

# Frühe informatische Bildung – Ziele und Gelingensbedingungen für den Elementar- und Primarbereich

Wissenschaftliche Untersuchungen zur Arbeit  
der Stiftung „Haus der kleinen Forscher“



# Frühe informatische Bildung – Ziele und Gelingensbedingungen für den Elementar- und Primarbereich

Stiftung Haus der kleinen Forscher:



PARTNER

Helmholtz-Gemeinschaft  
Siemens Stiftung  
Dietmar Hopp Stiftung  
Deutsche Telekom Stiftung

Wissenschaftliche Untersuchungen zur Arbeit der Stiftung  
„Haus der kleinen Forscher“

*Band 9*

Stiftung „Haus der kleinen Forscher“ (Hrsg.)

# **Frühe informatische Bildung – Ziele und Gelingensbedingungen für den Elementar- und Primarbereich**

Nadine Bergner, Hilde Köster, Johannes Magenheim, Kathrin  
Müller, Ralf Romeike, Ulrik Schroeder, Carsten Schulte

Mit einem Geleitwort von Peter Hubwieser und  
Johannes Magenheim

Verlag Barbara Budrich  
Opladen • Berlin • Toronto 2018

Herausgeber: Stiftung „Haus der kleinen Forscher“  
Verantwortlich: Dr. Janna Pahnke  
Projektleitung: Dr. Karen Bartling  
Konzeption und Redaktion: Dr. Claudia Peschke  
Redaktionelle Mitarbeit: Victoria Escobar Heredia, Christine Günther,  
Dr. Elena Harwardt-Heinecke, Mara Walgenbach

Weitere Informationen finden Sie unter: [www.haus-der-kleinen-forscher.de](http://www.haus-der-kleinen-forscher.de)

Haben Sie Fragen, Anmerkungen oder Anregungen zu diesem Band oder der wissenschaftlichen Begleitung der Stiftungsarbeit?  
Wenden Sie sich an: [forschung@haus-der-kleinen-forscher.de](mailto:forschung@haus-der-kleinen-forscher.de).  
Weitere Informationen und Studienergebnisse finden Sie auch unter [www.haus-der-kleinen-forscher.de](http://www.haus-der-kleinen-forscher.de), Rubrik Wissenschaftliche Begleitung.

Bibliografische Information der Deutschen Nationalbibliothek  
Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über <http://dnb.d-nb.de> abrufbar.

© 2018 Dieses Werk ist im Verlag Barbara Budrich erschienen und steht unter folgender Creative Commons Lizenz: <http://creativecommons.org/licenses/by-nc-nd/3.0/de/>  
Verbreitung, Speicherung und Vervielfältigung erlaubt, kommerzielle Nutzung und Veränderung nur mit Genehmigung des Verlags Barbara Budrich

Dieses Buch steht im Open-Access-Bereich der Verlagsseite zum kostenlosen Download bereit (<http://dx.doi.org/10.3224/84742107>).  
Eine kostenpflichtige Druckversion kann über den Verlag bezogen werden. Die Seitenzahlen in der Druck- und Onlineversion sind identisch.

**ISBN** 978-3-8474-2107-8  
**eISBN** 978-3-8474-1089-8  
**DOI** 10.3224/84742107

Umschlaggestaltung: Bettina Lehfeldt, Kleinmachnow – [www.lehfeldtgraphic.de](http://www.lehfeldtgraphic.de)  
Titelbildnachweis: Christoph Wehrer/Stiftung „Haus der kleinen Forscher“  
Lektorat und Satz: Ulrike Weingärtner, Gründau; [info@textakzente.de](mailto:info@textakzente.de)  
Druck: SDK Systemdruck, Köln  
Printed in Europe, gedruckt auf FSC-Papier

## Inhaltsverzeichnis

Informationen über die Autorinnen und Autoren .....	9
Vorwort .....	11
<b>Dialogisches Geleitwort</b> .....	13
<i>Peter Hubwieser, Johannes Magenheim</i>	
<b>Einleitung</b> .....	18
<b>1 Überblick zur Stiftung „Haus der kleinen Forscher“</b> .....	19
<b>2 Das „I“ in MINT – Relevanz der frühen informatischen Bildung</b> .....	28
<b>3 Fachliche Fundierung des Themenbereichs „Informatik“</b> .....	30
<b>Zusammenfassung zentraler Ergebnisse</b> .....	34
<b>Zieldimensionen informatischer Bildung im Elementar- und Primarbereich</b> .....	38
<i>Nadine Bergner, Hilde Köster, Johannes Magenheim, Kathrin Müller, Ralf Romeike, Ulrik Schroeder, Carsten Schulte</i>	
<b>1 Potenziale informatischer Bildung</b> .....	39
1.1 Was ist Informatik? .....	40
1.2 Informatik als Wissenschaft .....	41
1.3 Konstruktion in der Informatik .....	42
1.4 Gemeinsamkeiten und Unterschiede der Informatik im Vergleich ... ..	50
1.5 Informatik und informatische Bildung .....	56
1.6 Das Verhältnis von informatischer Bildung, Medienbildung und digitaler Bildung .....	68
1.7 Fazit: Informatische Bildung für alle .....	72
<b>2 Fundierung von Zieldimensionen auf Ebene der Kinder</b> .....	75
2.1 Kinder in digitalen Welten .....	75
2.2 Lernpsychologische Grundlagen .....	83
2.3 Zugänge zur Informatik für Kinder .....	85
2.4 Internationaler Vergleich: Curricula und deren Einordnung in das Kompetenzmodell. ....	101
2.5 Einordnung der internationalen Standards in den Ordnungsrahmen eines Kompetenzmodells für informatische Bildung in der Primarstufe .....	119
2.6 Ergebnis/Fazit .....	133

<b>3</b>	<b>Zieldimensionen auf Ebene der Kinder</b> .....	135
3.1	Übergreifende Basiskompetenzen. ....	136
3.2	Motivation, Interesse und Selbstwirksamkeit bezüglich Informatik ..	138
3.3	Informatische Kompetenzen der Kinder .....	141
3.4	Priorisierung konkreter Kompetenzerwartungen auf Ebene der Kinder. ....	152
<b>4</b>	<b>Zieldimensionen auf Ebene der pädagogischen Fach- und Lehrkräfte.</b> .....	166
4.1	Motivation, Interesse und Selbstwirksamkeit. ....	169
4.2	Einstellungen, Haltungen und Rollenverständnis .....	170
4.3	Informatische Fachkompetenz .....	174
4.4	Informatikdidaktische Kompetenzen. ....	192
4.5	Schlüsselkompetenzen zum Umgang mit digitalen Medien. ....	208
4.6	Fazit/Empfehlungen .....	210
<b>5</b>	<b>Beispiele priorisierter Kompetenzbereiche der frühen informatischen Bildung</b> .....	211
5.1	Beispiele früher informatischer Bildung .....	212
5.2	Zusammenfassende Heatmap der Priorisierung in den Beispielen. ...	239
<b>6</b>	<b>Gelingensbedingungen früher informatischer Bildung.</b> .....	241
6.1	Rahmenbedingungen für eine erfolgreiche Umsetzung. ....	241
6.2	Messinstrumente zur Evaluation der Gelingensbedingungen. ....	245
<b>7</b>	<b>Fazit</b> .....	250
<b>8</b>	<b>Anhang</b> .....	252
(1)	Information und Daten .....	252
	CAS (Computing at School; Großbritannien) .....	252
	CSTA (Computer Science Teachers Association; USA) .....	253
	Neuseeland .....	253
(12)	Algorithmen (und Programmierung) .....	254
	CAS. ....	254
	CSTA. ....	256
	Neuseeland .....	257
	Schweizer Lehrplan 21 .....	258
(13)	Sprachen und Automaten .....	258
	CAS. ....	258
	CSTA. ....	259
	Neuseeland .....	259
	Schweizer Lehrplan 21 .....	259

(14) Informatiksysteme .....	260
CAS .....	260
CSTA .....	261
Neuseeland .....	263
Schweizer Lehrplan 21 .....	263
(15) Informatik, Mensch und Gesellschaft .....	265
CAS .....	265
CSTA .....	265
Neuseeland .....	266
Schweizer Lehrplan 21 .....	267
<b>Fachempfehlung Informatiksysteme</b> .....	268
<i>Nadine Bergner, Kathrin Müller</i>	
<b>Einführung</b> .....	269
<b>1 Überblick über mögliche Informatiksysteme</b> .....	270
<b>2 Beschreibung und fachliche Einschätzung der einzelnen Informatiksysteme</b> .....	272
2.1 Roboter Cubetto der Firma Primo Toys .....	272
2.2 Bee-Bot der Firma Terrapin .....	274
2.3 KIBO von KinderLab Robotics .....	277
2.4 Ozobot/Ozobit von Evolve Inc. ....	279
2.5 LEGO WeDo 2.0 .....	283
2.6 Dash & Dot von Wonder Workshop .....	285
2.7 Scratch und ScratchJR .....	287
2.8 Makey Makey von JoyLabzLLC .....	290
2.9 LEGO Mindstorms (NXT & EV3) .....	292
2.10 Arduino-Mikrocontroller mit Ardublock .....	295
<b>3 Empfehlungen</b> .....	298
<b>4 Fazit</b> .....	301
<b>Fazit und Ausblick – Wie die Stiftung „Haus der kleinen Forscher“ mit den Erkenntnissen umgeht</b> .....	302
<b>1 Empfehlungen aus den Fachbeiträgen als Grundlage für die (Weiter-)Entwicklung der Stiftungsangebote</b> .....	303
1.1 Motivation, Interesse und Selbstwirksamkeit im Umgang mit Informatik .....	304
1.2 Informatische Prozessbereiche .....	308
1.3 Informatische Inhaltsbereiche .....	312



1.4 Informatikdidaktische Kompetenzen .....	315
1.5 Einstellungen, Haltungen und Rollenverständnis in Bezug auf die Gestaltung informatischer Bildung .....	317
<b>2 Ausblick und weitere wissenschaftliche Begleitung .....</b>	<b>320</b>
<b>Literatur .....</b>	<b>324</b>
<b>Anhang .....</b>	<b>342</b>
<b>Bildquellenverzeichnis .....</b>	<b>345</b>
<b>Über die Stiftung „Haus der kleinen Forscher“ .....</b>	<b>346</b>
<b>Bisher erschienen in der Wissenschaftlichen Schriftenreihe der Stiftung „Haus der kleinen Forscher“ .....</b>	<b>348</b>

## Informationen über die Autorinnen und Autoren

### **Dr. Nadine Bergner**

RWTH Aachen, Lehr- und Forschungsgebiet Informatik 9 (Lerntechnologien und Fachdidaktik Informatik), InfoSphere – Schülerlabor Informatik

*Arbeitsschwerpunkte:* Fachdidaktik Informatik, außerschulisches Lernen, Schülerlabor, Lehramtsaus- und -fortbildung, digitale Bildung, Informatik an Grundschulen, Lernspiele, E-Learning

*Kontakt:* Ahornstraße 55, 52074 Aachen, bergner@informatik.rwth-aachen.de

### **Prof. Dr. Peter Hubwieser**

TUM School of Education, Fachgebiet Didaktik der Informatik

*Arbeitsschwerpunkte:* Lehr-/Lernforschung zur Informatik in der Schule, Informatik-Lehrerausbildung

*Kontakt:* Marsstraße 20, 80335 München, Peter.Hubwieser@tum.de

### **Prof. Dr. Hilde Köster**

Freie Universität Berlin, Fachbereich Erziehungswissenschaft und Psychologie

*Arbeitsschwerpunkte:* Grundschulpädagogik, Sachunterricht, Bildungsprozesse und diagnosebasierte Begabungsförderung bei Kindern in den Bereichen Naturwissenschaften, Technik und Informatik, Professionalisierung angehender Grundschullehrkräfte

*Kontakt:* Habelschwerdter Allee 45, 14195 Berlin, hilde.koester@fu-berlin.de

### **Prof. Dr. Johannes Magenheim**

Universität Paderborn, Fakultät für Elektrotechnik, Informatik und Mathematik

*Arbeitsschwerpunkte:* Didaktik der Informatik, Informatik und Bildung, E-Learning

*Kontakt:* Fürstenallee 11, 33102 Paderborn, jsm@uni-paderborn.de

### **Kathrin Müller**

Universität Paderborn, Fakultät für Elektrotechnik, Informatik und Mathematik

*Arbeitsschwerpunkte:* Informatik in der Primarstufe und der Sekundarstufe I, Schülervorstellungen über Roboter und deren Funktionsweise, Lehrerausbildung, Lehrerfortbildung Informatik in der Primarstufe

*Kontakt:* Fürstenallee 11, 33102 Paderborn, kathrin.mueller@upb.de

**Prof. Dr. Ralf Romeike**

Friedrich-Alexander-Universität Erlangen-Nürnberg, Technische Fakultät, Department Informatik

*Arbeitsschwerpunkte:* Informatik in der frühen Bildung, agile Methoden für den Informatikunterricht, Schlüsselkonzepte der Informatik, Didaktik des Datenmanagements, Physical Computing, Didaktik visueller blockbasierter Programmierung

*Kontakt:* Martensstraße 3, 91058 Erlangen, ralf.romeike@fau.de

**Prof. Dr. Ulrik Schroeder**

RWTH Aachen, Lehr- und Forschungsgebiet Informatik 9 (Lerntechnologien und Fachdidaktik Informatik)

*Arbeitsschwerpunkte:* Theorien, Methoden und Werkzeuge für Lerntechnologien, Learning Analytics, Assessment und Feedback, spielbasiertes Lernen, mobiles Lernen, Fachdidaktik Informatik, außerschulische Lernorte, Entwicklung und Erforschung von Lernmaterialien für den Informatikunterricht und für außerschulische Lernorte, Informatik an Grundschulen

*Kontakt:* Ahornstraße 55, 52074 Aachen, schroeder@informatik.rwth-aachen.de

**Prof. Dr. Carsten Schulte**

Universität Paderborn, Fakultät für Elektrotechnik, Informatik und Mathematik

*Arbeitsschwerpunkte:* Didaktik der Informatik, Konzepte informatischer Bildung für die digitale Welt, Erarbeiten, Verstehen und Nutzen digitaler Geräte und Infrastrukturen, Grundschülervorstellungen über Informatiksysteme und wie diese durch die Beschäftigung mit der algorithmischen Arbeitsweise der Geräte geändert werden können

*Kontakt:* Fürstenallee 11, 33102 Paderborn, carsten.schulte@uni-paderborn.de

## Vorwort

### Liebe Leserinnen und Leser,

ob es nun das Ampelsystem ist, das Navigationsgerät im Auto, das Smartphone der Eltern, ein Tablet in der Kita oder der PC im Klassenzimmer – Kinder wachsen heute in einer Welt auf, die stark digital geprägt ist und sich rasend schnell weiterentwickelt. Mädchen und Jungen möchten sie erforschen und mitgestalten und haben viele Fragen: Wie funktioniert ein Roboter? Was passiert, wenn ich den Computer anschalte? Und was steckt eigentlich hinter all den Bildern und Informationen?

Seit vielen Jahren entwickelt und evaluiert die Stiftung „Haus der kleinen Forscher“ erfolgreich Konzepte und Materialien im MINT-Bereich für das forschende Lernen in der frühkindlichen Bildung. Nach den Naturwissenschaften, der Technik und der Mathematik wagt sich die Stiftung mit dem neunten Band der wissenschaftlichen Schriftenreihe zur Arbeit der Stiftung an das „I“ in MINT heran.

Damit haben wir Neuland betreten, denn im deutschsprachigen Raum gibt es kaum Forschungsansätze und Bildungskonzepte zu diesem Thema. Dabei wächst die Bedeutung von Informatik und damit verbundenen Kompetenzen stetig.

Bei den Kindern steht die Faszination der Welt im Vordergrund: Sie wollen ausprobieren, erforschen, mitgestalten und wissen, wie was zusammenhängt und funktioniert. Dabei gehen sie mit Computern und Co. nicht anders um als mit anderen spannenden Dingen: Sie wollen sie entdecken, erforschen und wissen, was dahintersteckt. Wie kommen alle diese Informationen in das kleine Handy? Warum kann mein Computer so schnell rechnen? Woher weiß die Ampel, wann wer fahren, wer gehen und wer stehen soll?

Unsere Aufgabe als Frühbildungsinitiative ist es, die Pädagoginnen und Pädagogen zu befähigen, gemeinsam mit den Mädchen und Jungen auf die Suche nach Antworten gehen zu können.

Daher stellt das „Haus der kleinen Forscher“ – beruhend auf den Empfehlungen der Expertinnen und Experten aus Wissenschaft und Forschung – in seinem Bildungsangebot die technologische Perspektive des Digitalen in den Mittelpunkt. Es gibt gut ausgearbeitete Fortbildungskonzepte, pädagogische Materialien und Online-Angebote für pädagogische Fach- und Lehrkräfte sowie für Kinder zum Entdecken von Informatik – mit und ohne Computer.

Denn es geht dabei nicht um die verstärkte Nutzung digitaler Medien, sondern um das Verstehen zugrunde liegender Konzepte. Mädchen und Jungen können in einem spielerischen Umgang mit Informatiksystemen algorithmisches Denken



kennenlernen und bauen damit eine wichtige Grundlage für einen reflektierten und kompetenten Umgang mit diesen Systemen auf.

Mit dem „I“ in MINT wurde Pionierarbeit geleistet – ich bedanke mich ganz herzlich bei den Autorinnen und Autoren dieses Bandes, dass sie uns auf diesem spannenden Terrain so gut unterstützt und geleitet haben. Ein verantwortungsvoller, aufgeschlossener und kreativer Umgang mit informatischer Bildung ist wichtig für unsere Kinder und die Gesellschaft von morgen.

Ich bin davon überzeugt, dass wir mit diesem Band und unserem Bildungsangebot einen Beitrag dazu leisten, dass das „I“ auch in den Bildungseinrichtungen bundesweit bald kein Neuland mehr ist, sondern ein wertvoller Teil einer guten und gelingenden frühen Bildung.

Michael Fritz

Vorstandsvorsitzender der Stiftung „Haus der kleinen Forscher“

## Dialogisches Geleitwort

*Peter Hubwieser, Johannes Magenheim*

Mit der Entwicklung von informatischen Bildungskonzepten für drei- bis sechsjährige Kinder und einem entsprechenden auf die Praxis frühkindlicher Erziehung in Kita, Hort und Grundschule bezogenen Fortbildungskonzept für die pädagogischen Fach- und Lehrkräfte betritt die Stiftung „Haus der kleinen Forscher“ weitgehend neues Terrain, da es zu diesem Bereich, zumindest in Deutschland, noch wenig Erfahrungen und nur rudimentäre Forschungsansätze gibt. Dem Einen oder Anderen mag es daher als gewagte Unternehmung erscheinen, informatische Bildung für Kinder im Kita- und Grundschulalter anzubieten. Geht das überhaupt? Falls ja, macht es auch Sinn?

Um diese Fragen bewusst zu reflektieren und mögliche Einwände aufzunehmen und Antworten zu suchen, diskutieren an dieser Stelle die beiden Experten Prof. Dr. Peter Hubwieser (Technische Universität München, Fachgebiet Didaktik der Informatik) und Prof. Dr. Johannes Magenheim (Universität Paderborn, Fakultät für Elektrotechnik, Informatik und Mathematik) mit einem fiktiven Fragesteller.

**P. Hubwieser und J. Magenheim (P. H. & J. M.):** Diese Expertise zur frühen informatischen Bildung hat es sich zur Aufgabe gemacht, einen Überblick zum Status quo der für diese Fragestellung relevanten Themenbereiche zu geben. Hierzu zählen u. a. Fragen nach der gesellschaftlichen Bedeutung von Informatiksystemen und dem Verhältnis von Informatik, informatischer Bildung und Medienbildung. Ferner wird versucht zu klären, welche frühen Erfahrungen Kinder teilweise schon mit digitalen Welten haben, welche kognitiven Voraussetzungen für die Verarbeitung derartiger Erfahrungen erforderlich sind, welche Spiel- und Lernmaterialien hierfür bereits zur Verfügung stehen und welche curricularen Konzepte es hierfür schon gibt. Dabei werden nicht nur nationale, sondern auch internationale Ansätze berücksichtigt. Auf der Basis der so gewonnenen Erkenntnisse werden unter Berücksichtigung einschlägiger internationaler Curricula Zieldimensionen, Gelingensbedingungen und Gestaltungsempfehlungen zur frühen informatischen Bildung auf der Ebene der Kinder und für pädagogische Fach- und Lehrkräfte formuliert.

**Fragesteller (FS):** Die Abstraktionsfähigkeit der Kinder und ihre kulturellen Grundtechniken in diesen Altersstufen sind möglicherweise noch nicht ausreichend, um

informatische Kompetenzen zu erlangen. Ernsthaftes Programmieren erfordert zumindest Grundfertigkeiten im Lesen und Schreiben, man denke nur an die Befehle einer Programmiersprache. Auch blockorientierte Sprachen wie bestimmte Versionen von Scratch kommen doch nicht ganz ohne textuelle Elemente aus. Für Wiederholungen benötigt man an der einen oder anderen Stelle auch noch grundlegende rechnerische Fähigkeiten.

**P. H. & J. M.:** Wir haben in der Expertise auf einige Studien verwiesen, die nahelegen, dass Kinder schon ab einem Alter von 3 bis 5 Jahren grundlegende informatische Konzepte begreifen und einfache Informatiksysteme spielerisch mit positiven Lerneffekten erkunden können. Vor allem im renommierten Media Lab des MIT (Massachusetts Institute of Technology) wurden auf diesem Gebiet jahrelang vielversprechende Erfahrungen gesammelt. Es geht nicht darum, den Kindern Programmierkenntnisse zu vermitteln oder gar eine Programmiersprache zu ‚erlernen‘. Vielmehr sollen algorithmisches Denken und der spielerische Umgang mit Informatiksystemen als wichtige kognitive Grundlage für den späteren reflektierten und kompetenten Umgang mit derartigen Systemen eingeübt werden. Hierzu werden kindgerechte und altersgemäße Lernszenarien vorgeschlagen, in deren Mittelpunkt vor allem das Erlernen informatischer Konzepte ohne Computer nach dem ‚Computer-Science-Unplugged‘-Ansatz (CS Unplugged) und das Identifizieren von ‚informatikhaltigen‘ Alltagssituationen aus der Erfahrungswelt der Kinder stehen. Als eine Option zur Vertiefung derartiger Lernprozesse können ggf. auch einfache, altersgemäße Informatiksysteme (z. B. einfache Spielroboter) erkundet und mit elementaren Befehlen gesteuert werden. Traditionelle digitale Medien wie Smartphones, Tablets und PCs spielen bei diesem didaktischen Konzept so gut wie keine Rolle.

**FS:** Welche informatischen Kompetenzen könnten die Kinder dieser Altersstufen denn ohne ernsthafte Programmierfähigkeit tatsächlich erlernen? Welche Änderungen ihres Verhaltens wären realistisch? Möglicherweise beschränkt sich der Lernprozess dann nur auf „Einsicht“ statt auf handfeste Kompetenzen und damit letztlich auf „totes Wissen“.

**P. H. & J. M.:** In der Expertise werden Zieldimensionen der frühen informatischen Bildung für Kinder beschrieben, die sich auf die oben erwähnten altersgemäßen Lernszenarien beziehen. Diese Zieldimensionen orientieren sich an Kompetenzbeschreibungen aus einschlägigen internationalen Curricula und nationalen Bildungsstandards zur Informatik in der Grundschule. Daher werden hier keine umfassenden informatischen Kompetenzerwartungen als potenzielle Ergebnisse früher informatischer Bildung formuliert, sondern es geht eher um die frühe

kognitive Präfiguration von informatischen Begriffen und Prozessen. In diesem Sinne werden in Orientierung am traditionellen Kompetenzbegriff neben altersgemäßen kognitiven informatikbezogenen Kompetenzen auch die Förderung von übergreifenden Basiskompetenzen (z. B. sprachliche und soziale Kompetenzen, Motivation, Interesse und Selbstwirksamkeit bezüglich Informatik) angestrebt. Es wird dabei berücksichtigt, dass der Kompetenzerwerb jeweils inhaltliche und prozessuale Aspekte vereint und sich in einem konkreten Handlungskontext manifestiert. Wenn Kinder z. B. eine Ampelanlage in ihrer grundlegenden Funktion verstehen und diese in vereinfachter Weise als gestaltbares, soziotechnisches Informatiksystem begreifen, das soziales Handeln reguliert, so kann damit eine wichtige Grundlage für das spätere tiefgehende Verständnis derartiger Systeme gelegt werden.

**FS:** Selbst wenn das so funktioniert, dann fehlt vielleicht die für Informatikbildung aufgewendete Zeit für andere Kompetenzentwicklungen oder Spieltätigkeiten wie Rechnen Schreiben, Lesen, Basteln, Raumorientierung, Natur erleben, soziale Kontakte etc. In späteren Jahren könnten die in diesem Alter mit viel Zeitaufwand erlernten informatischen Konzepte möglicherweise in wesentlich kürzerer Zeit nachgelernt werden. Zudem sollte der Kontakt mit Informatiksystemen in diesem Altersbereich meiner Meinung nach möglichst kurz gehalten werden. Die Kinder werden sich später noch lange genug überwiegend mit ihrem Smartphone beschäftigen. Damit werden längere Übungen am Gerät doch sehr problematisch.

**P. H. & J. M.:** Gerade in der Kita und im Grundschulunterricht geht es immer auch um den Erwerb von Primärerfahrungen mit der umgebenden Welt. Explizites Lernen kann nur auf Erfahrungen aufsatteln. Vertiefte und nachhaltige Interessen an einem Gegenstand werden oft gerade in der (frühen) Kindheit ausprägt. Dabei kommt es weniger auf das explizite Lernen als auf positive, spielerische Erfahrungen an. Die in der Expertise formulierten Zieldimensionen für Kinder sehen neben dem elementaren kognitiven, informatikbezogenen Kompetenzerwerb auch die spielerische Vermittlung von übergreifenden Basiskompetenzen, wie Sprachförderung, Kommunikationsfähigkeit, Umgang mit ikonografischen formalen Darstellungen als einer Form von Verschriftlichung von Information vor. Lernprozesse in der frühen informatischen Bildung, insbesondere in der hier vorgeschlagenen handlungsorientierten Form von CS Unplugged, dienen nicht nur der Grundlegung informatikbezogenen Wissens und erschließen den Kindern somit den Umgang mit digitalen Artefakten als ‚vierter Kulturtechnik‘, sondern ermöglichen als motivierende methodische Variation den kindgerechten Zugang zu den anderen grundlegenden Bereichen menschlicher Kommunikation und menschlichen Han-



delns. Sie können damit integrativer Bestandteil eines Gesamtkonzepts frühkindlicher Bildung sein.

**FS:** Es gibt bisher doch kaum belastbare Forschungsergebnisse über Lernprozesse zur Informatik in der frühkindlichen Bildung. Selbst informatische Bildung in der Primarstufe ist noch weitgehend unerforscht. Soll man die begrenzten Ressourcen nicht lieber auf Forschung und Entwicklung zur Informatikausbildung in sinnvolleren und dringenderen (höheren) Altersstufen konzentrieren? Zunächst müsste doch in ganz Deutschland ein entsprechendes Schulfach in der Mittelstufe eingeführt werden. Danach sollte man erforschen, welche Informatikbildung an Primarschulen sinnvoll ist und diese implementieren. Erst dann macht es Sinn, Ressourcen in die frühkindliche Bildung zu stecken.

**P. H. & J. M.:** Es ist in der Tat ausgesprochen wichtig, Lernprozesse in der frühen informatischen Bildung zu erforschen, da zu diesem Bereich bisher nur kleinere, rudimentäre Studien mit begrenzter Aussagekraft vorliegen. Im Gegensatz zu den anderen Schulfächern, insbesondere im naturwissenschaftlichen Bereich, weist die Informatik noch immer ein großes Defizit an empirisch fundierter Unterrichtsforschung auf. Dieses auszugleichen, ist eine wichtige Voraussetzung, um das Schulfach Informatik auf allen Schulstufen, und damit die informatische Bildung, nachhaltig zu etablieren. Allerdings hat in den letzten Jahren die Zahl der Informatikdidaktik-Professuren national wie auch international zugenommen, so dass hier neue Forschungskapazitäten entstanden sind. Die empirisch fundierte Erforschung von Lernprozessen in der informatischen Bildung sollte daher wie in anderen Fächern des naturwissenschaftlichen Bereichs auch alle Altersstufen umfassen und auch zu klären versuchen, welche frühen Erfahrungen Kinder mit Informatiksystemen machen und wie sich diese Erfahrungen auf ihre spätere Motivation und Einstellungen zur Informatik und zum Umgang mit digitalen Artefakten auswirken. Somit können die frühe informatische Bildung und deren Erforschung einen wichtigen Baustein für späteres, gelingendes schulisches und außerschulisches informatikbezogenes Lernen darstellen. Auf jeden Fall wollen wir versuchen, aus den ersten Umsetzungen unserer Konzepte belastbare Forschungsergebnisse zu gewinnen.

**FS:** Möglicherweise werden mit solchen Bemühungen zur frühkindlichen Bildung aber auch nur die übertriebenen Bemühungen von sog. „Helikopter-Eltern“ unterstützt. Falls sich die Lernprozesse aufgrund der geringen Abstraktionsfähigkeit der Kinder zu schwierig gestalten, könnte die Unterstützung der Eltern entscheidend sein. Dann könnten solche Angebote die ohnehin starke soziale Spreizung in der Bildung auch in der Informatik weiter verstärken.

**P. H. & J. M.:** Für die frühe informatische Bildung ist es wichtig, auch die Eltern und andere Entscheidungsträger mit einzubinden. Dies sehen auch die in der Expertise formulierten Zieldimensionen für die pädagogischen Fach- und Lehrkräfte vor. Hier geht es aber primär um den bewussten und reflektierten Umgang mit digitalen Medien im familialen Bereich der Kinder, der die CS-Unplugged-Methoden in den Einrichtungen der Elementarerziehung sinnvoll ergänzen und den ungezügelter Gebrauch elektronischer Medien in den Kinderzimmern eindämmen soll. Somit spielen die Verfügbarkeit derartiger Geräte und Medien im häuslichen Umfeld keine Rolle. Es wird von den Eltern auch nicht erwartet, dass sie ihre Kinder bei anfallenden Hausaufgaben unterstützen, da es diese im Kindergarten im Vergleich zum schulischen Lernen nicht gibt. Somit trägt die frühe informatische Bildung nicht zu einer Verstärkung einer ohnehin vorhandenen digitalen Spaltung der Gesellschaft bei, sondern kann im Gegenteil einen sinnvollen altersgemäßen Umgang mit digitalen Medien im Elternhaus unterstützen.

# Einleitung

Stiftung Haus der kleinen Forscher



- 1 Überblick zur Stiftung „Haus der kleinen Forscher“
- 2 Das „I“ in MINT – Relevanz der frühen informatischen Bildung
- 3 Fachliche Fundierung des Themenbereichs „Informatik“

## 1 Überblick zur Stiftung „Haus der kleinen Forscher“

Die gemeinnützige Stiftung „Haus der kleinen Forscher“ engagiert sich für gute frühe Bildung in den Bereichen Mathematik, Informatik, Naturwissenschaften und Technik (MINT) – mit dem Ziel, Mädchen und Jungen stark für die Zukunft zu machen und zu nachhaltigem Handeln zu befähigen. Gemeinsam mit ihren Netzwerkpartnern vor Ort bietet die Stiftung bundesweit ein Bildungsprogramm an, das pädagogische Fach- und Lehrkräfte dabei unterstützt, Kinder im Kita- und Grundschulalter qualifiziert beim Entdecken, Forschen und Lernen zu begleiten. Das „Haus der kleinen Forscher“ verbessert Bildungschancen, fördert Interesse am MINT-Bereich und professionalisiert dafür pädagogisches Personal. Die Bildungsinitiative leistet damit einen wichtigen Beitrag in folgenden Bereichen:

- zur Qualifizierung des frühpädagogischen Personals
- zur Qualitätsentwicklung von Einrichtungen
- zur Persönlichkeits- und Interessenentwicklung der Kinder
- zur Nachwuchsförderung in den MINT-Bildungsbereichen

Die Hauptaktivitäten der Stiftung sind:

- der Auf- und Ausbau tragfähiger lokaler Netzwerke unter Beteiligung von Akteuren vor Ort sowie Beratung und Service für die inzwischen rund 222 Netzwerkpartner,
- die Ausbildung von Multiplikatorinnen und Multiplikatoren (Trainerinnen und Trainern), die vor Ort pädagogische Fach- und Lehrkräfte kontinuierlich fortbilden,
- die Entwicklung und Bereitstellung von Fortbildungskonzepten und Materialien für pädagogische Fach- und Lehrkräfte sowie
- die Unterstützung der Qualitätsentwicklung von Bildungseinrichtungen durch die Zertifizierung als „Haus der kleinen Forscher“.

### Qualifizierungsinitiative für Pädagoginnen und Pädagogen

Das „Haus der kleinen Forscher“ ist bundesweit die größte Qualifizierungsinitiative für Pädagoginnen und Pädagogen im Bereich der frühen Bildung. Sie un-

terstützt Kitas, Horte und Grundschulen dabei, mathematische, informatische, naturwissenschaftliche und/oder technische Schwerpunkte zu setzen und förderliche Entwicklungs- und Lernumgebungen für Kinder in diesen Bereichen zu bieten. Der pädagogische Ansatz der Stiftung knüpft an den Ressourcen der Kinder an und betont das gemeinsame entdeckende und forschende Lernen im dialogischen Austausch (Stiftung Haus der kleinen Forscher, 2015). Die Stiftung fördert mit ihren Aktivitäten auch die Umsetzung vorhandener Bildungs- und Rahmenlehrpläne der jeweiligen Bundesländer in den Bereichen Mathematik, Informatik, Naturwissenschaften und Technik.

Die inhaltlichen Angebote der Stiftung umfassen neben den Fortbildungen für pädagogische Fach- und Lehrkräfte auch pädagogische Materialien, einen jährlichen Aktionstag sowie Anregungen für Kooperationen:

- **Pädagogische Materialien:** Für die praktische Umsetzung in den pädagogischen Einrichtungen stellt die Stiftung in den Fortbildungen kostenlos Materialien zur Verfügung, z. B. Themenbroschüren, Forschungs- und Entdeckungskarten, didaktische Materialien und Filmbeispiele.
- **Internetpräsenz:** Die Website [www.haus-der-kleinen-forscher.de](http://www.haus-der-kleinen-forscher.de) bietet Informationen für alle Interessierten.
- **Magazin „Forscht mit!“:** Pädagogische Fach- und Lehrkräfte erhalten quartalsweise praktische Tipps zum Entdecken und Forschen in der Einrichtung, Informationen zur Arbeit der Stiftung sowie Best-Practice-Berichte aus anderen Einrichtungen und Netzwerken.
- **„Tag der kleinen Forscher“:** An diesem bundesweiten Mitmachtag können Mädchen und Jungen in ganz Deutschland ein aktuelles Forschungsthema erkunden. Dazu stellt die Stiftung den pädagogischen Einrichtungen Material bereit und ruft Unterstützer und Unterstützerinnen aus Politik, Wirtschaft, Wissenschaft und Gesellschaft zum Mitmachen auf.
- **Anregungen zur Kooperation:** Interessierte Eltern, Patinnen und Paten sowie andere Bildungspartner unterstützen das gemeinsame Entdecken und Forschen in den Einrichtungen.
- **Zertifizierung:** Engagierte Einrichtungen werden anhand festgelegter Bewertungskriterien als „Haus der kleinen Forscher“ zertifiziert. Alle sich bewerbenden Einrichtungen erhalten eine detaillierte Rückmeldung mit Anregungen für die weitere Entwicklung des gemeinsamen Entdeckens und Forschens mit den Kindern.

- **Kinder-Website:** Unter [www.meine-forscherwelt.de](http://www.meine-forscherwelt.de) gelangen Kinder im Grundschulalter in einen interaktiven Forschergarten, der sie zu eigenständigen Entdeckungsreisen animiert. Für pädagogische Fach- und Lehrkräfte stehen Tipps für die Lernbegleitung zur Verfügung.
- **Service-Portal Integration:** Unter [www.integration.haus-der-kleinen-forscher.de](http://www.integration.haus-der-kleinen-forscher.de) erhalten Fach- und Lehrkräfte anhand vielfältiger Materialien, praxisnaher Impulse sowie durch inspirierenden Erfahrungsaustausch Unterstützung bei der Integration geflüchteter Kinder in Kita, Hort und Grundschule.

## Bundesweite Vernetzung

Das „Haus der kleinen Forscher“ lebt als bundesweite Bildungsinitiative vom Engagement vielfältiger Akteure vor Ort – den lokalen Netzwerken, die als dauerhafte Partner und Fortbildungsanbieter in den Regionen agieren. Zu den derzeit (Stand 30. Juni 2018) 216 Netzwerkpartnern zählen Kommunen und Kita-Träger, Wirtschaftsverbände, Science-Center, Museen, Unternehmen, Stiftungen, Vereine usw. Seit 2011 steht das Fortbildungsprogramm der Initiative auch Horten und Ganztagsgrundschulen offen.

Pädagogische Fach- und Lehrkräfte aus über 30.600 Kitas, Horten und Grundschulen haben bereits am Fortbildungsprogramm der Initiative teilgenommen, davon pädagogische Fachkräfte aus rund 24.800 Kitas sowie Fach- und Lehrkräfte aus über 1.400 Horten und über 4.400 (Ganztags-)Grundschulen.

Deutschlandweit sind über 5.000 Kitas, Horte und Grundschulen als „Haus der kleinen Forscher“ zertifiziert, darunter über 4.600 Kitas. Seit Herbst 2013 können sich auch Horte und Grundschulen zertifizieren lassen. Rund 200 Horte und über 200 Grundschulen haben bereits das Zertifikat „Haus der kleinen Forscher“.

## Das kontinuierliche Fortbildungsprogramm

Die Stiftung „Haus der kleinen Forscher“ konzentriert sich vor allem auf die Weiterqualifizierung von Pädagoginnen und Pädagogen im Hinblick auf das Entdecken und Erforschen mathematischer, informatischer, naturwissenschaftlicher und/oder technischer Themen mit Kindern. Das Ziel ist eine kontinuierliche Begleitung der pädagogischen Fach- und Lehrkräfte: Die Teilnahme an Fortbildungen zu unterschiedlichen Themen erweitert sukzessive das methodische Repertoire und vertieft das Verständnis des pädagogischen Ansatzes der Stiftung. Im Wechsel von Präsenzfortbildung und Transferphasen können die Pädagoginnen und Pädagogen das Gelernte in der Praxis ausprobieren und sich dazu in der nächsten Fortbildung austauschen.

Um möglichst vielen interessierten pädagogischen Fach- und Lehrkräften die Teilnahme an Fortbildungen zu ermöglichen, findet die Weiterqualifizierung über ein Multiplikatorenmodell statt: Die Stiftung „Haus der kleinen Forscher“ bildet an mehreren Standorten in Deutschland Trainerinnen und Trainer aus, die ihrerseits Fortbildungen für Pädagoginnen und Pädagogen in ihrem lokalen Netzwerk durchführen. Die Trainerinnen und Trainer qualifizieren sich durch die Teilnahme an den Präsenz- und Online-Fortbildungen der Stiftung dafür, Fortbildungen mit Pädagoginnen und Pädagogen durchzuführen. Als Unterstützung erhalten sie ausführliche Arbeitsunterlagen für ihre Aufgabe in der Erwachsenenbildung sowie die Möglichkeit, persönliches Feedback im Hospitationsprogramm der Stiftung oder in Form von Videocoaching zu bekommen. Für die Auffrischung und Vertiefung der Fortbildungsinhalte steht auch der Online-Campus für Trainerinnen und Trainer zur Verfügung. Die digitale Lernplattform hält neben einer Vielzahl von Online-Lernangeboten auch inhaltliche Informationen und Arbeitsunterlagen zu den einzelnen Fortbildungsmodulen bereit. Zu bestimmten Themen gibt es die Möglichkeit, eigenständig offene E-Learning-Module zu bearbeiten, an tutoriell begleiteten Kursen teilzunehmen sowie die Online-Begleitkurse zu Präsenzfortbildungen zu nutzen. Darüber hinaus können die Trainerinnen und Trainer in Themenforen oder offenen Chats miteinander in Kontakt treten und sich austauschen.

Sowohl für die pädagogischen Fach- und Lehrkräfte als auch für die Trainerinnen und Trainer werden in der Bildungsinitiative jedes Jahr unterschiedliche Fortbildungsthemen angeboten. Bis Ende 2016 besuchten neue Trainerinnen und Trainer bzw. erstmals teilnehmende Pädagoginnen und Pädagogen zunächst die Fortbildungen „Forschen mit Wasser“ (Workshop 1) und „Forschen mit Luft“ (Workshop 2), in denen der pädagogische Ansatz der Stiftung für das gemeinsame Entdecken und Forschen mit Kindern ausführlich thematisiert wird. Seit 2017 ist der Einstieg in das Bildungsangebot der Stiftung flexibel gestaltet. Sieht die Lernbegleitung noch Entwicklungsbedarf in ihrer pädagogischen Kompetenz bzw. möchte sich einen Überblick zum pädagogischen Konzept der Stiftung verschaffen, so erhält sie das Angebot, wie bisher mit den genannten Präsenzfortbildungen einzusteigen bzw. das Seminar oder den Online-Kurs „Grundlagen zur Pädagogik“ zu besuchen. Ebenso können die pädagogischen Fach- und Lehrkräfte bzw. die Trainerinnen und Trainer als Einstieg aber auch ein anderes Modul zu den mathematischen, informatischen, naturwissenschaftlichen oder technischen Themen wählen. Die Inhalte werden in verschiedenen Formaten angeboten: Fortbildungen vor Ort, Selbstbildungsformate (wie Online-Kurse oder gedruckte pädagogische Materialien) und Bildungsveranstaltungen. Das Zertifikat „Haus der kleinen Forscher“ unterstützt darüber hinaus bei der Qualitätsentwicklung der pädagogischen Arbeit in den Einrichtungen und macht das Engagement für gute frühe MINT-Bildung nach außen sichtbar. Die Stiftung orientiert sich somit stärker

an den Bedarfen ihrer Zielgruppen in Abhängigkeit von deren Vorwissen und Vorerfahrungen sowie Interessen und zeitlichen Flexibilität.

Inhaltlich wurde das Stiftungsangebot zum Start des Schuljahres 2017/18 mit dem Workshop „Informatik entdecken – mit und ohne Computer“<sup>1</sup> um den Bildungsbereich der informatischen Bildung ergänzt (vgl. Kapitel „Fazit und Ausblick“). Die Stiftung erweitert ihr Angebot um Fortbildungen, Inhalte und Materialien zur Bildung für nachhaltige Entwicklung. Das Konzept, das sich an pädagogische Fach- und Lehrkräfte wie auch erstmals an Kita-Leitungen wendet, wird seit 2017 in 29 Modellnetzwerken in der Praxis erprobt. Ab 2018 werden die Fortbildungen zur Bildung für nachhaltige Entwicklung überall dort zur Verfügung stehen, wo Netzwerkpartner diesen Bildungsschwerpunkt in ihr Angebot aufnehmen. Darüber hinaus wird ab Herbst 2018 das Thema „Technik – von hier nach da“ aus dem Bereich der technischen Bildung angeboten.

## Wissenschaftliche Begleitung und Qualitätsentwicklung

Alle Aktivitäten der Bildungsinitiative werden kontinuierlich wissenschaftlich begleitet und evaluiert. Die Stiftung „Haus der kleinen Forscher“ pflegt einen offenen Austausch mit Wissenschaft und Fachpraxis und versteht sich als lernende Organisation.

Ein umfangreiches Spektrum an Maßnahmen dient der Sicherung und Weiterentwicklung der Qualität im „Haus der kleinen Forscher“ (siehe Abbildung 1). Das stiftungseigene Qualitätsmanagement erfasst mit internen Evaluationsmaßnahmen und umfassendem Monitoring alle wichtigen Aktivitäten und Angebote. Dafür nutzt die Stiftung eine ganze Reihe an Datenquellen (wie z. B. anlassbezogene Befragungen der Zielgruppen<sup>2</sup>); eine Kombination aus Daten im Quer- und Längsschnittformat ermöglicht den Blick sowohl auf die aktuelle Situation als auch auf wichtige Veränderungen in den letzten Jahren. Um auf die Erkenntnisbedarfe der Stiftung flexibler reagieren zu können, wird die bisher jährliche Befragung sämtlicher Zielgruppen durch mehrere Erhebungen zu unterschiedlichen Zeitpunkten ersetzt. Die längsschnittliche Perspektive spielt in den internen Evaluations- und Monitoring-Maßnahmen der Stiftung eine zunehmend wichtigere Rolle, um dem Anspruch einer stärkeren Wirkungsorientierung gerecht zu werden. Mit dem regelmäßig erscheinenden Monitoring-Bericht stellt die Stiftung wichti-

---

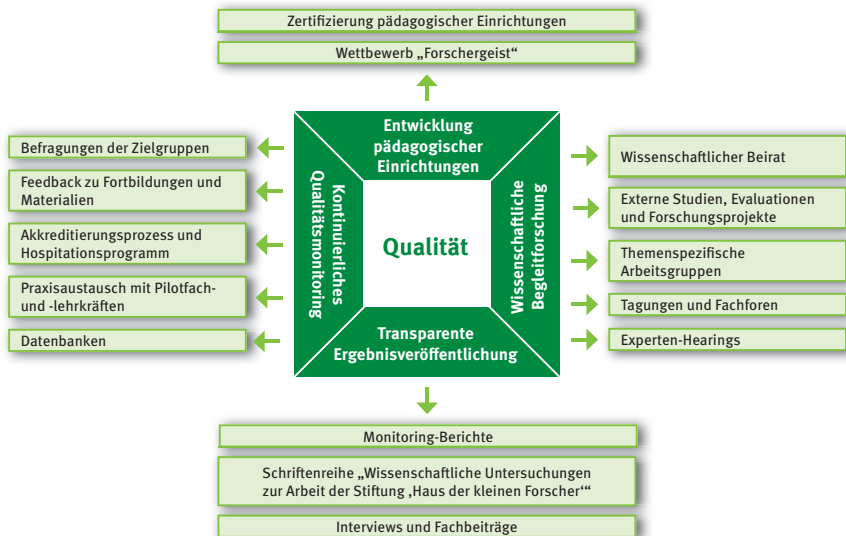
1 Der Workshop wurde auf Grundlage der vorliegenden Expertise „Zieldimensionen informatischer Bildung im Elementar- und Primarbereich“ entwickelt. Weitere Informationen zur inhaltlichen Ausgestaltung finden Sie im Kapitel „Fazit und Ausblick – Wie die Stiftung ‚Haus der kleinen Forscher‘ mit den Erkenntnissen umgeht“.

2 Netzwerkkoordinatorinnen und -koordinatoren, Trainerinnen und Trainer sowie pädagogische Fach- und Lehrkräfte.



ge Ergebnisse dieser Maßnahmen bereit. So beschreibt der Monitoring-Bericht 2016/2017 auf Grundlage einer Wirkungskette, wie das Fortbildungsangebot der Initiative zur Verbesserung der frühen MINT-Bildung in Deutschland beiträgt (vgl. Stiftung Haus der kleinen Forscher, 2017a).

Im Rahmen der inhaltlichen (Weiter-)Entwicklung werden neue Stiftungsangebote auch stets in der Praxis getestet. In Zusammenarbeit mit einer Gruppe pädagogischer Fach- und Lehrkräfte aus Kitas sowie aus Horten und Grundschulen findet für jedes neue Fortbildungsangebot eine ausführliche Pilotierung statt, bevor die Fortbildungskonzepte und Materialien in den regionalen Netzwerken verbreitet werden. Dabei prüfen die mitwirkenden pädagogischen Fach- und Lehrkräfte erste Praxisideen auf ihre Umsetzbarkeit und geben ein Feedback zu den Unterstützungsangeboten der Stiftung. Die Fortbildungskonzepte werden auf Basis dieser Rückmeldungen überarbeitet und weiterentwickelt.



**Abbildung 1.** Übersicht der Maßnahmen zu Sicherung und Weiterentwicklung der Qualität der Stiftungsangebote

Auf Einrichtungsebene ist die Zertifizierung als „Haus der kleinen Forscher“ ein weiteres wichtiges Instrument der Qualitätsentwicklung (Stiftung Haus der kleinen Forscher, 2017b). Über die Vergabe des Zertifikats entscheidet die Stiftung in einem standardisierten Verfahren, das in Anlehnung an das Deutsche Kindergarten Gütesiegel und unter Beteiligung eines Teams aus Wissenschaftlerinnen und Wissenschaftlern<sup>3</sup> entwickelt wurde. Die Reliabilität und Validität des Zerti-

<sup>3</sup> Prof. Dr. Yvonne Anders, Dr. Christa Preissing, Prof. Dr. Ursula Rabe-Kleberg, Prof. Dr. Jörg Ramseger, Prof. Dr. Wolfgang Tietze.

fizierungsverfahrens für Kitas wurde in einer externen wissenschaftlichen Studie bestätigt (Anders & Ballaschk, 2014). Mit dem bundesweiten Kita-Wettbewerb „Forschergeist“<sup>4</sup> möchten die Deutsche Telekom Stiftung und die Stiftung „Haus der kleinen Forscher“ das Engagement der Kita-Fachkräfte sowie die Qualität der Einrichtungen würdigen. Gesucht und prämiert werden herausragende Projekte, die Mädchen und Jungen für die Welt der Mathematik, Informatik, Naturwissenschaften oder Technik begeistert haben. Mit der Ausschreibung 2018 findet der Wettbewerb bereits zum vierten Mal statt. Die prämierten Projekte werden dokumentiert und veröffentlicht, damit sie als gute Beispiele auch andere Fachkräfte für das Forschen und Entdecken in der Kita begeistern.

Neben einem kontinuierlichen Monitoring zu Zwecken der Qualitätssicherung und der Qualitätsentwicklung wird die Stiftungsarbeit im Rahmen einer langfristig angelegten externen Begleitforschung mit renommierten Partnern fachlich fundiert und in Forschungsprojekten evaluiert. Zwei unabhängige Forschungsgruppen untersuchten von 2013–2017 die naturwissenschaftlichen Bildungswirkungen in der frühen Kindheit (Stiftung Haus der kleinen Forscher, in Vorbereitung)<sup>5</sup>.

Ziel des ersten Forschungsprojektes *Early Steps Into Science* (EASI Science, gefördert von der Stiftung „Haus der kleinen Forscher“ und dem Bundesministerium für Bildung und Forschung) war es, Erkenntnisse über Wirkungen früher naturwissenschaftlicher Bildungsangebote auf naturwissenschaftliche Kompetenzen von pädagogischen Fachkräften und Kindern in Kitas zu gewinnen. Die Ergebnisse zeigen, dass pädagogische Fachkräfte mit naturwissenschaftlichen Fortbildungen über ein höheres Fachwissen verfügen als eine Vergleichsgruppe ohne Fortbildungen. Zudem sind Motivation zu und Interesse an naturwissenschaftlicher Bildung fortgebildeter Erzieherinnen und Erzieher größer. Auch die Kinder zeigen mehr Lernfreude, Interesse an Naturwissenschaften sowie Selbstvertrauen in ihr eigenes Können, wenn ihre Kita einen naturwissenschaftlichen Schwerpunkt hat.

Das zweite Forschungsprojekt *Early Steps Into Science and Literacy* (EASI Science-L, gefördert von der Stiftung „Haus der kleinen Forscher“, der Baden-Württemberg Stiftung und der Siemens Stiftung) untersuchte sprachliche Bildungswirkungen und die Interaktionsqualität im Kontext naturwissenschaftlicher Bildungsangebote. In der Studie konnte gezeigt werden, dass sich forschendes Lernen gut für die Sprachbildung eignet. Fachkräfte mit naturwissenschaftlichen Fortbildungen gestalten sprachlich anregendere Lerngelegenheiten für Kinder als Fachkräfte ohne Fortbildungen in diesem Bereich. Die sprachlichen Fähigkeiten der Kinder waren umso größer, wenn die sie betreuenden Fachkräfte zuvor eine

---

4 [www.forschergeist-wettbewerb.de](http://www.forschergeist-wettbewerb.de)

5 Mehr Informationen finden Sie unter [www.haus-der-kleinen-forscher.de](http://www.haus-der-kleinen-forscher.de) in der Rubrik „Wissenschaftliche Begleitung“.

kombinierte Fortbildung zu Naturwissenschaften und Sprache der Stiftung „Haus der kleinen Forscher“<sup>6</sup> besucht hatten. Zudem zeigen sich positive Zusammenhänge zwischen der durch die Fachkraft gestalteten naturwissenschaftsbezogenen Prozessqualität und den naturwissenschaftlichen Kompetenzen der Kinder. Die Stiftung wird die Ergebnisse der Studien für eine systematische Reflexion ihrer bestehenden Bildungsangebote und die wirkungsorientierte Entwicklung zukünftiger Fortbildungen nutzen.

Im Hinblick auf die bedarfsorientierte Weiterentwicklung der Stiftungsangebote fördert die Stiftung derzeit gemeinsam mit dem Bundesministerium für Bildung und Forschung eine Studie zu den „Entwicklungsverläufen von pädagogischen Fach- und Lehrkräften in der MINT-Bildung“ (EpFL MINT, Laufzeit 2017–2018). Ziel dieser Studie ist es, Einblicke in typische Entwicklungsverläufe in der „MINT-biographischen“ Professionalisierung pädagogischer Fach- und Lehrkräfte zu erhalten. Insbesondere sollen deren Lernbedarfe in verschiedenen Phasen ihrer Entwicklung als pädagogische MINT-Fachkraft und die Umsetzung von Lerninhalten aus Fortbildungsaktivitäten in den pädagogischen Alltag der Einrichtung untersucht werden. Die Ergebnisse der Studie sollen die Stiftung „Haus der kleinen Forscher“ dabei unterstützen, ihre Angebote zukünftig noch passgenauer und möglichst bedarfsgerecht (weiter) zu entwickeln.

Die Ergebnisse der wissenschaftlichen Begleitung veröffentlicht die Stiftung transparent in der vorliegenden wissenschaftlichen Schriftenreihe, alle Publikationen sind zudem über ihre Website frei verfügbar<sup>7</sup>.

Ein Wissenschaftlicher Beirat berät die Stiftung zu Forschungsfragen sowie zur fachlichen Fundierung des Stiftungsangebots. Er setzt sich aus unabhängigen Wissenschaftlerinnen und Wissenschaftlern unterschiedlicher Fachgebiete zusammen und spricht Empfehlungen an den Vorstand und den Stiftungsrat aus. Seit 2017 wird der Wissenschaftliche Beirat im Bereich der informatischen Bildung durch Prof. Dr. Johannes Magenheim und im Bereich der Bildung für nachhaltige Entwicklung durch Prof. Dr. Armin Lude verstärkt. Die Mitglieder des Beirats sind hochkarätige Expertinnen und Experten relevanter Disziplinen:

---

6 „Forschen mit Sprudelgas – Chemie ist überall“: Stiftung Haus der kleinen Forscher (2013). *Sprudelgas und andere Stoffe – Mit Kita- und Grundschulkindern Chemie entdecken und dabei die sprachliche Entwicklung unterstützen*. Berlin: Stiftung Haus der kleinen Forscher. Verfügbar unter: [www.haus-der-kleinen-forscher.de](http://www.haus-der-kleinen-forscher.de).

7 Alle Ergebnisse und Publikationen zur wissenschaftlichen Begleitung sind als PDF verfügbar unter: [www.haus-der-kleinen-forscher.de](http://www.haus-der-kleinen-forscher.de), Rubrik „Wissenschaftliche Begleitung“. Alle Ergebnisse der externen Begleitforschung werden zudem in der vorliegenden wissenschaftlichen Schriftenreihe veröffentlicht. Eine Übersicht der bisher erschienenen Bände befindet sich auf [www.haus-der-kleinen-forscher.de](http://www.haus-der-kleinen-forscher.de) bzw. am Ende dieses Bandes.

- Vorsitz: Prof. Dr. Hans-Günther Roßbach, Universität Bamberg
- Prof. Dr. Fabienne Becker-Stoll, Staatsinstitut für Frühpädagogik (IFP), München
- Prof. Dr. Marcus Hasselhorn/Prof. Dr. Jan Lonnemann, Deutsches Institut für Internationale Pädagogische Forschung (DIPF), Frankfurt/Universität Potsdam, Empirische Kindheitsforschung
- Prof. Dr. Christoph Igel, Deutsches Forschungszentrum für Künstliche Intelligenz (DFKI) GmbH
- Prof. Dr. Bernhard Kalicki, Deutsches Jugendinstitut e. V. (DJI), München, und Evangelische Hochschule Dresden
- Prof. Dr. Alexander Kauertz, Universität Koblenz-Landau
- Prof. Dr. Armin Lude, Pädagogische Hochschule Ludwigsburg
- Prof. Dr. Johannes Magenheimer, Universität Paderborn
- Prof. Dr. Jörg Ramseger, Freie Universität Berlin
- Prof. Pia S. Schober, Ph.D/Prof. Dr. C. Katharina Spieß , Universität Tübingen/ Deutsches Institut für Wirtschaftsforschung (DIW Berlin), Freie Universität Berlin
- Prof. Dr. Mirjam Steffensky, Leibniz-Institut für die Pädagogik der Naturwissenschaften und Mathematik (IPN), Kiel
- Prof. Dr. Wolfgang Tietze/Prof. Dr. Catherine Walter-Laager, PädQUIS gGmbH, An-Institut der Alice Salomon Hochschule, Berlin/Universität Graz, Erziehungs- und Bildungswissenschaft
- Prof. Dr. Christian Wiesmüller, Pädagogische Hochschule Karlsruhe, Deutsche Gesellschaft für Technische Bildung (DGTB)
- Prof. Dr. Bernd Wollring, Universität Kassel

## 2 Das „I“ in MINT – Relevanz der frühen informatischen Bildung

Informatik begegnet uns im Alltag fast überall, nur bewusst ist uns dies oftmals nicht. Egal ob wir darauf warten, dass die digital gesteuerte Ampel endlich grün wird, wir unser Smartphone bedienen oder die vollautomatisierte Waschmaschine anstellen: Informatik ist stets dabei und hat somit auch Bedeutung im Leben von Kindern. Einer Studie im Auftrag des Deutschen Instituts für Vertrauen und Sicherheit im Internet (DIVSI) zufolge, ist „die zunehmende Digitalisierung des Alltags bereits bei kleinen Kindern fest im Familienleben verankert – als Thema und im konkreten Handeln“ (DIVSI U9-Studie, 2015). Allerdings werden die Kinder und Jugendlichen durch das Aufwachsen in einer digital geprägten Umwelt nicht automatisch zu kompetenten Nutzerinnen und Nutzern digitaler Technologien, wie eine Studie zu computer- und informationsbezogenen Kompetenzen bei Jugendlichen gezeigt hat (Eickelmann, 2015). Demnach verfügen fast 30 Prozent der Jugendlichen in Deutschland nicht über ausreichende Computer- und IT-Kenntnisse für eine erfolgreiche gesellschaftliche Teilhabe.

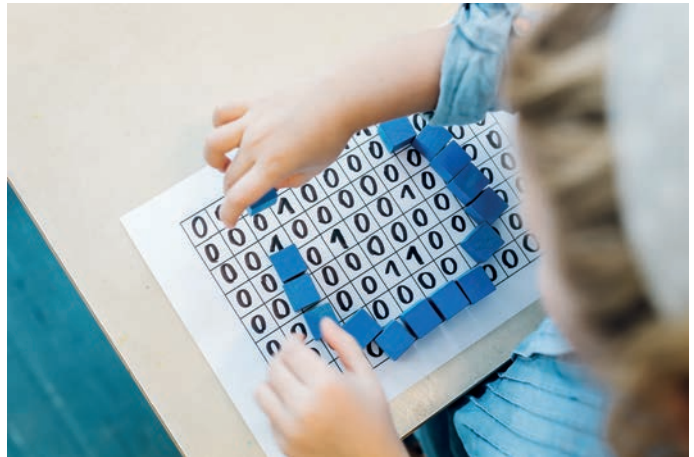
Im Rahmen des Nationalen IT-Gipfels, der am 16. und 17. November 2016 in Saarbrücken stattfand, appellierte die Deutsche Mathematiker-Vereinigung: „Nicht das bloße Verwenden digitaler Medien, sondern das Verständnis ihrer Grundlagen schafft die Voraussetzung für einen souveränen digitalen Wandel. [...] Ziel sollte sein, grundlegende Kompetenzen zu vermitteln, die Lernende zu einem mündigen Umgang mit digitalen Neuerungen befähigen.“<sup>8</sup> Damit rückt die informatische Bildung als gesellschaftliche Aufgabe immer mehr in den Fokus und sollte zukünftig ein fester Bestandteil einer grundlegenden Allgemeinbildung sein. Die Beherrschung elementarer informatischer Methoden und Werkzeuge bekommt somit eine ähnliche Bedeutung wie Schreiben, Lesen und Rechnen. Alle Mädchen und Jungen sollten die Chance zur frühzeitigen Bildung in diesem Bereich erhalten. Dies bedeutet, den Kindern Raum zu geben, ihre Fragen zu digitalen Medien und deren informatischen Grundlagen zu stellen und entdeckend und forschend Antworten zu suchen.

Die Stiftung „Haus der kleinen Forscher“ setzt an diesen Herausforderungen an und möchte über die Fort- und Weiterbildung pädagogischer Fach- und Lehrkräfte in Kitas, Horten und Grundschulen die Bildungschancen von Mädchen und Jungen in einem Kernbereich der digitalen Bildung stärken. Während es zunehmend Initiativen zur digitalen Mediennutzung gibt, fokussiert sich die Stiftung

---

8 Pressemitteilung der DMV zur „Bildungsoffensive zur digitalen Wissensgesellschaft“. *Mitteilungen der Deutschen Mathematiker-Vereinigung* (2017), 24(4), S. 191–191. doi:10.1515/dmvm-2016-0074

bei der Erschließung des Themenbereichs „I in MINT“ auf die informatische Bildung. Bisher existieren keine flächendeckenden Angebote zur informatischen Frühbildung für Mädchen und Jungen im Alter von 3 bis 10 Jahren in Deutschland. Während Bildung in Naturwissenschaften, Mathematik und Technik seit einigen Jahren in die Kita Einzug halten konnte, klappt bezogen auf die informatische Bildung eine



Lücke, die sich mit der Beschleunigung des digitalen Wandels der Gesellschaft beständig vergrößert. Das Ziel der Stiftung in diesem Bildungsbereich ist es, den Kindern erste Erfahrungen im Bereich der informatischen Bildung zu ermöglichen, um perspektivisch ein Grundlagenverständnis von Informatik(systemen) zu entwickeln. Somit wurde das Stiftungsangebot 2017 um ein Fortbildungsmodul zum Thema „Informatik“ ergänzt. Es geht dabei nicht um die verstärkte Nutzung digitaler Medien, sondern um das Verstehen zugrunde liegender Konzepte. Die Stiftung bietet ein Angebot für pädagogische Fach- und Lehrkräfte, um das Thema Informatik in ihrer Arbeit mit drei- bis zehnjährigen Kindern auch abseits von Computer oder Tablet zu thematisieren und, wie bisher gewohnt, mit Alltagsmaterialien umzusetzen.

Um dem hohen Qualitätsanspruch der Stiftung gerecht zu werden, wurde die inhaltliche Entwicklung auch im informatischen Bildungsbereich fachlich fundiert und wissenschaftlich begleitet. Seit 2015 steht die Stiftung daher im engen Austausch mit Fachexpertinnen und -experten der informatischen Bildung, die die Themenentwicklung in Fachforen und Expertentreffen kritisch begleiten und beraten (siehe folgendes Kapitel).

### 3 Fachliche Fundierung des Themenbereichs „Informatik“

Alle inhaltlichen Angebote der Stiftung „Haus der kleinen Forscher“ werden, aufbauend auf dem aktuellen wissenschaftlichen Forschungsstand, zu der jeweiligen Thematik erarbeitet. Wie bereits beschrieben, existieren in Deutschland in der informatischen Bildung bisher kaum fachlich fundierte oder gar evaluierte Konzepte und Bildungsansätze für den Primar- oder gar Elementarbereich<sup>9</sup>. Daher hat sich die Stiftung bei der fachlichen Fundierung und der Entwicklung von Stiftungsangeboten im Bereich der Informatik noch intensiver als schon bei den anderen MINT-Bildungsbereichen Mathematik, Naturwissenschaften und Technik von Fachexpertinnen und -experten beraten und begleiten lassen. Erstmals wurden sogar namhafte Fachleute aus internationalen Institutionen und Initiativen in die Erarbeitung des Themenbereichs miteinbezogen. Darüber hinaus pflegt die Stiftung einen kontinuierlichen Austausch mit relevanten Partnern und anderen Initiativen, die im Bereich der informatischen Bildung tätig sind, wie der Deutschen Telekom Stiftung und der Wissensfabrik – Unternehmen für Deutschland e. V.

Im Rahmen der fachlichen Fundierung initiierte die Stiftung die Arbeitsgruppe „Zieldimensionen informatischer Bildung im Elementar- und Primarbereich“<sup>10</sup> mit Expertinnen und Experten aus dem Bereich der Informatikdidaktik und der Grundschulpädagogik (Nadine Bergner, Hilde Köster, Johannes Magenheim, Kathrin Müller, Ralf Romeike, Ulrik Schroeder, Carsten Schulte). Die Arbeitsgruppe erarbeitete von 2015–2017 eine Expertise, in der sie theoretisch fundierte Zieldimensionen im Rahmen der informatischen Bildung für Kinder im Kita- und Grundschulalter, aber auch für pädagogische Fach- und Lehrkräfte formulierte und Instrumente für deren Messung sichtete. Darüber hinaus beschäftigte sich die Arbeitsgruppe mit Gelingensbedingungen für die Erreichung dieser Ziele und somit für eine effektive und wirkungsvolle frühe informatische Bildung in der Praxis. Im Vergleich zu den Bereichen Mathematik, Naturwissenschaften und Technik, wo schon deutlich mehr Forschungserkenntnisse vorlagen, hat die Expertengruppe hier eine hervorragende Pionierarbeit geleistet.

Um die ersten Ergebnisse der Arbeitsgruppe sowie den aktuellen Forschungsstand zur frühen informatischen Bildung mit einem erweiterten Expertenkreis zu diskutieren, veranstaltete die Stiftung im Herbst 2015 in Berlin das erste internati-

---

<sup>9</sup> Zu den wenigen fachlich fundierten und evaluierten Angeboten gehört die Experimentierkiste Informatik – ein informatisches Bildungsangebot für Kinder im Vor- und Grundschulalter, das seit 2015 von der Forschungsgruppe Elementarinformatik der Universität Bamberg (FELI) entwickelt wird (<http://www.uni-bamberg.de/kogsys/feli> [Zugriff am 05.03.2018]).

<sup>10</sup> Mehr Informationen zur Arbeitsgruppe unter [www.haus-der-kleinen-forscher.de](http://www.haus-der-kleinen-forscher.de)

onale Fachforum „Early Education in Computer Science“ („Frühe informatische Bildung“) mit führenden nationalen und internationalen Fachleuten aus Wissenschaft und Praxis. Expertinnen und Experten aus Deutschland, der Schweiz, Großbritannien und der Slowakei setzten sich mit der Frage auseinander, wie eine erfolgreiche informatische Bildung für Kita- und Grundschulkindern gelingen kann. Es wurde u. a. kontrovers diskutiert, ob die aktive Nutzung digitaler Geräte unabdingbar für die informatische Bildung ist oder ob erste Informatik-Kompetenzen auch ohne die Verwendung digitaler Geräte entwickelt werden können. Konsens bestand darüber, dass informatische Bildung frühzeitig implementiert werden sollte, um den Kindern erste Erfahrungen in diesem Bereich zu ermöglichen und perspektivisch ein Grundlagenvverständnis von Informatik(systemen) zu vermitteln.

Im Herbst 2016 fand das zweite internationale Fachforum in Berlin statt und fokussierte auf die Umsetzung der Fachempfehlungen in die inhaltlichen Angebote der Stiftung im Bereich der informatischen Bildung. Neben führenden Fachleuten aus der Informatikdidaktik waren dieses Mal vor allem Vertreterinnen und Vertreter aus nationalen und internationalen Praxisinitiativen der frühen informatischen Bildung (u. a. Neuseeland, Großbritannien) zu Gast und präsentierten ihre Ideen und Erfahrungen. Die Stiftung stellte erste, auf den Empfehlungen der Arbeitsgruppe basierende Praxisideen und -erfahrungen vor und diskutierte diese mit den Fachleuten. Die Expertinnen und Experten betonten den notwendigen Alltagsbezug für die Kinder. Sie bestärkten die Stiftung darin, auch in der informatischen Bildung die Umsetzung der Praxisideen mit Alltagsmaterialien zu ermöglichen. Das heißt, auch ohne die Verwendung digitaler Geräte können Kinder erste Erfahrungen im Bereich der informatischen Bildung machen. Des Weiteren wurde deutlich, dass es gerade im Elementarbereich nur sehr wenig Erfahrung gibt. Erste Angebote der Stiftung sollten daher wissenschaftlich begleitet werden.

Die Ergebnisse der Arbeitsgruppe sowie erste Praxisideen der Stiftung wurden weiterhin auf der 4. Sitzung des Wissenschaftlichen Beirats der Stiftung im Oktober 2016 in Berlin vorgestellt und diskutiert. Die Beiratsmitglieder würdigten die ausführliche Arbeit der Arbeitsgruppe und betonten die Notwendigkeit von Forschungsprojekten angesichts der fehlenden empirischen Grundlagen im Bereich der frühen informatischen Bildung. Zudem begrüßten sie die bisher geleistete Arbeit der Stiftung im Bereich der frühen informatischen Bildung, die auf Fachexpertise und internationalen Erfahrungen aufbaut.

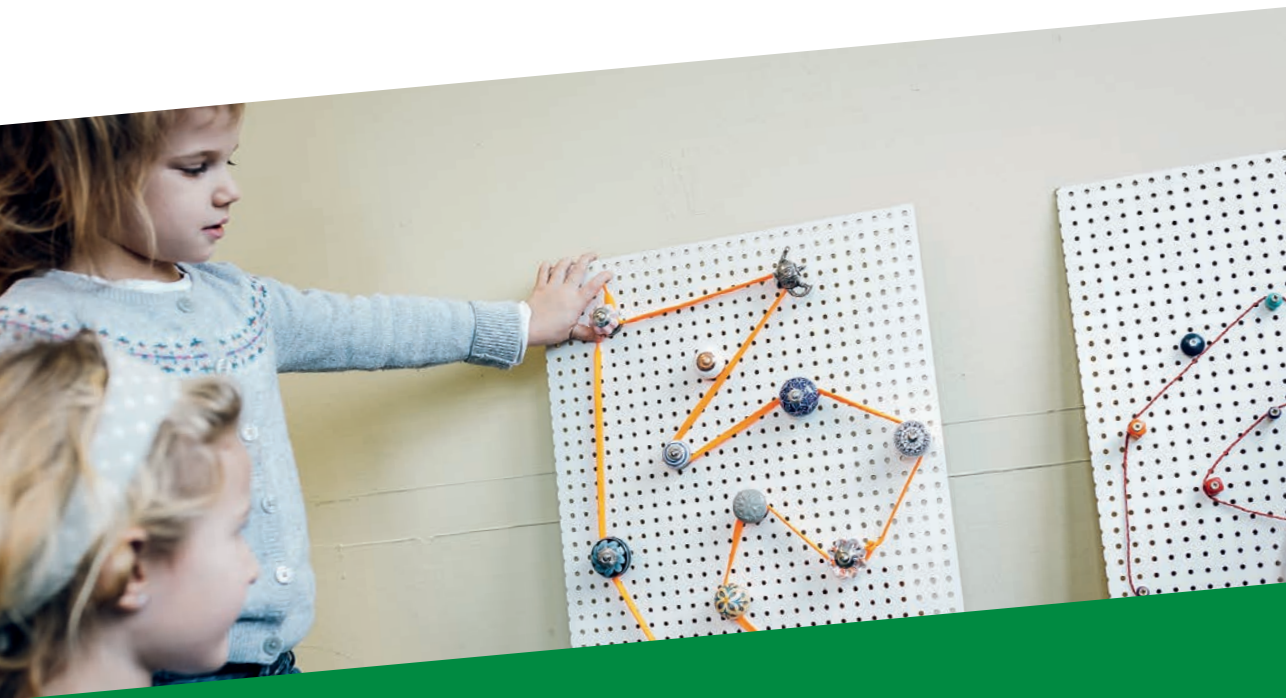
Ergänzend zur Expertise zu den Zieldimensionen informatischer Bildung haben Nadine Bergner und Kathrin Müller eine Fachempfehlung erstellt, in der sie eine Auswahl an Informatiksystemen für Kinder im Kita- und Grundschulalter vorstellen und beschreiben, wie und unter welchen Voraussetzungen diese im Elementar- und Primarbereich verwendet werden können.



Im vorliegenden Band werden die zentralen Ergebnisse der fachlichen Fundierung der frühen informatischen Bildung veröffentlicht. Die Beiträge stellen Ziele und Konzepte für eine gelingende informatische Bildung im Elementar- und Primarbereich in den Fokus und bilden das Fundament für die inhaltliche Entwicklung des Stiftungsangebots zum Themenbereich Informatik.

# Zusammenfassung zentraler Ergebnisse

Stiftung Haus der kleinen Forscher



## Zusammenfassung zentraler Ergebnisse

Der neunte Band der Schriftenreihe „Wissenschaftliche Untersuchungen zur Arbeit der Stiftung ‚Haus der kleinen Forscher‘“ stellt die informatische Bildung im Elementar- und Primarbereich in den Fokus. Er beinhaltet eine umfassende Expertise, die im Rahmen der fachlichen Fundierung des Themenbereichs Informatik von Fachexpertinnen und -experten für die Stiftung erstellt wurde und die theoretische Grundlage für die Entwicklung der inhaltlichen Angebote der Stiftung im Bereich Informatik bildet. Außerdem gibt eine Fachempfehlung einen Überblick zu Informatiksystemen für Kinder im Kita- und Grundschulalter und deren Verwendung im Elementar- und Primarbereich.

Im ersten Beitrag „Zieldimensionen informatischer Bildung im Elementar- und Primarbereich“ spezifizieren Nadine Bergner, Hilde Köster, Johannes Magenheim, Kathrin Müller, Ralf Romeike, Ulrik Schroeder und Carsten Schulte pädagogisch-inhaltliche Zieldimensionen für die frühe informatische Bildung. Aufgrund der bisher kaum vorhandenen theoretischen und empirischen Forschungserkenntnisse in diesem Bereich orientierten sich die Autorinnen und Autoren bei der Herleitung der fachspezifischen Zieldimensionen für den Elementar- und Primarbereich an den von der Gesellschaft für Informatik vorgeschlagenen Standards für die Sekundarstufe I (GI – Gesellschaft für Informatik e. V., 2008) sowie an existierenden internationalen Curricula der frühen informatischen Bildung. Die Analyse dieser vorhandenen Konzepte führte dazu, dass das Expertenteam im Vergleich zu den GI-Bildungsstandards einen neuen Prozessbereich ‚Interagieren und Explorieren‘ einführte, um die Bedeutung für den spielerisch erkundenden Umgang mit Informatiksystemen im Kita- und Grundschulbereich zu betonen. Die abgeleiteten Zieldimensionen werden sowohl auf Ebene der Kinder als auch auf Ebene der pädagogischen Fach- und Lehrkräfte aus dem Elementar- und Primarbereich erörtert.

Auf Ebene der Kinder empfehlen die Autorinnen und Autoren folgende Zielbereiche:

- Motivation, Interesse und Selbstwirksamkeit im Umgang mit Informatik(systemen)
- informatische Prozessbereiche
- informatische Inhaltsbereiche

Zudem werden Zielkompetenzen auf Ebene der Kinder priorisiert, die sich an beschriebenen Leitkriterien orientieren und die wichtigsten altersgemäßen Verknüpfungen von Prozess- und Inhaltsbereichen darstellen.

Auf Ebene der pädagogischen Fach- und Lehrkräfte werden folgende Zieldimensionen empfohlen:

- Motivation, Interesse und Selbstwirksamkeit bezogen auf informatische Bildung
- Einstellungen, Haltungen und Rollenverständnis in Bezug auf informatische Bildung
- informatische Prozessbereiche
- informatische Inhaltsbereiche
- informatikdidaktische Kompetenzen

Des Weiteren erörtern die Autorinnen und Autoren Gelingensbedingungen für eine effektive und wirkungsvolle frühe informatische Bildung in der Praxis. Diese beziehen sich zum einen auf die Kompetenzen und Einstellungen der pädagogischen Fach- und Lehrkräfte und zum anderen auf institutionelle Rahmenbedingungen. Eine gelingende Auseinandersetzung mit informatischen Themen im Alltag setzt insbesondere informatikdidaktische Kompetenzen bei den pädagogischen Fach- und Lehrkräften voraus. Dazu gehört u. a. das Erkennen und die Gestaltung effektiver Lernumgebungen, die auf die individuellen Entwicklungsstände der Kinder angepasste Auswahl an Materialien mit hohem informatischem Potenzial sowie die (Aufrechterhaltung der) Motivation der Mädchen und Jungen in den Lehr-Lern-Situationen. Eine wichtige Voraussetzung für den Erwerb dieser fachdidaktischen Kompetenzen ist nach Ansicht der Autorinnen und Autoren die subjektive Einstellung und Motivation der Pädagoginnen und Pädagogen in Bezug auf die frühe informatische Bildung. Das eigene Interesse an informatischen Themen sollte sich zudem positiv auf die Motivation der Kinder auswirken.

Ein weiterer wesentlicher Faktor für die erfolgreiche Umsetzung informatischer Bildung im Elementar- und Primarbereich ist die Vermeidung von falschen bzw. verzerrten Vorstellungen zur Informatik. Unter diesem Aspekt sieht das Expertenteam auch die Kooperation der Bildungseinrichtungen mit den Familien und ebenso Entscheidungsträgern als eine wichtige Gelingensbedingung an, um somit eventuellen Vorbehalten gegenüber der Thematik zu begegnen. Im Hinblick auf die Ausstattung der Bildungseinrichtungen betonen die Autorinnen und Autoren, dass vor allem in der Kita informatische Bildung auch ohne den Einsatz von digitalen Geräten möglich und sinnvoll ist. Insbesondere mit zunehmendem

Alter bietet die Verwendung von Informatiksystemen Chancen auf weitere Motivation und Lernerfahrungen und die Verknüpfung der vermittelten Inhalte mit den tatsächlich im Alltag der Kinder vorkommenden Geräten. Voraussetzung dafür ist dann natürlich eine entsprechende technische Ausstattung der Bildungseinrichtungen.

Schließlich unterstreicht die Expertise die Notwendigkeit der Instrumentenentwicklung für die Erfassung der definierten Zielbereiche. Die Autorinnen und Autoren führen wichtige Kriterien für die Erstellung empirischer Messinstrumente zur Evaluation von Umsetzung und Wirkung früher informatischer Bildung in der Praxis auf. Sie betonen dabei die Bedeutsamkeit der wissenschaftlichen Begleitung bei der Implementierung dieses Bildungsbereichs.

Im zweiten Beitrag des Bandes geben Nadine Bergner und Kathrin Müller eine Fachempfehlung zu Informatiksystemen für Kinder im Kita- und Grundschulalter und deren Verwendung im Elementar- und Primarbereich. Auch wenn der Einsatz solcher Systeme keine unabdingbare Voraussetzung für die Umsetzung informatischer Bildung in Kita, Hort und Grundschule ist, bieten sie eine weitere Möglichkeit, informatische Inhalte in der frühen Bildung zu vermitteln. Die Autorinnen beschreiben zunächst eine Auswahl an geeigneten Informatiksystemen für den Elementar- und Primarbereich und geben eine Einschätzung darüber, für welches Alter diese Geräte geeignet sind, welche Vorerfahrungen sowohl auf Seiten der Lernbegleitung als auch auf Seiten der Kinder notwendig sind und welche Lernziele mit den Systemen verfolgt werden können. Zudem stellen sie einen Bezug zu den Zieldimensionen informatischer Bildung her.

Das Fazit dieses Bandes beschreibt die Umsetzung der Fachempfehlungen in die inhaltlichen Angebote der Stiftung „Haus der kleinen Forscher“ und gibt einen Ausblick auf die weitere Stiftungsarbeit. Auf Grundlage der vorliegenden Fachempfehlungen hat die Stiftung ihr Angebot um den Bereich der frühen informatischen Bildung erweitert und ein Fortbildungskonzept sowie umfangreiche Materialien für die Praxis entwickelt.

# Zieldimensionen informatischer Bildung im Elementar- und Primarbereich

Nadine Bergner, Hilde Köster, Johannes Magenheim, Kathrin Müller,  
Ralf Romeike, Ulrik Schroeder, Carsten Schulte



- 1 Potenziale informatischer Bildung
- 2 Fundierung von Zieldimensionen auf Ebene der Kinder
- 3 Zieldimensionen auf Ebene der Kinder
- 4 Zieldimensionen auf Ebene der pädagogischen Fach- und Lehrkräfte
- 5 Beispiele priorisierter Kompetenzbereiche der frühen informatischen Bildung
- 6 Gelingensbedingungen früher informatischer Bildung
- 7 Fazit
- 8 Anhang

# 1 Potenziale informatischer Bildung

Viele Kinder in Deutschland wachsen zurzeit mit großartigen Möglichkeiten und Perspektiven auf. Das hat vielfältige Ursachen, zu denen nicht zuletzt ein gutes Bildungssystem gehört. Eine weitere, zwar im Alltag wirkmächtige, derzeit aber im Bildungssystem nicht in dem Ausmaß widergespiegelte Ursache sind die Veränderungen durch die Digitalisierung, die aktuell alle Lebensbereiche erfasst und transformiert. Dadurch werden sich nochmals die Möglichkeiten variieren, vervielfältigen und auch zu neuartigen Herausforderungen führen. Digitalisierung bedeutet im engen Sinn die Überführung analoger (d. h. stufenloser und damit theoretisch unendlich verschiedener) Daten in die digitale, d. h. auf Ziffern (engl. „digit“) abbildbare und damit durch Computer verarbeitbare Form. Sie führt dazu, dass grundsätzlich Daten aller digitalisierbaren Lebensbereiche zu geringen Kosten maschinell erfasst, gespeichert, verarbeitet, übermittelt und verbreitet werden können. Durch die damit einhergehende enorme Steigerung der Verfügbarkeit von Information ergeben sich weitreichende Möglichkeiten und Herausforderungen für die Gesellschaft.

Momentan ist noch unklar, wie auf diese umfassenden Veränderungen am besten reagiert werden kann. Wir werden in dieser Expertise anhand theoretischer Überlegungen den Blick in vergleichbare Länder und deren Ansätze sowie der Analyse verschiedener Praxisprojekte und des Forschungsstands in der Didaktik der Informatik untersuchen, welchen Beitrag die informatische Bildung zu einer zeitgemäßen und zukunftssicheren Bildung in Kita und Grundschulen leisten kann.

Aus unserer vor allem informatikdidaktisch geprägten Sichtweise ist klar: Die neuen digitalen Medien sind nicht nur eine weitere Erscheinungsform im Alltag der Kinder, sie stellen vielmehr einen neuen und eigenständigen Bildungsbereich dar – neben anderen, die in Bildungsplänen für Kindertageseinrichtungen oder dem Perspektivrahmen Sachunterricht aufgeführt werden (vgl. Berlin, 2014; Gesellschaft für Didaktik des Sachunterrichts, 2013). Dieser Bildungsbereich muss sich erst noch entwickeln, und dazu ziehen wir die Diskussionen aus den verschiedenen Bereichen (digitale Bildung, Medienbildung, informatische Bildung in höheren Jahrgangsstufen) heran, um so eine altersgemäße Konzeption zu entwerfen<sup>11</sup>.

Zentrales Ziel dieses neuen Bildungsbereichs bzw. dieser neuen Perspektive (zumindest teilweise neu, weil sie sich nicht vollständig durch die technische

---

<sup>11</sup> Die Begrifflichkeiten Informatik, informatische Bildung, Medienbildung und digitale Bildung sind in der allgemeinen Diskussion nicht sehr trennscharf definiert. Wir werden im Verlauf dieser Exposition versuchen – soweit möglich –, die verschiedenen Begriffe zu entfalten und zu entwirren.

Perspektive beschreiben lässt (s.u.)) ist das eigenständige, verantwortungsvolle Handeln in einer digital geprägten Lebenswirklichkeit. Um dieses zu erreichen, bedarf es der Kenntnis der grundlegenden Funktionsprinzipien und Wirkungsweisen digitaler Technologien, da diese – im wahrsten Sinne des Wortes – sonst nur oberflächlich genutzt, aber nur unzureichend erschlossen, (mit-)gestaltet und bewertet werden können.

Wir beginnen mit der Frage „Was ist eigentlich Informatik?“

## 1.1 Was ist Informatik?

Informatik gehört zur unmittelbaren Lebenswirklichkeit von Kindern – nicht nur in Form von Computern und anderen (offensichtlich) digitalen Geräten wie Smartphones, Tablets, Foto- und Videokameras, Fernsehgeräten, Musikabspielgeräten usw., sondern auch in Form von Geräten und Maschinen, denen man dies nicht sofort ansieht, wie Waschmaschinen, Uhren, Mikrowellengeräten, Ampeln und Autos.

Informatik ist überall dort, wo

- **Abläufe automatisiert gesteuert und geregelt** (die Ampelsteuerung, der Fahrplan der Bahn oder die Tour des Müllwagens, das Programm der Waschmaschine),
- **Daten digital gespeichert und ausgegeben** (Kamera, Hörbuch),
- **Daten übertragen** (Handy, Fernseher, Radio) oder
- **Daten verändert und berechnet werden** (die Wettervorhersage, der Taschenrechner, das Navigationssystem im Auto ...).

Informatische Bildung greift Erfahrungen im Umgang mit digitalen Geräten auf, um so Möglichkeiten für den (kritischen) Zugang (nicht gleichzusetzen mit *Umgang*) zu schaffen – insbesondere auch Zugänge, die über das Nachvollziehen vorgegebener Abläufe hinausgehen: Adaption, Konfiguration, Konstruktion und Gestaltung. Informatische Bildung beruht daher zu einem großen Teil auf den – sozusagen hinter der Benutzungsoberfläche verborgenen – Prinzipien und Konzepten, die zur Konstruktion und zur Beschreibung der Wirkungsweise digitaler Systeme benötigt werden und befähigt damit zu deren effektiven und effizienten Nutzung und Gestaltung.



Informatik beschäftigt sich mit Information, die eine Erscheinungsform der realen Welt darstellt neben Stoffen (Gegenstand der Chemie) und Energie (Gegenstand der Physik).

## 1.2 Informatik als Wissenschaft

Der Begriff Informatik wurde 1957 in Deutschland als Kunstwort aus Information und Automatik geprägt: „Sie [Ingenieure in den USA und Deutschland] fanden, dass man mit elektrischen Schaltungen Zahlenrechnungen durchführen kann, und zwar mit einer Schnelligkeit, wie sie bis dahin einfach unvorstellbar war. Damit begann die automatische Informationsverarbeitung. Wir nennen sie ‚INFORMATIK‘“ (Steinbuch, 1957, S. 171).

Informatik wird auch heute noch als die Wissenschaft der automatischen Informationsverarbeitung definiert, so heißt es im Duden Informatik (Claus & Schwill, 2006, S. 305): „**Informatik** ist die Wissenschaft von der systematischen Darstellung, Speicherung, Verarbeitung und Übertragung von Information, besonders der automatischen Verarbeitung mithilfe von Digitalrechnern.“

Neben den Wurzeln in der Mathematik und in der Ingenieurwissenschaft hat die Informatik aber auch ihre Wurzeln in den Methoden und Fragestellungen der Naturwissenschaften und ist ebenso eine empirische Disziplin. Nach Tedre und Apiola sind diese drei Traditionslinien ineinander verwoben, aber in den Teildisziplinen klar unterscheidbar (Tedre & Apiola, 2013). Bis heute sind diese drei Traditionslinien erkennbar:

- aus der Automatisierung bzw. den Ingenieurwissenschaften: Konstruktion von technischen Lösungen;
- aus der Mathematik: formale Strukturen, Abstraktion, Untersuchen von Algorithmen und zunehmend;
- aus der Tradition der Naturwissenschaften: Erklären der Welt, hier dann Erklären und Untersuchen der digitalen Welt.

Die Informatik bewegt sich dabei (auch) tief im Bereich psychologischer und soziologischer Fragestellungen, der Mediennutzungsforschung und der Usability (Nutzerfreundlichkeit): Wie gehen Menschen mit digitalen Geräten um? Wie müssen diese gestaltet werden, damit sie benutzbar sind? Welche Effekte und auch welche ggf. unerwünschten Auswirkungen hat die Verbreitung der Nutzung, und wie kann darauf reagiert werden?

Was bedeutet das für die informatische Bildung, um die es uns hier geht? Um die Frage zu beantworten, werden wir im Folgenden genauer einzelne Facetten der

Informatik beleuchten. Dabei schauen wir uns insbesondere die Perspektive der Konstruktion an.

### 1.3 Konstruktion in der Informatik

In der ingenieurwissenschaftlichen Perspektive ist die Informatik eine **konstruierende Wissenschaft**, die sich im Kern mit Fragen rund um und über die Konstruktion von digitalen Artefakten<sup>12</sup> – Hardware und Software – beschäftigt. Solche Fragen sind schnell genereller Natur, etwa die Frage danach, ob alle (berechenbaren) Problemstellungen mit Informatiksystemen effizient, d. h. in realistischer Zeit, lösbar sind. (Sind sie nicht – auf dieser Einsicht beruhen die meisten Verschlüsselungsverfahren.) Die Fragen sind oft sehr konkret und praxis- bzw. auf den Menschen bezogen. Beispielsweise die Frage, wie man am besten im Team Software entwickelt; so benötigen viele Softwareprojekte hunderte von Person Jahren: Wie kann man die Arbeit entsprechend aufteilen, so dass man nicht hunderte Jahre auf die Software warten muss?

Für Bildung sind jedoch weniger die Fragen nach technischen Details, sondern vor allem die grundsätzlichen Fragen interessant, wie diese Konstruktion im Prinzip vonstattengeht, was das eigentlich ist, was konstruiert wird, und wie dadurch das Leben des Einzelnen beeinflusst wird.

Ein Ziel der Informatik ist es, effiziente auf digitalen Artefakten ausführbare Algorithmen zu entwickeln, die Abläufe zu automatisieren und Daten zu transformieren. Aber was bedeutet das? Wieso ist das eine eigene, bis dato in der Geschichte der Menschheit nie dagewesene neuartige technologische Errungenschaft, die tatsächlich alle Lebensbereiche betrifft und sehr oft einschneidend verändert?

Bereits in den ersten informatikdidaktischen Abhandlungen Ende der 1960er-Jahre werden die neuen – digitalen – Technologien „als prinzipieller Abschluss der technikgeschichtlichen Entwicklung“ (Frank & Meyer, 1974, S. 592) der Menschheit verstanden. Nach dieser Sichtweise hat der Mensch immer mehr Funktionen an Werkzeuge, also an Objekte, delegiert. Drei Phasen werden unterschieden (vgl. Frank & Meyer, 1974)

1. Phase: die Objektivierung der Gliedmaßen bzw. Organe wie Faust, Zähne, Hand durch entsprechende, aus der Umwelt stammende Werkzeuge, die dazu meist verändert wurden (Stein, Faustkeil)

---

<sup>12</sup> Artefakt (lat. ars, artis ‚Handwerk‘ und factum ‚das Gemachte‘) steht für: von Menschen erzeugte Gegenstände (Wikipedia, 2014).

2. Phase: die Objektivierung der körperlichen Arbeit durch Maschinen

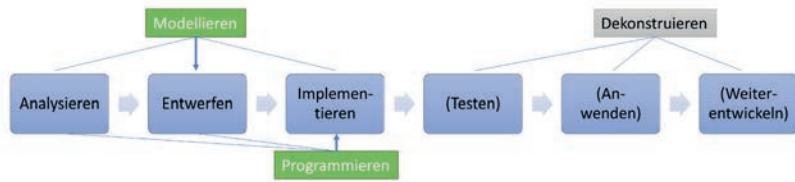
3. Phase: die Objektivierung der geistigen Arbeit durch Rechner

Während in den ersten Ideen zum Informatikunterricht – zu einer Zeit vor (!) Erfindung des PC – die Hardware dieser neuen Technologie im Mittelpunkt stand, entwickelte sich rasch ein Konsens, der in abstrakter Form bis heute gültig ist und in den 1976er-Empfehlungen der Gesellschaft für Informatik (GI) zu Zielen und Inhalten des Informatikunterrichts in knapper Form die wesentlichen Gedanken des algorithmenorientierten Ansatzes zusammenfasst: „Gegenstand des Informatikunterrichts ist in erster Linie nicht die technische Funktion des Rechners. Vielmehr erscheint es wesentlich, Möglichkeiten der Anwendung des Rechners sowie Auswirkungen und Grenzen des Einsatzes von Rechenanlagen zu kennen und zu erkennen“ (Eickel et al., 1969, S. 35).

1976 wird die Zielsetzung, „Möglichkeiten der Anwendung des Rechners“ zu erkunden, so verstanden, dass Fehlvorstellungen über den Rechner, etwa dass er ein „Elektronengehirn“ sei und eine „Knopfdruckautomatik“ enthalte, ab- und rationales Verstehen aufgebaut werden sollte. Im Einzelnen bedeutet dies: „Problemverständnis für die Möglichkeiten“, „Einordnen von Informatik-Kenntnissen in die Erlebniswelt“ sowie spezielle Informatik-Kenntnisse (Eickel et al., 1969, S. 35).

Methodisch schlugen die frühen Curricula vor, dazu das Programmieren in den Mittelpunkt zu stellen. Die Lernenden sollten also (im Prinzip) durch eigene Erfahrung des Konstruktionsprozesses grundlegende Kenntnisse über Computer und informatische Konzepte erwerben sowie Möglichkeiten, Grenzen und Auswirkungen beurteilen können und schließlich das Gelernte mittels algorithmischer Problemlösung und Automation selbst einsetzen und anwenden können.

In der Umsetzung wurde jedoch – zumindest anfänglich – anstelle des Problemlösens und Reflektierens dieser Art der Problemlösung allzu oft im Unterricht der Schwerpunkt auf die Einübung einzelner programmiersprachlicher Konstrukte gelegt: Anstelle Denk- und Problemlösefähigkeiten einzuüben, wurden Befehle einer Programmiersprache vorgestellt und isoliert eingeübt. Um das zu verhindern, wurde informatische Bildung im Weiteren sehr stark projektorientiert und damit auch – zumindest ansatzweise – interdisziplinär ausgelegt: Das Lösen realweltlicher Anwendungsprobleme sollte im Mittelpunkt stehen. Zudem wurde die eigentliche technische Konstruktion weniger stark gewichtet, um die eigentlich planerischen und gestalterischen Prozesse in den Mittelpunkt zu stellen (ausführlicher Schulte, 2001). Diese Orientierung wird auch als Modellieren im Gegensatz zum Programmieren bezeichnet (vgl. Abbildung 2).



**Abbildung 2.** Schritte des Konstruktionsprozesses in der Informatik (mittlere Kästen) und jeweilige Betonung: Programmieren betont die Implementation, Modellieren den Entwurf. Die Dekonstruktion sowie neuere zyklische und agile Modelle nehmen verstärkt die stete Adaption und Weiterentwicklung bestehender Produkte oder von Produktteilen auf.

Allgemein gesprochen geht es beim Analysieren darum, das Problem zu verstehen und in einzeln lösbare Teilprobleme zu zerlegen. Für diese wird dann im Entwurf eine Lösung entwickelt und in der Implementationsphase umgesetzt. Diese Schritte gelten so auch für Problemstellungen außerhalb der Informatik, so dass die Vermittlung solcher allgemeinen Problemlösefähigkeiten immer auch als ein wichtiger Beitrag der informatischen Bildung zur Allgemeinbildung gesehen wird. Das erklärt jedoch noch nicht den besonderen Stellenwert des informatischen Problemlösens für die digitale Welt. Daher werden wir nun diese Problemlösungs- und Konstruktionstätigkeiten genauer untersuchen.

Nach der weiter oben erwähnten Einteilung in verschiedene Entwicklungsstufen der Techniknutzung ist das Besondere an der Informatik, dass die Automation geistiger Prozesse im Mittelpunkt steht. Automation (geistiger Prozesse oder wie wir heute eher sagen würden: von Informationsverarbeitung) ist daher auch das Ziel der informatischen Konstruktionsprozesse.

Für die Modellierung – also vor allem die Schritte der Analyse und des Entwurfs – bedeutet das Folgendes: Ausgangspunkt ist eine bestimmte (Problem-) Situation. Diese muss zunächst hinsichtlich eines Zwecks abgegrenzt und verallgemeinert werden, denn (nur) verallgemeinerte und damit wiederkehrende Aspekte einer Situation lassen sich eher sinnvoll automatisieren, weil die Lösung dann auch wiederkehrend eingesetzt werden kann.

Diese Verallgemeinerung einer Situation bedeutet damit also auch, das dahinterliegende Konzept zu erkennen und zu verstehen. In der Modellierung entsteht damit ein Modell des generellen Ablaufs in der wirklichen Welt<sup>13</sup>.

Das kann man sich so vorstellen (vgl. Abbildung 3): Ausgehend von S entsteht ein Modell K. (S steht dabei für Situation, K für das verallgemeinerte Konzept.) Diese Analyse der Situation ist dabei eine zielgerichtete und zweckintentionale Abstraktion auf das Wesentliche. Was wesentlich ist, hängt vom gewünschten

<sup>13</sup> Die wirkliche Welt schließt hierbei auch gewünschte, noch nicht existierende, sondern zu schaffende Objekte und Situationen ein.

Zweck der späteren Lösung ab und kann immer nur im sog. soziotechnischen Kontext entschieden werden – da die Zwecke von und für Menschen entstehen. Daher findet auch Analyse und Abstraktion im Zusammenhang mit der Entwicklung eines Informatiksystems in einem Kontext von Auftraggeberinnen und Auftraggebern, Auftragnehmerinnen und Auftragnehmern, Benutzerinnen und Benutzern statt und ist nicht neutral oder wertfrei zu verstehen, sondern intentional.

Die Modellierung S->K bedeutet: Abstrahieren. Denn Abstrahieren ist:

a) verallgemeinern, indem von unwichtigen Details der jeweils unterschiedlichen Situation auf die entscheidende Gemeinsamkeit reduziert wird.

Gleichzeitig bedeutet abstrahieren

b) das Aufbewahren der wichtigen Aspekte hinsichtlich des Zwecks der Modellierung

und zudem auch

c) das Erkennen der Gemeinsamkeiten auf einem höheren konzeptionellen Niveau: sozusagen aus lauter Bäumen den Wald erkennen.

**Abbildung 3.** Definition von Abstraktion

Diese Schritte sind in der Abbildung 2 im linken Bereich angesiedelt. In der Programmierung bezeichnet man diese Schritte als Analysephase.

Anhand der Analyse kann eine Lösung entworfen werden (in Abbildung 4 der Schritt von K nach K'). In diesem Schritt werden das Modell oder Teile daraus mathematisiert und formalisiert. Das ist der entscheidende Vorgang in der informatischen Konstruktion: In der digitalen Welt, im Informatiksystem müssen alle Schritte, die automatisiert ablaufen sollen, eindeutig bestimmt und beschrieben sein.

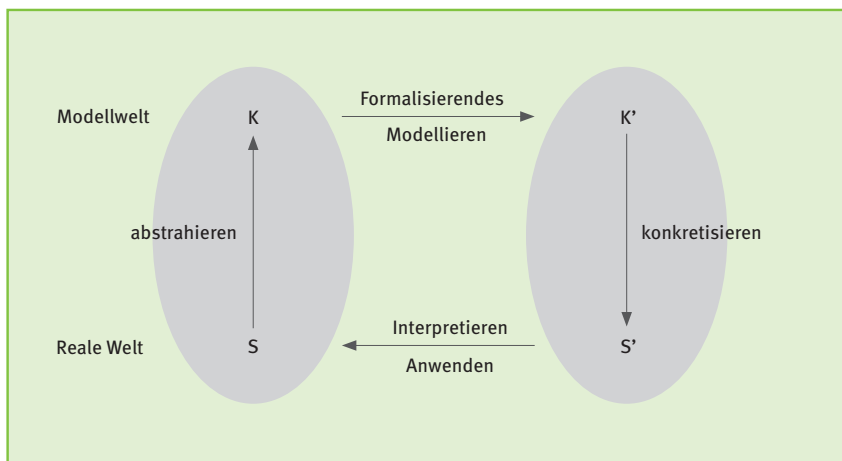
Dieses Modell kann dann implementiert werden, so dass eine Maschine es automatisiert ausführen und eine Benutzerin bzw. ein Benutzer es anwenden kann (siehe Abbildung 2).

Das Anwenden der Lösung erfolgt dann wieder situationsspezifisch, also beispielsweise mit konkreten Eingabedaten (in Abbildung 4 der Schritt K' nach S').

Über die Anwendung des implementierten Modells können nun beispielsweise Erkenntnisse über die reale Welt gewonnen werden (Simulation).

Wie komplex insgesamt ein solcher Vorgang ist, kann am Beispiel ‚Kalender‘ gezeigt werden: Soll das Datum einen Tag weiter geschaltet werden, kann dies mit folgender Eingabe geschehen: Tag  $n \rightarrow$  Tag  $n+1$ . Dieser Schritt erscheint zunächst einfach. Doch am Monatsende wird die Formel bereits komplizierter: Tag  $n+1$  genügt hier nicht mehr, sondern muss um Monat  $+1$  erweitert werden. Am Jahresende, in Schaltjahren oder bei unterschiedlicher Monatslänge müssen jeweils weitere Informationen hinzugefügt werden (ausführlich zu diesem Beispiel: Caspersen & Kolling, 2009). Das Beispiel zeigt, dass für eine informatische Lösung alle Schritte und Entscheidungen berücksichtigt und eindeutig sein müssen. Es wird daher sogar bei einfachen Beispielen schnell recht komplex und schwierig, den Überblick über die einzelnen Schritte und Entscheidungen zu behalten.

Abbildung 4 fasst die Konstruktionsschritte überblicksartig zusammen:



**Abbildung 4.** Informatisches Modellieren (vgl. Humbert & Puhlmann, 2004, S. 71; Schulte, 2003, S. 45, 72)

Zusammenfassend ist der informatische Konstruktionsprozess also wie folgt aufgebaut: Aus einer Situation oder einem einzelnen Phänomen (S) der realen Welt wird ein Modell abstrahiert, welches dieses als Konzept (K) beschreibt. Das Modell wird dann mathematisiert und formalisiert in die informatische bzw. die Modellwelt übertragen (K'). Das Modell K' kann dann konkretisiert werden, indem aus dessen abstrakter Beschreibung (der sog. Spezifikation) ein konkretes Programm erstellt wird, das auf einem Computer ausgeführt werden kann. Die Anwendung des Programms wirkt dann wiederum in der realen Welt.

Entscheidend ist nun der Unterschied zwischen der realen Welt und der Modellwelt. In der realen Welt ist intuitiv meist klar, worum es sich bei einem Gegenstand oder einer Erscheinung handelt, doch die genaue Grenzziehung fällt oft schwer: Wo fängt beispielsweise der Berg an, und wo hört das Tal auf? Idealtypen zu beschreiben ist meist leicht, aber es gibt immer wieder Unschärfen, z. B.: Ab welcher Anzahl nennen wir eine Ansammlung von Bäumen Wald? In der Modellwelt bzw. der informatischen Welt müssen alle diese Unschärfen auf eindeutige Aussagen bezogen werden können. In dieser Sichtweise lassen sich alle einzelnen Situationen und Phänomene auf ein eindeutiges Konzept zurückführen. Dieses Zurückführen ist Grundvoraussetzung für Digitalisierung und Automation. Peter Scheffe beschreibt diesen Sachverhalt aus der Perspektive der Softwarekonstrukteure wie folgt: „Das grundlegende Dilemma der Softwaretechnik ist, Nicht-Formalisierbares formal rekonstruieren zu müssen“ (Scheffe, 1999, S. 122). In der informatischen Konstruktion wird also eine Eindeutigkeit hergestellt, die nicht immer so gegeben ist. In der Entwicklung werden hier Entscheidungen getroffen, die oft aus rein softwaretechnischer Sicht relativ egal sind – etwa wie hoch der Wert  $n$  sein muss, um von Wald zu sprechen. Aus Anwenderperspektive kann das durchaus sehr relevant sein, etwa dann, wenn Eigentümerinnen und Eigentümer für ‚Wald‘ einen anderen Steuersatz als für eine Ansammlung von Bäumen bezahlen müssen.

Die Entscheidung hängt also von äußeren Faktoren, vom Einsatzzweck ab oder wird einfach während der Entwicklung festgelegt. Der wichtigste Aspekt ist, dass diese Entscheidungen normativ sind und wirken. Sie definieren, was später in der Anwendung gilt und damit auch wiederum die Ausgangssituation in der Wirklichkeit ( $S$  wird zu  $S'$ ) prägt.

Informatische Lösungen stellen also immer eine Eindeutigkeit her, die es ggf. vorher so nie gegeben hat. Und: Sie tun dies immer zweckbezogen, und dieser Einsatzzweck bedingt eine weitere wichtige Konsequenz: Eine Veränderung der Wirklichkeit, eine Wirkung ist immer beabsichtigt. Die automatische Ausführung soll etwas neu schaffen oder ersetzen – und damit wird das digitale Artefakt im Einsatz Teil der zuvor analysierten Wirklichkeit (in Abbildung 4 der Schritt von  $S'$  nach  $S$ ).

Allerdings greift das Modell (Abbildung 4) an entscheidender Stelle zu kurz: Die Lösung selbst bleibt nicht in der Modellwelt, sondern wird Teil der realen Welt und verändert die Ausgangssituation  $S$ . Die Lösung kommt also prinzipiell immer zu spät: Wenn sie eingesetzt wird, gibt es die Situation gar nicht mehr, für die sie einmal entwickelt wurde. Das treibt einen Kreislauf aus sich verändernden Softwareversionen an. Auch kommerziell vertriebene Produkte werden zumeist nicht nur mit Namen, sondern mit Name und Versionsnummer angeboten, und in rascher Folge kann man neue Versionen erwerben. Ein einzelnes Softwareprodukt

lässt sich daher immer nur dann verstehen, wenn man die Reihe der Vorgänger und der möglichen Nachfolger mit bedenkt. Verallgemeinert gilt: Die (Weiter-)Entwicklung von Programmen bzw. digitalen Systemen und Infrastrukturen erfolgt in einer Ko-Evolution: Auf der einen Seite ausgelöst von sich verändernden Bedingungen und neuen Ideen im Einsatzkontext, der sozialen Seite, und auf der anderen Seite ebenso getrieben von innertechnischen Anforderungen und Weiterentwicklungen. Digitale Systeme bzw. Informatiksysteme werden daher zunehmend als soziotechnische Systeme aufgefasst (Magenheim, 2000).

Mit der zunehmenden Bedeutung der Informatik in verschiedenen Alltagskontexten wird auch deutlicher, dass die rein innertechnische Sichtweise nicht ausreicht, um nützliche Systeme zu entwickeln. Daher hat sich auch die theoretische Sichtweise auf die Gegenstände bzw. den Gegenstandsbereich der Informatik geändert. Früher konnte man noch eher davon ausgehen, dass eindeutig und sauber definierte Anforderungen an die Konstruktion von Systemen gestellt werden können und diese Anforderungen sozusagen von außen kommen. Die Informatik selbst konzentrierte sich auf die technische Umsetzung dieser Anforderungen – den Weg von K nach K', und damit auch auf die Softwareentwicklung bzw. das Programmieren und den Aspekt der Implementation. Sie beschäftigte sich mit der dann als zweckfrei konzeptualisierten Konstruktionsweise informationsverarbeitender Technologien. Ihre Welt war eine mathematisch beschreibbare Welt, definiert durch die Möglichkeiten und Grenzen von Computern, die als rein mathematisches Modell vollumfänglich beschrieben werden konnten (das ist Grundlage der mathematischen Sichtweise bzw. Tradition der Informatik).

In dieser Sichtweise konnten Informatiksysteme mit dem Paradigma von Eingabe-Verarbeitung-Ausgabe betrachtet werden, mit dem Fokus auf dem mittleren Bereich: der Automatisierung (daher auch die früh geprägte Bezeichnung der wissenschaftlichen Disziplin als Informatik; siehe oben). Mittlerweile sind Informatiksysteme aber nicht mehr ausreichend durch diese Perspektive der Automatisierung erfasst, da die Interaktion mit verteilten digitalen Infrastrukturen wie z. B. sozialen Netzwerken, Flugbuchungssystemen etc. viel komplexere Prozesse hervorbringt.

Peter Wegner fasst diese Prozesse mit dem Begriff der **Interaktion** (Wegner, 1997) zusammen. Während auf der rein bzw. eng verstandenen technischen Seite solcher Systeme tatsächlich ‚nur‘ algorithmische Abläufe zu beobachten sind, die mit den entsprechenden mathematischen Beschreibungsmöglichkeiten erfasst werden können, entsteht das Verhalten des Gesamtsystems durch parallele Eingaben von ggf. Millionen von menschlichen Nutzerinnen und Nutzern. Damit aber geht das soziotechnische Gesamtsystem in seinen Handlungsmöglichkeiten über die vorab berechenbaren möglichen Zustände hinaus.

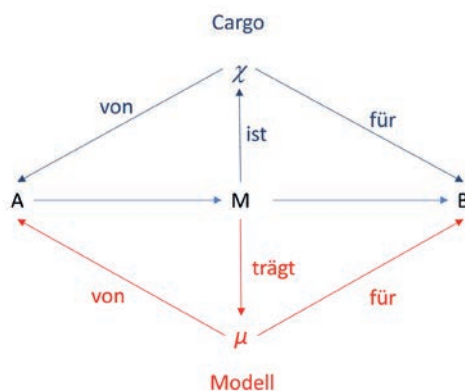


Insgesamt zeigt sich, dass am informatischen Konstruktionsprozess das Wesen der Informatik bestimmt werden kann – und dass die Informatik viel mit Modellketten unterschiedlicher Abstraktionsgrade und Modellierung zu tun hat. Bernd Mahr schlussfolgert sogar: „Wie immer man jedoch die Wissenschaft der Informatik betrachtet, immer spielen dabei Modelle eine dominierende Rolle“ (Mahr, 2009, S. 228).

Und weiter:

„Wie eng die Informatik in Praxis und Wissenschaft mit Modellen verbunden ist, wird auch deutlich, wenn man die Leitfrage betrachtet, die explizit oder implizit ihrer praktischen und wissenschaftlichen Arbeit zugrunde liegt. Diese für alle Ingenieursdisziplinen maßgebliche Leitfrage setzt in der Informatik den situativen Kontext einer informations-technischen Systementwicklung voraus und lautet: Erfüllt das System S die an dessen Anwendung gestellten Anforderungen? Es gibt in der Informatik wohl keine ernstzunehmende Aktivität, die nicht in irgendeiner Weise im Rahmen einer Systementwicklung steht, sei es, dass diese Entwicklung nur allgemein gedacht wird oder aber konkret gegeben ist. Und um ernst genommen zu werden, muss diese Aktivität, direkt oder indirekt, zur Beantwortung der damit aufgeworfenen Leitfrage beitragen“ (Mahr, 2009, S. 229).

Im Weiteren analysiert Mahr nun die Modellierung und die Rolle von Modellen. Wir hatten oben in Abbildung 4 den Modellierungsprozess beschrieben. Mittels der Analyse von Mahr kann das ergänzt werden (siehe Abbildung 5):



**Abbildung 5.** Struktur der Modellbeziehungen, durch die ein Gegenstand M als Modell  $\mu$  zum Träger eines Cargos  $\chi$  wird (nach Mahr, 2009, S. 336)

Demnach kann man einem Gegenstand nicht per se ansehen, ob er ein Modell ist, stattdessen erweist sich sein „Modellsein“ erst im Gebrauch (untere Hälfte der Grafik): Und zwar kann der Gegenstand M ein Modell von A sein oder (ggf. ohne Bezug zu einem tatsächlich vorhandenen A) ein Modell für B (oder beides). Entscheidend für die neue Sichtweise von Mahr ist nun die Funktion des Modellseins (obere Hälfte der Grafik): Ein Gegenstand wird (nur) dann als Modell benutzt, wenn er etwas transportiert, also eine Information trägt, die anders nicht oder nur schlecht übertragen werden kann. Er nennt diese Information Cargo. Wenn in der Informatik bzw. in der Konstruktion von Abstraktion gesprochen wird und diese als schwierig gilt, dann kann man anhand dieses Modells nun aufzeigen, dass die Schwierigkeit im Finden des passenden Cargos liegt. Damit wird nochmal klar, dass Modellieren immer nur in Bezug auf die Zwecke und die soziotechnische Einbettung der Konstruktion deutlich wird.

(Das Modell von Mahr ist auch deshalb interessant, weil es die Rolle der Informatik bzw. das Modellieren in der Informatik im Vergleich zu anderen Disziplinen abgrenzt – und weil es in der naturwissenschaftsdidaktischen Forschung, etwa in der Erarbeitung der „Modellkompetenz“ in Biologie und Physik, benutzt wird.)

## 1.4 Gemeinsamkeiten und Unterschiede der Informatik im Vergleich ...

### 1.4.1 ... zu den Naturwissenschaften

Auf den ersten Blick lässt sich der Unterschied der Informatik zu den Naturwissenschaften einfach fassen: Während die Naturwissenschaften die uns umgebende natürliche, d. h. von sich aus vorgegebene Umwelt – die Natur – untersuchen und erklären wollen, beschäftigt sich die Informatik mit dem Künstlichen: den technischen Systemen und Verfahren der automatisierten Informationsverarbeitung. Informatik und Technik erforschen die Beschaffenheit und das Gestalten künstlicher Gegenstände, die in diesem Zusammenhang Artefakte genannt werden.

Etwas zugespitzt könnte man sagen: Naturwissenschaft beschäftigt sich mit gegebenen Phänomenen oder Gegenständen, Informatik und Technik mit künstlichen bzw. von Menschen geschaffenen Gegenständen.

In beiden Fällen können also die Beschaffenheit, die Eigenschaften dieser Gegenstände untersucht werden: ihr innerer Aufbau, ihre Funktionsweise bzw. ihre Wirkprinzipien etc. – die Frage nach der **Struktur** des Gegenstands. Es gibt hier jedoch zwei entscheidende Unterschiede:

- 1) Die Informatik stellt als konstruierende Wissenschaft diese Frage vor allem aus der Perspektive des Entwicklers und Konstrukteurs: Die Antwort ist daher im

Grunde so etwas wie eine Konstruktionsbeschreibung. Die Naturwissenschaft stellt diese Frage, um zu verstehen und zu erklären. In Bildungszusammenhängen sollten digitale Artefakte aber angesichts der zunehmenden Digitalisierung aller Lebensbereiche – die von den Kindern ja ebenso wie die natürliche Umwelt als eine schon gegebene vorgefunden wird – u. E. ebenfalls unter dieser ‚naturwissenschaftlich-verstehenden‘ Perspektive untersucht werden. In dieser Sichtweise werden dann ‚informatische Phänomene‘ analog zu natürlichen Phänomenen untersucht. Hier wird informatische Bildung – insbesondere für junge Kinder – in Zukunft vermutlich noch stärker als fachdidaktische Forschungsaufgabe solche erklärenden Modelle für wichtige digitale Artefakte entwickeln müssen, da eine fachwissenschaftliche Erklärung für eine Konstrukteurin und einen Konstrukteur nicht unbedingt das geeignete Verstehensmodell darstellt.

- 2) Die Untersuchung der **Struktur** alleine greift jedoch entscheidend zu kurz und verkennt, dass ja nicht naturgegebene Phänomene untersucht werden, deren Existenz und deren Struktur als vorgegeben akzeptiert werden müssen. Eine solche weitreichende Übertragung würde bedeuten, was Kritikerinnen und Kritiker manchmal der informatischen Bildung vorwerfen: eine Anpassung der Kinder an die digitale Welt und deren Struktur. Daher muss immer auch das Gestaltete, das Gemachte mit einbezogen werden, wenn digitale Artefakte/Phänomene untersucht werden sollen. Deren Struktur ist eben nicht natürlich entstanden, sondern mit einer bestimmten Absicht so konstruiert worden. Somit muss immer zugleich die **Funktion** bzw. Intention des Artefakts berücksichtigt werden (Schulte, 2008a, 2008b). Spannend ist nämlich, dass dieselbe Funktion zumeist auch durch eine alternative Struktur erreicht werden könnte.

Die Informatik beschäftigt sich also nicht nur damit, wie die Welt ist, sondern auch damit, wie sie sein könnte – so entstehen unweigerlich gesellschaftsbezogene, d. h. ethische, moralische, normative Fragen. Informatik ist damit im Grunde näher an der Technik als an der Natur.

#### 1.4.2 ... zur Technik

In der Technikexpertise für die Stiftung „Haus der kleinen Forscher“ werden die folgenden Merkmale für Technik und Tech-



nikwissenschaft herausgestellt, die (zunächst) genau wie Merkmale der Informatik klingen – insbesondere in der oben vertretenen konstruierenden Sichtweise. Die Merkmale sind: „die Gestaltungsoffenheit der Technik, die Wertgebundenheit der Technik, der diskursive Charakter der Entscheidung über die Gestaltung eines konkreten Artefakts, die Bewertung der Angemessenheit technischer Lösungen, die kulturprägende Wirkung“ (Kosack, Jeretin-Kopf & Wiesmüller, 2015, S. 39). In der Auflistung lässt sich ohne Weiteres der Begriff Technik durch Informatik ersetzen. Nur, wo liegt dann der Unterschied in den Wissenschaften und der spezifische Beitrag der informatischen Bildung? Diese Unterschiede werden in der Konkretisierung deutlich; denn weiter heißt es in der Technikexpertise:

„Unterricht über Technik muss enthalten:

- Aspekte der naturalen Wirkungszusammenhänge, die in jedem spezifischen Artefakt wirksam werden,
- Aspekte der Gestaltungsoffenheit konkreter Problemlösungen im Verbund mit gegenständlicher Tätigkeit sowie
- Aspekte der Bewertung von Artefakten im Hinblick auf Funktionserfüllung und Nebenwirkungen und das Einbetten in einen kulturellen Zusammenhang.

Diese Aspekte müssen inhaltlich konkretisiert werden, z. B.:

- Naturale Wirkungszusammenhänge müssen beim Einsatz von Material und Werkzeug erkundet werden können insofern sie für den Zweck des Artefakts von Bedeutung sein könnten. Der Unterricht muss Raum geben für die Erfahrung grundlegender Gesetze wie dem Hebelgesetz, dem Hook'schen Gesetz, dem Ohm'schen Gesetz und ähnlichem, ohne dass die Gesetze mathematisch formuliert werden müssen. Qualitative Je-desto Beziehungen sind – besonders in den unteren Altersstufen – völlig ausreichend“ (Kosack et al., 2015, S. 39).

Die „Naturalen Wirkungszusammenhänge“ des „Materials“ sehen bei digitalen – im Gegensatz zu technischen – Artefakten deutlich anders aus. Insbesondere muss das abstrakte Material „Software“ überhaupt erst fassbar gemacht werden. Dazu sehen wir uns den Schritt aus dem Schema (siehe Abbildung 4) von  $K \rightarrow K'$  noch einmal genauer an: Das aus der Analyse und der Abstraktion entstandene Modell (K) wurde durch Formalisierung und Mathematisierung eindeutig gemacht, so hatten wir oben formuliert. Doch welche „Materialität“ des Baustoffs zwingt zu dieser Art der Formalisierung und Eindeutigkeit? Es liegt an der Art, wie Informatiksysteme Daten verarbeiten: in einzelnen, eindeutig festgelegten Schritten,

also beispielsweise: Jetzt fängt das Tal an, also gilt X. X könnte definiert sein als: Wir befinden uns auf einer Brücke. In der technischen Konstruktion einer Brücke gibt es dagegen zumeist mehr Spielräume, wo genau sie beginnt. Auch die Sicherheit einer Brücke kann technisch anders garantiert werden: Wenn die Tragkraft vielleicht doch nicht reicht, dann kann man etwa die Materialstärke der Brückenpfeiler etwas erhöhen. Das geht so in der Informatik nicht: Hier gibt es keinen zusätzlichen Gewinn, wenn man etwa eine Fallunterscheidung ein zweites Mal in ein Programm einbaut, um diese doppelt prüfen zu lassen. Ein Informatiksystem befindet sich immer in genau einem Zustand: Entweder hat die Prüfung Zustand X oder Zustand Y ergeben – eine zweite Prüfung verbessert nicht die „Tragfähigkeit“ des Ergebnisses.

Dieses Zustandskonzept bewirkt einen weiteren Unterschied in der „Materialität“: In „natürlichen“ Wirkungszusammenhängen haben kleine Abweichungen zumeist kleine Auswirkungen: Wenn etwa ein Brückenpfeiler ein klein wenig dünner ausgefallen ist, macht das nicht viel an der gesamten Stabilität der Konstruktion. Wenn aber in einem Informatiksystem der aktuelle Zustand „ein klein wenig“ abweicht, dann handelt es sich schlicht um einen anderen Zustand – und das kann unabhängig vom ‚Ausgangszustand‘ ein fataler Fehlerzustand sein. Das übliche Gesetz, dass kleine Abweichungen üblicherweise kleine Auswirkungen haben, gilt hier nicht.

Daher gibt es auch keine „Analogieschlüsse“ in der digitalen „Materialität“: In natürlichen Materialien verhalten sich ähnliche Zusammensetzungen zumeist auch ähnlich – in digitalen Systemen können ähnliche Anfangszustände eines Systems zu komplett anderen Zielzuständen führen.

Keil-Slawik (1994) nennt weitere Besonderheiten der Materialität von Software und schlussfolgert, dass Software weniger als ein „Produkt“, sondern eher als eine „Ansammlung von Plänen“ betrachtet werden sollte: Im Programmtext legen die Entwickler fest, „was zur Laufzeit des Programms wie, in welcher Reihenfolge und unter welchen Umgebungsbedingungen ausgeführt werden soll“ (Keil-Slawik, 1994, S. 4). Dabei kommt es darauf an, „alle Sonderfälle und Ausnahmbedingungen vollständig zu erfassen und ordnungsgemäß zu verarbeiten“ (Keil-Slawik, 1994, S. 5). Programmieren bzw. Softwareentwicklung kann daher nach Keil, unter Rückgriff auf Peter Naur (1985), als Theoriebildung aufgefasst werden: Softwareentwicklung sollte nicht(!)

„in erster Linie als Produktion von Programmen und zugehörigen Texten betrachtet werden sondern als ein Prozeß bei dem die Programmierer eine Theorie darüber entwickeln wie die vorhandenen Probleme durch die Programmausführung gelöst werden können. Da aber nicht alle bei der Systementwicklung auftretenden Probleme und Entscheidungen mit

all ihren Wechselbezügen dokumentiert werden können ist diese Theorie nur in den Köpfen der Entwickler vorhanden. Das Wiederherstellen der Theorie lediglich aufgrund der Dokumentation ist gänzlich unmöglich“ (Keil-Slawik, 1994).

Mittermeir fasst diesen Unterschied in der Materialität elegant zusammen und betont damit einen weiteren, für den Einstieg in informatische Bildung sicherlich sehr interessanten Aspekt:

„Im Unterschied zu anderen technischen Fächern konstruieren wir in der Informatik nicht durch physische Materialbearbeitung, sondern durch sprachliche Formulierung“ (Mittermeir, 2010, S. 59).

### 1.4.3 ... zur Mathematik

Formal hat die Informatik viel mit der Mathematik zu tun: An den Hochschulen sind Mathematik und Informatik oft in einem gemeinsamen Fachbereich zusammengelegt. Der Begriff Informatik wird (obwohl historisch falsch) oft als zusammengesetzt aus Information + Mathematik verstanden, das Konzept des Algorithmus stammt aus der Mathematik und ist in beiden Disziplinen sehr wichtig. Die GI-Standards für die Sekundarstufe I sind den amerikanischen Mathematikstandards nachempfunden. Das Konzept der Modellierung (Modellierung und Implementation anstelle von Programmierung) findet sich ebenfalls auch in der Mathematik.

Aus dieser Perspektive fällt es schwer, Informatik von Mathematik deutlich abzugrenzen. In beiden Disziplinen wird in dieser Sichtweise der Problemlöseprozess zentral gesetzt, der im Wesentlichen disziplinspezifisch als Besonderheit hat, dass die Problemlösung ein formales Modell darstellt. Aus dieser (engen) Wahrnehmung auf Informatik argumentierte beispielsweise Eberle wie folgt:

„Dabei [= im Problemlöseprozess] ist zu unterscheiden zwischen der reinen Übersetzung von Umgangssprachen in formale Schreibweise (das Niveau kann variieren) und dem antizipativen Verständnis der zeitlichen Abläufe (Prozeduren), während deren sich die Inhalte von Variablen ändern (z. B. Iterationen). Letzteres erfordert eine zusätzliche kognitive Leistung und unterscheidet auch informatisches Denken von großen Teilen des mathematischen Denkens (zumindest bezüglich der Mathematik auf der Sekundarstufe II), das statische Beziehungen formalisiert“ (Eberle, 1996, S. 329).

Diese Argumentation bezieht sich auf eine recht feinsinnige Unterscheidung eher statischer und eher dynamischer Aspekte der Algorithmik, die den Informatikunterricht von der Mathematik absetzen (vgl. Claus, 1977).

Informatische Bildung wurde dementsprechend, etwa in Form der allerersten didaktischen Ansätze (Algorithmenorientierung) und der rein mathematischen Tradition der Informatik, als zweckfreie ‚innertechnologische‘ Betrachtung der wesentlichen Konstruktionsbedingungen ohne Bezug zur Lebenswirklichkeit gesehen. Dabei gingen dann aber wichtige Aspekte wie Interaktion (vgl. „program or be programmed“ (Rushkoff, 2010)), Wechselwirkungen und Ko-Evolution von Handeln, individueller und gesellschaftlicher Entfaltung und Entwicklung im Zusammenspiel mit technischer Infrastruktur und deren Veränderungen verloren. Ebenso fehlte das systemkonstruierende Element. Informatik wirft aber immer – wie Technik – die Frage auf: Wie wollen wir leben?

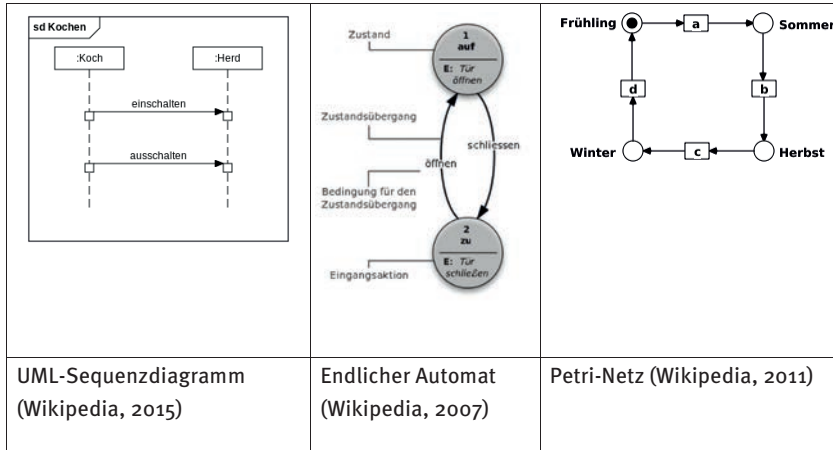
Die Mathematik ist sozusagen neutral bezüglich des Ziels der Formalisierung und Mathematisierung. In der Informatik geht es jedoch immer um die automatische Verarbeitung: um die (potenzielle) Implementierung und Nutzung – und damit immer auch um die Rückbezüglichkeit der Lösung:  $S \rightarrow S'$  in der Abbildung 4: die Implementierung (genauer: die Nutzung und Interaktion mit) der Lösung in der Ausgangssituation, die diese selbst verändert. Das konstruierte System wird selbst Teil der Problemsituation, wenn es dort eingesetzt wird und verändert damit die Ausgangslage entscheidend. Die mathematische Lösung kann als Anwenden im Sinne eines Beschreibens aufgefasst werden, informatische Anwendung bedeutet, ein Informatiksystem einzusetzen, dass es vorher innerhalb der Ausgangssituation so nicht gegeben hat.

Aber auch bezüglich der rein formal-abstrakten Lösung selbst gibt es Unterschiede – da die physikalischen Eigenschaften der Informatiksysteme berücksichtigt werden müssen, mit denen die Lösung ausgeführt werden kann: Fragen nach benötigtem Speicherplatz und Zeit für das Ausführen der notwendigen Berechnungsschritte verändern den Charakter der Lösung.

Zudem ist die Konstruktion einer Lösung weitaus mehr als die Konstruktion eines Automatismus, denn dieser muss auch benutzt, geprüft und gewartet werden können, d. h., *Benutzbarkeit*, *Wartbarkeit* spielen eine Rolle – ebenso *Erweiterbarkeit* und *Anpassbarkeit* (kursiv gedruckte Konzepte sind Kriterien für Softwarequalität (vgl. Wikipedia, 2016)).

Es gibt einen weiteren wichtigen Unterschied, der unserer Meinung nach entscheidend ist: Die formale Problemlösung wird in der Mathematik „mathematisch“ also in der (einen) Sprache der Mathematik dargestellt. Informatik bedient sich mathematischer Ausdrucksweisen und Darstellungen – zeichnet sich aber vor allem dadurch aus, stets neue formale Sprachen und Darstellungen zu entwickeln, die auf Automaten ausführbar sind bzw. zu deren Beschreibung genutzt

werden können. Diese Sprachen sind interessanterweise oft auch einfache grafische Notationen (vgl. Abbildung 6; ausführlicher dazu Hubwieser, 2007; Hubwieser & Broy, 1997).



**Abbildung 6.** Beispiele für grafische Notationen

Diese Sprachen werden oft als Modellierungssprachen bezeichnet, da mit ihnen das Ergebnis eines Modellierungsprozesses dargestellt wird und sie einen Zwischenschritt zur Implementierung in einer maschinennäheren Programmiersprache bilden.

## 1.5 Informatik und informatische Bildung

Nachdem Informatik als Disziplin eingeführt und zu anderen Wissenschaften abgegrenzt worden ist, kann nun die informatische Bildung charakterisiert werden. Dazu beginnen wir mit den Ergebnissen einer Studie, die von den beiden großen Informatik-Vereinigungen ACM und Informatics Europe in Auftrag gegeben wurde, um die Rolle der Informatik in der Bildung zu klären. Ein Ergebnis des Reports ist die Bekräftigung der folgenden Unterscheidung von „digital skills“ (Fertigkeiten im Umgang mit digitalen Geräten) und der Einführung in die Wissenschaft Informatik. Eine ähnliche Unterteilung findet sich auch im Report „Shut down or restart“ (The Royal Society, 2012) aus Großbritannien. Die sich daraus ergebende Unterscheidung von Computer Science, ICT und digital literacy (DL) wird in Abschnitt 2.4.1 vorgestellt.

Generell kann man sagen, dass in fast allen Curricula allgemeinbildender Schulen weltweit jeweils sowohl Anwendungsfertigkeiten als auch Konzeptwissen vermittelt wird.



Fertigkeiten im Umgang/Anwendungskennnisse beziehen sich auf die Nutzung vorhandener Systeme: „Any citizen of a modern country needs the skills to use IT and its devices intelligently. These skills, the modern complement to traditional language literacy in language (reading and writing) and basic mathematics, are called *digital literacy*“ (Gander et al., 2013, S. 7). Der Report nennt als Beispiele für solche Fertigkeiten z. B. das Tippen, das Erstellen und Überarbeiten von Dokumenten, Suchen, Ablegen und Wiederfinden von Daten, den ethischen und sicheren Umgang mit Daten, die Auswahl geeigneter Informatiksysteme u. Ä. (Gander et al., 2013, S. 8). Davon abgegrenzt werden Informatik und informatische Bildung mit eigenen Themen und Inhalten wie beispielsweise Algorithmen, Datenstrukturen und Abstraktion (Gander et al., 2013, S. 9). Die Beiträge der informatischen Bildung fasst der Report vor allem als Förderung von Kreativität, der Konstruktion von Artefakten, dem Umgang mit Komplexität und der Genauigkeit auf (Gander et al., 2013, S. 13).

In diesem Sinne aufgefasst wäre digitale Bildung ein Nebeneinander aus digital literacy und informatischer Bildung. Diese strikte Unterscheidung von Nutzungsfertigkeiten einerseits und informatischer Bildung andererseits hat unterschiedliche Ursachen. Eine ist sicherlich die im Report angedeutete Angst, informatische Bildung auf Umgangsfertigkeiten zu reduzieren. Eine andere Ursache ist der Wunsch, diesen neuen Bildungsbereich eng an die – auch noch recht junge – akademische Disziplin zu binden. Im deutschsprachigen Raum wird das oft mit der Aussage verknüpft, die Disziplin heiße ja auch Informatik und nicht Computerwissenschaft, habe also mithin wenig bis gar nichts mit dem Artefakt Computer zu tun.

Mittermeir argumentiert beispielsweise: „Informatik ist ein Kunstwort aus der Verschmelzung von Information und Automatik. Der Wortstamm Computer kommt dabei nicht vor. Das bedeutet, dass Informatik, obzwar ein technisches Fach, so doch kein gerätespezifisches Fach sein sollte“ (Mittermeir, 2010, S. 72). Problematisch ist es nämlich, wenn der Informatikunterricht auf Computernutzung reduziert wird und so ein Zerrbild der Disziplin vermittelt (Mittermeir, 2010, S. 55). In der Informatik gehe es „doch eigentlich nicht um den Computer, sondern um konstruierte (also technische) Systeme, die es erlauben, Daten (im weitesten Sinne) so zu interpretieren, dass Aktionen bewirkt werden“ (Mittermeir, 2010, S. 57).

Rechenberg widerspricht dieser Sichtweise: Es habe sich gezeigt, dass „die Informatik sich nicht mit Informationsprozessen in Gesellschaft und Natur beschäftigt“, daher müsse man „darauf beharren, dass Informatik heute die Wissenschaft und Technik vom Computer und seinen Anwendungen ist, also Computerwissenschaft“ (Rechenberg, 2010, S. 47). Interessanterweise ähneln sich aber die curricularen Vorschläge der beiden mehr, als dass sie sich unterscheiden. Dies ist kein Zufall, so stellt Brandhofer fest, dass die strikte Trennung von Infor-

matik und Computer(anwendung) in der Praxis und in der curricularen Diskussion wenig beachtet wird (Brandhofer, 2014, S. 3).

Unserer Meinung nach ist es sinnvoller, vom Zusammenhang der beiden vermeintlich getrennten Bereiche auszugehen. Eine Möglichkeit dazu wurde von Puhlmann und Humbert vorgeschlagen. Sie gehen von informatischen Phänomenen aus und beschreiben diese in drei Klassen (Humbert & Puhlmann, 2004):

1. Mit Informatiksystemen direkt verknüpfte Phänomene: Sie treten beim Nutzen auf und können dazu beitragen, dass Anwenden des Systems leichter, effizienter oder auch angenehmer zu machen.
2. Mit Informatiksystemen indirekt verbundene Phänomene: Sie sind nicht sofort ersichtlich, sondern werden erst bei der Analyse der Interaktion mit dem System deutlich oder wahrnehmbar.
3. Von Informatiksystemen unabhängige Phänomene: Sie sind unabhängig von digitalen Systemen und zeichnen sich dadurch aus, dass sie eine inhärente informatische Struktur aufweisen und/oder informatisches Denken nahelegen. Beispiele sind etwa Phänomene, bei denen Such- oder Sortierprozesse eine Rolle spielen.

Bei Humbert und Puhlmann ist die Nutzung bzw. die Interaktion mit einem Informatiksystem vorrangig als Auslöser oder Weg zur Wahrnehmung informatischer Phänomene gedacht, die anschließend im Mittelpunkt der Lernprozesse stehen sollen. Allerdings schreiben die beiden auch, dass die Relevanz bzw. der Bildungswert eben gerade im Rückbezug bzw. in der Anwendung auf die Interaktion besteht. Dieser Ansatz hält die Frage in der Schwebe, ob Informatik als Computerwissenschaft aufgefasst werden darf oder soll. Dementsprechend vage definieren die Autoren Informatik als die Wissenschaft, die „die Konstruktion und Gestaltung von Informatiksystemen adressiert“<sup>14</sup> (Humbert & Puhlmann, 2004, S. 75).

Interessant für diese Expertise ist der festgestellte Zusammenhang von informatischer Bildung und „Anwendungsfertigkeiten“. Als Ziel und Begründung für informatische Bildung wird immer wieder die positive Wirkung auf die Fertigkeiten im Umgang mit Informatiksystemen herangezogen (beispielsweise in Brandhofer, 2014; Gander et al., 2013; Humbert & Puhlmann, 2004; Mittermeir, 2010 u. a.) – diese Beziehung wird jedoch (zumeist) nur aus einem Blickwinkel gesehen: Während die informatischen Konzepte oder Phänomene zur Anwendungsfertigkeit

---

<sup>14</sup> Eigene Übersetzung. Im Original: „Informatics in this setting is the scientific discipline addressing the construction and design of informatics systems.“

beitragen, so wird teilweise (implizit) angenommen, dass die Anwendung selbst nicht zur Informatik als Disziplin gehöre und daher auch Anwendungsfertigkeiten nicht als Teil der Informatik oder informatischen Bildung gesehen werden sollten.

Allerdings ist die einfach und eindeutig scheinende Trennung zwischen Gestalten/Konstruieren als Teil der Informatik einerseits und dem Anwenden als Nicht-Informatik andererseits gar nicht so eindeutig und einfach (Crutzen, 2000). Fischer u. a. fordern ein entsprechendes Umdenken in der Softwaretechnik, ein Meta-Design, das Benutzerinnen und Benutzer zu Designerinnen und Designern werden lässt, die über die gesamte Lebensdauer eines digitalen Artefakts an dessen Adaption, Veränderung und Weiterentwicklung mit-gestalten, indem sie das jeweilige Artefakt an ihre eigenen Anforderungen anpassen und erweitern (können) – d. h. indem die Anwenderinnen und Anwender selbst zu Gestalterinnen und Gestaltern werden können (Fischer, Giaccardi, Ye, Sutcliffe & Mehandjiev, 2004). Insgesamt wird die postulierte Trennung von „design-time“ und „use-time“ mehr und mehr obsolet (Maceli & Atwood, 2011). Das gilt nicht nur auf Seiten der (ursprünglichen) Anwenderinnen und Anwender, sondern auch der Seite der Konstrukteurinnen und Konstrukteure: Im Prozess der Systemgestaltung und -entwicklung ist es Aufgabenbestandteil, bereits vorhandene Systeme zu untersuchen und ebenso die stets weiterentwickelten Werkzeuge zur Unterstützung der verschiedenen Aspekte der Softwareentwicklung kompetent anwenden zu können. Also, im Sinne Crutzens formuliert: Auch zum Entwickeln, d. h. zum Kern der Informatik, gehört das Anwenden und Nutzen als integraler Bestandteil dazu. Anwenden bzw. die Interaktion mit digitalen Artefakten bekommt in dieser Perspektive eine neue Bedeutung, die sich fundamental von der Idee des Nutzens der von anderen fertiggestellten Werkzeuge unterscheidet: Zur Interaktion zählen das Kennenlernen, Einarbeiten, Konfigurieren und insbesondere auch das Adaptieren, Anpassen und Erweitern dazu. Und die Interaktion gehört auch als integraler Bestandteil zur Konstruktion dazu, da diese nicht ohne Anwenden von Werkzeugen möglich ist.

Daraus ergibt sich ein weiterer wichtiger Aspekt: Interaktion schließt Explorieren ein: das Analysieren und Erkunden von digitalen Artefakten mit dem Ziel, diese zu verstehen. Das Verstehen schließt ein, die Möglichkeiten und Grenzen zur Adaption und Weiterentwicklung abschätzen zu können. Und das



wiederum geht nicht ohne Reflexion (in Bezug etwa auf den Einsatzzweck und mögliche Nebenwirkungen) und Bewertung. Verstehen meint hier insbesondere auch das Verstehen der Konstruktionsweise, denn diese ist es ja, die Veränderung erfährt, wenn das Artefakt erweitert wird.

Auch wenn die konkreten Konstruktionen und Konstruktionsweisen einer ständigen Weiterentwicklung unterworfen sind, so gibt es doch einen recht stabilen Kern an informatischen Konzepten, die in diesen konkreten Technologien sichtbar bzw. verkörpert werden und die unterrichtet werden können. Diese können als ‚Fundamentale Ideen‘ der Informatik bezeichnet werden (vgl. Schubert & Schwill, 2011).

Die Idee des *Computational Thinking* setzt einen anderen Akzent (Wing, 2006, 2008): Es baut auf dem Wissen der Informatik über informationsverarbeitende Prozesse auf und nutzt die Techniken, Modelle, Konzepte und Werkzeuge der Informatik für das problemlösende Denken. Im Kern sieht Wing dabei die Abstraktion, und das Denken in unterschiedlichen Abstraktionsschichten<sup>15</sup>. Es geht darum, zu unterscheiden, was jeweils zur Lösung eines Problems benötigt wird und was weggelassen werden kann. Komplexe Probleme werden dabei in Teile und Schichten zerlegt, so dass ein Teil auf ein anderes bzw. eine Schicht auf eine andere zurückgreifen kann. So muss beispielsweise ein Anwendungsprogrammierer nicht (immer) genau wissen, wie ein Befehl intern programmiert wurde, um ihn zu nutzen. Dies hat zwei wichtige Facetten: Erstens kommt es manchmal doch auf die Details der internen Funktionsweise an – immer dann, wenn es um Randbedingungen geht, etwa die Frage wie viel Speicherplatz etwas benötigt und ob der vorhandene dazu ausreicht. Zweitens ist es nach Wing beim Computational Thinking wichtig, die Beziehungen zwischen den Abstraktionsebenen im Auge zu behalten.

Dieses Denken in Abstraktionen kann glücklicherweise durch Automation unterstützt werden. Berechnen (das engl. Computing bedeutet berechnen) ist die „Automatisierung von Abstraktion“ (Wing, 2008). Computational Thinking beschäftigt sich mit der Frage, wie kann der Computer das Problem für mich lösen – für die Antwort bzw. eine gute Antwort müssen die richtigen Abstraktionen und der richtige „Computer“ gewählt werden (Wing, 2008). Wichtig ist, dass der Computer hier nicht unbedingt eine Maschine, sondern auch ein Mensch sein kann, der die „Berechnung“ durchführt (Wing, 2008).

Computational Thinking ist damit insgesamt nicht nur ein Ansatz oder eine Denkweise, sondern ein Ansatz, mit Problemen umzugehen, der informatische Konzepte und informatisches Wissen nutzt. Insofern kann man ihn mit dem Begriff informatisches Denken ins Deutsche übersetzen.

---

<sup>15</sup> Zum Begriff der Abstraktion siehe in Abbildung 3, S. 35.

Was dieses informatische Denken im Kern genau ist, ist schwer zu operationalisieren (National Research Council (U.S.) & Committee for the Workshops on Computational Thinking, 2010, 2011). In jedem Fall gehört dazu, informatische Konzepte zur Lösung von Problemen bzw. im Alltag anzuwenden.

Welche informatischen Konzepte sich eignen, lässt sich – unter der Annahme, die bisherigen fachdidaktischen Konzepte liegen zumindest nicht völlig daneben – aus der Entwicklungsgeschichte der informatischen Bildung zumindest teilweise ableiten. In den Anfängen der informatischen Bildung gab es – passend zu den Traditionslinien in der Informatik – die vorherrschende mathematische oder algorithmenorientierte Sichtweise der Turingmaschinen<sup>16</sup>, die sicherlich auch eine wesentliche Grundlage darstellt und die sich auf den Kernbereich der Algorithmen konzentriert (hat). Diese wurde jedoch schon recht früh (Ende der 1970er-, Anfang der 1980er-Jahre) durch die Anwendungsorientierung abgelöst, die nicht nur Algorithmen, sondern die Bedeutung von **Algorithmen** und **Automatisierung** für gesellschaftlich relevante Bereiche in den Mittelpunkt von Lernprozessen zu stellen versuchte. Dieser Ansatz zielte darauf, die mathematische Sichtweise der Wissenschaftsdisziplin mit der Betrachtung von gesellschaftlichen Auswirkungen zu vereinbaren (Schulte, 2001). In der Schulpraxis im gymnasialen Unterricht ist dieser Ansatz (vermutlich, es gibt unseres Wissens keine empirische Forschung dazu) immer noch weit verbreitet.

Davon abgegrenzt, teilweise auch durch politische Entscheidungen (van Lück, 1986), erfolgte die Einführung in **die Nutzung digitaler Systeme** als informationstechnische Grundbildung, die vor allem als Anwenden und Benutzen gedacht wurde und teilweise ebenso wie der Informatikunterricht, die **Wechselwirkungen zwischen Informatik und Gesellschaft** mit thematisieren wollte.

Die Einführung in das Nutzen von digitalen Systemen wird seitdem im schulischen Kontext teilweise als entgegengesetzt zur informatischen Bildung erlebt und verstanden: Als reine Benutzerschulung, die in das Anwenden vorgegebener Interaktionspfade einführt. Unter dem Titel „program or be programmed“ beschreibt Douglas Rushkoff (Rushkoff, 2010) diese Perspektive und zugleich eine Alternative: Entweder passen sich die Menschen an diese vorgegebenen Pfade an oder sie lernen, diese Interaktionspfade selbst zu gestalten.

Informatik-Curricula im internationalen Vergleich nehmen in vielen Fällen den Bereich Anwenden/**Interaktion** auf. Wir haben uns entschieden, diesen Bereich daher ebenfalls aufzunehmen und mit folgender Betonung zu verstehen: Interaktionspfade sind von Menschen gestaltet und damit auch veränderbar. Auch als

---

<sup>16</sup> Eine Turingmaschine ist ein vom Mathematiker Alan Turing 1936 vorgeschlagenes universelles Automatenmodell, welches auf einfache und damit gut zu analysierende Weise die Arbeitsweise eines Computers beschreibt.

„normaler“ oder „einfacher“ Anwender im Kindesalter muss ich nicht nur den vorgegebenen Pfaden folgen, sondern kann und soll auch mitgestalten (Gesellschaft für Didaktik des Sachunterrichts, 2013, S. 63). Interaktion ist für uns damit nicht „being programmed“, sondern als Einführung in aktives Mitgestalten gedacht.

Vor dem Mitgestalten durch Verändern des digitalen Artefakts gehört zu dieser Kompetenz auch das Wahrnehmen und Erkennen der (möglicherweise verschiedenen) Interaktionspfade (z. B. für eine Aufgabenlösung).

Entscheidend für unsere Perspektive hier ist: Digitale Artefakte sind von Menschen gemachte Artefakte, daher kann Bildung nur gelingen, wenn das Künstliche, das Gemachte in den Blick genommen wird. Kinder sollten bereits früh erkennen, dass die digitale Welt an menschlichen Bedürfnissen ausgerichtet werden kann und sollte. Sie sollten deren zweckbezogene Beschaffenheit soweit verstehen, dass sie beurteilen können, ob die Funktionen mit diesen Bedürfnissen übereinstimmen oder verändert werden sollten. Sie sollen sich selbst in der Rolle der Konstrukteurin/des Konstrukteurs erleben und die digitale Welt kreativ mitgestalten können!

Das eigene Konstruieren (bzw. Gestalten, Verändern, Adaptieren ...) aber setzt voraus, dass (einige) informatische Grundprinzipien bekannt sind und die Nutzerinnen und Nutzer den Wert des eigenen Gestaltens und Adaptierens für ihr zukünftiges Handeln ein- und abschätzen bzw. bewerten können. Selbst wenn sie noch keine Methoden kennen, die es ihnen erlauben, sich selbst in ein System einzuarbeiten, um es zu verändern, können schon Kinder ein Verständnis dafür entwickeln, dass diese Systeme prinzipiell gestaltbar sind: Es gibt diverse Ansätze und Systeme, mit denen Kinder aktiv gestaltend tätig werden können (vgl. die Abschnitte 2.3.2, 2.3.3 sowie die Beispiele in Kapitel 5). Wenn informatische Bildung dazu schon früh Grundlagen legt, dann hat sie unserer Meinung nach ihren Auftrag zur Erziehung von Vor- und Grundschulkindern in Richtung Mündigkeit und Ermöglichung von Teilhabe auch in der digitalen Welt erfüllt.

Im Folgenden skizzieren wir in Anlehnung an die Methode „Forschungskreis“ der Stiftung „Haus der kleinen Forscher“ (Stiftung Haus der kleinen Forscher, 2015, Stiftung Haus der kleinen Forscher, 2013b) den Prozess des Konstruierens und des Explorierens sowie deren Verschränkung. Der Explorationskreis geht dem Gestaltungskreis voraus – es sind aber auch Gestaltungen denkbar, die ohne Vorbild direkt beginnen.

Der Explorations- und der Gestaltungskreis wechseln zwischen zwei Perspektiven hin und her, verdeutlicht durch Farben: Rot bezieht sich auf die Bauweise bzw. die Struktur, Grün auf den Zweck bzw. die Funktion. Diese Dualität wird in verschiedenen Varianten und Formulierungen in der Didaktik der Informatik beschrieben.

Im Ansatz ‚Informatik im Kontext‘ kann der Kontext so gesehen werden, dass er sich vor allem auf den Nutzen, die Einsatzmöglichkeiten bezieht. Im Verlauf des Unterrichts erfolgt dann eine Dekontextualisierung, in der die technische bzw. innere Perspektive (rot) eingenommen wird. Am Ende wird dann beides verknüpft, indem rekontextualisiert wird (Koubek, Schulte, Schulze & Witten, 2009).

Stechert (2009) entwickelte ein Konzept zum „systematischen Erkunden des Verhaltens von Informatiksystemen“. Im Sinne des Black-Box-Prinzips kann systematisch das nach außen sichtbare Verhalten (grün) analysiert werden und schließlich das Wechselspiel des nach außen sichtbaren Verhaltens mit der inneren Struktur (rot) erkundet werden – in diesem Schritt wird dann das System als White-Box oder Glas-Box analysiert.

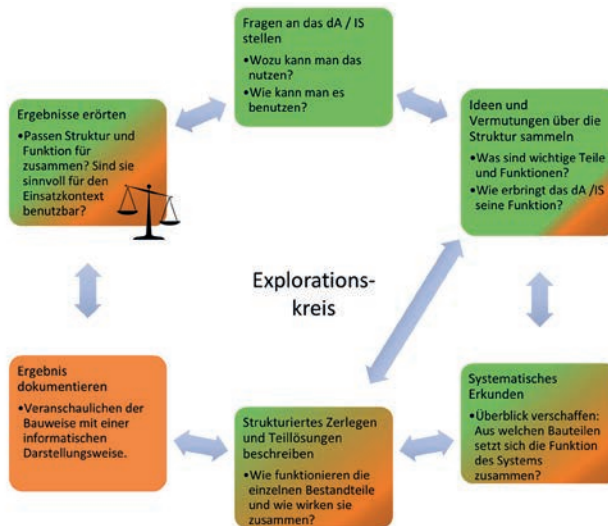
Die Dekonstruktion von Informatiksystemen (Magenheim, 2000) geht ebenfalls von der Analyse „fertiger“ Informatiksysteme (IS) aus. Der springende Punkt ist, dass diverse Aspekte eines IS nur dann zu verstehen sind, wenn man die ursprünglichen Design-Entscheidungen der Entwicklerinnen und Entwickler kennt. Bestimmte Eigenschaften sind ggf. zufällig oder Nebenprodukt, andere eine bewusste Design-Entscheidung bzw. Interpretation der Entwicklerinnen und Entwickler über den Zweck des Systems. Da diese aber nicht mehr eindeutig zu rekonstruieren sind, müssen sie anhand der vorhandenen Spuren nachempfunden werden. Insofern ist es eher eine Interpretation als eine Analyse. Diese bezieht sich auf die verschiedenen erreichbaren Sichten und Materialien: Benutzungsschnittstelle, Quelltext und Quelltext-Kommentare, Benutzungshandbücher, Entwickler-Dokumentationen etc. Dekonstruktion versucht also gewissermaßen anhand der Dokumentation der inneren Bauweise (rot), z. B. dem Quelltext die damit verbundenen Einsatzzwecke und Intentionen abzuleiten (grün) und darauf aufbauend das IS ggf. zu erweitern.

Die Dualitätsrekonstruktion versucht, direkt die beiden Bereiche Rot und Grün, also innere Struktur und Bauweise (rot) sowie Zweck, Intention, Nutzen (grün) zu verknüpfen (Schulte, 2008b). Eine Methode ist das Experimentieren (Schulte, 2012).

Anhand eines Experiments mit Robotik beschreibt Mioduser das Vorhandensein dieser beiden Perspektiven bei vier- bis fünf- sowie sechs- bis siebenjährigen Kindern. Er nennt Str (rot) die technische Sichtweise und Fkt (grün) die psychologische Sichtweise. In der psychologischen Sichtweise erklären die Kinder das Verhalten des Roboters mit psychologischen Kategorien wie Wille, Intention oder Persönlichkeit – in der technologischen erklären sie das Verhalten mit kausalen Ursache-Wirkungs-Beschreibungen und Regeln bzw. regelgeleitetem Verhalten (vgl. Levy & Mioduser, 2008).

### 1.5.1 Explorieren

Die Exploration eines gegebenen Artefakts (dA für digitales Artefakt) bzw. eines Informatiksystems (IS) ähnelt dem Experimentieren und damit ein wenig dem Forschungskreis der Naturwissenschaften. Hier geht es darum, anhand der nach außen wahrnehmbaren Funktion bzw. dem Einsatzzweck oder anhand der Nutzung des Systems – also aus der Benutzungsperspektive – zunehmend die innere Bauweise, die einzelnen Bestandteile und deren Wirkmechanismen zu erkunden (siehe Abbildung 7 in diesem Abschnitt). Das funktioniert rein durch äußere Beobachtung nur bis zu einem bestimmten Grad, aber die meisten Aspekte der Benutzungsoberfläche sind ja so gestaltet, dass sie die inneren Systemzustände oder Aspekte davon widerspiegeln. Es geht darum, die Aufmerksamkeit weg von der Aufgabe (einen Text verfassen, den Roboter fahren lassen, ein Foto machen) hin zur technischen Realisierung zu lenken (der Text wird nicht bildlich gespeichert, sondern durch Zeichen und Metazeichen, die beschreiben, wie der Text am Bildschirm aussieht ...).



**Abbildung 7.** Der Explorationskreis für die Untersuchung von digitalen Artefakten (Apps, Digitalkamera, Roboter, Textverarbeitungssoftware, ...)

Dazu kann, wie in der Abbildung 7 dargestellt, Explorieren als ein Prozess mit verschiedenen Schritten durchgeführt werden, wobei die Aufmerksamkeit immer stärker von der Funktion des dA (grün dargestellt) zur Struktur (man könnte auch sagen zur Funktionsweise bzw. den Wirkprinzipien) gelenkt werden. Diese Aspekte sind rot markiert. Insgesamt geht es darum, den Zusammenhang von Funktion und Funktionsweise, also ein ausgewogenes Verhältnis der beiden Far-



Schritt	Fragen an das dA / IS stellen	Ideen und Vermutungen über die Struktur sammeln	Systematisches Erkunden	Strukturiertes Zerlegen und Teillösungen beschreiben	Ergebnis dokumentieren	Ergebnisse erörtern
Leitfrage	Wozu kann man das nutzen? Wie kann man es benutzen?	Wie erbringt das dA / IS seine Funktion? Was sind wichtige Teile/Aufgaben?	Überblick verschaffen: Aus welchen Bauteilen setzt sich die Funktion des Systems zusammen?	Wie funktionieren die einzelnen Bestandteile und wie wirken sie zusammen?	Veranschaulichen der Bauweise mit einer informatischen Darstellungsweise.	Passen Struktur und Funktion für die Anwendung zusammen? Sind sie sinnvoll für den Einsatzkontext benutzbar? Passt vermutete Struktur zum tatsächlichen Systemverhalten?
Prozesskompetenz	P0: Interagieren und Explorieren (niedrige Stufe)	P0: Interagieren und Explorieren P4: Kommunizieren und Kooperieren	P0: Interagieren und Explorieren P3: Strukturieren und Vernetzen P4: Kommunizieren und Kooperieren	P3: Strukturieren und Vernetzen	P5: Darstellen und Interpretieren	P2: Begründen und Bewerten
Perspektive	Außen: Funktion	Vor allem Funktion, grobe Ideen, Struktur	Von Funktion zur Struktur	Innen: Struktur	Innen Struktur, mit Bezug zu Funktion	Balance von Struktur und Funktion
Erläuterungen	Zunächst Klären, wozu das Artefakt/ System da ist und gebraucht wird. Beim alltägl. Nutzen wird ggf. nur ein eingeschränkter Bereich erkannt, hier soll ein Überblick über den Einsatzbereich entwickelt werden. Dieser hilft, das dA zu zergliedern und Ideen über interne Arbeitsweise zu entwickeln	In diesem Schritt werden Ideen für interne Arbeitsweise entwickelt, diskutiert und verfeinert oder bereits verworfen. Hier werden auch erste sprachliche Beschreibungen für interne Arbeitsweisen und Konzepte erprobt.	Ideen werden systematisiert und Funktionsbereichen zugeordnet. Generelle Bereiche wie Eingabe, Verarbeitung, Ausgabe, Speicherung... Ideengeleitet, ggf. Bedienungsanleitung lesen; von Bedienung her untersuchen	Datenstrukturen und Algorithmen, Objektinteraktion Hier stärkere Abstraktion, Dekontextualisierung und Formalisierung; hier Notizen machen	Dokumentation der inneren Struktur. Use Case, Statechart, Flussdiagramm, ...	Einsatzkontexte und Rollen versch. Stakeholder, prüfen der Ergebnisse im Einsatz, ... <i>Kann hier in den Gestaltungskreis übergehen</i>

Abbildung 8. Die einzelnen Schritte des Explorationskreislaufs

ben, zu erlangen – nur dann kann verstanden werden, dass die Wirkmechanismen nicht einfach so, sondern bezogen auf den Einsatzzweck hin konstruiert wurden. Abbildung 8 zeigt die Fragestellungen der einzelnen Schritte genauer. Dabei wird hier auch schon der Bezug zu den in den folgenden Kapiteln erläuterten Prozesskompetenzen hergestellt.

Ein Hilfsmittel dazu ist die Darstellung mit passenden Ausdrucksweisen. Die Informatik hat zur Darstellung von Informatiksystemen bzw. einzelner Aspekte von Informatiksystemen verschiedene Ausdrucksweisen und verschiedene grafische Modellierungssprachen entwickelt, die hier eingesetzt werden können (vgl. Abbildung 6 in diesem Abschnitt sowie ausführlich Hubwieser, 2007).

### 1.5.2 Konstruieren

Das Konstruieren bezieht sich in der Informatik auf das Erstellen eines digitalen Artefakts bzw. eines Informatiksystems. Die verschiedenen Schritte beginnen grundsätzlich mit der Frage, was benötigt wird, um dann zu überlegen, was dazu konstruiert werden muss (siehe Abbildung 10). Noch stärker als beim Explorieren sind die Schritte ineinander verzahnt, so dass Zyklen, Rücksprünge d. h. also das schrittweise Annähern an die gewünschte Lösung hier natürliche Vorgänge sind.

**Abbildung 9.** Der Gestaltungskreis für die Konstruktion von digitalen Artefakten



Neben der Konstruktion eines komplett neuen Systems kann auch auf der Grundlage eines bestehenden Informatiksystems aufgebaut werden, das ggf. dazu zunächst exploriert werden muss. In der Konstruktion kann es sowohl beim Analy-

## Gestaltungs- Erläuterungen

Schritt	Gestaltungsziel entwickeln	Erkunden der inneren Struktur	Gestaltungs Ideen sammeln	Strukturiertes Lösen der Teilprobleme	Zusammenführen der Teillösungen	Lösung im Kontext ausprobieren und bewerten
Leitfrage	Was sind Voraussetzungen und Zweck des dA / IS ?	Wie erbringt das dA /IS seine Funktion? Was sind wichtige Teile/Aufgabe	Wie kann die Funktion umgesetzt werden?	Entwerfen einer System-Struktur für die Lösung Entwickeln von Teil-Lösungen	Lösung aus Einzelteilen zusammenbauen	Erproben und Abgleich der Struktur mit dem Zweck Bewerten in Bezug auf Kontexte
Prozesskompetenz	P0: Interagieren und Explorieren	P4: Kommunizieren und Kooperieren	P4: Kommunizieren und Kooperieren	P3:Strukturieren und Vernetzen	P1: Modellieren u Implementieren	P2: Begründen und Bewerten P5: Darstellen und Interpretieren
Perspektive	Fkt Mit Bezug zu Str	Von Fkt zu Str	Fkt Anhänger, vor allem Str	Fkt Anhänger, vor allem Str	Str (in Bezug zur Fkt)	Balance Fkt und Str
Erläuterungen	Entwickeln einer Idee für den Nutzen eines Systems, der so noch nicht da ist.	Prüfen der Randbedingungen für Verwirklichungsmöglichkeiten anhand der bestehenden Str und Fkt. <b>Kann hier in den Explorationskreis übergehen</b>	Str-Ideen für Umsetzung der Fkt entwickeln	Str aufteilen und präziser bestimmen, dabei auch Fkr in Teil-Fkt zerlegen und einzeln Str-Lösungen dafür entwickeln	Implementationsphase	Benutzungs- und Beurteilungsphase

Abbildung 10. Die einzelnen Schritte des Gestaltungsreislaufs

sieren der benötigten Funktionen als auch beim Konstruieren von Teillösungen sozusagen lokale Kreisläufe geben, in denen dann die Teilprobleme nacheinander einzeln bearbeitet werden (vgl. Abbildung 9). Abbildung 10 zeigt die einzelnen Schritte des Gestaltungskreislaufs im Detail.

## 1.6 Das Verhältnis von informatischer Bildung, Medienbildung und digitaler Bildung

In den Anfängen der Diskussion über Informatik in der Bildung nannte man Computer (engl. „to compute“: rechnen) gerne auch Rechner; wie wir im vorangegangenen Abschnitt gesehen haben mit enger Anbindung an die Mathematik. Doch zunehmend wurden Computer in verschiedensten Zusammenhängen eingesetzt, etwa in Bildung, Handel, Produktion, Wissenschaft, Militär, Gesundheitswesen oder Politik (vgl. Magenheim & Schulte, 2006) – und damit veränderten sich sowohl die Anwendungen als auch die Auswirkungen und die Sichtweise auf Computer. Nach einer Debatte, ob Computer vor allem als Rechner oder eher als Werkzeuge, als Instrumente oder als Medien eingeschätzt werden sollten, änderte sich die Sicht zumindest insofern, dass digitale Artefakte nun zumindest als interaktive ‚Medien‘ im persönlichen Umgang und mittlerweile auch zumeist als vernetzte Online-Medien gesehen werden.

In Bezug auf den Umgang mit digitalen Artefakten als „persönliche Medien“ charakterisiert Keil den „Computer als Medium – Medien als Denkzeug des Geistes“. Er formuliert:

„Der Sichtweise des Computers als Automaten, der ohne Eingreifen des Menschen Prozesse ausführt, und der Sichtweise des Computers als Werkzeug, das speziell unter dem Gesichtspunkt der Interaktivität den Nutzern einen Handlungs- und Entscheidungsspielraum zur Ablaufstrukturierung überlässt, stellt Coy 1995 die bis dahin ungebräuchliche Sichtweise des Computers als Medium zur Seite. Die Unterstützung verteilter kooperativer Arbeitsprozesse sei nicht mehr mit der Sichtweise des Werkzeugs verträglich bzw. könne damit nicht hinreichend begründet werden, denn: ‚Vernetzte Kooperation ist die Basis der modernen arbeitsteiligen Produktionsweise, ihre angemessene technische Unterstützung geschieht über vernetzte Rechner: Der Computer wird zum Medium (Coy, 1995a, 36)“ (Keil, 2012, S. 147).

Denkzeuge sind diese interaktiven Medien, da sie Rückmeldung und somit Differenzenerfahrung ermöglichen. Durch die Repräsentation etwa eines Rechenvor-

gangs kann dieser wahrgenommen und auf Differenzen zum eigentlich gewollten Ergebnis geprüft werden: Das schriftliche Denken ist daher leistungsfähiger als das Kopfrechnen. Diese Unterstützung macht digitale Artefakte als persönliche Medien zu Denkzeugen.

Der Informatiker und Vordenker Alan Kay geht soweit, digitale Artefakte durch ihre Kombination von Interaktion und automatisierter Verarbeitung als neue kulturelle Ausdrucksform zu begreifen, die das Buchzeitalter ablöst.

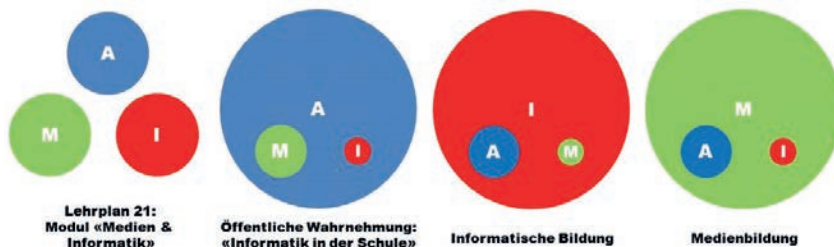
1977 haben Goldberg und Kay ihre Idee des Dynabook wie folgt beschrieben: ein Gerät in der Größe eines Notizbuches (im englischen: Notebook)

„which could be owned by everyone and could have the power to handle virtually all of its owner's information-related needs. Towards this goal we have designed and built a communications system: the Small-talk language, implemented on small computers we refer to as ‚interim Dynabooks‘. We are exploring the use of this system as a programming and problem solving tool; as an interactive memory for the storage and manipulation of data; as a text editor; and as a medium for expression through drawing, painting, animating pictures, and composing and generating music“ (Kay & Goldberg, 1977, S. 1).

Die Dynabook-Idee drückt genau das aus, was Keil als Denkzeug bezeichnet: ein interaktives Werkzeug für das Verfertigen und Ausdrücken von Gedanken, das Ausdrücken vor allem in dynamischer und interaktiver Form. Diese Medien werden daher nicht in herkömmlichen Sinn geschrieben, noch in herkömmlichen Sinn programmiert (im Sinne codiert), sondern mittels des ebenfalls interaktiven Werkzeugs in einem heuristischen Schreib- und Programmierprozess erzeugt. Damit entsteht eine neue mediale Ausdrucksform, die nur diejenigen zu nutzen wissen, die programmieren können. Kay und Goldberg haben dazu Squeak entwickelt, die erste objektorientierte Programmiersprache.

Das Verfertigen von Äußerungen ist dann zweifach auf Medien angewiesen: einmal auf Medien zur Verbreitung und zum Transport, zum anderen aber auch für den interaktiven Erstellungsprozess. Medienhandeln ist damit oft oder sogar immer auf Produkte angewiesen und findet etwa in sozialen Netzwerken statt, also in von kommerziellen Firmen bereitgestellten digitalen Infrastrukturen. Darauf verweist die Medienbildung und verknüpft das gerne mit dem Argument, eine rein ‚technische‘ Betrachtungsweise dieser digitalen Welt reiche deshalb nicht aus.

Döbeli fasst diese Diskussion in folgender Abbildung zusammen:



**Abbildung 11.** Verschiedene Möglichkeiten, den Zusammenhang zwischen (A)nwendung von IT, (I)nformatik und (M)edienbildung zu konzipieren (nach Döbeli Honegger, 2015)

In der öffentlichen Wahrnehmung stehen demzufolge die Anwendungen im Vordergrund: Das effiziente und effektive Nutzen von digitalen Artefakten bzw. Informations- und Kommunikationstechnologien. Aus der Perspektive der informatischen Bildung steht die Informatik im Mittelpunkt: das Verstehen der „Grundkonzepte der automatisierten Informationsverarbeitung“, das Nutzen der Grundkonzepte für Problemlösungen und zum Verstehen der Informationsgesellschaft. Aus der Perspektive der Medienbildung stehen die Medien im Mittelpunkt: Digitale Inhalte produzieren, kritische Reflexion von „Nutzung, Bedeutung und Wirkung“; vgl. auch Döbeli Honegger, 2016).

Die Idee, die Abbildung 11 zugrunde liegt, ist, aufzuzeigen, dass aus den jeweiligen disziplinären Sichtweisen heraus der jeweils „eigene“ Kreis als konstitutiv für die Betrachtung der anderen Kreise gesehen wird. Damit wird die Annahme verbunden, dass die jeweils ‚eingebetteten‘ Themen bzw. Kompetenzen (aus den anderen Bereichen) einfach nebenbei mit erworben werden. Dagegen scheint es sinnvoller, wie es in der Schweiz mit dem Lehrplan 21 versucht wird (Deutschschweizer Erziehungsdirektoren-Konferenz, 2016), die Bereiche eher gleichberechtigt nebeneinander zu berücksichtigen.

Es gibt daneben auch Versuche, weniger aus der eigenen disziplinären Sicht zu argumentieren und stattdessen allgemein oder übergreifend Kompetenzen für die digitale Welt zu beschreiben: Eine Expertengruppe aus Informatikerinnen und Informatikern sowie Medienwissenschaftlerinnen und Medienwissenschaftlern hat 2010 (für das BMBF) Kompetenzen für die digitale Welt beschrieben. Diese sind in vier Themen- und Aufgabenfelder gegliedert:

1. Information und Wissen
2. Kommunikation und Kooperation
3. Identitätssuche und Orientierung
4. Digitale Wirklichkeiten und produktives Handeln

Die einzelnen Kompetenzen werden dann ohne direkten Bezug zur Disziplin recht abstrakt beschrieben. Eine Kompetenz im Bereich Information und Wissen wird etwa wie folgt gefasst: „[...] die Herstellung und Verbreitung von Informationen und deren Erschließung als interaktive Prozesse begreifen und sich adressatengerecht, situationsbezogen und verantwortlich beteiligen“ (Deutschschweizer Erziehungsdirektoren-Konferenz, 2016, S. 9).

Aktuelle Ansätze suchen nach Wegen, die verschiedenen Aspekte oder Sichtweisen so zu vereinbaren, dass sie in Bildungsprozessen fruchtbar aufeinander bezogen werden können. Sie zielen darauf, einerseits eine häufig als sinnlos erlebte Einführung in innertechnische Wirkprinzipien ohne Lebensweltbezug zu verhindern und andererseits eine Bedienschulung, die nur ein Nutzen und damit Anpassen individueller Handlungsmöglichkeiten an vorgegebene digitaltechnische Systeme ermöglicht, zu vermeiden. Insbesondere soll so auch die gesellschaftlich-kulturelle Perspektive gestärkt werden:



**Abbildung 12.** Das Dagstuhl-Dreieck: Erkenntnisperspektiven auf die digitale Welt (Brinda et al., 2016)

Das Dagstuhl-Manifest beschreibt die drei Perspektiven mit folgenden Worten (Brinda, Diethelm, Gemulla, Romeike, Schöning & Schulte, 2016):

- Die **technologische Perspektive** hinterfragt und bewertet die Funktionsweise der Systeme, die die digitale Welt ausmachen. Sie gibt Antworten auf die Frage nach den Wirkprinzipien von Systemen, auf Fragen nach deren Erweiterungs- und Gestaltungsmöglichkeiten. Sie erklärt verschiedene Phänomene mit immer wiederkehrenden Konzepten. Dabei werden grundlegende Problemlösestrategien und -methoden vermittelt. Sie schafft damit die technologischen Grundlagen und das Hintergrundwissen für die Mitgestaltung der digitalen Welt.

- Die **gesellschaftlich-kulturelle Perspektive** untersucht die Wechselwirkungen der digitalen Welt mit Individuen und der Gesellschaft. Sie geht z. B. den Fragen nach: Wie wirken digitale Medien auf Individuen und die Gesellschaft? Wie kann man Informationen beurteilen, eigene Standpunkte entwickeln und Einfluss auf gesellschaftliche und technologische Entwicklungen nehmen? Wie können Gesellschaft und Individuen digitale Kultur und Kultivierung mitgestalten?
- Die **anwendungsbezogene Perspektive** fokussiert auf die zielgerichtete Auswahl von Systemen und deren effektive und effiziente Nutzung zur Umsetzung individueller und kooperativer Vorhaben. Sie fragt, wie und warum Werkzeuge ausgewählt und genutzt werden. Dies erfordert eine Orientierung hinsichtlich der vorhandenen Möglichkeiten und Funktionsumfänge gängiger Werkzeuge in der jeweiligen Anwendungsdomäne und deren sichere Handhabung.

Diesem Manifest folgend stellen wir hier ein Modell vor, das die im Dagstuhl-Dreieck aufgezeigten Perspektiven entfaltet. Wichtig ist uns dabei, dass die einzelnen Perspektiven ineinandergreifen. Wir gehen dabei ebenfalls von der digitalen, vernetzten Welt aus, in der Kinder heutzutage aufwachsen. Die vorrangige Perspektive ist für uns die technologische und damit die Frage: „Wie funktioniert das?“ Trotzdem sind die anderen beiden Perspektiven auf diesen Gegenstand der digitalen Welt ebenso wichtig.

## 1.7 Fazit: Informatische Bildung für alle

Informatische Bildung hat u. E. in Übereinstimmung mit den Bildungsplänen für den Elementarbereich und den Lehrplänen bzw. dem Perspektivrahmen für den Sachunterricht im Sinne grundlegender Bildung (Gesellschaft für Didaktik des Sachunterrichts, 2013, S. 9) das zentrale Ziel, Kinder dabei zu unterstützen, ihre Lebenswirklichkeit „sachbezogen zu verstehen, sie sich auf dieser Grundlage bildungswirksam zu erschließen und sich darin zu orientieren, mitzuwirken und zu handeln“ (Gesellschaft für Didaktik des Sachunterrichts, 2013).

Dabei geht es hier ganz spezifisch um die Orientierung in und das Verstehen der digitalen Welt, die heute und in Zukunft einen bedeutsamen Teil der Lebenswirklichkeit von Kindern ausmacht. Außerdem geht es auch darum, neue Ausdrucksmöglichkeiten kennenzulernen und gestalterisch, partizipativ und verantwortungsvoll an der Weiterentwicklung dieser digitalen Welt mitwirken zu können.

Wir orientieren uns dabei im Rahmen der Bildungsziele für Elementar- und Primarstufe im Wesentlichen an den von der Gesellschaft für Informatik vorgeschlagenen Standards für die Sekundarstufe I (GI – Gesellschaft für Informatik



e. V., 2008), die im Grunde ebenfalls bereits die drei im Dagstuhl-Dreieck explizierten Perspektiven aufweisen. Wir werden diese Standards jedoch um einen Aspekt erweitern: die Interaktion und Exploration als neuen Prozess (PO). Dieser bezieht sich auf Anwendungskompetenzen, aber nicht rein auf das Anwenden-Können vorgegebener konkreter Systeme, sondern das Einarbeiten-Können in unbekannte Systeme, das Verfügen über und konkret Anwenden-Können von allgemeinen Benutzungskompetenzen und die Fähigkeit, altersangemessen über Interaktionsmöglichkeiten und -konsequenzen nachzudenken. Das bezieht Aspekte des Reflektierens, des Beurteilens und des Bewertens von Interaktionsmöglichkeiten ein und zielt direkt auf die genannten allgemeinen Prinzipien und Teilhabemöglichkeiten an der digitalen Welt sowie die Stärkung der Selbstständigkeit.

Insgesamt wollen wir daher **Interaktion und Exploration** als eigenen Kompetenzbereich aufnehmen. Dieser wird im Unterschied zu Anwendungsfähigkeiten/digital literacy, die sich ausschließlich auf den kompetenten Umgang mit vorhandenen technologischen Geräten der aktuellen Generation beziehen (so wie sich Lesen und Schreiben auf die aktuelle vorhandene Landessprache und die aktuellen Rechtschreibregeln bezieht), in einem erweiterten Verständnis als Kompetenz zur Exploration und Bewusstmachung der dahinterliegenden informatischen Konzepte und Absichten definiert. Explorieren meint dabei nicht allein das freie Spielen, sondern auch durch Arbeitsaufträge angeleitetes Erkunden, mit dem Ziel, die Systeme zu „be-greifen“, d. h. erste Zugänge zum Verstehen der Funktionsweise zu entwickeln (vgl. dazu die Abschnitte 3.3.2 und 4.3.2, in denen der Prozessbereich genauer erklärt und diskutiert wird). Das Umgehen mit einem digitalen Artefakt ist daher nur die basale Ebene dieses Kompetenzbereichs. Im Kern steht die Systemerkundung, die Exploration durch zielgerichtete Interaktion, die auf das Verstehen und Zu-eigen-Machen abzielt und dabei insbesondere auch Aspekte der Anpassung und Gestaltung im Sinne eines End-User-Designs einschließt.

Informatik ermöglicht, verschiedene Sichten auf ein Informatiksystem bzw. ein digitales Artefakt zu erschließen, beispielsweise das Konzept des Automaten aus der theoretischen Informatik: ein Modell, das die Funktionsweise digitaler Artefakte als eine Menge festgelegter Zustände beschreibt, zwischen denen gewechselt werden kann. Diese Zustände müssen nicht alle direkt an der Benutzungsschnittstelle angezeigt werden. Durch gezieltes Erkunden kann aber ein entsprechender Zustandswechsel initiiert und die Bedeutung bzw. Auswirkung studiert werden.

Interaktion als Kompetenz bezieht sich also nicht nur auf den Umgang mit dem System, sondern auf das Erschließen und Nachdenken über die Bedeutung der Interaktion. Diese Bedeutung kann über die einzelne Anwenderin/den einzelnen Anwender und den einzelnen Anwendungsfall hinausgehen und sich auf

eine Gruppe von Personen beziehen, auf einfachem Niveau etwa in der Wahl eines Werkzeugs für die Nutzung durch die eigene Kita-Gruppe.

Offen ist, wie weit dieses mit Kindern direkt möglich ist. Sie sollten jedoch nicht nur den konkreten Umgang mit einem digitalen Artefakt erlernen, sondern sich allgemeine und übertragbare Strategien aneignen und sich zutrauen, ein unbekanntes System zu erkunden, und dabei auch über dessen Möglichkeiten, Grenzen und Auswirkungen nachdenken können. Ein wichtiger Aspekt dieser „Erkundungskompetenz“ ist die Einsicht, dass die meisten Artefakte das Anpassen und Einstellen in Bezug auf die eigenen Wünsche ermöglichen, also selbst gestaltbar sind. Daraus folgt die Möglichkeit, diese Anpassung vornehmen zu können und sich das konkret zuzutrauen. Das kann über Konfigurieren, Parametrisieren bis hin zu kleineren Programmieraktivitäten im Sinne des End-User-Programming geschehen. Dabei können die Kinder zudem erkennen, dass Informatik nicht nur Interaktion, also Umgehen mit einem System, ist, sondern das Gestalten und Realisieren der Umgangsmöglichkeiten einschließt. Diese Art der Programmierkompetenz ist nicht nur für (professionelle) Softwareentwicklerinnen und Softwareentwickler, sondern für alle Nutzerinnen und Nutzer digitaler Systeme sinnvoll.

## 2 Fundierung von Zieldimensionen auf Ebene der Kinder

Zur Fundierung der Zieldimensionen informatischer Bildung werden Forschungslage und Erfahrungen im Kontext zum informatischen Kompetenzerwerb herangezogen. Ausgehend von täglichen Erfahrungen der Kinder im Umgang mit Informationssystemen werden wir in Abschnitt 2.1 zunächst die Ausgangslage betrachten, d. h. wir versuchen, auf Basis von Studien zu ergründen, ob und wie Kinder digitale Medien bereits nutzen, entsprechende Erfahrungen gesammelt haben und wo informatische Bildung anzusetzen hat. Im Folgenden erörtern wir die Forschungslage hinsichtlich der lernpsychologischen Grundlagen: Welche Voraussetzungen müssen zum erfolgreichen Kompetenzerwerb erfüllt sein, und wie entwickelt sich Interesse für Informatik? In Abschnitt 2.3 werden die existierenden Ansätze zur Vermittlung informatischer Bildung skizziert und systematisiert. Das Kapitel schließt mit einem Vergleich internationaler Curricula zur informatischen Bildung und deren Einordnung in das gewählte Kompetenzmodell.

### 2.1 Kinder in digitalen Welten

#### 2.1.1 Nutzungserfahrungen

Kinder sammeln heute schon früh Erfahrungen mit verschiedenen digitalen Artefakten wie Computern, Tablets oder Smartphones: Sie beobachten, wie Erwachsene oder Kinder diese Geräte benutzen, und sie nutzen sie auch selbst. Die Erforschung dieser Nutzungserfahrungen von Kindern und Kleinkindern mit digitalen Artefakten hat gerade erst begonnen. Die vorliegenden, aktuellen Studien nehmen allerdings keine informatische Perspektive ein, sondern untersuchen Nutzungserfahrungen in Bezug auf medienpädagogische und medienerzieherische Aspekte sowie im Hinblick auf Chancen und Gefahren der digitalen Welt für Kinder.

Da die regelmäßige Nutzung digitaler Artefakte in Schulen und Kindergärten derzeit noch keine Selbstverständlichkeit ist, liegt die Verantwortung für diesbezügliche Nutzungserfahrungen von Kindern weitest-



gehend bei den Eltern bzw. Erziehungsberechtigten. Wie die DIVSI U9-Studie deutlich macht, hängt der Einfluss dabei von den Perspektiven und Einstellungen der Eltern ab, die sich, abhängig vom Milieu, bezüglich ihres Umgangs mit und hinsichtlich ihrer Überzeugungen zur digitalen Welt deutlich unterscheiden (DIVSI U9-Studie, 2015). Die Studie von Chaudron kommt zu ähnlichen Befunden: Kindliche Nutzungserfahrungen sind demzufolge „shaped partly by parents’ beliefs, values and ethnotheories“ (Chaudron, 2015, S. 42). Dabei kann nicht davon ausgegangen werden, dass bildungsnahe Elternhäuser besonders prädestiniert für häufige oder qualitativ hochwertige Nutzungserfahrungen bei den Kindern sind. Im Gegenteil: Gerade diese Eltern sind durch negative oder dramatisierende Medienberichterstattung verunsichert oder besorgt, insbesondere bezüglich des Datenschutzes und der Wahrung der Privatsphäre. Eltern aus eher bildungsfernen Milieus sehen im Gegensatz dazu oft keine Notwendigkeit für aktive Eingriffe in die kindliche Nutzung digitaler Artefakte oder für regulierende oder erzieherische Einflussnahme (Chaudron, 2015).

Die Nutzung selbst findet oft isoliert und unbeobachtet statt – im Gegensatz zu den meisten anderen kindlichen Aktivitäten, die viel häufiger gemeinsam mit Eltern, Geschwistern oder im Familienkreis ausgeführt werden:

„New (online) technologies were not perceived as an integral part of shared family life in most families. Rather, engaging with a digital device was considered as an individual activity, unlike offline family activities such as going to the park or playing a board game“ (Chaudron, 2015, S. 28).

Chaudron macht auch darauf aufmerksam, dass viele Eltern die Fähigkeiten ihrer Kinder bezüglich ihrer ‚digital skills‘ überschätzen, dagegen aber die Zeitdauer, die ihre Kinder mit digitalen Artefakten verbringen, unterschätzen. Die damit verbundene Vermutung erscheint plausibel: „This might be because so much digital use was to fill time when the parent was otherwise engaged“ (Chaudron, 2015, S. 30). Zugang zu Artefakten wird Kindern also oft genau dann gegeben, wenn die Erziehungspersonen abgelenkt sind – also um kindliche Wartezeiten zu überbrücken und das Kind zu beschäftigen, wenn Erziehungsberechtigte gerade keine Zeit haben.

Inwiefern Eltern ihren Kindern die Nutzung digitaler Artefakte gestatten bzw. deren Nutzung überwachen, ist wohl auch von deren eigenen technischen Kenntnissen und Fähigkeiten abhängig. In den Basisuntersuchungen zur Mediennutzung von Kindern und Jugendlichen des medienpädagogischen Forschungsver-

bunds Südwest<sup>17</sup> (MPFS – Medienpädagogischer Forschungsverbund Südwest, 2014a) wird – zumeist als einziges Beispiel für konkrete technische oder im weitesten Sinne informatische Kenntnisse – auf die Einrichtung von Schutzmaßnahmen bzw. Begrenzungen von Zugriffsmöglichkeiten auf Online-Angebote eingegangen (Filter parental control, Begrenzung von Nutzungszeiten durch Software, Einstellungen von Routern, Vorauswahl und Installation kindgerechter Apps auf Smartphones und anderen Artefakten o. Ä.). Teilweise ziehen die durchaus berechtigten Befürchtungen der Eltern bisher oft kaum Konsequenzen bezüglich ihrer Maßnahmen oder Verhaltensweisen nach sich (MPFS – Medienpädagogischer Forschungsverbund Südwest, 2014a, S. 73). Insgesamt zeichnen die Studien ein Bild, demzufolge Eltern mit der Gestaltung kindlicher Nutzungserlebnisse bzw. insgesamt mit der Gestaltung kindlicher Sozialisation in die digitale Welt eher überfordert sind (jedoch abhängig von den jeweiligen Milieus mit großen Unterschieden). Es wird teilweise empfohlen, dass Kindergärten und Schulen einen proaktiven Ansatz verfolgen sollten, um Eltern zu helfen, und selbst erzieherisch und bildend eingreifen sollen.

Die derzeit verfügbaren Studien, die das Erleben und Verhalten von (kleinen) Kindern in der digitalen Welt untersuchen, suggerieren ein relativ übereinstimmendes Bild der Nutzungserfahrungen im Umgang mit „Computern“: Im Rahmen der miniKim-Studie von 2014 wird festgestellt:

„Computer und Internet spielen im Alltag der Zwei- bis Fünfjährigen eine sehr untergeordnete Rolle. Mit 85 Prozent hat die deutliche Mehrheit noch überhaupt keine Erfahrungen mit dem Computer gesammelt. Nur jedes zehnte Kind nutzt den Computer – alleine oder gemeinsam mit den Eltern – mindestens einmal pro Woche. Dabei nutzen fast doppelt so viele Jungen (13 %) wie Mädchen (7 %) den PC mindestens wöchentlich. Bei den Vier- bis Fünfjährigen hat schon jeder Vierte (24 %) Erfahrungen mit dem Computer gemacht, während die Computernutzung bei den Zwei- bis Dreijährigen (6 %) die absolute Ausnahme ist“ (MPFS – Medienpädagogischer Forschungsverbund Südwest, 2014b, S. 23).

Berücksichtigt man allerdings, dass Smartphones, Tablets und andere digitale Medien zunehmend anstelle von „Personalcomputern“ Verwendung finden, ändert sich das Bild. So zeigt die DIVSI U9-Studie, dass sich bereits 23 Prozent der dreijährigen Kinder mit Smartphones beschäftigen und sich aus der Perspektive der Eltern „recht versiert und intuitiv beim Erlernen der Bedienung von Endgeräten

---

<sup>17</sup> Die Ergebnisse der regelmäßigen Untersuchungen (JIM-Studie, KIM-Studie, miniKIM- und FIM-Studie) werden veröffentlicht auf <https://www.mpfs.de>.

mit Touchscreen“ zeigen (DIVSI U9-Studie, 2015). Für die Zukunft ist hier nochmal eine deutliche Steigerung der frühen Nutzung digitaler Medien durch Kinder zu erwarten. So zeigt sich auch im europäischen Vergleich in einer Studie in vier europäischen Ländern: „[...] children under five are heavy users of a number of digital technologies at home“ (Palaiologou, 2016, S. 2). Palaiologou folgert aus einer Übersicht verschiedener Studien, dass die meisten drei- bis vierjährigen Kinder eine Art frühe „digital literacy“ erworben haben, mit der sie verschiedene digitale Technologien verstehen und nutzen können, um Informationen zu gewinnen und um erste Fähigkeiten im Umgang mit gedruckter Sprache („print literature“) weiterzuentwickeln (Palaiologou, 2016, S. 6).

Nach Plowman et al. können Vorschulkinder durch den Umgang mit digitalen Technologien zu Hause in vier Bereichen lernen (Plowman, Stevenson, Stephen & McPake, 2012):

1. manuelle Bedienfertigkeiten im engeren Sinn (etwa Umgang mit der Maus oder Gesten auf Touchscreens u. Ä.), wobei dieser Bereich eher nachrangig gewertet wird;
2. Erwerb von Wissen und Fähigkeiten in verschiedenen Bereichen wie Sprache und Mathematik, vor allem über die Inhalte in digitalen Spielzeugen und Medien. Das ist der Bereich, der von der Spieleindustrie in der Werbung herausgestellt wird;
3. Erweitern der sozial-emotionalen und kognitiven Fähigkeiten; etwa durch Erfolgserlebnisse beim Nutzen;
4. Verstehen der Rolle von Technologie im Alltag durch eigene Nutzung und Beobachtung von Familienmitgliedern beim Umgang mit Technologien. Die Autoren stellen ein Fallbeispiel heraus, um zu zeigen, dass diese Lernmöglichkeiten nicht unbedingt von der Verfügbarkeit der Technologie zu Hause abhängt, sondern von den spezifischen familiären Umgangsweisen damit und Einstellungen der Eltern (Plowman et al., 2012, S. 36). Dabei zeigt sich auch, dass z. B. der sozioökonomische Status der Familie kein genereller Prädiktor für die Art des familiären Umgangs mit Technologie ist.

Rosen et al. befragten Eltern nach negativen Auswirkungen. Der Studie zufolge führt erhöhter „Technologiekonsum“ (TV, Internet, Videospiele) zu Beeinträchtigungen (Rosen et al., 2014). Andererseits stellt Palaiologou fest:

„Analysis of the qualitative data emerging from the interviews suggested that children are ‘digitally fluent from a very young age’. One of the key findings was that parents felt that their definition of an illiterate person no longer corresponded to the traditional view of someone who cannot read and write, but rather was considered as a person who cannot learn, unlearn, relearn and use digital technologies as part of their everyday lives“ (Palaiologou, 2016, S. 2).

Gutnick et al. (2011) konstatieren in einer Zusammenschau, dass das Fernsehen immer noch Leitmedium bleibt (vgl. dazu auch MPFS – Medienpädagogischer Forschungsverbund Südwest, 2014a, S. 73). Allerdings stellen sie einen Wechsel fest, der sich im Alter von ca. 8 Jahren vollzieht, wenn Kinder mehr und mehr digitale und vor allem mobile Medien entdecken. Vor allem die mobile Nutzung bzw. mobile Medien üben die größte Faszination aus. Zu ähnlichen Schlussfolgerungen bzw. zu einer Bestätigung dieser Schlussfolgerungen kommt Palaiologou (2016).

### **2.1.2 Kinder nutzen selbstbewusst verschiedene digitale Artefakte**

Nach der miniKim-Studie nutzen Kinder bis 5 Jahre vor allem Spiele. „An zweiter Stelle steht das Malen oder Zeichnen am Computer, etwas weniger Jungen und Mädchen nutzen ein spezielles Lernprogramm. Ein paar der zwei bis- fünfjährigen Computernutzer schreiben nach Angabe der Haupterzieher auch schon Texte oder Worte am PC“ (MPFS – Medienpädagogischer Forschungsverbund Südwest, 2014b, S. 23). Zu ähnlichen Ergebnissen kam auch Palaiologou (2016, S. 11).

Die KIM-Studie macht deutlich, dass Kinder im Alter von 6 bis 13 Jahren verschiedene Angebote und Apps nutzen:

„Zu den Anwendungen, die mindestens einmal pro Woche von den Nutzern (unabhängig vom Nutzungsort) ausgeübt werden, zählt in erster Linie das Schreiben von Wörtern und Texten (50 %), von einem etwas geringeren Anteil (45 %) werden mit dieser Intensität Fotos und/oder Videos betrachtet. Ein Viertel malt oder zeichnet – Mädchen (30 %) häufiger als Jungen (21 %). Mit zunehmendem Alter gewinnen das Schreiben (6–7 Jahre: 27 %, 8–9 Jahre: 35 %, 10–11 Jahre: 51 %, 12–13 Jahre: 66 %) und das Betrachten von Bildern bzw. Videos an Bedeutung hinzu (6–7 Jahre: 31 %, 8–9 Jahre: 37 %, 10–11 Jahre: 44 %, 12–13 Jahre: 57 %), Malen und Zeichnen hingegen büßen an Attraktivität ein (6–7 Jahre: 45 %, 8–9 Jahre: 34 %, 10–11 Jahre: 20 %, 12–13 Jahre: 16 %). Bei den ab Zehnjährigen bearbeitet knapp ein Viertel mindestens einmal pro Woche Bilder und Videos – Mädchen (27 %) etwas häufiger als Jungen (19

%), die Älteren mehr (27 %) als die Jüngeren (19 %)“ (MPFS – Medienpädagogischer Forschungsverbund Südwest, 2014a, S. 34).

In einigen Studien wird betont, dass z. B. aufgrund des engen Spektrums und des repetitiven Charakters der Nutzung die jeweils beobachtete Aktivität kompetent und selbstbewusst durchgeführt wird, dies aber keinen Indikator für generelle Nutzungskompetenz oder gar das Verstehen von dahinterliegenden internen Vorgängen oder das Verstehen von Funktionsprinzipien impliziert: Dazu heißt es in der KIM-Studie 2014:

„In der Differenzierung nach Alter wird deutlich, dass sich die technischen Kompetenzen der Kinder jenseits des Abspielens einer DVD meist erst ab dem zehnten Lebensjahr ausprägen. Bei den Acht- bis Neunjährigen kann zwar ein Drittel gut alleine ins Internet gehen und ein Viertel sieht sich in der Lage etwas auszudrucken, aber nur ein knappes Fünftel kann Lieder auf den MP3-Player laden und nur jeder Zehnte kennt sich mit der Ablagestruktur eines Computers aus. Auch im Alter von 10 bis 11 Jahren werden die meisten Aufgaben nur von jedem dritten bzw. vierten Kind beherrscht. Und selbst unter den ältesten Kindern gibt es für einige Tätigkeiten noch Nachholbedarf“ (MPFS – Medienpädagogischer Forschungsverbund Südwest, 2014a, S. 60).

Allerdings ändert sich die Ausgangslage langsam, aber stetig:

„Der Langzeitvergleich zeigt, dass der Zuwachs an technischer Kompetenz in den letzten Jahren trotz Omnipräsenz der Medien im Alltag der Kinder nur partiell vonstattengegangen ist. So können heute deutlich mehr Kinder als noch 2010 alleine ins Internet gehen. Ein deutlicher Entwicklungsschub ist hinsichtlich des Herunterladens von Handybildern zu beobachten. Aber die Fähigkeit etwas auszudrucken, Lieder auf den MP3-Player zu laden oder eine Dateisystematik aufzubauen, hat sich ebenso wenig positiv entwickelt wie das Herunterladen von Dateien aus dem Internet“ (MPFS – Medienpädagogischer Forschungsverbund Südwest, 2014a, S. 61).

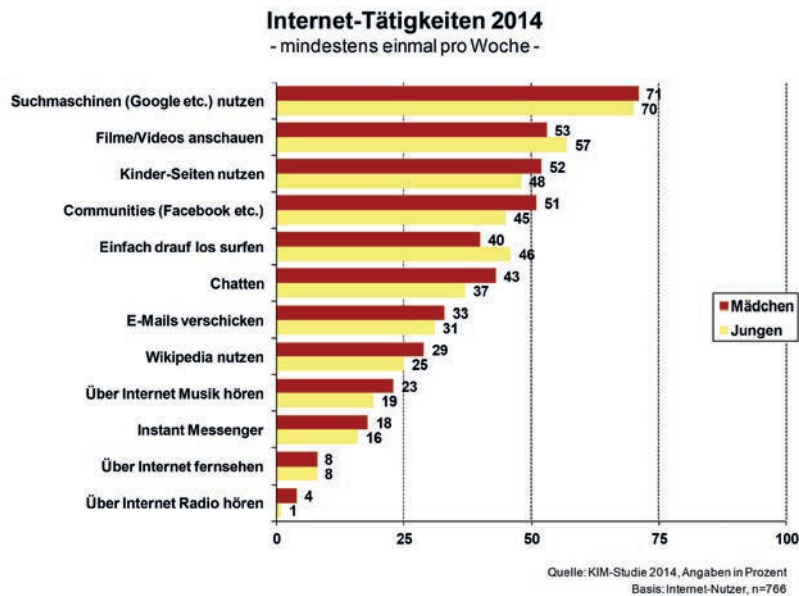
### **2.1.3 Nutzen von Online-Angeboten durch Kinder**

Bezüglich etwa der Fähigkeit, nach Online-Angeboten zu suchen, scheinen Kinder (altersabhängig) eher vorgeschlagenen Angeboten zu folgen. Diese ‚Suchstrategie‘ könnte man ‚Stöbern‘ oder ‚Browsen‘ nennen, wenn sie beispielsweise auf YouTube die vorgeschlagenen Folgefilme anschauen.



Online nutzen Kinder vor allem Suchmaschinen und Videoportale wie YouTube. Dabei können sie oft nicht erkennen oder unterscheiden, ob sie offline oder online sind, wenn sie sich ein Video ansehen. Sie verwenden auch recht häufig soziale Apps oder Dienste; hier vor allem WhatsApp.

Sie nutzen ebenfalls Internetsuchmaschinen, vor allem Google oder frag-FINN. Dabei greifen sie meist auf visuelle Vorschläge zurück (etwa bei YouTube), als Suchbegriffe einzugeben. Ähnlich, d. h. durch Folgen von visuellen Hervorhebungen, können sie auch im Web navigieren (siehe Abbildung 13).



**Abbildung 13.** Ergebnisse der KIM-Studie zu Tätigkeiten von 6- bis 13-jährigen Kindern im Internet (ohne Altersdifferenzierung)

Eltern äußern hier oft Befürchtungen, dass Kinder mit nicht altersgerechten Inhalten konfrontiert werden könnten (MPFS – Medienpädagogischer Forschungsverbund Südwest, 2014a). Eltern sowie Wissenschaftlerinnen und Wissenschaftler äußern teilweise auch Befürchtungen in Bezug auf den entstehenden digitalen Fußabdruck und die Wahrung der Privatsphäre.

Interessant ist hier, dass diese Befürchtungen sich ausschließlich auf explizit erstellte Daten beziehen, jedoch nicht auf Verbindungs- oder Metadaten. Auch Chaudron stellt fest, dass Eltern das Wissen über Online-Risiken fehlt (Chaudron, 2015, S. 5). Insgesamt scheint unklar, wie auf die „digitalen Fußabdrücke“ der Kinder durch deren Online-Nutzung reagiert werden kann oder sollte (Chaudron, 2015, S. 7). Die Rolle von Bildungseinrichtungen in dieser Frage ist nach den Studien ebenfalls unklar.

Für den Schutz vor nicht altersangemessenen Inhalten oder die Begrenzung der Zugangszeit werden teils technische Lösungen als Möglichkeit genannt, die jedoch nicht von allen Eltern umgesetzt werden können (vgl. Chaudron, 2015, S. 42).

#### **2.1.4 Zusammenfassung**

Insgesamt agieren Kinder (und Eltern) selbstsicher mit digitalen Artefakten. Diese Art der Nutzung führt zum Eindruck, sich insgesamt mit der digitalen Welt auszukennen und eventuell auch die wesentlichen Prinzipien und Wirkungsweisen zu verstehen. Nach den betrachteten Studien ist aber die tatsächliche Bandbreite der Nutzung gering, so dass auch die beobachtete sichere Nutzung eher einen Nebeneffekt der milieuspezifischen (eingeschränkten) Wahrnehmung der insgesamt vorhandenen Nutzungsmöglichkeiten darstellt. Dies trifft umso stärker auf Konsequenzen und Effekte der Interaktion zu, die nicht direkt dem WYSIWYG<sup>18</sup>-Paradigma zuzuordnen sind. Diese werden dann weniger wahrgenommen und bleiben im eigenen Nutzungsverhalten unberücksichtigt. Obwohl insgesamt eine recht verbreitete allgemeine Besorgnis um Datenschutz und Privatsphäre vorhanden ist, wurde die Rolle von Meta- und Verbindungsdaten nicht einmal von den Autoren der genannten Studien erkannt.

Durch die zunehmende Verbreitung von Smartphones und Tablets mit Touch-Bedienung sowie die ebenfalls zunehmende Bedienung durch Spracheingaben verschieben sich bereits wahrnehmbar die Nutzungsweisen von Kindern. Konsequenzen davon werden allerdings nicht benannt. Bezüglich smartphonerorientierter Betriebssysteme kann jedoch davon ausgegangen werden, dass etwa die Fähigkeiten zur Organisation und Verwaltung von Daten damit weniger oft trainiert werden, da diese Betriebssysteme zumeist den direkten Zugriff auf das Dateisystem erschweren oder ganz verhindern. Möglicherweise erklärt das die oben genannten widersprüchlichen Interpretationen, ob etwa Kinder den Zugriff auf selbstgemachte Fotos mit dem Smartphone organisieren können, so z. B. die Möglichkeit, Fotos vom Handy auf dem PC oder dem TV-Gerät anzusehen. Durch Weitergabe der Fotos an kommerzielle Cloud-Dienste geht das oft erstaunlich einfach. Wenn aber diese privaten Daten ohne den Umweg über Dritte direkt übertragen und dann auch einfach wiedergefunden werden sollen, erfordert das mehr Verständnis über die technischen Möglichkeiten und den Aufbau der entsprechenden Betriebssysteme.

---

<sup>18</sup> WYSIWYG: *What-You-See-Is-What-You-Get*: Nutzer erhalten unmittelbar visuell das Ergebnis der Interaktion, siehe auch („WYSIWYG“, 2016).

Insgesamt ist eine große Bandbreite in den beobachteten Nutzungsfähigkeiten festzustellen, das betrifft auch die Eltern, die dann, je nachdem, ihren Kindern mehr oder weniger hilfreich zur Seite stehen können.

## 2.2 Lernpsychologische Grundlagen

### 2.2.1 Kognitive Voraussetzungen

Informatisches Denken und Handeln in Bezug auf Kinder ist ein bislang vernachlässigtes Forschungsfeld (vgl. Borowski, Diethelm & Mesaros, 2010). Lernumgebungen, die es Kindern ermöglichen, ein Grundverständnis für die Funktionsweisen von Computern oder sogar kreative und gestalterische Potenziale zu entwickeln, liefern erste Ansätze, wie der Lerngegenstand ‚Informatik‘ Kindern zugänglich gemacht werden könnte (Borowski & Diethelm, 2009; Borowski et al., 2010). Im Vordergrund der aktuellen Bemühungen um informatische Bildung im Kindesalter stand jedoch bis vor ein paar Jahren verstärkt die Medienbildung (Borowski et al., 2010).

Wenn auch die Forschungslage insgesamt noch unbefriedigend ist, kann hinsichtlich zentraler informatikbezogener Kompetenzen auf Erkenntnisse aus mathematikbezogenen Studien (Benz, Grüßing, Lorenz, Selter & Wollring, 2017) sowie auf Ergebnisse der Lern- und Entwicklungspsychologie zurückgegriffen werden.

Um kognitive Voraussetzungen von Lernenden beschreiben zu können, unterscheidet man Novizinnen und Novizen und Expertinnen und Experten, die es auf unterschiedlichen Altersstufen geben kann (vgl. Stern, 2002, S. 29). Kindliche Expertinnen und Experten können dabei sogar erwachsenen Novizinnen und Novizen kognitiv überlegen sein (Stern, 2002). Empirische Untersuchungen zum bereichsspezifischen Wissen zeigen, dass Kinder bei ihnen vertrauten Sachverhalten bessere Leistungen erbringen können und eine effektivere Organisationsstrategie hervorbringen als Erwachsene, denen die Inhalte fremd sind (Sodian, 2002). Es wird angenommen, dass jüngere Kinder älteren deshalb noch oft unterlegen sind, weil sie weniger Zeit und Gelegenheiten hatten, sich Wissen und Können anzueignen und Aufgaben von ihnen oft nur deshalb nicht adäquat gelöst werden, weil ihnen das notwendige bereichsspezifische Wissen bisher fehlt (vgl. Sodian, Koerber & Thoermer, 2006). Schneider, Körkel und Weinert konnten schon früh am Beispiel ‚Fußball‘ nachweisen, dass Kinder, die ein bereichsspezifisches Vorwissen haben, bezüglich des Verstehens und Behaltens diesbezüglicher Inhalte sowohl gegenüber intelligenteren als auch gegenüber älteren Kindern im Vorteil sind (Schneider, Körkel & Weinert, 1989). Dies wäre für informatische Aufgabenbeispiele zu überprüfen.

Aufgrund erster Erfahrungen mit informatikbezogenen Lernumgebungen für Kinder im Vor- und Grundschulalter sowie einigen wenigen Studienergebnissen gibt es aber auch bereits spezifische Hinweise darauf, dass Kinder im Vor- und Grundschulalter informatikbezogene Kompetenzen erwerben können. Schwill stellte basierend auf Studienergebnissen Überlegungen dazu an, inwiefern jüngere Kinder dazu in der Lage sind, fundamentale Ideen der Informatik wie ‚Rekursion‘, die ‚Greedy-Methode‘ (sukzessiver Ausbau von Teillösungen zur Gesamtlösung), ‚Strukturiertes Zerlegen‘ und die ‚Nachbildung hierarchischer Strukturen‘ nachzuvollziehen bzw. zu verstehen oder anzuwenden (Schwill, 2001). Er schlussfolgert aus seinen theoretischen Betrachtungen, dass „eine Reihe wichtiger fundamentaler Ideen der Informatik bereits von Kindern im Grundschulalter erfasst werden kann, vorausgesetzt, die Gegenstände werden altersgemäß aufbereitet und im Unterricht unter Berücksichtigung der kognitiven Strukturen der Kinder und unterstützt durch Handlungen oder reale Gegenstände vermittelt“ (Schwill, 2001, S. 17). Gibson fand heraus, dass selbst Grundschul Kinder, die des Lesens und Schreibens noch nicht mächtig sind, lernen können, mit grafischen Algorithmen umzugehen (Gibson, 2012). Das Verstehen von Algorithmen wird für die informatische Bildung als zentral angesehen (vgl. Modrow, 2010; Schwill, 1995).

Grenzen für das Verständnis von informatischen Konzepten bei Kindern stellen dagegen die Autoren einer Studie fest, in der Kinder unter Nutzung der grafischen Programmiersprache „PiktoMir“ Programmieraufgaben lösen sollten (Rogozhkina & Kushnirenko, 2011). Von den sechs Kindern im Alter von unter 6 Jahren konnten nur zwei alle Aufgaben lösen. Daraus schlussfolgern die Autoren, dass Kinder dieser Altersgruppe noch nicht dazu in der Lage seien, informatische Konzepte adäquat zu verstehen. Bedenkt man jedoch das Alter der Kinder, so könnte man das Ergebnis durchaus auch positiv bewerten: Immerhin schafft es ein Drittel der unter Sechsjährigen schon, alle Aufgaben richtig zu bearbeiten! Zu untersuchen wäre, ob das Ergebnis sich noch verbessern ließe, wenn die Kinder bereits auf Vorerfahrungen zurückgreifen könnten. Darüber hinaus hängen solche Studienergebnisse eng mit der Wahl der verwendeten Programmierumgebung zusammen. So berichten Portelance, Strawhacker und Bers, dass Kinder ab 5 Jahren in der Lage sind, Algorithmen bei der Erstellung von Spielen und Animationen mit der speziell für die Altersgruppe entwickelten Programmierumgebung ScratchJr zu entwickeln (Portelance, Strawhacker & Bers, 2016). So wiesen Weintrop und Wilensky nach, dass die grafische Darstellung der algorithmischen Strukturen beim Lernen Vorteile gegenüber der textuellen Darstellung aufweist (Weintrop & Wilensky, 2015).

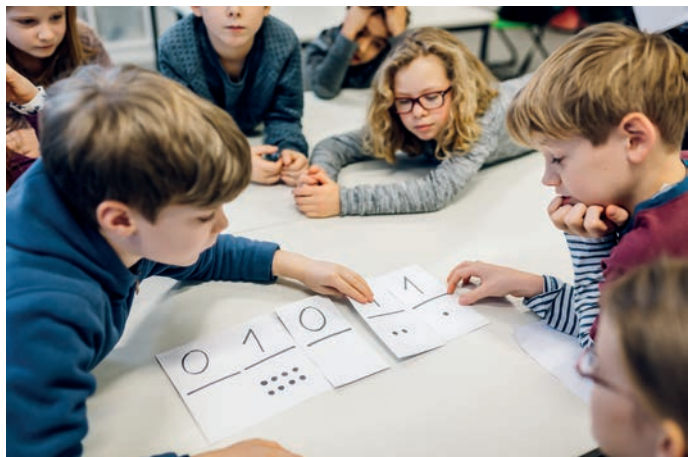
### 2.2.2 Interesse an Informatik

Grundschul Kinder sind im Hinblick auf die Interessengeneese an Informatik eine erfolgversprechende Zielgruppe. Kinder und Jugendliche interessieren sich für Neue Medien und Zukunftstechnologien. Dies kann auch aus Studien wie KIM und JIM geschlossen werden, im Rahmen dessen wird untersucht, welche Medien von welchen Kindern und Jugendlichen genutzt werden (MPFS – Medienpädagogischer Forschungsverbund Südwest, 2014a, 2015). Der rasche Anstieg an genutzten Medien, auch bei jüngeren Kindern, zeigt, dass zumindest hinsichtlich der Nutzung von Medien ein großes Interesse besteht. Ob sich Kinder und Jugendliche zugleich für die informatischen Hintergründe interessieren, ist dagegen noch weitgehend offen. Die Begeisterung, die Kinder zeigen, wenn sie kleine Roboter Fußball spielen sehen oder z. B. im Rahmen des ‚Roberta‘-Projekts selbst programmieren dürfen, kann aber vielleicht als ein Zeichen dafür gewertet werden, dass das Interesse sich wecken lässt (Petersen, Theidig, Bördig, Leimbach & Flintrop, 2007).

Yardi und Bruckmann untersuchten in einer qualitativen Studie die Wahrnehmungen von ‚Computing‘ bzw. die Einstellungen zu Informatik bei Kindern und Jugendlichen ab einem Alter von 11 Jahren im Vergleich zu Studierenden der Informatik (Yardi & Bruckman, 2007). Sie stellten fest, dass bei den befragten Teenagern die Vorstellung vorherrschte, Informatik sei langweilig, eine einsame Angelegenheit und ohne Bezug zur realen Welt. Informatik-Studierende dagegen äußerten sich begeistert und fasziniert von den Möglichkeiten: „[...] graduate students described their research as exciting, social, and having a direct and meaningful impact on the world around them“ (Yardi & Bruckman, 2007, S. 39).

## 2.3 Zugänge zur Informatik für Kinder

Wie im vorherigen Abschnitt festgestellt wurde, wenden sich Untersuchungen im Kontext der informatischen Bildung zunehmend Jugendlichen und Kindern jüngerer Alters zu. Einhergehend mit diesem Forschungsinteresse entstehen Initiativen und Projekte, die schulische und außerschulische Lernangebote zur



Beschäftigung mit Informatik zur Verfügung zu stellen. Solche Angebote verfolgen insbesondere drei Ziele:

- a) frühzeitige Interessenanregung für Informatik,
- b) Vermittlung eines Grundverständnisses für Phänomene der durch Informatiksysteme geprägten Welt (digitale Gesellschaft) und
- c) Förderung informatischen Denkens (auch „Computational Thinking“).

Über verschiedene Zugänge werden unterschiedliche Motivationen zur Beschäftigung mit Informatik in den Vordergrund gestellt. Sie unterscheiden sich vor allem darin, ob Computer/Informatiksysteme im Lernprozess als Werkzeug verwendet werden oder ob sich der Zugang auf nicht-informationstechnisches Lernmaterial (unplugged<sup>4</sup>) beschränkt. Die Verwendung von Computern lässt sich grob unterscheiden in Zugänge unter der Verwendung von Software auf universellen Computern (wozu auch z. B. Tablets zählen) und in Zugänge, die programmierbares Spielzeug, also spezielle Informatiksysteme für Kinder, verwenden. Als weiteren Zugang werden außerschulische Lernorte und Communities vorgestellt, die Computer oder Unplugged-Zugänge verwenden, dabei aber das gemeinsame, in der Regel kollaborative Erlebnis als Motivation und Leitlinie des Lernens aufgreifen.

### **2.3.1 Zugänge ohne Computer**

Die Informatik stellt als Disziplin Konzepte, Methoden und Arbeitsweisen zur Verfügung, die sich in unterschiedlicher Form auch ohne unmittelbaren Bezug zu Informatiksystemen in Lehr-Lern-Materialien wiederfinden. Gegenüber den vielen softwarebasierten Werkzeugen für den Einstieg in die Informatik ist das Anliegen der sog. Unplugged-Materialien, den Kindern vor allem die dem informatischen Problemlösen zugrunde liegenden Ideen und Strategien zu verdeutlichen und dabei Phänomene der digitalen Welt zu erklären. Allen Materialien/Zugängen ist gemeinsam, dass für die Durchführung kein Computer notwendig ist. Gleichzeitig wird in den Materialien verdeutlicht, dass die vorgestellten informatischen Denkweisen dem Alltagsdenken entspringen (vgl. auch „Computational Thinking“). Der Zugang erfolgt auf eine meist spielerische und handlungsorientierte Art und Weise mit dem Ziel, Kinder und Jugendliche unterschiedlicher Altersstufen zu erreichen. In den letzten Jahren sind einige wenige Bücher veröffentlicht worden, die versuchen, mit einem vergleichbaren Ansatz Kinder ab dem Kindergartenalter zu erreichen.

Neben diesen aus dem Informatikkontext stammenden Materialien fördern auch Ansätze angrenzender Disziplinen, z. B. der frühen mathematischen Bil-

dung, Kompetenzen, die sich ebenso der frühen informatischen Bildung zuordnen lassen.

### *CS Unplugged*

Das von den neuseeländischen Informatikern Bell, Witten und Fellows 1998 herausgegebene Buch „Computer Science Unplugged ... off-line activities and games for all ages“ (Bell, Witten & Fellows, 1998) war Namensgeber und internationaler Vorreiter darin, Kindern Ideen der Informatik ohne Computer zu vermitteln. Mit den im Buch vorgestellten Aktivitäten und Spielen wenden sich die Autoren insbesondere an Grundschul Kinder im Alter von 5 bis 12 Jahren, indem ein breites Themenspektrum der Informatik unter Verwendung von Alltagsmaterialien wie Spielkarten, Schnüre, Stifte und anderen Haushaltsmaterialien erklärt wird. Die Themen umfassen u. a. Algorithmen, künstliche Intelligenz, Binärzahlen, logische Schaltungen, Kompression, Kryptographie, Informationsdarstellung und Parallelverarbeitung.

Die Beispiele können auch dazu dienen, ähnlich den aus den Naturwissenschaften bekannten Vorführ- und Mitmachexperimenten, in einer „Informatik-Show“ für Familien Ideen der Informatik auf anregende und Spaß vermittelnde Art und Weise zu demonstrieren (vgl. Bell, 1999). Im deutschsprachigen Raum sind die Ideen und Materialien des CS-Unplugged-Ansatzes u. a. im Informatikjahr im Projekt „Einstieg Informatik“ aufgegriffen und teilweise weiterentwickelt worden, insbesondere auch mit Blick auf Kinder ab 5 Jahren (Pohl, Kranzdorf & Hein, 2007). Unter dem Titel „Abenteuer Informatik“ entwickelte Gallenbacher eine Mitmachausstellung zur Informatik, die von Kindern ab der dritten Klasse „bespielt“ und verstanden wurde (vgl. Gallenbacher, 2009).



**Abbildung 14.** Ein Beispiel für CS Unplugged: Sortieren nach Gewicht mittels Balkenwaage: Da ein intuitives Ordnen wie mit Zahlen nicht möglich ist, werden aufwändige Vergleichsoperationen explizit durchgeführt.

CSUnplugged zählt inzwischen zu den populärsten und einflussreichsten Ansätzen für einen spielerischen Einstieg in die Informatik ohne Computereinsatz der Informatik-

didaktik weltweit (siehe Abbildung 14). Er wird in schulischen und außerschulischen Lernsituationen verwendet, wird international in Curricula zur Informatik herangezogen und ist in verschiedenen Studien evaluiert worden. So konnte u. a. festgestellt werden, dass sich vor allem jüngere Kinder auf die Ideen einlassen und Spaß daran haben, diese weiter zu erforschen (vgl. Bell, Rosamond & Casey, 2012), es mit CS Unplugged gelingen kann, bei Viertklässlerinnen und Viertklässlern Interesse für Informatik zu wecken (Lambert & Guiffre, 2009), und auch Lehrkräften der Zugang hilft, Ideen der Informatik zu vermitteln (Morreale & Joiner, 2011). Fraglich bleibt allerdings, ob CS Unplugged tatsächlich langfristig die Ziele erreicht, Kinder zur weiteren Beschäftigung mit Informatik zu motivieren und die Sichtweise dahingehend zu ändern, dass der Computer zwar Werkzeug, aber nicht Mittelpunkt der Informatik darstellt (Taub, Ben-Ari & Armoni, 2009).

#### *Beispiele der Informatik in Materialien zur frühen Bildung*

Die wissenschaftliche Diskussion um informatische Bildung in der frühen Bildung ist noch sehr jung. Dennoch existieren bereits verschiedene Materialien zur frühen Bildung und Grundschulbildung, die sich mit Themen der Informatik auseinandersetzen. So finden sich beispielsweise im Angebot der Stiftung „Haus der Kleinen Forscher“ verschiedene Aktivitäten, die inhaltlich klare Bezüge zu Gegenständen der Informatik aufweisen und teilweise dem CS-Unplugged-Ansatz entlehnt sind (vgl. Abbildung 15). Hierbei handelt es sich zum einen um Materialien, die thematisch unmittelbar der Informatik zuzuordnen sind (z. B. „Digitale Kommunikation“<sup>19</sup>, „Einen Roboter steuern“<sup>20</sup>), und zum anderen um Materialien, die Sachverhalte und Vorgehensweisen beschreiben, die sowohl in anderen Wissenschaften als auch in der Informatik von Bedeutung sind. So finden sich z. B. im Lehr-Lern-Material zum Thema „Mathematik entdecken: Modelle und Karten – Vom Gegenstand zum Symbol“<sup>21</sup> Aspekte der informatischen Modellierung und Repräsentation von Information wieder. In der Themenbroschüre „Kommst Du mit die Zeit entdecken?“ (Stiftung Haus der kleinen Forscher 2013a) werden die in der Informatik wichtigen Themen Aufwandsabschätzung, Effizienz und Optimierung untersucht, und das Material zum Schwerpunktthema Mathematik „Spannende

19 In Themenbroschüre „Kannst Du mich verstehen? Die Vielfalt der Kommunikation erkunden und erforschen“ (Stiftung Haus der kleinen Forscher, 2014).

20 Experiment im Rahmen der Praxisanregungen: <http://www.haus-der-kleinen-forscher.de/de/praxisanregungen/experimente-themen/lebensraeume/experiment/einen-roboter-steuern/>

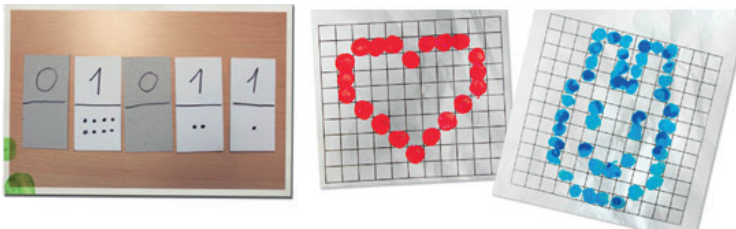
21 Vgl. [http://www.haus-der-kleinen-forscher.de/fileadmin/Redaktion/6\\_Experimente/Mathematik/Downloads/MATHE\\_\\_KARTENSET\\_6.2014.pdf?pk\\_campaign=Newsletter%20August%202014&pk\\_kwd=Karten-Set-Mathe](http://www.haus-der-kleinen-forscher.de/fileadmin/Redaktion/6_Experimente/Mathematik/Downloads/MATHE__KARTENSET_6.2014.pdf?pk_campaign=Newsletter%20August%202014&pk_kwd=Karten-Set-Mathe) [Zugriff am 05.03.2018].



Wiederholungen“<sup>22</sup> betrifft z. B. die informatischen Ideen Algorithmus und Iteration.

Informatiknahe Aufgaben lassen sich auch in anderen Lernmaterialien für Kinder finden, etwa im Kontext von logischem Denken. So adressiert das gleichnamige Buch der Reihe „Kindergarten-Lernraupe“ für Kinder ab 3 Jahren (Wiesner, 2008) z. B. die Beschreibung von Abläufen (Algorithmisierung) und die Identifizierung der Eingaben eines Verarbeitungsprozesses (EVA-Prinzip) (vgl. Abbildung 16).

In diesem Zusammenhang ist zu konstatieren, dass nur unzureichende Forschungsergebnisse im Bereich solcher elementaren, der Informatik zuzuordnen den Fähigkeiten und hinsichtlich Vorläuferfähigkeiten vorliegen (wie sie z. B. für die Mathematik existieren).



**Abbildung 15.** CS-Unplugged-Aufgaben in den Materialien der Stiftung „Haus der kleinen Forscher“: Auf Papier werden Binärzahlen (links) und die Bilddarstellung mit Rastergrafiken (rechts) dargestellt (Stiftung Haus der kleinen Forscher, 2014).



**Abbildung 16.** Informatiknahe Aufgaben in der „Lernraupe“ (Wiesner, 2008).

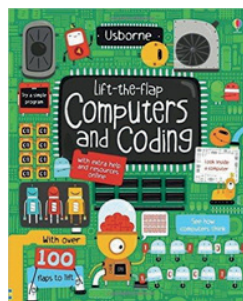
22 Vgl. [http://www.haus-der-kleinen-forscher.de/uploads/tx\\_hdkfexp/110831\\_Spannende\\_Wiederholungen\\_Web.pdf](http://www.haus-der-kleinen-forscher.de/uploads/tx_hdkfexp/110831_Spannende_Wiederholungen_Web.pdf) [Zugriff am 05.03.2018].

### *Kinderliteratur zur Informatik*

Erst seit relativ kurzer Zeit wird auch in der Kinderliteratur Informatik explizit thematisiert. Bücher hierzu gibt es vor allem im englischen Sprachraum. „Hello Ruby: Adventures in Coding“ (Liukas, 2015; siehe Abbildung 17) versucht, zum einen Computational Thinking, zum anderen aber auch Grundlagen des Aufbaus von Informatiksystemen, des Programmierens sowie Begriffe und Phänomene aus der IT-Welt auf Kindergartenniveau zu vermitteln. Neben der altersgerecht gestalteten Geschichte beinhaltet das Buch verschiedene Übungen und Bastelmaterialien. An Grundschulkinder richtet sich das Buch „Lift-the-Flap Computers and Coding“ (Dickins, Nielsen, Barden & Lamont, 2015; siehe Abbildung 18), das kindgerecht den Aufbau von Hardware sowie einzelne Phänomene rund um den Computer (Programme, Binärsystem, Zeichencodierung etc.) erklärt.



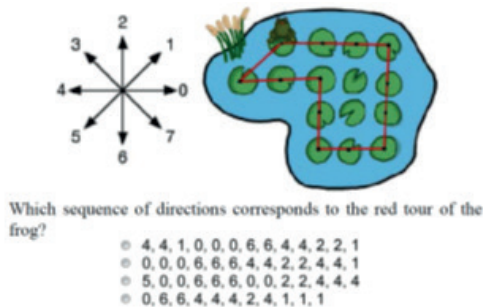
**Abbildung 17.** „Hello Ruby“ vermittelt auf Kindergarteniveau Konzepte der Informatik mit besonderem Fokus auf „Computational Thinking“ (Liukas, 2015; <http://www.helloruby.com/press>)



**Abbildung 18.** Aufklappbuch „Lift-the-Flap Computers and Coding“.

### Informatik-Biber für Kinder

Der jährlich stattfindende internationale Wettbewerb Informatik-Biber<sup>23</sup> (international „Bebras“) wird seit dem Jahr 2016 auch in Deutschland für Kinder ab Klassenstufe 3 angeboten. Auch wenn es sich hierbei um einen Online-Wettbewerb handelt, also zur Darstellung der Aufgaben und zur Eingabe der Lösungen ein Computer verwendet wird, zeichnet sich der Informatik-Biber vor allem durch seine zum großen Teil ohne Computer durchführbaren Knobelaufgaben aus. In anderen Ländern, z. B. der Slowakei seit 2010, wird der Wettbewerb bereits seit einigen Jahren in der Grundschule erfolgreich eingesetzt (vgl. Gujberova & Kalas, 2013). Gekennzeichnet ist der Wettbewerb durch eine Sammlung von jeweils in wenigen Minuten zu bearbeitenden, altersgemäßen, teilweise interaktiven Aufgabenstellungen der Informatik, die die Vielfalt der Informatik vermitteln, einen ersten Kontakt herstellen und zur weiteren Beschäftigung mit Informatik anregen sollen (vgl. Abbildung 19). Die im Wettbewerb gesammelten Daten können zukünftig interessante Forschungsergebnisse hinsichtlich der kognitiven Leistungsfähigkeit von Kindern bei der Bearbeitung solcher Informatikaufgaben liefern.



**Abbildung 19.** Beispielaufgabe im Informatikbiber für die Grundschule (Gujberova & Kalas, 2013).

### 2.3.2 Softwarebasierte Einstiege in die Programmierung

Oftmals sind es Werkzeuge, die bei Kindern Neugier und Interesse für die Beschäftigung mit Informatik wecken. Interaktivität, direkte Rückmeldungen und umfangreiche Gestaltungsmöglichkeiten machen Informatiksysteme zu attraktiven Lernmedien, die zahlreich auch für die informatische Bildung entwickelt wurden. Insbesondere existieren viele Softwarewerkzeuge, die Kindern den Einstieg in die Informatik über die Programmierung ermöglichen. Kelleher und Pausch analysieren Werkzeuge zum Erlernen der Programmierung und unterscheiden diese hinsichtlich zweier unterschiedlicher, mit dem Programmieren verbundener Ziele: „Teaching systems“ helfen dabei, möglichst korrekt programmieren zu lernen

<sup>23</sup> <http://informatik-biber.de/>

(Kelleher & Pausch, 2005). „Empowering systems“ sollen die Lernenden vor allem darin unterstützen, Programmieren als kreatives Werkzeug verwenden zu können. Die im Folgenden beschriebenen Werkzeuge sind dieser zweiten Kategorie zuzuordnen. Sie zeichnen sich dadurch aus, dass die Komplexität sinnvoll reduziert wird, z. B. durch die Verwendung visueller Bausteine, die keine Syntaxfehler zulassen.

### *Hintergrund*

Bereits in den 1960er-Jahren wurden Computer in Lernkontexten verwendet. Papert (1987) stellte fest, dass sich diese Anwendungen insbesondere drei Bereichen zuordnen lassen: (1) Tutorials, in denen der Computer als Instruktor fungiert, (2) Softwarewerkzeuge wie Taschenrechner, Textverarbeitung oder Simulationen und (3) Mikrowelten, die als Realitätsausschnitt eine neue Art des Lernens ermöglichen, die erst durch Computertechnologie realisierbar ist. Ein großer Teil der für den Einstieg in die Informatik zur Verfügung stehenden Softwarewerkzeuge fußt auf der Idee der Mikrowelten, die Papert definierte als

„subset of reality or a constructed reality whose structure matches that of a given cognitive mechanism so as to provide an environment where the latter can operate effectively. The concept leads to the project of inventing microworlds so structured as to allow a human learner to exercise particular powerful ideas or intellectual skills“ (Papert, 1980, S. 240).

Mikrowelten stellen einen überschaubaren virtuellen Lernraum dar, in dem Kinder die vorhandenen virtuellen Objekte erkunden und in den gegebenen Möglichkeiten weiterentwickeln können. Programmierung wird hierbei als intellektuelles Werkzeug in der Erkundung und für das Verständnis der oft abstrakten Sachverhalte verstanden.

Mit der Programmiersprache Logo (Feurzeig, Papert & Lawler, 1970) wurde eine Mikrowelt für das Programmieren erschaffen, zu der umfangreiche Berichte zum Einsatz in der Grundschule aus den 1980er-Jahren vorliegen (z. B. Hoppe & Löthe, 1984; Ziegenbalg, 1985). Ziel von Logo war zunächst weniger das Vermitteln informatischer Bildung, sondern eines Verständnisses für mathematische und geometrische Strukturen im Sinne des konstruktionistischen Lernansatzes als Erweiterung der konstruktivistischen Lerntheorie. Papert unterstreicht hierin die Bedeutung des handelnden Lernens anhand von konkreten, persönlich bedeutsamen Gegenständen: „[Learning] happens especially felicitously in a context where the learner is consciously engaged in constructing a public entity, whether it's a sand castle on the beach or a theory of the universe“ (Papert & Harel, 1991, S. 1). Gegenüber dem Konstruktivismus wird hervorgehoben, dass

die Wissenskonstruktion nicht nur im Kopf der Lernenden geschieht, sondern auf einem tatsächlichen Konstruktionsprozess in der realen oder virtuellen Welt basieren sollte („learning by making“ statt „learning by doing“). Daraus ergibt sich als Vorteil für die Lernenden, dass das entstehende Produkt ausprobiert, gezeigt, diskutiert, analysiert und auch bewundert werden kann und die Lernenden die Möglichkeit haben, hierbei „eins“ mit dem betrachteten Phänomen zu werden, statt es nur „von außen“ zu sehen. Mithilfe von Mikrowelten als Lernumgebungen auf Computern können solche „Lernerfahrungen stattfinden, unbehindert von den Komplexitäten der realen Welt“ (Papert, 1998, S. 66). So werden Mikrowelten zu „Brutwelten für Wissen“ und können als „Gewächshaus für eine bestimmte Spezies tragender Ideen oder intellektueller Strukturen“ (Papert, 1982, S. 157) dienen.

Basierend auf dem Vorbild „Logo“ wurden in den 1980er- und 1990er-Jahren verschiedene weitere Mikrowelten entwickelt, von denen einige bis heute im Informatikunterricht vor allem in der Sekundarstufe I verwendet werden (z. B. Robot Karol, Kara, Java-Hamster (Boles, 2005; Reichert, Nievergelt & Hartmann, 2005)). Als problematisch stellt sich allerdings die Eindimensionalität vieler dieser Mikrowelten dar: So entspricht es einem fragwürdigen Informatikbild, wenn Programmierung und Informatik vor allem dazu dienen sollen, Marienkäfer oder Roboter durch Labyrinth zu bewegen, um diverse Gegenstände aufzusammeln. Ebenso verletzen einige der Mikrowelten die zugrunde liegenden Ideen: Während beim konstruktionistischen Lernen anhand einer Mikrowelt Lernende persönlich bedeutsame Produkte erschaffen, sind die Möglichkeiten in vielen Mikrowelten derart eingeschränkt, dass nicht ein Produkt erschaffen, sondern nur eine Lösung für gegebene, oft künstliche Probleme gefunden werden muss.

Viele in jüngerer Zeit entwickelte Werkzeuge beheben diesen Mangel, so z. B. die Programmierumgebung ‚Scratch‘ (s. u.). Im Unterrichtseinsatz lassen sich bei den neueren Werkzeugen u. a. Vorteile für die Motivation der Schülerinnen und Schüler nachweisen (vgl. Ruf, Mühling & Hubwieser, 2014).

### *Scratch/ScratchJr*

Ein gutes Beispiel für einen kindgerechten Einstieg in die Programmierung stellt die auf den Ideen Paperts basierende Entwicklungsumgebung Scratch<sup>24</sup> (vgl. Abbildung 20) dar, die insbesondere folgende Prinzipien verwirklicht: Low Floors, High Ceilings, Wide Walls (vgl. Resnick & Silverman, 2005), d. h. bei möglichst geringen Einstiegshürden sollen (schrittweise) auch komplexe anspruchsvolle Projekte realisierbar sein und unterschiedliche Interessen von Kindern angesprochen werden. Erreicht wird die geringe Einstiegshürde durch eine weitgehend intuitiv erfassbare visuelle Programmiersprache, die keine Syntaxfehler erlaubt –

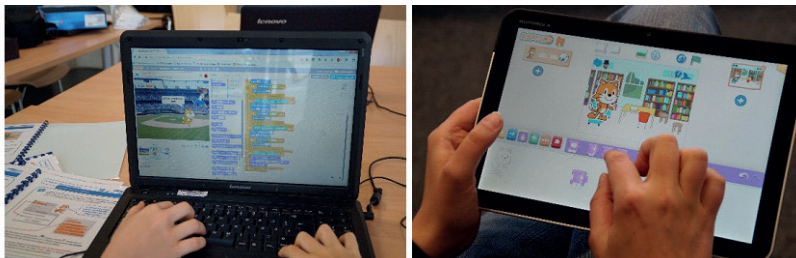
---

<sup>24</sup> <http://scratch.mit.edu>

eine der Haupthürden für Programmieranfängerinnen und -anfänger (vgl. Myers, 1990). Die Programme sind jederzeit ausführbar, so dass die Lernenden zu jedem Zeitpunkt den Erfolg ihrer Arbeit selbst beurteilen können.

Gegenüber älteren Mikrowelten ermöglicht Scratch die Umsetzung unterschiedlichster multimedialer Projekte: Animationen und Geschichten erzählen, Spiele, Simulationen und vieles mehr. Zusätzlich können Projekte in einer Online-Galerie geteilt und kollaborativ weiterbearbeitet werden. Damit war es den Entwicklerinnen und Entwicklern von Scratch weniger ein Anliegen, Programmierinnen und Programmierer auszubilden, als Kinder zu einem kreativen Umgang mit digitalen Medien zu befähigen (vgl. Resnick et al., 2009). Der von Scratch gewählte Ansatz hat sich als sehr erfolgreich erwiesen: Innerhalb kürzester Zeit hat sich die Verwendung der Programmierumgebung weltweit verbreitet.

Dem Beispiel folgend wurden verschiedene andere Programmierumgebungen entwickelt, die mit einem ähnlichen Ansatz, aber teilweise anderen Ausrichtungen Kindern das Programmierenlernen ermöglichen wollen, etwa bei der Entwicklung von Apps (z. B. mit App Inventor<sup>25</sup>) oder mit Online-Denkspielen (z. B. Blockly<sup>26</sup>). Da die grafische Darstellung der Programmierkonstrukte auch Vorteile für elaboriertere Programmierung bieten kann, werden derzeit verschiedene, auf der Idee von Scratch basierende Programmierumgebungen entwickelt, die im fortführenden Unterricht und für Gelegenheitsprogrammiererinnen und -programmierer verwendet werden können (z. B. Snap<sup>27</sup>, GP<sup>28</sup>). Seit 2012 wird versucht, mit dem Ansatz bereits Kinder ab 5 Jahren zu erreichen, indem ein altersgerechtes Userinterface mit Touch-Bedienung und ausschließlich ikonischer Darstellung entwickelt wurde, für das Kinder keine Lesekompetenz benötigen (Flannery et al., 2013).



**Abbildung 20.** Visuelle Programmierung mit Scratch-Bausteinen (links) und in ScratchJr (rechts)

<sup>25</sup> <http://appinventor.mit.edu>

<sup>26</sup> <https://blockly-games.appspot.com/>

<sup>27</sup> <http://snap.berkeley.edu/>

<sup>28</sup> [http://scratch-dach.info/wiki/GP\\_\(Programmiersprache\)](http://scratch-dach.info/wiki/GP_(Programmiersprache)) [Zugriff am 05.03.2018]

### *Online-Tutorials*

Mit dem Ziel, Nachwuchsprogrammierer zu motivieren und dabei einen möglichst einfachen Zugang zu bieten, stellen verschiedene Websites Sammlungen angeleiteter Aktivitäten und Lerneinheiten im Kontext des Programmierens zur Verfügung, die auch für Schulen interessant sind (<https://code.org/>, <http://start-coding.de/>). Populär wurden diese Kurzeinblicke in die Programmierung mit der Aktion „Hour of code“, die bis 2015 fast 100 Millionen Menschen weltweit erreichte (Wilson, 2015).

### **2.3.3 Programmierbares Spielzeug (Robotik)**

McNerney berichtet von Untersuchungen Seymour Paperts in den 1970er-Jahren mit dem Ziel, Programmieren mit der physischen Welt zu verknüpfen, um Kindern die Welt der Informatik nahezubringen (McNerney, 2004). Da Grundschul Kinder noch nicht gut in der Lage waren, selbst Programmieranweisungen per Tastatur zu schreiben, wurde die Logo-Schildkröte prototypisch als malender Roboter implementiert und alternative physische Möglichkeiten der Programmierung erprobt, z. B. mit Knöpfen und Steckkarten (vgl. Perlman, 1976). Die Zugänglichkeit über physische Erfahrungen ist seitdem regelmäßig Gegenstand der Forschung. So zeigt es sich als Vorteil, dass eine direkte Interaktion mit physischen Objekten für Kinder einfacher und damit weniger frustrierend ist (Xie, Antle & Motamedi, 2008). Daher erstaunt es nicht, dass zunehmend Versuche unternommen werden, gerade für jüngere Kinder „tangible programming interfaces“ zu entwickeln (Brauner, 2009; Leonhardt, 2015). Als solches kommerziell verfügbar sind heutzutage direkt am Gerät programmierbare Spielzeuge wie z. B. der Bee-Bot<sup>29</sup>.

Als Weiterentwicklung der physischen Logo-Schildkröte sind Robotikbaukästen wie z. B. LEGO Mindstorms<sup>30</sup> oder WeDo<sup>31</sup> erfolgreich, aus denen in letzter Zeit Weiterentwicklungen für programmierbare, interaktive Kreativbaukästen hervorgegangen sind (beispielsweise Pico Crickets; vgl. Resnick, 2007). Gemeinsam ist diesen Werkzeugen, dass sie eine Reihe von Sensoren (z. B. für Lautstärke, Helligkeit, Entfernung) bereitstellen, deren Daten über eine Programmierschnittstelle erfasst und verarbeitet werden können, um damit wiederum Aktionen, z. B. durch die Steuerung von Motoren, LEDs und Lautsprechern hervorzurufen.

Vor dem Hintergrund des zunehmenden Verschwindens von Desktopcomputern und der Allgegenwart von Computern als sog. eingebettete Systeme in Alltagsgegenständen stellen sich solche Werkzeuge als spannend und zeitgemäß für die informatische Bildung dar, da Kinder damit lernen können, Computertechnologie im Alltag zu entdecken und auch selbst, z. B. als interaktives Spielzeug,

---

29 <https://www.bee-bot.us>

30 <https://www.lego.com/de-de/mindstorms> [Zugriff am 05.03.2018]

31 <https://education.lego.com/de-de/grundschule/shop/wedo-2> [Zugriff am 05.03.2018]

zu gestalten. Im Folgenden wird auf verschiedene Vertreter von Robotik-Kits und programmierbarem Spielzeug näher eingegangen.

#### *LEGO Mindstorms & WeDo*

LEGO Mindstorms haben als konkrete Umsetzung der konstruktionistischen Idee im Informatikunterricht weite Verbreitung gefunden (vgl. Wiesner & Brinda, 2007). Mit ihnen wird die Idee der Mikrowelten von der virtuellen auf die reale Welt übertragen. Kritisiert wird an Mindstorms-Robotern, dass die Konstruktionen schnell sehr komplex und damit für jüngere Kinder schwer erfassbar sind. Borowski und Diethelm vermuten, basierend auf eigenen Erfahrungen, dass dieser Zugang noch nicht in vollem Umfang für Kinder im Alter von 10 Jahren geeignet ist (Borowski & Diethelm, 2009). Auch wenn mit LEGO Mindstorms sehr unterschiedliche Konstruktionen gebaut werden können, kommen häufig nur Roboterfahrzeuge zum Einsatz (vgl. Wiesner & Brinda, 2007). LEGO Mindstorms zählen in Deutschland zu den am weitesten verbreiteten programmierbaren Robotern, die sowohl im schulischen wie auch außerschulischen Bereich eine breite Anwendung finden (Hartmann & Schecker, 2005; Leonhardt, 2015).

Eine einfachere Variante des LEGO Mindstorms Systems stellt LEGO WeDo dar, das für Kinder ab 7 Jahren empfohlen wird. Zum Einsatz gibt es noch wenig Erfahrungsberichte. Mayerová beobachtete in der Arbeit mit Drittklässlerinnen und Drittklässlern, dass die Kinder grundsätzlich in der Lage waren, die gestellten Aufgaben zu lösen, teilweise aber mehr Zeit mit dem Zusammenstecken des mechanischen Modells als mit der Programmierung verbringen (Mayerová, 2012).

#### *PicoCrickets*

PicoCrickets<sup>32</sup> können als Neuentwicklung, basierend auf den Erfahrungen mit LEGO-Mindstorms-Robotern, verstanden werden. Mit PicoCrickets wird die Idee programmierbarer Bausteine auf die Welt der allgegenwärtigen Computer ausgedehnt. PicoCrickets sind kleine programmierbare Bausteine, an welche Sensoren (Licht, Berührung, Widerstand, Geräusch) als Eingabequellen angeschlossen werden können. Kinder können mit diesen Bausteinen Kreationen erstellen, die sich bewegen, leuchten, Musik machen und vieles mehr. Mit der einfach gehaltenen visuellen Programmiersprache PicoBlocks werden dafür die Ausgabemöglichkeiten (farbige Lichter, Motoren, Soundgenerator und LED-Display) angesteuert. Hieraus ergibt sich eine Vielzahl an realisierbaren Projekten, beispielsweise interaktive Gärten, reagierende Plüschtiere, oder „Technokleidung“, etwa die von Resnick beschriebenen Stiefel, die passend zur Schrittgeschwindigkeit farbig blinken (Resnick, 2007). Zum Umfang der PicoCrickets gehört ein Set mit LEGO-Bausteinen

---

<sup>32</sup> <http://www.picocricket.com>



und Bastelmaterialien. Die Trennung von elektronischer Welt und Bastelwelt soll damit durchbrochen werden. Dies ermöglicht Kindern, sich auf verschiedenste Arten in Kontexten aus ihrer Erfahrungswelt informatiknah und kreativ zu betätigen. So sind PicoCrickets nicht unmittelbar eine Form der Robotik, sondern werden als „invention kit that integrates art“ vorgestellt:

„You can plug lights, motors, and sensors into a Cricket, then write computer programs to tell them how to react and behave. With Crickets, you can create musical sculptures, interactive jewellery, dancing creatures, and other artistic inventions – and learn important math, science, and engineering ideas in the process“ (MIT, 2011).

Positive Erfahrungen mit dem Einsatz von Pico Crickets in der 4. Klasse einer Grundschule wurden von den Autoren selbst gesammelt (Romeike & Reichert, 2011).

#### *Bee-Bot*

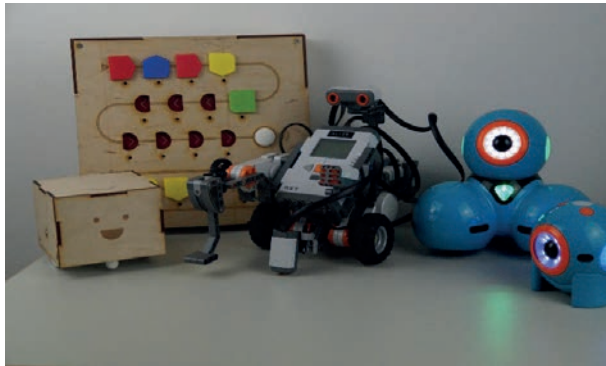
Der Bee-Bot ist ein fahrbarer Roboter, der speziell für junge Kinder entwickelt wurde und optisch mit den schwarz-gelben Streifen an eine Biene erinnert (siehe Abbildung 21). Als Eingabe stehen sieben Knöpfe zur Verfügung. Zur Steuerung gibt es je einen für eine Vorwärts- und eine Rückwärtsbewegung, außerdem jeweils einen für die Rechts- bzw. Linksdrehung um 90 Grad. Nach der Eingabe von maximal vierzig dieser vier Befehle in beliebiger Kombination kann der Bee-Bot mit Drücken der „Go“-Taste gestartet werden. Die beiden verbleibenden Knöpfe dienen dem Anhalten und dem Zurücksetzen der Anweisungen. Was man bei allen Anwendungen mit dem Bee-Bot lernen und üben kann, ist die gedankliche Ausführung von Befehlen. Da alle Anweisungen vor dem Drücken der „Go“-Taste eingegeben werden müssen, wird immer zuvor eine Strategie ausgearbeitet und diese in eine Befehlssequenz umgesetzt.



**Abbildung 21.** Der programmierbare Bodenroboter Bee-Bot

### Weitere Robotersysteme: Cubetto, Dash & Dot

Auch der PRIMO-Roboter Cubetto (siehe Abbildung 22 links) soll insbesondere jungen Informatikeinsteigerinnen und Informatikeinsteigern die ersten Schritte erleichtern. So lässt sich dieser über das Stecken von Holzklötzchen programmieren. Mit diesem einfachen Lern-Spielzeug können sich bereits kleine Kinder Grundlagen hinsichtlich des Verstehens von Algorithmen (bzw. Programmabläufen) und Programmierung spielerisch aneignen.



**Abbildung 22.** Verschiedene Robotersysteme für Kinder (von links nach rechts): PRIMO Cubetto, LEGO Mindstorms, WONDER-Roboter Dash & Dot

Besonders bei jungen Lernenden ist eine direkte Rückmeldung zu ihrer Arbeit sehr motivationsfördernd (Leonhardt, 2015). So bieten auch die etwas komplexeren Robotermodelle wie die WONDER-Roboter Dash & Dot (siehe Abbildung 22, rechts), die über Symbole auf einem Tablet gesteuert werden, die Möglichkeit, das Ergebnis der eigenen Programmierversuche direkt selbst zu bestaunen und zu bewerten.

### 2.3.4 Außerschulische Lernorte und Communities

Während die oben dargestellten Zugänge sich an konkreten Beispielen und Themen (CS Unplugged) oder Werkzeugen orientieren, wird bei anderen Zugängen das soziale und kollaborative Erlebnis betont, um Kinder und Jugendliche für Informatik zu motivieren, insbesondere im außerschulischen Bereich. Dabei bieten verschiedene Arten der Online- und Offline-Communities Kindern und Jugendlichen die Möglichkeit des Austauschs mit anderen Interessierten und auch ein gemeinschaftliches Lernerlebnis.

### *Computer Clubhouses*

Ein Beispiel hierfür sind die *Computer Clubhouses*<sup>33</sup>, welche auch jungen Kindern bereits Mitmach-Angebote im Bereich Lernen mit Computern ermöglichen. Das Projekt der „Computer Clubhouses“ verfolgt das Ziel, Schülerinnen und Schülern ein kreatives Lernen mit Computern zu ermöglichen (Resnick & Rusk, 1996). Dem Clubhouse-Lernansatz liegen vier Leitprinzipien zugrunde: Lernen durch Design, Folgen eigener Interessen, Bilden einer Gemeinschaft und Gestalten einer Umgebung des Respekts und Vertrauens. Auch wenn informatische Bildung nicht als primäres Ziel des Clubhouse-Lernansatzes benannt wird, sondern vielmehr interkulturelle Verständigung und ein tolerantes Miteinander angestrebt werden, so lernen die Teilnehmerinnen und Teilnehmer hier quasi „nebenbei“ grundlegende Verfahren, Konzepte und Anwendungen der Informatik.

### *Schülerlabore*

Eine weitere wachsende Institution im Bereich informatischer Bildung sind *Schülerlabore* wie das Schülerlabor Informatik *InfoSphere*<sup>34</sup> an der RWTH Aachen, welches sich ausschließlich die Vermittlung informatischer Aspekte an Kinder und Jugendliche ab Klassenstufe 3 zum Ziel gesetzt hat (Bergner, 2015). Schülerlabore bieten häufig sowohl kompletten Schulklassen als auch einzelnen Interessierten Workshopangebote.

### *Wettbewerbe*

Neben dem bereits in Abschnitt 2.3.1 dargestellten Wettbewerb Informatik-Biber gibt es verschiedene weitere Angebote. So stellt der Bundeswettbewerb Informatik<sup>35</sup> fortgeschrittene Informatikinteressierte vor neue Herausforderungen. Weiterhin gibt es zahlreiche regionale, aber auch internationale Roboterwettbewerbe, welche teils direkt von den Firmen (z. B. LEGO) aber auch durch (unterschiedlich große und professionelle) Gruppen, Netzwerke oder gar Schulen selbst angeboten werden.

### *Online Communities*

Brennan (2013) beobachtete, dass nicht nur der Prozess des Erstellens von Scratch-Projekten einen wesentlichen Grund für Kinder darstellt, sich mit Programmieren zu beschäftigen, sondern auch die durch die Plattform zur Verfügung gestellte Möglichkeit, sich mit anderen zu verbinden und auszutauschen. Die Scratch-Community umfasst inzwischen hunderttausende Mitglieder, überwiegend im Alter von 8 bis 16 Jahren, die täglich mehr als 2000 Projekte auf die

---

<sup>33</sup> <http://www.computerclubhouse.org/>

<sup>34</sup> <http://schuelerlabor.informatik.rwth-aachen.de/>

<sup>35</sup> <http://www.bwinf.de/>

Website laden, die dort kommentiert, gelobt und weiterbearbeitet werden können (Brennan, 2013). Der Quellcode der Projekte kann von jedem eingesehen werden, so dass Lernen voneinander möglich ist. Kafai und Burke (2014) betonen, dass nach ihren Beobachtungen weniger die Werkzeuge als die soziale Erfahrung für das Programmierenlernen von Bedeutung sind. Programmieren stellt sich zunehmend als ein hochsoziales und nicht individuelles Unterfangen dar. Der Austausch und vor allem das positive Feedback stärkt offenbar, ähnlich wie bei Präsentationen vor Publikum, die positive Würdigung des eigenen Schaffensprozesses und vermittelt den Kindern eine neue Selbstkompetenz im Umgang mit Technik.

### *Maker Culture*

In der sog. Maker Culture geht es darum, Menschen jeden Alters zu ermöglichen, insbesondere mit neuen Technologien Dinge zu erfinden und zu erschaffen. Typischerweise geschieht dies nicht in der Schule, sondern in Arbeitsgemeinschaften, öffentlichen Einrichtungen (wie z. B. FabLabs; siehe Walter-Herrmann & Büching, 2013), auf Messen und Ausstellungen (Maker Faire<sup>36</sup>) oder privat. Da auf Informatiksystemen basierende Technologien wie 3D-Drucker, Mikrocontrollerboards oder Robotik eine große Resonanz in der Maker Community erfahren, gibt es viele Überschneidungen zu Beispielen der informatischen Bildung (vgl. Libow Martinez & Stager, 2013).

### **2.3.5 Zusammenfassung**

Auch wenn informatische Bildung für Kinder im Kindergarten- und Grundschulalter ein relativ junges Forschungsgebiet darstellt, existieren bereits verschiedene erprobte Zugänge, die Kinder motivieren und Kompetenzen der Informatik vermitteln können. Hinsichtlich der Eignung für die verschiedenen Zielgruppen lassen sich Zugänge finden, die bereits für Kinder im Kindergartenalter intendiert und erprobt sind (beispielsweise Lernmaterialien der frühen Bildung, „Hello Ruby“, Bee-Bot, ScratchJr), sowie Angebote, die sich vor allem an Kinder ab dem Grundschulalter (häufig etwa ab 3. Klasse) richten (beispielsweise CS Unplugged, Informatik-Biber, LEGO WeDo, Scratch, Schülerlabore). Allerdings ist die Forschungslage hierzu als ungenügend zu bezeichnen, da die verschiedenen Erfahrungsberichte über den Praxiseinsatz allenfalls erste Hinweise zur Gestaltung und Wirkung informatischer Bildung in dieser Altersgruppe geben können.

---

<sup>36</sup> Für Deutschland: <https://maker-faire.de/>

## 2.4 Internationaler Vergleich: Curricula und deren Einordnung in das Kompetenzmodell

Im Gegensatz zur Mathematik und den Naturwissenschaften gibt es weder national noch international ein lange etabliertes durchgängiges Pflichtfach Informatik in der Sekundarstufe I. Daher gibt es sehr viel weniger Erfahrungen und Studien, welche informatischen Kompetenzen in welchen Altersstufen erlangt werden können und wie dies zu erreichen ist. Daraus folgt, dass es erst Recht kaum gesicherte Erkenntnisse gibt, welche informatischen oder dafür grundlegenden Kompetenzen in der Primarstufe oder gar im Vorschulalter erworben werden können (vgl. Abschnitt 2.2).

Dennoch gibt es, wie in den vorherigen Abschnitten dargestellt, zum einen Studien, die belegen, dass informatische Konzepte von Kindern erlernt werden können und dass dies im jungen Alter auch sinnvoll ist, und zum anderen konkrete Beispiele und Ansätze, diese Konzepte altersgerecht zu vermitteln.

Darauf basierend etablieren aktuell international einige fortschrittliche Bildungssysteme informatische Bildung in der Sekundarstufe I und in der Primarstufe als durchgängige Pflichtfächer, um Schülerinnen und Schülern informatische Denkweisen und Problemlösekompetenzen mittels informatischer Methoden (international „Computational Thinking“) zu vermitteln. Diese Denkweisen qualifizieren sowohl für eine fundierte und reflektierte Nutzung von als auch die kreative Gestaltung der eigenen Lebenswelt mit Informatiksystemen. Daher werden die Fächer allgemein mit Begriffen wie Informatik (Computer Science, Informatics, Programming, Computing (UK, NZ)), informatisches Denken (Computational Thinking (USA)) oder Digitale Technologien (Digital Technologies (AUS)) bezeichnet. Unabhängig von ihrer Bezeichnung beinhalten die internationalen Initiativen für informatische Bildung zum einen Bedien- und Medienkompetenz (digital literacy und ICT-Grundbildung) sowie zum anderen Problemlösefähigkeiten und technische Gestaltungskompetenzen (Computing, Computational Thinking), die jeweils im Nutzungskontext reflektiert werden müssen (vgl. Dagstuhl-Dreieck, Abbildung 12 aus Kapitel 1).

Die meisten bekannten Ansätze, die jetzt Eingang in inter-



nationale Curricula erfahren, entstammen außerschulischen Aktivitäten (Clubhouses, Summer Camps, Schülerlabore, Coding-Initiativen, Wettbewerbe, Online Communities, programmierbare Spielzeuge). Trotz derartiger, im Einzelnen nachweislich erfolgreicher Maßnahmen (vgl. Abschnitt 2.3) ist es über lange Sicht nicht gelungen, das Interesse an Informatik stabil zu wecken. Im Gegensatz dazu ist dies in den letzten Jahren in den anderen MINT-Disziplinen, die ebenfalls durch außerschulische Maßnahmen unterstützt werden, gelungen. Mathematik und Physik werden stärker gewählt, während Informatik weiter abnimmt (Brown, Sentence, Crick & Humphreys, 2014, S. 3). Der offensichtliche Unterschied zwischen den Disziplinen ist, dass Informatik kein reguläres Schulfach in der Primar- und Sekundarstufe I ist.

Einige der genannten internationalen Prinzipien und Standards sowie Curricula werden im Folgenden analysiert, um für die Expertise international anerkannte Kerninhalte und Kompetenzen zu identifizieren. Anschließend werden sie in der Struktur des Kompetenzmodells für informatische Bildung in der Sekundarstufe I der Gesellschaft für Informatik (GI – Gesellschaft für Informatik e. V., 2008) dargestellt. Aus dieser Analyse und Darstellung kann geschlossen werden, ob und wie sich das Kompetenzmodell für die Sekundarstufe auf eines der Primarstufe abbilden lässt. Ziel ist es, ein Kompetenzstrukturmodell für die Primarstufe zu definieren, das anschlussfähig an das etablierte GI-Modell der Sekundarstufe I ist, für das die informatische Vorbildung in Kita und Grundschule qualifizieren sollte.

#### **2.4.1 Computing in Großbritannien**

Die CAS-Initiative<sup>37</sup> (Computing at School) wurde 2008 von BCS (British Chartered Institute of IT; vergleichbar mit der deutschen Gesellschaft für Informatik) und den IT-Unternehmen Microsoft, Google und Intellect mit der Absicht gegründet, Informatik in die Schule zu bringen. Zuvor wurde in britischen Schulen eher ICT im Sinne einer Nutzungsschulung von Informatiksystemen betrieben. ICT-Curricula waren genau wie in Deutschland wenig erfolgreich (u. a. Empfehlungen der The Royal Society, 2012). Sie haben Kinder und Jugendliche eher davon abgeschreckt, sich für Informatik und deren Methoden zu interessieren und von der Disziplin ein falsches Bild vermittelt.

Die CAS-Initiative umfasste zu Beginn wenige engagierte Lehrkräfte und Akademikerinnen und Akademiker, wuchs aber schnell auf hunderte und inzwischen über 6.000 Mitglieder. Das Konsortium CAS hat zunächst vor allem politischen Einfluss genommen, indem verdeutlicht wurde, dass Informatik eine wissenschaftliche Disziplin ist, die eigene Ideen, Prinzipien, Techniken und Methoden entwickelt, die in weiten Bereichen des Lebens Anwendung finden. Dabei basiert

---

<sup>37</sup> <http://www.computingatschool.org.uk/>

sie auf einer eigenen Art informatischen Denkens (Computational Thinking), die generell zur Lösung von Problemstellungen (auch ohne die Anwendung von Informatiksystemen) herangezogen werden kann und zunehmend Einzug in viele andere Disziplinen findet.

Damit wurde erreicht, dass der bis dahin bestehende ICT-Unterricht (Bedienkompetenzen von Informatiksystemen) hinterfragt und neue nationale Curricula für „Computing“ ab der Grundschule ausgearbeitet wurden und ab dem Jahr 2014 das Fach an Grundschulen eingeführt wurde. Darin wurden nach wie vor wichtige Nutzungskompetenzen (ICT und digital literacy) als wesentliche Grundlagen beibehalten, aber eben um informatische Kompetenzen erweitert, damit Informatiksysteme und in ihnen realisierte Informatikkonzepte auch verstanden und reflektiert eingesetzt werden können, um selbst kreativ Lösungen für vorliegende Aufgabenstellungen zu gestalten.

Zur Darstellung der angestrebten Kompetenzen informatischer Bildung werden Themenbereiche der Informatik (Key Concepts: *Languages, Machines, and Computation, Data and Representation; Communication and Coordination; Abstraction and Design; Computers and Computing are part of a wider context*) und typische Handlungen und Denkweisen (Key Processes: *Abstraction, Modelling, Decomposing, Generalising, and Programming*) beschrieben. Daraus werden Inhalte und Umfang für eine Vorbildung in der Primar- und Sekundarstufe abgeleitet, die die Themen Algorithmen, Programme, Daten, Computer, Kommunikation und Internet sowie „vertiefende Themen“ umfassen.

Für die zu erreichende Ausbildungsniveaus werden im CAS-Konzept für die Altersstufen (Key stages 1–4) (je nach in der Schule verfügbaren Stunden) ab der Primarstufe neun angestrebte Kompetenzniveaus (Level 1–8 plus exceptional) unterschieden:

- Key stage 1: In Vorschule bis 2. Klasse (Alter 5–7) können die Kompetenzstufen 1 bis ggf. 3 erreicht werden.
  - Level 1: über Abläufe (storyboards) sprechen, Gegenstände anordnen, Gegenstände, die Eingaben verarbeiten, erkennen, programmierbare Spielzeuge nutzen.
  - Level 2: eigene Abläufe (storyboards) zeichnen, direkte Anweisungen geben, mit programmierbaren Spielzeugen bestimmte Aufgaben erfüllen, Gegenstände klassifizieren.
  
- Key stage 2: In der 3.–6. Klasse (Alter 7–11) werden entsprechend die Kompetenzstufen 2 bis 4 angestrebt.

- Level 3: Gemeinsamkeiten in Storyboards erkennen, eine Serie von Anweisungen planen, lineare Anweisungen geben, systematisch Daten präsentieren.
- Level 4: eine Sequenz von Ereignissen analysieren und repräsentieren, verschiedene Datentypen (Text, Zahl) erkennen, die Präzision einer programmiersprachlichen Syntax verstehen, Anweisungen mit Auswahl und Wiederholung geben, einen Algorithmus durchdenken und das Ergebnis voraussagen.

Darauf aufbauend sollen im durchgängigen Informatikunterricht in der Sekundarstufe die weiteren Kompetenzebenen erreicht werden:

- Key stage 3: In den Klassen 7–9 (Alter 11–14) sind dies die Kompetenzstufe 5 bis 6.
  - Level 5: Probleme zerlegen und in einer geeigneten Notation präsentieren, ausgewählte Algorithmen analysieren und erläutern, ähnliche Problemstellungen und dafür verwendbare Algorithmen erkennen, Variablen in einem Programm explorieren, Anweisungsfolgen schrittweise entwickeln und erproben.
  - Level 6: etwas komplexere Probleme (Suchen und Sortieren) beschreiben, Systemkomponenten grafisch darstellen, Modelle für ähnliche Problemstellungen angeben, Programme analysieren und Verhalten vorher-sagen, verschiedene Lösungen vergleichen, parametrisierte Prozeduren und Funktionen nutzen.
- Key stage 4: In den Klassen 10–12 (Alter 14–17) entsprechend unserer Sekundarstufe II werden die weiteren Kompetenzstufen 7 bis 8 und für besonders begabte Schülerinnen und Schüler in vertiefenden Informatikkursen die Stufe *exceptional* angestrebt.
  - Level 7: Such- und Sortier-Algorithmen beschreiben, Probleme mit geeigneter Notation zerlegen, Ähnlichkeiten in etwas komplexeren Problemen erkennen, vorgegebene Programmbausteine zusammenfügen, komplexere Datenstrukturen, einschließlich relationale Datenbanken, nutzen, Programmierwerkzeuge geeignet auswählen, textbasiertes Programmieren.
  - Level 8: geeignete Programmierkonstrukte wählen, passende Modelle für komplexere Probleme finden, weitergehende Fehlersuche und Debugging, komplexere Datenstrukturen analysieren und optimieren, Beziehung zwischen Realität, Modell, Logik, Algorithmus und Visualisierung verstehen.



- Exzeptionell: professionelle Programmiersprache nutzen, für Problemklassen generelle Modelle angeben.

Das englische Bildungsministerium hat die Vorschläge der Kommission aufgegriffen und ab 2014 ein durchgehendes Fach *Computing* eingeführt, das in den für den Bereich der Expertise relevanten Bereich (Key stages 1 und 2) folgende Kompetenzen verpflichtend benennt (Department for Education, 2013):

- Key stage 1: Vorschule bis 2. Klasse (Alter 5–7)
  - Verstehen, was Algorithmen sind und wie sie in Form von Programmen auf Digitalrechnern (Geräten) implementiert werden, verstehen, dass Programme präzisen und eindeutigen Anweisungen folgen.
  - Einfache Programme selbst entwickeln und austesten.
  - Logische Schlüsse ziehen, um das Verhalten von einfachen Programmen vorherzusagen.
  - Digitaltechnologien zweckgerichtet einsetzen, um digitale Inhalte zu erzeugen, zu organisieren, zu speichern, darauf zuzugreifen und diese anzupassen.
  - Im Alltag den Gebrauch von digitalen Technologien wahrnehmen.
  - Digitale Technologien sicher und respektvoll nutzen und dabei Daten privat halten und wissen, an wen man sich wenden kann, wenn es Bedenken gegenüber Inhalten oder Kontaktanfragen aus dem Internet gibt.
- Key stage 2: In der 3.–6. Klasse (Alter 7–11)
  - Programme entwerfen, entwickeln, implementieren und testen, die spezifischen Ziele erfüllen, u. a. Simulationen physikalischer Systeme; Probleme durch Dekomposition lösen.
  - In Programmen Sequenz, Verzweigung und Schleifen nutzen; Programme mit Variablen und verschiedenen Formen von Ein- und Ausgaben entwickeln.
  - Durch logische Schlüsse das Verhalten von einfachen Algorithmen erklären, Fehler in Algorithmen und Programmen finden und diese korrigieren.
  - Rechnernetzwerke inklusive das Internet verstehen, wie diese verschiedene Dienste zur Verfügung stellen, wie das World Wide Web und deren Möglichkeiten für Kommunikation und Kollaboration einschätzen.
  - Suchtechnologien effektiv nutzen und einschätzen, wie Suchergebnisse ausgewählt und angeordnet (ranked) werden; digitale Inhalte kritisch bewerten.

- Softwarewerkzeuge (inklusive Web-Dienste) auf verschiedenen Informatiksystemen (PC, Tablet, Smartphone) auswählen, kombinieren und nutzen, um eine Bandbreite digitaler Objekte (Programme, Systeme, Inhalte) zielgerichtet zu kreieren und zu verwalten. Dies umfasst das Sammeln, Analysieren, Evaluieren und Präsentieren von Information und Daten.
- Digitale Technologien sicher (safe), respektvoll und verantwortlich nutzen; akzeptables und inakzeptables Verhalten erkennen und Wege zur Meldung von Bedenken kennen.

Für das Curriculum wurden eine Menge von Unterrichtsideen und Materialien entwickelt<sup>38</sup> und entsprechende Weiterbildungen für Lehrkräfte konzipiert. Alle inhaltlichen Punkte betreffen immer sowohl Aspekte der reflektierten Nutzung – engl. digital literacy (DL) und der Informationstechnologie (ICT) als auch der grundlegenden informatischen Konzepte (Computer Science – CS). Dabei bilden anfangs eher Nutzungskompetenzen und mit zunehmendem Alter und steigenden Kompetenzstufen die informatischen Konzepte den Schwerpunkt (siehe Tabelle 1). Für den in dieser Expertise betrachteten Altersbereich sind vor allem die Kompetenzstufen 1 bis 3 (ggf. teilweise 4) relevant.

---

<sup>38</sup> <http://primary.quickstartcomputing.org/> und <http://community.computingschool.org.uk/resources>  
[Zugriff am 05.03.2018]

	CS	ICT	DL
1	<p>Understand what algorithms are</p> <p>Create simple programs</p>	<p>Use technology purposefully to create digital content</p> <p>Use technology purposefully to store digital content</p> <p>Use technology purposefully to retrieve digital content</p>	<p>Use technology safely</p> <p>Keep personal information private</p> <p>Recognise common uses of information technology beyond school</p>
2	<p>Understand that algorithms are implemented as programs on digital devices</p> <p>Understand that programs execute by following precise and unambiguous instructions</p> <p>Debug simple programs</p> <p>Use logical reasoning to predict the behaviour of simple programs</p>	<p>Use technology purposefully to organise digital content</p> <p>Use technology purposefully to manipulate digital content</p>	<p>Use technology respectfully</p> <p>Identify where to go for help and support when they have concerns about content or contact on the internet or other online technologies</p>
3	<p>Write programs that accomplish specific goals</p> <p>Use sequence in programs</p> <p>Work with various forms of input</p> <p>Work with various forms of output</p>	<p>Use search technologies effectively</p> <p>Use a variety of software to accomplish given goals</p> <p>Collect information</p> <p>Design and create content</p> <p>Present information</p>	<p>Use technology responsibly</p> <p>Identify a range of ways to report concerns about contact</p>
4	<p>Design programs that accomplish specific goals</p> <p>Design and create programs</p> <p>Debug programs that accomplish specific goals</p> <p>Use repetition in programs</p> <p>Control or simulate physical systems</p> <p>Use logical reasoning to detect and correct errors in programs</p> <p>Understand how computer networks can provide multiple services, such as the World Wide Web</p> <p>Appreciate how search results are selected</p>	<p>Select a variety of software to accomplish given goals</p> <p>Select, use and combine internet services</p> <p>Analyse information</p> <p>Evaluate information</p> <p>Collect data</p> <p>Present data</p>	<p>Understand the opportunities computer networks offer for communication</p> <p>Identify a range of ways to report concerns about content</p> <p>Recognise acceptable/unacceptable behaviour</p>

	CS	ICT	DL
5	Solve problems by decomposing them into smaller parts Use selection in programs Work with variables Use logical reasoning to explain how some simple algorithms work Use logical reasoning to detect and correct errors in algorithms Understand computer networks, including the internet Appreciate how search results are ranked	Combine a variety of software to accomplish given goals Select, use and combine software on a range of digital devices Analyse data Evaluate data Design and create systems	Understand the opportunities computer networks offer for collaboration Be discerning in evaluating digital

**Tabelle 1.** Die Kompetenzstufen 1–5 aufgefächert nach informatischen Kompetenzen (CS), reflektierter Nutzung (DL), und Grundwissen zu Informationstechnologien (ICT) (Computing at School Working Group, 2013, S. 25).

#### 2.4.2 Computational Thinking in den USA

Auch in den Vereinigten Staaten wird diskutiert, ob und wie ein durchgängiges Fach Informatik auch ab der Primarstufe eingeführt werden kann. Die Bestrebungen von Verbänden und Lehrervereinigungen erhalten prominente Unterstützung: Der ehemalige Präsident Obama nahm an einer „Hour of Code“ teil und wurde laut den Websites des Weißen Hauses zum ersten amerikanischen Präsidenten, der selbst programmiert hat. Dabei kündigte er ein 4 Mrd. schweres Programm für die nächsten Jahre an, um Informatik in Schulen zu etablieren:

*„In the coming years, we should build on that progress, by ... offering every student the hands-on computer science and math classes that make them job-ready on day one‘. [...] give all students across the country the chance to learn computer science (CS) in school [...] recognizing that CS is a ‚new basic‘ skill necessary for economic opportunity and social mobility“ (The White House, 2016).*

Die aktuelle Situation bezüglich informatischer Bildung in allgemeinbildenden Schulen wird in den USA generell als problematisch und ungenügend eingeschätzt (Wilson, Sudal, Stephenson & Stehlik, 2010). Genau wie in Deutschland gibt es dabei Unterschiede in den einzelnen Bundesstaaten. In dem Bericht „Running on Empty“ wird deutlich, dass über ein Drittel der Bundesstaaten in ihren Schulen

weder ICT noch informatische Bildung, ein weiteres Drittel lediglich ICT-Grundbildung und das verbleibende Drittel zum Teil auch Informatikwahlangebote aufweist. Durch diese Situation hat die Informatik in den USA in den letzten Jahren an Nachwuchs verloren.

Seitens der ACM (*Association for Computing Machinery*, vergleichbar mit der deutschen GI) gibt es ein Curriculum für Sekundarstufen und studienvorbereitende Vertiefungskurse (Advanced Placement Prüfungen). Die Informatiklehrkräftevereinigung CSTA (Computer Science Teachers Association) hat Empfehlungen für Informatikstandards erarbeitet, die als Rahmenwerk für künftige Lehrplanausgestaltung seitens der Bundesstaaten dienen sollen („CSTA K-12 Computer Science Standards“, 2011). Der Vorschlag zielt auf ein durchgängiges Curriculum K-12. In dem Dokument wird der untragbare Zustand der mangelnden informatischen Bildung in den USA kritisiert. In den meisten Staaten werden derzeit lediglich Nutzungskompetenzen vermittelt (*digital literacy* und ICT), während die informatischen Kompetenzen fehlen. Häufig herrsche bei Politikern, Eltern und Lehrkräften sogar die Fehlvorstellung, dass sich diese Disziplinen gar nicht unterscheiden: „... *general public is not as well educated about computer science as it should be, to the point that the nation faces a serious shortage of computer scientists at all levels that is likely to continue into the foreseeable future.*“ Genau wie bei den zuvor dargestellten britischen Vorschlägen werden die, wenn überhaupt in Curricula vorhandenen, vorherrschenden Angebote für reine IT-Nutzungskompetenzen um den Schwerpunkt informatischer Bildung angereichert. Die amerikanischen und britischen Arbeitsgruppen standen im Austausch und haben sich gegenseitig bereichert.

Die CSTA-Standards thematisieren die Felder (1) *Collaboration*, (2) *Computational Thinking*, (3) *Computing Practice and Programming* und (4) *Community, Global and Ethical Impacts*. Dabei wird betont, dass Kinder informatische Bildung (*Computing*) wegen der vielseitigen, oft kreativen Facetten lieben: „... *the combination of art, narrative, design, programming, and sheer enjoyment that comes from creating their own virtual worlds.*“

Die Empfehlungen für Informatikstandards unterscheiden drei Kompetenzstufen (Level), die durch Abstufung in Alterssegmente in insgesamt sechs Stufen ausdifferenziert werden. Davon sind die beiden ersten (Level 1:3 für das Alter 5–8



(K-3) und Level 1:6 für das Alter 8–11 (Klassen 3–6)) relevant für die in dieser Expertise betrachtete Altersspanne und das damit zu erreichende Kompetenzniveau. Für die erste Kompetenzstufe unter der Bezeichnung „*Computer Science and Me*“ (K-6; Vorschule bis Klasse 6 im Alter von 5–11) sollen die fundamentalen Konzepte der Informatik allen Schülerinnen und Schülern ab der Primarstufe präsentiert werden. Die Expertengruppe der CSTA-Empfehlungen geht davon aus, dass dies erreicht werden kann, indem Technologienutzung, Anwendungen und zugrunde liegende Informatikkonzepte in anderen Schulfächern integriert unterrichtet werden.

Im ersten Altersbereich (K-3) vergleichbar zu Key stage 1 des britischen Curriculums werden die vier Kompetenzbereiche beschrieben:

- Im Kompetenzbereich *Collaboration* geht es auf der ersten Altersstufe darum, Programme zur Sammlung von Informationen einzusetzen und elektronisch zu kommunizieren und zusammenzuarbeiten.
- Der Kompetenzbereich *Computational Thinking* umfasst das Lösen altersgerechter logischer Aufgaben unter Zuhilfenahme von Programmen und Apps, das Arrangieren und Verwalten von Information, das Nutzen von Programmen, um eigene Geschichten in Bilder und Texte zu fassen, und die Erkenntnis, dass Rechner Information in Form von in Form der Ziffern 0 und 1 verarbeiten und Programme entwickelt werden, um Computer (inklusive Smartphones und Tablets) zu steuern.
- *Computing Practice and Programming* umfasst Kompetenzen, die eher der Nutzung von Werkzeugen zum Erzeugen, Arrangieren, Verwalten von Information und digitalen Repräsentationen und explizit dem Einsatz beim Lernen zuzuordnen sind (in unserer Einordnung eher ICT/digital literacy).
- Auch der Kompetenzbereich *Community, Global and Ethical Impacts* ist eher der ICT literacy zuzuordnen und beschreibt das Erkennen ethisch korrekter Nutzung von Informatikwerkzeugen (einschließlich des Internet).

In der Altersstufe bis Klasse 6 (vergleichbar Key stage 2 des britischen Curriculums) werden die folgenden Kompetenzen angestrebt:

- Im Bereich *Collaboration* sollen Standardwerkzeuge wie Textverarbeitung, Präsentationssoftware, Tabellenkalkulation individuell und kollaborativ genutzt werden und dabei auch auf Online-Ressourcen zurückgegriffen werden.

- Der Kompetenzbereich *Computational Thinking* ist am ehesten der informatischen Bildung zuzuordnen. Er umfasst auf dieser Altersstufe das Verständnis, wie Algorithmen aus Einzelschritten konstruiert werden, die Dekomposition von Problemstellungen sowie den Einsatz von Simulationen zur Problemlösung.
- Erneut bildet der Kompetenzbereich *Computing Practice and Programming* den Schwerpunkt, der im Wesentlichen die kompetente Zusammenstellung und Nutzung von Informatikwerkzeugen zum Erzeugen, Manipulieren und Verwalten von digitalen Objekten sowie deren Bedeutung für das eigene Lernen sowie Anwendungen in Berufen umfasst. Programmieren wird lediglich in einem der neun Punkte in Form visueller Sprachen beschrieben.
- Der Kompetenzbereich *Community, Global, and Ethical Impacts* erweitert die Basiskompetenzen der Vorstufe um die Diskussion ethischer Nutzung sowie der kritischen Reflexion der Auswirkungen und Sicherheit von Informatiksystemen.

Beispiele zu allen Themen finden sich im CSTA Web Repository of teaching materials: <http://drupdev.csc.villanova.edu/csta7/>.

Im Vergleich zum britischen Curriculum fällt insgesamt auf, dass in den CSTA-Empfehlungen in beiden Altersbereichen mehr Gewicht auf der Nutzung von Werkzeugen liegt (*digital literacy & ICT*) und deutlich weniger informatische Kompetenzen explizit beschrieben werden (Programmieren, Programmkonstrukte, Algorithmen, Vorhersage des Verhaltens von Programmen). Dafür wird mehr Wert gelegt auf Nutzungskompetenzen von Informationstechnologie zur Zusammenarbeit (explizites Themenfeld *Collaboration*) und zum Erkennen von Informatikwerkzeugen im Alltag und beruflichen Kontext, in denen Informatikbildung hilfreich sind. Außerdem wird bei der CSTA explizit die Binärdarstellung erwähnt. Die drei zuletzt genannten Bereiche sind in den britischen Vorschlägen nicht zu finden.

	Level 1:3 K-3 (Age 5–8)	Level 1:6 3–6 (Age 8–11)
<b>Collaboration</b>	Gather information and communicate electronically with others with support from teachers, family members, or student partners.	Use productivity technology tools (e. g., word processing, spreadsheet, presentation software) for individual and collaborative writing, communication, and publishing activities.
	Work cooperatively and collaboratively with peers, teachers, and others using technology.	Use online resources (e. g., email, online discussions, collaborative web environments) to participate in collaborative problem-solving activities for the purpose of developing solutions or products.
		Identify ways that teamwork and collaboration can support problem solving and innovation.
<b>Computational Thinking</b>	Use technology resources (e. g., puzzles, logical thinking programs) to solve age-appropriate problems.	Understand and use the basic steps in algorithmic problem solving (e. g., problem statement and exploration, examination of sample instances, design, implementation, and testing).
	Use writing tools, digital cameras, and drawing tools to illustrate thoughts, ideas, and stories in a step-by-step manner.	Develop a simple understanding of an algorithm (e.g., search, sequence of events, or sorting) using computer-free exercises.
	Understand how to arrange (sort) information into useful order, such as sorting students by birth date, without using a computer.	Demonstrate how a string of bits can be used to represent alphanumeric information.
	Recognize that software is created to control computer operations.	Describe how a simulation can be used to solve a problem.
	Demonstrate how 0s and 1s can be used to represent information.	Make a list of sub-problems to consider while addressing a larger problem.
		Understand the connections between computer science and other fields.



	Level 1:3 K-3 (Age 5–8)	Level 1:6 3–6 (Age 8–11)
<b>Computing Practice, Programming</b>	Use technology resources to conduct age-appropriate Research.	Use technology resources (e. g., calculators, data collection probes, mobile devices, videos, educational software, and web tools) for problem-solving and self-directed learning.
	Use developmentally appropriate multimedia resources (e. g., interactive books and educational software) to support learning across the curriculum.	Use general-purpose productivity tools and peripherals to support personal productivity, remediate skill deficits, and facilitate learning.
	Create developmentally appropriate multimedia products with support from teachers, family members, or student partners.	Use technology tools (e. g., multimedia and text authoring, presentation, web tools, digital cameras, and scanners) for individual and collaborative writing, communication, and publishing activities.
	Construct a set of statements to be acted out to accomplish a simple task (e. g., turtle instructions).	Gather and manipulate data using a variety of digital tools.
	Identify jobs that use computing and technology.	Construct a program as a set of step-by-step instructions to be acted out (e. g., make a peanut butter and jelly sandwich activity).
	Gather and organize information using concept-mapping tools.	Implement problem solutions using a block-based visual programming language.
		Use computing devices to access remote information, communicate with others in support of direct and independent learning, and pursue personal interests.
		Navigate between webpages using hyperlinks and conduct simple searches using search engines.
		Identify a wide range of jobs that require knowledge or use of computing.

	Level 1:3 K-3 (Age 5–8)	Level 1:6 3–6 (Age 8-11)
<b>Community, Global, and Ethical Impacts</b>	Practice responsible digital citizenship (legal and ethical behaviors) in the use of technology systems and software.	Discuss basic issues related to responsible use of technology and information, and the consequences of inappropriate use.
	Identify positive and negative social and ethical behaviors for using technology.	Identify the impact of technology (e. g., social networking, cyber bullying, mobile computing and communication, web technologies, cyber security, and virtualization) on personal life and society.
		Evaluate the accuracy, relevance, appropriateness, comprehensiveness, and biases that occur in electronic information sources.
		Construct a program as a set of step-by-step instructions to be acted out (e. g., make a peanut butter and jelly sandwich activity).

**Tabelle 2.** Die in diese Expertise eingeflossenen CSTA-Kompetenzlevel und -bereiche.

### 2.4.3 Digital Technologies in Neuseeland (& Australien)

Parallel zur Erstellung dieser Expertise wurden auch für Australien (<http://www.australiancurriculum.edu.au/technologies/digital-technologies/curriculum/f-10> [Zugriff am 05.03.2018]) und Neuseeland Curricula ausgearbeitet und zur Umsetzung vorgeschlagen (Duncan & Bell, 2015). Sie orientieren sich an den zuvor dargestellten Initiativen der USA und Großbritannien, strukturieren die Kompetenzbereiche aber ein wenig anders, um den Anschluss an die existierenden Kompetenzstufenmodelle der Sekundarstufe im Bereich digitaler Technologien zu erleichtern. In beiden Entwürfen werden neben den rein informatischen Aspekten ebenfalls die Kompetenzen der Nutzung von Informatikwerkzeugen und digitalen Technologien beschrieben. Das australische Curriculum betont in der ersten Stufe ähnlich wie der CSTA-Vorschlag für die USA stärker Aspekte, wie digitale Technologien funktionieren (Binärdarstellung).

Im neuseeländischen Entwurf werden die Inhalte in sechs Kategorien eingeteilt: (1) *Algorithms*, (2) *Programming*, (3) *Data representation*, (4) *Digital device infrastructure*, (5) *Digital applications* und (6) *Humans and computers*. Diese Kategorien sollen die Kernprinzipien der Informatik widerspiegeln: „This classification corresponds to the key ideas in computation, since digital devices apply algorithms to data through the practical means of programming, and they produce

digital content which must then be considered in the context of its impact on the individual and society“ (Duncan & Bell, 2015).

Das vorgeschlagene Modell zielt auf die existierenden Standards ab, die informatische Kompetenzen für ein Alter ab 15 Jahren beschreiben. Die angestrebten Kompetenzen werden in sechs Kategorien für fünf darunterliegende *Level* ausdifferenziert, wobei im Vorschlag ein eher konservativer und ein fortschrittlicherer, weitergehender Ansatz unterschieden werden. Für diese Expertise sind wiederum die ersten beiden Kompetenzstufen interessant.

Auf der New Zealand Curriculum-Stufe (NZC-Stufe) 1 (Alter 5–7)

- (1) Verstehen, was *Algorithmen* sind, und diesen folgen können (interpretieren).
- (2) Im Bereich *Programmierung* sollen erste kleine Programme für kybernetische Systeme (Roboter, Turtle) entwickelt werden. Beim „fortschrittlichen Vorschlag“ soll dabei schon einfache Iteration erfolgen und das Suchen von Fehlern behandelt werden.
- (3) Der Bereich *Data representation* wird auf dieser Altersstufe im konservativen Vorschlag noch nicht betrachtet; im fortschrittlichen sollen in Form der Ziffern 0 und 1 als Repräsentation von Text und Bildern verstanden werden.
- (4) *Digital device infrastructure* beschreibt die (physischen, motorischen) Bedienkompetenzen digitaler Systeme (Gesten, Klicken, Maus, Tastatur und Touchscreen). Im weitergehenden Vorschlag sollen die Software- und Hardwarekomponenten zusätzlich beschrieben werden.
- (5) *Digital applications* umfasst die Kompetenzen, um digitale Inhalte zu erzeugen, zu organisieren, zu verändern und darauf zugreifen zu können. Der weitergehende Vorschlag nennt explizit Multimedia als digitalen Inhalt.
- (6) Der Bereich *Humans and computers* soll vor allem den sicheren Gebrauch der Informationstechnologie ermöglichen, indem persönliche Informationen geschützt und Befürchtungen des Missbrauchs bekannt gemacht werden. Der erweiterte Vorschlag umfasst zusätzlich das Erkennen des Einsatzes im Alltag.

Auf der folgenden NCZ-Stufe 2 (Alter 7–9) sollen diese Kompetenzbereiche alterssprechend erweitert werden:

- (1) Probleme sollen zerlegt und Fehler in *Algorithmen* gefunden und korrigiert werden. Beim weitergehenden Ansatz sollen Schritte zur Problemlösung mit betrachtet werden.
- (2) Die *Programmierung* in visuellen Umgebungen soll auf Variablen erweitert werden, Iteration und Fallunterscheidung. Beim weniger konservativen Ansatz sollen dabei interaktive Programme entwickelt und getestet werden.
- (3) Es soll verstanden werden, wie die Binärdarstellung als *Data representation* genutzt wird. Im erweiterten Vorschlag soll dies für verschiedene Informationsformen (Text, Bild, Symbole) erfolgen.
- (4) Die Bedienung der *Digital device infrastructure* soll auf Datenerfassung und -transfer ausgeweitet werden und im erweiterten Fall die Problemsuche und -behebung beinhalten.
- (5) *Digital applications* sollen, aufbauend auf den ersten Erfahrungen, kombiniert werden und auch das Suchen umfassen, im weniger konservativen Ansatz auch die erweiterte Suche und Darstellung von Ergebnissen in einfachen Tabellen und Charts berücksichtigen.
- (6) Der Bereich *Humans and computers* soll um respektvollen und verantwortlichen Umgang erweitert werden. Im erweiterten Vorschlag beinhaltet dies auch die Analyse gesellschaftlicher Bedarfe und die Kommunikation von Information.

Das neuseeländische Modell stimmt in großen Teilen mit dem amerikanischen und britischen überein. Im Vergleich zum CSTA-Vorschlag wird genau wie beim britischen Vorschlag in der ersten Altersstufe darauf verzichtet, den Aspekt der Repräsentation von Information in Form digitaler Daten zu thematisieren. Dafür wird ähnlich wie beim amerikanischen Vorschlag der Bereich der Bedienung digitaler Technologien mehr Augenmerk gewidmet. Der „weniger konservative“ Vorschlag geht in den einzelnen Bereichen etwas weiter als der amerikanische.

Das australische Modell unterscheidet „Verständnis von und Wissen über“ von „Fähigkeiten und Fertigkeiten im Umgang mit“ digitalen Technologien. Die Altersstufen werden wie im Neuseeländischen Modell vorgenommen: F-2 (Alter

5–7) und Year 3 and 4 (Alter 7–9). Insgesamt entspricht dieser in 2016 umgesetzte Vorschlag weitgehend den anderen Modellen.

In der Stufe F-2 sollen Schülerinnen und Schüler Erfahrungen im Kreieren, Verwalten und Nutzen von digitalen Objekten durch spielerischen Umgang mit Informatiksystemen (inklusive Robotern, programmierbaren Spielzeugen etc.) und Sammlung und Organisation von Daten und deren multimedialer Repräsentation sammeln. Ferner sollen sie Schritte zur algorithmischen Problemlösung kennenlernen und den Nutzen von Informatiksystemen im Alltag beschreiben können. Der ethisch verantwortungsvolle und vor allem für die eigene Person sichere Umgang soll bei der Nutzung zur Kommunikation und beim Lernen mit digitalen Medien eingeübt werden.

Am Ende der darauffolgenden Stufe (Years 3–4) sollen Schülerinnen und Schüler die Facetten von Informatiksystemen (Hard- und Software) für ihren Gebrauch einschätzen können und erklären, wie Informationen in Form von Daten zielgerichtet repräsentiert werden können. Sie sollen für einfache Problemstellungen einfache (zum Teil interaktive) Programme entwickeln und implementieren und dabei Entwurfsentscheidungen erläutern können. Sie können Informatikwerkzeuge sicher (*safe*) und zielgerichtet einsetzen.

#### 2.4.4 Schweizer Lehrplan 21

Im Projekt Lehrplan 21 hat die Deutschschweizer Erziehungsdirektoren-Konferenz (D-EDK) von 2010–2014 den Lehrplan 21 erarbeitet. Zunächst war darin für die frühen Stufen keine informatische Bildung enthalten. Nach Protesten einiger Verbände und Studien der Hasler Stiftung hat daraufhin eine Kommission den Lehrplan 21 um ein Modul „Medien und Informatik“ ab der Klasse 3 erweitert. Das Modul enthält Anwendungs- (eher ICT-Nutzung und Medienbildung) und Grundlagenkompetenzen (eher Informatische Hintergründe).

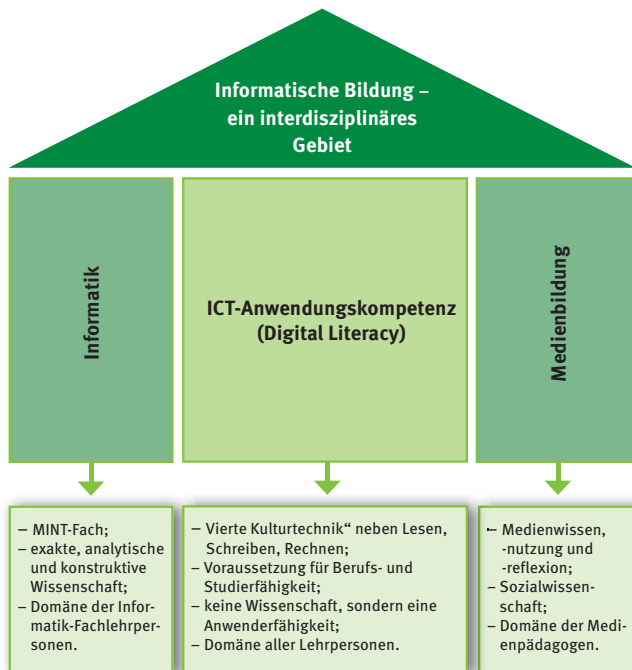
Der Bereich *ICT literacy* wird als Voraussetzung informatischer Erfahrungen und Kompetenzen bereits für den Zyklus 1 (Klassenstufe 1-2) thematisiert. Folgende Kompetenzen sind beschrieben:

- Geräte ein- und ausschalten, Programme starten, bedienen und beenden sowie einfache Funktionen nutzen,
- sich ein- und ausloggen
- Dokumente ablegen und wiederfinden
- mit einfachen Bedienelementen (GUI) umgehen

Für den Bereich der informatischen Bildung werden im Wesentlichen ab Zyklus 2 (Klassenstufe 3–4) für die Bereiche (1) Daten, (2) Algorithmen und (3) Informationssysteme folgende Kompetenzen beschrieben: Die Schülerinnen und Schüler

- können Daten aus ihrer Umwelt
  - in unterschiedlichen Darstellungsformen darstellen, strukturieren und auswerten,
  - entsprechende Dateiformate unterscheiden
  - und kennen Baum(-Verzeichnisse) und Netzstrukturen (Web).
  
- können einfache Problemstellungen analysieren, mögliche Lösungsverfahren beschreiben und in Programmen umsetzen (und testen).
  - Abläufe mit Verzweigungen und Schleifen
  - Einfache Algorithmen mit Parametern interpretieren
  - Programme als eindeutige Anweisungen verstehen
  
- verstehen Aufbau und Funktionsweise von informationsverarbeitenden Systemen und können Konzepte der sicheren Datenverarbeitung anwenden
  - Betriebssystem- und Anwendungsebene unterscheiden
  - verschiedene Speicherarten (z. B. Festplatten, Flashspeicher, Hauptspeicher) und deren Vor- und Nachteile und Größenbeschränkungen verstehen
  - können bei Problemen mit Geräten und Programmen Lösungsstrategien anwenden (z. B. Hilfe-Funktion, Recherche).
  - können erklären, wie Daten verloren gehen können, und kennen die wichtigsten Maßnahmen, sich davor zu schützen

Im Vergleich zu den zuvor aufgeführten Ansätzen, greift der Schweizer Vorschlag informatisch etwas kürzer, ist aber dennoch weiter gefasst, als der Status quo in Deutschland. Im Schweizer Entwurf wird deutlicher als in den anderen internationalen Vorschlägen, dass eine digitale Bildung auf drei unterscheidbaren aber nicht trennscharf abgrenzbaren Kompetenzbereichen beruht: (digitale) Medienbildung, ICT-Anwendungskompetenz und informatische Bildung (siehe Abbildung 23). Diese Ansicht wird ebenfalls in der Dagstuhlerklärung der GI und in weiteren Strategiepapieren der Fachbereiche Informatik und Ausbildung der GI vertreten (vgl. Kapitel 1.6).



**Abbildung 23.** Informatische Bildung als interdisziplinäres Gebiet (aus Deutschschweizer Erziehungsdirektion, 2015)

## 2.5 Einordnung der internationalen Standards in den Ordnungsrahmen eines Kompetenzmodells für informatische Bildung in der Primarstufe

Im Folgenden soll zunächst das Kompetenzstrukturmodell vorgestellt werden, das den 2008 veröffentlichten Empfehlungen für Bildungsstandards in der Informatik der Sekundarstufe I (GI – Gesellschaft für Informatik e. V., 2008) zugrunde liegt. Diese Empfehlungen bilden derzeit die wesentliche Basis aller Ausgestaltungen der Bundesländer, die Informatik in der Sekundarstufe I vorsehen. Allerdings bleibt kritisch anzumerken, dass die GI-Empfehlungen von einem durchgängigen Fach Informatik in der Sekundarstufe I ausgehen, während in den meisten Bundesländern Informatik lediglich als Wahlfach für die Klassenstufen 8 und 9 existiert. Zusätzlich gibt es in den Stufen 5 und 6 an vielen Schulen ICT-Anwendungskurse entweder als eigenständige Fächer oder integriert in andere. Daher sind die in den GI-Empfehlungen formulierten Kompetenzen faktisch nicht erreichbar. Bei der Ausgestaltung der Sekundarstufen-I-Informatik erfolgt daher in der Regel eine Auswahl aus dem gesamten Katalog.

### 2.5.1 Kompetenzstrukturmodell für informatische Bildung in den Sekundarstufen

Die Empfehlungen für Informatikstandards orientieren sich an dem NCTM-Kompetenzstrukturmodell (NCTM – National Council of Teachers of Mathematics, 1989, 1991, 1995, 2000) der US-amerikanischen Mathematik, die ebenfalls den Bildungsstandards der Mathematik in Deutschland zugrunde liegen. In den NCTM- und den GI-Empfehlungen werden jeweils fünf Inhalts- und Prozessbereiche unterschieden. „Die Inhaltsbereiche charakterisieren mindestens zu erwerbende fachliche Kompetenzen. Die Prozessbereiche beschreiben, auf welche Art und Weise die Schülerinnen und Schüler mit den genannten Fachinhalten umgehen sollen“ (GI – Gesellschaft für Informatik e. V., 2008). Daher sind die zu erwerbenden Kompetenzen jeweils als eine Kombination von Inhalts- und Prozessbereichen zu beschreiben (s. Abbildung 24). Die jeweiligen Bereiche sind dabei nicht trennscharf voneinander abzugrenzen.

Die Kompetenzen werden für die Klassenstufen 5–7 und 8–10 beschrieben. Übergeordnetes Ziel ist es, alle Schülerinnen und Schüler für ein Leben in einer digital geprägten Welt vorzubereiten. Dabei sollen sie den grundlegenden Aufbau von Informatiksystemen und deren Funktionsweise verstehen, um damit

„einerseits deren zielgerichtete Anwendung bei der Lösung von Problemen, aber auch die leichte Erschließung anderer Systeme der gleichen Anwendung zu ermöglichen. Die schulische Auseinandersetzung mit dem Aufbau und der Funktionsweise von Informatiksystemen darf dabei aber nicht nur auf der Ebene der Benutzungsschnittstelle erfolgen, die sich bereits bei einer nächsten Produktversion oder bei Verwendung eines Produkts eines anderen Herstellers ändern kann“ (GI – Gesellschaft für Informatik e. V., 2008, S. 19).

Den Ausgangspunkt für einen allgemeinbildenden Zugang bildet die Repräsentation von „Information“ zu Fragestellungen aus der Lebenswelt der Schülerinnen und Schüler durch „Daten“ (Inhaltsbereich 1 – I1). Des Weiteren gehört zur Allgemeinbildung die Erkenntnis, dass Daten und Algorithmen (I2) in einer bestimmten, präzise festgelegten Sprache (I3) formuliert werden müssen, um von einem Rechner oder einem einfachen Modell, wie das der Automaten (I3), automatisch verarbeitet werden zu können. Dabei können Informatiksysteme verschiedener Anwendungsbereiche (I4) modelliert und einfache Teile implementiert (Prozessbereich 1 – P1) werden, deren Einsatz das Zusammenleben in unserer Gesellschaft wesentlich gestalten (I5). Dabei sind Chancen und Risiken abzuwägen und mögliche Entwurfsentscheidungen zu begründen und zu bewerten (P2). Diese Inhaltsbereiche sollen in einem Unterricht behandelt werden, der Schülerinnen



und Schüler zu sachgerechter „Kommunikation“ (P4) unter Verwendung informatischer Fachsprache, zu informatischem „Strukturieren“ (P3), „Begründen“, „Bewerten“ und zur „Kooperation“ anregt und innerinformatische Erkenntnisse mit solchen außerhalb der Informatik „vernetzt“.



**Abbildung 24.** Empfehlungen für Bildungsstandards der Informatik in der Sekundarstufe I aus [www.informatikstandards.de](http://www.informatikstandards.de)

Diese Struktur ist in Anlehnung an die NCTM-Standards über viele Workshops mit Lehrkräften und Fachdidaktikern in ca. 5 Jahren entstanden. Im Einzelnen sind die Kompetenzbereiche über alle Jahrgangsstufen zusammenfassend in den Empfehlungen folgendermaßen definiert (GI – Gesellschaft für Informatik e. V., 2008, S. 20). Die fünf Inhaltsbereiche beschreiben, womit sich die Schülerinnen und Schüler beschäftigen:

#### *Information und Daten (I1)*

Schülerinnen und Schüler aller Jahrgangsstufen

- verstehen den Zusammenhang von Information und Daten sowie verschiedene Darstellungsformen für Daten,
- verstehen Operationen auf Daten und interpretieren diese in Bezug auf die dargestellte Information,
- führen Operationen auf Daten sachgerecht durch.

*Algorithmen (12)*

Schülerinnen und Schüler aller Jahrgangsstufen

- kennen Algorithmen zum Lösen von Aufgaben und Problemen aus verschiedenen Anwendungsgebieten und lesen und interpretieren gegebene Algorithmen,
- entwerfen und realisieren Algorithmen mit den algorithmischen Grundbausteinen und stellen diese geeignet dar.

*Sprachen und Automaten (13)*

Schülerinnen und Schüler aller Jahrgangsstufen

- nutzen formale Sprachen zur Interaktion mit Informatiksystemen und zum Problemlösen,
- analysieren und modellieren Automaten.

*Informatiksysteme (14)*

Schülerinnen und Schüler aller Jahrgangsstufen

- verstehen die Grundlagen des Aufbaus von Informatiksystemen und deren Funktionsweise,
- wenden Informatiksysteme zielgerichtet an,
- erschließen sich weitere Informatiksysteme.

*Informatik, Mensch und Gesellschaft (15)*

Schülerinnen und Schüler aller Jahrgangsstufen

- benennen Wechselwirkungen zwischen Informatiksystemen und ihrer gesellschaftlichen Einbettung,
- nehmen Entscheidungsfreiheiten im Umgang mit Informatiksystemen wahr und handeln in Übereinstimmung mit gesellschaftlichen Normen,
- reagieren angemessen auf Risiken bei der Nutzung von Informatiksystemen.

Die Prozessbereiche beschreiben die Art und Weise, wie sich Schülerinnen und Schüler mit den Fachinhalten auseinandersetzen. Sie geben Hinweise, wie die Inhalte in einem Unterricht didaktisch umgesetzt werden können.

#### *Modellieren und Implementieren (P1)*

Schülerinnen und Schüler aller Jahrgangsstufen

- erstellen informatische Modelle zu gegebenen Sachverhalten,
- implementieren Modelle mit geeigneten Werkzeugen,
- reflektieren Modelle und deren Implementierung.

#### *Begründen und Bewerten (P2)*

Schülerinnen und Schüler aller Jahrgangsstufen

- stellen Fragen und äußern Vermutungen über informatische Sachverhalte,
- begründen Entscheidungen bei der Nutzung von Informatiksystemen,
- wenden Kriterien zur Bewertung informatischer Sachverhalte an.

#### *Strukturieren und Vernetzen (P3)*

Schülerinnen und Schüler aller Jahrgangsstufen

- strukturieren Sachverhalte durch zweckdienliches Zerlegen und Anordnen,
- erkennen und nutzen Verbindungen innerhalb und außerhalb der Informatik.

#### *Kommunizieren und Kooperieren (P4)*

Schülerinnen und Schüler aller Jahrgangsstufen

- kommunizieren fachgerecht über informatische Sachverhalte,
- kooperieren bei der Lösung informatischer Probleme,
- nutzen geeignete Werkzeuge zur Kommunikation und Kooperation.

*Darstellen und Interpretieren (P5)*

Schülerinnen und Schüler aller Jahrgangsstufen

- interpretieren unterschiedliche Darstellungen von Sachverhalten,
- veranschaulichen informatische Sachverhalte,
- wählen geeignete Darstellungsformen aus.

Seit ca. 2009 beschäftigte sich eine GI-Arbeitsgruppe dann mit der Entwicklung von Standards für die Informatik in der Sekundarstufe II. Diese Standards wurden nach mehrjährigen Diskussionen in den Fachgremien im Januar 2016 von der GI verabschiedet und publiziert (GI – Gesellschaft für Informatik e. V., 2016a). Die Struktur der Sekundarstufe-II-Standards wurde beibehalten, die inhaltliche Beschreibung innerhalb der Bereiche allerdings leicht angepasst (insbesondere bezüglich des Modellierungsaspekts). Ergänzend werden für die Sekundarstufe II entsprechend der Anforderungen an einheitliche Abiturprüfungen als dritte Dimension drei *Anforderungsbereiche* unterschieden:

- (1) Reproduktion
- (2) Reorganisation und Transfer
- (3) Reflexion und Problemlösung.

Daraus ergibt sich das dreidimensionale Kompetenzmodell (siehe Abbildung 25):

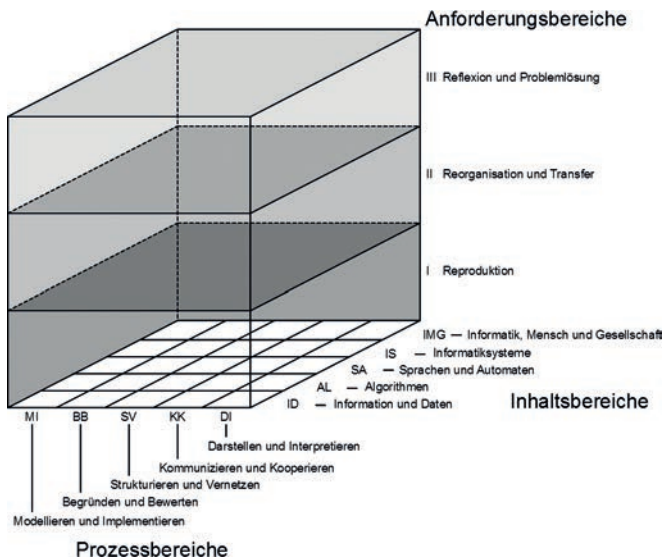


Abbildung 25. Kompetenzmodell der Bildungsstandards Informatik für die Sekundarstufe II

Die jeweiligen Kompetenzbereiche werden in den Standards entsprechend nach Anforderungsbereichen unterschieden und für die Stufen EF und Q1/Q2 dabei jeweils nach Anspruchsniveaus für einen Grund- oder Leistungskurs differenziert. Für die inhaltliche Beschreibung sei auf die Standards selbst verwiesen.

### 2.5.2 Abbildung der Komponenten internationaler Curricula in den Ordnungsrahmen eines vorgeschlagenen Kompetenzmodells für informatische Bildung in der Primarstufe

Die im vorherigen Abschnitt analysierten Kompetenzen aus den internationalen Standards und Curricula werden im Folgenden in den Ordnungsrahmen der GI-Standards der Sekundarstufen dargestellt. Dabei wird herausgearbeitet, dass der Rahmen weitgehend genutzt werden kann, die Bereiche im Altersbereich der Primarstufe allerdings unterschiedlich umfangreich bzw. mit deutlichen Schwerpunkten zu behandeln sind. Ferner wird deutlich, dass der gesamte Schnittstellenbereich zur ICT-Anwendungskompetenz und *digital literacy* innerhalb der informatischen Kompetenzen nur abgedeckt werden kann, wenn ein zusätzlicher Prozessbereich definiert wird. Dieser Prozessbereich, den wir im Folgenden als *PO: Interagieren und Explorieren* einführen, lässt sich in den internationalen Standards häufig finden.

Im Anhang sind die in den internationalen Standards und Curricula genannten Kompetenzen unter den jeweiligen Zuordnungen zu den Inhaltsbereichen im Original zusammengefasst. Im Folgenden werden zunächst diese Zuordnungen zu

den informatischen Prozessbereichen und anschließend den Inhaltsbereichen zusammenfassend beschrieben.

### **Einordnung der internationalen Standards in die Prozessbereiche**

Aus den Kompetenzbeschreibungen der internationalen Ansätze geht häufig nicht direkt hervor, wie sich Schülerinnen und Schüler mit den Fachinhalten auseinandersetzen. Daher lässt sich im Folgenden eine Zuordnung aller beschriebenen Kompetenzen eher zu den Inhaltsbereichen angeben und dabei die möglichen Prozessbereiche andeuten, da die Art der Auseinandersetzung mit einem Fachinhalt häufig von der didaktischen Auseinandersetzung in der unterrichtlichen Situation abhängt. Meist sind die Kompetenzen lediglich als „Verstehen“ oder „Erkennen“ ausgewiesen. Das Verständnis können die Kinder aktiv erlangen, indem sie etwas modellieren und implementieren (P1) und dabei Teile begründen und bewerten (P2). Dabei werden häufig Aspekte strukturiert und vernetzt (P3) und je nach didaktischer Methode dabei kooperiert und kommuniziert (P4). Fachinhalte müssen auch jeweils dargestellt und die fachliche Darstellung interpretiert werden (P5). Vor allem müssen sich Kinder in der aktiven Auseinandersetzung mit Fachkonzepten, Algorithmen, Programmen oder Informatiksystemen Strukturen und Abläufe erschließen, indem sie mit diesen interagieren und diese explorieren (P0), um sich ein mentales Modell aufzubauen. Daher erweitern die Autorinnen und Autoren der Expertise das GI-Kompetenzmodell der Sekundarstufen um diesen Prozessbereich.

#### *(P0) Interagieren und Explorieren*

Das Autorenteam schlägt nach Analyse und Diskussion der in den internationalen Standards vorgefundenen Kompetenzbeschreibungen vor, einen zusätzlichen Prozessbereich für das Kompetenzmodell in der Primarstufe einzuführen. Dieser Prozessbereich beschreibt grundlegende Kompetenzen, sich den Umgang mit Informatiksystemen oder -komponenten zu erschließen. Wie kann man systematisch und fundiert mit Informatiksystemen interagieren? Wie können diese systematisch erschlossen werden, um ein mentales Modell von Funktions- und Wirkungsweisen und Strukturen aufzubauen? Diese Grundfertigkeiten müssen als Voraussetzung erworben werden, um die anderen Prozesse beim Lernen im Kontext zu unterstützen. Wenn sie im Primarbereich ausreichend erworben sind, können sie in der Sekundarstufe I angewendet und vertieft werden, daher tauchen sie in diesem Modell nicht mehr auf, sondern werden als orthogonale Kompetenz in den anderen fünf Prozessbereichen mitgedacht.

Der Prozessbereich wird häufig mit dem Inhaltsbereich I4 (Informatiksysteme) kombiniert (vgl. Kapitel 3.4). Zusammenfassend werden derartige Kompetenzen

in allen internationalen Standards für die betrachteten Altersstufen beschrieben: **Informatiksysteme** oder -komponenten und typische informatische Abläufe (I4) ...

- fundiert nutzen (Komponenten, Architektur, Devices, Dokumentablage etc.)
- gezielt/adäquat anwenden (insbesondere Kommunikation und Kooperation und Erstellung digitaler Artefakte)
- auswählen, vergleichen (Voraussetzung für Begründen und Bewerten)
- explorieren und ein mentales Modell aufbauen, dabei von der Außensicht auf Systeme auf Funktionsweisen und erklärende innere Strukturen schließen
- mit Informatiksystemen lernen

Auch wenn sie in den Standards weniger ausführlich gelistet werden, so lassen sich alle anderen Inhaltsbereiche immer auch mit dem Prozessbereich P0 Interagieren und Explorieren kombinieren:

- **I1 Information und Daten:** Daten gezielt ablegen und finden (Dateien, Namen, Typen), Datenbanken verwenden, Information suchen, ein Modell entwickeln, wie Informationen in Form von Daten organisiert werden können etc.
- **I2 Algorithmen und Programme:** z. B. einfache Algorithmen anwenden (einen Roboter steuern), dabei schrittweise explorieren, Algorithmen durchspielen, dabei mit den Gegenständen interagieren, sich Programme erschließen.
- **I3 Sprachen und Automaten:** dieser Inhaltsbereich wird in den internationalen Standards kaum beschrieben. Beispiele sind aber auch in der jungen Altersstufe vorstellbar, z. B. eine Webadresse (URL) im korrekten Format eingeben, einen Wecker stellen, ...
- **I5 Informatik, Mensch und Gesellschaft:** zu erkennen, in welchen Bereichen Informatiksysteme eine Rolle spielen, kann z. B. durch Rollenspiele erfolgen.

*(P1) Modellieren und Implementieren*

Der **gesamte Inhaltsbereich I2 (Algorithmen und Programme)** wird im Wesentlichen (aber nicht exklusiv) mit diesem Prozessbereich verknüpft (ausführliche Auflistung im Abschnitt ‚Einordnung der Inhaltsbereiche‘ und im Anhang), z. B.

- A pupil should know how to write executable **programs in at least one language** (CAS, S.14, **Programming**).
- **Construct** a set of **statements** to be acted out to accomplish a simple task (e. g., turtle instructions) (CSTA S.14, L1:3.CPP 4.).
- **Implement** problem solutions using a block-based **visual** programming language (CSTA S.14, L1:6.CPP 6.)
- können **Programme** mit Schleifen, bedingten Anweisungen und Parametern schreiben und testen (CH LP21, KL. 3-4).

*(P2) Begründen und Bewerten*

Viele der Inhaltsbereiche bieten Möglichkeiten des Begründens und Bewertens (s. mögliche Zuordnungen im folgenden Abschnitt). In den Kompetenzbeschreibungen sind allerdings oft nur die Themen explizit aufgeführt, ohne dass dargestellt wird, auf welche Weise das Verstehen erfolgt. Dies kann häufig die Möglichkeit des Begründens und Bewertens eines Sachverhalts beinhalten, auch wenn es auf der jungen Altersstufe eher die Ausnahme sein wird, z. B.

- I4 (Informatiksysteme): Apply strategies for **identifying** simple hardware and software **problems** that may occur during use (CSTA S.14, L1:6.CD 3.).

*(P3) Strukturieren und Vernetzen*

Der Prozessbereich P3 ist häufig ein impliziter Bestandteil von P1, da beim Modellieren zunächst ein Sachverhalt analysiert und strukturiert wird. Außerdem wird beim Modellieren und beim Darstellen und Interpretieren (P5) mit Informatiksachverhalten vernetzt. Daher kann P3 in allen Inhaltsbereichen eine Rolle spielen, insbesondere im Bereich I2 (Algorithmen und Programme), sicher auch in I1 (Daten) und I4 (Informatiksysteme). In I5 spielt gerade das Vernetzen mit Sachverhalten außerhalb der Informatik eine Rolle. Vorzufindende Beispiele aus den internationalen Standards sind:

- I1 (Information und Daten): Gather and **organize information** using concept-mapping tools (CSTA S.14, L1:3.CPP 6.).
- I2 Algorithmen (und Programme): Algorithms are developed according to a **plan** and then **tested**. Algorithms are corrected if they fail these tests (CAS, S.13, Algorithms, Key stage 2).



- I1, I2, I4 (Informatiksysteme): Understand how to **arrange (sort) information** into useful order, such as sorting students by birth date, without using a computer (CSTA S.13, L1:3.CT 3.).
- I1, I2, I4, ggf. I5 (Informatik, Mensch und Gesellschaft): Make a **list of sub-problems** to consider while addressing a larger problem (CSTA S. 13, L1:6.CT 5.).
- I5: Understand the **connections between computer science and other fields** (CSTA S.13, L1:6.CT 6.).
- I2, ggf. I4, I5: It can be easier to **plan, test and correct** parts of an algorithm **separately** (CAS, S.13, Algorithms, Key stage 2).

*(P4) Kommunizieren und Kooperieren*

Generell ist der Prozessbereich P4 eher eine Frage der didaktischen Methode und eine überfachliche Kompetenz. Daher taucht sie ohne konkrete Beispiele mit methodischer Umsetzung in den Inhaltsbereichen der internationalen Standards kaum explizit auf. Allerdings ist erwähnt, dass die informatische Methode des Problemlösens (I1, I2, vor allem I4) häufig kooperativ erfolgt. Explizit genannt ist ferner, dass zur kooperativen Problemlösung auch Kommunikations- und Kooperationswerkzeuge genutzt werden (I4, I5). Beispiele sind:

- Gather information and **communicate electronically with others** with support from teachers, family members, or student partners (CSTA S. 13, L1:3.CL 1.).
- **Work cooperatively and collaboratively** with peers, teachers, and others **using technology** (CSTA S. 13, L1:3.CL 2.).
- **Use productivity technology** tools (e. g., word processing, spreadsheet, presentation software) for individual and collaborative writing, communication, and publishing activities (CSTA S.13, L1:6.CL 1.).
- Use online resources (e. g., email, online discussions, collaborative web environments) to participate in **collaborative problem-solving activities** for the purpose of developing solutions or products (CSTA S.13, L1:6.CL 2.).
- **Identify** ways that **teamwork** and collaboration can support problem solving and innovation (CSTA S.14, L1:6.CL 3.).

- Use computing devices to **access** remote information, **communicate** with others in support of direct and **independent learning**, and pursue personal interests (CSTA S.14, L1:6. CPP 7.).

#### *(P5) Darstellen und Interpretieren*

Das Darstellen und/oder Interpretieren informatischer Beschreibungen kommt in allen Inhaltsbereichen und häufig in Kombination mit anderen Prozessbereichen vor. P1 ist ohne P5 kaum möglich.

- Information can be stored and communicated in a variety of forms e.g. numbers, text, sound, image, video (CAS, S.16, Data, Key stage 1).
- Use writing tools, digital cameras, and drawing tools to illustrate thoughts, ideas, and stories in a step-by-step manner (CSTA S.13, L1:3. CT 2.) .
- Structured data can be stored in **tables** with rows and columns. Data in tables can be **sorted**. Tables can be **searched** to answer **questions**. Searches can use one or more columns of the table (CAS, S.16, Data, Key stage 2).

#### **Einordnung der internationalen Standards in die GI-Inhaltsbereiche**

Im Gegensatz zu den Prozessbereichen fällt eine Einordnung der in den internationalen Standards genannten Kompetenzen in den inhaltsbezogenen Ordnungsrahmen der GI einfacher, da die Inhalte jeweils explizit aufgeführt sind.

#### *(1) Information und Daten*

Der Inhaltsbereich der Darstellung von Information in Form von Daten wird in allen Vorschlägen, außer dem neuseeländischen, konservativen Vorschlag, aufgegriffen. Allerdings stellt er keinen Schwerpunkt in den Ansätzen dar. Bei der Umsetzung der CAS Vorschläge in das englische Curriculum hat der Bereich zusätzlich an Stellenwert eingebüßt und wird nur am Rande erwähnt. Bei der CSTA wird die Kompetenz explizit beschrieben.

Zusammengefasst wird in einigen Ansätzen beschrieben, dass Kinder in den unteren Stufen verstehen, dass sich verschiedene Formen von Information (Text, Grafik, Bild, Audio) in Form binärer Daten auf Computern darstellen und verarbeiten lassen. Ferner sollen sie mit verschiedenen Geräten, Information sammeln und digital verwalten können (Fotos, Scan, Audioaufnahmen, ...).

In der zweiten Altersstufe werden bereits Fehler in der Darstellung und ein möglicher Umgang damit thematisiert. Eine explizit genannte Datenstruktur ist die Tabelle; die genannten Verfahren sind das Sortieren von Objekten (oder Repräsentationen) nach gewählten Eigenschaften und das Suchen und Finden digital

verwalteter Information. Ein Informatiksystem, das im Umfeld der Darstellung von verteilter Information genannt wird, ist das World Wide Web.

### *(12) Algorithmen (und Programmierung)*

Der Inhaltsbereich Algorithmen (und hier ergänzt um die Programmierung, weil dieser Bereich in vielen internationalen Curricula in einem Atemzug genannt wird) bildet in fast allen internationalen Entwicklungen einen Schwerpunkt. Es gibt zahlreiche angestrebte Kompetenzen und viele ausgearbeitete Beispiele.

Auf der jüngsten Altersstufe soll zunächst ein Verständnis entwickelt werden, dass Algorithmen und Programme in Form eindeutiger vordefinierter Anweisungen formuliert werden. Für deren Beschreibung wird eine präzise Sprache benötigt. Diese Prinzipien sind auch in der Lebenswelt der Kinder, unabhängig von der Implementierung in Informatiksystemen, zu beobachten. Erste Programmierkonzepte (Sequenz, ggf. schon Wiederholung, strukturierte Zerlegung) können mit geeigneten Werkzeugen oder Alltagsbeispielen erarbeitet werden. Erste Algorithmen, wie das systematische Sortieren nach Eigenschaften können durchgeführt und exploriert werden. Kinder können lernen, eindeutigen Anweisungen nach Regeln zu folgen (Interpretieren von einfachen Algorithmen).

In der zweiten Altersstufe sollten dann auch weitergehende Programmierkonzepte wie Variablen, bedingte Verzweigung und Schleifen genutzt werden. In der Regel sollte die Programmierung von Algorithmen mit programmierbaren Spielzeugen und visuellen Programmierumgebungen eingeführt werden. Die Notwendigkeit von Programmiersprachen (mit vordefinierten Bedeutungen und der Möglichkeit eindeutiger Formulierungen) soll verstanden werden. Auch der Prozess einer systematischen Entwicklung (Strukturierte Zerlegung, Einsatz digitaler Werkzeuge zur Problemlösung, Entwicklung und Test) sollen bereits eingeübt werden.

### *(13) Sprachen und Automaten*

Im eigentlichen Sinn der GI-Standards der Konzepte der theoretischen Informatik (Automaten, Grammatiken) findet sich zu diesem Inhaltsbereich naturgemäß in den internationalen Ansätzen für die unteren Altersstufen nichts. Es gibt einen aktuellen Artikel über einen Unterricht bezüglich Automatentheorie für die „Grundschule“, der sich aber auf Experimente in den Klassen 5 bis 7 bezieht (Isayama, Ishiyama, Relator & Yamazaki, 2016). Wenn man den Bereich dahingehend interpretiert, dass ein Verständnis der Darstellung von Sprachen und deren Eigenschaften sowie Regeln und Werkzeuge zum zustandsgesteuerten Programmieren in diesem Bereich beschrieben werden, könnten Teile der im vorherigen Inhaltsbereich I2 (Algorithmen & Programmierung) genannten Aspekte auch hier zugeordnet werden, insbesondere wenn es um Steuerung von Robotern etc. geht.

Dieser Bereich wird vor allem im britischen Ansatz betrachtet, weniger im amerikanischen oder neuseeländischen.

Wir formulieren in Kapitel 3 und 4 eigene Ideen für mögliche Umsetzungen, die in dieser Form nicht in den internationalen Curricula vorkommen, z. B.

- formaler Aufbau und Regeln, z. B. von Straßennamen, Gruppen in Kitas, Aufbau von URLs
- systematisches Erschließen und Konfigurieren von Informatiksystemen als „formale“ Sprache (P0)

#### *(14) Informatiksysteme*

Der Bereich Informatiksysteme bildet wiederum einen Schwerpunkt in allen internationalen Ansätzen. Dabei geht es häufig darum, Komponenten und Funktionsweisen von Informatiksystemen zu erkennen, sich diese erschließen zu können, fundiert und reflektiert zu nutzen und dabei die notwendige Vorsicht walten zu lassen (Schnittstelle zum folgenden Inhaltsbereich ‚Informatik, Mensch und Gesellschaft‘). Ein großer Teil der mit diesem Inhaltsbereich verbundenen Kompetenzen ist eng mit dem neu vorgeschlagenen Prozessbereich P0 ‚Interagieren und Explorieren‘ verbunden. Insgesamt liegt hier die Schnittstelle zur *IT literacy* und fundierten Nutzungskompetenzen nah, die naturgemäß in den jungen Altersstufen einen Schwerpunkt bilden und eine fundamentale Voraussetzung für die weiterzuentwickelnden informatischen Kompetenzen bilden, außer wenn diese rein ‚unplugged‘ verfolgt werden.

#### *(15) Informatik, Mensch und Gesellschaft*

Der Inhaltsbereich ‚Informatik, Mensch und Gesellschaft‘ findet sich in allen internationalen Ansätzen mit einem Schwerpunkt auf einem sicheren und ethisch korrekten Umgang mit Informatiksystemen, insbesondere dem World Wide Web und darin verfügbaren sozialen Medien. Die Kinder sollen als erstes lernen, dass zu überlegen ist, wem welche Daten preisgegeben werden und hinter welchen Interaktionen Gefahren lauern können. Sie sollen lernen, sich bei Zweifeln an Eltern und Lehrkräfte zu wenden. Auf der anderen Seite sollen sie verstehen, dass am anderen Ende eines Kommunikationskanals Menschen sitzen und dass daher auch bei der Interaktion mit Programmen Kommunikationsregeln gelten.

Ferner umfasst der Inhaltsbereich das Erkennen, wie sehr der Alltag mit Informatiksystemen durchdrungen ist. Die Kinder sollen lernen, Informatiksysteme zu identifizieren und deren Rollen wahrzunehmen.

## 2.6 Ergebnis/Fazit

Die in Abschnitt 2.1 genannten Studien zu Kindern in digitalen Welten belegen, dass Kinder zwar insgesamt selbstsicher mit Informatiksystemen agieren, dabei allerdings häufig die zugrunde liegenden Prinzipien nicht kennen und daher die Konsequenzen ihres Handelns nicht immer richtig einschätzen können. Ferner ist ihre Mediennutzung weitgehend konsumierender Art (Medienangebote im Web nutzen, Videos ansehen, spielen). Sie vernachlässigen die mögliche konstruktive und kreative Möglichkeit, mit und innerhalb von Informatiksystemen Neues zu schaffen und sich ihre digitalen Welten nützlich zu gestalten. Unserer Einschätzung nach könnte eine besondere Rolle von Schulen und ggf. auch Kindergärten darin bestehen, neben den vorherrschenden konsumierenden Nutzungsaktivitäten weitere Interaktionsmöglichkeiten und -typen zu fördern und aufzuzeigen. Zudem könnten Interaktionen nicht nur isoliert und individuell, sondern Informatiksysteme durch Eltern und Kinder zusammen erlebt werden, so dass die Erfahrungen versprachlicht, verarbeitet und altersgerecht reflektiert und eingeordnet werden können.

Die festgestellten lernpsychologischen Voraussetzungen sprechen dafür, dass informatische Bildung in Kitas und in der Grundschule sowohl sinnvoll als auch möglich ist. Die Kinder sind prinzipiell kognitiv dazu in der Lage, ausgewählte informatische Konzepte nachzuvollziehen, zu verstehen und umzusetzen. Sie sind für informatische Aspekte zu begeistern, und zwar Mädchen und Jungen gleichermaßen.

Ferner existiert eine Reihe von erprobten Lerntools für unterschiedliche Zugangsweisen, die für den Einsatz an Kitas und Grundschulen grundsätzlich geeignet erscheinen. Dabei können Zugänge mit und ohne Computer, Tablets und Smartphones genutzt werden. Häufig erlauben programmierbare Spielzeuge, Erfahrungen im kreativen, gestalterischen, programmierenden Umgang mit technischen Geräten spielerisch zu machen. Alternativ stehen Mikrowelten und Entwicklungsumgebungen mit visuellen, blockorientierten Programmiersprachen zur Verfügung. Diese Alternativen bieten Kindern die Möglichkeit, frühzeitig die Phänomene der durch Informatiksysteme geprägten Welt aktiv zu erkunden und erste Schritte des informatischen Denkens zu erfahren.

Welche Kompetenzen in den jungen Altersstufen als Vorbildung für eine durchgängige schulische informatische Bildung anzustreben sind, ist bislang weitgehend unerforscht. Dennoch etablieren derzeit international einige fortschrittliche Bildungssysteme informatische Bildung in der Primarstufe und in der Sekundarstufe I als durchgängige Pflichtfächer, um Schülerinnen und Schülern informatische Denkweisen und Problemlösekompetenzen mittels informatischer Methoden frühzeitig zu vermitteln. Diese Denkweisen qualifizieren sowohl für



eine fundierte und reflektierte Nutzung von als auch die kreative Gestaltung der eigenen Lebenswelt mit Informatiksystemen. Wir haben die angestrebten Kompetenzen der Prinzipien, Standards und Curricula aus Großbritannien, den USA, Neuseeland und der Schweiz zusammengetragen und im Detail analysiert und diese in den Ordnungsrahmen des in Deutschland etablierten Kompetenzstrukturmodells für infor-

formatischen Bildung (GI – Gesellschaft für Informatik e. V., 2008) eingeordnet. Eine wichtige Erkenntnis dieser Analyse und Zuordnung war die Erweiterung des vorgegebenen GI-Modells um einen weiteren Prozessbereich, der das Interagieren und Explorieren von Informatiksystemen und informatischen Methoden beschreibt.

Das resultierende erweiterte Kompetenzmodell dient in den folgenden Kapiteln 3 und 4 dazu, die Zieldimensionen für Kinder und Pädagoginnen und Pädagogen strukturiert darzustellen und anhand konkreter Beispiele und Umsetzungsvorschläge zu diskutieren.

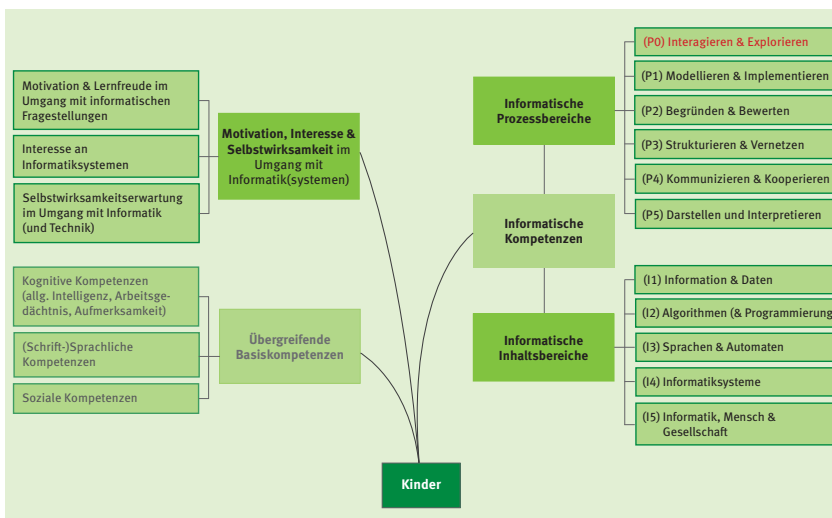
### 3 Zieldimensionen auf Ebene der Kinder

Die im Folgenden vorgestellten Zieldimensionen auf Ebene der Kinder beschreiben mögliche Zielkompetenzen, die von Kita- und Grundschulkindern durch entsprechende Maßnahmen erreicht werden können. Die Struktur dieser Zielkompetenzen orientiert sich am *Kompetenzbegriff von Weinert*. Demnach sind Kompetenzen

„die bei Individuen verfügbaren oder durch sie erlernbaren kognitiven Fähigkeiten und Fertigkeiten, um bestimmte Probleme zu lösen sowie die damit verbundenen motivationalen, volitionalen und sozialen Bereitschaften und Fähigkeiten, um die Problemlösungen in variablen Situationen erfolgreich und verantwortungsvoll nutzen zu können“ (Weinert, 2001, S. 27).

Übertragen auf informatische Kompetenzen bestehen diese demnach aus den notwendigen kognitiven Fähigkeiten und Fertigkeiten, die sowohl informatikspezifisch, als auch überfachlich sein können und motivationale, volitionale und soziale Bereitschaft und Fähigkeiten beinhalten.

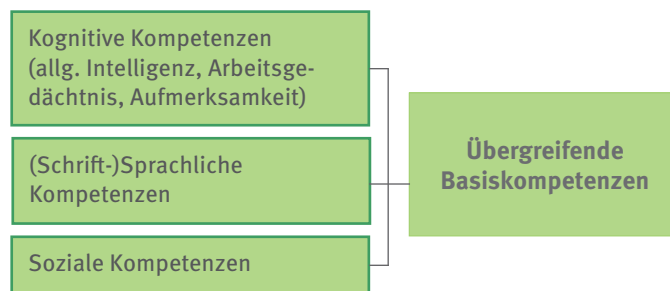
Zur strukturierten Darstellung von Kompetenzen werden hier Kompetenzmodelle, speziell Kompetenzstrukturmodelle, verwendet. Typischerweise orientiert sich die Strukturierung eines Kompetenzmodells an den zu bewältigenden Anforderungen. Die verschiedenen „Kompetenzen und Teilkompetenzen werden vor allem nach den Inhalten der relevanten Aufgaben und den zur Lösung dieser Aufgaben zu bewältigenden Anforderungen definiert“ (Hartig & Klieme, 2006, S. 131).



**Abbildung 26.** Zieldimensionen informatischer Bildung auf Ebene der Kita- und Grundschul Kinder

Das im Folgenden vorgestellte Modell orientiert sich dabei u. a. an dem von Benz et al. (2017) für die frühe mathematische Bildung formulierten und überträgt übergreifende Basiskompetenzen, Motivation, Interesse und Selbstwirksamkeit auf den Umgang mit Informatik(systemen) sowie informatische Kompetenzen (Benz et al., 2017). Letztere werden in Form eines Kompetenzstrukturmodells beschrieben, das sich, wie in Kapitel 2 dargelegt, an das der GI-Empfehlungen für Standards im Fach Informatik für die Sekundarstufe I (GI – Gesellschaft für Informatik e. V., 2008) anlehnt. Der parallel zur Entstehung dieser Expertise durch die Gesellschaft für Informatik verabschiedete Entwurf für Bildungsstandards zum Ende der Sekundarstufe II hat diese Struktur der Prozess- und Inhaltsbereiche ebenfalls übernommen. Der aktuell im Entwurf befindliche Vorschlag für GI-Empfehlungen für die Primarstufe hält sich ebenfalls an die etablierte Struktur der erstgenannten Standards. Aus Sicht der Autorinnen und Autoren ist es daher naheliegend, zu versuchen, diese Struktur für den Elementar- und den Primarbereich weitgehend zu übernehmen, um die Anschlussfähigkeit für die Sekundarstufenkompetenzmodelle zu gewährleisten. Darüber hinaus sollen aber auch die internationale Diskussion und internationale Curricula zur informatischen Bildung, insbesondere in der Grundschule und Elementarerziehung, berücksichtigt werden. Die einschlägigen Ergebnisse unserer Analyse aus Kapitel 2.4 dieser Expertise und grundsätzliche Überlegungen zu einem informatischen Explorations- und Gestaltungskreis haben uns veranlasst, die etablierte Struktur der GI-Bildungsstandards um die Prozessdimension P0 ‚Interagieren und Explorieren‘ zu erweitern (siehe Abbildung 26).

### 3.1 Übergreifende Basiskompetenzen



**Abbildung 27.** Übergreifende Basiskompetenzen

Die im Folgenden genannten Basiskompetenzen stellen keine ausschließlichen Zielkompetenzen informatischer Bildung dar. Sie können einerseits informatische Bildungsprozesse beeinflussen und andererseits durch Lernprozesse im Rahmen



informatischer Bildung gefördert werden. Dies bezieht sich vor allem auf die Bereiche Problemlösekompetenz und Sozialkompetenz.

Da diese fachunabhängigen Basiskompetenzen, bei entsprechender Kompetenzstufe, den informatischen Lernprozess positiv beeinflussen können, sind sie ggf. auch als Kontrollvariablen in der Begleitforschung von Relevanz. Die übergreifenden Basiskompetenzen umfassen die drei Kategorien: Kognitive, sprachliche und soziale Kompetenzen (siehe Abbildung 27).

### **3.1.1 Kognitive Kompetenzen**

Kognitive Fähigkeiten im Rahmen der allgemeinen Intelligenz oder auch der Aufmerksamkeit spielen beim Lernen jeglicher Art eine Rolle. Es ist davon auszugehen, dass entsprechend stärkere Ausprägungen auch Auswirkungen auf das Lernen informatischer Inhalte, insbesondere im Bereich des Problemlösens, haben. Kognitive Kompetenzen bieten sich daher im Rahmen einer Evaluation als Kontrollvariablen an.

### **3.1.2 (Schrift-)Sprachliche Kompetenzen**

Der Erwerb informatischer Kompetenzen ist eng verbunden mit der Entwicklung einer informatischen Fachsprache, die u. a. zahlreiche Anglizismen enthält (Computer, Hardware, Software, ...). Die Kinder können diese Begriffe bei der Einführung durch pädagogische Fachkräfte, die ihnen als Modell dienen, aufnehmen, auch ohne englische Sprachkenntnisse zu haben. Dabei ist es essenziell, dass die pädagogischen Fachkräfte die Terminologie beherrschen und konsequent anwenden. Spezielle Messinstrumente zur Erhebung informatischer Sprachentwicklung sind derzeit nicht bekannt.

### **3.1.3 Soziale Kompetenzen**

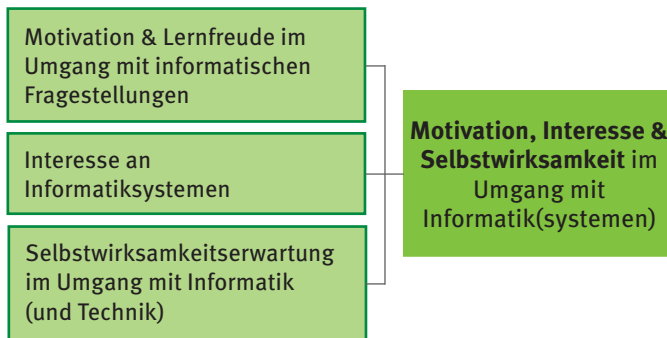
Soziale Kompetenzen spielen immer dann eine große Rolle, wenn im Team bzw. in Gruppen gelernt und gearbeitet wird. Gerade in der Informatik spielt diese Lern- und Arbeitsform eine große Rolle, sei es in der gemeinsamen Bearbeitung von Problemen, Pairprogramming oder der Diskussion, z. B. über die Bewertung eines Systems, einer Software etc. oder der Beurteilung von gesellschaftlichen und ethischen Rahmenaspekten. Soziale Kompetenzen sind immer auch ein Ziel informatischer Bildung, da sie für die typischen projektartigen Lern- und Arbeitsformen benötigt und dadurch gefördert werden.

### 3.2 Motivation, Interesse und Selbstwirksamkeit bezüglich Informatik

Die psychologischen Konstrukte ‚Motivation‘ und ‚Interesse‘ werden, wie von Benz et al. (2017) für die frühe mathematische Bildung dargestellt, als nah verwandt und unter dem Oberbegriff ‚Motivation‘ betrachtet (Anders, Hardy, Pauen & Steffensky, 2013a; Anders, Hardy, Sodian & Steffensky, 2013b; Benz et al., 2017; Schiefele, 2009):

**Interesse** bezieht sich auf ein klar abgegrenztes Inhaltsgebiet (etwa auf Fußball, Schmetterlinge, klassische Musik oder Überraschungseier), wohingegen Motivation auf eine auszuführende Handlung abzielt. Individuelles Interesse ist ein dauerhaftes, dispositionelles Merkmal des Individuums. Für das Schulalter lassen sich durchaus Interessen für einzelne Schulfächer (und damit Inhaltsbereiche wie die Mathematik, Sachunterricht/Naturwissenschaften oder Technik/Informatik o. Ä.) ausmachen. Für das Vorschulalter halten die Autoren solche spezifischen Interessen für eher unwahrscheinlich, sieht man von außergewöhnlichen Fähigkeiten und damit einhergehendem Interesse (etwa für Sport oder das Spielen eines Musikinstruments) ab.

Die **Motivation** allerdings, die in der Literatur häufig mit „situationalem Interesse“ gleichgesetzt wird (Seel, 2003) (z. B. eine bestimmte Handlung auszuführen), ist gängige Triebkraft auch im Vorschulalter. Motivation wird aber nicht durch ein persönlichkeitsimmanentes Interesse ausgelöst, sondern ergibt sich im Vorschulalter aus dem Spielgeschehen, aus dem Anreiz des Spiels selbst sowie aus der emotionalen Nähe zu den Spielkameradinnen und -kameraden. Dies bietet insbesondere für informatische Bildung die Möglichkeit, sich mit Informatiksystemen und -konzepten spielerisch auseinanderzusetzen. Erst im Laufe der Spielerfahrung dürfte sich eine positive Einstellung bzw. eine Ablehnung für einzelne Spiele bzw. Spielarten und -formen ausbilden, was durch die positiven bzw. negativen Erlebnisse in Vergleichssituationen bedingt ist. Situationales Interesse in Bezug auf Informatik ist nach Ansicht der Autoren im Vorschulalter nicht zu erwarten, kann sich aber im Grundschulalter ausbilden, wenn Fächer als wichtig erachtet werden (Benz et al., 2017).



**Abbildung 28.** Motivation und Interesse

Die bei Benz et al. (2017) ausführlich diskutierten Konstrukte ‚Motivation‘, ‚Interesse‘ und ‚Selbstwirksamkeit‘ können analog für Informatik angenommen werden (siehe Abbildung 28). Allerdings liegen für die Informatik weit weniger gesicherte Erkenntnisse aus entsprechenden Studien vor, da Informatikgrundlagen bislang institutionell kaum im Vorschulalter oder der Primarstufe angelegt werden.

Es gibt einige Untersuchungen zur Selbstwirksamkeit bezogen auf die reine Nutzung von Computern, die Einstellungen von Kindern im Vorschulalter mit denen aus der Grundschule (Klassenstufe 2) vergleicht:

„Overall, children express positive attitudes towards computers ( $M = 3.23$ ,  $SD = 0.56$ ). Significant differences in attitudes (...) towards computers were found for gender (in favor of boys), grade level (in favor of the older children in grade 2) and socio-economic status (in favor of the children from a lower socio-economic neighborhood, Southside.“ (McKenney & Voogt, 2010, S. 16)

Ähnliche Ergebnisse für die Primarstufe fanden auch Vekiri und Chronaki (2008).

Der Vorteil der Informatik liegt darin, dass eine altersgerechte Beschäftigung mit Informatiksystemen immer spielerisch erfolgen kann, da Informatiksysteme und darin verwirklichte informatische Konzepte in Form von Spielzeugen repräsentiert werden können (vgl. Kapitel 2.3). Wie die Autoren der Mathematikexpertise halten wir es für sinnvoll, dass die Beschäftigung mit informatischen Inhalten auf spielerischem Weg, z. B. durch programmierbares Spielzeug, erfolgt (Benz et al., 2017). Dies gilt vor allem für das Vorschulalter, aber auch für die Grundschule:

„Prinzipiell ist für den Vorschulbereich zu konstatieren, dass der spielerische Zugang für die Kinder genuin ist, wohingegen Trainingsprogram-

me zum einen keine motivationale Kraft entfalten können und sie zum anderen auch nicht effektvoller sind als der spielerische Zugang“ (Benz et al., 2017, S. 47; vgl. auch Hauser & Rechsteiner, 2011; Pauen & Pahnke, 2008).

Den Autoren ist nur eine Studie bekannt, die den Einfluss von außerschulischen Maßnahmen für informatische Bildung auf die Ausbildung von Selbstwirksamkeit untersucht hat. In seiner Dissertation hat Leonhardt messbare Unterschiede in grundlegenden Einstellungen bezüglich Technik und Informatik zwischen Jungen und Mädchen identifiziert (Leonhardt, 2015). Um diese auszugleichen, hat er gezielt didaktische Maßnahmen für Mädchen im Alter von 11 bis 12 Jahren konzipiert und dafür mittlere bis starke Effekte der Stärkung ihrer Selbstwirksamkeit empirisch nachgewiesen. Diese Maßnahmen bestehen aus einem Interventionsworkshop (zweitägige Roboter-Programmierkurse für Mädchen in ihren Schulen) und privater Folgemaßnahmen (dreitägige weiterführende Programmierworkshops) in den Ferien, um das einsetzende individuelle Interesse zu stabilisieren. Die empirische Pre- und Post-Untersuchung des Interventionsworkshops lieferte zusammen mit der des Aufbauworkshops Erkenntnisse hinsichtlich der Verstetigung und Stabilisierung von Effekten auf das technische, informatische Selbstkonzept sowie auf die Selbstwirksamkeitserwartung und die Kontrollüberzeugung im Umgang mit Technik. Die Ausbildung eines stabilisierenden *emerging individual interest* ist bei den Teilnehmenden nach dem Aufbauworkshop vorhanden. Der Wunsch, sich wiederholt mit dem Lerngegenstand auseinandersetzen zu wollen, ist stark ausgeprägt, so dass die Voraussetzung für ein stabiles persönliches Interesse in Bezug auf den Lerngegenstand vorhanden ist. Die Interventionsmaßnahme ist ein erster Schritt, die entstehende Geschlechterdifferenz im Selbstkonzept im Umgang mit Technik und Informatik zu nivellieren, und zeigt, dass die Lücke zwischen auftretender Umweltsozialisation und zu spät ansetzenden interessefördernden Maßnahmen geschlossen werden kann. Die Studien geben Anlass zur Annahme, dass rechtzeitig beginnende und kontinuierliche informatische Bildung dazu führen kann, dass sich ein Geschlechterunterschied der Selbstwirksamkeitserwartung und Interesse an Informatik gar nicht erst ausbildet.

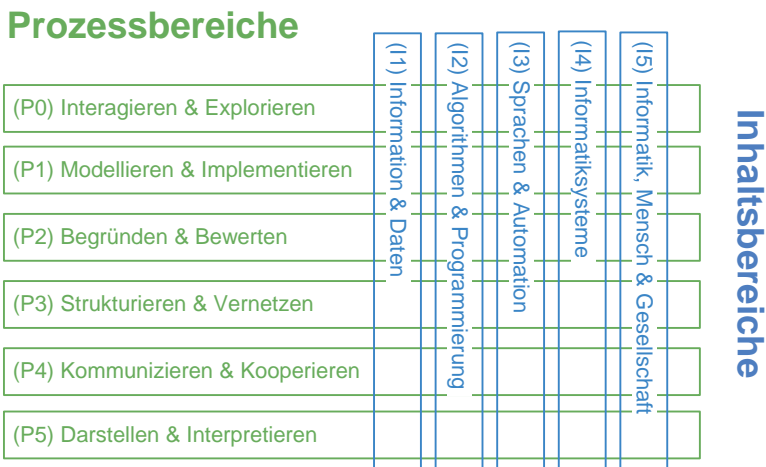
#### *Empfohlene Zielaspekte zur informatischen Selbstwirksamkeit*

Für die Ausbildung eines positiven informatischen Selbstkonzepts ist es günstig, dass Kinder positive Erfahrungen beim Umgang mit informatischen Inhalten machen. Dazu sind realistisch lösbare Aufgaben geeignet, die auf keinen Fall überfordern, aber auch nicht unterfordern sollen. Durch einen spielerischen Umgang mit diesen Aufgaben können Kinder ihre informatischen Kompetenzen erweitern, informatische Darstellungen und regelhafte Abläufe erleben. Sie erfahren, dass

dabei erlernte Begriffe und das Verständnis typischer Abläufe geeignet sind, die (virtuelle) Welt der Informatiksysteme zu ordnen und zu durchschauen. Das gibt ihnen Vertrauen, die (digitale) Welt zu erkunden und mitgestalten zu können. Sie spüren die Wirksamkeit ihres Tuns, was sie in ihrem Selbstwirksamkeitsempfinden wachsen lässt. Für Rückmeldungen sollten gemäß Attributionstheorie Erfolge der eigenen Begabung zugeschrieben werden können, um ein hohes Selbstkonzept aufzubauen bzw. zu erhalten, während Misserfolge eher auf mangelnde Anstrengung oder externale Faktoren zurückgeführt werden sollten (Leonhardt, 2015).

### 3.3 Informatische Kompetenzen der Kinder

Im Folgenden beschreiben wir die Inhalts- und Prozessbereiche zur Charakterisierung von informatischen Kompetenzen auf der Ebene der Kinder. So können wichtige Zieldimensionen informatischer Bildung für diese Adressatengruppe dargestellt werden (siehe Abbildung 29). Diese Darstellung steht im engen Zusammenhang mit den entsprechenden Zieldimensionen für die pädagogischen Fach- und Lehrkräfte. In Kapitel 4.3 werden daher die Inhalts- und Prozessbereiche unter Berücksichtigung der spezifischen Anforderungen für diese Zielgruppe erneut aufgegriffen und spezifiziert.



**Abbildung 29.** Verzahnung der Prozess- und Inhaltsbereiche zur Darstellung informatischer Kompetenzen

*Inhaltsbereiche* charakterisieren die informatischen Inhalte, die von den Kindern erworben werden sollen. *Prozessbereiche* beschreiben, auf welche Art und Weise die Kinder diese fachlichen Inhalte beherrschen sollen (vgl. GI – Gesellschaft für

Informatik e. V., 2008, 2016a). Mit anderen Worten: Inhaltsbereiche beschreiben, was Kinder über Informatik erfahren sollten und informatische Prozesse beinhalten, wie Informatik „gemacht/ausgeführt“ wird. Die hier dargestellten Inhalts- und Prozessbereiche resultieren zum einen aus den Informatik-Bildungsstandards und zum anderen aus der Analyse, die Grundlage des vorherigen Kapitels war. Dabei kamen die Autoren dieser Expertise zu dem Schluss, dass es sinnvoll ist, die in den GI-Empfehlungen für informatische Bildungsstandards beschriebenen informatischen Prozesse um den Bereich Anwenden & Explorieren (P0) zu ergänzen. Grundsätzlich lassen sich alle Inhaltsbereiche mit allen Prozessbereichen kombinieren.

### 3.3.1 Inhaltsbereiche

#### *Information und Daten (I1)*

In der Informatik spielt die systematische Darstellung und automatische Verarbeitung von Daten als Träger der Information eine zentrale Rolle. Dabei wird Information als kontextbezogener Bedeutungsgehalt einer Aussage, Beschreibung etc. verstanden, während Daten Information in formalisierter Art darstellen. Sie eignen sich damit zur Kommunikation, Interpretation und Verarbeitung.

Daten können Information in verschiedenen Darstellungsformen codieren (Bilder, Text, Symbole). Eine Repräsentation in einem Programm ist jeweils eine Zeichenfolge, die einer festen Syntax folgt, um automatisch verarbeitet und interpretiert werden zu können. Für Kinder bekannte Beispiele sind hier etwa Morsen, Zeichensprachen wie z. B. Flaggensignale oder auch die Blindenschrift. Auf Informatikebene spielt der Binärcode eine große Rolle. Daten werden als Binärcode dargestellt, dieser kann wiederum auf physikalischer Ebene in Ein-/Aus-Signale umgewandelt werden.

Zur Information werden Daten dann, wenn sie in einem Bedeutungskontext interpretiert werden. Ein bekanntes Beispiel aus der Lebenswelt von Schülerinnen und Schülern sind Buchstaben und daraus zusammengesetzte Wörter (Daten) und deren Bedeutung (Information). Ein weiteres Beispiel könnten Bilder sein, die sich aus vielen Pixeln zusammensetzen.

Kinder sollten den Zusammenhang zwischen Information und Daten verstehen sowie zwischen Darstellungsform und Bedeutung unterscheiden können. Beim Übersetzen von Information in eine bestimmte Darstellungsform (Daten), z. B. SOS in Morsecode, vollziehen Kinder den Prozess des Darstellens oder im umgekehrten Fall des Interpretierens (P5). Kinder können Dinge nach selbst gewählten Eigenschaften repräsentieren (P1), beispielsweise um sie zu ordnen (P3).

### *Algorithmen und Programme (I2)*

Der Begriff des Algorithmus ist ein zentraler Begriff in der Informatik. Er beschreibt konkrete Handlungsanweisungen, in einer festen Reihenfolge, die notwendig sind, um ein Problem zu lösen bzw. ein Ziel zu erreichen. Für die Zieldimensionen auf Ebene der Kinder bedeutet dies, dass sie Algorithmen begegnen, indem sie altersgerechte Probleme mit Hilfe von Handlungsvorschriften lösen und beschreiben. Dies können einfache Aufgaben sein, z. B. einen Weg durch ein Labyrinth zu finden, oder komplexere Problemstellungen, etwa das Sortieren von Daten.

In der Lebenswelt der Kinder kommen Handlungsvorschriften, z. B. in Form von Spielregeln, Schrittfolgen für Schuhebinden oder Zähneputzen, als Malanleitungen, Wegbeschreibungen, Kochrezepte etc., vor. Sie können diese Vorschriften in verschiedenen Repräsentationsformen, z. B. grafisch, als Text oder in einer Programmiersprache, lesen und interpretieren (P5) und damit symbolisch ausführen und erkunden (P0). Ferner können derartige Vorschriften umgangssprachlich sowie durch eigene selbstausedachte Regeln erweitert werden (P1).

Je nach Altersstufe der Kinder können bereits Darstellungsformen von Algorithmen erarbeitet werden. Handlungsvorschriften können z. B. in Form von Symbolen (Bausteine, Karten z. B. zur Steuerung eines Roboters) oder später in Form von konkreten schriftlichen Anweisungen vorliegen (Verknüpfung zu I3). Der Einstieg kann ‚unplugged‘ erfolgen, indem einfache Symbole auf Papier genutzt und in die richtige Reihenfolge gebracht werden, um beispielsweise einen Papierroboter zu steuern. In weiteren Stufen können auch altersgerechte (z. B. visuelle) Programmierumgebungen genutzt werden. Grundlegende Programmierkonzepte wie z. B. Sequenzen, bedingte Anweisungen und Schleifen/Wiederholungen können so mit den Kindern erarbeitet werden (vgl. Kinder-Website der Stiftung „Haus der kleinen Forscher“ <http://www.meine-forscherwelt.de/>, ScratchJr, AgentSheets). Sie können in vorgegebenen Abläufen auch Fehler finden oder diese Abläufe ergänzen/vervollständigen, um eine Aufgabe zu lösen. Eine wichtige Erkenntnis soll es sein, dass es auf klare und eindeutige Formulierungen sowie exakte und gleichartige Ausführung ankommt. Des Weiteren sollen Kinder die Erfahrung machen, dass sie selbst Abläufe in einer virtuellen Welt gestalten können, indem sie Algorithmen anpassen und weiterentwickeln (P1).

### *Sprachen und Automaten (I3)*

Sprachen sind bekannt als Mittel der Kommunikation zwischen Menschen, aber auch zwischen Menschen und Informatiksystemen (Eingabe und Ausgabe) sowie zwischen verschiedenen Informatiksystemen selbst. In der Informatik ist eine Sprache eine formalisierte Darstellung von Information. Eine solche formale Sprache, die eine feste Syntax verfolgt, ist die Voraussetzung für die maschinelle Verarbeitung durch Automaten (vgl. GI – Gesellschaft für Informatik e. V., 2008).

Automaten kennen Kinder z. B. in Form von Spielautos, Fahrkarten- oder Getränkeautomaten, Smartphones etc. Es handelt sich dabei um zustandsbasierte Systeme, die eine Eingabe lesen und verarbeiten. Erhält ein Automat eine Eingabe, ändert er seinen Zustand. In diesem neuen Zustand kann er Informationen verarbeiten und anschließend etwas ausgeben oder auf eine neue Eingabe warten. Beides würde erneut zu einem Wechsel des Zustands führen.

Kinder erfahren, dass diese Automaten durch Eingaben gezielt gesteuert werden können und dass Automaten sich immer in einem bestimmten Zustand befinden. Um über Zustände von Automaten zu kommunizieren, können grafische Darstellungen von Zustandsübergängen genutzt werden.

Ziel einer informatischen Bildung sollte sein, dass die Kinder Automaten aus dieser informatischen Sicht verstehen. Das bedeutet, dass sie Zustände und Zustandsübergänge in Alltagssituationen identifizieren (P1) und ggf. auch grafisch darstellen können (P5).

#### *Informatiksysteme (I4)*

Informatiksysteme sind aus dem heutigen Alltag nicht mehr wegzudenken. Steckt doch häufig in jeder Waschmaschine oder gar Kaffeemaschine bereits ein Informatiksystem. Schon Kleinkinder spielen mit Smartphones und Tablets und machen so intuitive und spielerische Erfahrungen mit Informatiksystemen (P0), wobei sie diese nicht immer als solche erkennen (Spielkonsole, Smartphone, Fernseher). Informatiksysteme sind spezifische Zusammenstellungen aus Hardware- und Softwarekomponenten zur Lösung eines oder mehrerer Probleme (P1) sowie ggf. Ein- und Ausgabegeräten. Dies kann bereits der kleine Spielzeugroboter sein, der sich mit Hilfe unterschiedlicher Knöpfe steuern lässt, oder eine Puppe, die durch Einstellungen in verschiedene Modi versetzt werden und somit unterschiedliche Musik abspielen kann (Bezug zu P1 und I2, teilweise I3). Im Gegensatz zu dem im vorherigen Abschnitt beschriebenen abstrakten Konzept der Automaten sind Informatiksysteme konkrete Realisierungen, mit denen Kinder interagieren können, um konkrete Erfahrungen zu machen (P0).

Informatiksysteme sind häufig miteinander vernetzt und kommunizieren mittels technischer Kommunikationsprotokolle (Bezug I3) untereinander oder über Aktoren und Sensoren mit anderen technischen Komponenten des Systems (eingebettete Systeme). Wichtig ist auch die ‚Kommunikation‘ mit den Nutzerinnen und Nutzern des Systems. Dies geschieht über Benutzungsschnittstellen (z. B. Display), die es Menschen ermöglichen, mit dem Informatiksystem zu ‚kommunizieren‘ und es zu steuern. Man spricht dann von ‚Mensch-Maschine-Kommunikation‘. Diese ist essenziell für die effiziente Nutzung des Systems. Man bezeichnet Informatiksysteme daher oft auch als soziotechnische Systeme, da bei der Modellierung und Systemgestaltung, die mit dem System assoziierten sozialen



Handlungssysteme von interagierenden Personen mit bedacht werden müssen (Magenheim, 2008; Ropohl, 1999). Informatikdidaktische Konzepte wie der ‚systemorientierte Ansatz‘ oder ‚Informatik im Kontext‘ versuchen, diese soziale Komponente von Informatiksystemen in Informatikbezogenen Lernprozessen zu berücksichtigen (Koubek et al., 2009)

Der Aufbau und Umgang mit diesen Systemen (Bezug zu I5 und P0) ist ein Teil informatischer Bildung. Aus welchen Bestandteilen bestehen Informatiksysteme? Wie spielen diese zusammen? Welche Funktionsweisen werden durch welche Bestandteile realisiert? Wenn Probleme auftauchen, wie können diese eingegrenzt und behoben werden?

Die Informatik führt jedoch noch weiter hinter die „Kulissen“. Wie geschieht die Ein- und Ausgabe, also die Kommunikation mit einem Informatiksystem (Bezug zu I1 und P5)? Hier spielt insbesondere das Eingabe-Verarbeitung-Ausgabe-Prinzip, kurz EVA-Prinzip, eine wesentliche Rolle. Die über ein Eingabegerät induzierte Eingabe wird vom Informatiksystem verarbeitet. Dieses reagiert daraufhin z. B. mit einer Ausgabe auf dem Monitor. Hier wird schnell deutlich, neben der Hardware spielt auch die Software bei der Betrachtung eines Informatiksystems eine Rolle. So existieren unterschiedliche Betriebssysteme mit häufig ähnlichen Grundstrukturen und weitere Software, die bereits im Alltag der Kinder eine Rolle spielen. Kinder müssen grundlegend lernen, dass Informatiksysteme mit Strom arbeiten (I1), wie Programme gestartet und ohne Datenverlust beendet werden (P0), dass ein Login zu einem System eine Rolle spielen kann.

Ein weiterer großer Aspekt ist die Vernetzung von Informatiksystemen bis hin zum Internet. Sein Aufbau und seine Funktionalität sollten ebenfalls Inhalt einer informatischen Bildung sein. Ein Verständnis über den Aufbau und die Funktionalität des Internets unterstützt beispielsweise die Erklärung von Phänomenen wie das Stocken von Videos im Internet. Es unterstützt ein Verständnis darüber, ob sich Daten lokal oder auf einem oder mehreren Servern im Internet befinden. Gerade die Frage, ob sich Daten auf dem eigenen Rechner oder auf einem Server im Internet befinden, ist eine wesentliche informatische Grundkompetenz. Die Vernetzung von Informatiksystemen wird deutlich, wenn Kinder z. B. sich gegenseitig Nachrichten schicken oder Digitalkameras nutzen, Bilder auf Computer überspielen, um sie dann mit Grafik- und Zeichenprogrammen weiter zu bearbeiten. In kindgerechten Kameras ist die Bearbeitungsfunktionalität bereits integriert und verdeutlicht, dass auch Kameras Informatiksysteme sind.

Wesentliche Voraussetzung für die Erschließung von Informatiksystemen ist die Kompetenz, diese systematisch zu erkunden und sich ein geeignetes *mentales Modell* der Systeme zu entwickeln (P0). Dabei spielt die Erkenntnis, dass sich Informatiksysteme anpassen lassen (P2) und daher geeignet sind, sich kreativ zu betätigen und auszudrücken, eine wesentliche Rolle. Dies ermöglicht den Trans-



fer von Wissen auf verschiedene Systeme und legt die Grundlage dafür, Informatiksysteme kompetent für den gewünschten Zweck zu nutzen. Damit verbunden sind auch Fragen, wie Kinder Systeme sicher nutzen und in welchen Fällen sie sich an welche Personen wenden, um Ereignisse in der digitalen Welt zu melden. Schließlich sind auch motorische Fähigkeiten der Nutzung von Gesten oder Eingabegeräten zu erlernen.

#### Informatik, Mensch und Gesellschaft (I5)

Die immer weitreichendere Rolle von Informatiksystemen in unserer Welt hat wesentliche Auswirkungen auf unsere Gesellschaft. Einige dieser Aspekte sollten ebenfalls Teil früher informatischer Bildung sein. So ist es möglich, bereits mit Kindern über positives und negatives soziales Verhalten auch im Umgang mit Technologie und sozialen Medien zu diskutieren. Weitere ethische Aspekte sowie Legalität im Umgang mit Informatiksystemen können Inhalte einer informatischen Bildung sein. Wie kann ich meine persönlichen Daten schützen? Warum muss ich sie überhaupt schützen? Warum sollte ich beim Versenden von Post einen Briefumschlag verwenden? Dies sind ebenfalls wichtige Aspekte im Umgang mit Informatiksystemen, die möglichst früh vermittelt werden sollten.

Darüber hinaus kann über die Rolle von Informatiksystemen im täglichen Leben diskutiert werden. Welche Bedeutung haben diese Systeme für den Alltag (insb. der Kinder selbst)? Welche Auswirkungen können sie haben? Welche Bedeutung spielen sie in der Berufswelt und in welchen Berufen spielt Informatik eine Rolle? Kinder sollen die Durchdringung der Welt durch Informatiksysteme erkennen. Sie sollen informatische Prinzipien an unterschiedlichen Informatiksystemen wie Kamera, Smartphone oder Spielsachen identifizieren und daher die Grundlagen der fortschreitenden Integration der Medien und Systeme (Fernseher, Smartphone) verstehen.

Eine wichtige Kompetenz besteht zunächst darin, zwischen Menschen und Informatiksystemen unterscheiden zu lernen und festzustellen, wann Menschen über das Medium Informatiksystem involviert sind und wann ein Programm (P2). Nicht zuletzt sollten auch Verbindungen von der Informatik zu anderen Wissenschaften, z. B. zur Elektrotechnik oder Physik, gezogen werden (P3).

### 3.3.2 Prozessbereiche

Die fünf Prozessbereiche, die in den Vorschlägen der GI für Bildungsstandards für die Sekundarstufen I und II sowie im Entwurf der GI für die Primarstufe genannt werden, übernehmen wir für unser Kompetenzmodell, um wie bei den Inhaltsbereichen direkte Anschlussfähigkeit an die etablierten Kompetenzmodelle der Sekundarstufen zu ermöglichen:

**(P1)** Modellieren und Implementieren

**(P2)** Begründen und Bewerten

**(P3)** Strukturieren und Vernetzen

**(P4)** Kommunizieren und Kooperieren

**(P5)** Darstellen und Interpretieren

Wie in Kapitel 2.4 begründet, ergänzen wir um den zusätzlichen Prozessbereich

**(P0)** Interagieren und Explorieren

Dieser zusätzliche Prozessbereich macht deutlich, dass als Grundlage einer kompetenten Nutzung die systematische und aktive Exploration steht.

#### *Interagieren und Explorieren (P0)*

Das spielerische Erkunden und Ausprobieren von Objekten ihrer Alltagswelt ist ein wesentliches Element kindlichen Lernens (vgl. Sachser, 2004). Kinder ‚begreifen‘ derartige Objekte im wörtlichen Sinne und gewinnen über den spielerischen Umgang mit ihnen ein rudimentäres mentales Modell über deren technische Funktionen und ihr mögliches ‚Innenleben‘ (vgl. Schwarzkopf & Zolig, 1997; Zolig, 2006). Diese lernpsychologischen Erkenntnisse über den kindlichen Umgang mit simplen technischen Artefakten können auch beim Lernen mit und über altersgemäße Informatiksysteme (z. B. Bee-Bot, ...) genutzt werden. Mittels Interagieren mit und Explorieren von Informatiksystemen können sich Kinder einen ersten Zugang zu diesen Systemen erschließen. Wenn es gelingt, diese Erfahrungen in moderierten Lernprozessen zu systematisieren, können sie die Grundlagen für vertiefte Erkenntnisse der Kinder über Funktion und Struktur derartiger Systeme schaffen.

Neben der lernpsychologischen Begründung dieses Prozessbereiches gibt es auch zahlreiche fachdidaktische und fachwissenschaftliche Argumente, die die Relevanz dieses Prozessbereichs für die informatische Bildung begründen. Zahlreiche internationale Ansätze zur frühen informatischen Bildung, z. B. der Lifelong Kindergarten am MIT (Resnick, 2013) oder die CAS-Initiative in Großbritannien (vgl. Kapitel 2.4), enthalten mit dem ‚Tinkering‘, dem spielerischen Erkunden

von Informatiksystemen, ein dem Prozessbereich P0 ‚Interagieren und Explorieren von Informatiksystemen‘ vergleichbares methodisches Konzept.

In der fachdidaktischen Diskussion gibt es zahlreiche Ansätze, die dem Erkunden und Anwenden von Informatiksystemen eine hohe Bedeutung in der informatischen Bildung beimessen. Im anwendungsorientierten (Körper & Peters, 1988) und im benutzungsorientierten Ansatz (z. B. Forneck, 1992) zur Didaktik der Informatik wird das reflektierte Nutzen und Anwenden von Informatiksystemen als relevant für Lernprozesse in der informatischen Bildung erachtet. Im systemorientierten didaktischen Ansatz soll mit Hilfe Methode der Dekonstruktion die nach außen sichtbare Funktion und die innere Struktur von Informatiksystemen systematisch erkundet werden (Magenheim, 2008). In ähnlicher Weise soll im fachdidaktischen Ansatz zur Dualitätsrekonstruktion an die Alltagserfahrungen der Lernenden mit Informatiksystemen angeknüpft und die Dualität von Struktur und Funktion digitaler Artefakte mittels systematischer Nutzung ermittelt werden (Schulte, 2009). Auch im fachdidaktischen Ansatz von Stechert soll über das Erkunden von Informatiksystemen von deren äußerlich sichtbarem Verhalten auf ihre innere Struktur geschlossen werden (Stechert, 2009).

Schließlich hat die fachdidaktische Kompetenzforschung in der Informatik die Fähigkeit zum Erkunden und Anwenden von Informatiksystemen als eine wesentliche Voraussetzung zum Verständnis von Informatiksystemen identifiziert (z. B. Magenheim, Nelles, Rhode, Schaper, Schubert & Stechert, 2010).

Aus fachwissenschaftlicher Perspektive ergibt sich durch den ständigen technischen Wandel im Bereich der Informatiksysteme sowohl für Expertinnen und Experten als auch für Laiinnen und Laien die Notwendigkeit zur Interaktion und Erkundung von Informatiksystemen. Laiinnen und Laien sollten in der Lage sein, durch benutzungsfreundliche Schnittstellen sich die Funktionsweise von Informatiksystemen zu erschließen und diese ggf. entsprechend ihren spezifischen Bedürfnissen zu konfigurieren. Diese Form der kompetenten Nutzung von Informatiksystemen wird so zunehmend zu einem Teil von Allgemeinbildung.

Bei der Entwicklung neuer Informatiksysteme explorieren Fachleute zunächst meist ein bereits existierendes System und den zugehörigen Nutzungskontext, um dessen grundlegende Funktionsweise und Konstruktionsprinzipien zu erkennen. Auf der Basis dieser Erkenntnisse können dann neue Systemkomponenten konstruiert bzw. das System insgesamt weiterentwickelt werden.

In allgemeinbildender Perspektive eröffnet das Erkunden und Anwenden von Informatiksystemen Kindern die Möglichkeit, sich ein mentales Modell von Informatiksystemen aufzubauen, das im Laufe von weiteren Lernprozessen systematisch erweitert und verfeinert werden kann. Dies ermöglicht ihnen auch in Zukunft, sich mit neuer Software oder neuen Technologien und ggf. mit neuen Interaktionsmustern auseinanderzusetzen und technische Veränderungen reflek-

tiert als Chance und als Risiko zu begreifen. Es befähigt so Kinder, im Laufe ihres Lebens auch bei der Nutzung und Anwendung künftiger Informatiksysteme kompetent zu handeln und einen Nutzen aus neuen Technologien zu ziehen.

Naturgemäß sind die Kompetenzen dieses Prozessbereichs kaum ‚unplugged‘ zu realisieren, da es gerade um die Interaktion mit Informatiksystemen geht (I4), deren Nutzung zu beobachten ist und über deren Verhalten Vermutungen geäußert (P1, P5) werden können, die dann zu validieren sind.

#### *Modellieren und Implementieren (P1)*

Modellieren und Implementieren sind die zentralen Bestandteile des Software-Entwicklungszyklus und damit ein Kernprozessbereich der Informatik. Für die Softwareentwicklung werden die Ausgangssituation und ein Problem darin analysiert, in der Regel mehrere informatische Modelle mit zunehmendem Formalisierungsgrad entworfen, die schließlich auf einem Informatiksystem implementiert, getestet und reflektiert werden. Alle anderen Inhalts- und Prozessbereiche sind normalerweise eng mit diesem verknüpft.

Beim **Modellieren** bildet man relevante Aspekte der Realität oder die eines geplanten Informatiksystems, das die Realität verändern soll, durch Abstraktion zu einem bestimmten Zweck ab. Dazu werden in einer Anforderungsanalyse Sachverhalte und Abläufe unter informatischer Perspektive untersucht. Ziel dabei ist es, die wesentlichen und verallgemeinerbaren Komponenten und Parameter eines Systems sowie ihre Beziehungen untereinander möglichst eindeutig zu beschreiben. In weiteren Modellierungsschritten werden durch strukturierte Zerlegung des Ausgangsproblems, der Identifizierung technisch abbildbarer Teilkomponenten sowie dem Konstruieren und Vernetzen der Komponenten zu einer Systemarchitektur weitere zunehmend formale, textuelle und/oder grafische Modelle entworfen.

Durch das **Implementieren** wird ein formales Informatikmodell so auf einem Informatiksystem beschrieben (in der Regel in einer oder mehreren Programmiersprachen), dass es auf diesem ausgeführt werden kann. Zu diesem Prozessschritt gehört neben der eigentlichen Programmierung auch das systematische Testen und Bewerten der geschaffenen Lösung nach funktionalen und nicht-funktionalen Kriterien (Evaluation) sowie die Reflexion des Systems im Einsatzkontext (Validierung), die deren Möglichkeiten und Grenzen bewertet und häufig zu neuen Anforderungen und einer nächsten Iteration des Modellierungs- und Implementierungszyklus führt.

Auf Ebene der Kinder im Elementar- und Primarbereich können die Aspekte der professionellen Softwareentwicklung nicht vollständig berücksichtigt werden. Eine Zerlegung des Ausgangsproblems in kleinere Probleme (I4, P3) und die Identifizierung von Komponenten sind aber je nach Aufgabenstellung möglich.

Ausgangsprobleme bzw. Aufgabenstellungen stammen aus der Erfahrungswelt der Kinder. Ebenso die strukturierte Darstellung von Lösungswegen (I2, I3, auch P5). Dabei geht es nicht darum, Informatiksysteme zur Lösung von Aufgaben formal zu beschreiben, sondern informatische Perspektiven einzunehmen und Schritte der Modellierung für Alltagsaufgaben und das Beschreiben der Alltagsphänomene zu nutzen. Zudem sollen Kinder die Kompetenz erwerben, einfache informatische (häufig grafische) Modelle zu „lesen“ (z. B. I3) und zu interpretieren (P5), um eine Aufgabe zu lösen. Ferner sollen sie eigene oder vorgegebene Modellbeschreibungen reflektieren. Wie ändert sich das Verhalten, wenn ein Teil der Modellbeschreibung verändert wird? Wie muss ich eine Beschreibung ändern, um einen bestimmten Effekt zu erzielen?

Auch erste Schritte der Implementierung können in speziellen, kindgerechten Arbeitsumgebungen (ScratchJr, MicroBits etc., vgl. Kapitel 2.3) vollzogen werden. Besonders im Vordergrund stehen die Programmerstellung und ihre Methoden und Denkweisen. Sie können bereits im Elementar- und Primarbereich Teil der informatischen Bildung sein. So können kleine Programme entwickelt oder vorgegebene angepasst werden, indem Algorithmen in einer entsprechenden kindgerechten (i. d. R. visuellen) Programmiersprache umgesetzt werden. Diese können darüber hinaus einfach auf Korrektheit überprüft werden, da die geschaffenen Produkte erlebbar sind und mit der eigenen Erwartung an das System abgeglichen werden können. Ein großer Vorteil ist, dass Informatiksysteme damit selbst Rückmeldung an die Kinder geben. Eine Implementierung kann auch in Form von physischen Elementen wie Blöcken oder Karten erfolgen („unplugged“). Allerdings ist dann die Ausführung der eigenen Implementierung nicht mehr gegeben. Das System kann nicht selbst Rückmeldung geben und nicht so einfach auf Korrektheit getestet und validiert werden.

#### *Begründen und Bewerten (P2)*

Für die Erfüllung von Aufgaben oder für die Lösung von Problemen mittels informatischer Methoden gibt es in der Regel grundsätzlich verschiedene Wege. Informatische Zusammenhänge, Vorgehensweisen und Lösungsansätze müssen daher immer erläutert und begründet werden. Nur so lassen sich beispielsweise unterschiedliche Ansätze, Lösungen oder Systeme miteinander vergleichen und nach unterschiedlichen Kriterien bewerten. Dies setzt voraus, dass Fachsprache zusammen mit allgemeinen Kommunikations- und Argumentationsfähigkeiten erlernt wird, mit dem Ziel der reflektierten Nutzung und Anpassung von Informatiksystemen.

Kinder sollen Entscheidungen sachgerecht begründen, informatische Sachverhalte nach Werten des Fachs beurteilen und eigene Positionen bilden. Grundlage

hierfür ist die Kenntnis von Begriffen, Regeln, Methoden und Verfahren sowie ein Verständnis für informatische Sachverhalte. Beim Begründen erfolgt eine bewusste Auseinandersetzung und ein Abwägen der Vor- und Nachteile unterschiedlicher informatischer Vorgehensweisen (I2) oder Repräsentationen (I1). Das Bewerten erfordert eine eigene Position bezüglich informatischer Sachverhalte auf der Basis sachgerechter Bewertungskriterien und -maßstäbe (I5).

Für die Kinder bedeutet dies, dass sie anhand konkreter Beispiele lernen, Fragen zu stellen und begründete Vermutungen über ihnen bekannte informatische Zusammenhänge zu äußern und diese am Informatiksystem zu überprüfen (auch Teil von P0). Mit Hilfe von einfachen Kriterien können sie informatische Sachverhalte bewerten.

#### *Strukturieren und Vernetzen (P3)*

Beim Strukturieren müssen die einzelnen Bestandteile von Sachverhalten oder Abläufen erkannt und miteinander in Beziehung gesetzt werden. Dies bezieht sich auf verschiedene Bereiche der Informatik. So können beispielsweise Informationen organisiert und Daten für einen bestimmten Zugriff und Verarbeitung strukturiert (I1), Teilprobleme konstruiert und oder zerlegt (I2) oder Teilfunktionen oder Komponenten beschrieben werden (I4). Es können Verbindungen unterschiedlicher Art innerhalb und außerhalb der Informatik hergestellt werden (I5). Auch die Erstellung von Planungsprozessen fällt in diesen Bereich. In der Informatik werden häufig komplexe Anforderungen mit informatischen Mitteln, zu denen insbesondere auch Strukturieren und Vernetzen gehören, bearbeitet.

Für die Kinder bedeutet dies, dass sie erste Schritte zur Zerlegung von Sachverhalten oder Abläufen, mit der Absicht der Strukturierung und der Lösung von Aufgaben in Teilschritten, spielerisch erlernen. Erste Anordnungen von Alltagsobjekten können anhand von Kriterien erfolgen, z. B. Zuordnung von Kindern in Lerngruppen oder Klassen. Umgekehrt können Kinder Strukturen und Zusammenhänge anhand einfacher Beispiele ihrer Alltagswelt erkunden.

#### *Kommunizieren und Kooperieren (P4)*

Dieser Prozessbereich ist in der Regel eine Frage des didaktischen Ansatzes und daher fachübergreifend. Dennoch spielt er in der informatischen Bildung eine besondere Rolle. Zum einen kann Kommunikation über informatische Inhalte stattfinden. Zum anderen kann eine Kommunikation mit Hilfe eines Informatiksystems als Medium erfolgen. Zudem kann gemeinsames Arbeiten ohne ein Informatiksystem geschehen, wenn man z. B. gemeinsam ein informatisches Problem im Team erarbeitet. Aber auch hier kann mit Hilfe eines Informatiksystems kollaborativ z. B. an einem Text gearbeitet werden.

Altersentsprechend gewählt können die Kinder die unterschiedlichen Arten des Kommunizierens und Kooperierens kennenlernen/vertiefen/erweitern. Eine besondere Rolle können Multi-Touch-Geräte und darauf abgestimmte Lernprogramme spielen. Als Kompetenz in der informatischen Bildung sollen Kinder lernen, ihre Denkprozesse beim Lösen informatischer Aufgaben umgangssprachlich und zunehmend mit Begriffen einer Fachsprache darzustellen und umgekehrt zu verstehen. Dies ist die Grundlage dafür, informatische Aufgaben zusammen zu lösen.

#### *Darstellen und Interpretieren (P5)*

In der Informatik existieren für unterschiedliche Probleme und Ansätze verschiedene grafische und textuelle Darstellungsformen informeller, semi-formaler oder formaler Natur. Auf der Ebene der Kinder sollten hier altersgerechte Darstellungsformen gewählt werden, die die Kinder selbst nutzen und auch interpretieren können. Dabei können sie ihre Denk- und Vorgehensweisen beim Lösen informatischer Aufgaben formulieren. Dies kann verbal, zeichnerisch oder schriftlich unter Einsatz verschiedener Darstellungsarten wie Graphen (Objekte mit Verbindungen), Tabellen oder Listen etc. erfolgen. Sie lernen, zeichnerisch oder in einfachen, altersgerechten Formalismen dargestellte informatische Sachverhalte zu erläutern (Geschichten erzählen, Abläufe nachvollziehen). Dabei können altersgerechte Lernumgebungen zum Einsatz kommen (Storytelling Alice, ScratchJr).

### **3.4 Priorisierung konkreter Kompetenzerwartungen auf Ebene der Kinder**

Im Folgenden führen wir Zielkompetenzen auf Ebene der Kinder auf, die unserer Meinung nach die wichtigsten altersgemäßen Verknüpfungen von Prozess- und Inhaltsbereichen demonstrieren. Diese Priorisierung erfolgt anhand von Leitkriterien, die sich sowohl an der fachdidaktischen Relevanz des auszuwählenden Kompetenzbereichs als auch an lern- und entwicklungspsychologischen Kriterien orientieren. Neben den fachlich informatischen Kompetenzen, die sich aus einer Kombination von Inhalts- und Prozessbereichen ergeben und in die vor allem fachdidaktische Überlegungen einfließen, spielt bei der Priorisierungsempfehlung der Aspekt der Vermittlung von allgemeinen Basiskompetenzen und der damit zu erwartende Beitrag zur Allgemeinbildung der Kinder eine wichtige Rolle.

Insgesamt ist zu beachten, dass diese Empfehlungen vor allem aus der Perspektive der Informatik-Didaktik anhand der derzeit noch recht übersichtlichen Forschungslage entwickelt worden sind. Die Perspektive der Elementar- und Primarpädagogik und -didaktik auf dieses Themenfeld entwickelt sich gerade erst, so dass die im Folgenden beschriebenen Vorschläge als Impulse für Erprobungen



und für die Entwicklung eines Bildungsbereichs ‚Informatik für die Elementar- und Primarstufe‘ verstanden werden sollten.

In der Zusammenstellung der Kriterien für eine Priorisierung spielen auch die folgenden Überlegungen eine nicht unwesentliche Rolle: Zwar spielt der Computer bereits seit Längerem eine Rolle im Rahmen der Medienbildung in Kita und Grundschule, jedoch geht es hierbei



in erster Linie um die sachgerechte, kritische und reflektierte Mediennutzung (Gesellschaft für Didaktik des Sachunterrichts, 2013, S. 9; Senatsverwaltung für Bildung, Jugend und Wissenschaft Berlin, 2014, S. 103). Die zentrale Aufgabe des Sachunterrichts in der Grundschule ist es jedoch, „Schülerinnen und Schüler darin zu unterstützen, ihre natürliche, kulturelle, soziale und technische Welt sachbezogen zu verstehen, sie sich auf dieser Grundlage bildungswirksam zu erschließen und sich darin zu orientieren, mitzuwirken und zu handeln“ (Gesellschaft für Didaktik des Sachunterrichts, 2013, S. 9). Ohne informatische Bildung können diese Ziele in einer durch Computer geprägten Welt heute jedoch nicht (mehr) erreicht werden.

Der Sachunterricht bedarf daher in Bezug auf das Thema ‚Computer‘ dringend einer Innovation: Um Kinder darin zu unterstützen, ihre technisierte Lebenswelt zu verstehen und sie diesbezüglich angemessen auf die Zukunft vorzubereiten, muss die Thematisierung sog. „Neuer Medien“ über das Anwendungslernen und auch über die Vermittlung allgemeiner Medienkompetenzen hinausgehen:

„Within the computer science community (academia [sic!] and industry) there is general agreement that computer science is not about how to use a computer through the applications that it can execute, but it is about knowing how these applications work“ (Gibson, 2012, S. 34).

Dieses Ziel stimmt mit den Forderungen zur Kompetenzentwicklung im Sachunterricht überein (vgl. Gesellschaft für Didaktik des Sachunterrichts, 2013, S. 63), so dass die zukünftige Aufnahme informatischer Bildung in die Bildungsbereiche der Elementarstufe und insbesondere in den Sachunterricht nur folgerichtig und den technischen und gesellschaftlichen Entwicklungen adäquat erscheint.

Gründe dafür, weshalb eine über die Mediennutzung hinausgehende informatische Bildung bisher einen geringen Stellenwert in der Grundschulpraxis aufweist, sind vielfältig und denen ähnlich, die früher auch zu einer eher randständigen Thematisierung von naturwissenschaftlichen und technischen Inhalten führten (vgl. Döbeli Honegger, 2010; Köster, 2006, S. 11). Neben persönlichen Berührungsängsten bzw. einer Vermeidungshaltung bei den pädagogischen Fach- und Lehrkräften können fehlende Kompetenzen bezüglich informatischer Inhalte und als vordringlicher empfundenen Themen und Herausforderungen im Schulalltag als Gründe für die eher randständige Thematisierung angenommen werden. Bei einer Implementation informatischer Bildung in der Grundschule sind daher Hemmschwellen bzw. persönliche ‚Lehrgrenzen‘ (Köster, 2006) zu beachten, die dazu führen können, dass die Kinder ‚Lerngrenzen‘ erfahren, obwohl sie sich selbst für informatische Bildungsinhalte interessieren.

Zusammenfassend liegen die folgenden Leitkriterien der Priorisierung der ausgewählten Kompetenzbereiche zugrunde:

- (1) **Fachliche und fachdidaktische Konzepte:** Welche Bedeutung hat der ausgewählte Kompetenzbereich für das Verständnis von zentralen informatischen Konzepten und deren Anwendung bei Informatiksystemen?
- (2) **Lern- und entwicklungspsychologische Aspekte:** Können die informatischen Konzepte auf einem der Zielgruppe angemessenen kognitiven Niveau vermittelt werden?
- (3) **Alltagsbezug:** Kann der Kompetenzbereich im Hinblick auf die Alltagssituation der Kinder altersgemäß veranschaulicht werden?
- (4) **Motivation:** Ist der priorisierte Kompetenzbereich anhand eines für die Kinder motivierenden Beispiels darstellbar?
- (5) **Fachliches Interesse:** Sind die priorisierten Kompetenzbereiche geeignet, das Interesse der Kinder an Informatik zu wecken?
- (6) **Selbstwirksamkeit:** Befähigt der Kompetenzbereich und das ausgewählte Beispiel die Kinder dazu, ihre Selbstwirksamkeit im Umgang mit Informatiksystemen zu stärken?
- (7) **Allgemeinbildung:** Können der Kompetenzbereich und das ausgewählte Beispiel einen Beitrag zur Allgemeinbildung der Kinder leisten?
- (8) **Übergreifende Basiskompetenzen:** Welche Basiskompetenzen, z. B. Kommunikations- und Kooperationsfähigkeit, Empathie, Problemlösefähigkeit, werden mit dem ausgewählten Kompetenzbereich gefördert?
- (9) **Bezug zu didaktischen Konzepten** (Grundschule, Sachunterricht, frühkindliche Erziehung): Kann der priorisierte Kompetenzbereich in den curricularen Kontext der Grundschule und in Konzepte der frühkindlichen Erziehung eingeordnet werden?

**(10) Existierende Praxiserfahrungen:** Gibt es beim ausgewählten Kompetenzbereich bereits positive Erfahrungen hinsichtlich der pädagogischen Umsetzung von zielgruppenspezifischen Lernszenarien?

Beim Begriff der Allgemeinbildung orientieren wir uns an Konzepten von Klafki und Heymann:

Klafkis Allgemeinbildungsbegriff beinhaltet (Klafki, 1993):

- A1: Mitbestimmungsfähigkeit
- A2: Selbstbestimmungsfähigkeit
- A3: Solidaritätsfähigkeit

Allgemeinbildung bedeutet

- A4: Bildung für alle
- A5: Allseitige Bildung
- A6: Bildung im Sinne des Allgemeinen

Heymanns Allgemeinbildungsbegriff definiert sich über (Heymann, 1997):

- A7: Vorbereitung auf zukünftige Lebenssituationen
- A8: Stiftung kultureller Kohärenz
- A9: Aufbau eines Weltbildes
- A10: Anleitung zum kritischen Vernunftgebrauch
- A11: Entfaltung eines verantwortlichen Umgangs mit den zu erwerbenden Kompetenzen
- A12: Stärkung des Schüler-Ichs

### **3.4.1 Erläuterung der priorisierten Kompetenzbereiche**

Die Inhalts- und Prozessbereiche lassen sich kombinieren und sollen kombiniert werden. Die einzelnen Möglichkeiten für die Kombination jeweils eines Inhalts- mit einem Prozessbereich zeigt die folgende Tabelle.

Inhaltsbereiche	(I1) Information & Daten	(I2) Algorithmen & Program- mierung	(I3) Sprachen & Automation	(I4) Informatik- systeme	(I5) Informatik, Mensch & Gesellschaft
Prozessbereiche					
(P0) Interagieren & Explorieren					
(P1) Modellieren & Implementieren					
(P2) Begründen & Bewerten					
(P3) Strukturieren & Vernetzen					
(P4) Kommunizieren & Kooperieren					
(P5) Darstellen & Interpretieren					

**Abbildung 30.** Kombinationen von Inhalts- und Prozessbereich. Dunkelgrün für die Zielgruppe wichtige und naheliegende Kombinationen. Hellgrün: Kombinationen, für die wir weitere Beispiele vorschlagen. (Die Beispiele werden in Kapitel 4 und 5 dargestellt.)

Insgesamt scheinen uns die folgenden Kombinationen aus Inhalts- und Prozessbereichen besonders vielversprechend für gelingende informatische Bildung im Elementar- und Primarbereich (in Abbildung 30 dunkelgrün hervorgehoben):

- Modellieren und Implementieren (P1) von Algorithmen und Programmen (I2)
- Interaktion mit und Exploration von (P0) Informatiksystemen (I4)
- Repräsentieren/Darstellen und Interpretieren (P5) von Daten und Information (I1)
- Nachdenken über und Bewerten (P2) des Zusammenhangs von Informatik, Mensch und Gesellschaft (I5)

Es gibt für ein Praxisbeispiel üblicherweise auch Bezüge zu weiteren Inhalts- und Prozessbereichen. Das heißt: Grundsätzlich sind in konkreten Beispielen mehrere Inhaltsbereiche und mehrere Prozessbereiche zu einem unterschiedlichen Grad betroffen. Die folgende Priorisierung hat daher vor allem den Zweck einer Fokussierung.

Die Priorisierung erfolgt zunächst für die Zielgruppe der Kinder. Für die pädagogischen Fach- und Lehrkräfte wird sie später analog vorgenommen.

Im Folgenden werden wir zunächst herausarbeiten und begründen, was die vier ausgewählten Kombinationen auszeichnen und weshalb wir sie als besonders wichtig empfinden.

#### *Modellieren und Implementieren (P1) von Algorithmen und Programmen (I2)*

Dieser Bereich kann – leicht vereinfachend und zugespitzt – als Programmieren zusammengefasst werden. Das Programmieren ist aber ein umstrittener Begriff, da es unterschiedliche Sichtweisen darüber gibt, was als Programmieren angesehen wird.

In den 1940er- und 1950er-Jahren bezogen sich Programmierprobleme und -aufgaben fast grundsätzlich auf mathematische Probleme. In den 1970er-Jahren wurde Programmieren als ein „fundamentally easy task“ gesehen: Man müsse nur die (Konstrukte der) Programmiersprache lernen und es könne losgehen (engage in). Programmierkurse waren daher fast reine Sprachkurse, in der die Syntaxelemente nacheinander vorgestellt wurden. Seit Ende der 1970er- und Anfang der 1980er-Jahre wird Programmieren zunehmend als ein komplexer Problemlöseprozess gesehen, in dem es zum einen um die (technische) Lösung des Problems geht – zum anderen aber auch darum, zunächst einmal überhaupt das Problem zu verstehen (van Merriënboer & Krammer, 1987).

Konsequenterweise werden seitdem zwei Tätigkeiten (und Fähigkeiten) unterschieden: 1) Entwerfen (bzw. Modellieren) und 2) Implementieren. Im ersten Schritt wird das Problem analysiert und eine algorithmische Lösung entworfen, das ist die Entwurfsphase. Im zweiten Schritt wird die Lösung in eine Programmiersprache übertragen, die Implementations- oder Codierungsphase (vgl. van Merriënboer & Krammer, 1987). Seitdem wird der Begriff des Programmierens oft in unterschiedlicher Weise benutzt (vgl. Abbildung 31).



**Abbildung 31.** Sichtweisen auf das Programmieren

In Deutschland wird scheinbar ein eher enges Verständnis gepflegt: Die deutschsprachige Wikipedia fokussiert den Begriff Programmieren auf die Implementationsphase: „Programmierung (von griechisch próγραμμα ‚Vorschrift‘) bezeichnet die Tätigkeit, Computerprogramme zu erstellen. Dies umfasst vor Allem (sic!) die Umsetzung (Implementierung) des Softwareentwurfs in Quellcode sowie – je nach Programmiersprache – das Übersetzen des Quellcodes in die Maschinsprache, meist unter Verwendung eines Compilers.“<sup>39</sup> Die englischsprachige Wikipedia fasst den Begriff weiter: „Computer programming (often shortened to programming, scripting, or coding) is the process of designing, writing, testing, debugging, and maintaining the source code of computer programs.“<sup>40</sup> In diesen beiden Definitionen spiegelt sich die oben erwähnte Unterscheidung von Entwurf und Implementation. Wir verstehen den Begriff in diesem weitergefassten Sinn (ausführlicher Schulte, 2013) und beziehen die Bereiche des Modellierens, Implementierens sowie der Algorithmik und der Programmiersprache ein.

Dabei geht es darum, einfache Handlungsabläufe zu verstehen, nachzuvollziehen und selbst entwickeln zu können. Diese Fähigkeit ist für die Kinder nicht nur im Kontext informatischer Bildung nützlich, sondern auf andere Bildungsbereiche, in denen Problemlösekompetenzen gefragt sind, übertragbar.

Beispielhafte Kompetenzerwartungen in diesem Bereich sind:

- Umgang mit Handlungsvorschriften (vor allem Entwerfen, aber auch Lesen und Verstehen)
  - Die Kinder benennen und formulieren Handlungsvorschriften zur Steuerung eines altersentsprechenden Informatiksystems (auch P5).
  - Die Kinder erklären gelesene Handlungsvorschriften und -abläufe für die Steuerung eines altersentsprechenden Informatiksystems (auch P4).
  - Die Kinder entwerfen eine Vorschrift zur Verschlüsselung von Nachrichten (Daten) mit altersentsprechenden Verfahren (z. B. Skytale).
  
- Implementieren (nutzen einer formalen Notation)
  - Die Kinder entwerfen Handlungsvorschriften/-abläufe mit vorgegebenen altersentsprechenden Bausteinen bzw. Befehlen.
  
- Einfache Algorithmik (Handlungsvorschriften anwenden und untersuchen)
  - Die Kinder wenden ein gegebenes Verfahren zur Lokalisierung einer fehlerhaften Stelle an (auch P5).

<sup>39</sup> <http://de.wikipedia.org/w/index.php?title=Programmierung&oldid=113866966>

<sup>40</sup> [http://en.wikipedia.org/w/index.php?title=Computer\\_programming&oldid=537629774](http://en.wikipedia.org/w/index.php?title=Computer_programming&oldid=537629774)

- Die Kinder erläutern gegebene Algorithmen.
- Die Kinder führen Algorithmen schrittweise aus (simulieren, auch P5).

#### *Interaktion und Exploration (P0) von und mit Informatiksystemen (I4)*

Das spielerische Erkunden und Ausprobieren als wesentliches Element kindlichen Lernens kann insbesondere durch die Kombination des Prozessbereichs P0 bezogen auf Informatiksysteme und deren Funktionsweisen umgesetzt werden. Dabei sollen Kinder Erfahrungen sammeln, um die Grundlagen für vertiefte Erkenntnisse über Funktion und Struktur derartiger Systeme zu schaffen.

Das Umgehen mit einem digitalen Artefakt ist daher nur die basale Ebene dieses Kompetenzbereichs. Im Kern steht die Systemerkundung und zielgerichtete Interaktion, mit der ein Informatiksystem oder ein Algorithmus erschlossen wird. Die Exploration umfasst auch das Anpassen und Gestalten im Sinne eines End-User-Designs. Das Wissen über informatische Konzepte ermöglicht, verschiedene Sichtweisen auf ein Informatiksystem bzw. ein digitales Artefakt zu erschließen. Da nicht alle relevanten Eigenschaften und Zustände eines Informatiksystems immer direkt an der Benutzungsschnittstelle ablesbar sind, ist dies eine wichtige Voraussetzung für das zu entwickelnde Systemverständnis.

Interaktion als Kompetenz ist nicht auf den Umgang mit dem System beschränkt, sondern umfasst das Nachdenken über die Bedeutung der Interaktion. Diese kann über die einzelne Anwenderin/den einzelnen Anwender und den einzelnen Anwendungsfall hinausgehen und sich auf eine Gruppe von Personen beziehen, auf einfachem Niveau etwa in der Wahl eines Werkzeugs für die Nutzung durch die eigene Kita-Gruppe.

Die Kinder sollten nicht nur den konkreten Umgang mit einem digitalen Artefakt erlernen, sondern sich allgemeine und übertragbare Strategien aneignen und sich zutrauen, ein unbekanntes System zu erkunden, und dabei auch über dessen Möglichkeiten, Grenzen und Auswirkungen nachzudenken.

Ein wichtiger Aspekt dieser „Erkundungskompetenz“ ist die Einsicht, dass die meisten Artefakte das Anpassen und Einstellen in Bezug auf die eigenen Wünsche ermöglichen, also selbst gestaltbar sind. Daraus folgen die Möglichkeit und das Zutrauen in die eigenen Fähigkeiten, diese Anpassung auch vornehmen zu können. Das kann über Konfigurieren, Parametrisieren bis hin zu kleineren Programmieraktivitäten im Sinne des End-User-Programming gehen.

Dabei sollen die Kinder vor allem erkennen, dass Informatik nicht nur Interaktion, also Umgehen mit einem System, ist, sondern das Gestalten und Realisieren der Umgangsmöglichkeiten.

Dazu gehören beispielsweise folgende Kompetenzen:

- Exploration (Erkunden, Ausprobieren mit dem Ziel, Erkenntnisse über das digitale Artefakt zu gewinnen)
  - Die Kinder erschließen sich einfache Funktionen der entsprechenden Programme oder Websites und finden diese bei wiederholter Nutzung (auch P3).
  - Sie identifizieren in ihrer Umgebung Informatiksysteme und wie sie diese vorteilhaft nutzen können (auch P2).
  - Die Kinder lernen von der Außensicht der Systeme auf innere Strukturen und Abläufe sowie Zustände eines Informatiksystems zu schließen (auch P2 & P3).
  - Die Kinder wenden Strategien zur Identifizierung einfacher Hard- und Softwareprobleme an, die während der Nutzung auftreten können.
  
- Interaktion (Interaktionspfade erkennen, nutzen, anpassen)
  - Die Kinder zeigen einen altersentsprechenden sicheren Umgang mit Ein- und Ausgabegeräten, vor allem Audioanweisungen sowie Gesten bei Touchscreens und spezieller Symbole.
  - Die Kinder verwenden technische Tools (wie z. B. Multimedia-Anwendungen, Texteditoren, Webtools etc.) zur Präsentation altersentsprechender Aufgaben (auch P4).
  - Die Kinder navigieren zwischen Websites durch die Nutzung von Hyperlinks und konstruieren einfache Suchanfragen mit Hilfe von Suchmaschinen (auch I3).
  
- Bewusstmachung (Bewerten des Artefakts oder eines Interaktionspfades)
  - Sie lernen typische Muster in Informatiksystemen kennen, z. B. die Anordnung von Symbolen und Auswahlmenüs, und erkennen eine personalisierte Nutzung nach Anmeldung an ein System (auch P3).

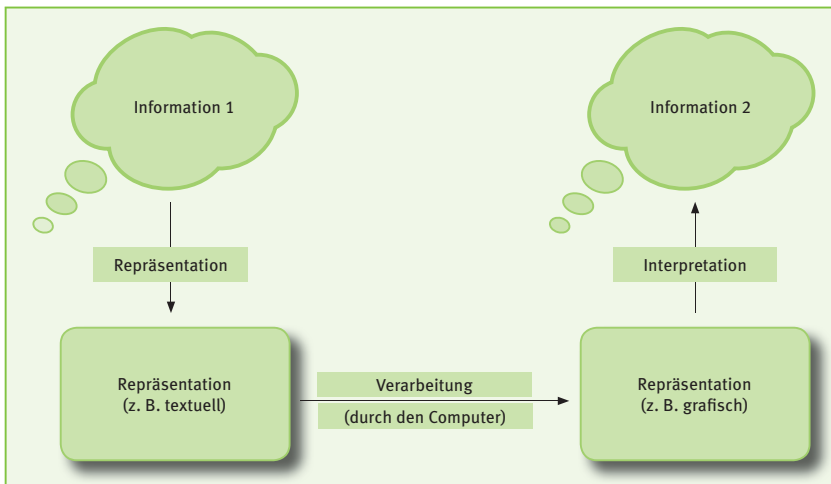
#### *Repräsentieren und Interpretieren (P5) von Information und Daten (I1)*

Kern dieses Bereichs ist der Begriff der Information, der ja namensgebend für die Disziplin ist – und dennoch nur schwer gefasst werden kann. Nach Breier ist Information der „Bedeutungsinhalt einer Aussage, Belehrung, Benachrichtigung, Botschaft, Mitteilung, Nachricht o. ä.“ (Breier, 2004a, S. 69). In Bezug auf informatische Bildung nennt Breier einige wesentliche Merkmale von Information (Breier, 2004a, S. 69):



- Information existiert nicht ‚an sich‘, sondern muss auf einem Trägermedium repräsentiert sein.
- Dazu wird zwischen Sender und Empfänger der Information ein Code vereinbart (Syntax und Semantik).
- Wird die Repräsentation vernichtet, kann auch die Information gelöscht werden.
- Information kann verarbeitet werden, indem ihre Repräsentation verarbeitet wird.

Diese Merkmale zeigen, dass streng zwischen zwei Aspekten einer Nachricht unterschieden wird: zwischen dem Bedeutungsinhalt einerseits und der Form, Darstellung andererseits (vgl. auch Hubwieser, 2007, S. 78). Letzteres ist die Repräsentation einer Nachricht – (nur diese) Repräsentationen können vom Computer verarbeitet werden. Ersteres, also nur der Bedeutungsinhalt, ist die Information. Sie ist (nur) dem Menschen oder allgemeiner: biologischen Organismen, zugänglich (vgl. Abbildung 32).



**Abbildung 32.** Schema der Informationsverarbeitung (nach Breier, 2004a, S. 74; Hubwieser, 2007, S. 80; GI – Gesellschaft für Informatik e. V., 2008, S. 23)

Die Repräsentation von Information kann als ein Datum aufgefasst werden – streng genommen verarbeiten Computer daher immer nur Daten und nicht Infor-

mationen. Die Information gewinnt der Mensch – nach diesem Schema – durch Interpretation der veränderten Repräsentation bzw. der veränderten Daten.

Der Kompetenzbereich Repräsentieren und Interpretieren von Information und Daten beschäftigt sich daher mit unterschiedlichen Darstellungsformen von Daten bzw. unterschiedlichen Repräsentationsarten von Information. Eine wichtige Frage dabei ist z. B., wie Informationen in digitale Daten umgewandelt werden und wie man diese Daten übertragen und interpretieren kann. Eine konkrete Fragestellung ist etwa, wie Bilder auf den Monitor oder „in den Rechner“ kommen.

Beispielhafte konkrete Kompetenzerwartungen können sein:

#### ■ Information repräsentieren (digitalisieren)

Die Kinder ...

- können erklären, wie Informationen in digitale Daten umgewandelt werden.
- führen die Umwandlung von Dezimalzahlen, Buchstaben und Grafiken in Binärzahlen wie auch in umgekehrter Richtung selbst durch.
- können Dinge nach selbst gewählten Eigenschaften repräsentieren (P1) und ordnen (P3), damit sie ein Objekt mit einer bestimmten Eigenschaft schneller finden (z. B. Farbe, Form, Größe).
- tragen Daten zu einer Aufgabenstellung zusammen und repräsentieren sie unter Nutzung von kindgerechten Informatikwerkzeugen (Editoren, digitale Kameras, Zeichenwerkzeuge, Concept Maps), um Gedanken und Geschichten schrittweise zu konstruieren.
- lernen, Antworten auf Aufgaben als eine Reihe von Ja-/Nein-Entscheidungen zu repräsentieren.

#### ■ Daten verarbeiten

Die Kinder ...

- erkennen, dass Daten in einer Vielzahl unterschiedlicher Formen durch ein Informatiksystem repräsentiert, gespeichert, verarbeitet und übertragen werden können, z. B. als Texte, Bilder, Videos, Audio, Kombinationen daraus, ...
- analysieren die Schritte zur Verschlüsselung von Nachrichten (Daten) mit altersentsprechenden Verfahren (z. B. Skytale).

- Daten interpretieren

Die Kinder ...

- können Fehler in binär repräsentierten Daten erkennen und identifizieren.
- können erläutern, dass das Internet eine sehr große Menge von Daten enthält.

*Nachdenken über und Bewerten (P2) des Zusammenhangs von Informatik, Mensch und Gesellschaft (15)*

Ein Grund dafür, mit informatischer Bildung bereits im frühen Kindesalter zu beginnen, liegt in den Veränderungen durch die Digitalisierung vieler Lebensbereiche: in den sog. Auswirkungen der technologischen Weiterentwicklung bzw. Durchdringung der Lebenswelt mit digitalen Artefakten – und in der wissenschaftlichen Disziplin, die das zur Entwicklung dieser Artefakte notwendige Wissen und Können bereitstellt. Aber wie ist das Verhältnis von Informatik, Mensch und Gesellschaft?

Oft wird nur von einer allgemeinen Verknüpfung ausgegangen, wenn von dem Bereich gesprochen wird: Informatik und Gesellschaft bzw. die Fortentwicklung der Technik werden als Treiber gesellschaftlicher Veränderungen angesehen – allerdings kann ein solcher Technik-Determinismus die Veränderungen nur unzureichend erklären. Es scheint vielmehr komplexe Wechselwirkungen zwischen Informatik, Mensch und Gesellschaft zu geben.

Ein interessanter Versuch zu Strukturierung dieser Wechselwirkungen ist im Projekt ‚Kontextuelle Informatik‘ entstanden. Kernidee ist die folgende:

„Technologieentwicklung drückt sich nicht nur in den jeweiligen Artefakten aus, sondern auch in von uns so genannten Soziefakten (geschriebene und ungeschriebene Gesetze und Vereinbarungen) und Kognifakten (in Anlehnung an Foucaults ‚Technologien des Selbst‘ also Kompetenzen, Methoden und damit auch Techniken im ursprünglichen Wortsinn)“ (Engbring & Selke, 2013, S. 113).

Diesem Ansatz zufolge werden im Prozess der Technikgenese nicht nur neue digitale Artefakte (=technische Produkte), sondern auch andere „Produkte“ erstellt: Auf Ebene der Gesellschaft entstehen neue Regeln, Gesetze, Umgangsformen für die Interaktion mit der Technik (und darüber auch die Interaktion zwischen Menschen) – und auf Ebene des Individuums entstehen neue Wissensbestände, Fertigkeiten und Ideen. Diese verschiedenen Prozesse und Produkte durchdringen und beeinflussen sich gegenseitig – und sie beeinflussen den Prozess der Genese

neuer Artefakte und Technologien und damit die Gestaltung des Alltags bzw. der Lebenswelt.

Diese Sichtweise auf Technikgenese vereinfacht und strukturiert zunächst den Ablauf, ist aber inhaltlich neutral. Die Autoren schlagen vor, die Bereiche Arbeit, Kultur und Wissen unter dieser Perspektive in den Blick zu nehmen und im Themenbereich Informatik, Mensch und Gesellschaft schlussendlich also die Technikgenese in den drei Feldern a) **Arbeitsprozesse** (Entwicklung von Maschinen und Werkzeugen), b) **Kommunikationsmedien** (Veränderungen im Bereich Kultur/Kulturtechniken: Schrift, Rechnen, Medien, Kommunikation und Kooperation) und c) **Wissensgesellschaft** (Instrumente und Dienste zum Umgang mit Wissen) zu untersuchen.

Auf diese Weise entsteht eine Auswahl von Inhalten und deren Strukturierung, wie in Abbildung 33 dargestellt:

Techniken	Gestaltung/ Erschließung	Regulation/ Gestaltung	Erschließung/ Regulation	Technikgenese
<b>Arbeit</b> Maschinen Werkzeuge	<i>Partizipative System- entwicklung</i>	<i>Arbeitsschutz/ Ergonomie</i>	Berufs- ausbildung	Arbeitsprozesse
<b>Kultur</b> Schrift Rechnen  Medien Kommunikation Kooperation	<i>Gestaltung interaktiver Medien</i>	<i>Datenschutz/ Informationelle Selbst- bestimmung</i>	(allgemeine) Bildung/ (Hoch-)Schule	Kommunikations- medien
		Netiquette TK-Gesetzte		
<b>Wissen</b> Instrumente Dienste	<i>System- entwicklung als Anpassung</i>	Patentrecht Copyright	<i>Fachgesell- schaften/Verant- wortung</i>	Wissens- gesellschaft
„Informatik im Kontext“			„Kontext der Informatik“	

**Abbildung 33.** Vorschlag für eine Strukturierung des Grundlagenbereichs (nach Engbring & Selke, 2013)

Des Weiteren schlagen wir vor, dazu auch historische Bezüge heranzuziehen. Die Technikgenese könnte an einzelnen Beispielen nachvollzogen werden: etwa wie Textverarbeitung mit zunehmenden Layoutfähigkeiten und die Weiterentwicklung der Drucktechnik das Berufsbild des Setzers/Druckers verändert hat und mehr und mehr die Anwenderinnen und Anwender das Erscheinungsbild ihrer Druckerzeugnisse selbst gestalten und produzieren.

Dieser Bereich beschäftigt sich also insgesamt vor allem mit gesellschaftlichen, gesetzlichen und ethischen Aspekten im Zusammenhang mit der Informatik. Beispielhafte Kompetenzerwartungen in diesem Bereich sind:

### ■ Arbeit/Arbeitsprozesse

Die Kinder ...

- können altersentsprechend Vor- und Nachteile des Einsatzes von Robotern oder anderen Informatiksystemen benennen.
- kennen Schritte der Entwicklung und Digitalisierung von Schreibsystemen.

### ■ Kultur/Kommunikationsmedien

Die Kinder ...

- finden Unterschiede der Interaktion/Kommunikation mit einem Informatiksystem oder mit einem Menschen.
- können erläutern, wie sie mit ihren persönlichen Daten im Internet umgehen sollen.
- können altersentsprechend die Auswirkungen von Cybermobbing benennen.
- können Nutzungsregeln für den Umgang mit sozialen Netzwerken aufstellen und altersentsprechend formulierte Regeln bewerten.

### ■ Wissen/Wissensgesellschaft

Die Kinder ...

- erkennen, dass Informatiksysteme „intelligentes“ Verhalten modellieren.

### ■ Allgemein/einführend

Die Kinder ...

- benennen Vor- und Nachteile der Durchdringung der Alltagswelt mit Informatiksystemen und -werkzeugen, z. B. Sprachnachrichten, Videos abspielen oder herunterladen, Internetzugriff auf mobilen Geräten, Navigationssysteme etc.
- benennen, in welchen Bereichen in ihrer Umgebung Informatiksysteme genutzt werden.

## 4 Zieldimensionen auf Ebene der pädagogischen Fach- und Lehrkräfte

Die professionelle Kompetenz von Lehrkräften und ihre berufsbezogenen Einstellungen sind wesentliche Einflussfaktoren für die Qualität von Unterricht und Lernangeboten. Dies hat nicht zuletzt die breit angelegte empirische Meta-Studie von Hattie gezeigt (Hattie, 2009). Für verschiedene Unterrichtsfächer, auch im MINT-Bereich, wurden als theoretische Basis für die empirische Analyse von Unterrichtsqualität Kompetenzmodelle für Lehrkräfte konzipiert, die sich in ihrem Ursprung auf ein von Shulman entwickeltes Konzept professioneller Kompetenz beziehen (Shulman, 1986, 1987). Es sieht als wesentliche Komponenten das fachliche Wissen („Content Knowledge“ (CK)), das pädagogisch-psychologische Wissen („Pedagogical Knowledge“ (PK)) und als Kombination von beiden das fachdidaktische Wissen („Pedagogical Content Knowledge“ (PCK)) vor. Carlsen hat den Ansatz von Shulman dahingehend verändert, indem das curriculare Wissen der Lehrkräfte in eine fachspezifische und eine interdisziplinäre Komponente aufgeteilt wurde (Carlsen, 2002). Eine gute Übersicht über die Weiterentwicklung und Differenzierung der PCK-Modelle liefern Magnusson et al. (1999) und Fernandez (Fernandez, 2014; Magnusson, Krajcik & Borko, 1999). Im Fach Physik (z. B. Riese, 2009), in Biologie (z. B. Rozenszajn & Yarden, 2014) und besonders im Fach Mathematik (z. B. Lindmeier, 2011) wurden diese generischen Konzepte in einer Reihe von Studien wie MT21 (Blömeke, 2008; Blömeke, Kaiser & Lehmann, 2011), TEDS-M (Döhrmann, Kaiser & Blömeke, 2010) und COACTIV (Kunter & Baumert, 2011) jeweils unter fachdidaktischen Gesichtspunkten spezifiziert und partiell empirisch überprüft.

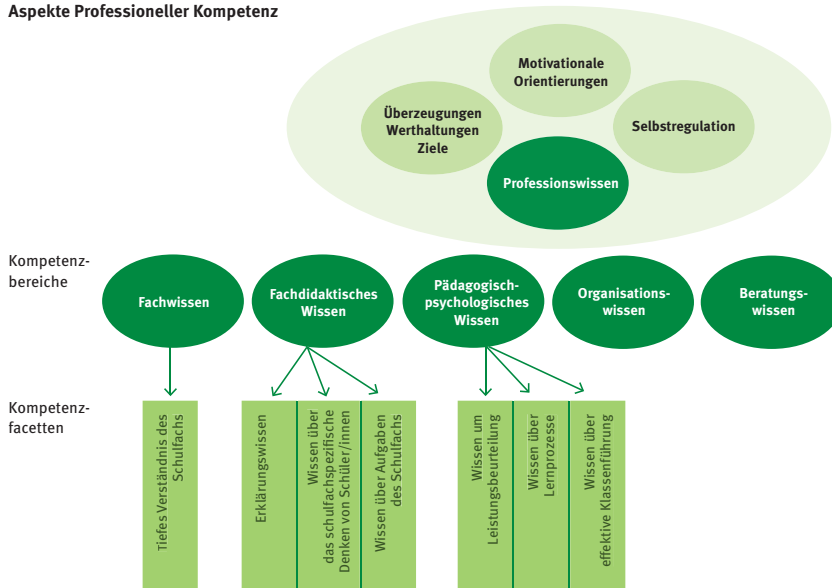
Modelle zur Beschreibung der professionellen Kompetenz von pädagogischen Fach- und Lehrkräften für das Fach Informatik und die informatische Bildung sind bisher kaum vorhanden und empirisch noch wenig gesichert (Hubwieser, Magenheimer, Mühling & Ruf, 2013; vgl. z. B. Saeli, 2012). In dieser Hinsicht ist die Forschungslage zur Professionalisierung von pädagogischen Fach- und Lehrkräften des Faches Informatik eher mit der Situation in der technischen Bildung (Kosack et al., 2015) als mit der des Faches Mathematik vergleichbar (vgl. Benz et al., 2017).

Als Ausgangspunkt für die Beschreibung von Zieldimensionen für pädagogische Fach- und Lehrkräfte dient uns das Modell der professionellen Kompetenz von Baumert und Kunter (Kunter & Baumert, 2011), auf das sich auch andere Konzepte (z. B. Döhrmann, Kaiser & Blömeke, 2012) beziehen.

Wichtige Aspekte der professionellen Kompetenz sind demnach ‚motivationale Orientierungen‘, ‚Überzeugungen, Werthaltungen und Ziele‘, die Fähigkeit zur ‚Selbstregulation‘ und auf kognitiver Ebene das ‚Professionswissen‘. Letzteres

untergliedert sich in das Fachwissen (CK), das fachdidaktische Wissen (PCK) und das pädagogisch-psychologische Wissen (PK). Ferner sind Organisations- und Beratungswissen für eine Tätigkeit im pädagogischen Kontext von Bedeutung (siehe Abbildung 34).

#### Aspekte Professioneller Kompetenz



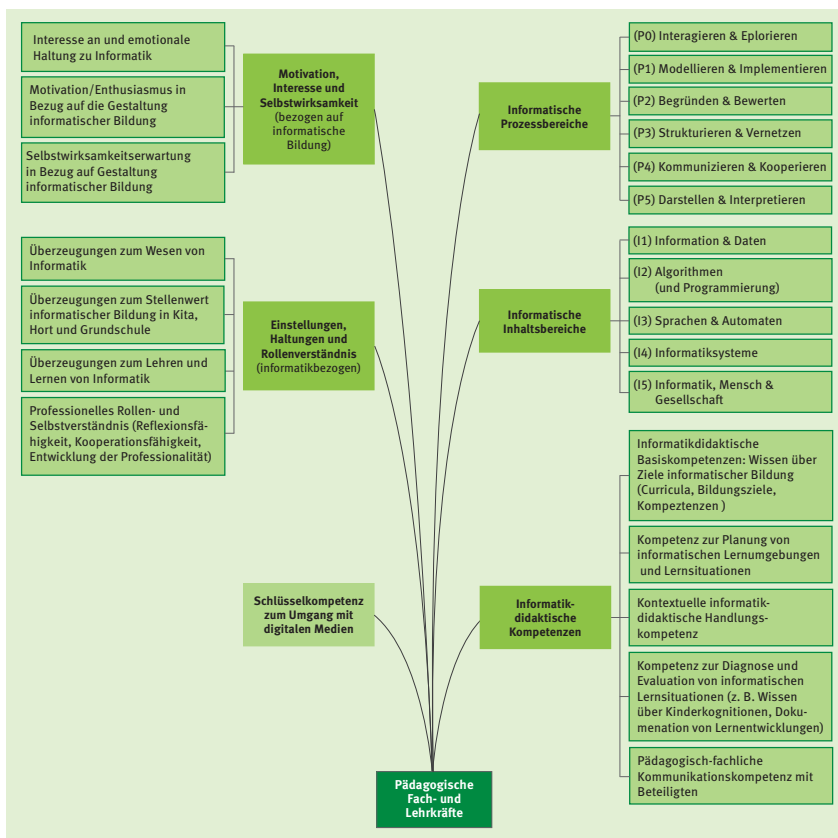
**Abbildung 34.** Aspekte professioneller Kompetenz (nach Baumert & Kunter, 2011, S. 32)

Mit motivationaler Orientierung wird die Disposition und Ausdauer von Lehrkräften beschrieben, mit der sie sich ihrer beruflichen Tätigkeit widmen. Die Selbstregulation beschreibt die Fähigkeit zur situativen Anpassung des eigenen pädagogischen Verhaltens an die jeweiligen spezifischen Rahmenbedingungen des Lernkontextes, wie er etwa mit Kategorien des ‚Hamburger Modells‘ (Schulz, 1997) charakterisiert werden kann. Schließlich sind die Vorstellungen der Fach- und Lehrkräfte vom eigenen Rollenbild, die damit verbundenen Überzeugungen und Werthaltungen zum Unterrichten sowie die Bedeutung des zu unterrichtenden Faches für die Bildung von Kindern und Jugendlichen von hoher Relevanz für die professionelle Kompetenz für ihre pädagogische Praxis.

Dieses generische Modell der professionellen Kompetenz von Lehrkräften wurde primär in Bezug auf die Lehrtätigkeit im schulischen Unterricht entwickelt. Insofern muss es im Hinblick auf die organisatorischen und inhaltlichen Rahmenbedingungen für Fach- und Lehrkräfte an Grundschulen und in der Elementarerziehung angepasst und für das Fach Informatik und die informatische Bildung spezifiziert werden. Soweit es die Forschungslage zulässt, soll das in den folgenden Abschnitten dieses Kapitels geschehen.

Im Unterkapitel 4.1 werden wir uns zunächst mit der Motivation, dem Interesse und der Selbstwirksamkeit von pädagogischen Fach- und Lehrkräften auseinandersetzen. Abschnitt 4.2 ist den Einstellungen, Haltungen und dem Rollenverständnis der Pädagoginnen und Pädagogen gewidmet. Die Bedeutung des pädagogisch-psychologischen Wissens (PK) der Fach- und Lehrkräfte im Hinblick auf die Gestaltung von informatikbezogenen Lernprozessen in Kita und Grundschule wird ebenfalls knapp in Abschnitt 4.2 diskutiert. In den Abschnitten 4.3 und 4.4 werden erforderliche informatische Fachkompetenzen (CK) und fachdidaktische Kompetenzen (PCK) der Fach- und Lehrkräfte zur Gestaltung von Lernsituationen der informatischen Bildung im Elementar- und Primarbereich vorgestellt. Abschließend werden fachübergreifende interdisziplinäre Aspekte des Lernens mit digitalen Medien aus informatischer Perspektive erörtert.

Abbildung 35 gibt einen Überblick über die Zieldimensionen informatischer Bildung für pädagogische Fach- und Lehrkräfte, deren einzelne Komponenten nun näher beschrieben und begründet werden.



**Abbildung 35.** Zieldimensionen früher informatischer Bildung auf Ebene der pädagogischen Fach- und Lehrkräfte



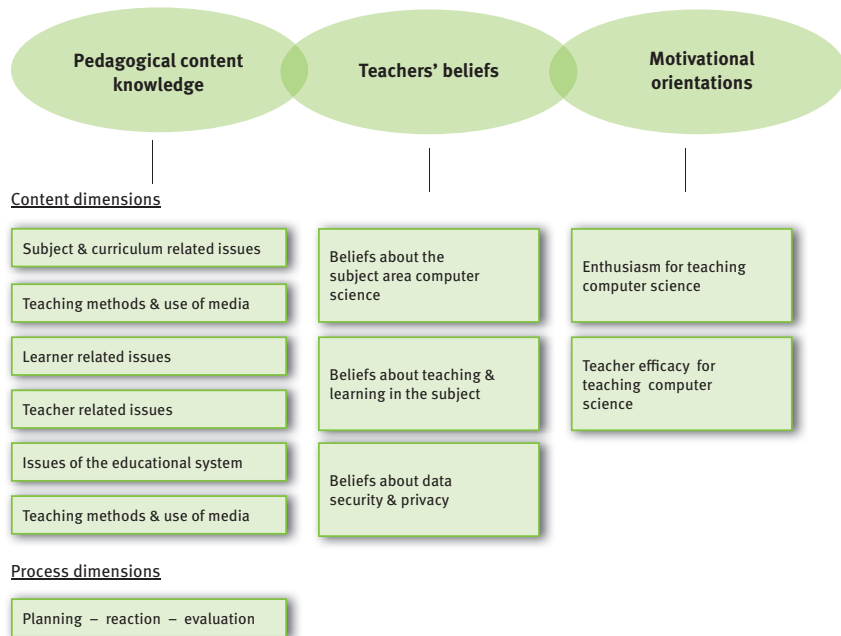
## 4.1 Motivation, Interesse und Selbstwirksamkeit

Die Motivation und das Interesse von Lehrkräften an ihrem Fach und deren Selbstwirksamkeitskonzept sind u. a. nach Baumert und Kunter von erheblicher Bedeutung für deren pädagogische Praxis (Kunter & Baumert, 2011). Für die informatische Bildung gibt es hinsichtlich dieser Kompetenzkomponenten bisher vor allem empirische Befunde auf der Ebene von Schülerinnen und Schülern, jedoch erst wenige für Lehrkräfte.

So wurde im DFG-Forschungsprojekt MoKoM (Modellierungskompetenzmessung) (Magenheim et al., 2010) ein theoretisch fundiertes, fachliches Kompetenzmodell entwickelt, das in Anlehnung an den Kompetenzbegriff von Weinert auch nicht-kognitive Kompetenzdimensionen wie motivationale und volitionale Elemente enthält (Weinert, 2001). In einer breit angelegten empirischen Studie mit Schülerinnen und Schülern des Fachs Informatik in der Sekundarstufe II konnte die Relevanz dieser Kompetenzfacetten für das Kompetenzniveau der Probandinnen und Probanden nachgewiesen werden (Neugebauer, Magenheim, Ohrndorf, Schaper & Schubert, 2015). In verschiedenen anderen kleineren empirischen Studien wurden z. B. bei Studierenden der Fächer Informatik/Wirtschaftsinformatik der Einfluss der Technikakzeptanz nach dem Technikakzeptanzmodell TAM (Bagozzi, 2007) bezüglich der Nutzung eines Informatiksystems in einem Lernkontext nachgewiesen (Beutner, Kundisch, Magenheim & Zoyke, 2014). Auf der Ebene von Schülerinnen konnten in einer empirischen Studie positive Selbstwirksamkeitseffekte in einem Lernszenario nachgewiesen werden, in dem die Schülerinnen positive Erfahrungen mit Informatiksystemen und deren Gestaltung machen konnten (Leonhardt, 2015). Ausführlich wurden die Ergebnisse der Studie in Kapitel 3 dargestellt. Kind konnte in einer Studie mit Schülern und Schülerinnen der 2. Klasse zeigen, dass der Umgang mit Informatiksystemen und deren Programmierung (Scratch) positiven Einfluss auf die Selbstwirksamkeit der Schülerinnen und Schüler hatte und einen Beitrag zur Überwindung geschlechtsspezifischer Differenzen beim Umgang mit Informatiksystemen leisten konnte (Kind, 2015).

Die Bedeutung von Selbstwirksamkeitskonzepten für die unterrichtspraktische Arbeit und die professionelle Identität von ‚Computer Science Teacher‘ sowie deren Berücksichtigung bei Konzepten der Informatiklehrerausbildung haben Ni und Guzdial (2015) hervorgehoben. Zu ähnlichen Ergebnissen kommen Bender et al. bei der Auswertung von empirischen Daten zur Kompetenz von Informatiklehrerinnen im Rahmen des BMBF-Forschungsprojekts KUI ‚Kompetenzen für das Unterrichten in Informatik‘ (Bender et al., 2015). Die Daten der Studie liefern auch Hinweise auf die Bedeutung von intrinsischer Motivation bezüglich informatischer Themen und deren unterrichtlicher Umsetzung. Die Autoren spezifizieren das professionelle Kompetenzmodell von Baumert und Kunter (2011) für Informa-

tiklehrkräfte, indem sie den persönlichen Einstellungen und Haltungen der Lehrkräfte (beliefs) und ihrer motivationalen Orientierung – auch im Hinblick auf die wahrgenommene Selbstwirksamkeit beim Unterrichten von Informatik – als wichtige Kompetenzkomponente ausweisen (siehe Abbildung 36).



**Abbildung 36.** PCK zum Unterrichten von Informatik (Bender et al., 2015, S. 3)

Für die Zieldimensionen der pädagogischen Fach- und Lehrkräfte in der informatischen Bildung kann aus diesen Ergebnissen geschlossen werden, dass sowohl die eigene aktive Erfahrung mit Informatiksystemen als auch deren mediale Nutzung in von den Fach- und Lehrkräften organisierten Lernsituationen mit Kindern Bestandteil ihrer Ausbildung sein sollte.

## 4.2 Einstellungen, Haltungen und Rollenverständnis

Die Bedeutung von Einstellungen, Haltungen und dem beruflichen Rollenverständnis von Lehrkräften auf deren pädagogische Praxis sind seit langen in verschiedenen empirischen Studien nachgewiesen worden. Eine gute zusammenfassende Darstellung findet sich z. B. bei Eulenberger (2015). Auf der Ebene der Persönlichkeitsmerkmale wurden die ‚Big Five‘ herangezogen: Dies sind die Dimensionen Neurotizismus (N), Extraversion (E), Offenheit für Erfahrungen (O), Verträglichkeit (V) und Gewissenhaftigkeit (G) (vgl. Eulenberger, 2015, S. 1). Anders et al. (2013b) beschreiben wichtige Komponenten des Rollen- und Selbst-

verständnisses von pädagogischen Fach- und Lehrkräften mit den Kategorien ‚Reflexionsfähigkeit‘, ‚Offenheit‘, ‚forschende Haltung‘, ‚Kooperationsfähigkeit‘ und ‚Entwicklung der Professionalität u. a. durch Fortbildungsbereitschaft‘. Diese allgemeinen Kompetenzfacetten treffen sicher auch auf pädagogische Fach- und Lehrkräfte in der informatischen Bildung zu.

Kleickmann (2008) hat die Zusammenhänge von Lehrervorstellungen und Unterrichtspraxis für den naturwissenschaftlichen Unterricht in der Grundschule untersucht, die in unterrichtsmethodischer Hinsicht auch für die informatische Bildung von Bedeutung sein könnten:

„Je stärker Grundschullehrkräfte eine Vorstellung vertreten, der zufolge Schüler im naturwissenschaftlichen Grundschulunterricht am besten aus Erklärungen der Lehrkraft lernen und Schüler das Wissen eher passiv-rezipierend aufnehmen (‚Transmission‘), praktische Aktivitäten (‚hands-on-activities‘) wie das Durchführen von Versuchen eine hinreichende Bedingung für das Erreichen konzeptuellen Verständnisses seitens der Schüler darstellen (‚Praktizismus‘), Schüler im naturwissenschaftlichen Grundschulunterricht weitgehend selbstständig arbeiten bzw. lernen sollten, ohne dass prozessbezogene Hilfestellungen und Strukturierungsmaßnahmen durch die Lehrkraft notwendig wären (‚Laissez-faire‘), desto geringer fallen die Fortschritte der Schüler im naturwissenschaftlichen konzeptuellen Verständnis aus“ (Kleickmann, 2008, S. 172).

Davis (2008) weist in einer fallbasierten Langzeitstudie zur Einstellung von Grundschullehrerinnen und Grundschullehrern über ‚Effective Science Teaching‘ nach, dass derartige Einstellungen zwar relativ stabil sind, sich aber mit der Zeit in einer Rückkopplung zur Praxiserfahrung der Lehrkräfte tendenziell weg von ‚reform-orientierten‘ Konzepten zu eher ‚konservativen‘ lehrerzentrierten Vorstellungen von Unterricht bewegen.

Fulton hatte in einer Untersuchung von ‚beliefs‘ von Lehrerinnen und Lehrern der Mittelstufe bereits früher darauf hingewiesen, dass die Einstellungen der Lehrkräfte zu Lehrstilen deren Einsatz von digitaler Unterrichtstechnologie dominieren:

„Results suggest that technology use did match teaching beliefs. Teachers with constructivist teaching beliefs adopted technology for a learner-centered teaching style, while those with more traditional (non-constructivist) teaching beliefs used technology in a more teacher-centered transmission style. Teachers said that technology has not changed their

pedagogical beliefs, but the opportunities they have had to work in this learning community also affected teaching practice.“ (Fulton, 1999, S. 1).

Gil, Schwarz und Asterhan (2007) haben den Einfluss von ‚beliefs‘ der Lehrkräfte im Hinblick auf den Einsatz digitaler Medien in Lernsituationen untersucht und gezeigt, dass die kognitiven Kommunikationsstile (intuitive moderation (intuitive Moderation), synchronous discussion organizing (Organisation synchroner Diskussion), guiding (begleiten), observing (beobachten), involved and authoritative (eingebunden und verbindlich)), der Lehrpersonen in diesen Situationen von ihren ‚beliefs‘ ihren ‚technischen Fähigkeiten‘ und den ‚Constraints‘ der Lernumgebung abhängen.

Der Einfluss von ‚beliefs‘ auf die Unterrichtspraxis wurde von Schulte und Bennedsen bei Informatiklehrerinnen und -lehrer im Kontext von Kursen zur objektorientierten Programmierung nachgewiesen sowie von Fessakis und Karakiza (2011) anhand einer Studie bei griechischen Informatiklehrerinnen und -lehrern (siehe auch Schulte & Bennedsen, 2006).

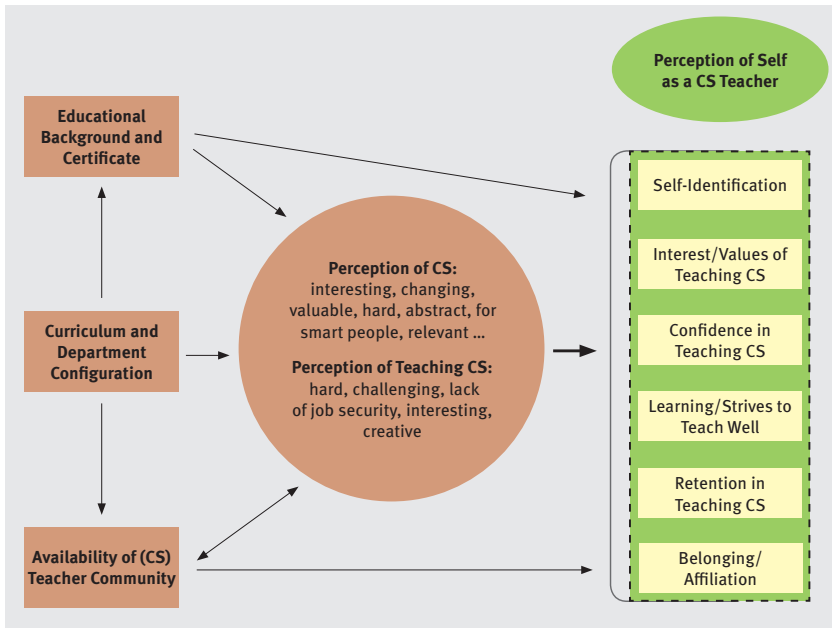
Für Informatiklehrkräfte hat Ni in ihrer Untersuchung zur professionellen Identität von ‚CS Teachern‘ den Einfluss der ‚beliefs‘ auf die Unterrichtspraxis spezifiziert (Ni, 2011). Sie ermittelte neben den generellen Einflussfaktoren ‚Perception of CS‘ und ‚Perception of Teaching‘ die Kriterien ‚Self-Identification‘, ‚Interest/Values of teaching CS‘, ‚Confidence in CS Teaching‘, ‚Learning Styles to Teach Well‘, ‚Retention in Teaching CS‘, ‚Belonging/Affiliation‘. Besondere Bedeutung weist sie dabei der guten Einbettung in eine Bildungseinrichtung und in ein professionelles soziales Netzwerk zu (siehe Abbildung 37), das durchaus im Sinne von Wenger (1998) als ‚Community of Practice‘ zum Kompetenzerwerb entsprechend des Konnektivismus verwendet werden kann (Siemens, 2008).

Schließlich kamen Bender et al. im KUI-Projekt hinsichtlich des Einflusses von nicht-kognitiven Kompetenzstrukturelementen zu folgender zusammenfassender Einschätzung (Bender, Schaper, Caspersen, Margaritis & Hubwieser, 2016):

„The described theoretical analysis leads to assumptions that two main areas of beliefs (beliefs about the subject and its teaching and learning, and beliefs about data security and privacy) and two main areas of motivational orientations (intrinsic motivational orientations and teaching efficacy) are particularly important for teaching computer science“ (Bender et al., 2016, S. 6).

Die Reflexion des möglichen Einflusses von Einstellungen der Lehrkräfte zu eingesetzten Lernmedien, insbesondere zu digitalen lernunterstützenden Medien auf

ihr pädagogisches Verhalten in Lernsituationen der informatischen Bildung sollte als Element der Zieldimensionen ebenso Beachtung finden wie die Einbettung des Diskurses und des Informationsaustauschs über dieses pädagogische Handeln in sozialen Netzwerken einer ‚Communities of Practice‘.



**Abbildung 37.** Main Influencing Factors for CS Teacher Identity Formation (Ni, 2011, S. 77)

Die lern- und entwicklungspsychologischen Voraussetzungen für die Realisierung von informatischen Lernprozessen mit Kindern im Kindergarten und in der Grundschule wurden bereits in Abschnitt 2.2 beschrieben. Die pädagogischen Fach- und Lehrkräfte sollten über solides Wissen in diesem Bereich verfügen und in der Lage sein, zielgruppenadäquate Lernszenarien zur informatischen Bildung unter Berücksichtigung der entwicklungs- und lernpsychologischen Fähigkeiten der Kinder zu organisieren. Dazu gehört auch eine allgemeine lernpsychologische Diagnosefähigkeit bezüglich der Lernfähigkeiten der konkret zu betreuenden Lerngruppe als Voraussetzung für die Planung von Lernszenarien. Diese Form des Wissens kann zusammenfassend als pädagogisch-psychologisches Wissen (PK) der Fach- und Lehrkräfte bezeichnet werden (siehe auch Abbildung 34).

### 4.3 Informatische Fachkompetenz

Zur Charakterisierung der von pädagogischen Fach- und Lehrkräften erwarteten informatischen Fachkompetenz kann zunächst auf verschiedene Modelle und Empfehlungen zurückgegriffen werden. Diese wurden jedoch primär für Informatik-Fach- und -Lehrkräfte entwickelt, die in der außerschulischen Arbeitswelt oder in höheren Stufen des schulischen Bildungswesens eingesetzt werden. Somit können sie nur bedingt der Begründung von Zieldimensionen für pädagogische Fach- und Lehrkräfte im Elementar- und Primarbereich dienen. Da die meisten der neueren Empfehlungen fachliche Kompetenzmodelle verwenden, sollen diese hier zumindest referenziert und zu dem von uns verwendeten Modell in Beziehung gesetzt werden. Eine aktuelle kompetenzorientierte Darstellung der fachlichen Kompetenzen von Informatikerinnen und Informatikern auf Bachelor- und Masterniveau liefern die GI-Empfehlungen zur Gestaltung von Bachelor- und Masterstudiengängen (GI – Gesellschaft für Informatik e. V., 2016b). Die dort beschriebenen Kompetenzerwartungen liegen in Umfang und Niveau natürlich weit über den erforderlichen Kompetenzerwartungen für die hier adressierte Zielgruppe der pädagogischen Fach- und Lehrkräfte. Allerdings werden Teile dieser Kompetenzen in die Kompetenzerwartungen für Informatiklehrkräfte an allgemeinbildenden Schulen einfließen. Für diese Informatiklehrkräfte liegen im fachlichen und fachdidaktischen Bereich Empfehlungen der Ständigen Kultusministerkonferenz der Länder vor (Kultusministerkonferenz, 2015), die u. a. die folgenden, gegenüber den GI-Empfehlungen von 2016, deutlich reduzierten Inhaltsbereiche enthalten:

- Formale Sprachen und Automaten
- Algorithmen und Datenstrukturen
- Datenmodellierung und Datenbanksysteme
- Programmierung und Softwaretechnik
- Rechnerstrukturen und Betriebssysteme und
- Informatik, Mensch und Gesellschaft sowie
- die Fachdidaktik

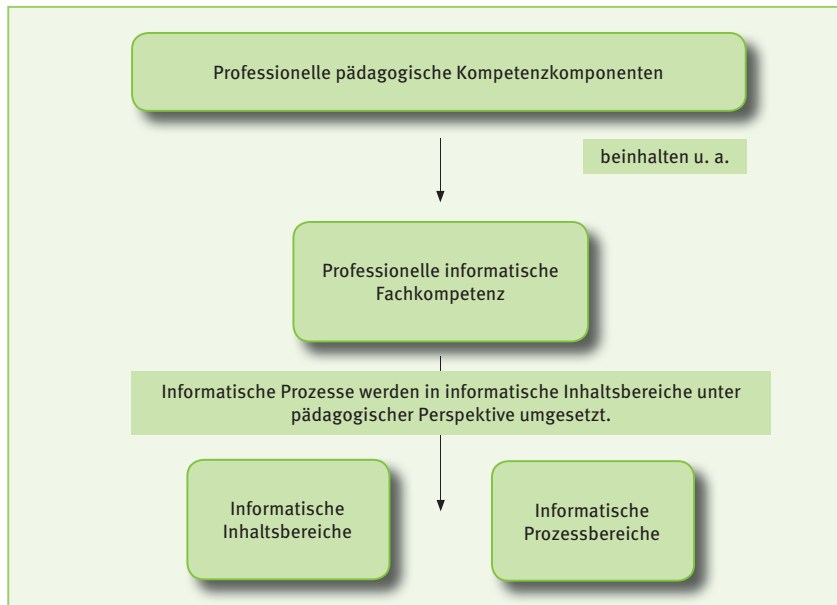
Hierbei zeigt sich eine starke Übereinstimmung mit den Inhaltsbereichen der GI-Bildungsstandards (GI – Gesellschaft für Informatik e. V., 2008, 2016a).

Während die GI-Empfehlungen für Informatikstudiengänge (GI – Gesellschaft für Informatik e. V., 2016b) eine Stufung nach Kompetenzniveaus entsprechend einem adaptierten AKT-Modell (Krathwohl, 2002) vornehmen (Bröker, Kastens & Magenheimer, 2014), enthalten die ‚KMK-Empfehlungen‘ und die GI-Bildungsstandards keine solche Stufungen. Andere Formen der Gradierung von fachlichen informatischen Kompetenzen finden sich bei Fuller et al. (2007), in der SOLO-Taxonomy (Biggs & Collis, 1982) oder im empirisch fundierten MoKoM-Kompetenz-Niveaumodell (Neugebauer et al., 2015). Empirische Untersuchungen zeigen, dass sich Elemente dieser Kompetenzstufenmodelle und der oben beschriebenen Fachinhalte sowohl in den fachlichen Ausbildungscurricula von Informatik-Bachelorstudiengängen an Hochschulen (Brabrand & Dahl, 2009; Müller, 2015) als auch in den fachlichen Teilen der Ausbildungscurricula für Informatik-Lehramtsstudierende wiederfinden (Hubwieser et al., 2013).

Es macht im Zusammenhang mit der Zielsetzung dieser Expertise jedoch wenig Sinn, diese zum Teil recht komplexen Kompetenzstufenmodelle zur Kompetenzbeschreibung der pädagogischen Fach- und Lehrkräfte im Elementar- und Primarbereich heranzuziehen, da für diese Zielgruppe bisher weder ein theoretisch hinreichend begründetes Kompetenzstrukturmodell vorliegt noch daraus abgeleitete, empirisch überprüfbare Stufungsmodelle entwickelt wurden. Um dies leisten zu können, bedarf es zunächst weiterer, teilweise langwieriger fachdidaktischer empirischer Forschung.

Wir orientieren uns bei den Zielsetzungen für pädagogische Fach- und Lehrkräfte hinsichtlich der fachlichen Anteile der zu erwerbenden Kompetenzen zunächst an den GI-Bildungsstandards für die Sekundarstufe I bzw. dem Entwurf für die GI-Bildungsstandards in der Grundschule (vgl. auch Kapitel 3). Es werden darüber hinaus aber auch weitere fachwissenschaftliche Argumente und fachdidaktische Kriterien mit berücksichtigt. Mit dieser Vorgehensweise stehen wir auch im Einklang mit den entsprechenden Empfehlungen in der Expertise für das Fach Mathematik (Benz et al., 2017). Beim Erwerb von informatischen Fachkompetenzen soll jeweils ein enger Bezug zu den später für die Kinder zu organisierenden Lernszenarien hergestellt werden, so dass die gewonnenen Fachkenntnisse schnell in das eigene pädagogische Handeln integriert werden können. Daher orientieren sich die Zieldimensionen für pädagogische Fach- und Lehrkräfte, deren Auswahl und Priorisierung maßgeblich an den in Kapitel 3 dargestellten Zieldimensionen für Kinder. Auch hinsichtlich der methodischen Umsetzung in Lernszenarien steht die Orientierung an den Interessen und Fähigkeiten der Kinder im Mittelpunkt. Die informatischen Fachkompetenzen werden in Anlehnung an die GI-Bildungsstandards, wie bereits für die Ebene der Kinder dargestellt, aus der Integration von Inhalten und Prozessen erzeugt, indem die jeweiligen Kompetenzerwartungen als konkrete Kontextualisierung von informatischem

Prozesshandeln von Lernenden in einem Inhaltsbereich beschrieben werden (vgl. Kapitel 3 und Abbildung 38).



**Abbildung 38.** Professionelle Pädagogische Kompetenzkomponenten in der informatischen Bildung

Dieses kontextualisierte informatische Prozesshandeln von pädagogischen Fach- und Lehrkräften erfolgt in Lernszenarien mit Kindern zudem immer unter der Perspektive ihrer professionellen pädagogischen Kompetenz. Daher wird es in den folgenden Darstellungen der Prozess- und Inhaltsbereiche einerseits immer den Verweis auf die entsprechenden Kompetenzerwartungen an die Kinder geben (vgl. Kapitel 3), andererseits sollen aber jeweils auch die professionellen pädagogischen Kompetenzerwartungen in diesen Lernkontexten erörtert werden.

#### 4.3.1 Inhaltsbereiche

##### *Information und Daten (11)*

Die Fähigkeit zur Unterscheidung zwischen Information und Daten ist eine wichtige inhaltliche Kompetenzfacette, über die auch die pädagogischen Fach- und Lehrkräfte verfügen müssen, wenn sie entsprechende informatische Lernsituationen für Kinder gestalten wollen. Dabei sollten verschiedene symbolische Repräsentationsformen von Daten (z. B. Buchstaben, Zahlen, Zeichen ...) bekannt sein, die auch in Lernsituationen mit Kindern verwendet werden können. Wichtig ist zudem das Ver-



ständnis der Transformation von Information in Daten und umgekehrt, die über den Mechanismus einer Bedeutungszuweisung und Interpretation erfolgt.

So kann die Buchstabenfolge ‚CAB‘ als ein Akronym in lateinischer oder kyrillischer Schreibweise angesehen werden oder die Zahl 3243 in Hexadezimaldarstellung repräsentieren. Entscheidend für die Interpretation der Daten ist ein geeigneter Interpretationskontext. Die Fach- und Lehrkräfte sollten Grundkonzepte der binären Codierung von Daten kennen (Zahlen, Buchstaben, Bilder, Grafiken), die deren automatische Verarbeitung und Speicherung in einem Computer ermöglichen. Es sollte ein Verständnis dafür entwickelt werden, dass ...

- in Abhängigkeit von der Datenrepräsentation eines Objekts der realen Welt unterschiedliche Operationen auf den Daten ausgeführt werden können.
- die Datenrepräsentation eines realen Objekts Prozesse der Abstraktion und Dekontextualisierung beinhalten.
- diese Prozesse zum Teil irreversibel sind.
- sich unterschiedliche Darstellungen für bestimmte Operationen unterschiedlich gut eignen.

Neben den in Kapitel 3.4 bzw. Kapitel 5 dokumentierten Beispielen für Kinder können für Fach- und Lehrkräfte zur Veranschaulichung dieses Inhaltsbereichs z. B. folgende Darstellungen verwendet werden: die Luftaufnahme einer Landschaft – eine Flurkarte derselben Landschaft – die abstrakte Repräsentation der Karte mittels Flurbezeichnungen und Grundstücksmaßen – ein Datensatz mit diesen Grundstücksdaten, wobei man aus Letzteren zwar die Grundstücksgröße nicht aber den Bewuchs des Grundstücks ermitteln kann. Als Ausgangspunkt für weitere Beispiele, die näher an der Alltagswelt der Kinder sind, können z. B. Bilder einer Pizza, einer Torte oder eines Hauses dienen.

#### *Algorithmen und Programmierung (12)*

Der für die Informatik essenzielle Begriff des Algorithmus, ist auch für die Konstruktion von Kompetenzerwartungen für die pädagogischen Fach- und Lehrkräfte ein wesentlicher Inhaltsbereich. Er kann in Orientierung an die mit den Kindern spielerisch zu erkundenden Szenarien erarbeitet werden, z. B. mit Hilfe von Wegbeschreibungen, Spielregeln, Pantomime, Schatzsuche, Kochrezepten oder Malanleitungen. Hierbei sind die Unschärfe von Alltagsalgorithmen herauszuarbeiten (‚Man nehme  $\frac{1}{2}$  Tasse Milch und etwas Butter.‘) und mit den Anforderungen an einen von einem Operator präzise auszuführenden Algorithmus zu kontrastieren:

Eindeutigkeit der Anweisungen, endliche Abfolge von Anweisungen, Terminierung des Programms. Hierbei sollten die wesentlichen Kontrollstrukturen eines Algorithmus wie Sequenz, bedingte Verzweigung und Schleifen/Wiederholungen ebenso kennengelernt werden wie ein elementares Variablenkonzept mit elementaren Datentypen. Es ist für die Organisation informatischer Lernprozesse eine wichtige Voraussetzung, dass die pädagogischen Fach- und Lehrkräfte diese grundlegenden Konzepte sicher verstanden haben und entsprechend sprachlich gegenüber den Kindern verbalisieren können.

Bei der spielerischen Umsetzung des Algorithmusbegriffs in Lernszenarien kann der Operator zunächst ein Mensch sein, der den Anweisungen eines auf Papier geschriebenen ‚Programms‘ folgt und z. B. einen Papierroboter steuert (Unplugged-Version). Später kann der Operator auch ein Computer sein, der zunächst ein in Form von Puzzleteilen entwickeltes Programm auf Papier, das dann computerlesbar als Programmcode eingegeben wurde, auf dem Bildschirm ausführt, z. B. mit der visuellen Programmiersprache Scratch. Hier wird ein virtueller Roboter auf dem Bildschirm in ähnlicher Weise bewegt (virtuelle Version). Schließlich könnte mit dem gleichen Programm in der gleichen Programmiersprache auch ein kleiner Roboter in der realen Welt gesteuert werden (reale Welt Version). Es liegt nahe, den Einstieg in die Algorithmik mit einer Unplugged-Version zu beginnen. Für die pädagogischen Fach- und Lehrkräfte ist es empfehlenswert, dass sie wegen der oben beschriebenen Effekte der Selbstwirksamkeit und Motivation zudem die weiteren Versionen der algorithmischen Steuerung kennenlernen (siehe auch Abschnitt 4.1).

### *Sprachen und Automaten (13)*

Formale Sprachen und Automaten besitzen in der Fachwissenschaft Informatik eine große Bedeutung. So kann z. B. mit dem für die Informatik zentralen theoretischen Konzept der Turingmaschine, einem abstrakten Automaten, die Ausführbarkeit eines Algorithmus getestet und dessen Komplexität beschrieben werden. Für pädagogische Fach- und Lehrkräfte kann es nicht darum gehen, diese zentralen Konzepte der theoretischen Informatik im Detail zu verstehen. Allerdings sollten sie die Grundprinzipien dieser Konzepte begreifen, um sie dann in pädagogischen Lernkontexten mit Kindern altersangemessen umsetzen zu können.

So sollte der Unterschied zwischen einer natürlichen Sprache und der von einem ‚Interpretationskontext‘ unabhängigen formalen Sprache, mit der man z. B. ein Computerprogramm schreiben kann, verstanden sein. Hier können Bezüge zu (11) hergestellt werden. Im Hinblick auf die Kontextabhängigkeit natürlicher Sprache können mit den pädagogischen Fach- und Lehrkräften Kommunikationsmodelle diskutiert werden, wie etwa das des Symbolischen Interaktionismus (Blumer, 1969). Aufgrund ihrer pädagogischen Ausbildung sind derartige Modelle

den Fach- und Lehrkräften ggf. bereits bekannt. Im Gegensatz dazu kann dann das Konzept formaler Sprachen in seiner Unterschiedlichkeit herausgearbeitet werden. Dies kann durchaus spielerisch und ‚unplugged‘ erfolgen. Zur Veranschaulichung der Kontextabhängigkeit der natürlichen Sprache kann z. B. das Spiel ‚Stille Post‘ herangezogen werden. Grundprinzipien einer formalen Sprache können im Zusammenhang mit Kommunikationsprotokollen durch die Kommunikation mittels Morsezeichen (Trommeln) oder Lichtzeichen vermittelt werden, wobei die Kommunikation via Lichtzeichen auch mittels Binärcode erfolgen kann. Bei Bedarf können hier auch Prinzipien der Verschlüsselung oder Regeln für ‚Geheimsprachen‘ zum Einsatz kommen. Auch die Übermittlung von E-Mails kann im Rollenspiel anhand von ‚Routingprotokollen‘ spielerisch als Repräsentation einer formalen Sprache erkundet werden.

Im Zusammenhang mit (I2) können außerdem Programmiersprachen wie z. B. Scratch mit ihren Schlüsselworten und festen syntaktischen Regeln für die Gestaltung von Anweisungen als formale Sprachen in elementarer Form kennengelernt und angewendet werden.

Als Ausgangspunkt für das Erkunden des informatischen Konzepts eines Automaten können Automaten der realen Welt dienen, denen die pädagogischen Fach- und Lehrkräfte im Alltag begegnen: Fahrkartenautomaten, Geldautomaten, Ampeln, Getränkeautomaten etc.

Den pädagogischen Fach- und Lehrkräften muss der Unterschied zwischen einem Automaten als abstraktem theoretischen Konzept und einem Informatiksystem als dessen mögliche technische Realisierung verdeutlicht werden. Das nicht sichtbare abstrakte theoretische Konzept eines Automaten lässt sich z. B. über die sichtbaren Ein- und Ausgabemöglichkeiten eines Informatiksystems (z. B. Geldautomat) erschließen. Auch hier können die pädagogischen Fach- und Lehrkräfte anhand eines Rollenspiels wesentliche Elemente des Konzepts Automat kennenlernen: Ein- und Ausgaben mit entsprechenden Symbolen, innere Zustände, Übergangsregeln für Zustände in Form von Tabellen oder Graphen.

Wichtig ist bei den spielerischen Zugängen zu den Konzepten ‚formale Sprachen‘ und ‚Automat‘, dass diese am Ende des Spiels bezüglich ihrer wesentlichen Charakteristika auch formal, z. B. mit einer Grafik, beschrieben und verstanden werden. Damit kann zugleich der Bezug zu einem wichtigen Konzept des Modellierens (P1), der zustandsorientierten Modellierung, hergestellt werden.

#### *Informatiksysteme (I4)*

Für pädagogische Fach- und Lehrkräfte ist es wichtig, dass sie technische Artefakte der realen Welt (Phänomene der Informatik) als Informatiksysteme identifizieren, deren wesentliche sichtbaren und nicht sichtbaren Eigenschaften beschreiben und sie auch als spezifische technische Realisierungen von abstrakten

Automaten begreifen können. Auf dieser Grundlage sind sie zudem in der Lage, Lernszenarien zu organisieren, die den Kindern wichtige Charakteristika von Informatiksystemen näherbringen.

Informatiksysteme sind seit Langem ein fester Bestandteil unseres Alltags, ohne dass wir sie möglicherweise noch als solche wahrnehmen. Im Haushalt (Wasch- und Spülmaschine, Mikrowelle, Kaffeemaschine ...), bei Kommunikation und Freizeit (Smartphone, Tablet, Laptops, Fernseher, digitale Medienrecorder, Spielkonsole ...), unterwegs (Auto, Flugzeug, Parkleitsystem, GPS-System, Ampelanlage ...) oder beim Einkaufen (Scannerkassen, Bankautomaten, Ticketautomaten, Ausgangskontrollsysteme, Fahrstühle ...) begegnen sie uns.

Das Konzept des soziotechnischen Informatiksystems mit seinen technischen und sozialen Komponenten wurde bereits ausführlich in Kapitel 3 beschrieben. Informatiksysteme sind in der Regel sehr komplex und selbst von Fachleuten oft nur schwer zu durchschauen. Ein grundlegendes Verständnis über den Aufbau und die Funktionsweise von Informatiksystemen ist allerdings ein wesentlicher Aspekt von informatischer Bildung.

Diese grundlegenden Prinzipien sollten für die pädagogischen Fach- und Lehrkräfte an einfachen Beispielen erfahrbar gemacht werden. Die technischen Systeme in der Alltagswelt der Kinder und der pädagogischen Fach- und Lehrkräfte als soziotechnische Informatiksysteme zu identifizieren, einige Eingabemöglichkeiten, Systemreaktionen sowie Handlungsmöglichkeiten der Nutzerinnen und Nutzer zu beschreiben wäre ein erster Zugang zu Informatiksystemen.

Die Beispiele sollten nah an jenen Konzepten orientiert sein, die auch bei Lernszenarien mit den Kindern verwendet werden können. Dabei sollte sowohl die Außen- als auch die Innensicht auf ein Informatiksystem ermöglicht werden. Die Außensicht kann über die Analyse von Benutzungsschnittstellen (GUI: Graphical User Interfaces) und die damit verbundene Mensch-Maschine-Kommunikation (MMK) erfolgen, etwa indem man vertraute Software, z. B. oft genutzte Apps oder Websites, hinsichtlich der Eingabemöglichkeiten und Rückmeldungen analysiert. Optional könnte überlegt werden, ob der Aufbau und die Gestaltung einer Website anhand von MMK-Gesichtspunkten unter Beachtung der Trennung von Struktur, Layout und Inhalt, von pädagogischen Fach- und Lehrkräften zunächst ‚unplugged‘ zeichnerisch konzipiert, und dann mit einfachen HTML-Befehlen als Beispiel für eine Auszeichnungssprache (Markup Language) umgesetzt wird.

Über das EVA-Prinzip (Eingabe-Verarbeitungs-Ausgabe-Prinzip) kann man auf innere Vorgänge des Systems schließen und das Rückmeldeverhalten analysieren (z. B. anhand von Apps wie WhatsApp oder Google). Innere Zustände des Systems, die Verarbeitung von Eingaben und Erzeugung von Ausgaben, die u. a. mit Algorithmen beschrieben werden, können anhand einfacher Informatiksysteme (z. B. LEGO-Roboter, Computerspiel) mit spielerischen Konzepten modelliert und

mit einer visuellen Programmiersprache wie Scratch implementiert werden (vgl. Kapitel 3).

Ein weiterer wichtiger Bereich stellt die Vernetzung von Informatiksystemen und deren Kommunikation mittels technischer Protokolle dar. Mit einfachen spielerischen Mitteln sollten ‚unplugged‘ Grundprinzipien der Kommunikation im Internet erkundet werden, da ein grundlegendes Verständnis für die Funktionsweise des Internets ein wichtiger Bestandteil informatischer Bildung ist. Hier bieten sich Rollenspiele zum Mailversand über Router, der Aufruf und die Weiterleitung von Eingaben auf Websites zur Demonstration des Klient-Server-Prinzips oder als Beispiel für das Schichtenprotokoll des Internets die Telefonkommunikation zwischen Menschen in unterschiedlichen Sprachen mittels Dolmetscherinnen und Dolmetscher an.

#### *Informatik, Mensch und Gesellschaft (15)*

Auch wenn dieser sehr wichtige Bereich informatischer Bildung für Kinder nicht immer einfach verständlich in Lernszenarien umzusetzen ist, sollten pädagogischen Fach- und Lehrkräfte fundamentale Kenntnisse über diese Zusammenhänge erwerben.

Die allgegenwärtige Informationstechnologie hat in der Fachwissenschaft Informatik zu einer intensiven Diskussion über die gesellschaftliche Bedeutung von Informatiksystemen und deren Gestaltung für Individuum und Gesellschaft geführt. Die Begriffe ‚Informationsgesellschaft‘ oder ‚ubiquitous computing‘ sind wichtige Konzepte und Indikatoren für diese Diskussion. Die gesellschaftlichen Anwendungen und Auswirkungen von Informatiksystemen sowie grundsätzliche Fragen nach der Gestaltung und Verantwortbarkeit von Informationstechnologie werden als wichtiger Bestandteil von Allgemeinbildung angesehen und sind deshalb in den Informatiklehrplänen auf allen Bildungsebenen ab der Sekundarstufe I enthalten. Da es sich hierbei jedoch in der Regel um sehr komplexe wechselseitige Beziehungen handelt, die für Laiinnen und Laien nur schwer zu durchschauen sind, werden selbst für diese Bereiche des schulischen Bildungssystems derartige Themen oft nur exemplarisch und sehr grundlegend behandelt, ohne zu tief in die informatischen Aspekte der Probleme einzudringen. Bei der Zielgruppe der Kinder im



Grundschul- und Kindergartenalter stellen die kognitive Entwicklung und die Fähigkeit zur Bewertung ethisch-moralischer Fragen eine Hürde für die Gestaltung von Lernszenarien mit derartigen Themen dar. Falls mobile Geräte mit Apps im familiären Umfeld der Kinder oder in der Schule verwendet werden, könnte man die Weitergabe von persönlichen Daten in diesem Kontext kindgerecht thematisieren.

Für die pädagogischen Fach- und Lehrkräfte ist dies ebenfalls eine wichtige Fragestellung, wobei man z. B. die Nutzungsprofile und Datenschutzeinstellungen von oft genutzten Apps oder Programmen (z. B. Facebook, WhatsApp) analysieren könnte. Auf diese Weise könnten für diese Zielgruppe am konkreten Beispiel Grundprinzipien des Daten- und Urheberrechtsschutzes exemplarisch erfahrbar gemacht werden. Auf jeden Fall sollte vermittelt werden, welche Datenspuren man beim Surfen im Internet hinterlässt und wie diese für diverse, nicht für den ursprünglichen Verwendungszweck gedachten Auswertungen, genutzt werden können. Dies kann z. B. anhand von Einstellungen hinsichtlich Cookies in oft genutzten Browsern oder handlungsorientiert und ‚unplugged‘ mit einem Planspiel zum Datenschutz geschehen (Medienwissenschaft Universität Bayreuth, 2014).

#### **4.3.2 Prozessbereiche**

Wie oben in den Abschnitten 2.5.1 und 3.3 dargestellt, gehört zu der Beschreibung von Kompetenzerwartungen in Lernprozessen der informatischen Bildung auch die Darstellung von Prozessen, die in den jeweiligen Inhaltsbereichen angesiedelt sind. Schon bei der Beschreibung der Beispiele in den Inhaltsbereichen wurde deutlich, dass diese nicht ohne entsprechende Prozesse, die in den jeweiligen Anwendungskontexten von den Lernenden ausgeführt werden, darstellbar sind. In Anlehnung an die GI-Bildungsstandards für die Sekundarstufen I und II sowie den Entwurf der GI für die Primarstufe sollen auch bei den Kompetenzerwartungen und Zielsetzungen für die Lehrkräfte entsprechende Prozessbereiche formuliert werden.

##### **Interagieren und Explorieren (P0)**

Bereits in Kapitel 3 ist die Bedeutung dieses Prozessbereichs für die informatische Bildung der Kinder ausführlich dargestellt worden. Die pädagogischen Fach- und Lehrkräfte sollten die Argumentationsebenen für die Einführung dieses Prozessbereichs in der informatischen Bildung mit Kindern kennen und in der Lage sein, auf der Grundlage dieser Kenntnisse für die Kinder altersangemessene Lernszenarien zur informatischen Bildung zu organisieren.

Für die pädagogischen Fach- und Lehrkräfte ist es wichtig, dass sie den Kindern einerseits genügend Spielraum für deren eigene Erkundungen lassen, sie andererseits aber auch die von den Kindern gemachten Erfahrungen in geeigneter Weise systematisieren, damit diese sich ein mentales Modell zum Umgang, zur

Nutzung und zur Gestaltung von Informatiksystemen aufbauen können. Folgende Perspektiven sollten die pädagogischen Fach- und Lehrkräfte bei der Organisation von Lernszenarien hinsichtlich des Prozessbereichs ‚Interagieren und Explorieren‘ berücksichtigen:

#### *Lern- und entwicklungspsychologische Perspektive*

Kindliches Lernen in frühen Lebensphasen findet oft über Ausprobieren und Erkunden von Objekten der Lebenswelt statt. Daher liegt es nahe, diese Form des Lernens auch in der frühen informatischen Bildung zu nutzen. Durch Beobachten und Ausprobieren von geeigneten Informatiksystemen können Kinder Handlungs- und Gestaltungskonzepte im Umgang mit diesen kennenlernen. Ausgehend von den beobachteten Funktionen können Kinder Vermutungen über deren innere Struktur anstellen. Die pädagogischen Fach- und Lehrkräfte können dann durch die geeignete Gestaltung von Lernszenarien diese Erfahrungen und Vermutungen der Kinder aufgreifen, sie mit den Kindern systematisieren und verallgemeinern, um so den Kindern sukzessiv ein differenzierteres Bild von Informatiksystemen zu erschließen.

#### *Internationale Perspektive*

Wie bereits für die Zieldimensionen auf der Ebene der Kinder anhand zahlreicher Praxisbeispiele aus Projekten in verschiedenen Ländern dargestellt wurde, ist es international üblich, Kinder in der Grundschule und der Früherziehung Informatiksysteme spielerisch erkunden zu lassen und diese Systeme auf vielfältigste Art zu nutzen (vgl. Kapitel 3). Pädagogische Fach- und Lehrkräfte sollten einige der internationalen Konzepte informatischer Bildung zum Interagieren mit und Explorieren von Informatiksystemen kennen. Diese für die informatische Früherziehung entwickelten Konzepte können im Hinblick auf ihre explorativen und systemerkundenden Elemente auch in der informatischen Bildung von pädagogischen Fach- und Lehrkräften verwendet werden. So kann z. B. Resnicks methodischer Ansatz des ‚Tinkering‘, der sich auf handlungsorientierte Ansätze von Dewey (1938), Fröbel (1826) und Pappert (1993) bezieht, mittels der Prozessdimension des ‚Interagierens und Explorierens‘ praktisch umgesetzt werden.

„Froebel filled his kindergarten with physical objects (such as blocks, beads, and tiles) that children could use for designing, creating, and making. These objects became known as Froebel’s Gifts. Froebel carefully designed his Gifts so that children, as they played and constructed with the Gifts, would learn about common patterns and forms in nature“ (Resnick, 2013, S. 50).

Im Zeitalter digitaler Artefakte wird dieses Konzept von Resnick im Sinne von Papert auf digitale Mikrowelten im Computer (z. B. Scratch Programme) und digitale Objekte in der realen Welt (z. B. Mindstorms) erweitert. Dieses in Lernszenarien mit Kindern angewandte Konzept ist auch für die informatische Bildung pädagogischer Fach- und Lehrkräfte tragfähig.

„We see tinkering as a valid and valuable style of working, characterized by a playful, exploratory, iterative style of engaging with a problem or project. When people are tinkering, they are constantly trying out ideas, making adjustments and refinements, then experimenting with new possibilities, over and over and over“ (Resnick, 2013, S. 164).

#### *Fachdidaktische Perspektive*

In der Fachdidaktik Informatik wird dem Interagieren und Explorieren von Informatiksystemen seit jeher eine große Bedeutung beigemessen. Dabei wird an lernpsychologische Erkenntnisse angeknüpft wie etwa an die Cognitive Flexibility Theory (Spiro, Feltovich, Jacobson & Coulson, 1991), nach der man einen Gegenstandsbereich aus unterschiedlicher konzeptioneller Perspektive zu unterschiedlichen Zeitpunkten betrachten sollte, um ein vertieftes Verständnis über ihn zu gewinnen. Ein spielerischer, explorativer Zugang zu Informatiksystemen kann in diesem Sinne einen Ausgangspunkt für später nachfolgende weitere Betrachtungen darstellen und zusätzlich im Sinne von Selbstwirksamkeitskonzepten den selbstbewussten und reflektierten Umgang mit Informatiksystemen fördern. So könnte eine Fundierung für nachhaltige informatische Lernprozesse erfolgen.

In der fachdidaktischen Kompetenzforschung legen zudem theoretisch und empirisch abgesicherte Kompetenzmodelle zum informatischen Systemverständnis und zum Modellieren von Informatiksystemen nahe, dass das Erkunden und Anwenden von Informatiksystemen eine wesentliche Kompetenzfacette für das Verständnis von Informatiksystemen darstellt (z. B. Magenheim et al., 2010). Verschiedene fachdidaktische Konzepte haben diese Zugangsweise zur informatischen Bildung zuvor schon mit unterschiedlichen Argumenten zu begründen versucht. So betonen Vertreterinnen und Vertreter des anwendungsorientierten Ansatzes zur Didaktik der Informatik (Körber & Peters, 1988), dass Informatiksysteme stets in einem Anwendungskontext betrachtet werden sollten und die Nutzung und Erprobung eines Informatiksystems eine wesentliche Phase von informatikbezogenen Lernprozessen darstellt.

Noch einen Schritt weiter geht der benutzerorientierte fachdidaktische Ansatz (z. B. Forneck, 1992), in dem die konstruktive Entwicklung von Informatiksystemen zugunsten der fast ausschließlichen Nutzung und Bewertung von Informatiksystemen auf ein Minimum reduziert wird. Im systemorientierten Ansatz wird



in Anlehnung an die Cognitive Flexibility Theory ein vielfältiger Zugang zu Informatiksystemen empfohlen, indem man über die Methode der Dekonstruktion in einer initialen Explorationsphase Informatiksysteme und deren Funktionen erkundet, um später auch auf deren innere Struktur schließen zu können (Magenheim, 2008).

Im fachdidaktischen Ansatz zur Dualitätsrekonstruktion soll an die Alltagserfahrungen der Lernenden mit Informatiksystemen (z. B. Handy, digitale Media-Recorder, Standardsoftware) angeknüpft und die Funktionsweise digitaler Artefakte erkundet werden. In einem Prozess didaktischer Rekonstruktion soll die Dualität von Struktur und Funktion digitaler Artefakte erschlossen werden, wobei die Funktion sich auf den Einsatzzweck und die Nutzung des Artefakts im Alltag, die Struktur auf dessen inneren Aufbau bezieht (Schulte, 2009).

Das Erschließen informatischer Konzepte über das Erkunden der Funktionsweise von Informatiksystemen und des Wechselspiels von Verhalten und Struktur, von Außen- und Innensicht über den stufenweisen Wandel des Lernszenarios mit dem Informatiksystem von der ‚Black-Box‘ (nur Außensicht ist zugänglich) zur ‚White-Box‘ (innere Struktur des Informatiksystems, wie z. B. Sourcecode ist für Lernende zugänglich) wird in einem weiteren Ansatz von Stechert begründet (Stechert, 2009).

Auch in der fachdidaktischen Kompetenzforschung werden empirisch und theoretisch begründete Kompetenzmodelle diskutiert, die die Fähigkeit zum Erkunden und Anwenden von Informatiksystemen als eine wichtige Kompetenzfacetten zum Verständnis von Informatiksystemen ansehen (z. B. Magenheim, 2008).

Schließlich kann die Methodik des Experiments für das Erkunden von Informatiksystemen und digitalen Artefakten genutzt werden. Dabei zeigt sich, dass durch die zielgerichtete Interaktion anhand der Rückmeldungen des Systems auf innere Strukturen geschlossen werden kann (Schulte, 2012).

Als Konsequenz aus diesen fachdidaktischen Argumenten und empirischen Befunden sollte auch für die Zieldimensionen auf der Ebene der pädagogischen Fach- und Lehrkräfte neben einem Unplugged-Zugang zu informatischen Konzepten ein Umgang zumindest mit jenen Informatiksystemen eingeübt werden, die in Lernszenarien mit Kindern verwendet werden können. Ferner sollten die pädagogischen Fach- und Lehrkräfte einige der wesentlichen fachdidaktischen Argumente für das Interagieren und Explorieren mit Informatiksystemen kennen und so in der Lage sein, entsprechende Lernszenarien fachdidaktisch fundiert zu organisieren.

#### *Informatisch fachliche Perspektive*

Das Explorieren von und Interagieren mit bestehenden Informatiksystemen durch Expertinnen und Experten und Nutzerinnen und Nutzer ist bei der Weiter- und

Neuentwicklung von Informatiksystemen und deren Implementierung in der Praxis eine wichtige Phase der Systementwicklung. Auf diese Weise können über die grundlegende, äußerlich erkennbare Funktionsweise der Informatiksysteme deren Gestaltungsprinzipien erkannt und neue Systemkomponenten konstruiert bzw. das System insgesamt weiterentwickelt werden (vgl. z. B. Brandt-Pook & Kollmeier, 2008). Informatiksysteme und Software sollten überdies so konstruiert sein, dass sich die Funktionsweise der Systeme ebenso Laiinnen und Laien bei der Anwendung weitgehend selbsterklärend erschließt und für deren spezifische Nutzungsinteressen leicht konfigurierbar ist. Informatische Bildung sollte daher auch Laiinnen und Laien in die Lage versetzen, Informatiksysteme für ihre Zwecke zu spezifizieren und kompetent anzuwenden. Damit leistet informatische Bildung einen wichtigen Beitrag zur Allgemeinbildung.

Für die Zieldimensionen der pädagogischen Fach- und Lehrkräfte kann aus dieser Perspektive formuliert werden, dass sie auch die fachbezogenen informatischen Argumente für das Anwenden und Erkunden von Informatiksystemen kennen und sie praktisch in der Lage sind, diese im eigenen Arbeitskontext oder anhand eines kleinen Projekts umzusetzen. Dies könnte z. B. mit einem sehr einfachen, in Scratch programmierten Spiel geschehen. Die Funktionalität des Spiels könnte erkundet, eine mögliche Funktionserweiterung aus der Perspektive von Kindern antizipiert und diese dann durch eine leichte Modifikation des Quellcodes realisiert werden.

#### *Allgemeinbildende Perspektive*

Pädagogische Fach- und Lehrkräfte sollten sich der schon oben genannten allgemeinbildenden Aspekte des Erkundens und Anwendens von Informatiksystemen durch Kinder bewusst sein und sie in geeigneten Lernszenarien fördern. Mit dem Explorieren von Informatiksystemen kann bei geeigneter Moderation durch die pädagogischen Fach- und Lehrkräfte den Kindern ermöglicht werden, sich sukzessive ein differenzierteres mentales Modell von Informatiksystemen aufzubauen. Auf diese Weise werden sie auch in Zukunft befähigt, Informatiksysteme kompetent zu nutzen und technologischen Wandel hinsichtlich seiner Chancen und Risiken zu bewerten. Damit dieses anspruchsvolle Langfristziel gelingen kann, sollten pädagogische Fach- und Lehrkräfte in der Lage sein, Kinder beim Umgang mit altersgemäßen Informatiksystemen hinsichtlich ihrer Selbstwirksamkeit und Motivation zu stärken sowie ihre Reflexionsfähigkeit im Hinblick auf den Gebrauch der Informatiksysteme zu fördern. Entsprechende Kompetenzen bei den Kindern, aber auch bei den pädagogischen Fach- und Lehrkräften erfordern den interaktiven Umgang mit derartigen Systemen, der nicht nur in Lernszenarien mit Unplugged-Methoden zu realisieren ist.

### *Modellieren und Implementieren (P1)*

Das Modellieren eines Informatiksystems und dessen (software-)technische Implementierung mit Hilfe einer Programmiersprache sind wichtige Aufgabenbereiche der Informatik und stellen wesentliche Arbeitsschritte bei der Gestaltung eines Informatiksystems dar. Es kann für die Zielgruppe der pädagogischen Fach- und Lehrkräfte hier nicht darum gehen, unterschiedliche Vorgehensmodelle der Softwareentwicklung kennenzulernen. Allerdings sollten einige wichtige Phasen und die darin angewandten Methoden und Prozesse identifiziert werden, da diese z. B. auch bei der Entstehung von Lern- und Spielesoftware von hoher Bedeutung sind. Nach einer Phase der Anforderungsanalyse an die Software, sollten sich die Phasen des Modellierens und des Implementierens anschließen. Schließlich sollte die entstandene Software getestet und bewertet werden. Hier könnte auch ein Zusammenhang zu einem spezifischen Entdeckungs- und Forschungskreis für die informatische Bildung hergestellt werden (siehe Kapitel 1.5).

Denkbar wäre an dieser Stelle die ‚projektartige‘ Entwicklung eines kleinen Spiels mittels der Programmiersprache Scratch. Zahlreiche Beispiele für ein derartiges Vorhaben finden sich unter <https://scratch.mit.edu>. In dem ‚Projekt‘ könnten verschiedene wichtige Phasen durchgespielt werden. Ein solches Konzept würde vor allem die Prozessbereiche P2–P5 betreffen.

In der der eigentlichen Modellierung vorgeschalteten Phase des Ermitteln der Anforderungen an eine Software kann kooperativ festgelegt werden, welche Funktionen die zu entwickelnde (Spiele-)Software erfüllen soll. In der Modellierungsphase werden auf der Ebene formaler Beschreibung wesentliche Komponenten und Parameter des Informatiksystems sowie die Beziehungen zwischen ihnen festgelegt. Auch das Aussehen der Benutzungsschnittstelle wird in dieser Phase grundlegend festgelegt.

Für die pädagogischen Fach- und Lehrkräfte kann zu dieser Thematik auch ein spielerischer Unplugged-Zugang verwendet werden, indem die Modellierung von Teilen des Systems mittels CRC-Karten (Ambler, 1998) oder mit einem Rollenspiel (Objektspiel) erfolgt (Börstler & Schulte, 2005). Allerdings sollten diese der objektorientierten Modellierung zuzuordnenden Methoden nicht dazu führen, dass bei einer möglichen anschließenden Implementierung auch eine objektorientierte Programmiersprache verwendet wird. Weder kann es darum gehen, mit dieser Zielgruppe unterschiedliche Programmierparadigmen zu erarbeiten, noch darum, verschiedene Programmiersprachen kennenzulernen. Selbst eine visuelle Programmierumgebung mit ansatzweise objektorientiertem Konzept wie Alice, würde die Zielgruppe vermutlich mit unnötigen Lernbarrieren konfrontieren, wie entsprechende Erfahrungen mit Schülerinnen und Schülern der Sekundarstufe I gezeigt haben (Dohmen, Magenheimer & Engbring, 2009). Um mediale Brüche zu vermeiden, sollte man im Wesentlichen bei den pädagogischen Fach- und Lehrkräften

bei der Programmiersprache bleiben, die ggf. auch in Lernszenarien mit Kindern eingesetzt werden kann (z. B. Scratch oder ein verwandter Dialekt). So können beim Modellieren kleine Problemstellungen zunächst in noch kleinere Aufgaben zerlegt und strukturiert werden (siehe P3). Anschließend werden die so ermittelten Algorithmen als Handlungsvorschriften dargestellt (einfaches Diagramm) und in einer kindgerechten Programmiersprache wie z. B. Scratch implementiert. Dies kann zu visuellen Lösungen auf dem Bildschirm (z. B. Computerspiel) oder zum Bewegen von Objekten in der realen Welt führen (Roboter).

#### *Begründen und Bewerten (P2)*

Insbesondere bei der kooperativen Ermittlung von Systemanforderungen und in der Phase des Modellierens müssen Entscheidungen zur Gestaltung eines Informatiksystems getroffen werden. Gleiches gilt, wenn ein (selbstentwickeltes) Informatiksystem getestet und in seiner Funktionsweise und im Hinblick auf die angestrebte Problemlösung bewertet wird. In beiden Fällen sollten Kriterien erstellt werden, nach denen die informatischen Sachverhalte eingeschätzt werden können. Bei der Ermittlung der Systemanforderungen kann der Funktionsumfang (z. B. eines Spiels) festgelegt und eine Aufwandsschätzung zu dessen Realisierung abgegeben werden. Bei der Bewertung der fertigen Software können z. B. ergonomische Kriterien der Benutzungsoberfläche (z. B. Herczeg, 2009) und die Nützlichkeit der Systemfunktionen in einer gegebenen sozialen Situation beurteilt werden (z. B. die Möglichkeit, ein Spiel mit einer Gruppe zu spielen). In der Kommunikation mit anderen Nutzerinnen und Nutzern sowie Bewerterinnen und Bewertern können Modellierungsentscheidungen begründet und Bewertungen argumentativ fundiert werden. In Lernszenarien mit Informatiksystemen sollten pädagogische Fach- und Lehrkräfte in der Lage sein, Erkundungs- oder Gestaltungsprozesse mit Informatiksystemen oder Handlungsstrategien bei Unplugged-Konzepten mit den Kindern kooperativ zu planen und sie zu bewerten.

#### *Strukturieren und Vernetzen (P3)*

Strukturieren und Vernetzen können als Prozesse in unterschiedlichen Inhaltsbereichen der Informatik erforderlich sein. Wie bereits in P2 geschildert, erfordert das Modellieren eines Informatiksystems ggf. eine strukturierte Zerlegung in einzelne Komponenten. Auch bei der Auswahl geeigneter Datenstrukturen ist eine sinnvolle Strukturierung erforderlich. Eine Vernetzung von Systemkomponenten kann bei Informatiksystemen auf vielfältigste Art innerhalb eines Systems in Erscheinung treten und auch die Außenkommunikation mit anderen Systemen beinhalten, insbesondere bei Informatiksystemen die miteinander über lokale Netze oder das Internet verbunden sind. Anhand kleiner Roboter mit Aktoren und Sensoren, die auf ‚Impulse‘ aus ihrer technischen Umwelt reagieren, kann dieses

Konzept veranschaulicht werden. Pädagogische Fach- und Lehrkräfte sollten in informatikbezogenen Lernszenarien, seien sie ‚plugged‘ oder ‚unplugged‘, die Strukturierung eines Problem-bereichs als Voraussetzung zu einer Lösung verdeutlichen, etwa wenn ein Roboter verschiedene Bewegungsrichtungen kennen muss (links, rechts, geradeaus), um sich in einem Labyrinth zu bewegen. Die Vernetzung eines Informatiksystems kann etwa beim spielerischen Übermitteln einer ‚E-Mail‘ verdeutlicht werden.



#### *Kommunizieren und Kooperieren (P4)*

Informatische Problemlösungen in den verschiedenen Inhaltsbereichen erfolgen oft nicht individuell, sondern kooperativ in einem Team. Hierzu werden von den Mitgliedern des Teams auf einer formalen Ebene allgemeine kooperative und kommunikative Fähigkeiten verlangt bzw. eingeübt. Inhaltlich wird die Fähigkeit erwartet, fachgerecht unter Nutzung informatischer Fachbegriffe über informatische Sachinhalte zu kommunizieren, um kooperativ zur Lösung informatischer Probleme beizutragen. Diese Kommunikation und Kooperation kann ‚unplugged‘ mittels diverser, nicht digitaler Materialien als Hilfsmittel und im Medium zwischenmenschlicher Kommunikation erfolgen. Andererseits kann die Kommunikation und Kooperation auch mit Hilfe eines Informatiksystems organisiert werden. Hierzu können synchrone (z. B. Chat, Instant Messaging ...) und asynchrone Kommunikationsmöglichkeiten (z. B. E-Mail) von Informatiksystemen zum Austausch von Information und zur Organisation kooperativer Arbeit genutzt werden. Elektronische (webbasierte) Plattformen können darüber hinaus dem Austausch und der Archivierung von Dokumenten dienen. Pädagogische Fach- und Lehrkräfte sollten diese informatikbezogenen Kommunikations- und Kooperationsformen kennen, anwenden und in Lernszenarien mit Kindern zielgruppenbezogen einsetzen können. Sie sollten dabei auf eine korrekte Verwendung von Fachbegriffen und der Fachsprache achten, um beim Aufbau mentaler Modelle von Informatiksysteme die Begriffsvorstellungen der Kinder korrekt zu präfigurieren.

### *Darstellen und Interpretieren (P5)*

Die in den verschiedenen informatischen Inhaltsbereichen individuell oder kooperativ gewonnenen Erkenntnisse müssen zum Zwecke der individuellen Rezeption und Interpretation oder, um sie mit anderen zu diskutieren, entsprechend kommuniziert werden. Hierbei bieten sich unterschiedliche symbolische, grafische oder bildliche Darstellungsformen an. Die Auswahl geeigneter Darstellungsformen zur Veranschaulichung und Interpretation von informatischen Sachverhalten ist somit ein wichtiger Prozessbereich zur Beschreibung informatischer Fachkompetenzen. Pädagogische Fach- und Lehrkräfte sollten einerseits in der Lage sein, altersgerechte Darstellungsformen zu wählen, die sie auch in Lernszenarien mit Kindern nutzen können. Andererseits sollten Darstellungsformen bekannt sein und teilweise genutzt werden, mit denen etwas komplexere informatische Sachverhalte visualisiert werden können (z. B. Wissensnetze, Ablaufdiagramme, Struktogramme, Graphen).

### **4.3.3 Kontextualisierte Kompetenzerwartungen**

Im folgenden Abschnitt sollen die als besonders relevant erachteten Kombinationen aus einzelnen Inhalts- und Prozessbereichen für kontextualisierte, fachliche Kompetenzerwartungen an die pädagogischen Fach- und Lehrkräfte in knapper Form dargestellt werden.

Im Prinzip gilt hier die gleiche Argumentation wie auch schon auf der Ebene der Zieldimensionen für Kinder (vgl. Kapitel 3.4). Jeder der Inhaltsbereiche lässt sich prinzipiell mit jedem der Prozessbereiche kombinieren. Somit könnten Kompetenzerwartungen formuliert werden, die sich auf jede dieser Kombinationen beziehen. Für jede dieser Kombinationen lassen sich auch geeignete Beispiele für Lernszenarien finden, deren Darstellung aber den Rahmen dieser Expertise sprengen würde. Darüber hinaus erscheint es kaum machbar für unsere Zielgruppe ein Curriculum zu entwickeln, das in realistischer Zeit alle diese Kombinationen abdeckt. Dies ist aber auch nicht erforderlich. Viele der oben im Text genannten Beispiele betreffen oft mehrere Inhalts- und Prozessbereiche, so dass man nicht für jede Kombination einen neuen Zugang (Lernszenario) benötigt.

Bei einem Curriculum für pädagogische Fach- und Lehrkräfte sollte man lediglich darauf achten, dass möglichst viele Inhalts- und Prozessbereiche zumindest einmal in einem der Lernszenarien adressiert werden. Für die pädagogischen Fach- und Lehrkräfte gelten bei einer Auswahl und Priorisierung der fachlichen Kompetenzerwartungen ähnliche Kriterien. Insofern kann hier insbesondere auf die Argumentation in Kapitel 3.4 verwiesen werden. Die pädagogischen Fach- und Lehrkräfte sollten bei den ausgewählten Kombinationen von Inhalts- und Prozessbereichen (I/P) allerdings grundlegende fachliche Kompetenzen erwerben, während dies bei den Kindern, je nach Altersgruppe, sicher zunächst nur erste

fachbezogene Kompetenzfacetten sein werden, die erworben werden können. Als auszuwählende Bereichskombinationen werden daher für die pädagogischen Fach- und Lehrkräfte vorgeschlagen:

- Modellieren und Implementieren (P1) von Algorithmen und Programmen (I2)
- Interaktion mit und Exploration von (P0) Informatiksystemen (I4)
- Repräsentieren und Interpretieren (P5) von Information und Daten (I1)
- Nachdenken über und Bewerten (P2) des Zusammenhangs von Informatik, Mensch und Gesellschaft (I5)

Die Auswahl erfolgt anhand der schon in Kapitel 3.4 und Kapitel 5 angewandten Leitkriterien zur Auswahl und Priorisierung von zu vermittelnden Kompetenzen und Beispielen:

- nach fachlicher und fachdidaktischer Bedeutung
- Möglichkeit der Realisierung unter Lern- und entwicklungspsychologischen Aspekten
- Bezug zu Alltagssituationen der Kinder
- Fähigkeit, die Lernenden zu motivieren
- fachliches Interesse der Kinder an Informatik zu wecken und einen Beitrag zu deren Allgemeinbildung zu leisten
- Möglichkeit, auch übergreifende Basiskompetenzen zu vermitteln
- Bezug zu didaktischen Konzepten der Grundschule bzw. des Kindergartens
- Möglichkeit, sich an der Praxis bereits realisierter Konzepte zu orientieren

Da jede der oben genannten vier Kombinationen anhand von konkreten Praxisbeispielen kontextualisiert werden muss, sind bei einem Beispiel zumeist mehrere Kombinationen aus Inhalts- und Prozessbereichen betroffen, wobei eine oder zwei der Kombinationen einen Schwerpunkt bilden, andere Kombinationen bei dem gegebenen Beispiel nicht so stark thematisiert werden. In Kapitel 5 sind die

zu der Priorisierung gewählten Beispiele deshalb mittels einer ‚Heatmap‘ dargestellt, wobei die Farben indizieren, wie stark bei dem gegebenen Beispiel eine I/P-Kombination und damit zusammenhängende Kompetenzen betroffen sind.

Für die pädagogischen Fach- und Lehrkräfte ergeben sich aus der oben genannten Priorisierung und der Priorisierung für die Zielgruppe der Kinder vor allem Kompetenzerwartungen in den grün markierten Kompetenzfeldern (I/P) (siehe Abbildung 39).

Inhaltsbereiche	(I1) Information & Daten	(I2) Algorithmen & Program- mierung	(I3) Sprachen & Automation	(I4) Informatik- systeme	(I5) Informatik, Mensch & Gesellschaft
Prozessbereiche					
(P0) Interagieren & Explorieren					
(P1) Modellieren & Implementieren					
(P2) Begründen & Bewerten					
(P3) Strukturieren & Vernetzen					
(P4) Kommunizieren & Kooperieren					
(P5) Darstellen & Interpretieren					

**Abbildung 39.** Kompetenzerwartungen an pädagogische Fach- und Lehrkräfte

Die in Kapitel 5 dargestellten Beispiele verdeutlichen, wie das Konzept praktisch in zielgruppenadäquate Lernszenarien umgesetzt werden könnte. Für die Gruppe der pädagogischen Fach- und Lehrkräfte können anhand der für die Kinder vorgesehenen Lernszenarien sämtliche in diesem Kapitel angesprochenen Kompetenzbereiche adressiert werden, wenn sie über die fachliche Ebene hinaus auch die lernpsychologischen, fachdidaktischen und fachübergreifenden Kompetenzbereiche beispielhaft in der Lerngruppe thematisieren.

#### 4.4 Informatikdidaktische Kompetenzen

Wie in Kapitel 4 eingangs bereits beschrieben, werden wir uns bei der Darstellung der fachdidaktischen Kompetenzerwartungen in den Zielvereinbarungen für pädagogische Fach- und Lehrkräfte an dem PCK-Konzept von Shulman und den entsprechenden Umsetzungen in der Mathematikdidaktik orientieren. Von zentraler Bedeutung in diesem Abschnitt sind aber die in jüngster Zeit in der fach-



didaktischen Forschung der Informatik entwickelten Konzepte und Modelle. Wir werden sie an dieser Stelle etwas ausführlicher darstellen und in den folgenden Unterabschnitten des Kapitels in eher knapper Form im Hinblick auf die jeweiligen Kategorien und den spezifischen Kontext der Bildung im Elementar- und Primarbereich spezifizieren.

Mit den informatikdidaktischen Kompetenzen werden jene Fähigkeiten von Lehrkräften beschrieben, ihre fachwissenschaftlichen Kompetenzen in eine für die Lernenden ihrer Zielgruppe effektive Gestaltung von Unterricht und von Lernszenarien umzusetzen. In der Lehrerausbildung werden daher fachwissenschaftliche, fachdidaktische und praxisorientierte Ausbildungsphasen eng miteinander verzahnt. Dabei gehen fachwissenschaftliche Phasen den fachdidaktischen Phasen zumeist voraus, da die fachdidaktische Aufbereitung von informatischen Inhalten entsprechende Fachkenntnisse voraussetzt. Allerdings ist das nicht als linearer und einseitiger Prozess zu verstehen. Auch das situative Lernen fachlicher Inhalte in fachbezogenen Lernszenarien in der Bildung im Elementar- und Primarbereich kann für pädagogische Fach- und Lehrkräfte zum Erwerb von fachlicher und fachdidaktischer Kompetenz beitragen. Dies gilt es bezüglich der Organisation von Lernszenarien für diese Zielgruppe im Auge zu behalten.

Shinners-Kennedy und Fincher (2013) weisen in diesem Zusammenhang darauf hin, dass die fachliche Reflexion eines wichtigen informatischen Gegenstandsbereichs durch Lehrkräfte, deren subjektive Sichtweisen auf diesen und mit ihm zusammenhängende Gegenstandsbereiche verändern kann. Sie machen das an den ‚Big Ideas‘ des Faches Informatik deutlich und verwenden dabei die Idee der ‚Threshold Concepts‘ von Meyer und Land (2005):

„It represents a transformed way of understanding, or interpreting, or viewing something without which the learner cannot progress. As a consequence of comprehending a threshold concept there may thus be a transformed internal view of subject matter, subject landscape, or even world view“ (Meyer & Land 2005, S. 1).

Schwill hat mit seinem Konzept der fundamentalen Ideen der Informatik Kriterien zur Identifikation derartiger ‚Big Ideas‘ geliefert, die gleichzeitig als Kategorien zur Auswahl geeigneter Lerninhalte dienen können: Horizontalkriterium (Fachinhalt ist auf unterschiedlichen kognitiven Niveaustufen erschließbar), Vertikalkriterium (Fachinhalt ist in vielen Teildisziplinen des Faches Informatik relevant), Zeitkriterium (Fachinhalt ist über einen längeren Zeitraum von Bedeutung für das Fach Informatik), Sinnkriterium (Fachinhalt lässt sich über die Erfahrung der Zielgruppe in ihrer Alltagswelt erschließen) und Zielkriterium (Fachinhalt erschließt einen Bezug zu aktuellen Forschungsfragen des Faches Informatik) (Schwill, 1993). Für

Lernszenarien mit Kindern in der Grundschule und im Kindergarten, aber auch für die pädagogischen Fach- und Lehrkräfte sollten Inhalte vor allem im Hinblick auf das Sinnkriterium mit seinem Alltagsbezug ausgewählt werden. Ferner sollte bei dieser Zielgruppe neben der fachlichen Bedeutung des Themas vor allem auch sein Beitrag zur Allgemeinbildung berücksichtigt werden. Zur Beurteilung dieses Aspekts können Kriterien von Klafki (1993) sowie Bussmann und Heymann (1987) herangezogen werden. Klafki schlägt u. a. die folgenden Kriterien vor: (sinnvolle) Bildung für alle, alle menschlichen Fähigkeitsdimensionen betreffend, ein epochaltypisches Schlüsselproblem betreffend, problemorientiert, sachbezogenes und soziales Lernen verbindend. Bussmann und Heymann benennen als Kriterien für die Allgemeinbildung: Lebensvorbereitung, Stiftung kultureller Kohärenz, Weltorientierung, Anleitung zum kritischen Vernunftgebrauch, Entfaltung von Verantwortungsbereitschaft, Einübung in Verständigung und Kooperation sowie Stärkung des Schüler-Ichs.

Buchholz, Saeli und Schulte (2013) schlagen im Sinne von Shinners-Kennedy und Fincher (2013) sowie Saeli (2012) jeweils ein für einen bestimmten Inhalt spezifiziertes PCK-Modell für den Kompetenzerwerb von Informatiklehrkräften vor. Während Saeli am Beispiel der Programmierung zeigt, wie Lehrkräfte pädagogische Fachkompetenzen erwerben können, gehen Buchholz, Saeli und Schulte allgemeiner vor und verwenden das empirische Instrument CoRe (Content Representation), um das PCK einer fundamentalen Idee zu identifizieren. Mit diesem Instrument sollen mittels geeigneter Fragestellungen sowohl die Begründung für die Auswahl des entsprechenden Inhalts als Lerngegenstand erfolgen als auch die Art und Weise beschrieben werden, wie es in einem Lernszenario vermittelt werden kann.

„CoRe involves the following series of questions:

1. What do you intend the students to learn about this Big Idea?
2. Why is it important for the students to know this Big Idea?
3. What else do you know about this Big Idea (and you don't intend students to know yet)?
4. What are the difficulties/limitations connected with the teaching of this Big Idea?
5. Which knowledge about students' thinking influences your teaching of this Big Idea?
6. Which factors influence your teaching of this Big Idea?
7. What are your teaching methods (any particular reasons for using these to engage with this Big Idea)?

8. What are your specific ways of assessing students' understanding or confusion around this Big Idea?“ (Buchholz et al., 2013, S. 10)

Die auf ein spezifisches Informatikthema bezogenen Fragen tangieren zugleich wichtige pädagogische Handlungsfelder unterrichtlicher Praxis, die es im Sinne eines PCK-bezogenen Kompetenzerwerbs zu reflektieren gilt. Durch Theorie-Praxis-Verzahnung und dem Sammeln von unterrichtspraktischer Erfahrung kann durch einen zyklischen Prozess reflektierter Praxis auf der Basis von fachwissenschaftlicher und fachdidaktischer Theorie von Informatik-Lehramtsstudierenden ein vertieftes PCK erworben werden, das über rein oberflächlich formale Festlegungen zur medialen und methodischen Gestaltung von Lernszenarien hinausgeht. Basierend auf diesen theoretischen Überlegungen und mittels in einer Lehrveranstaltung der Informatiklehrerausbildung gewonnener empirischer Daten schlagen die Autoren ein zweidimensionales PCK-Entwicklungsmodell vor.

Die eine Dimension bezieht sich auf die didaktisch-methodische Entscheidungsebene, die mit den oben zitierten Fragestellungen erschlossen werden kann, und ist nach drei Teilbereichen gruppiert:

- Lehren (was? warum? inhaltliche Tiefe?)
- Lernen (Vorwissen der Lerner, Auswertungsmethoden)
- andere Faktoren (Lehrmaterial, institutionelle Bedingungen, inhaltsbezogenes methodisches Arrangement)

Die zweite Dimension stellt eine Gradierung der Fähigkeiten in den verschiedenen Bereichen der ersten Dimension auf 3 Stufen dar, wobei auf Stufe 3 jeweils ein hohes Reflexionsniveau unter Berücksichtigung von fachwissenschaftlichen und fachdidaktischen Konzepten und Praxisbezügen erwartet wird (Buchholz et al., 2013, S. 15; siehe Abbildung 40).

Level	Teaching nexus			Learning nexus			Other	
	Q1 (what)	Q2 (why)	Q3 (SMK, reduc.)	Q4 (difficulties)	Q5 (prior knowledge)	Q8 (assessment)	Q6 (forces)	Q7 (methods)
1	topic is named	goals is named but not justified	(nearly) no knowledge about the topic	teacher centric: tries to cope with the topic herself	mentions part of the content	teacher just knows (observes)	focus on organizational issues and teaching material; vague impression that pupils perspective is important	few methods
2	topic and CS connected	NOT CLEAR [not: names more than one goal!]	deeper content knowledge, but no justification what to leave out	teacher centric: how to reduce/reconstruct	NOT CLEAR	teacher inquires	NOT CLEAR	more methods, and justification for Content
3	connected to every day life	goal is justified; connection between CS and real world is made	focus on specific parts of the deep content knowledge, no justification what to leave out	learner centered; knows specific, content related learning obstacles	NOT CLEAR	teacher has methods/ aspects of inquiring	NOT CLEAR	broad knowledge of methods; use of methods is partially justified

**Abbildung 40.** Vorläufiges Kompetenzmodell (Buchholz et al., 2013, S. 15)

Einen weiteren Bezugspunkt für die Darstellung informatikdidaktischer Kompetenzen von Lehrkräften liefert das im BMBF-Projekt KUI ‚Kompetenzen für das Unterrichten in Informatik‘ entwickelte Kompetenzmodell (Hubwieser et al., 2013). Ausgehend von einem aus dem ‚Darmstädter Modell‘ (Hubwieser et al., 2011) hergeleiteten Kategoriensystem, das als informatikbezogene Weiterentwicklung der lerntheoretischen Modelle zur Unterrichtsplanung entstand (‚Berliner Modell‘ (Heimann, Otto & Schulz, 1979); Hamburger Modell (Schulz, 1997)), wurde zunächst ein Kompetenzstrukturmodell normativ-deduktiv hergeleitet. Dies geschah anhand der qualitativen Inhaltsanalyse (Mayring, 2010) von Ausbildungscurricula von Informatik-Lehramtsstudiengängen vor allem an deutschen, aber auch ausländischen Universitäten und von einschlägiger fachdidaktischer Literatur. Das so entwickelte Kompetenzstrukturmodell wurde im Anschluss mit Hilfe von Experteninterviews nach der ‚Critical Incident Methode‘ (vgl. Hettlage & Steinlin, 2006) empirisch ausdifferenziert und überprüft. In einem weiteren Schritt wurde ein Messinstrument entwickelt und eingesetzt, das mit Daten aus unterschiedlichen Zielgruppen (erfahrene Informatiklehrerinnen und -lehrer, Studienreferendarinnen und -referendare, Informatik-Lehramtsstudierende) einerseits Hinweise auf die entsprechende Expertise der befragten Gruppen lieferte, andererseits aber auch Gelegenheit bot, die Validität des zugrunde liegenden Kompetenzmodells zu überprüfen (Hubwieser et al., 2013). Zwar sind Kompetenzmodell und zugehöriges Messinstrument auch hier wiederum primär für Informatiklehrkräfte im

schulischen Sekundarbereich entwickelt worden, die im Kompetenzstrukturmodell enthaltenen Kategorien, können aber auch ,entsprechend kontextuell adaptiert, wichtige Hinweise für Kompetenzerwartungen der pädagogischen Fach- und Lehrkräfte im (schulischen und außerschulischen) Elementar- und Primarbereich liefern. Als Ergebnis der normativ-deduktiven Analysen ergab sich ein zweidimensionales Kompetenzstrukturmodell, das später aus Gründen der Praktikabilität bei der empirischen Verfeinerung noch etwas reduziert wurde (vgl. Abbildung 41).

Die ursprünglichen Kategorien des Modells sind:

- *Dimension 1: Pädagogische Handlungsfelder (FPO: Fields of Pedagogical Operation)* mit den Subkategorien
  - FPO 1 (Vorbereitung und Planung des Unterrichts)
  - FPO 2 (Pädagogisches Handeln im Informatikunterricht) und
  - FPO 3 (Evaluation und Bewertung des Unterrichts).
  
- *Dimension 2: Aspekte des Lehrens und Lernens (ATL: Aspects of Teaching and Learning)* mit 15 Subkategorien, die wiederum in 5 Gruppen unterteilt sind:
  - Gruppe 1: inhalts- und curriculumbezogene Aspekte
  - Gruppe 2: Lehrmethoden und Mediengebrauch
  - Gruppe 3: lernerbezogene Aspekte
  - Gruppe 4: lehrerbezogene Aspekte
  - Gruppe 5: Aspekte des Bildungswesens.

	FPO 1	FPO 2	FPO 3
ATL 1			
ATL 2			
...	...	...	...
ATL 15			

Cat. Nr.	Field descriptor	Subcategories
FPO 1	Planning and design of learning situations	<ul style="list-style-type: none"> <li>- Time planning (Time allocation),</li> <li>- Explanation of the planning: subject specific consistency, reasonability of the approach, psychological argumentation</li> <li>- Granularity: long term lesson planning, planning the entire curriculum, planning a lesson</li> </ul>
FPO 2	Reacting on student's demands during teaching processes	<ul style="list-style-type: none"> <li>- Reacting based on understanding: flexible use of connected knowledge in critical situations, responding to students appropriately, responding flexibly</li> <li>- Mastering complexity</li> <li>- Keeping compliant with planning</li> </ul>
FPO 3	Evaluation of teaching processes	<ul style="list-style-type: none"> <li>- Techniques,</li> <li>- Criteria</li> <li>- Derive consequences</li> </ul>

Cat. Nr.	Category	Subcategories
<b>Group 1: Subject and Curriculum related Issues</b>		
ATL 1	Learning content	<ul style="list-style-type: none"> <li>- Multiple representations</li> <li>- Category systems for learning content</li> <li>- Specific school-related content</li> <li>- Selection and justification of learning content</li> <li>- Didactical (re-) construction of subject-matter knowledge</li> </ul>
ATL 2	Subject	<ul style="list-style-type: none"> <li>- Relations to other subjects</li> <li>- Definition of computer science education</li> <li>- History of computer science education</li> <li>- Relationship of the subject to the scientific discipline</li> <li>- Objectives of the subject</li> <li>- Legitimacy and relevance of the subject</li> </ul>

ATL 3	Curricula and standards	<ul style="list-style-type: none"> <li>- Curriculum development</li> <li>- Relation to other subjects</li> <li>- Approach and structure of the curriculum</li> <li>- Selection and commitment</li> <li>- Actual examples of curricula</li> </ul>
ATL 4	Objectives of lessons	<ul style="list-style-type: none"> <li>- Focus on education standards</li> <li>- Competencies</li> <li>- Learning objectives</li> </ul>
ATL 5	Extracurricular activities	<ul style="list-style-type: none"> <li>- External collaboration</li> <li>- Contests</li> </ul>
ATL 6	Science	<ul style="list-style-type: none"> <li>- Subject discipline</li> <li>- Computer science education as a scientific discipline</li> <li>- Relationship between teaching of the subject and the scientific discipline</li> </ul>

#### Group 2: Teaching methods and use of Media

ATL 7	Teaching Methods	<ul style="list-style-type: none"> <li>- Organizational arrangements</li> <li>- Methodological principles</li> <li>- Subject-specific teaching methods</li> </ul>
ATL 8	Subject-specific teaching concepts	<ul style="list-style-type: none"> <li>- Introductory lessons</li> <li>- Programming classes</li> <li>- Historical approach</li> </ul>
ATL 9	Specific teaching elements	<ul style="list-style-type: none"> <li>- Lab-based teaching</li> <li>- Experiments</li> <li>- Tasks and assignments</li> </ul>
ATL 10	Media and educational material	<ul style="list-style-type: none"> <li>- Application of hardware and software</li> <li>- Textbooks</li> <li>- Unplugged media</li> </ul>

#### Group 3: Learner related Issues

ATL 11	Heterogeneity in the context of subject-specific learning	<ul style="list-style-type: none"> <li>- Age</li> <li>- Gender</li> <li>- Ethnical background</li> <li>- Family socialization</li> <li>- Disabilities</li> </ul>
ATL 12	Student cognition	<ul style="list-style-type: none"> <li>- General subject-related cognitive aspects</li> <li>- Individual learning Diagnostics, performance evaluation and assessment</li> <li>- Cognitive activation</li> </ul>

Group 4: Teacher related Issues		
ATL 13	Teachers' perspective	<ul style="list-style-type: none"> <li>- Collaboration</li> <li>- Core tasks</li> <li>- Qualification</li> <li>- Motivation</li> <li>- In-service training</li> <li>- Teaching experience</li> </ul>

Group 5: Issues of the Educational System		
ATL 14	School development	<ul style="list-style-type: none"> <li>- Policies</li> <li>- Quality management</li> <li>- School profile</li> </ul>
ATL 15	Educational system	<ul style="list-style-type: none"> <li>- School type</li> <li>- Enrollment</li> <li>- Organizational aspects of subject</li> </ul>

**Abbildung 41.** Kompetenzstrukturmodell für Informatiklehrkräfte (Hubwieser et al., 2013, S. 3)

Im Detail sieht das Modell für die ATL-Dimension wie folgt aus:

Bei der empirisch fundierten Weiterentwicklung des Kompetenzstrukturmodells zeigte sich, dass, ähnlich wie im oben beschriebenen Modell von Buchholz, Saeli und Schulte, die Fähigkeit der Lehrkräfte fachliches und fachdidaktisches Wissen zu verbinden und in praktisches, schülerbezogenes Unterrichtshandeln umzusetzen, ein wichtiges Unterscheidungskriterium für den Grad der erworbenen informatikdidaktischen Kompetenz ist (Buchholz et al., 2013).

Wir werden im Folgenden die Erkenntnisse aus den vorgestellten Kompetenzmodellen und Publikationen zu den fachdidaktischen Kompetenzen von Informatiklehrkräften im Hinblick auf einige relevante Kompetenzaspekte für die pädagogischen Fach- und Lehrkräfte in knapper Form spezifizieren. Dabei werden wir auch adaptierte Indikatoren aus dem KUI-Messinstrument sowie adaptierte Kompetenzbeschreibungen aus den fachspezifischen KMK-Kompetenzbeschreibungen (Kultusministerkonferenz, 2015) verwenden. Wegen der teilweise wechselseitigen Abhängigkeiten sind die Beschreibungen der Kompetenzkategorien nicht immer vollständig disjunkt. Auf der fachlichen Ebene beziehen wir uns auf die in Kapitel 4.4 beschriebenen Kompetenzerwartungen. Im Hinblick auf die organisatorische Gestaltung von informatikbezogenen Lernszenarien für pädagogische Fach- und Lehrkräfte sollte nach den Erkenntnissen der vorgenannten Studien eine zyklische Gestaltung von Kursen in Betracht gezogen werden, die ...



- den Erwerb von informatischen Kompetenzen eng angelehnt an den Lernszenarien für die Kinder ausrichtet,
- fachliche und fachdidaktische Problemstellungen miteinander verbindet sowie

die praktische Umsetzung von zuvor gemeinsam entwickelten Lernkonzepten und Lernszenarien für Kinder in einer praktischen Erprobungsphase und deren anschließender Reflexion mit einschließt.

#### **4.4.1 Informatikdidaktische Basiskompetenzen**

In diesem Abschnitt werden grundlegende informatikdidaktische Basiskompetenzen beschrieben, über die die pädagogischen Fach- und Lehrkräfte verfügen sollten, damit sie zielgruppenbezogene Lern- und Bildungsprozesse der informatischen Bildung in Grundschule und Kita selbstständig erfolgreich gestalten können (vgl. auch Kultusministerkonferenz, 2015, S. 32). Die pädagogischen Fach- und Lehrkräfte sollten über solides informatikdidaktisches und pädagogisch-psychologisches Wissen zur informatischen Bildung von Kindern im Grund- und Vorschulalter verfügen und in der Lage sein, zielgruppenadäquate Lernszenarien zur informatischen Bildung unter Berücksichtigung der entwicklungs- und lernpsychologischen Fähigkeiten der Kinder zu organisieren (vgl. auch Kapitel 2.1).

Die pädagogischen Fach- und Lehrkräfte ...

- sind in der Lage, Bildungsstandards, Empfehlungen und Curricula zur informatischen Bildung in der Grundschule und der Kita zu interpretieren, zu bewerten und ihre Lernszenarien im Sinne dieser Empfehlungen zu planen und durchzuführen;
- können diese Empfehlungen in geeigneter Weise mit Anforderungen aus Orientierungsplänen im Elementarbereich in Beziehung setzen, um daraus Kriterien zur Gestaltung von Lernszenarien abzuleiten;
- kennen wichtige fachdidaktische Konzepte ‚informatischer Bildung‘ (z. B. fundamentale Ideen der Informatik, CS Unplugged, ‚Informatik im Kontext‘) und deren Bedeutung für die Allgemeinbildung und sind in der Lage, diese bei der Planung ihrer Lernszenarien zu berücksichtigen;
- kennen wichtige informatiktypische Lernformen (z. B. exploratives Lernen, anwendungsorientiertes Lernen) sowie wichtige Phasen der Entwicklung von

kleinen Softwareprojekten und sind in der Lage, diese bei der Planung ihrer Lernszenarien zu berücksichtigen;

- sind in der Lage, aktuelle grundlegende fachwissenschaftliche und fachdidaktische Entwicklungen in der Informatik für den Primarbereich aufeinander zu beziehen, zu reflektieren und in die Planung von Lernprozessen zur informatischen Bildung einzubeziehen;
- sind in der Lage, durch die begriffliche Vernetzung von informatischen Konzepten bei den Kindern erste Grundvorstellungen über das Fach Informatik zu erzeugen;
- kennen wichtige informatische Inhalts- und Prozessbereiche und können diese mit einer zielgruppenbezogenen angemessenen Fach- und Alltagssprache darstellen, um auf diese Weise bei den Kindern zu einer informatischen Begriffs- und Konzeptbildung beizutragen;
- können informatische Sachverhalte in verschiedenen Anwendungsbezügen und Sachzusammenhängen aus der Erfahrungswelt der Kinder identifizieren, zumindest in Ansätzen deren gesellschaftliche Bedeutung erkennen und sie für die Gestaltung von Lernszenarien der informatischen Bildung nutzen;
- kennen Möglichkeiten zur Illustration von grundlegenden informatischen Prinzipien, die die visuelle, auditive und haptische Wahrnehmung der Kinder ansprechen;
- sind befähigt, ihre Lehrerfahrung in der Grundschule bzw. im Elementarbereich auch für die informatische Bildung zu nutzen und eigene Routinen für pädagogisches Handeln in informatischen Lernszenarien zu entwickeln;
- sind in der Lage, sich selbst und die Kinder für Informatikthemen zu motivieren, indem sie authentische Lernsituationen schaffen und ihr Interesse am Fach Informatik mit einbringen.

#### **4.4.2 Kompetenz zur Planung von informatischen Lernumgebungen und Lernsituationen**

Ein wesentliches Element der fachdidaktischen Kompetenz von pädagogischen Fach- und Lehrkräften ist ihre Fähigkeit zur Gestaltung geeigneter Lernszenarien mit Bezug zur informatischen Bildung. Die hierzu erforderlichen Teilkompetenzen werden in diesem Abschnitt beschrieben. Dabei kommt es sowohl auf die gene-

relle pädagogische und fachdidaktische Kompetenz zur Planung informatischer Lernszenarien an als auch darauf, die Planung für konkret gegebene organisatorische und pädagogische Rahmenbedingungen zu kontextualisieren.

Die pädagogischen Fach- und Lehrkräfte ...

- kennen wesentliche Elemente von informatischen Lernumgebungen und nutzen diese Kenntnis zur zielgerichteten Konstruktion von Lernszenarien mittels geeigneter Auswahl von für Kinder authentischen Themen sowie adäquaten Medien und Methoden zu deren Erschließung;
- sind in der Lage, ihre Planung informatischer Lernprozesse mit Kindern hinsichtlich der erforderlichen zeitlichen Aspekte von Lernphasen und unter Berücksichtigung der individuellen Lernprozesse der Kinder einzuschätzen;
- sind in der Lage, ihre Planung informatischer Lernprozesse anhand fachlicher und lernpsychologischer Kriterien, insbesondere im Hinblick auf die kognitive Entwicklung der Kinder, einzuschätzen;
- sind in der Lage, geeignete Lerninhalte für Lernszenarien der informatischen Bildung auszuwählen und die Auswahl fachdidaktisch („Big Idea“) und lernpsychologisch zu begründen;
- sind in der Lage, die von den Kindern in informatischen Lernprozessen zu erwerbenden Kompetenzen individuell je nach deren Leistungsfähigkeit auf der Basis von Bildungsstandards der Informatik zu definieren;
- sind in der Lage, typische informatische Präkonzepte, Verstehens- und Lernbarrieren der Kinder zu identifizieren und erforderliche Formalisierungen und Abstraktionen informatischer Prinzipien entsprechend den kognitiven Fähigkeiten der Kinder umzusetzen;
- sind in der Lage, Aufgabenstellungen und Lerninhalte



in der informatischen Bildung mit Bezug auf die Zielgruppe kindgemäß und auf unterschiedliche Weise vereinfacht, aber korrekt darzustellen sowie abstrakte informatische Konzepte (z. B. Algorithmus, Datum, Information, Variable ...) durch verschiedene Beispiele und mittels spielerischer und explorativer Erkundungsanlässe für die Kinder erfahrbar zu machen;

- sind in der Lage, für die informatische Bildung motivierende informatiktypische Sozialformen wie z. B. Gruppenarbeit an Informatiksystemen, Rollenspiele oder CS-Unplugged-Methoden für die Kinder kompetenzfördernd einzusetzen;
- sind in der Lage, kindgemäße Software wie z. B. Programmiersprachen oder Modellierungswerkzeuge und einfache altersgemäße Informatiksysteme (siehe Kapitel 2) für die informatische Bildung im Elementar- und Primarbereich auszuwählen und sinnvoll in Lernszenarien zu integrieren;
- sind in der Lage, in der frühen informatischen Bildung auf geschlechtsspezifische Unterschiede, insbesondere in der Aufgabenauswahl einzugehen und der Stereotypenbildung entgegenzuwirken.

#### **4.4.3 Kontextuelle informatisch-didaktische Handlungskompetenz**

Mit diesem Kompetenzbereich werden erwartete Fähigkeiten von pädagogischen Fach- und Lehrkräften beschrieben, ihre getroffenen Planungen für informatische Lernszenarien in die Praxis der Grundschulbildung und in Lernprozesse im Kindergarten konkret umzusetzen und diese in geeigneter Weise zu steuern. Dazu gehören die folgenden Kompetenzen:

Die pädagogischen Fach- und Lehrkräfte ...

- sind in der Lage, im Rahmen ihres pädagogischen Handelns in informatischen Lernszenarien Kinder für das Lernen von Informatik zu motivieren sowie individuelle Lernfortschritte zu fördern und zu bewerten;
- kennen fachspezifische Interventionsmöglichkeiten und können diese situativ in informatischen Lernszenarien anwenden (z. B. mit informatischen Fehlvorstellungen, Umgang mit vorläufigen und ungenauen Fachbegriffen, Reaktion auf kindliche Denkkonstruktionen, heuristische Hilfestellungen);

- erkennen in informatischen Bildungsprozessen Situationen mit informatischem Gehalt, greifen diesen ggf. in angemessener Weise auf und initiieren somit weitergehende informatikbezogene Handlungen und Gespräche;
- können auch in Gruppenarbeitsituationen mit Informatiksystemen auf Lernschwierigkeiten einzelner Kinder eingehen und gezielte individuelle Hilfestellungen geben;
- sind in der Lage, Konzepte für das Erlernen einer einfachen Programmiersprache in informatischen Lernprozessen mit einem spielerischen, auch Unplugged-Ansatz, praktisch umzusetzen;
- sind in der Lage, in kreativer und motivierender Weise Methoden zur Aktivierung der Kinder in informatischen Lernprozessen im Elementar- und Primarbereich anzuwenden, um den Lernprozess zu steuern und zu fördern;
- sind in der Lage, unterschiedliche individuelle Leistungsstände der Kinder in informatischen Bildungsprozessen während des praktischen pädagogischen Handelns zu identifizieren und auf diese Heterogenität durch geeignetes situatives Handeln angemessen zu reagieren (z. B. Prozesshilfen, natürlich differenzierende Spiel- und Erkundungsumgebungen und Lernarrangements);
- regen die Kinder in informatischen Lernprozessen zum selbstständigen Lernen und zur eigenständigen Begutachtung ihrer so erzielten informatischen Lösungen an und fördern damit deren Selbstwirksamkeit in informatisch-technischen Kompetenzbereichen;
- fördern durch Gruppengespräche die Reflexion der Kinder über ihr eigenes Handeln mit Informatiksystemen und sorgen für eine kooperative Ermittlung des Lernerfolgs und für ein positives Feedback über die erzielten Ergebnisse an die Kinder.

#### **4.4.4 Kompetenz zur Diagnose und Evaluation von informatischen Lernsituationen**

Neben kontextuellen informatisch-didaktischen Handlungskompetenzen, die die Fähigkeit zu angemessenem situativen Handeln in informatischen Bildungsprozessen beschreiben, benötigen pädagogische Fach- und Lehrkräfte auch die Kompetenz, informatische Lernsituationen mit angemessenen empirischen Methoden zu analysieren, um daraus Schlussfolgerungen für das eigene künftige pädagogische Handeln in der Lerngruppe ziehen zu können. Gegenstandsbereiche

sind dabei Indikatoren, die den Kompetenzerwerb der Kinder auf unterschiedlichen Ebenen (informatisch-fachliche, soziale, motivationale und volitionale Kompetenzstrukturelemente (vgl. Magenheim et al., 2010) sowie relevante Kriterien des informatikbezogenen pädagogischen Handlungskontextes (vgl. Kapitel 4.4) betreffen.

Voraussetzung für diese Kompetenz ist insgesamt eine allgemeine Diagnosefähigkeit der pädagogischen Fach- und Lehrkräfte. Dazu gehört außerdem eine allgemeine lernpsychologische Diagnosefähigkeit bezüglich der Lernfähigkeiten der konkret zu betreuenden Lerngruppe als Voraussetzung für die Planung von Lernszenarien.

Als erwartete Kompetenzen in diesem Bereich sind hier zu nennen:

Die pädagogischen Fach- und Lehrkräfte ...

- kennen die Grundlagen empirischer Kompetenzmessung, der Leistungsdiagnose und Leistungsbeurteilung in der informatischen Bildung, können deren Ergebnisse verstehen und im Hinblick auf die Organisation informatischer Bildungsprozesse in ihrer Bildungseinrichtung interpretieren (z. B. Intelligenz- und informatische Leistungstests);
- können Kinder in Bezug auf informatische Lernprozesse, deren individuelle Denkwege und Vorstellungen sowie hinsichtlich ihrer persönlichen Lernvoraussetzungen, Vorerfahrungen und Fähigkeiten mit informatikbezogenen Diagnosewerkzeugen (wie z. B. Bibertests <http://www.informatik-biber.de>) analysieren;
- sind in der Lage, geeignete Bewertungsschemata für die von ihnen organisierten informatischen Lernsituationen unter Berücksichtigung der verwendeten methodischen und medialen Arrangements zu erstellen und die Leistungen der Kinder in diesen Lernsituationen zu bewerten;
- sind in der Lage, mittels empirisch gewonnener Daten aus der eigenen Lerngruppe die von ihnen organisierten informatischen Lernsituationen zu reflektieren und, darauf aufbauend, bei der Wiederholung einzelner Sequenzen ihre ursprüngliche Planung für das Lernszenario zu überarbeiten;
- sind in der Lage, Kinder für das Lernen von Informatik zu motivieren sowie individuelle Lernfortschritte der Kinder auf der Basis geeigneter Daten zu fördern und zu bewerten;

- sind in der Lage, die kindliche Entwicklung in der informatischen Bildung beobachtend zu begleiten und möglichen Interventionsbedarf im Hinblick auf informatische Lernprozesse in der Bildungseinrichtung und im familiären Umfeld (z. B. im Umgang mit digitalen Medien) zu diagnostizieren.

#### **4.4.5 Pädagogisch-fachliche Kommunikationskompetenz mit Beteiligten**

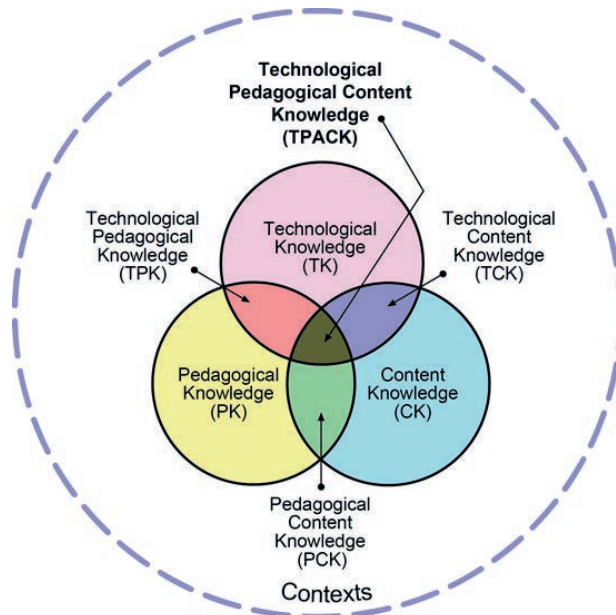
Die pädagogischen Fach- und Lehrkräfte im Elementar- und Primarbereich verrichten ihre Arbeit oft in enger Abstimmung mit ihren Kolleginnen und Kollegen und mit dem familialen Umfeld der Kinder. Auch in diesem pädagogischen Handlungsfeld sind aus der Perspektive informatischer Bildung pädagogisch-fachliche Kommunikationskompetenzen mit allen Beteiligten erforderlich.

Die pädagogischen Fach- und Lehrkräfte ...

- sind in der Lage, den Eltern der Kinder die informatikbezogenen Lernszenarien und angestrebten informatischen Bildungsziele verständlich zu erklären und so auf motivierende Weise das familiale Umfeld der Kinder in den Bildungsprozess mit einzubeziehen;
- sind in der Lage, informatikbezogene Lernprozesse in ihren Bildungseinrichtungen in Kooperation mit ihren Kolleginnen und Kollegen zu organisieren und bei der Gestaltung von Lernszenarien der informatischen Bildung auf fachlicher und fachdidaktischer Ebene angemessen mit diesen zu kommunizieren;
- können zur Organisation und Abstimmung informatikbezogener Lernprozesse in ihren Bildungseinrichtungen neben der direkten zwischenmenschlichen auch Formen der synchronen und asynchronen Kommunikation mittels digitaler Medien nutzen sowie Dokumente über digitale Lernplattformen austauschen;
- können ihr Wissen über grundlegende Funktionsweisen und pädagogische Einsatzmöglichkeiten von digitalen Medien im Elementar- und Primarbereich nutzen, um Kolleginnen und Kollegen und Eltern hinsichtlich der Mediennutzung durch Kinder im pädagogischen Kontext zu beraten.

## 4.5 Schlüsselkompetenzen zum Umgang mit digitalen Medien

Wie oben erläutert, beinhalten die fachlichen und fachdidaktischen Kompetenzen der pädagogischen Fach- und Lehrkräfte auch fundamentales Wissen über die Funktionsweise digitaler Medien – als spezifische Informatiksysteme – und deren Einsatzmöglichkeiten im Unterricht und in informatikbezogenen Lernszenarien (vgl. Schulte & Knobelsdorf, 2011; vgl. auch Kapitel 1). Einerseits ist dieses Wissen im Hinblick auf seine Anwendung in pädagogischen Kontexten fachlich spezifiziert. Andererseits können die in informatischen Lernzusammenhängen erworbenen Kompetenzen bezüglich des Verständnisses und der Nutzung digitaler Medien von Lehrkräften auch in verschiedenen anderen Fachkontexten und z. B. in Beratungsgesprächen mit Eltern genutzt werden. Mishra und Köhler (2006) haben mit ihrem Konzept des TPACK (Technological Pedagogical Content Knowledge) die Zusammenhänge zwischen der fachbezogenen pädagogischen Nutzung digitaler Medien und den fachlichen, fachdidaktischen, technischen sowie pädagogischen Kompetenzkomponenten deutlich herausgearbeitet (siehe Abbildung 42).



**Abbildung 42.** Struktur des TPACK (Mishra & Koehler, 2006; vgl. auch <http://www.matt-koehler.com/tpack/using-the-tpack-image/> [Zugriff am 05.03.2018])



Die von pädagogischen Fach- und Lehrkräften erworbenen, auf Informatik bezogenen fachlichen und fachdidaktischen Kompetenzen (siehe oben: z. B. Programmieren mit Scratch, Nutzung von Tangibles wie Ozobot, MakeyMakey, Mindstorms etc.) stellen nach diesem Modell ein grundlegendes fachspezifisches TPACK dar, das aber auch in andere pädagogische Kontexte transferiert werden kann. Insofern kann diese Art von informatikbezogenem Kompetenzerwerb der pädagogischen Fach- und Lehrkräfte einen Beitrag zu deren digitaler Medienbildung und ihrem kompetenten Umgang mit digitalen Medien in verschiedenen Nutzungskontexten leisten.

Dies ist ein wichtiger Aspekt der Ausbildung. So werden in den ISTE ‚Teacher Standards‘ für alle Lehrkräfte Kompetenzen im Bereich ‚Design and develop digital age learning experiences and assessments‘, ‚Model digital age working and learning‘, und ‚Promote and model digital citizenship‘ gefordert (ISTE – The International Society for Technology in Education, 2008). Magenheim, Schulte und Scheel (2002) haben ein Konzept vorgelegt, das relevante informatische Inhalte als wichtige Elemente einer digitalen Medienbildung für Lehrkräfte aller Fachrichtungen ausweist und so den Zusammenhang zwischen ‚digitaler Medienbildung‘ für Lehrkräfte und deren Kenntnissen über fundamentale informatische Konzepte verdeutlicht.

Die Bedeutung derartiger Kompetenzen zum Verständnis und der Nutzung digitaler Medien in pädagogischen Handlungskontexten ist ausführlich dargestellt worden: hinsichtlich der vielfältigen Einsatzmöglichkeiten digitaler Medien in der Schule z. B. in Albers, Magenheim und Meister (2011), bei Tillman, Fleischer und Hugger (2014) oder Breiter, Welling und Stolpmann (2010) und hinsichtlich der daraus resultierenden Kompetenzerwartungen an Lehrkräfte auf nationaler Ebene etwa im BMBF-Forschungsprojekt M3K zur Medienkompetenz von Lehrkräften (Grafe & Breiter, 2014), in den KMK-Beschlüssen zur Medienbildung (Kultusministerkonferenz, 2012) oder international seitens der UNESCO (2012) mit ihrem ‚ICT-Literacy Concept for Teachers‘.

In diesem Sinne können die hier beschriebenen Zieldimensionen für die pädagogischen Fach- und Lehrkräfte auch einen spezifischen Beitrag zur digitalen Medienbildung für diese Zielgruppe leisten.

Darüber hinaus können den Pädagoginnen und Pädagogen im Sinne des dreidimensionalen DeSeCo Konzepts der OECD auch Schlüsselqualifikationen zum Umgang mit digitalen Medien in der Informationsgesellschaft vermittelt werden: Fähigkeit zum autonomen Handeln mit digitalen Medien in verschiedenen Handlungskontexten, Fähigkeit zum Interagieren in heterogenen Gruppen, kontextspezifische Auswahl geeigneter digitaler Tools (OECD, 2005).

## 4.6 Fazit/Empfehlungen

Die Ausführungen in Kapitel 4 haben gezeigt, dass eine adäquate informatikbezogene Aus- und Fortbildung der pädagogischen Fach- und Lehrkräfte eine wesentliche Gelingensbedingung für die erfolgreiche Umsetzung informatischer Bildung im Kita- und Grundschulbereich darstellt. Die Aus- und Fortbildung der pädagogischen Fach- und Lehrkräfte sollte praxisbezogen in Orientierung an den für die Kinder zu organisierenden Lernszenarien erfolgen (vgl. Beispiele und Priorisierungsvorschläge in Kapitel 5.1). Hierbei sollten die pädagogischen Fach- und Lehrkräfte in den dabei adressierten Kompetenzfeldern grundlegende fachliche und fachdidaktische Basiskompetenzen erwerben, die ihnen eine souveräne pädagogische Handlungskompetenz bei der Organisation und Evaluation der informatischen Lernprozesse mit den Kindern ermöglicht. Ferner sollten in der praxisbezogenen (praxisintegrierten) Ausbildung der pädagogischen Fach- und Lehrkräfte deren Motivation zum Gestalten informatischer Bildungsprozesse und deren grundsätzlich positive Einstellungen gegenüber Informatik- und Informatiksystemen gefördert werden. Dies schließt den kompetenten und kritisch-reflektierten Umgang mit digitalen Medien, insbesondere in der Funktion als Lernmedien der informatischen Bildung, mit ein. Auf dieser Basis können die pädagogischen Fach- und Lehrkräfte dann auch die erforderliche fachliche Kommunikationskompetenz mit verschiedenen Gruppen von an Bildungsprozessen Beteiligten innerhalb ihrer Einrichtung erlangen, um frühe informatische Bildung im Kita- und Grundschulbereich erfolgreich zu implementieren.

## 5 Beispiele priorisierter Kompetenzbereiche der frühen informatischen Bildung

Wie bereits in Kapitel 3.4 und 4.3 dargestellt, können Kompetenzen nur in spezifischen kontextualisierten Lernsituationen erworben werden, in denen bereichsspezifische Handlungsprozesse in ausgewählten Inhaltsbereichen verwirklicht werden. Wie eine erfolgreiche Umsetzung derartiger Lernszenarien der informatischen Bildung realisiert werden kann, soll in diesem Kapitel an einigen ausgewählten Beispielen gezeigt werden. Die Auswahl der Kompetenzbereiche als Kombination von Prozess- und Inhaltsbereichen wurde in den Kapiteln 3.4 und 4.3 begründet. Sie erfolgt anhand der dort genannten 10 Kriterien, die sich u. a. an fachlichen, fachdidaktischen, lernpsychologischen und allgemeinbildenden Aspekten orientieren. Die Umsetzungsbeispiele wurden so gewählt, dass sie die bereits priorisierten Kompetenzfelder (I/P) abdecken:

- Modellieren und Implementieren (P1) von Algorithmen und Programmen (I2)
- Interaktion mit und Exploration von (P0) Informatiksystemen (I4)
- Repräsentieren und Interpretieren (P5) von Daten und Information (I1)

Nachdenken über und Bewerten (P2) des Zusammenhangs von Informatik, Mensch und Gesellschaft (I5)

Wie bereits oben dargestellt, haben die Beispiele zwar einen primären Bezug zu einem oder zwei Kompetenzfeldern, berühren meist jedoch mehrere. Zur Illustration dieses Sachverhalts wird zu jedem Beispiel eine ‚Heatmap‘ von adressierten Kompetenzen erstellt. Dabei symbolisieren dunkle Farben eine starke Repräsentanz der betreffenden Kompetenz in dem jeweiligen Beispiel. Umgekehrt indizieren hellere Farben, dass die betreffende Kompetenz wenig oder eher nur marginal in dem Beispiel berührt wird.

Um die empfohlene Priorisierung von Kompetenzen insgesamt zu verdeutlichen, wird am Ende von Kapitel 5 eine zusammenfassende ‚Heatmap‘ vorgestellt (vgl. Kapitel 5.2). Sie veranschaulicht kontextunabhängig, welche Kompetenzen (Kombinationen aus Prozess- und Inhaltsbereichen) insgesamt für die pädagogische Umsetzung empfohlen werden. Dabei fließt aus jedem Beispiel nur die primär ausgewählte Kompetenz in die zusammenfassende Heatmap ein. Der Inhaltsbereich ‚Informatik, Mensch & Gesellschaft‘, den wir für sehr wichtig halten und dem in vielen nationalen und internationalen Curricula für die Sekundarstufe I und II eine hohe Bedeutung beigemessen wird, ist für die Zielgruppen dieser Empfeh-

lung zwar kein priorisierter Bereich. Er sollte dennoch in einzelnen Beispielen, wie unten gezeigt, kindgerecht in die Lernszenarien eingebunden werden.

Da es aus fachdidaktischer Perspektive keine inhaltlichen Abhängigkeiten von zu erwerbenden Kompetenzen bei den einzelnen Beispielen gibt und deren Komplexität von den pädagogischen Fach- und Lehrkräften in adaptierten Lernszenarien je nach kognitiven Fähigkeiten der Lerngruppe sowie deren Bedürfnissen und Interessen gestaltet werden kann, fällt es dem Autorenteam schwer, eine Zuordnung der Beispiele zu Altersstufen wie Kita- oder Grundschule vorzunehmen. Hier müsste zunächst durch empirisch fundierte Begleitforschung eine Auswertung der praktischen Umsetzung der vorgeschlagenen Beispiele erfolgen, um auf der Basis so gewonnener Erfahrungen dann eventuell Stufungsempfehlungen aussprechen zu können.

## 5.1 Beispiele früher informatischer Bildung

### 5.1.1 Interagieren mit und Explorieren von Informatiksystemen

**Priorisierung: I4 (Informatiksysteme) und P1 (Explorieren und Interagieren)**

Außerdem adressiert: **I2+P0** und **I2, I4+P1**

**Beispiel: Exploration eines Informatiksystems mit LEGO WeDo**

#### **Kurzfassung**

Kinder bauen mit LEGO WeDo und einer Anleitung ein Informatiksystem, dieses kann ein Roboter oder gesteuertes Tier sein. Alternativ können sie auch ein bereits zusammengebautes System nutzen.

Dieses System besteht aus Motoren, Sensoren und Aktoren die in Kombination mit einer zugehörigen Software programmiert und somit gesteuert werden können.

Die Kinder können die einzelnen Komponenten des Systems sowie ihr Zusammenspiel auch in Kombination mit der Programmierung explorieren. Dadurch lernen sie die Komponenten des Systems und ihre Interaktion kennen. In weiteren Schritten kann das System programmiert und Kompetenzen im Bereich Algorithmen und Programmierung können aufgebaut werden. Hier kann ebenfalls explorativ vorgegangen werden.

#### **Ziele**

Durch die Exploration des Informatiksystems lernen die Kinder die verschiedenen Bestandteile des Systems und ihre Funktionalitäten kennen. Das System besteht beispielsweise aus Sensoren und Aktoren. Die Kinder lernen durch die Explora-

tion des Systems, dass beispielsweise die Aktoren immer dann reagieren, wenn die Sensoren ein Signal aufnehmen (z. B. ein Geräusch oder eine Helligkeit). An dieser Stelle kann beispielsweise auf das EVA-Prinzip eingegangen werden.

Darüber hinaus können z. B. durch den Austausch von Zahnrädern Entdeckungen hinsichtlich mechanischer Funktionalität vs. informatischer Funktionalität gemacht werden.

In einem weiteren Schritt kann die Programmierung hinzukommen. Hier können die Kinder erkunden und ausprobieren, wie beispielsweise die Motoren angesteuert werden oder Aktoren generell auf Eingaben eines bestimmten Sensors warten.

### Heatmap

Inhaltsbereiche	(11) Information & Daten	(12) Algorithmen & Program- mierung	(13) Sprachen & Automation	(14) Informatik- systeme	(15) Informatik, Mensch & Gesellschaft
<b>Prozessbereiche</b>					
(P0) Interagieren & Explorieren		Verhalten- muster explorieren, Experimente durchführen		Orientieren und Explo- rieren von Bestand- teilen und Funktiona- lität	
(P1) Modellieren & Implementieren		Systema- tisches Auspro- bieren und Entwickeln von Algorith- men		Erstellen von Modellen zur Funktio- nalität des Systems	
(P2) Begründen & Bewerten					
(P3) Strukturieren & Vernetzen					
(P4) Kommunizieren & Kooperieren				Kommunizie- ren über die entdeckte Funktiona- lität	Diskutieren der Auswir- kungen von Automatisie- rung für die Gesellschaft
(P5) Darstellen & Interpretieren					

### **Beschreibung**

Mit LEGO WeDo kann sehr gut auf explorative Art und Weise gearbeitet werden. Lerntheoretischer Fokus ist hierbei der Konstruktivismus, der durch Papert (1991) begründet wurde.

Kinder untersuchen eine häufig vorkommende reale Problemstellung, z. B. Schussweiten eines Fußballspielers oder Sortieren von Gegenständen. Hierzu können sie zunächst zahlreiche Experimente und Untersuchungen durchführen, die entweder bereits mit dem gebauten System realisiert werden (siehe Beispiel Fußballspieler) oder zur Entwicklung von Ideen zur Konstruktion eines eigenen Systems führen (siehe Beispiel Sortieren). In einem weiteren Schritt können sie das vorhandene System verändern oder ein eigenes neues System konstruieren. So lernen Kinder den Aufbau und die Funktionalität (inklusive einem kleinen Einblick in die Programmierung) des Systems kennen, ähnlich auch der Idee des „Tinkering“.

Mögliche Beispiele:

**Fußballspieler:** Dabei kann z. B. die Schussweite mit einem vorgegebenen Ball ermittelt werden. In weiteren Schritten kann z. B. der Hebel verändert und auch hier die Schussweite ermittelt werden. Ein weiterer Schritt wäre, z. B. einen Sensor einzubauen, der eine bestimmte Zeit auf einen Ball wartet, oder System so zu ergänzen, dass der Spieler automatisch in die Ausgangsposition zurückfährt etc.

**Sortiermaschine:** Die Schülerinnen und Schüler lösen ein reales Problem, untersuchen, wie Maschinen zum Sortieren von Gegenständen funktionieren, und entwickeln eine Sortiermaschine, überprüfen diese auf ihre Funktionalität und experimentieren mit unterschiedlichen Ausgangssituationen.

### **Begründung**

Durch das Explorieren und gezielte Erkunden eines Informatiksystems (sowohl auf Hardware- als auch auf Softwareebene) lernen die Kinder neue Informatiksysteme kennen und können sich für diese begeistern. Sie probieren die Systeme aus und entdecken dabei neue Phänomene. Um das Verhalten besser nachvollziehen zu können, erstellen sie Modelle zur Beschreibung des Verhaltens. In weiteren Schritten können die Kinder Informatiksysteme mitgestalten, in denen sie beispielsweise Anpassungen vornehmen. In diesem Zusammenhang lernen sie auch mit Fehlern umzugehen.

Es ist anzunehmen, dass diese Kompetenzen im Umgang mit neuen Informatiksystemen hilfreich sind. In unserer heutigen Welt sind wir permanent mit neuen

Informatiksystemen konfrontiert. Durch die Bedienung dieser Systeme konstruieren wir mentale Modelle dieser Systeme. Da der Grundaufbau informatischer Systeme immer gleich ist, ist anzunehmen, dass die erworbenen Kompetenzen zu einem selbstsichereren Umgang mit Informatiksystemen beitragen.

### 5.1.2 Programmieren und Algorithmen

**Priorisierung:** I2 (Algorithmen und Programmierung) und P1 (Modellieren und Implementieren)

Optional adressiert: P3 + P4

#### **Beispiel A: Geschichten erzählen (engl. Storytelling)**

##### **Kurzfassung**

Kinder erstellen mit einem geeigneten Programmierwerkzeug eine kleine Geschichte bzw. eine Animation. Dazu gucken sie sich Vorlagen an, ändern diese ab und entwickeln auf dieser Basis eigene kleine Geschichten.

Dabei lernen sie die Programmierumgebung, eine Auswahl grundlegender Programmieranweisungen und den Aufbau der Beispiele kennen und gezielt an eigene Ideen anzupassen (Anpassen fremder Beispiele in der (engl.) Literatur auch als Remix bezeichnet).

##### **Ziele**

Das Programmieren auf diesem Niveau und die Erweiterung der Ausdrucksfähigkeit sowie das Verstehen des Programmieren als: „jetzt definieren, später ausführen [oder] durch andere ausführen lassen“ sind die Ziele.

Die Ausdruckskompetenz beinhaltet die Fähigkeit, eigene Ideen und Gedanken auszudrücken und anderen mitzuteilen.

Algorithmen: algorithmische Grundbausteine: Anweisung, Verzweigung, Schleife, ggf. Interaktion (vgl. z. B. Übersichtsseite Scratch)

## Heatmap

Inhaltsbereiche	(I1) Information & Daten	(I2) Algorithmen & Program- mierung	(I3) Sprachen & Automation	(I4) Informatik- systeme	(I5) Informatik, Mensch & Gesellschaft
Prozessbereiche					
(P0) Interagieren & Explorieren		Adaptieren	Schrittwei- ses Verfei- nern	Explorieren <sup>41</sup>	Kommuni- zieren durch Programme
(P1) Modellieren & Implementieren		Ablauf planen			
(P2) Begründen & Bewerten					
(P3) Strukturieren & Vernetzen					
(P4) Kommunizieren & Kooperieren		Kommuni- zieren durch Programme			Geschichten programmie- ren
(P5) Darstellen & Interpretieren		Story- boarding			

## Beschreibung

Die Idee, den Einstieg in das Programmieren mit dem Erzählen von (animierten) Geschichten zu verknüpfen, ist vermutlich an unterschiedlichen Orten entstanden. Die Idee wurde von Caitlin Kelleher, die nach Möglichkeiten gesucht hat, Mädchen an das Programmieren heranzuführen, zu ihrem Promotionsprojekt (2006) ausgebaut. Sie hat mit Alice (einer 3D-Umgebung) experimentiert und beobachtet, dass Kinder und Jugendliche im Alter von etwa 9 bis 15 Jahren das System gerne nutzten, um kleine Geschichten zu erstellen, und dass sie dabei die wichtigen Grundbausteine des Programmierens einsetzten. Als Vorteile des Ansatzes formuliert sie in ihrer Dissertation:

„Storytelling is a good context for middle school girls to learn about computer programming: 1. Given a little bit of time, most girls can come up with a story they would like to tell. Storytelling is, at its core, a form of communication which is an important activity to most middle school girls. 2. Stories are naturally sequential, allowing users to begin by creating sequences of instructions and gradually progress to more advanced programming concepts as they gain experience and confidence.

<sup>41</sup> Explorieren der Funktionen der Programmierumgebung sowie z. B. beim Finden von Fehlern auch generelles Explorieren der Funktionsweise des Computers (für Spezialistinnen und Spezialisten: Erkunden der ‚notional machine‘).



3. Stories are a form of self-expression and provide girls an opportunity to experiment with different roles, a central activity during adolescence.
4. Non-programming friends can readily understand and appreciate an animated story, which provides an opportunity for girls to get positive feedback from their friends“ (Kelleher, 2006, S. 32).

Im Vergleich zur Arbeit mit einer eher traditionellen Programmierumgebung/Ansatz (Kelleher, 2006, S. 38) lernten die Kinder zwar nicht mehr, verbrachten aber mehr Zeit mit der Tätigkeit des Programmierens und bekundeten ein stärkeres Interesse daran, für sich selbst und/oder in weiteren Kursen weiter mit dem System zu programmieren. Kelleher folgert:

„Storytelling can provide a gentle, motivating introduction to programming concepts. Girls often begin by creating sequences of instructions and, as they gain confidence, create new scenes and new actions for their characters, tasks which often require more complex programming constructs. Girls' storyboards commonly included motivation to use methods, parameters, loops, and parallel execution“ (Kelleher, 2006, S. 39).

Umgesetzt wird die Idee als sog. ‚Storyboarding‘, das dem Kompetenzbereich „Modellieren“ zugeordnet werden kann:

„In the first step, they (die Kinder) wrote a single-paragraph description of the story they were planning to create. In the packet, I encouraged them to think of this paragraph as being similar to the description one might find on the back of the DVD box. In the second step, the worksheet directed girls to break their story into 3–5 separate scenes. For each scene, girls wrote a description of the setting, what happens during the scene (in 1–2 sentences), and the purpose for the scene (what the audience should learn from the scene). Finally, girls created a series of storyboard frames for each of their scenes. The worksheet provided 9 frames per scene and directed girls to both draw the frame and provide a short textual description of the action in that frame beneath it. In practice, most scenes contained 4 to 6 frames and accompanying textual descriptions“ (Kelleher, 2006, S. 75).

Solche Storyboards müssen nicht unbedingt schriftsprachlich umgesetzt werden, sondern können auch aus kleinen Zeichnungen bestehen:

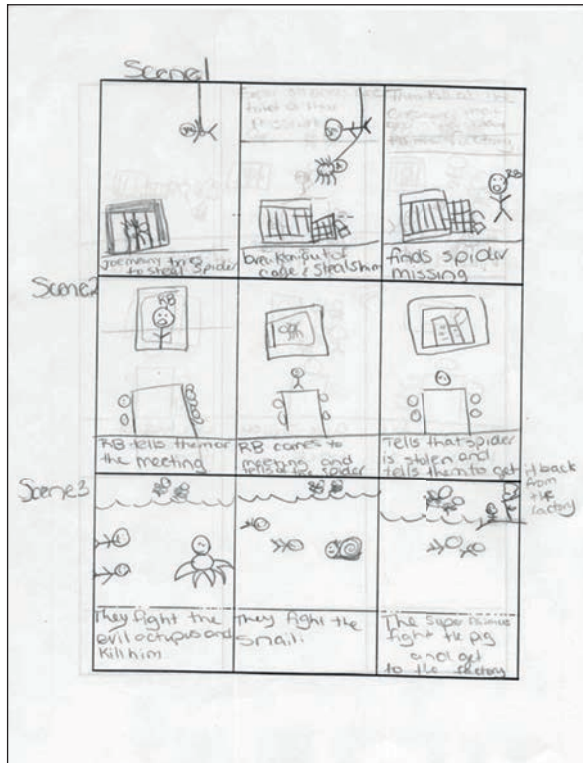


Abbildung 43. Storyboard-Beispiel (Kelleher, 2006, S. 54)

Storyboards können eingesetzt werden, um Abläufe zu **modellieren** und zu lernen, dass die Technik das Anpassen an neue Ideen während des weiteren Verlaufs fördert.

Als das Potenzial des Storytelling-Ansatzes deutlich wurde, begann Kelleher mit der Entwicklung speziell dazu geeigneter Programmierumgebungen. Die Umgebung aus dem Promotionsprojekt hieß StoryTellingAlice. Kelleher arbeitet mittlerweile mit eigenem Team an LookingGlass<sup>42</sup> und versucht, das Geschichtenschreiben in den Mittelpunkt zu stellen (Kelleher, 2015). Es erlaubt die einfache Darstellung grafischer Figuren, die Interaktion mit diesen Figuren (vgl. LookingGlass-Beispiele), grafische Elemente einfach zu benutzen und mit Programmweisungen anzusteuern, z. B. sie zu bewegen oder Figuren sprechen zu lassen etc.

Da Geschichten zumeist einfache lineare Abläufe darstellen, ist die Auswahl der verwendeten Programmierbefehle etwas eingeschränkt. Adams und Webster vergleichen drei verschiedene Einstiege in das Programmieren (Jugendliche im Al-

<sup>42</sup> <https://lookingglass.wustl.edu>

ter ab ca. 16 Jahren) und stellen fest, dass im Storytelling-Ansatz eher einfachere algorithmische Abläufe verwendet werden:

„Students creating games used the most variables, if statements, and loops. Students creating music videos used nearly as many loops as games, but far fewer variables and if statements. Students creating storytelling projects used the fewest loops, variables, and if statements, but made the most use of dialog<sup>43</sup>“ (Adams & Webster, 2012, S. 648).

Kelleher zufolge ergeben sich dennoch viele Möglichkeiten:

„Users of both Storytelling Alice and Generic Alice experimented with programming constructs beyond simple sequences. A majority of the participants in both groups used Do Togethers to have multiple animations occur simultaneously. 53 % of the users of Storytelling Alice created a new method and used it in their program as opposed to 30 % of the users of Generic Alice. 33 % of the users of Generic Alice used loops as compared to 12 % of the users of Storytelling Alice“ (Kelleher, 2006, S. 184).

Scratch<sup>44</sup> ist eine ähnliche Umgebung. In einem Vortrag<sup>45</sup> wies Michael Resnick, einer der Begründer von Scratch, auf die zahlreichen Ausdrucksmöglichkeiten dieser Programmierumgebung hin. Am meisten berührt hätten ihn Beispiele von Glückwunsch- und Grußkarten, die Kinder z. B. für ihre Großeltern gestaltet hatten. Diese Beispiele (die auf der Scratch-Website veröffentlicht werden können) verweisen auf die Bedeutung der Programmierumgebung für die Erweiterung der Ausdruckskompetenz und deuten zudem darauf hin, dass die Kinder sich das Werkzeug und die Tätigkeit des Programmierens soweit zu eigen gemacht haben, dass sie es kreativ für ihre unmittelbaren Bedürfnisse nutzen können.

Ähnliche Erfahrungen werden immer wieder berichtet: Ausgehend vom Erkunden ähnlicher Beispiele werden die Kinder schnell selbst aktiv und fangen an, kleinere Dinge zu ändern, zuerst ggf. die Farbe und Form der Figur, schließlich immer stärker deren Verhalten. Das eigenständige Explorieren und das Experimentieren mit einzelnen Änderungsmöglichkeiten (**schrittweises Verfeinern**) können zu einem sehr großen Teil (ggf. sogar vollständig) selbstgesteuert durch die

---

43 Dialog bezeichnet Methodenaufrufe, die die Figuren etwas sagen oder sie laut denken lassen

44 <https://scratch.mit.edu>

45 [https://www.ted.com/talks/mitch\\_resnick\\_let\\_s\\_teach\\_kids\\_to\\_code#](https://www.ted.com/talks/mitch_resnick_let_s_teach_kids_to_code#) [Zugriff am 05.03.2018]

Kinder erfolgen. Die Umgebungen bieten hier auch immer mehr Hilfen für einen schrittweisen Einstieg an<sup>46</sup>.

Um diese Eigenaktivitäten zu fördern, sollten Kinder zunächst im Sinne des Ansatzes des ‚Freien Explorierens und Experimentierens‘ (FEE; vgl. Köster, 2006) die Gelegenheit erhalten, sich spielerisch und ohne Anforderungen von außen mit (für Kinder entwickelten) Programmierumgebungen zu beschäftigen. Die Beobachtung der Kinder während dieser ersten Orientierungsphase und dem anschließenden, oft bereits etwas systematischeren Explorieren hilft zu erkennen, wie die Kinder vorgehen, welchen (oft zunächst nicht verbalisierten) Fragen sie nachgehen und wo sie sich Unterstützung wünschen. Erst wenn die Kinder das Bedürfnis nach Unterstützung äußern oder deutlich wird, dass sie allein nicht vorankommen, können Impulse hilfreich sein oder eher offene Aufgaben (wie z. B. „Erzähle deine eigene Geschichte“; „Male dein eigenes bewegtes Bild.“ o. Ä.), um die kreative Ausdrucksfähigkeit und den eigenen Bezug zum Programmierprojekt zu stärken.

Ein wichtiges Feature solcher Werkzeuge und des Vorgehens ist daher die einfache Möglichkeit, sich von anderen Geschichten inspirieren zu lassen und diese als Vorlage für eigene Geschichten zu nutzen (**Adaptieren bzw. Remixen**).

Falls die Kinder dies nicht ohnehin tun, bietet es sich an, die Kinder zwischendurch zum Austausch anzuregen und zu sammeln, welche Möglichkeiten und interessanten Programmierbefehle bereits exploriert wurden. Auch den Aufbau eines Beispiels vorstellen zu lassen, zeigt die Wertschätzung für das bereits Erreichte und kann zu gemeinsamen weiteren Ideen führen<sup>47</sup>. Die Kinder können beschreiben, wie sie den Ablauf einer Geschichte gestaltet haben oder auch weiter gestalten wollen. Dazu können wie bei Filmen/Comics Storyboards gezeichnet werden (vgl. Beispiel oben, Abbildung 43).

Auf diese Weise können die Kinder auch eine erste Vorstellung davon erlangen, was ein Algorithmus ist. Je nach Lerngruppe kann eine Reflexion der verwendeten Ausdrucksmittel im Hinblick auf das Gestalten eines ‚Ablaufs‘ (als Begriff für Algorithmus) erfolgen, um zu verdeutlichen, dass die erzählte Geschichte einen Algorithmus darstellt und entsprechende Befehle verwendet werden können: einfache Anweisungen und Aufrufe von Methoden, ggf. auch Schleifen und Verzweigungen etc.

---

46 ScratchJr (<https://www.scratchjr.org/>) ist eine solche aktuelle Entwicklung. Die Programmierumgebung richtet sich an Kinder im Alter von 5 bis 8 Jahren.

47 Google bietet (englischsprachige) Materialien zum Storytelling-Ansatz an, die ggf. für unsere Zielgruppe angepasst werden könnten: <https://www.cs-first.com/materials> [Zugriff am 05.03.2018].

### **Begründung (erworbene Kompetenzen)**

Programmieren kann und soll als Beitrag zur Entwicklung der Ausdrucksfähigkeit gesehen werden (Schulte 2013), als Teil der **Allgemeinbildung**: Ein Programm ist in diesem Sinne ein ästhetisches Artefakt, ein Ausdrucks- und Kommunikationsmedium. Kinder können und sollen lernen, diese Form des Ausdrucks zu nutzen, und verstehen, wie sie funktioniert. Programmierübungen sind auf das Erstellen eines meist visuellen und animierten kreativ und künstlerisch gestalteten Ergebnisses bezogen und zumeist Selbstzweck, ähnlich wie auf Papier gemalte Bilder.

Während des Gestaltens des Produkts (= des Programmierens) kann sich das Ziel ändern, zum einen, weil neue Ausdrucksmöglichkeiten erkannt werden, zum anderen weil neue Ideen entstehen. Diese Art des Programmierens ist daher (zumeist in einem Rahmen) selbstgesteuert und trägt so zur **Selbstwirksamkeit** im Umgang mit Informatiksystemen und zur Entwicklung von **Interesse** am Programmieren und der Funktionsweise von Programmen bei und fördert die **Motivation**, auch eigenständig weiter zu programmieren. Ein **Lebenswelt-** und **Alltagsbezug** ergibt sich durch Verwendung in Kommunikations- und Gestaltungszusammenhängen, in denen kleine, selbst erstellte ästhetische Produkte eine Rolle spielen (z. B. Glückwunsch- und Grußkarten zu bestimmten Anlässen).

### **Beispiel B: Programmieren unplugged**

#### **Kurzfassung**

Um erste Erfahrungen mit der Programmierung zu machen, existieren zahlreiche Unplugged-Ansätze. Viele dieser Ansätze legen einen Fokus auf die Steuerung eines Roboters.

In einigen Ansätzen sind Kinder selbst die Roboter und werden von anderen Kindern mittels konkreter Befehle durch einen Parcours gesteuert. In anderen Ansätzen werden Roboterfiguren z. B. über Spielfelder gesteuert. Die Programmierung geschieht beispielsweise über Karten oder puzzleähnliche Bausteine.

#### **Ziele**

Ziel ist es, dass die Kinder spielerisch durch Ausprobieren und Experimentieren erste Schritte der Programmierung als wichtiges Konzept der Informatik kennen und verstehen lernen. So lernen sie am Beispiel Roboter, dass dieser konkrete Handlungsanweisungen zur Steuerung benötigt.

## Heatmap

Inhaltsbereiche	(I1) Information & Daten	(I2) Algorithmen & Program- mierung	(I3) Sprachen & Automation	(I4) Informatik- systeme	(I5) Informatik, Mensch & Gesellschaft
Prozessbereiche					
(P0) Interagieren & Explorieren					
(P1) Modellieren & Implementieren	Abläufe erstellen	Handlungs- abläufe formulieren		Präzise Befehle als Eingabe zur Steuerung	
(P2) Begründen & Bewerten					
(P3) Strukturieren & Vernetzen					
(P4) Kommunizieren & Kooperieren		Erklären von Handlungs- vorschriften			
(P5) Darstellen & Interpretieren		Interpre- tieren Handlungs- vorschriften		Zuordnen von Bestand- teilen des Roboters	

## Beschreibung

Eine Einführung in die Programmierung über den Unplugged-Ansatz geschieht, wie üblich, grundsätzlich zunächst ohne den Einsatz eines echten Informatiksystems. Häufig wird für die Unplugged-Vermittlung der Programmierung aber eine Art Pseudo-Informatiksystem benutzt. Dies ist häufig ein Roboter, der von Kindern selbst oder durch eine Figur symbolisiert wird. Grundlegende Programmierkonstrukte und einfache Befehle befinden sich auf Befehlskarten, die auf unterschiedliche Weise über Symbole, Textkarten, Puzzlebausteine o. Ä. visualisiert werden können.

Kinder können verschiedene Aufgabenstellungen bekommen, die der Roboter erfüllen muss, z. B. Mitschülerinnen und Mitschüler durch einen Parcours führen (Dworschak, 2015) oder Roboter über ein Spielfeld steuern. Diese Aufgaben müssen sie mit einer Programmierung lösen, die mit Hilfe der Befehlskarten durchgeführt wird. Dabei kann die Schwierigkeit und die Anzahl der Befehle bzw. Programmierkonstrukte stufenweise erhöht werden. Die so formulierten Handlungsabläufe werden anschließend überprüft, in dem das Partnerkind anhand der Befehle durch den Parcours oder der Roboter über das Spielfeld gesteuert wird (siehe Abbildung 44).



**Abbildung 44.** Kinder programmieren einen Roboter über das Spielfeld (Beispiel aus dem Projekt Informatik an Grundschulen NRW)

Ein weiteres bekanntes Beispiel zum Verständnis von Algorithmen und konkreten Befehlen ist „Jam Sandwich Algorithm“<sup>48</sup>. Hier wird den Kindern die Aufgabe gestellt, einen Roboter (meist von der Lernbegleitung umgesetzt) zu instruieren, ein Brot mit Marmelade zu bestreichen. Ein Teil der Aufgabe besteht darin, einen Algorithmus zu schreiben. Ein ähnliches Beispiel wird bereits auf der Website der Stiftung „Haus der kleinen Forscher“ unter der Rubrik „Experimente + Themen“<sup>49</sup> vorgestellt.

Dieser Unplugged-Ansatz hat den großen Vorteil, dass er kostengünstig ist und keine technische Ausstattung des Lernortes notwendig macht. Dies könnte für einige Lernbegleiterinnen und Lernbegleiter auch eine geringe Einstiegshürde darstellen, da sie beispielsweise keine Angst haben müssen, dass die Technik versagt. Dennoch benötigen sie ein informatisches Grundwissen im Bereich der Programmierung, um entsprechende Lerneinheiten durchführen zu können.

Grundsätzlich halten wir es für sinnvoll, beim Start im Bereich der Programmierung mit einem Unplugged-Ansatz zu arbeiten und in einem weiteren Schritt ein Informatiksystem einzusetzen.

### Begründung

Wesentliches zu entdeckendes Konzept in diesem Beispiel ist das Konzept der Programmierung. Ein Grundverständnis dessen, was Programmierung überhaupt ist, wann ich programmiere und wie sie grundlegend funktioniert, ist wichtig, um in unserer digitalen Welt gut zurechtzukommen, besonders, wenn es darum geht,

48 <http://code-it.co.uk/unplugged/jamsandwich> [Zugriff am 05.03.2018]

49 <http://www.haus-der-kleinen-forscher.de/de/praxisanregungen/experimente-themen/kommunikation/experiment/einen-roboter-steuern/> [Zugriff am 05.03.2018]

Programme an die eigenen Bedürfnisse anzupassen, sei es durch konkrete Einstellungsparameter oder durch Makros.

Rushkoff (2010) hebt mit seiner Formulierung „program or be programmed“ die Bedeutung grundlegender Programmierkenntnisse in unserer digitalen Welt ebenfalls hervor. So ist anzunehmen, dass grundlegende Programmierkenntnisse dazu beitragen, digitale Systeme besser zu verstehen, und darüber hinaus die Kompetenz im Bereich der Endnutzer-Programmierung erhöhen. Diese Kompetenzen sind in unserer heutigen Zeit immer entscheidender, wenn man nicht möchte, dass man für jede Einstellungsänderung im Smart Home etc. eine Technikerin oder einen Techniker rufen muss.

Da auch Kinder immer häufiger Informatiksystemen begegnen, an denen sie selbst zumindest Programmierung auf Endnutzerebene vornehmen können, unterstützen die erworbenen Kompetenzen die Kinder in einem sicheren Umgang mit diesen Systemen.

### **Beispiel C: Aufgaben/Puzzles mit einem Informatiksystem algorithmisch lösen**

#### **Kurzfassung**

Kinder beschäftigen sich mit vorgegebenen kleinen Aufgaben, die algorithmisch mit Hilfe eines Informatiksystems (z. B. App oder Roboter) zu lösen sind und deren Lösung implementiert und ausprobiert werden kann.

#### **Ziele**

- a) Anwenden von Grundstrukturen des Programmierens als Folge elementarer, eindeutiger, meist einfacher Anweisungen (z. B. Schritt vor, drehen)
- b) Hinführung zur Verwendung von Wiederholungen, bedingten Anweisungen, Prozeduren

#### **Heatmap**

Inhaltsbereiche	(1) Information & Daten	(12) Algorithmen & Program- mierung	(13) Sprachen & Automation	(14) Informatik- systeme	(15) Informatik, Mensch & Gesellschaft
Prozessbereiche					
(P0) Interagieren & Explorieren		Zielgerichtet und geführt Lösungs- möglichkei- ten auspro- bieren		Präzise Befehle als Eingabe zur Steuerung	
(P1) Modellieren & Implementieren		Abläufe mo- dellieren			



(P2) Begründen & Bewerten					
(P3) Strukturieren & Vernetzen					
(P4) Kommunizieren & Kooperieren					
(P5) Darstellen & Interpretieren					

### Beschreibung

Vergleichbar zum Unplugged-Beispiel sollen die Kinder einfache Aufgaben lösen, indem sie eine roboterartige Figur mit (i. d. R. Bewegungs-)Anweisungen steuern, um ein bestimmtes Ziel zu erreichen. Die Anweisungen sind in der Regel sofort ausführbar und werden animiert und mit Hilfe des Roboters nachvollzogen. Diese Beispiele knüpfen an Paperts Lernansatz des Konstruktivismus (vgl. Abschnitt 2.3) an, mit dem Ziel, an konkreten Objekten zu lernen.

### Konkrete Ausgestaltung:

- Grafiken zum Thema „Frozen/Die Eiskönigin“ angeleitet erstellen:  
<https://studio.code.org/s/frozen/stage/1/puzzle/1>
- Ein Spiel zum Thema Angry Birds angeleitet erstellen:  
<https://studio.code.org/hoc/1>
- Mit dem Bee-Bot Zahlen „malen“:  
<http://barefootcas.org.uk/wp-content/uploads/2014/09/Bee-Bots-1-2-3-Activity-Barefoot-Computing2.pdf>
- Mit dem Lightbot Aufgaben lösen:  
<https://lightbot.com/>

### Begründung

Siehe Beispiel B (oben).

### 5.1.3 Darstellung und Übertragung von Information

**Priorisierung: I1 (Information & Daten) und P5 (Darstellen und Interpretieren)**

Außerdem adressiert: P0 + P2 + P5/I1 + I4 + I5 und optional: P3 + P4

Dieser Bereich wird mit Beispieleinheiten illustriert, die vermutlich eher in der oberen Altersstufe der Grundschule angesiedelt sind (3./4. Klasse). Für diese Einheiten liegen Erfahrungen aus dem Schülerlabor InfoSphere vor, und eine entsprechende Folge von Lerneinheiten wird aktuell im Projekt „Informatik an Grundschulen“ in Kooperation mit dem Schulministerium Nordrhein-Westfalens an drei Grundschulen erprobt. Ähnliche Kompetenzen (eventuell auf einem niedrigeren Niveau) lassen sich durch die gleiche Kombination von Prozess- und Inhaltsbereichen und altersentsprechenden Beispielen auch für jüngere Kinder konstruieren.

#### Kurzfassung

Information ist neben Stoff und Energie die dritte Grundgröße der Natur (vgl. Wiener, 1948). Der Umgang mit Information führt zu einem permanenten gesellschaftlichen Wandel und sollte bereits von Grundschülerinnen und Grundschülern kennengelernt und verstanden werden. So ist das Ziel, Kindern der Grundschule die Möglichkeit zu eröffnen, Primärerfahrungen im Bereich der digitalen Welt zu sammeln.

Dabei sollen Alltagsfragen der Kinder im Bereich der Datenübertragung aufgegriffen und erläutert, aber auch abstrakte Alltagsprobleme wie z. B. Fehler bei der Datenübertragung und die Möglichkeit ihrer Erkennung erarbeitet und für die Kinder erlebbar gemacht werden. Als Grundlage der digitalen Datenübertragung bildet die Binärdarstellung von digitalen Daten in 0 und 1 das verbindende Element.

In der im Folgenden beschriebenen Beispielumsetzung anhand von vier Bausteinen soll einleitend an die Erfahrungen der Kinder angeknüpft werden. Darauf aufbauend können die Themen Binärdarstellung, Datenübertragung und Fehlererkennung thematisiert werden.

#### Ziele

Ziel möglicher Bausteine zum Thema Informationsdarstellung und -interpretation ist es, Kindern einen Einblick in die digitale Welt zu ermöglichen. Die Kinder sollen erkennen, dass Informationen auf verschiedene Weisen repräsentiert werden können. Für die Verwendung in der digitalen Welt müssen alle Informationen binär codiert werden, was zunehmend auch geschieht: Texte, Bilder, Ton und Video werden durch eine Menge von „0“ und „1“ dargestellt und sind somit durch Informatiksysteme übertrag- und verarbeitbar.

### 1. Baustein: Themenzugang

Die Kinder erarbeiten an QR-Codes, dass Information in verschiedenen Formen (Buchstaben, Zahlen, Codes) dargestellt werden kann.

Es gibt andere altersgerechte Zugänge wie das verteilte Ausmalen von Kästchen, die zusammengesetzt ein Bild ergeben.

### 2. Baustein (eventuell zu verbinden mit Mathematik): Zahldarstellung

Die Kinder lernen den Binärcode kennen. Es erfolgt die Erkenntnis, dass im vierstelligen Binärcode alle Zahlen von 0–15 darstellbar sind.

### 3. Baustein: Datenübertragung

Die Kinder werden in die Lage versetzt, mit dem angeeigneten Wissen zum Binärcode einfache Botschaften zu übertragen und auch empfangene Botschaften zu entschlüsseln.

### 4. Baustein: Fehlererkennung

Die Kinder lernen mit Hilfe eines Kontrollbits eine Möglichkeit der Fehlererkennung kennen.

Die Lernenden

- beschreiben ihren persönlichen Umgang mit Informatiksystemen,
- erläutern das Prinzip der Dateneingabe und der Datenübertragung,
- begründen die Einschränkung auf die Zustände Strom an (1) und Strom aus (0),
- erklären, wie Information in digitale Daten umgewandelt werden kann,
- führen die Umwandlung von Zahlen und Buchstaben in Binärcode wie auch in umgekehrter Richtung selbst durch,
- erläutern die Notwendigkeit von Fehlererkennungsmechanismen bei der Datenübertragung und
- beschreiben ein vorgegebenes Verfahren zur Lokalisierung einer fehlerhaften Stelle und wenden dies an.

## Heatmap (Zusammenfassung der 4 Einheiten):

Inhaltsbereiche	(11) Information & Daten	(12) Algorithmen & Programmierung	(13) Sprachen & Automation	(14) Informatiksysteme	(15) Informatik, Mensch & Gesellschaft
<b>Prozessbereiche</b>					
<b>(P0) Interagieren &amp; Explorieren</b>	verschiedene Darstellungsformen entdecken	Umwandlung von Dezimal in Binärzahlen		Informatiksysteme stellen Information für User verständlich dar	Information wird repräsentiert und muss (korrekt) interpretiert werden
<b>(P1) Modellieren &amp; Implementieren</b>					
<b>(P2) Begründen &amp; Bewerten</b>	Unterschied zwischen Information & Daten			Informatiksysteme arbeiten mittels der Zustände Strom an (1) und aus (0)	manche Darstellungsformen für Menschen, andere für Informatiksysteme einfacher zu lesen
<b>(P3) Strukturieren &amp; Vernetzen</b>	Repräsentieren von Information gibt es bei Sprachen, Bildern, ...				
<b>(P4) Kommunizieren &amp; Kooperieren</b>	Datenübertragung				
<b>(P5) Darstellen &amp; Interpretieren</b>	Dezimalzahlen und Buchstaben in Binärcode darstellen	Umwandlung von Dezimal in Binärzahlen			Bei Übertragung oder Interpretation können Fehler passieren

**Beschreibung**

Mögliche konkrete Umsetzungsideen für eine Reihe aus 4 Einheiten mit den Themen: Codierung, Binärdarstellung, Datenübertragung, Fehlererkennung:

## Einheit 1 – QR-Code

In einer ersten Einheit sollen die Kinder den Bezug der im Verlauf thematisierten Darstellungsformen zu ihrem eigenen Leben erkennen. Dazu dienen QR-Codes als Einstieg. Die Kinder beschreiben diese (zentral ist dabei die reine Färbung in

Schwarz und Weiß, also zwei Zustände) und erörtern, wo sie diese im Alltag bereits entdeckt haben.

### Einheit 2 – Binärdarstellung

Ziel der zweiten Einheit ist es, dass die Kinder erkennen, dass es neben der bekannten Dezimaldarstellung auch eine bestimmte Darstellungsform gibt, mit der Informatiksysteme arbeiten: die Binärdarstellung. Auch Vorschulkindern sind bereits unterschiedliche Repräsentationen von Zahlen bekannt, z. B. Finger, Striche, Streichhölzer oder Äpfel. Wichtig ist dabei, dass die Kinder den Vergleich bzw. die Analogien zwischen dem bekannten zehnstelligen Dezimalsystem und dem zweistelligen Binärsystem bemerken. In beiden Systemen können mit einer bestimmten Zifferanzahl nur eine bestimmte Anzahl verschiedener Zahlen dargestellt werden (mit drei Ziffern im Dezimalsystem die Zahlen von 0 bis 999 und entsprechend im Binärsystem nur die Zahlen von 0 bis 7). Ebenso wichtig ist es, dass durch die Lernbegleitung der Bezug zu bekannten Informatiksystemen (Computer, Smartphone, Tablet) hergestellt wird. Es soll thematisiert werden, dass Informatiksysteme über das ‚Ein- und Ausschalten von Strom‘ arbeiten und durch die (sehr schnellen) Wechsel Daten über ein Kabel übertragen werden können (siehe Einheit 3).

Um ein Gefühl für diese Form der Zahldarstellung zu erwerben, könnten die Kinder eine Liste mit allen möglichen Kombinationen anlegen und somit feststellen, dass durch den vierstelligen Code die Ziffern 0 bis 15 dargestellt werden können.

Wichtig ist weiter, dass die Schülerinnen und Schüler verstehen, dass die Form der Binärdarstellung auch auf andere Dinge mit zwei Zuständen (Lampe an/aus, Kärtchen in Blau/Rot) übertragbar ist.

### Einheit 3 – Datenübertragung

Ziel dieser Einheit ist es, das erworbene Wissen über Zahldarstellungen auf Buchstaben und Satzzeichen zu erweitern und Möglichkeiten der Übertragung dieser zu erfahren.

Die Einheit knüpft direkt an das Ergebnis der vorherigen Einheit an, indem mit einer Codetabelle zur Binärdarstellung von Buchstaben und Satzzeichen statt Dezimalzahlen gearbeitet wird. Damit können Wörter oder auch Sätze codiert werden. Dies kann entweder händisch auf Papier notiert und anschließend als Botschaft weitergereicht werden oder auch mittels Schaltern, Kabeln und Lämpchen elektrisch geschehen. So können sich die Kinder gegenseitig „wie ein Computer“ Nachrichten senden. Damit wird der gesamte Prozess von der Information über die Binärdarstellung, die Übertragung und schließlich die Rücküberführung

mehrfach wiederholt und gefestigt. Auch hier sollte wieder der Bezug zu realen Informatiksystemen hergestellt werden.

#### Einheit 4 – Fehlererkennung

Diese letzte Einheit zielt darauf ab, dass den Lernenden das Problem der Übertragungsfehler bewusst wird und sie eine erste Idee erfahren (beispielsweise mittels Prüfbits), wie diese Fehler erkannt werden können.

Um den Kindern die Thematik der Übertragungsfehler intuitiv zugänglich zu machen, braucht es nur eine fehlerhafte Nachricht (z. B. könnte statt einem E (00101) ein A (00001) im Wort KINDE/AR übertragen werden). Anhand eines solchen Beispiels wird in einem Unterrichtsgespräch das Thema Übertragungsfehler diskutiert. Auch kann am Beispiel einer Robotersteuerung verdeutlicht werden, dass ein Übertragungsfehler nicht nur zu einem falschen Buchstaben in einer Nachricht führen, sondern weit schwerwiegendere Auswirkungen haben kann. Anschließend können von den Kindern selbst Ideen zur Fehlererkennung entwickelt werden. Schlussendlich soll die Einheit auf die Möglichkeit eines Prüfbits hinauslaufen, was den Kindern anhand einer weiteren Stelle in der Codetabelle verdeutlicht werden kann, die so mit 0 oder 1 gefüllt wird, dass die Anzahl der Ziffern jeweils gerade ist.

#### **Begründung**

Das wesentliche zu entdeckende Konzept ist, dass **Information** auf verschiedene Weisen dargestellt werden kann. Für die digitale Welt ist entscheidend, dass es reicht, zwischen zwei Zuständen zu unterscheiden, und damit beliebige Information codiert werden kann. Diese beiden Zustände können (einfach) über verschiedene Wege übertragen werden. Binärdarstellungen können beliebige Texte, Zahlen, Farben, Adressen, Bilder, Videos, strukturierte **Daten** wie Tabellen, Sportergebnisse etc. repräsentieren.

Durch die einheitliche Codierung kann Information einfach übertragen werden und von verschiedenen Informatiksystemen „überall auf der Welt“ verstanden und für Menschen in ihrer Sprache und einem geeigneten Format (z. B. Bilder, Grafiken, Tabellen) dargestellt werden (Codierung, Übertragung, Verarbeitung, Decodierung). Das Prinzip der Darstellung von Daten in verschiedenen Repräsentationen, z. B. verschiedene natürliche Sprachen, gesprochenes oder geschriebenes Wort, Bilder oder symbolische Darstellung, Tabellen, Fußballergebnisse etc., ist essenziell für Kommunikation und damit auch außerhalb der Informatik bedeutend.

Der **Alltagsbezug** für die Kinder kann hergestellt werden, indem bekannte Informationsdarstellungen aufgegriffen werden, z. B. Ampelsignale, QR-Codes, Farbmarkierung bei Zuordnung von Gegenständen unter Geschwistern etc.

### 5.1.4 Explorieren und Strukturieren des Internets

**Priorisierung: P0/I4**

Außerdem adressiert: P2 + P3/I1 + I5

#### **Kurzfassung**

Täglich nutzen wir das Internet. Wir schauen uns Websites an, schreiben E-Mails, chatten mit unseren Freundinnen und Freunden usw. Aber wie funktioniert das alles? Wie kommt die Website auf den eigenen Computer?

Unter dieser Fragestellung könnten Kinder die einzelnen Bestandteile und Ideen hinter der Technik kennenlernen. Eine Idee wäre, die Beschäftigung mit dem Thema in zwei große Bestandteile zu gliedern. Im ersten Teil lernen die Kinder in Kleingruppen die verschiedenen Technologien und Verfahrensweisen kennen, die für die Funktionsweise des Internets wichtig sind. Hierzu eignet sich besonders ein Stationenlernen, wobei sich jede Station einem bestimmten Thema widmet und dieses anhand von kleinen Experimenten erlebbar macht. Die Stationen sollten sich dabei mit den folgenden Themen beschäftigen:

- Umrechnung von Dezimalzahlen in Binärzahlen (siehe Abschnitt 5.1.3)
- Übertragung von Daten (siehe Abschnitt 5.1.3)
- Das schnelle Netz im Hintergrund
- Zerlegung und Bauplan einer Website
- Die Funktionsweise eines DNS-Servers
- Das Klient-Server-Prinzip
- Aufbau von Internetadressen
- Sicherheit bei der Datenübertragung

Zur Sicherung eignet sich beispielsweise ein Quiz, indem die Schülerinnen und Schüler ihr erworbenes Wissen noch einmal wiederholen.

#### **Ziele**

Hauptziel ist hierbei, dass die Kinder die grobe Struktur des Internets erkennen und ein Gefühl dafür entwickeln, dass Daten über das Netzwerk verschickt und auch außerhalb des eigenen Computers (oder Tablets, Smartphones) gespeichert

werden. Sie erkennen die Notwendigkeit des Einloggens auf bestimmten Websites. Dieses Grundwissen über die Technik des Internets befähigt die Kinder, zukünftig dazu ihr Verhalten im Internet fundiert zu reflektieren und sich so über Panikmache und Verherrlichung hinweg eigenständig ein Bild zu machen. Im Einzelnen können die folgenden Zielsetzungen und Themenbereiche in dem Beispiel adressiert werden:

■ **Bezug zum eigenen Umgang mit dem Internet**

Den Kindern soll der eigene momentane und zukünftige Bezug zum Internet vor Augen geführt werden.

■ **Internet als Zusammenspiel der Komponenten**

Durch die Dekonstruktion des Informatiksystems Internet in seine Komponenten können sich die Lernenden einen Gesamtzusammenhang erarbeiten.

■ **Repräsentation und Transport von Informationen als Daten**

Der Lerninhalt zur Abstraktion von Information (z. B. E-Mails und Websites) in Form von Daten stellt ein grundlegendes Prinzip dar.

■ **Handlungsabfolgen und Klient-Server-Anwendungen**

Das Klient-Server-Prinzip und das Prinzip eines DNS-Servers sind als Handlungsabfolgen im Rahmen des Internets als grundlegende Prinzipien zu vermitteln.

■ **Sicherheit der Datenübertragung und Verschlüsselung**

Die Kinder können bereits in der Grundschule für die Problematik und die Gefahren bei der Benutzung des Internets sensibilisiert werden.

■ **Aufbau einer Internetadresse (URL)**

Die URL als zentrales Element des Internets soll in ihrer Struktur vermittelt werden, um ein Feststellen ihrer Gültigkeit zu ermöglichen.



## Heatmap

Inhaltsbereiche	(I1) Information & Daten	(I2) Algorithmen & Program- mierung	(I3) Sprachen & Automation	(I4) Informatik- systeme	(I5) Informatik, Mensch & Gesellschaft
<b>Prozessbereiche</b>					
(P0) Interagieren & Explorieren					
(P1) Modellieren & Implementieren					
(P2) Begründen & Bewerten					Alltagsbe- deutung des Internets
(P3) Strukturieren & Vernetzen			Aufbau einer URL	Struktur von Netzwerken/ des Internets	Datensi- cherheit
(P4) Kommunizieren & Kooperieren	Datenüber- tragung in Netz			Internet- adressen/ URL	E-Mails ver- schicken
(P5) Darstellen & Interpretieren	Binär darstellung			Klient- Server- Modell	

## Beschreibung

Die nun folgende Beschreibung stellt Ideen aus dem Modul „Wie funktioniert das Internet?“ des Schülerlabors InfoSphere dar.

Zur Motivation könnten mit den Kindern in einem Brainstorming verschiedene Möglichkeiten des Internets anhand der Fragestellung ‚Was kann das Internet?‘ gesammelt werden. Dadurch erfahren und reflektieren sie zu Beginn, welche Fähigkeiten und Anwendungen im und mit dem Internet möglich sind. Anschließend kann zu der Fragestellung: ‚Wie kann das Internet das alles?‘ übergeleitet werden.

Als Grundlage für die Bearbeitung der Stationen wird der Schritt von den Daten hin zur Repräsentation als Binärzahlen zum Transport dieser Daten aufgezeigt. Als Beispiel aus dem Alltag für das Strom-An- und Ausschalten kann eine Glühbirne, die mit einem Schalter ausgestattet werden kann, dienen. Auf diesem Wissen aufbauend wird dann beispielhaft die Darstellung eines Bildes als Zahlenkette erklärt. Hierbei wird das Bild bis auf den einzelnen Pixel reduziert und dieser in eine Binärzahl codiert. Damit könnte diese Einheit an das vorausgegangene Beispiel zum Thema Binärdarstellung anschließen.

Im weiteren Verlauf der Einheit arbeiten die Kinder in Kleingruppen an Stationen und eignen sich jeweils einen Aspekt des Internets an. Aufgrund der im ersten Teil gelegten inhaltlichen Basis ist es möglich, dass jede Station des Rundgangs als Startpunkt für die Bearbeitung der Stationen genutzt werden kann. Um den Kindern

eine Orientierung bei der Bearbeitung der Stationen zu geben, sollten diese im Vorfeld vorgestellt und deren jeweilige Startpunkte und Abläufe kurz erläutert werden. Außerdem könnten die Kinder Laufzettel erhalten, auf denen sie für jede bearbeitete Station einen Stempel erhalten, um selber den Ablauf zu überblicken.

Über alle Stationen hinweg tauchen einige Modelle immer wieder auf. Damit behalten diese modellhaften Abbildungen ihre Bedeutung für die Dauer des Stationenlernens. Binärzahlen könnten in Form von kleinen Holzmurmeln verkörpert werden. Dabei stellt die Farbe Weiß die Zahl 0 und Schwarz die Zahl 1 dar. Diese Kugeln werden auf verschiedene Arten transportiert. Hauptsächlich kommen als Modell für die Kabelverbindungen Rohre zum Einsatz, durch die die Kugeln geschickt werden können.

Im Anschluss an das Stationenlernen könnte das neu erworbene Wissen über das Internet in einem Quiz im Stil der Fernsehsendung ‚1, 2 oder 3‘ gefestigt werden. Hierbei werden Fragen zu den einzelnen Stationen mit drei möglichen Antworten präsentiert, und die Kinder müssen sich in das Feld stellen, hinter dem sie die richtige Antwort vermuten. Bei richtigen Antworten bekommen die Kinder Punkte. Auf diese Weise wird der Wettbewerbscharakter dieses Quiz gefördert und damit Leistungsmotivation erzeugt und aufrechterhalten.

Dieser Teil hat vor allen Dingen zwei Absichten: Die in den Stationen gelernten Inhalte sollen wiederholt und gefestigt werden. Die Fragen sind daher so gewählt, dass sie jeweils zu dem Lernziel einer Station passen.

Weitere Informationen zum InfoSphere-Modul sind unter <http://schuelerlabor.informatik.rwth-aachen.de/module/internetspiel> zu finden.

### **Begründung**

Als weltweites Kommunikations- und Informationsmedium hat das Internet eine umfassende gesellschaftliche Bedeutung. Zur Teilhabe am gesellschaftlichen Leben sind Fähigkeiten und Fertigkeiten und vor allem ein Verständnis dieser Technologien unabdingbar geworden. Insbesondere der Zugang zu gesamtgesellschaftlichem Wissen findet vermehrt über das Internet statt. Ohne eine entsprechende Vorbildung können Kinder zukünftig nicht mehr uneingeschränkt am gesellschaftlichen Leben (im Alltag wie im Beruf) teilhaben.

Das Smartphone löst das herkömmliche Handy ab, und der Computer zieht mehr und mehr in die Kinder- und Jugendzimmer ein. Das zeigt eine Studie der Initiative D21 mit dem Titel „Bildung via Internet: Wie vernetzt sind Deutschlands Kinder“ (Initiative D21 e. V., 2008). In der Altersgruppe der 7- bis 10-Jährigen zeigt sich eine Nutzung des Computers im Elternhaus von 86,2 %. Bei den 11- bis 15-Jährigen steht sogar zu 93,7 % ein Computer zu Hause zur Verfügung. Von diesen nutzen die meisten (87,1 %) den Computer, um ins Internet zu gehen (vgl. Initiative D21 e. V., 2008, S. 7).

### 5.1.5 Modellierung von Automaten

**Priorisierung:** P1 (Modellieren und Implementieren)/I3 (Sprachen und Automaten)

Außerdem adressiert: P0 + P5/I4 + I5 und optional: P1/I2

**Beispiel:** Eine Ampelanlage an einem Fußgängerüberweg (Kreuzung) als Informatiksystem erkunden und modellieren

#### Kurzfassung

Das Beispiel beschreibt exemplarisch eine mögliche Umsetzung des Kompetenzbereichs P1/I3. Mit der Erkundung einer realen Ampelanlage können in dem Beispiel verkehrserzieherische und informatische Aspekte miteinander verbunden werden. Das Beispiel soll dazu beitragen, dass Kinder die Fähigkeit entwickeln, Informatiksysteme in ihrer realen Welt zu entdecken und damit zusammenhängende informationsverarbeitende Prozesse zu erkennen. Das Beispiel ist geeignet, den Kindern die wichtigen informatischen Konzepte eines endlichen Automaten und der zustandsorientierten Modellierung zu erschließen. Gleichzeitig kann die gesellschaftliche Bedeutung solcher Systeme, etwa durch den Vergleich mit einer Verkehrsregulierung durch einen Menschen, erfahrbar gemacht werden.

#### Ziele:

Die Kinder sollen...

- eine Ampelanlage und ihre Funktionen beschreiben können,
- die Bedeutung einer Ampelanlage für die Verkehrsregelung beschreiben können,
- die Ampelanlage als Informatiksystem mit Eingabe- und Ausgabegeräten beschreiben können,
- die Ampelanlage als Automaten mit Eingabezeichen, Ausgabezeichen, Zuständen, Regeln für Zustandsübergänge und einer dafür verwendeten ‚Sprache‘ beschreiben können,
- ein einfaches Nutzungsszenario (Fußgängerampel) als Automat modellieren können und
- (optional) eine virtuelle Ampelsimulation am Bildschirm oder mittels eines Modells mit einer visuellen Programmiersprache (z. B. Scratch/Arduino) programmieren können.

## Heatmap

Inhaltsbereiche	(11) Information & Daten	(12) Algorithmen & Program- mierung	(13) Sprachen & Automation	(14) Informatik- systeme	(15) Informatik, Mensch & Gesellschaft
<b>Prozessbereiche</b>					
(P0) Interagieren & Explorieren	Rot als Information für Stopp, Grün als Go erkennen		Ampelmodell auf Zustände, Ein- und Ausgaben erkunden	Ampelanlage als Informatiksystem betrachten	Reale Ampelanlage erkunden
(P1) Modellieren & Implementieren	Straßenbahn- vs. Auto- vs. Fußgängerampel	Optionale Erweiterung: Implementierung mit Scratch/Arduino	Ampelanlage zustandsorientiert modellieren		
(P2) Begründen & Bewerten					
(P3) Strukturieren & Vernetzen					
(P4) Kommunizieren & Kooperieren					
(P5) Darstellen & Interpretieren			Grafisches Automatenmodell der Ampel vorstellen		

## Beschreibung

Aus informatischer Sicht können an diesem Beispiel wesentliche Aspekte der zustandsorientierten Modellierung verdeutlicht werden. Im Gegensatz zur realen Welt wird im Prozess der Systemmodellierung durch kontextuelle Reduktion (De-kontextualisierung), Abstraktion und Formalisierung ein abstraktes Modell eines Informatiksystems erzeugt.

Folgende Aktivitäten der Kinder können (teilweise optional) in einem pädagogischen Lernszenario initiiert werden:

- Kinder erkunden, wie die Ampelanlage an einer Kreuzung funktioniert (eventuell in Kombination mit Verkehrserziehung),
- Anlage kann mit Papier oder Holzklötzchen nachgebaut werden,

- Nutzungsszenarien können (z. B. mit Playmobilfiguren o. Ä. oder im Rollenspiel) simuliert werden,
- Zustände einer Ampel werden betrachtet,
- Abhängigkeiten der Zustände der Ampeln an der Kreuzung werden betrachtet,
- Ampel kann als endlicher Automat mit Zuständen und Zustandsübergängen betrachtet werden,
- Ampelkreuzung kann als Automat mit Übergängen betrachtet werden,
- Fußgängerampel mit Eingabe (gedrückt; nicht gedrückt) kann als Einführung in die zustandsorientierte Modellierung verwendet werden,
- Zusammenhang zu ‚Sprache‘ mit Eingabe- und Ausgabealphabet herstellen,
- mögliche andere Beispiele: Eisenbahnschranke, Baustellenampel,
- gesellschaftlicher Aspekt: Vergleich der Verkehrsregelung durch Polizistin/Polizist und Ampel,
- Optionale Erweiterung auf Programmierung/Simulation mit Scratch am Bildschirm bzw. Steuerung eines Kreuzungsmodells, z. B. mit Arduino.

Ausgangspunkt kann die Erkundung eines realen ampelgesicherten Fußgängerüberwegs mit Druckknopf sein. Im weiteren Verlauf dieses didaktischen Bausteins kann auch auf ein Bild (Video) des Fußgängerüberwegs zurückgegriffen werden. Folgender weiterer Verlauf des Lernszenarios wäre denkbar: Aus dem Foto des Fußgängerüberwegs kann eine Zeichnung oder ein Modell aus Pappe mit Farbplättchen erstellt werden. Das Szenario „Straße überqueren“ kann im Rollenspiel oder mit Figuren gespielt werden. Die Zustände der Ampeln und des Druckknopfs können in eine Tabelle eingetragen werden. Zustandsübergänge in Abhängigkeit vom Druckknopf sowie Ein- und Ausgabesymbole (gedrückt, nicht gedrückt, rot, gelb, grün) können durchgespielt und aufgeschrieben werden. Ein variabler Timer (Uhr) als Impulsgeber für Zustandsübergänge kann berücksichtigt werden. Damit kann das Prinzip eines endlichen Automaten als abstraktes Konzept der Informatik verdeutlicht werden, das sich von der technischen Realisierung eines Automaten (z. B. Getränkeautomat) klar unterscheidet.

Weitere einfache Beispiele wären z. B. eine Eisenbahnschranke, eine Baustellenampel, eine automatische Tür oder ein einfacher Getränkeautomat.

Je nach Lerngruppe kann das Beispiel komplexer gestaltet werden (Kreuzung mit Ampel, Kreuzung mit Fußgängerampel, Kreuzung mit Kontaktschwelle für Autos). Optional kann der Automat auch algorithmisch realisiert werden, indem man z. B. mit Scratch ein entsprechendes Programm erstellt und auf dem Bildschirm visualisiert bzw. ein Ampelmodell mit LEDs mittels eines Arduino ansteuert.

Unter dem Gesichtspunkt der Übernahme von menschlichen Tätigkeiten durch Automaten kann eine Verkehrsregelung durch die Polizei mit der durch eine Ampel verglichen werden (oder: Getränkeautomat und Verkäuferin/Verkäufer).

Exemplarisch soll an diesem letzten Beispiel auch gezeigt werden, wie die in Kapitel 3.4 beschriebenen Kriterien für die Auswahl von Inhalts- und Prozessbereichen angewendet werden können, um die priorisierte Auswahl eines Inhaltsbereichs zu begründen. Ferner können die Kriterien wichtige Hinweise zur fachdidaktischen Umsetzung des Beispiels unter Berücksichtigung der kognitiven und motivationalen Voraussetzungen der Lerngruppe liefern.

### **Anwendung der Kriterien auf die Priorisierungsempfehlung P1/I3**

- **Fachliche und fachdidaktische Konzepte:** Zustandsorientierte Modellierung und das Konzept des endlichen Automaten sind wichtige informatische Konzepte, die für die Modellierung und das Verstehen von Informatiksystemen große Bedeutung haben.
- **Lern- und entwicklungspsychologische Aspekte:** Die Kinder sollten in der Lage sein, den Abstraktions- und Formalisierungsprozess vom realen Informatiksystem hin zum abstrakten Modell (mit Hilfestellung) nachvollziehen zu können.
- **Alltagsbezug:** Verkehrssituationen und Ampelanlagen sind wichtige Elemente der kindlichen Alltagserfahrung.
- **Motivation:** Wie eine Ampel und das Konzept eines Automaten funktioniert, kann motivierend und spielerisch erschlossen werden.
- **Fachliches Interesse:** Grundlegende Funktionsprinzipien eines Informatiksystems und dessen innere Zustände spielerisch zu erkunden und zu gestalten, können das Interesse an Informatik wecken.

- **Selbstwirksamkeit:** Das Modellieren und ggf. (Unplugged-)Programmieren eines aus dem Alltag bekannten Informatiksystems, z. B. eine Ampelanlage, können das Vertrauen der Kinder in ihrem Umgang mit und ihre Gestaltungsfähigkeit von Informatiksystemen fördern.
- **Allgemeinbildung:** Der selbstbewusste Umgang der Kinder mit dem Informatiksystem ‚Ampelanlage‘, der Erwerb eines Konzeptverständnisses (Automat, Innensicht- und Außensicht auf Informatiksysteme) und erste Erfahrungen mit Automatisierungsprozessen betreffen vor allem die folgenden allgemeinbildenden Aspekte: A2, A4, A6, A7, A9, A12 (siehe Kapitel 3.4).
- **Übergreifende Basiskompetenzen:** Das Erkunden der Ampelanlage, das Gestalten des Modells und dessen Erklärung sowie die Erprobung von Anwendungssituationen an der Ampelkreuzung im Rollenspiel können Kommunikations- und Teamfähigkeit fördern.
- **Bezug zu didaktischen Konzepten:** Es ergeben sich Anknüpfungspunkte zur Verkehrserziehung und zum Sachunterricht (Steuerung des Ampelmodells).
- **Existierende Praxiserfahrungen:** Es existieren entsprechende Unterrichtsmodelle in der informationstechnischen Grundbildung; allerdings in Bezug auf eine ältere Adressatengruppe.

## 5.2 Zusammenfassende Heatmap der Priorisierung in den Beispielen

Fast man die Heatmaps aus den einzelnen oben vorgestellten Beispielen zusammen, so ergibt sich eine Empfehlung für die Priorisierung von Kompetenzfeldern, die in der frühen informatischen Bildung in der Grundschule und im Kindergarten umgesetzt werden kann (siehe Abbildung 45). Diese Empfehlung enthält jedoch keine Vorschläge für eine Abfolge der Kompetenzfelder und damit zusammenhängend für eine Sequenzierung von Themen, da die adressierten Kompetenzfelder in den Beispielen weitgehend unabhängig voneinander behandelt werden können und sich demzufolge keine hierarchische Abhängigkeit von zu erwerbenden Kompetenzen in diesen Bereichen ergibt. Allerdings sind die oben vorgestellten Beispiele unterschiedlich komplex und sollten daher von den pädagogischen Fach- und Lehrkräften in Kenntnis der kognitiven Fähigkeiten und motivationalen Präferenzen ihrer Lerngruppen ausgewählt werden.

Inhaltsbereiche	(11) Information & Daten	(12) Algorithmen & Program- mierung	(13) Sprachen & Automation	(14) Informatik- systeme	(15) Informatik, Mensch & Gesellschaft
Prozessbereiche					
<b>(P0) Interagieren &amp; Explorieren</b>		Verhaltensmuster explorieren, Experimente durchführen (1: Interagieren und Explorieren von Informationssystemen)		Orientieren und Explorieren von Bestandteilen und Funktionalität (1: Interagieren und Explorieren von Informationssystemen)	
<b>(P1) Modellieren &amp; Implementieren</b>	Abläufe erstellen (2: Algorithmen und Programmierung)	Ablauf planen, modellieren, Handlungsabläufe formulieren (2: Algorithmen und Programmierung)	Ampelanlage zustandsorientiert modellieren (5: Modellierung von Automaten)	Präzise Befehle als Eingabe zur Steuerung (2: Algorithmen und Programmierung)	
<b>(P2) Begründen &amp; Bewerten</b>					
<b>(P3) Strukturieren &amp; Vernetzen</b>				Struktur von Netzwerken/des Internets (4: Explorieren und Strukturieren des Internets)	
<b>(P4) Kommunizieren &amp; Kooperieren</b>	Datenübertragung im Netz (4: Explorieren und Strukturieren des Internets)				
<b>(P5) Darstellen &amp; Interpretieren</b>	Dezimalzahlen und Buchstaben in Binärcode darstellen (3: Darstellung und Übertragung von Information)	Interpretieren Handlungsvorschriften (2: Algorithmen und Programmierung)			

Abbildung 45. Zusammenfassende Heatmap der Priorisierung von Kompetenzfeldern



## 6 Gelingensbedingungen früher informatischer Bildung

### 6.1 Rahmenbedingungen für eine erfolgreiche Umsetzung

Da bisher hierzu nur wenige Forschungsergebnisse vorliegen, beziehen wir uns im Folgenden u. a. auf vergleichbare Gelingensbedingungen, wie sie bei Benz et al. (2017) für die frühe mathematische Bildung dargestellt worden sind, und wenden diese, soweit sinnvoll, auf den Bereich der informatischen Bildung an.

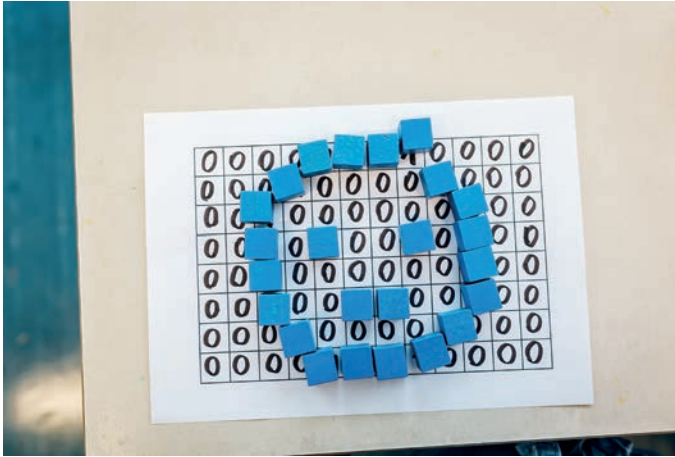
Eine erfolgreiche Förderung informatischer Bildung hat die Erfüllung verschiedener Bedingungen zur Voraussetzung, die zum einen die Kompetenzen und Einstellungen der pädagogischen Fach- und Lehrkräfte betreffen, zum anderen aber auch institutionelle und Ausstattungsrahmenbedingungen einschließen. Die Rahmenbedingungen bezogen auf die Institutionen betreffen vor allem die Schaffung der zeitlichen, räumlichen, organisatorischen und ausstattungsbezogenen Voraussetzungen.

Zusammenfassend lässt sich besonders als Fazit aus den Ausführungen in Kapitel 4 festhalten, dass pädagogische Fach- und Lehrkräfte:

- a) Fachkompetenz besitzen sollten, insbesondere um angebotenes Lehr-Lern-Material zu verstehen und darauf basierend eigene Lerneinheiten zu planen und zu gestalten,
- b) pädagogisch-didaktische Handlungskompetenz benötigen, um Lerngelegenheiten bezogen auf informatische Bildung zu erkennen und zu nutzen und angebotene Materialien lerngruppenadäquat auszuwählen, anzupassen und ggf. weiterzuentwickeln,
- c) positive Einstellungen zur Informatik besitzen, aber auch gegenüber existierenden Vorstellungen offen für neue Sichtweisen sein sollten.

#### 6.1.1 Subjektive Theorien und Einstellungen der pädagogischen Fach- und Lehrkräfte zur Informatik

Bisher liegen nur wenige Erkenntnisse zum Einfluss von Einstellungen zur Informatik und von subjektiven Theorien auf den informatischen Kompetenzerwerb vor. Es ist allerdings, basierend auf den Erkenntnissen aus anderen Fächern, zu vermuten, dass diese einen direkten Einfluss darauf haben, wie und mit welcher Begeisterung ein Thema von den Fach- und Lehrkräften vermittelt wird.



Nicht zuletzt wird das Thema auch durch in eigenen Nutzungserfahrungen mit digitalen Medien erworbene mentale Barrieren und Ängste beeinflusst. So identifizierten Knobelsdorf und Schulte (2007) bei Studienanfängerinnen und -anfängern entwickelte Haltungen als „Insider“ oder „Outsider“, die dazu führten, dass sich „Outsider“ vor allem auf das Anwenden von Computerprogrammen beschränkten

und Probleme im Wesentlichen hinnahmen, „Insider“ dagegen den Computer, und damit die Informatik, als gestaltbares Werkzeug auffassten.

Da davon auszugehen ist, dass bei vielen Fach- und Lehrkräften im Kita- und Grundschulbereich unvollständige und falsche Vorstellungen von Informatik und, damit korrespondierend, informatischer Bildung existieren, sollten diese offen für eine Änderung der Sichtweise und damit verbundenen eigenen Lernerfahrungen sein. Sie sollten bestenfalls einen ‚Schritt weit‘ entfernt sein, um sich vom „Outsider“ zum „Insider“ zu wandeln.

Da zunehmend Kreativität als Ziel informatischer Bildung in den Vordergrund rückt, ist eine dem kreativen Potenzial entsprechende Gestaltung informatischer Bildung anzustreben (vgl. Romeike, 2008).

### 6.1.2 Fachdidaktische Kompetenzen der pädagogischen Fach- und Lehrkräfte

Der Alltag bietet vielfältige Anknüpfungsmöglichkeiten für informatische Bildung, nicht zuletzt, da die Lebenswelt der Kinder zunehmend durch digitale Artefakte geprägt ist (vgl. Kapitel 2.1). Borowski, Diethelm und Wilken (2016) erhoben entsprechend bei Kindern der dritten und vierten Klasse, welche Fragen sie über „Computer, Handys, Roboter usw.“ gern beantwortet haben würden. Die sich daraus ergebenden Fragen weisen eine große Bandbreite informatikbezogener Themen auf: über die Geschichte, Zukunft und Funktion von Informatiksystemen bis hin zu Sicherheits- und rechtlichen Aspekten. Solche intrinsisch verankerten Fragen bieten Potenzial für Lerngelegenheiten zur informatischen Bildung. Für die pädagogischen Fach- und Lehrkräfte liegt allerdings eine Schwierigkeit darin, das informatische Potenzial hinter den Fragen der Kinder zu erkennen sowie geeignete Alltagssituationen für die Gestaltung von Lernsituationen zu identifizieren. Zusätzlich besteht die Gefahr dieser Sichtweise, dass informatische Lerngele-

genheiten ausschließlich mit Informatiksystemen verknüpft werden und darüber hinaus ein verzerrtes Bild entsteht, dass grundlegende Ideen und Konzepte, die keinen unmittelbaren Bezug zu Informatiksystemen aufweisen, vernachlässigt werden.

Analog zu den Anforderungen, die in der Expertise zur Mathematik formuliert wurden, kommen den pädagogischen Fach- und Lehrkräften anspruchsvolle Aufgaben zu (Benz et al., 2017):

1. Aus dem vorhandenen *Lehr-Lern-Material* zur informatischen Bildung, aber auch aus anderen Kontexten (vgl. Kapitel 2.3) muss das *Material* ausgewählt werden, das hohes informatisches Potenzial erkennen lässt und dem aktuellen Entwicklungsstand der Kinder entspricht.
2. Es müssen *Lernsituationen* geschaffen oder aufgegriffen werden, so dass informatische Bildungsprozesse mit dem zur Verfügung stehenden Material umgesetzt werden können.
3. Digitalen Artefakten, aber auch Unplugged-Materialien zur informatischen Bildung wohnt ein *Aufforderungscharakter* inne, der bei den Kindern Neugier und Interesse zur Auseinandersetzung mit dem Lerngegenstand wecken soll. Dies ist durch die Fach- und Lehrkräfte zu unterstützen, z. B. indem diese in eine Spielsituation eingebettet werden.
4. Die *Motivation der Kinder* sollte angesprochen werden, indem die Fach- und Lehrkräfte sich selbst an den Lerngegenständen interessiert zeigen und die Kinder mit ihrer Begeisterung für das jeweilige Thema anstecken.
5. Durch die *Schaffung von kommunikativen Situationen* soll der begründete Austausch der Kinder über Zusammenhänge, Vermutungen und Schlussfolgerungen geübt und die Reflexion über das Erlebte gestärkt werden. Hierfür ist es erforderlich, dass die Fachbegriffe der Informatik sicher, aber auch in einer altersgemäßen Form verwendet werden.
6. Basierend auf Erfahrungen mit informatischen Bildungsprozessen älterer Kinder und Jugendlicher lässt sich vermuten, dass sich mit zunehmendem Alter *geschlechtsspezifische Unterschiede* im Interesse und Umgang mit Technik und Informatik entwickeln, die im Bereich des Kindergartens und der Grundschule noch nicht existent oder weniger stark ausgeprägt sind. Diesbezüglich kommt den Fach- und Lehrkräften die Aufgabe zu, mögliche Vorurteile nicht zu bestärken, die eigenen Einstellungen und Klischees zu reflektieren und Jungen und

Mädchen gleichermaßen in den Lehr-Lern-Situationen zu motivieren und zu berücksichtigen.

### **6.1.3 Kooperation zwischen Bildungseinrichtung, Familie und Entscheidungsträgern**

Auch im Bereich der informatischen Bildung ist eine Kooperation zwischen Bildungseinrichtung und Familie als essenziell anzusehen. Es ist anzustreben, dass eine Verständigung über gemeinsame Werte und Ziele zur informatischen Bildung erreicht wird, auch wenn dies wohl eher das Ergebnis als die Voraussetzung des Bildungsprozesses sein wird. So ist davon auszugehen, dass viele Eltern unklare Vorstellungen davon besitzen, was Ziele und Potenziale informatischer Bildung in Kindergarten und Grundschule tatsächlich ausmachen. Bei den Erziehungsberechtigten kann teilweise auch Unkenntnis über Informatik bzw. ein anderweitig manifestiertes Bild der Informatik vorhanden sein, insgesamt werden vermutlich sehr verschiedene Vorstellungen von Informatik existieren. So wäre es ungünstig, wenn die Chance auf informatische Bildung vorab von elterlicher Seite aus durch eine demotivierende Haltung verbaut wird, so dass „das Kind nicht Computerspielen soll“ oder falsche Erwartungen geweckt werden („Im Kindergarten wird die Bedienung von Computern erlernt.“).

Damit einher geht auch, dass die Kinder von Hause aus sehr unterschiedliche Umgangsweisen mit digitalen Medien mitbringen, die von der Ablehnung bis hin zur extensiven Nutzung reichen (vgl. Kapitel 2.1). Es wäre demnach anzustreben, dass die Kooperation zwischen Bildungseinrichtung und Elternhaus auch die Information und Zusammenarbeit dahingehend umfasst, wie mit digitalen Medien nicht nur rezeptiv, sondern auch gestalterisch-kreativ umgegangen werden kann und wie informatische Bildung hierfür förderlich ist. Für die pädagogischen Fach- und Lehrkräfte ergibt sich ein wichtiges Aufgabenfeld, informatische Bildung in Grundschule oder Kindergarten für die Eltern transparent zu gestalten und sie in die informatischen Lernprozesse in den Bildungseinrichtungen produktiv einzu beziehen. Die transparente Gestaltung von informatischen Bildungskonzepten durch die pädagogischen Fach- und Lehrkräfte ist außerdem wichtig, um die für die Bildungseinrichtung relevanten Entscheidungsträger von diesen Konzepten zu überzeugen und sie für eine lernförderliche Gestaltung der organisatorischen Rahmenbedingungen zu gewinnen.

### **6.1.4 Organisationsbedingungen in der pädagogischen Einrichtung**

Die Organisation von Lerngelegenheiten zur informatischen Bildung kann auf unterschiedliche Art und Weise erfolgen: Erfahrungen existieren mit Gruppenarbeit, Partnerarbeit, im formellen Organisationsrahmen oder als Arbeitsgemeinschaft,

als Vorstellung bzw. Mitmach-Show (vgl. Unplugged-Shows) oder auch im Rahmen einer Ausstellung. Im Kita-Bereich sollten Lernprozesse zur informatischen Bildung in Übereinstimmung mit den dort bewährten kindgemäßen, spielerischen Arbeitsformen organisiert werden.

Für den Grundschulbereich empfiehlt es sich, informatikbezogene Unterrichtsprojekte zu organisieren. Für eine nachhaltige Förderung informatischer Bildung sollte die Schaffung eines Lernbereichs, als eigenes Fach oder als Teil des Sachunterrichts angestrebt und die Qualifizierung der Fachlehrer ermöglicht werden. Dabei ist zu berücksichtigen, dass der Ansatz der Integrierung informatischer Fachinhalte in andere Fächer zumindest im Bereich der Sekundarstufe I aus verschiedenen Gründen gescheitert ist (vgl. z. B. Breier, 2004b).

### 6.1.5 Anforderungen an die Ausstattung

Informatische Bildung kann grundsätzlich auch ohne Informatiksysteme stattfinden: So sind beispielsweise Aspekte der Zieldimensionen ‚Algorithmen‘, ‚Sprachen und Automaten‘ oder ‚Information und Daten‘ durchaus ‚unplugged‘ vermittelbar. Auch finden sich heutzutage viele Informatiksysteme im Alltag, so dass diese gemeinsam betrachtet und analysiert werden können. Dennoch bietet die Verwendung von Informatiksystemen Chancen auf Motivation und Lernerfahrungen, die ohne sie kaum möglich sind. So kann z. B. die Ausführung eines Algorithmus auf einem Informatiksystem nachvollziehbar und glaubhaft die Notwendigkeit von Eindeutigkeit und Ausführbarkeit von Algorithmen verdeutlichen. Das Erschaffen eines eigenen Spiels, das stolz Freundinnen und Freunden oder Eltern gezeigt werden kann, lässt Kinder erahnen, welche kreativen Möglichkeiten hinter der Beschäftigung mit Informatik stehen. In Kapitel 2.3 sind verschiedene Möglichkeiten skizziert worden, die konkrete Möglichkeiten zur Umsetzung informatischer Bildung mit Informatiksystemen aufzeigen. Dies setzt natürlich auch eine entsprechende sächliche Ausstattung der Bildungseinrichtungen voraus, die aber nicht als unabdingbare Voraussetzung für erfolgreiche informatische Bildung insbesondere im Kita-Bereich angesehen werden muss.

## 6.2 Messinstrumente zur Evaluation der Gelingensbedingungen

Für den Bereich der informatischen Bildung im Elementar- und Primarbereich liegen bisher keine empirisch fundierten Kompetenzmodelle vor. Daraus resultiert ein Mangel an verlässlichen Instrumenten zur Messung der von Kindern erworbenen Kompetenzen im Elementar- und Primarbereich der informatischen Bildung und, damit zusammenhängend, eine empirisch fundierte Evaluation der informa-

tischen Bildungskonzepte im Kita- und Grundschulbereich. Eine fundierte wissenschaftliche Begleitung und eine formative Evaluation der praktischen Umsetzung der vorgestellten Beispiele und fachdidaktischen Konzepte zur frühen informatischen Bildung sind ein wesentliches Element von deren erfolgreicher Implementierung in der pädagogischen Praxis von Kitas und Grundschulen.

Nicht zuletzt deshalb sollte die wissenschaftliche Begleitung der Stiftungsarbeit vielschichtig angelegt sein und weitere Forschung zur frühen informatischen Bildung angeregt werden. In Anlehnung an für die Informatik in der Sekundarstufe II und in der Lehrerbildung partiell entwickelten Kompetenzmodelle (vgl. Kapitel 4) lassen sich für die Evaluation von Lernprozessen und Gelingensbedingungen informatischer Bildung im Kita- und Grundschulbereich verschiedene Aufgabenfelder identifizieren, für die dann geeignete bereichsspezifische Messinstrumente zu entwickeln wären.

### **6.2.1 Konzeptionelle Evaluation**

Die vorgeschlagenen Maßnahmen und Beispiele sollten durch weitere Expertinnen und Experten der Informatikdidaktik sowie der frühen Bildung und Grundschulbildung einer kritischen Analyse unterzogen werden.

### **6.2.2 Materialevaluation (in Analogie zur Mathematikdidaktik)**

Die von der Stiftung „Haus der kleinen Forscher“ entwickelten Materialien zur frühen informatischen Bildung sollten einer kritischen Analyse durch eine Gruppe von Personen aus der Informatikdidaktik, aus relevanten Nachbardisziplinen sowie Vertreterinnen und Vertreter der Praxis unterzogen werden. Die regelmäßige Ausrichtung von Fachforen zur Informatik mit Fachwissenschaftlern, Fachdidaktikerinnen und Fachdidaktikern und Praktikerinnen und Praktikern, die mit der Arbeit der Stiftung vertraut sind, könnten hierzu hilfreich sein. Außerdem ist die Materialqualität durch das Wechselspiel von theoretischer Fundierung, Entwicklungsarbeit und empirischer, meist qualitativer Forschung im Paradigma von Design Research kontinuierlich zu erhöhen.

Des Weiteren sollten Forschungsprojekte zur Wirkungs- und Anwendungsforschung der Materialien etabliert werden mit den Zielgruppen:

- Kinder,
- pädagogische Fach- und Lehrkräfte sowie
- Multiplikatoren.

### 6.2.3 Maßnahmenevaluation

Einzelne informatische Bildungskonzepte im Kita- bzw. Grundschulbereich sollten evaluiert werden unter Berücksichtigung folgender Zielgruppen:

- Kinder,
- pädagogische Fach- und Lehrkräfte,
- Multiplikatoren und
- Eltern.

### 6.2.4 Wirksamkeitsforschung

Die Wirksamkeitsforschung steht in Verknüpfung mit der Maßnahmenevaluation und adressiert unter besonderer Berücksichtigung der Zielgruppen folgende Untersuchungsgegenstände:

- Kinder
  - A: Nicht-kognitive Kompetenzen (Motivation, Interesse, Selbstwirksamkeit, Einstellungen) → hier gibt es beispielsweise schon Studien bezüglich geschlechtlicher Unterschiede (vgl. Kapitel 2).
  - B: Spezifische fachliche, auf ein Unterrichtsmodul bezogene Kompetenzen (modulbezogene Analyse der Lernergebnisse).
  - C: Allgemeinbildende informatische Kompetenzen → z. B. erkennen von informatischen Phänomenen, erkennen von Informatiksystemen und ihrer Funktionalität etc.
  - D: Bewertung der Lernprozesse (z. B. im Hinblick auf die Überwindung von Lernbarrieren/Transition von Lernkonzepten, etwa von ‚unplugged‘ zu plugged).
- Pädagogische Fach- und Lehrkräfte:
  - A: Einstellungen/Curriculare beliefs  
Die Bedeutung von Einstellungen, Haltungen und des beruflichen Rollenverständnisses von Lehrkräften auf die eigene pädagogische Praxis ist seit Langem in verschiedenen empirischen Studien nachgewiesen worden (vgl. Kapitel 4 (z. B. Eulenberger, 2015)). Für Informatiklehrkräfte hat Ni (2011) in ihrer Untersuchung zur professionellen Identität von ‚CS Teachern‘ den Einfluss der ‚beliefs‘ auf die Unterrichtspraxis spezifiziert. Sie ermittelte u. a. die beiden generellen Einflussfaktoren ‚Percep-

tion of CS' und ‚Perception of Teaching‘. Zusätzlich spielt (ähnlich wie auf der Ebene der Kinder, s. o.) die Einstellung bzw. generelle Akzeptanz von Technik eine Rolle für den Unterricht (etwa Gil et al., 2007). Hier können ebenfalls vorhandene Instrumente angepasst werden (z. B. TAM 2.0 (Bagozzi, 2007) oder INCOBI-R (Richter & Naumann, 2010; Richter, Naumann & Groeben, 2001).

Mögliche Erhebungsdimensionen in diesem Bereich sind:

- Einstellungen gegenüber Informatiksystemen;
  - Einsatz von Informatiksystemen im Alltag und in der Schule;
  - Einstellungen zu Informatik in der Grundschule;
  - Curricular beliefs/Curricular emphasis.
- B: Fachdidaktische Kompetenzen
- Um die fachdidaktischen Kompetenzen bzw. den Kompetenzerwerb der pädagogischen Fach- und Lehrkräfte zu evaluieren, sollten auch bei dieser Zielgruppe Kompetenzmessungen vorgenommen werden. Die Items der Testinstrumente könnten zielgruppenbezogen aus theoretisch und empirisch fundierten Kompetenzmodellen zur Informatiklehrerbildung adaptiert, aus dort bereits vorhandenen Messinstrumenten abgeleitet werden (vgl. z. B. KUI, CoRe in Kapitel 4; Bender et al., 2015, 2016; Buchholz et al., 2013; Linck et al., 2013; Magenheim et al., 2010; Neugebauer et al., 2015; Williams & Lockley, 2012) und sich auf fachliche, fachdidaktische, motivationale und volitionale Kompetenzaspekte beziehen (Weinert, 2001). Dies schließt die Evaluation der praxisbezogenen Handlungskompetenz in informatikbezogenen Lernsituationen durch Beobachtungen mit ein.

■ Eltern:

Wie bereits in Kapitel 6.1.3 beschrieben, haben Eltern und familiales Umfeld der Kinder vermutlich einen nicht unerheblichen Einfluss auf die Einstellungen der Kinder gegenüber Informatiksystemen. Daher ist es sinnvoll, Indikatoren des informatikbezogenen familiären Umfelds der Kinder und die Einstellungen der Eltern zu Informatik und Informatiksystemen als relevante Faktoren der Gelingensbedingungen informatischer Bildung im Kita- und Grundschulbereich mit zu erheben. Konkret umfasst dies die folgenden Aspekte:

- Nutzungsmöglichkeiten:  
Informations- und Kommunikationssysteme im häuslichen Umfeld: Existenz, Nutzungsformen und -häufigkeiten,

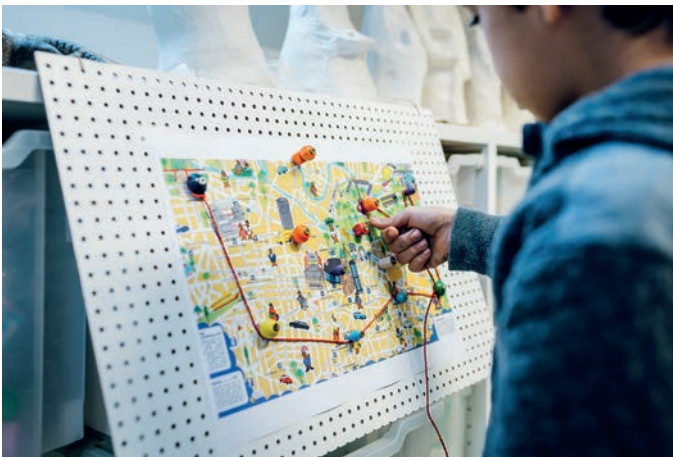


- Einstellungen:
  - Einstellungen gegenüber Informatiksystemen;
  - Einstellungen zum Einsatz von Informatiksystemen im Alltag und im Umfeld der Kinder;
  - Einstellungen zu Informatik in der Grundschule.

Es sollte mit ähnlichen, z. T. adaptierten Instrumenten wie bei den Einstellungen der Lehrkräfte untersucht werden, ob die Einstellungen der Eltern und die häusliche Nutzung von Informatiksystemen einen Einfluss auf den informatischen Kompetenzerwerb der Kinder und deren Einstellungen zu Informatik und Informatiksystemen haben.

## 7 Fazit

Mit dieser Expertise hat das Autorenteam sowohl aus der Perspektive fachdidaktischer Forschung als auch unter Gesichtspunkten pädagogischer Praxis der informatischen Bildung im Kita- und Grundschulbereich Neuland betreten. Das hat mehrere Ursachen, u. a. die im Vergleich zu anderen etablierten Fächern noch wenig ausgeprägte fachdidaktische Forschung zur informatischen Bildung sowie die vergleichsweise schwache Verankerung der informatischen Bildung als Unterrichtsfach im Sekundar- und Grundschulbereich, vor allem an deutschen Schulen. Vor diesem Hintergrund hat das Autorenteam den Versuch unternommen, wichtige Ziele und Gelingensbedingungen für eine erfolgreiche informatische Bildung im Kita- und Grundschulbereich herauszuarbeiten. Dazu wurde zunächst der Begriff informatische Bildung, seine Beziehung zur Fachwissenschaft Informatik und die Gemeinsamkeiten und Unterschiede zur digitalen (Medien-)Bildung charakterisiert. Ferner wurden bereits existierende, vor allem internationale Ansätze zur informatischen Bildung im Kita- und Grundschulbereich sowie hierfür geeignete Lernumgebungen und Softwaretools im Hinblick auf ihre Verwendbarkeit in der frühen informatischen Bildung analysiert. Einschlägige Curricula wurden zwecks Vergleichbarkeit hinsichtlich ihrer Bezüge zu informatischen Bildungsstandards bewertet und auf diese Weise ein empirisch zumindest teilweise begründetes Kategoriensystem zur Beschreibung von Kompetenzfeldern informatischer Bildung entwickelt. Aus dieser Analyse existierender internationaler Curricula zur frühen informatischen Bildung und aufgrund von einschlägigen Konzepten verschiedener informatikdidaktischer Ansätze wurde im Vergleich zu den GI-Bildungsstandards eine neue Prozesskategorie ‚Interagieren und Explorieren‘ mit und von Informatiksystemen eingeführt. Deren Bedeutung für den spielerisch erkundenden



Umgang mit Informatiksystemen im Kita- und Grundschulbereich wurde ausführlich erörtert. Hierzu wurde auf einschlägige Studien zu den erforderlichen kognitiven Voraussetzungen und Entwicklungen der Kinder Bezug genommen, die zum Erwerb von grundlegenden Kompetenzen informatischer Bildung als relevant eingeschätzt werden. Einstellungen, Motivation und beliefs von pädagogischen Fach- und

Lehrkräften gegenüber Informatik und informatischer Bildung wurden als wichtige Voraussetzungen für den Erwerb fachdidaktischer Kompetenzen im Bereich früher informatischer Bildung beschrieben. Insgesamt wurden gut qualifizierte pädagogische Fach- und Lehrkräfte als ein wesentlicher Faktor für die erfolgreiche Umsetzung informatischer Bildungskonzepte im Kita- und Grundschulbereich charakterisiert. Schließlich wurden als ein Fazit dieser Ausführungen wichtige Kriterien für die Konstruktion von empirischen Messinstrumenten zur Evaluation von Gelingensbedingungen und von pädagogischer Praxis früher informatischer Bildung hergeleitet. Die praktische Erprobung der Beispiele und der vorgestellten Konzepte sowie eine begleitende formative Evaluation werden von den Autoren dabei selbst als ein wesentlicher Faktor für die erfolgreiche Implementierung früher informatischer Bildung in der Praxis von Kitas und Grundschulen angesehen.

Das Autorenteam hofft, mit dieser Expertise die wichtigen Aktivitäten der Stiftung „Haus der kleinen Forscher“ in der frühen informatischen Bildung unterstützen zu können und damit einen Beitrag zur Implementierung von informatischer Bildung im Kita- und Grundschulbereich zu leisten.

## 8 Anhang

### Abbildung der Komponenten internationaler Standards und Curricula in den Ordnungsrahmen eines Kompetenzmodells für informatische Bildung in der Primarstufe

Im Folgenden sind die Originalformulierungen der Kompetenzen der internationalen Standards und Curricula den Inhaltsbereichen des in Kapitel 3 diskutierten Kompetenzmodells zugeordnet, um einerseits die Interpretation durch das Autorenteam zu belegen und andererseits die in den internationalen Ansätzen vorgenommenen Schwerpunktsetzungen darzustellen.

Dabei ist auf die Prozessbereiche, wie in Abschnitt 2.5.2 dargelegt, eher indirekt zu schließen, während die Inhaltsbereiche explizit benannt werden.

#### (I<sub>1</sub>) Information und Daten

*CAS (Computing at School; Großbritannien)*

Key stage 1 (Vorschule–2. Klasse, Alter 5–7)

- A pupil should **understand** how computers **represent data** (CAS, S.16, Data).
- **Information** can be **stored** and **communicated** in a variety of forms e.g. numbers, text, sound, image, video (CAS, S.16, Data).
- Computers use **binary** switches (on/off) to store information (CAS, S.16, Data).
- Binary (yes/no) answers can directly provide useful information (e. g. present or absent), and be **used for decision** (CAS, S.16, Data).

Key stage 2 (3.–6. Klasse, Alter 7–11)

- Similar information can be represented in multiple ways (CAS, S.16, Data).
- Introduction to **binary representation** [representing **names, objects** or **ideas** as sequences of 0s and 1s] (CAS, S.16, Data).
- Difference between **data** and **information** (CAS, S.16, Data).

- Structured data can be stored in **tables** with rows and columns. Data in tables can be **sorted**. Tables can be **searched** to answer **questions**. Searches can use one or more columns of the table (CAS, S.16, Data).
- Data may contain errors and that this affects the search results and decisions based on the data. **Errors** may be reduced using **verification** and **validation** (CAS, S. 16, Data).
- A pupil should understand the principles underlying how **data** is transported on the **Internet** (CAS, S.18, Communication and the Internet).

*CSTA (Computer Science Teachers Association; USA)*

Level 1 (Vorschule–3. Klasse, Alter 5–8)

- **Demonstrate** how 0s and 1s can be used to **represent information** (CSTA S.13, L1:3.CT 5.).
  - Prozessbereiche P5 Darstellen & Interpretieren, ggf. auch P1 Modellieren oder P2 Begründen bzw. P3 Strukturieren

Level 2 (3.–6. Klasse, Alter 8–11)

- Demonstrate how a string of bits can be used to **represent alphanumeric information** (CSTA S.13, L1:6. CT 3.).
  - vor allem Prozessbereich P5 Darstellen & Interpretieren
- **Gather and manipulate** data using a variety of digital **tools** (CSTA S.14, L1:6. CPP 10.).
  - P5 und P1

*Neuseeland*

Level 1 (1.–3. Klasse, Alter 5–7)

- Konservativ: nicht enthalten
- Fortgeschritten: How 0s and 1s represent information; patterns and symbols; pixels and file size.
  - P5

Level 2 (3.–5. Klasse, Alter 7–9)

- How two different symbols can **represent information** (e. g. binary numbers).
  - P5 (Beispiele darstellen, Darstellungen interpretieren), ggf. P1 (geeignete Repräsentation für Problemstellung finden => Modellieren), ggf. P2

(aus möglichen Repräsentationen für Problemstellung auswählen=> Begründen, Bewerten)

- **Representation of text and images** using binary; codes and symbols.
  - P1 oder P5, ggf. P2 und P3

#### *Schweizer Lehrplan 21*

Die Schülerinnen und Schüler können **Daten** aus ihrer Umwelt **darstellen** (P1, P5), **strukturieren** (P3) und **auswerten** (P5) (Ml.2.1, S.14).

1.–2. Klasse, Alter 6–7

- Schülerinnen und Schüler können Dinge nach selbst gewählten Eigenschaften **ordnen** (P3), damit sie ein Objekt mit einer bestimmten Eigenschaft schneller **finden** (P5) (z. B. Farbe, Form, Größe).

3.–4. Klasse, Alter 8–9

- Schülerinnen und Schüler können unterschiedliche **Darstellungsformen** für Daten verwenden (P5) (z. B. Symbole, Tabellen, Grafiken).
- Schülerinnen und Schüler können Daten mit Hilfe **selbstentwickelter Geheimschriften** verschlüsseln (P5).
- Schülerinnen und Schüler kennen **analoge** und **digitale Darstellungen** von Daten (Text, Zahl, Bild und Ton) und können die entsprechenden Dateitypen **zuordnen** (P3).

4.–6. Klasse, Alter 9–11

- Schülerinnen und Schüler **erkennen** und **verwenden** (P1, P3, P5) **Baum-** und **Netzstrukturen** (z. B. Ordnerstruktur auf dem Computer, Stammbaum, Mindmap, Website).
- Schülerinnen und Schüler **verstehen** (P2, P5) die Funktionsweise von **fehlererkennenden** und **-korrigierenden Codes**.

### **(I2) Algorithmen (und Programmierung)**

CAS

Key stage 1 (Vorschule–2. Klasse, Alter 5–7)

- Algorithms are **sets of instructions** for achieving goals, made up of **pre-defined steps** [the ‘how to’ part of a recipe for a cake] (CAS, S.13, Algorithms).

- Algorithms can be **represented in simple formats** [storyboards and narrative text] (CAS, S.13, Algorithms).
- They can describe **everyday activities** and can be **followed by humans** and **by computers** (CAS, S.13, Algorithms).
- A pupil should **understand what an algorithm is**, and what algorithms can be **used for** (CAS, S.13, Algorithms).
- **Steps** can be **repeated** and some steps can be **made up of smaller steps** (CAS, S.13, Algorithms).

Key stage 2 (3.–6. Klasse, Alter 8–11)

- Algorithms can be **represented symbolically** [flowcharts] or using instructions in a clearly defined **language** [turtle graphics] (CAS, S.13, Algorithms).
- Algorithms can include **selection (if)** and **repetition (loops)** (CAS, S.13, Algorithms).
- Algorithms may be **decomposed** into component parts (**procedures**), each of which itself contains an algorithm (CAS, S.13, Algorithms).
- Algorithms should be stated **without ambiguity** and care and **precision** are necessary to avoid errors (CAS, S.13, Algorithms).
- Programs can include **repeated instructions** (CAS, S. 14, Programs).
- A computer **program** is a sequence of instructions written to perform a specified **task** with a computer (CAS, S.14, Programs).
- The idea of a program as a **sequence** of statements written in a **programming language** [Scratch] (CAS, S.14, Programs).
- One or more mechanisms for **selecting** which **statement sequence** will be executed, **based upon** the **value** of some data item (CAS, S.14, Programs).
- One or more mechanisms for **repeating** the execution of a sequence of statements, and using the **value** of some data item **to control the number** of times the sequence is repeated (CAS, S.14, Programs).

- Programs can be created using **visual tools**. Programs can work with different types of data. They can use a **variety of control structures** [selections and procedures] (CAS, S.15, Programs).
- Programs are **unambiguous** and that care and precision is necessary to avoid errors (CAS, S. 15, Programs).
- Programs are **developed according to a plan** and then **tested**. Programs are corrected if they fail these tests (CAS, S.15, Programs).
- The **behaviour** of a program should be **planned** (CAS, S.15, Programs).
- A well-written program tells a reader the story of how it works, both in the code and in **human-readable comments** (CAS, S.15, Programs).
- A web page is an **HTML** script that constructs the **visual appearance**. It is also the carrier for other **code** that can be **processed** by the browser (CAS, S.15, Programs).
- Computers can be **programmed** so they appear to **respond ‘intelligently’ to certain inputs** (CAS, S.15, Programs).
- The difference between **constants** and **variables** in programs (CAS, S.16, Data).

#### CSTA

Level 1 (Vorschule–3. Klasse, Alter 5–8)

- **Use technology resources** (e. g., puzzles, logical thinking programs) to solve age-appropriate problems (CSTA S.13, L1:3.CT 1.).
  - P0, P1, ggf. P3 und P5
- Understand how to **arrange (sort) information** into useful order, such as sorting students by birth date, without using a computer (CSTA S.13, L1:3.CT 3.)
  - P0, P1, P3, P5; ggf. P2 und P4 Kommunizieren & Kooperieren.
- **Construct** a set of **statements (P1)** to be acted out to accomplish a simple task (e. g., turtle instructions) (CSTA S.14, L1:3. CPP 4.).



## Level 2 (3.–6. Klasse, Alter 8–11)

- Understand and use the **basic steps in algorithmic problem-solving** (e. g., problem statement and exploration, examination of sample instances, design, implementation, and testing) (CSTA S. 13, L1:6.CT 1.).
  - P0, P1, P5, ggf. P2, P3, P4
- Develop a **simple understanding of an algorithm** (P1, (e. g., search, sequence of events, or sorting) using computer-free exercises (CSTA S.13, L1:6. CT 2.).
  - P1, P5, ggf. P2, P3, P4
- Make a **list of sub-problems** (P3, P1) to consider while addressing a larger problem (CSTA S.13, L1:6. CT 5.).
- **Construct a program** (P1, P5) as a set of **step-by-step instructions** to be acted out (e. g., make a peanut butter and jelly sandwich activity) (CSTA S.14, L1:6. CPP 5.).

*Neuseeland*

## Level 1 (1.–3. Klasse, Alter 5–7)

- Understand **what algorithms are** (P2, P5) and **follow an algorithm** (P5).
- **Sorting** (P1, P5) and **patterns** (P3).
- **Create** and **debug** (P1, P5, ggf. P3) simple programs (e. g. turtle instructions) with simple **sequencing** and **repetition**.

## Level 2 (3.–5. Klasse, Alter 7–9)

- **Decompose** (P1, P5) problems into steps.
- **Explain** (P5, ggf. P2) and **correct** (P1) **errors** in algorithms.
- Block-based/visual programming including simple **iteration**.
- Design, implement, test and debug (P1 inklusive P3 und P5, ggf. P2 und P4) an interactive application in a visual programming language (sequence and **selection; input**).

*Schweizer Lehrplan 21*

Die Schülerinnen und Schüler können einfache Problemstellungen **analysieren** (P2/P3/P5), mögliche **Lösungsverfahren beschreiben** (P5) und in Programmen **umsetzen** (P1).

1.–2. Klasse, Alter 6–7

- Schülerinnen und Schüler können **formale Anleitungen erkennen** und ihnen **folgen** (P5, AFB I-II) (z. B. Koch- und Backrezepte, Spiel- und Bastelanleitungen, Tanzchoreographien).

3.–4. Klasse, Alter 8–9

- Schülerinnen und Schüler können durch **Probieren Lösungswege** (P1, P5) für einfache Problemstellungen suchen und auf **Korrektheit prüfen** (P1, P5) (z. B. einen Weg suchen, eine Spielstrategie entwickeln). Sie können verschiedene **Lösungswege vergleichen** (P2).
- Schülerinnen und Schüler können **Abläufe mit Schleifen und Verzweigungen** aus ihrer Umwelt **erkennen** (P3), beschreiben und **strukturiert darstellen** (P5) (z. B. mittels Flussdiagrammen).
- Schülerinnen und Schüler können einfache Abläufe mit Schleifen, bedingten Anweisungen und Parametern **lesen** und **manuell ausführen** (P5, ggf. P4).
- Schülerinnen und Schüler **verstehen** (P2, ggf. P5), dass ein Computer **nur vordefinierte Anweisungen** ausführen kann und dass ein Programm eine Abfolge von solchen Anweisungen ist.
- Schülerinnen und Schüler können **Programme** mit Schleifen, bedingten Anweisungen und Parametern schreiben und testen (P1+Testen).

**(I3) Sprachen und Automaten**

CAS

Key stage 1 (Vorschule–2. Klasse, Alter 5–7)

- Computers need more **precise instructions** than humans do (CAS, S.13, Algorithms).
- Computers (understood here to include all devices controlled by a processor, thus including programmable toys, phones, game consoles and PCs) are controlled by **sequences of instructions** (CAS, S.14, Programs).

- A computer program is like the narrative part of a story, and the computer's job is to do what the narrator says. Computers have **no intelligence**, and so **follow** the narrator's **instructions blindly** (CAS, S.14, Programs).

Key stage 2 (3.–6. Klasse, Alter 7–11)

- Programs can model and **simulate environments** to answer “**What if**” questions.(CAS, S.14, Programs).
  - Dies könnte auch in Form von Automaten dargestellt werden.
- Programs are **unambiguous** and that care and precision is necessary to avoid errors (CAS, S.15, Programs).
- A web page is an **HTML** script that constructs the **visual appearance**. It is also the carrier for other **code** that can be **processed** by the browser (CAS, S.15, Programs).
- The format of URLs (CAS, S.19, Communication and the Internet).

*CSTA*

Level 1 (Vorschule–3. Klasse, Alter 5–8)

- Construct a set of statements to be acted out to accomplish a simple task (e. g., **turtle instructions**) (CSTA S.14, L1:3. CPP 4.).

*Neuseeland*

Level 1 (1.–3. Klasse, Alter 5–7)

- k. A.

Level 2 (3.–5. Klasse, Alter 7–9)

- k. A.

*Schweizer Lehrplan 21*

3.–4. Klasse, Alter 8–9

- Schülerinnen und Schüler **verstehen** (P2, ggf. P5), dass ein Computer **nur vordefinierte Anweisungen** ausführen kann und dass ein Programm eine Abfolge von solchen Anweisungen ist.

#### (I4) Informatiksysteme

CAS

Key stage 1 (Vorschule–2. Klasse, Alter 5–7)

- Particular **tasks** can be accomplished by creating a program for a computer. Some computers allow their users to create their own programs (CAS, S.14, Programs).
- Computers typically accept **inputs**, follow a stored sequence of **instructions** and produce **outputs** (CAS, S.14, Programs).
- A pupil should know the main **components** that make up a **computer** system, and how they fit together (their **architecture**) (CAS, S.17, Computers).
- **Computers** are electronic devices using stored sequences of instructions (CAS, S.17, Computers).
- Computers typically accept input and produce outputs, with examples of each in the context of PCs (CAS, S.17, Computers).
- Many devices now contain computers (CAS, S. 17, Computers).
- Web **browser** is a program used to use view pages (CAS, S.18, Communication and the Internet).

Key stage 2 (3.–6. Klasse, Alter 8–11)

- Computers are devices for executing programs (CAS, S.17, Computers).
- Application software is a computer program designed to perform user tasks (CAS, S.17, Computers).
- The **operating system** is a software that manages the relationship between the application software and the hardware (CAS, S.17, Computers).
- Computers consist of a number of **hardware components** each with a specific role [e. g. CPU, Memory, Hard disk, mouse, monitor] (CAS, S.17, Computers).
- Both the operating system and application software **store data** (e. g. in memory and a file system) (CAS, S.17, Computers).

- The above applies to devices with embedded computers (e. g. digital cameras), handheld technology (e. g. smart phones) and personal computers (CAS, S. 18, Computers).
- A **variety of operating systems** and application software is typically available for the same hardware (CAS, S. 18, Computers).
- Users can **prevent or fix problems** that occur with computers (e. g. connecting hardware, protection against viruses) (CAS, S. 18, Computers).
- A pupil should understand the principles underlying how **data** is transported on the **Internet** (CAS, S.18, Communication and the Internet).
- The **Internet** is a collection of computers connected together sharing the same way of communicating. The internet is not the web, and the web is not the internet (CAS, S.18, Communication and the Internet).
- These connections can be made using a **range of technologies** (e. g. network cables, telephone lines, wifi, mobile signals, carrier pigeons) (CAS, S.19, Communication and the Internet).
- The Internet supports multiple **services** (e. g. the Web, e-mail, VoIP) (CAS, S.19, Communication and the Internet).
- The relationship between web **servers**, web **browsers**, **websites** and web **pages** (CAS, S.19, Communication and the Internet).
- The role of **search engines** in allowing users to find specific web pages and a basic understanding of how results may be ranked (CAS, S.19, Communication and the Internet).
- Issues of **safety and security** from a technical perspective (CAS, S.19, Communication and the Internet).

#### CSTA

Level 1 (Vorschule–3. Klasse, Alter 5–8)

- **Use writing tools**, digital **cameras**, and **drawing** tools to illustrate thoughts, ideas, and stories in a step-by-step manner (CSTA S.13, L1:3.CT 2.).

- **Recognize** that software is created to control computer operations (CSTA S.13, L1:3. CT 4.).
- Use standard **input and output devices** to successfully operate computers and related technologies (CSTA S.14, L1:3. CD 1.).
- Use technology resources to conduct age-appropriate **research** (CSTA S.14, L1:3. CPP 1.).
- Use developmentally appropriate multimedia resources (e. g., interactive books and educational software) to support **learning** across the curriculum (CSTA S.14, L1:3. CPP 2.).
- Create developmentally appropriate **multimedia products** with support from teachers, family members, or student partners (CSTA S.14, L1:3. CPP 3.).

#### Level 2 (3.–6. Klasse, Alter 8–11)

- Demonstrate an appropriate level of proficiency with **keyboards** and other **input and output devices** (CSTA S.14, L1:6. CD 1.).
- Apply strategies for **identifying** simple **hardware and software problems** that may occur during use (CSTA S.14, L1:6. CD 3.).
- Identify that **information** is coming to the computer from many **sources over a network** (CSTA S.14, L1:6. CD 4.).
- Describe how a **simulation** can be used to **solve a problem** (CSTA S.13, L1:6. CT 4.).
- Use technology resources (e. g., calculators, data collection probes, mobile devices, videos, educational software, and web tools) for **problem-solving** and **self-directed learning** (CSTA S.14, L1:6. CPP 1.).
- Use general-purpose productivity tools and peripherals to support personal productivity, remediate skill deficits, and facilitate learning (CSTA S.14, L1:6. CPP 2.).

- Use technology tools (e. g., multimedia and text authoring, presentation, web tools, digital cameras, and scanners) for individual and collaborative **writing, communication, and publishing** activities (CSTA S.14, L1:6. CPP 3.).
- Use computing devices to **access** remote information, **communicate** with others in support of direct and **independent learning**, and pursue personal interests (CSTA S.14, L1:6. CPP 7.).
- **Navigate** between webpages using hyperlinks and conduct simple **searches** using search engines (CSTA S.14, L1:6. CPP 8.).

### *Neuseeland*

Level 1 (1.–3. Klasse, Alter 5–7)

- **Use** (P0) common input/output **devices** (e.g. keyboard, pointing device, touch screen).
- **Describe** (P3) hardware and software components; input/output devices.
- Create, organise, store, manipulate and retrieve **content** (P0) including **multi-media** (P0).

Level 2 (3.–5. Klasse, Alter 7–9)

- Peripheral devices; data capture; data transfer (P0).
- Identify simple **software problems** (P3, P5).
- Use **search** technology (P0).
- **Select, use and combine software** (P3, ggf. P2) to collect, organise and present **data**; simple **spreadsheets** and **charts**.

### *Schweizer Lehrplan 21*

Die Schülerinnen und Schüler verstehen **Aufbau** und **Funktionsweise** von informationsverarbeitenden Systemen und können Konzepte der **sicheren Datenverarbeitung** anwenden.

1.–2. Klasse, Alter 6–7

- Schülerinnen und Schüler können **Geräte** ein- und ausschalten, **Programme** starten, **bedienen und beenden** sowie einfache **Funktionen** nutzen (P0).

- Schülerinnen und Schüler können sich mit eigenem **Login** in einem lokalen Netzwerk oder einer Lernumgebung anmelden (P0).
- Schülerinnen und Schüler können **Dokumente** selbstständig **ablegen** und **wiederfinden** (P0).
- Schülerinnen und Schüler können mit grundlegenden Elementen der **Bedienoberfläche** umgehen (Fenster, Menü, mehrere geöffnete Programme) (P0).

#### 3.–4. Klasse, Alter 8–9

- Schülerinnen und Schüler können **Betriebssystem** und **Anwendungssoftware unterscheiden** (P0, P1).
- Schülerinnen und Schüler kennen verschiedene **Speicherarten** (z. B. Festplatten, Flashspeicher, Hauptspeicher) und deren Vor- und Nachteile und verstehen **Größeneinheiten** für Daten (P0, P2).
- Schülerinnen und Schüler können bei **Problemen** mit Geräten und Programmen **Lösungsstrategien** anwenden (z. B. Hilfe-Funktion, Recherche) (P0, P3).
- Schülerinnen und Schüler können erklären, wie **Daten verloren** gehen können und kennen die wichtigsten **Maßnahmen**, sich davor zu schützen (P0).

#### 4.–6. Klasse, Alter 9–11

- Schülerinnen und Schüler verstehen die grundsätzliche Funktionsweise von **Suchmaschinen** (P0).
- Schülerinnen und Schüler können **lokale** Geräte, lokales **Netzwerk** und das Internet als **Speicherorte** für private und öffentliche Daten unterscheiden (P0, P2).
- Schülerinnen und Schüler haben eine Vorstellung von den **Leistungseinheiten** informationsverarbeitender Systeme und können deren Relevanz für konkrete Anwendungen einschätzen (z. B. Speicherkapazität, Bildauflösung, Rechenkapazität, Datenübertragungsrate) (P0, P2).



## (15) Informatik, Mensch und Gesellschaft

### CAS

Key stage 2 (3.–6. Klasse, Alter 7–11)

- **Personal information** should be accurate, stored **securely**, used for **limited purposes** and treated with respect (CAS, S.16, Data, Key stage 2).
  - P2, P3
  
- Social and **ethical issues** raised by the role of computers in our lives (CAS, S.18, Computers, Key stage 2).
  - P3, P2, ggf. P4 & P5

### CSTA

Level 1 (Vorschule–3. Klasse, Alter 5–8)

- Practice responsible digital citizenship (legal and **ethical** behaviors) in the use of technology systems and software (CSTA S.15, L1:3.CI 1.).
  
- Identify **positive** and **negative** social and **ethical** behaviors for using technology (CSTA S.15, L1:3.CI 2.).
  - P3, P2, ggf. P4
  
- **Identify jobs** that use computing and technology (CSTA S.14, L1:3. CPP 5.).

Level 2 (3.–6. Klasse, Alter 8–11)

- Understand the **connections** between computer science and **other fields**. (CSTA S.13, L1:6. CT 6.).
  - P2, P3
  
- Identify a wide range of jobs that require knowledge or use of computing (CSTA S.14, L1:6. CPP 9.).
  - P3
  
- Understand the **pervasiveness** of computers and computing in daily life (e. g., voice mail, downloading videos and audio files, microwave ovens, thermostats, wireless Internet, mobile computing devices, GPS systems) (CSTA S.14, L1:6. CD 2.).
  - P3

- Identify factors that **distinguish humans from machines** (CSTA S.14, L1:6. CD 5.).
  - P3, ggf. P2
- Recognize that computers **model intelligent behavior** (as found in robotics, speech and language recognition, and computer animation) (CSTA S.15, L1:6. CD 6.).
  - P3, ggf. P2
- Discuss basic issues related to **responsible use** of technology and information, and the consequences of inappropriate use (CSTA S.15, L1:6. CI 1.)
  - P3, P2, ggf. P4
- Identify the **impact** of technology (e.g., social networking, cyber bullying, mobile computing and communication, web technologies, cyber security, and virtualization) on **personal life and society** (CSTA S.15, L1:6.CI 2.).
  - P3, P2, ggf. P4
- **Evaluate** the accuracy, relevance, appropriateness, comprehensiveness, and biases that occur in electronic information sources (CSTA S.15, L1:6.CI 3.).
  - P2
- Understand ethical issues that relate to computers and networks (e. g., equity of access, security, privacy, copyright, and intellectual property) (CSTA S.15, L1:6.CI 4.).
  - P2, ggf. P3

### *Neuseeland*

Level 1 (1.–3. Klasse, Alter 5–7)

- **Ergonomics**, digital devices in everyday life; **ethical and safe use** (P0).
- Use technology safely and respectfully, **keeping** personal information **private** (P0).
- Identify **where to go for help** and support when they have concerns about content or contact on the internet or other online technologies (P0, P4).
- **Recognise** use beyond school (P0, P3).

Level 2 (3.–5. Klasse, Alter 7–9)

- **Use** safely, respectfully and responsibly; appropriate behaviour; reporting concerns (P0, P2).
  
- How systems meet community and personal needs; **evaluate adequacy** of a solution (P0, P2, P3).
  
- **Collaboratively plan creation** and communication of information (P4).
  
- **Ethical decisions** and behaviour; social media (P2, P3).

*Schweizer Lehrplan 21*

Die Schülerinnen und Schüler verstehen **Aufbau** und **Funktionsweise** von informationsverarbeitenden Systemen und können Konzepte der **sicheren Datenverarbeitung** anwenden.

4.–6. Klasse, Alter 9–11

- Schülerinnen und Schüler verstehen die grundsätzliche Funktionsweise von **Suchmaschinen**.
  
- Schülerinnen und Schüler können **lokale** Geräte, lokales **Netzwerk** und das Internet als **Speicherorte** für private und öffentliche Daten unterscheiden.

# Fachempfehlung Informatiksysteme

Nadine Bergner, Kathrin Müller



## Einführung

- 1 Überblick über mögliche Informatiksysteme
- 2 Beschreibung und fachliche Einschätzung der einzelnen Informatiksysteme
- 3 Empfehlungen
- 4 Fazit

## Einführung

Informatiksysteme spielen in unserer digitalen Welt eine immer größere Rolle. Dabei geht es schon lange nicht mehr ausschließlich um große Rechenanlagen, die von Fachleuten bedient werden, sondern um unzählige kleine und größere Alltagsgeräte wie Smartphone, Waschmaschine oder Auto. Teilweise sind diese Systeme eindeutig als Informatiksysteme zu erkennen (wie bei einem Laptop oder Smartphone), aber in einigen Fällen (wie bei Kaffeemaschinen) ist nicht immer direkt erkennbar, dass sich auch hier ein Informatiksystem versteckt. Bereits einige (Spiel-)Geräte für Kleinkinder (z. B. Babyphone und sprechende Puppen) beinhalten ein vollständiges Informatiksystem. An diesen Beispielen wird deutlich, dass Kinder mit Informatiksystemen in Kontakt kommen, lange bevor sie einmal vor einem Computer sitzen. Das wohl verbreitetste und bereits sehr komplexe Informatiksystem ist das Smartphone. Hier kann man beobachten, dass bereits Kinder im Kleinkindalter vollkommen intuitiv mit den Geräten umgehen, obwohl diese in erster Linie für Erwachsene entwickelt wurden. Daher hat die Generation auch ihren Namen: digital Natives.

Warum aber gerade für diese Generation eine fundierte informatische Bildung entscheidend ist, wird schnell klar, wenn die intuitive Bedienung nicht ziel führend geschieht. Kinder vollziehen häufig Abläufe nach, die sie bei den Eltern beobachtet haben; sie erkennen aber nicht die dahinterliegende Logik, verstehen beispielsweise nicht, ob sie Dateien lokal auf einem Gerät oder verteilt im Internet speichern, was öffentlich und privat ist, wie die Vernetzung von Geräten funktioniert und wie dies unser Leben beeinflusst.

Darum sollten Kinder auch bereits in Kita und Grundschule erfahren, was es bedeutet, mit Informatiksystemen umzugehen. Sie sollten digitale Geräte nicht nur nutzen, sondern auch verstehen, dass diese Geräte das machen, was man ihnen „sagt“. Daher muss neben dem „Wie“ auch immer das „Warum“ vermittelt werden. Für das „Warum“ ist es unerlässlich, ein Grundverständnis über informatische Systeme und ihre Funktionsweise zu besitzen. Das Hauptziel für Kinder im Kita- und Grundschulalter ist es, erste Erfahrungen im Bereich der informatischen Bildung zu sammeln, um perspektivisch ein Grundlagenverständnis von Informatik(systemen) zu entwickeln. Nur so können die Kinder später einmal mündige Mitgestalterinnen und Mitgestalter (und nicht nur passive Nutzer) unserer digital geprägten Lebensumwelt werden.

## 1 Überblick über mögliche Informatiksysteme

Inzwischen existieren zahlreiche Informatiksysteme für den Einsatz im Elementar- und Primarbereich. Diese Systeme können auf unterschiedliche Art und Weise kategorisiert werden. Zum einen existieren Systeme, die autonom verwendet werden können, dazu gehören beispielsweise der Cubetto, der Bee-Bot oder der KIBO (nähere Informationen zu den Systemen folgen im nächsten Kapitel). Andere Systeme können zusätzlich in Verbindung mit einer App, wie z. B. der Ozobot, oder dem Computer, wie z. B. LEGO WeDo, verwendet werden. Systeme wie die Sprachen Scratch oder ScratchJr (Scratch Junior) benötigen neben einem Tablet oder einem Computer keine zusätzliche Hardware.

Aber für welches Alter ist welches System angemessen? Welche Vorerfahrungen bedarf es seitens der Kinder, aber auch seitens der Lernbegleitungen? Und welche Ziele im Bereich der informatischen Bildung lassen sich mit welchem System am besten verfolgen?

Diesen Fragen widmet sich die vorliegende Fachempfehlung und versucht, den aktuellen Stand zu den vorhandenen Informatiksystemen für Kinder zu geben. Die Einschätzungen zur Nutzerfreundlichkeit, Vorerfahrungen etc. basieren auf unseren Erfahrungen sowie Herstellerinformationen.

Die im Folgenden beschriebenen Informatiksysteme sind nach Schwierigkeitsgrad bzw. Komplexität geordnet.

*Systeme mit geringer Komplexität:* Diese Systeme können in der Regel autonom verwendet werden und benötigen häufig eine geringe informatische Kompetenz der Lernbegleitung.

- Cubetto
- Bee-Bot
- KIBO
- Ozobot ohne visuelle Sprache

*Systeme mit mittlerer Komplexität:* Diese Systeme werden häufig in Verbindung mit einer App (fürs Smartphone oder Tablet) oder auch mit einem Computer ein-

gesetzt. Aufgrund der Verwendung einer Programmiersprache benötigt die Lernbegleitung je nach System informatische Kompetenzen im Bereich der Programmierung.

- ScratchJr
- Dash & Dot
- WeDo
- Makey Makey
- Ozobot mit visueller Sprache (Besonderheit hier: Dieses System kann die komplette Bandbreite der Komplexität abdecken.)

*Systeme mit höherer Komplexität:* diese Systeme benötigen ein höheres Fachwissen bei der Lernbegleitung:

- Scratch
- LEGO Mindstorms
- Arduino-Mikrocontroller mit ArduBlock

## 2 Beschreibung und fachliche Einschätzung der einzelnen Informatiksysteme

### 2.1 Roboter Cubetto der Firma Primo Toys

#### Überblick

Der Roboter Cubetto<sup>50</sup> der Firma Primo Toys wurde 2013 mit dem Ziel entwickelt, bereits Kindern ab 3 Jahren einen spielerischen Einstieg in die Welt der Programmierung zu bieten. Cubetto ist ein würfelförmiger Holzroboter mit lachendem Gesicht und zwei Rädern zur Fortbewegung. Und Bewegen ist auch alles, was Cubetto kann. Das Besondere an diesem System ist, dass es mit einem eigenen, ebenfalls hölzernen Programmierbrett ausgeliefert wird. In dieses Brett können die Kinder die vier verschiedenfarbigen und unterschiedlich geformten Holzpuzzlestücke (entsprechen einzelnen Programmierbefehlen/Bewegungsanweisungen) einsetzen. Ein Pfeil auf dem Brett zeigt die Reihenfolge der Ausführung an. Haben die Kinder ihr Programm, bestehend aus bis zu 12 Befehlen (Puzzlesteinen), fertiggestellt, überträgt ein Druck auf den einzigen Knopf dieses auf den Roboter. Dieser arbeitet die eingesteckten Befehle nacheinander ab. Gleichzeitig zeigen kleine Lämpchen auf dem Programmierfeld an, welcher Befehl aktuell ausgeführt wird.

Mit dem Roboter und dem Programmierbrett wird eine Plane mit einem Spielfeld ausgeliefert, auf welcher Orte in Form von einfachen Grafiken (z. B. ein Burgturm oder Segelschiff) markiert sind, so dass Kinder Aufgaben im Stil von „Fahre von der Burg zum See“ für den Roboter kreieren können. Auch ohne Plane können Aufgaben im Raum, z. B. „Fahre um den Stuhl“, von den Lernenden entwickelt und gelöst werden.



Als Erweiterung können die Lernenden eine sog. Methode programmieren. Dies ist ein informatisches Konstrukt, um Programmcode effektiv zu gestalten, indem sich wiederholende Elemente nicht mehrfach programmiert werden. Dazu werden bis zu vier Befehle in einen abgetrennten Bereich ausgelagert (Umrandung). Neben den Bausteinen „vorwärts“, „links“ und

<sup>50</sup> <https://www.primotoys.com/de/>



„rechts“ gibt es einen weiteren Baustein, der die Methode aufruft und damit zur Ausführung der bis zu vier Befehle führt. Dieser zusätzliche Baustein kann nun an mehreren Stellen im Hauptprogramm eingefügt werden, an denen dann die immer gleiche Methode wiederholt durchgeführt wird. Insgesamt ergibt sich durch den eingeschränkten Befehlsumfang und die maximale Programmlänge ein recht überschaubares Einsatzgebiet, was den Cubetto als Anfänger-Werkzeug auszeichnet.

### **Zielgruppe**

Der Cubetto eignet sich, ausgehend von unseren bisherigen Erfahrungen und den auf der Händlerseite präsentierten Einsatzszenarios, besonders für Kinder im Kita-Alter (insbesondere mit Montessori-Pädagogik), da er einem Holzspielzeug entspricht und ohne Anleitung von den Kindern erkundet werden kann. Das Ziel, Holzklötze in passende Löcher zu stecken, ist den meisten Kindern bereits aus dem Kleinkindalter bekannt. Auch das Drücken auf den Knopf ist für die Lernenden naheliegend. Einzig ein Denkanstoß zur Verknüpfung von Programm und Ausführung durch den Roboter muss durch die Lernbegleiterin/den Lernbegleiter geleistet werden, da die Übertragung kabellos funktioniert und daher den Kindern nicht auf Anhieb bewusst ist.

### **Nutzerfreundlichkeit**

Dieses Robotersystem ist extrem nutzerfreundlich und unabhängig jeglichen Vorwissens auf Seiten der Kinder wie Lernbegleiterinnen und Lernbegleiter einsetzbar. Durch die Lernbegleitung müssen lediglich im Vorfeld Batterien in den Roboter und das Programmierbrett eingesetzt und beides durch einen kleinen Schiebeschalter eingeschaltet werden. Die Verbindung funktioniert automatisch, so dass direkt los programmiert werden kann. Es wird kein weiteres Informatiksystem (Computer, Laptop, Tablet) benötigt.

### **Nötige Vorerfahrungen und Vorkenntnisse der Lernbegleitung**

Wie oben beschrieben, ist dieses System ohne jegliches informatisches bzw. technisches Vorwissen einsetzbar. Lediglich die Idee des linearen Ablaufs der Befehle im Hauptprogramm sowie das Konstrukt der Methode müssen vorher verinnerlicht sein. Ansonsten werden nur kreative Aufgabenstellungen für Cubetto benötigt, die nach einer ersten Erkundung aber von den Kindern selbst mit großer Begeisterung entwickelt werden.

### **Nötige Vorerfahrungen und Vorkenntnisse der Kinder**

Auch hier sind keine Vorerfahrungen zwingend erforderlich. Die Idee, Holzpuzzlesteine in entsprechende Löcher zu stecken, sollte verinnerlicht sein. Die Thematik

Roboter, Programm, Befehle etc. kann anhand dieses Lernwerkzeugs behandelt werden.

### Lernziele

Das übergeordnete Lernziel besteht darin, ein Grundverständnis für automatisiert ausführbare Programme zu erzeugen. Die Kinder lernen, dass Programmieren bedeutet, dass ein Mensch einem Informatiksystem, in diesem Fall einem Roboter, eine Abfolge von Befehlen eingibt und dieser das Programm auf Knopfdruck ausführt. Im Detail lernen die Kinder somit Algorithmen kennen und diese selbst zu entwickeln. Da der Cubetto zudem wunderbar kollaborativ durch mehrere Kinder programmiert werden kann, kommen sie indirekt auch mit der zielgerichteten Kommunikation über Programme und den Herausforderungen der Teamarbeit in Berührung.

### Bezug zu den Zieldimensionen informatischer Bildung

Mit dem Roboterset Cubetto wird in erster Linie der *Inhaltsbereich* „I2 Algorithmen & Programmierung“ adressiert. Durch das zielgruppengerechte Design können mit Cubetto bereits Kita-Kinder selbst Algorithmen entwickeln. Je nach Ausgestaltung der Gespräche in den Lerngruppen bzw. weiterer Lernphasen können auch Aspekte von „I4 Informatiksysteme“ und „I5 Informatik, Mensch und Gesellschaft“ thematisiert werden.

Bei den *Prozessbereichen* liegt der Fokus hier auf „P0 Interagieren & Explorieren“, indem der programmierbare Roboter sehr frei von den Kindern erkundet werden kann. Weiter können die Lernenden bereits eigene Programme implementieren (ein Aspekt von „P1 Modellieren & Implementieren“). Beim Spielen in Teams findet auch der Prozessbereich „P4 Kommunizieren & Kooperieren“ Anwendung.

## 2.2 Bee-Bot der Firma Terrapin

### Überblick

Der Bee-Bot<sup>51</sup> ist ein programmierbarer Bodenroboter für Kinder im Vorschul- und Grundschulalter in Gestalt einer Biene. Durch ihre einfache und kindgerechte Gestaltung sind Bee-Bots ein guter erster Einstieg in das Programmieren. Programmiert werden können Bewegungsabläufe des Bee-Bot. Hierzu sind auf der „Biene“ sieben Tasten angebracht. Bee-Bots können sich vorwärts und rückwärts bewegen (jeweils um genau 15 cm) sowie jeweils eine 90-Grad-Drehung nach links oder rechts durchführen. Durch Drücken der Richtungsknöpfe können so bis zu 40 aufeinanderfolgende Befehle als Ablauf einprogrammiert werden. Über den

---

<sup>51</sup> <https://www.bee-bot.us>

„Go-Button“ in der Mitte wird die eingegebene Sequenz gestartet und abgearbeitet. Darüber hinaus existiert ein Button zum Löschen der vorherigen Sequenz. Wird eine Sequenz nicht gelöscht, werden die darauffolgenden Befehle an die bisherige Sequenz angehängt. Dies ermöglicht den Kindern, sich Schritt für Schritt vorzuarbeiten. Unterstützung in der Nachvollziehbarkeit der Eingaben gibt es durch entsprechende Geräusche oder Leuchtsignale des Bee-Bot So blinkt er auf und gibt ein Geräusch von sich, wenn ein Befehl des eingegebenen Programms ausgeführt wurde.



Betrieben wird der Bee-Bot mit einem Akku, der über eine USB-Schnittstelle geladen werden kann.

Für die Verwendung des Bee-Bot existieren inzwischen zahlreiche Materialien sowohl vom Hersteller selbst als auch von unterschiedlichen Forschungs- und Bildungseinrichtungen. Viele dieser Materialien legen Wert auf fächerübergreifendes Lernen und Problemlösen.

Neben dem Bee-Bot existieren mittlerweile weitere Varianten wie der Blue-Bot oder der Pro-Bot. Der Blue-Bot kann zusätzlich über ein Tablet gesteuert werden. Der Pro-Bot kommt mit einem gänzlich neuen Design als Rennauto daher und bietet zahlreiche neue Funktionen und somit mehr Möglichkeiten als der Bee-Bot.

### **Zielgruppe**

Der Bee-Bot ist für Kinder im Vorschul- und Grundschulalter von etwa 4 bis 9 Jahren geeignet. Bei älteren Kindern wirkt der Bee-Bot im Zuge seiner Aufmachung als Biene schnell zu kindlich.

### **Nutzerfreundlichkeit**

Der Bee-Bot ist ein sehr nutzerfreundliches System. Die Bedeutung der Bedientknöpfe wird auch ohne eine Anleitung zumindest etwas älteren Kindern schnell klar, jüngere Kinder benötigen ggf. eine kurze Erläuterung.

### *Nötige Vorerfahrungen und Vorkenntnisse der Lernbegleitung*

Die nötigen Vorerfahrungen und Vorkenntnisse der Lernbegleitung beschränken sich auf die Funktionalität des Bee-Bot und die Idee des linearen Ablaufs der Befehle. Mit diesen Aspekten vertraut, benötigt es nur entsprechende (kreative) Herausforderungen, die mit Hilfe des Bee-Bots gelöst werden sollen. Dies können einfach Wegfindungen sein, aber auch die Zusammensetzung von Wörtern auf Buchstabenmatten. Hier finden aber sicher die Kinder nach einer Erkundungsphase auch selbst erste Herausforderungen. Darüber hinaus gibt es bereits zahlreiche Materialien für die Anwendung des Bee-Bots, z. B. über die Homepage des Herstellers (<https://www.bee-bot.us/bee-bot/bee-bot-curriculum.html> [Zugriff am 05.03.2018]).

### **Nötige Vorerfahrungen und Vorkenntnisse der Kinder**

Auf der Ebene der Kinder sind keine Vorerfahrungen notwendig. Gegebenenfalls muss die Funktionalität der Bedienknöpfe erläutert werden. Dies hängt aber vom Alter und den Vorerfahrungen der Kinder ab. Für die Programmierung des Bee-Bot (je nach Komplexität des zu programmierenden Weges) ist eine gewisse Arbeitsgedächtnisleistung der Kinder notwendig. Der Weg muss vorgeplant werden, da die Eingabe in den Bee-Bot für die Kinder nicht sichtbar ist und somit nicht direkt nachvollzogen werden kann. Zur Unterstützung existieren allerdings Karten, mit denen Wege z. B. vorgeplant und anschließend eingegeben werden können.

### **Lernziele**

Das übergeordnete Lernziel ist, ein Grundverständnis für automatisiert ausführbare Programme zu erzeugen. Wie auch beim Cubetto lernen die Kinder, dass ein Mensch durch Eingabe von Befehlen (programmieren) ein Informatiksystem steuert und dieses das Programm ausführt. Die Kinder lernen dabei den Umgang mit einem einfachen Informatiksystem und darüber hinaus das Problemlösen durch Algorithmen. Da der Bee-Bot auch gut kollaboratives Arbeiten mehrerer Kinder ermöglicht, lernen sie indirekt auch die zielgerichtete Kommunikation über Programme und die Herausforderungen der Teamarbeit.

### **Bezug zu den Zieldimensionen informatischer Bildung**

Auch mit dem Bee-Bot wird in erster Linie der *Inhaltsbereich* „I2 Algorithmen & Programmierung“ adressiert. Durch das zielgruppengerechte Design können mit dem Bee-Bot bereits Kita-Kinder im Rahmen der Problemlösung Algorithmen entwickeln und diese in den Bee-Bot eingeben (*Prozessbereich* „P1 Modellieren & Implementieren“). Je nach Ausgestaltung der Gespräche in den Lerngruppen bzw. weiterer Lernphasen können auch Aspekte von „I4 Informatiksysteme“ und „I5 Informatik, Mensch und Gesellschaft“ thematisiert werden.

Bei den weiteren *Prozessbereichen* liegt der Fokus hier auf „P0 Interagieren & Explorieren“, indem der Bee-Bot sehr frei von den Kindern erkundet werden kann und sie ihre Interaktionskompetenz im Umgang mit diesem System anwenden oder ausbilden können. Beim Spielen in Teams findet auch der Prozessbereich „P4 Kommunizieren & Kooperieren“ Anwendung.

## 2.3 KIBO von KinderLab Robotics

### Überblick

Beim KIBO<sup>52</sup> handelt es sich um ein pädagogisches Roboterkit, welches den Kindern das Konzept der Programmierung interaktiv näherbringen soll. Die Programmierung für den Roboter KIBO erfolgt über beschriftete, ineinander steckbare Holzklötze.

Der KIBO bietet den Kindern zahlreiche kreative Möglichkeiten. Die vorhandenen Bauelemente bieten viele unterschiedliche Gestaltungsmöglichkeiten, z. B. als Karussell, Roboter mit Ohr, Auge, Lampen oder aber mit zusätzlichem Bastelmaterial frei nach der Phantasie der Kinder. Dies ermöglicht eine breite Anwendungspalette, von konkreten, von der Lernbegleitung gestellten Aufgaben bis hin zu freiem kreativen Arbeiten mit dem KIBO.

Der KIBO besitzt Sensoren, Lichter und Motoren. So kann er beispielsweise auf Klatschen reagieren, seine Lampen können unterschiedlich leuchten und er kann sich in verschiedene Richtungen bewegen.

Gesteuert wird der KIBO, wie bereits erwähnt, über Holzklötze. Auf diesen Klötzen befinden sich die Anweisungen für den KIBO, die mit einer Steckverbindung sequenziell aneinandergereiht werden. Anschließend werden sie mit einem Infrarot-Scanner, der sich am KIBO befindet, selbst eingescannt. Mit einem Knopfdruck wird dann das Programm gestartet und vom KIBO ausgeführt. Die Anweisungen beziehen sich z. B. auf die Fahrtrichtung des KIBO, die Farbe der Lampe, Reaktionen auf Klatschen etc.

Auch wenn die Klötze sequenziell aneinandergereiht wer-



<sup>52</sup> <http://kinderlabrobotics.com/kibo/>

den, gibt es bereits Konstrukte wie eine Wiederholung und bedingte Anweisungen.

Die Anweisungen auf den Holzklötzen sind sowohl grafisch als auch schriftlich, so dass bereits Kinder ohne Lesekompetenz mit dem KIBO erste Erfahrungen im Bereich der Programmierung machen können.

### **Zielgruppe**

Laut Hersteller ist der KIBO für Kinder im Alter von 4 bis 7 Jahren angedacht. Aus unseren Erfahrungen würden wir den Einsatz im Vorschul- bzw. Grundschulalter empfehlen.

### **Nutzerfreundlichkeit**

Der KIBO ist sehr benutzerfreundlich und unabhängig von jeglichem Vorwissen auf Seiten der Kinder wie auch der pädagogischen Fach- und Lehrkräfte einsetzbar. Im Vorfeld müssen von der Lernbegleitung lediglich Batterien in den KIBO eingesetzt werden. Die einzelnen Elemente des KIBO lassen sich recht intuitiv an den KIBO stecken. Bei kleineren Kindern könnte dieser Schritt ggf. auch im Vorfeld von der Lernbegleitung durchgeführt werden. Zur Programmierung müssen lediglich Holzbausteine zusammengesteckt werden. Diese werden dann mit einem Infrarotscanner am KIBO selbst eingescannt. Die richtige Haltung des KIBO haben die Kinder schnell erkannt. Fehler meldet er sofort zurück. Nach dem erfolgreichen Scannen geht es auf Knopfdruck los.

### **Nötige Vorerfahrungen und Vorkenntnisse der Lernbegleitung**

Für den Einsatz des KIBO sind grundlegende informatische Vorkenntnisse nötig. Die Idee des linearen Ablaufs der Befehle sowie die Funktionen von einfachen Schleifen und Bedingungen sollten bekannt sein, um die Kinder hier ggf. unterstützen zu können.

### **Nötige Vorerfahrungen und Vorkenntnisse der Kinder**

Auf Ebene der Kinder sind keine Vorerfahrungen zwingend erforderlich. Die Idee, die Holzbausteine entsprechend zusammenzustecken, sollte verinnerlicht sein. Die Thematik Roboter, Programm, Befehle etc. kann mittels dieses Lernwerkzeugs behandelt werden.

### **Lernziele**

Übergeordnetes Lernziel ist auch beim KIBO, ein Grundverständnis für automatisiert ausführbare Programme zu erzeugen. Kinder erlernen erste Grundkonzepte der Programmierung und vor allem, dass Programmierung bedeutet, in ein Informatiksystem eine Abfolge von Befehlen einzugeben, welche dieses dann ausführt.

Kinder lernen so neben der sequenziellen Abfolge von Befehlen auf spielerische Art und Weise bereits Bedingungen und Schleifen kennen. Da der KIBO außerdem wunderbar kollaborativ durch mehrere Kinder programmiert werden kann, lernen sie indirekt auch die zielgerichtete Kommunikation über Programme und die Herausforderungen der Teamarbeit.

### **Bezug zu den Zieldimensionen informatischer Bildung**

Mit dem KIBO werden in erster Linie der *Inhaltsbereich* „I2 Algorithmen & Programmierung“ sowie der *Prozessbereich* „P1 Modellieren & Implementieren“ adressiert. Durch das zielgruppengerechte Design können mit dem KIBO selbst erste Algorithmen entwickelt und über die Holzklötze implementiert werden. Je nach Ausgestaltung der Gespräche in den Lerngruppen bzw. weiterer Lernphasen können auch Aspekte von „I4 Informatiksysteme“ und „I5 Informatik, Mensch und Gesellschaft“ thematisiert werden.

Bei den *Prozessbereichen* liegt ein weiterer Fokus auf „P0 Anwenden & Explorieren“, indem der programmierbare Roboter und seine Elemente sehr frei von den Kindern erkundet werden kann. Beim Spielen in Teams findet auch der Prozessbereich „P4 Kommunizieren & Kooperieren“ Anwendung.

## **2.4 Ozobot/Ozobit von Evolve Inc.**

### **Überblick**

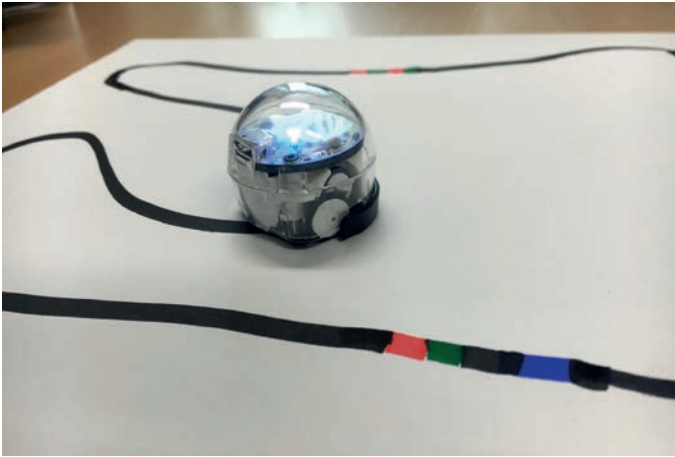
Beim im Jahr 2014 vorgestellten Ozobot<sup>53</sup> handelt sich um einen ca. 2,5 cm großen Roboter. Ein Ozobot kann sowohl Linien folgen als sich auch – in der neuen Generation dem Ozobot-Ozobit<sup>54</sup> – frei im Raum bewegen. Zusätzlich ist er auf zwei unterschiedliche Arten programmierbar, was eine breite Einsatzmöglichkeit in unterschiedlichen Altersstufen ermöglicht. Betrieben wird der Ozobot mit einem Akku, der über eine USB-Schnittstelle aufgeladen werden kann. An der Unterseite besitzt dieser Roboter einen Fünffarbsensor mit dessen Hilfe er in der Lage ist, unterschiedliche Farben auf dem Boden zu erkennen. Anhand von Linien und Farben auf dem Boden kann sich der Ozobot in unterschiedliche Richtungen mit unterschiedlichen Geschwindigkeiten bewegen und selbst in verschiedenen Farben leuchten.

Programmierbar ist der kleine Roboter auf zwei unterschiedliche Art und Weisen:

---

<sup>53</sup> <http://ozobot.com>

<sup>54</sup> Da der Ozobit mehr Möglichkeiten bietet als die erste Generation des Ozobot, wird im Folgenden immer Bezug auf diese neue Generation Ozobit genommen.



Programmierung mit Farbcodes:  
Die Programmierung mit Farbcodes kann ganz ohne weiteres Informatiksystem erfolgen. Neben dem Ozobot werden lediglich Stifte und Papier benötigt. Die Abfolgen von Linien und Farbcodes werden einfach auf Papier gezeichnet. Anhand dieser Codes kann der Ozobot seine Farbe, seine Geschwindigkeit oder seine Richtung verändern. Sogenannte „Special Moves“ er-

möglichen zudem Drehungen oder Zickzack-Fahren. Somit können für den Ozobot beispielsweise ganze Parcours gebaut werden. Der Kreativität sind keine Grenzen gesetzt. Wer möchte, kann Linien und Farbcodes aber auch auf einem Tablet zusammenstellen und den Ozobot auf diesem fahren lassen.

Programmierung mit visueller Programmiersprache OzoBlockly:

Die neue Generation der Ozobots, der sog. Ozobit, kann mit der visuellen Programmiersprache OzoBlockly programmiert werden. Diese erinnert an andere visuelle Programmiersprachen wie z. B. Scratch (s. u.), ist aber spezifisch für den Ozobot entwickelt. Sie ist sowohl über einen Internetbrowser am Rechner als auch über ein Tablet oder gar ein Smartphone anwendbar. Zur Übertragung des erstellten Programms auf den Ozobot muss dieser lediglich an den Bildschirm gehalten bzw. auf das Tablet gestellt werden.

Sehr nützlich ist der Aufbau in unterschiedliche Niveaustufen. So stehen zu Beginn lediglich Programmierbausteine mit Symbolen zur Verfügung, so dass ggf. bereits Kinder ohne Lesekompetenz den Ozobot über diesen Weg programmieren können. In den weiteren Niveaustufen kommen weitere und immer komplexere Programmierbausteine hinzu. Programmierbar ist sowohl das Verhalten des Ozobot auf Linien und Farben als auch völlig freie Bewegungen im Raum.

Die Motoren des Ozobot sind sehr präzise, so dass auch Aufgaben mit Messungen oder synchrone Anwendungen etc. möglich sind.

Zusätzlich bietet der Ozobot selbst, über Gestaltungsmöglichkeiten der Schutzhüllen mit Papier oder Ozobot-spezifischem Zubehör, kreative Anwendungen.

Alles in allem ist der Ozobot ein kleiner begeisternder Roboter mit vielen kreativen Möglichkeiten und unterschiedlichen Schwierigkeitsgraden für Klein und Groß.



### **Zielgruppe**

Der Ozobot eignet sich für einen Einsatz in der Grundschule und darüber hinaus. Hier hängt es von der Gestaltung des Angebots ab, in welcher Klasse man startet. Laut Hersteller ist der Ozobot für Kinder ab 8 Jahren geeignet. Aufgrund der unterschiedlichen Möglichkeiten für die Programmierung und damit der unterschiedlichen Schwierigkeitsgrade gibt es nach oben keine Grenzen für den Einsatz des Ozobot

Inwieweit der Ozobot auch für jüngere Kinder geeignet ist, wäre zu erproben. Die Bedeutung vieler Farbcodes lässt sich auch gut durch Symbole anstatt durch Schrift darstellen, so dass der Einsatz ggf. auch bei Kindern ohne oder mit sehr geringer Lesekompetenz möglich ist.

### **Nutzerfreundlichkeit**

Der Ozobot ist an sich sehr benutzerfreundlich. Allerdings muss bei der Anwendung auf eine regelmäßige Kalibrierung geachtet werden. Bei der Verwendung von nicht Original-Stiften muss beachtet werden, dass die Farbtöne sehr nah an das Original herankommen, da sonst die Farben nicht immer zu 100 % richtig erkannt werden.

Für den Ozobot existieren ein gutes, ausführliches Handbuch und viele Beispiele. Darüber hinaus ist es möglich, sich den entsprechenden JavaScript Code anzeigen zu lassen.

Nachteil von OzoBlockly, inklusive Handbuch, ist zurzeit, dass es lediglich in englischer Sprache zur Verfügung steht.

### **Nötige Vorerfahrungen und Vorkenntnisse der Lernbegleitung**

Bei den nötigen Vorerfahrungen bzw. Vorkenntnissen der Lernbegleitung ist zu unterscheiden, wie der Ozobot angewendet werden soll.

Für die Anwendung mit Papier und Stiften sind keine Vorkenntnisse bei der Lernbegleitung notwendig. Sie muss im Vorfeld die Ozobots per USB aufladen und sollte den Ozobot selbst natürlich bereits ausprobiert haben. Für die Bedeutung der Farbcodes stellt der Hersteller der Ozobots bereits Material zur Verfügung. Zusätzlich existieren einige Arbeitsblätter und im Internet bereits viele kreative Ideen zum Einsatz des Ozobot.

Für die Verwendung der visuellen Programmiersprache sollten je nach Schwierigkeitsstufe Vorkenntnisse über grundlegende Programmierstrukturen vorhanden sein, um die Kinder bei Schwierigkeiten oder Fragen entsprechend unterstützen zu können.

### **Nötige Vorerfahrungen und Vorkenntnisse der Kinder**

Die Kinder benötigen für die Steuerung des Ozobot über Farbcodes keinerlei Vorkenntnisse.

Soll der Ozobot mit OzoBlockly programmiert werden, sollten die Kinder sicher im Umgang mit dem Computer bzw. dem Tablet sein. Programmiervorerfahrungen sind allerdings nicht notwendig.

Die Thematik Roboter, Programm, Befehle etc. kann mittels dieses Lernwerkzeugs behandelt werden.

### **Lernziele**

Die Lernziele bei der Verwendung des Ozobot können je nach Anwendung unterschiedlich gestuft sein. Übergeordnetes Lernziel ist, ein Grundverständnis für die Funktionsweise von automatisierten, ausführbaren Programmen zu erzeugen. Die Kinder lernen, dass Programmieren bedeutet, dass ein Mensch einem Informatiksystem, in diesem Fall, einem Roboter, eine Abfolge von konkreten Befehlen eingibt. Diese Befehle können auf unterschiedliche Weise codiert sein, z. B. mit Farben oder durch eine visuelle Programmiersprache. Im Detail lernen die Kinder somit unterschiedliche Möglichkeiten für die Programmierung eines Systems kennen. Darüber hinaus lernen sie Algorithmen kennen und diese selbst zu entwickeln. Bei der Verwendung von OzoBlockly werden sie zusätzlich mit grundlegenden Strukturen von Programmiersprachen vertraut gemacht. Da der Ozobot auch wunderbar kollaborativ durch mehrere Kinder programmiert werden kann, lernen sie indirekt auch die zielgerichtete Kommunikation über Programme und die Herausforderungen der Teamarbeit.

### **Bezug zu den Zieldimensionen informatischer Bildung**

Mit dem Ozobot werden in erster Linie der *Inhaltsbereich* „I2 Algorithmen & Programmierung“ sowie der *Prozessbereich* „P1 Modellieren & Implementieren“ adressiert. Durch das zielgruppengerechte Design können mit ihm auf unterschiedliche Art und Weise selbst Algorithmen entwickelt und diese auf unterschiedliche Art und Weise implementiert werden. Je nach Ausgestaltung der Gespräche in den Lerngruppen bzw. weiterer Lernphasen können auch Aspekte von „I4 Informatiksysteme“ und „I5 Informatik, Mensch und Gesellschaft“ thematisiert werden.

Bei den *Prozessbereichen* liegt ein weiterer Fokus auf „P0 Interagieren & Explorieren“, indem der programmierbare Roboter sehr frei von den Kindern erkundet werden kann. Beim Spielen in Teams findet auch der Prozessbereich „P4 Kommunizieren & Kooperieren“ Anwendung.

## 2.5 LEGO WeDo 2.0

### Überblick

LEGO WeDo 2.0<sup>55</sup> ist ein handlungsorientiertes Lernkonzept von LEGO, was auf dem Vorgängerkonzept WeDo (siehe Abbildung) basiert. WeDo wurde speziell für den Sachunterricht der Grundschule entwickelt und bietet bereits an den Lehrplan angepasste, ausgearbeitete Unterrichtsmaterialien in Form von Projekten. Der Bausatz zur Gestaltung dieser Modelle enthält neben unterschiedlichen LEGO-Steinen, Rädern, Sensoren z. B. zur Neigung und Bewegung, einen Motor und einen zentralen Steuerbaustein, den sog. Smarthub. Per Bluetooth kann eine Verbindung zwischen dem Smarthub und der Software auf einem PC, Mac, Android Tablet oder iPad hergestellt werden. Diese Software bietet u.a. eine Programmieroberfläche. Mit dieser können die erstellten Modelle, programmiert werden. Für den Vorgänger WeDo gab es zusätzlich eine Anbindung mittels der visuellen Programmiersprache Scratch. Diese existiert für WeDo 2.0 zurzeit nur für MacOS, ist aber auch für Windows in Planung. Darüber hinaus liefert die Software Material, das durch alle von LEGO angebotenen Projekte führt: digitale Schritt-für-Schritt-Bauanleitungen, Filme, ein digitales Lerntagebuch und die oben genannte Programmierumgebung. Zurzeit scheint die App allerdings ausgereifter zu sein als die Software für den Computer.

Die Konzipierung von WeDo ermöglicht es, die Kinder an informatisches Arbeiten und Forschen heranzuführen. Quasi ganz nebenbei lernen und erkennen sie, wie viele Informatiksysteme bzw. programmierbare Geräte in ihrer eigenen Lebenswelt existieren. Sie lernen im Kleinen, wie Geräte programmiert werden und dass mit diesen Geräten und durch Programmierung unterschiedliche Aufgaben gelöst werden können.

### Zielgruppe

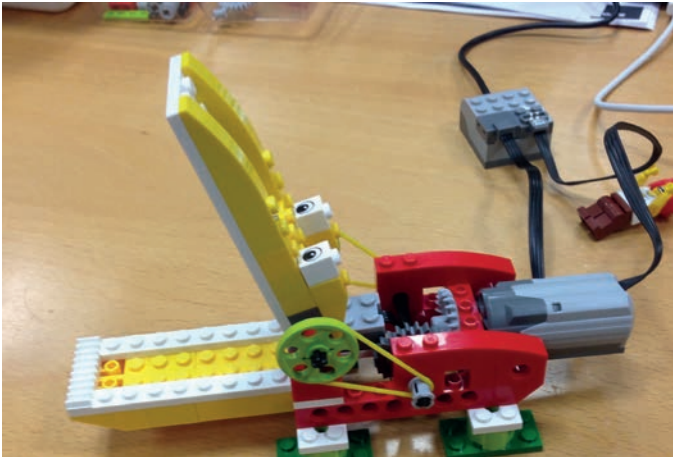
LEGO WeDo 2.0 eignet sich laut Hersteller und unseren eigenen Erfahrungen für Grundschulkindern ab der 2. Klasse.

### Nutzerfreundlichkeit

Auch wenn LEGO WeDo 2.0 im Vergleich zu vielen anderen hier vorgestellten Systemen sehr komplex ist, ist es auf Grund der zahlreichen Materialien und der unterstützenden Software sehr nutzerfreundlich.

---

55 <https://education.lego.com/de-de/grundschule/entdecken/sachunterricht>



### **Nötige Vorerfahrungen und Vorkenntnisse der Lernbegleitung**

Die Lernbegleitung sollte sich im Vorfeld mit dem LEGO WeDo-Baukasten und dem Modell, welches sie einsetzen möchte, auseinandergesetzt haben, um den Kindern bei entsprechenden Fragen oder Problemen unterstützend zur Seite stehen zu können. Eine spezifische informatische Vorbildung ist unter Verwendung

der LEGO Software nicht notwendig. Eine Auseinandersetzung mit der Programmiersprache für LEGO WeDo 2.0 sollte aber stattgefunden haben. Soll statt der LEGO-eigenen Programmiersprache die Sprache Scratch verwendet werden, sollte die Lernbegleitung die dafür notwendigen Vorerfahrungen besitzen.

### **Nötige Vorerfahrungen und Vorkenntnisse der Kinder**

Von Seiten der Kinder sind keine spezifischen Vorerfahrungen und Vorkenntnisse nötig. Viele Kinder werden LEGO aus ihrem Kinderzimmer kennen. Bauanleitungen für die vorgeschlagenen Modelle liefert die Software.

### **Lernziele**

Die informatikspezifischen Lernziele im Kontext von WeDo beziehen sich u. a. auf den Aufbau eines beispielhaften Informatiksystems. Die Kinder lernen, wie Sensoren und Aktoren in Verbindung mit Motoren interagieren können. Darüber hinaus bekommen sie in Verbindung mit der Programmierung ein Grundverständnis über Programmierung und grundlegende Programmierkonzepte.

Durch die unterschiedlichen Beispiele lernen die Kinder nach und nach unterschiedliche programmierbare Systeme kennen und können so einen Bezug zu Systemen in ihrer Lebenswelt herstellen.

Durch die Möglichkeit des kollaborativen Arbeitens mehrerer Kinder, lernen sie indirekt auch die zielgerichtete Kommunikation über Programme und die Herausforderungen der Teamarbeit.

Nicht zuletzt lernen die Kinder den Umgang mit einem Informatiksystem kennen.

### Bezug zu den Zieldimensionen informatischer Bildung

Mit LEGO WeDo 2.0 werden in erster Linie die *Inhaltsbereiche* „I4 Informatiksysteme“ sowie „I2 Algorithmen & Programmierung“ adressiert. Durch das zielgruppengerechte Design können Grundschul Kinder ihr eigenes Informatiksystem gestalten und im Rahmen der Problemlösung Algorithmen für dieses System entwickeln (*Prozessbereich* „P1 Modellieren & Implementieren“). Je nach Ausgestaltung der Gespräche innerhalb der Lerngruppe und den unterschiedlichen Lernphasen können auch Aspekte aus dem Bereich „I5 Informatik, Mensch und Gesellschaft“ thematisiert werden.

Bei den weiteren *Prozessbereichen* liegt der Fokus hier auf „P0 Interagieren & Explorieren“, indem die Kinder ihr eigenes Informatiksystem erstellen und erkunden können. In diesem Kontext können die Kinder Interaktionskompetenz im Umgang mit diesem System anwenden oder ausbilden. Beim Spielen in Teams findet auch der Prozessbereich „P4 Kommunizieren & Kooperieren“ Anwendung.

## 2.6 Dash & Dot von Wonder Workshop

### Überblick

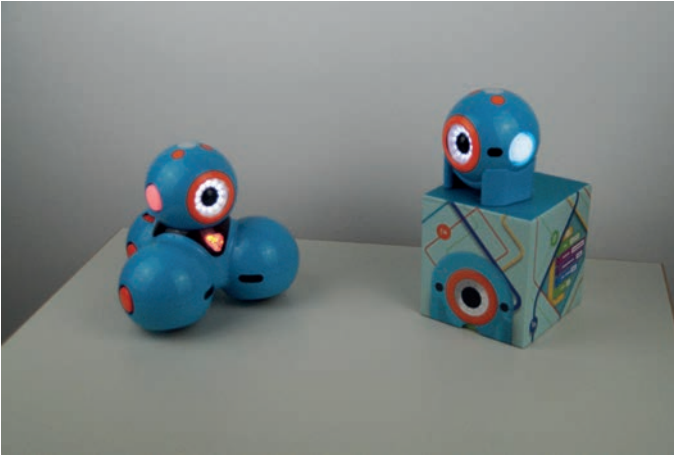
Dash & Dot<sup>56</sup> sind ein zweiteiliges, türkisfarbenes Roboterpaar, welches auf den ersten Blick einem klassischen Spielzeug gleicht. Sie sind optisch sehr kindgerecht (auch für den Kita-Bereich) gestaltet und sehr robust gebaut. Die beiden Roboter können mit Hilfe verschiedener Apps gesteuert und programmiert werden. Der größere Roboter Dash kann den Kopf bewegen, fahren, sprechen, leuchten und auf Geräusche wie auch Hindernisse reagieren. Sein kleiner Bruder Dot steht fest und dient in erster Linie als Kommunikationspartner für Dash. Darüber hinaus sind zahlreiche Ergänzungssets (z. B. ein Adapter für weitere Lego-Anbauten oder auch eine Handyhalterung) verfügbar, so dass den kreativen Einsatzszenarien wenig Grenzen gesetzt werden. Die verschiedenen Apps lassen Kinder im Vor- und Grundschulalter die Roboter selbstständig erkunden. Die App „Go“ ermöglicht eine Handsteuerung, die von Kindern intuitiv bedient werden kann. Andere Apps (wie „Wonder“) ermöglichen die Programmierung mittels verschiedener Ansätze. Auch die verbreitete grafische Programmierumgebung Blockly kann verwendet werden.

### Zielgruppe

Je nach verwendeter App können die Roboter Dash & Dot bereits von Kindern im Kita-Alter problemlos wie mit einer Fernsteuerung gesteuert werden. Andere

---

<sup>56</sup> <https://www.makewonder.de>



Apps ermöglichen darüber hinaus einen tieferen Einstieg in die Programmierung, so dass auch Kinder der Unter- und sogar Mittelstufe mit den Robotern herausfordernde Aufgaben lösen und selbst kreieren können. Laut Herstellerseite werden Dash & Dot weltweit bereits in über 7.000 Grundschulen eingesetzt.

### **Nutzerfreundlichkeit**

Die Roboter Dash & Dot können ausgepackt, geladen und sofort eingesetzt werden. Da es sich hier um ein sehr robustes Komplettsystem handelt, sind diese extrem nutzerfreundlich. Ein weiterer Aspekt der Nutzerfreundlichkeit sind die sehr professionell und kindgerecht entwickelten Apps, die allerdings nur auf größeren (neueren) Smartphones und Tablets funktionieren. Diese sind kostenfrei für Android und iOS herunterzuladen.

### **Nötige Vorerfahrungen und Vorkenntnisse der Lernbegleitung**

Für den simplen Einsatz einer Fernsteuerungsapp muss die Lernbegleitung keinerlei informatische Vorkenntnisse haben, sondern lediglich das Tablet (oder Smartphone) bedienen können. Um die Roboter selbst programmieren zu können, bedarf es grundlegender Programmierkenntnisse, wobei die Apps so kindgerecht gestaltet sind, dass sich auch eine unerfahrene Lernbegleitung schnell zurechtfinden wird.

### **Nötige Vorerfahrungen und Vorkenntnisse der Kinder**

Die Kinder sollten den Umgang mit einem Smartphone oder Tablet gewöhnt sein bzw. es sollte Zeit für die Erkundung dieses Mediums eingeplant werden. Darüber hinaus sind keine Vorkenntnisse zwingend erforderlich. Auch kann das System erst durch die Fernsteuerungsapp erkundet und in folgenden Schritten über einfache Programme selbst weiterentwickelt werden.

### **Lernziele**

Das übergeordnete Lernziel ist ein Grundverständnis für automatisiert ausführbare Programme und die Kommunikation zweier Roboter miteinander. Die Kinder lernen, wie Informatiksysteme (Roboter) mittels Sensoren aufeinander reagieren und somit kommunizieren können. Je nach didaktischer Ausgestaltung erweitern

die Kinder ihre Problemlösekompetenz, entwickeln eigene Algorithmen oder formulieren gar selbst Herausforderungen.

Mögliche Bestandteile (z. B. Sensoren, Motoren) eines Informatiksystems können mittels Dash & Dot von den Kindern selbstständig erforscht werden, wobei dieses Informatiksystem nicht veränderbar ist.

### **Bezug zu den Zieldimensionen informatischer Bildung**

Mit dem Roboterpaar Dash & Dot wird in erster Linie der *Inhaltsbereich* „I2 Algorithmen & Programmierung“ adressiert. Durch das zielgruppengerechte Design und die leicht zu bedienenden Apps können diese bereits mit Kita-Kindern auch zur Erkundung der Inhaltsbereiche „I4 Informatiksysteme“ und „I5 Informatik, Mensch und Gesellschaft“ genutzt werden

Bei den *Prozessbereichen* liegt der Fokus zu Beginn des Einsatzes eindeutig auf „P0 Interagieren & Explorieren“, da Dash & Dot über die Apps auf unterschiedlichen Niveaustufen sehr frei von den Kindern erkundet werden können. In der weiteren Beschäftigung kann dann auch der Bereich „P1 Modellieren & Implementieren“ bedient werden, wobei die Modellierung durch die Lernbegleitung außerhalb der Apps fokussiert werden könnte (z. B. durch Kärtchen, die den Zustand der Roboter beschreiben, und dann passend angeordnet werden können, je nachdem in welcher Reihenfolge die Zustände durchlaufen werden sollen). Je nach didaktischer Ausgestaltung können auch die anderen Prozessbereiche angesprochen werden.

## **2.7 Scratch und ScratchJR**

### **Überblick**

Scratch<sup>57</sup> und ScratchJR<sup>58</sup> (Junior-Variante von Scratch) sind grafische Programmierumgebungen. Das bedeutet, dass die Kinder keinen textuellen Programmcode schreiben müssen, sondern Puzzlesteine (Codefragmente) passend zusammensetzen. Die wichtigsten drei Elemente der Oberfläche beider Anwendungen sind eine Bühne, die Menüs mit den Befehlen und eine Programmierfläche.

Auf der Bühne führen die Objekte (z. B. eine Katze) die Programme aus (z. B. bewegen sich oder sprechen). Die Befehle müssen von den Kindern aus dem Menü per Drag-and-drop auf die Programmierfläche gezogen und passend verbunden werden. Das so entstehende Programm kann jederzeit getestet werden. So können die Kinder ihre Ergebnisse selbstständig bewerten und entsprechend ihren Zielen verbessern.

---

<sup>57</sup> <https://scratch.mit.edu>

<sup>58</sup> <https://www.scratchjr.org>



Bei Scratch beinhalten die Puzzle-Elemente textuelle Befehle (teilweise kombiniert mit Symbolen), z. B.: Drehe dich nach rechts um 90 Grad. In der App ScratchJR bestehen die Befehle ausschließlich aus Symbolen, auch ist der Befehlsumfang deutlich eingeschränkter.

Neben einfachen Anweisungen zur Bewegung gibt es in beiden Varianten die Möglichkeit, die Objekte optisch zu verändern

(z. B. zu verkleinern), zudem können Geräusche abgespielt werden oder mehrere Objekte miteinander agieren. Bei Scratch können die Objekte auch auf zahlreiche Sensoren reagieren (z. B. wenn das Objekt an den Rand der Fläche stößt). Auch sind alle grundlegenden Programmierkonstrukte (Wiederholungen, Verzweigungen etc. verfügbar). Insgesamt sind durch das umfangreiche Spektrum der Befehle (insbesondere bei Scratch) der eigenen Phantasie und Kreativität der Kinder keine Grenzen gesetzt.

Neben diesem Einsatz zur Programmierung von Objekten in einer Mikrowelt gibt es für Scratch Erweiterungen, mit denen es möglich ist, auch Informatiksysteme wie den Mikrocontroller Raspberry Pi oder auch LEGO WeDo zu programmieren.

### Zielgruppe

Die Original-Variante Scratch eignet sich für Kinder ab der vierten Klasse, wobei die Entwicklerinnen und Entwickler selbst die Zielgruppe der 8- bis 16-jährigen ansprechen wollen. Die App ScratchJR kann bereits von Vorschulkindern problemlos und zielgerichtet genutzt werden, da keine Lesekompetenz erforderlich ist. Hier empfehlen die Entwicklerinnen und Entwickler die Altersstufen 5 bis 7 Jahre.

### Nutzerfreundlichkeit

Scratch gibt es bereits seit 2007. Ursprünglich handelte es sich um ein Programm, welches kostenfrei heruntergeladen und installiert werden konnte. Seit ein paar Jahren wird Scratch browserbasiert (als Website) angeboten, wodurch der Aufwand der Installation wegfällt. Bereits ohne einen Account können die Kinder mit Scratch programmieren, allerdings bedarf es zum Speichern der eigenen Projekte einer Anmeldung (ebenfalls kostenfrei). Falls eine Online-Nutzung nicht möglich ist, steht weiterhin eine Offline-Variante zum Download zur Verfügung. Erleichtert



wird der Einsatz in der Grundschule dadurch, dass Scratch in unzählige Sprachen übersetzt wurde und damit komplett auf Deutsch zur Verfügung steht.

ScratchJR ist eine kostenfreie App für Tablets (Android, iPad), die ohne Text verwendet werden kann.

Beide Anwendungen haben eine sehr hohe Qualität und sind technisch wie didaktisch sehr gut zu verwenden.

### **Nötige Vorerfahrungen und Vorkenntnisse der Lernbegleitung**

Generell sollte die Lernbegleitung beim Thema Programmierung ein Verständnis für grundlegende Programmierkonstrukte aufweisen. Sie sollte also wissen, wie eine Anweisung aussieht, wie verschiedene Anweisungen kombiniert werden können, und die Konstrukte Wiederholung (Schleife) und Verzweigung (z. B. if-Abfrage) kennen und anwenden können.

### **Nötige Vorerfahrungen und Vorkenntnisse der Kinder**

Auf Seiten der Kinder sind keine Vorkenntnisse zwingend erforderlich. Zur Vorbereitung wäre es sinnvoll, Abläufe von Alltagsprozessen (z. B. Zähneputzen) zu thematisieren, so dass die Kinder ein grundlegendes Verständnis davon bekommen, wie sie selbst verschiedene Anweisungen nacheinander ausführen und unter Umständen sogar wiederholen können.

### **Lernziele**

Beide Varianten Scratch und ScratchJR haben das Ziel, einen Einstieg in die Welt der Programmierung zu ermöglichen. Dabei geht es nicht um die Syntax von Programmiersprachen, sondern um deren grundlegende Konzepte sowie um einfache Algorithmen. Die Kinder erlernen, eigene Abläufe in einer Mikrowelt zu programmieren. Dabei wenden sie die Basiskonzepte Anweisung, Wiederholung, Verzweigung (auch in Abhängigkeit von Sensorwerten) an. Darüber hinaus stehen Algorithmen im Fokus.

### **Bezug zu den Zieldimensionen informatischer Bildung**

Mit beiden Scratch-Varianten wird der *Inhaltsbereich* „I2 Algorithmen & Programmierung“ fokussiert. Durch die kindgerechten Programmierumgebungen können selbstständig Algorithmen geplant und umgesetzt werden. Am Rande können auch Aspekte von „I4 Informatiksysteme“ und „I5 Informatik, Mensch und Gesellschaft“ thematisiert werden.

Bei den *Prozessbereichen* liegt der Fokus auf „P0 Anwenden & Explorieren“ sowie auf „P1 Modellieren & Implementieren“. Das Anwenden und Explorieren bezieht sich hierbei auf die Informatiksysteme Tablet bzw. Laptop wie auch auf die Arbeit mit einem komplexen Programmfenster mit mehreren Bereichen. Da die

Spiele zum Abschluss der Lerneinheit unbedingt präsentiert werden sollten, wird auch der Bereich „P4 Kommunizieren (& Kooperieren)“ erweitert.

## 2.8 Makey Makey von JoyLabzLLC

### Überblick

Makey Makey<sup>59</sup> wurde von Jay Silver und Eric Rosenbaum am MIT entwickelt. Es handelt sich dabei um eine Platine, mit deren Hilfe unzählige kreative Wege gefunden werden können, wie alltägliche Dinge in berührungsempfindliche Flächen verwandelt und zur Interaktion mit dem Computer benutzt werden können. So verwandelt sich ein Apfel, eine Banane oder ein Stück Knete z. B. in eine linke Maustaste.

Optisch erinnert Makey Makey an eine Spielkonsolensteuerung. Er besitzt standardmäßig 6 Eingänge: die Pfeiltasten für hoch, runter, rechts und links sowie die Leertaste und die linke Maustaste. Weitere 12 Möglichkeiten erlauben z. B. eine komplette Maussteuerung.

Das Set des Makey Makey besteht neben der Platine selbst aus Krokodilklemmen, Jumperkabeln und einem USB-Kabel.

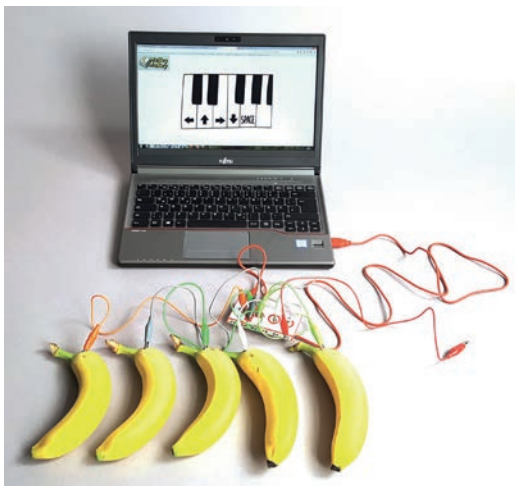
Eine einfache Steuerung lässt sich z. B. schon mit einem Apfel bauen. Der Makey Makey wird per USB-Kabel mit einem Rechner verbunden. Zwei Kabel mit Krokodilklemmen werden an den Makey Makey angeschlossen; eins zum Erden, eins z. B. für den Mausklick. Die Enden der Kabel werden zum einen z. B. mit dem Apfel und mit der Person selbst verbunden. Sobald die Person den Apfel berührt, wird ein Stromkreis geschlossen und der entsprechende Befehl, in diesem Fall

der Mausklick, ausgeführt. Auf diese Art und Weise können unterschiedlichste Gegenstände zu Interaktionsgegenständen werden. Warum also nicht einmal das Computerspiel mit Obst steuern?

In Verbindung mit Programmierumgebungen wie Scratch können weitere informatische Aspekte besprochen werden.

### Zielgruppe

Die Zielgruppe des Makey Makey spannt sich von Kindern ab dem 8. Lebensjahr bis hin zu Erwachsenen.



<sup>59</sup> <https://scratch.mit.edu>

### **Nutzerfreundlichkeit**

Nach der Einrichtung des Makey Makey als Eingabegerät am Rechner (dies geschieht genauso wie bei einer handelsüblichen Maus oder Tastatur) kann schon mit der Entdeckung neuer Interaktionsgegenstände begonnen werden. Der Makey Makey ist daher nach einer kurzen Einführung in die Handhabung sehr benutzerfreundlich.

### **Nötige Vorerfahrungen und Vorkenntnisse der Lernbegleitung**

Die Lernbegleitung muss im Vorfeld den Makey Makey als Eingabegerät am Rechner einrichten können. Darüber hinaus sollte sie mit der Funktionalität des Makey Makey vertraut sein, um bei Fragen oder Schwierigkeiten als Ansprechpartnerin/Ansprechpartner zur Verfügung zu stehen. Soll der Makey Makey auch in Verbindung mit Scratch eingesetzt werden, sind die entsprechenden Vorerfahrungen und Vorkenntnisse für Scratch ebenfalls nötig.

### **Nötige Vorerfahrungen und Vorkenntnisse der Kinder**

Vorerfahrungen auf Ebene der Kinder sind nicht notwendig. Kenntnisse über einfache Stromkreise sind aber für das Verständnis der Funktionalität hilfreich.

### **Lernziele**

Auf die informatische Bildung bezogen lernen Kinder mit dem Makey Makey auf experimentelle Art und Weise Interaktionen mit unterschiedlichen Programmen kennen. Darüber hinaus lernen sie, wie unterschiedlichste Gegenstände als Interaktionsgeräte benutzt werden können. Neben dem Aspekt der Eingabe- bzw. Interaktionsgeräte kann auch eine Verbindung zum Thema Stromkreis und Leitfähigkeit von Gegenständen gezogen werden.

In Verbindung mit Scratch können zusätzlich die Scratch-bezogenen Lernziele erreicht werden (siehe oben).

Auch mit dem Makey Makey kann wunderbar kollaborativ gearbeitet werden, so dass entsprechende Kompetenzen auf der Ebene der Zusammenarbeit gefördert werden.

### **Bezug zu den Zieldimensionen informatischer Bildung**

Auf der Ebene der Zieldimensionen informatischer Bildung wird mit dem Makey Makey vor allem der *Prozessbereich* „P0 Interagieren und Explorieren“ im Zusammenhang mit dem *Inhaltsbereich* „I4 Informatiksystem“ abgedeckt. Der Fokus der Anwendungen mit dem Makey Makey liegt auf der Entdeckung unterschiedlicher Interaktionsmöglichkeiten mit dem Computer bzw. Programmen auf dem Compu-

ter. Beim Arbeiten in Teams findet auch der *Prozessbereich* „P4 Kommunizieren & Kooperieren“ Anwendung.

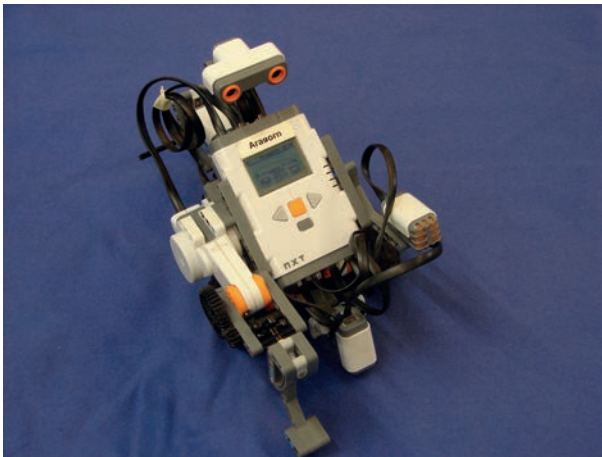
Bei der Verwendung des Makey Makey in Verbindung mit Scratch kommen entsprechend die dort angegebenen Inhalts- und Prozessbereiche ebenfalls zum Tragen.

## 2.9 LEGO Mindstorms (NXT & EV3)

### Überblick

Die LEGO-Mindstorms-Roboter<sup>60</sup> sind in der Reihe der hier dargestellten Informatiksysteme die komplexesten. Sie können als der große Bruder des LEGO WeDo angesehen werden und erobern bereits seit 1998 (in verschiedenen Versionen) schulische wie auch außerschulische Lernorte. Die aktuelle Version, der EV3, kam im September 2013 auf den deutschen Markt, aber auch der Vorgänger, der NXT, ist aktuell noch weit verbreitet.

Grundsätzlich sind die Lernziele des LEGO Mindstorms analog zu denen des WeDo sowohl im Bereich der Konstruktion wie auch der Programmierung des Roboters zu sehen. In beiden Bereichen (Hard- und Software) ist der LEGO Mindstorms entsprechend variabler und dadurch auch komplexer. Der EV3 umfasst insgesamt über 600 Einzelteile, wohingegen der WeDo mit nur 280 Bauteilen auskommt. Entscheidend sind hier vor allem die drei (statt nur einem) Motoren, abhängig von der Variante mit bis zu vier verschiedenen Sensoren ausgestattet (Farb-, Infrarot-, Gyro-, Berührungssensor oder Mikrofon). Daher können die Kinder beim Bau der Roboter neben vorgegebenen Modellen auch selbst sehr kreativ in der Schaffung eigener Modelle werden.



Ebenso im Bereich der Programmierung stehen den Lehrenden wie Lernenden vielfältige Möglichkeiten zur Verfügung. Ähnlich wie beim WeDo können die Roboter mit der Software NXG (für den NXT) bzw. EV3-Programmiersoftware (für PC und Tablet) grafisch programmiert werden. Hierbei stehen vielfältigere Befehle zur Verfügung, die auch differenzierter verwendet werden können (z. B. können bei Verwendung des Ultraschallsensors exakte Ent-

<sup>60</sup> <https://www.lego.com/de-de/mindstorms> [Zugriff am 05.03.2018]

fernungswerte verarbeitet werden). Darüber hinaus können die LEGO Mindstorms mit zahlreichen textuellen Programmiersprachen programmiert werden (u.a. Java, C++).

### **Zielgruppe**

Die LEGO Mindstorms sind vorrangig für Schülerinnen und Schüler an Sekundarschulen konzipiert und werden teilweise sogar an Universitäten für Einführungskurse in die Programmierung (dann meist in Java oder C) verwendet. Auch sind sie für Kinder am Ende der Grundschulzeit bzw. in kleinen Gruppen mit hohem Betreuungsschlüssel einsetzbar. Die Altersempfehlung für das aktuelle Modell EV3 liegt bei 10 Jahren.

### **Nutzerfreundlichkeit**

Die LEGO Mindstorms sind für den dauerhaften Einsatz mit Schülerinnen und Schülern konzipiert und entsprechend stabil gebaut. Auch die Software ist speziell für Programmieranfängerinnen und -anfänger konzipiert und einfach zu bedienen, wobei der Umgang mit klassischer Software (Öffnen, Speichern etc. von Dateien) im Vorfeld bekannt sein muss.

Eine häufig unüberwindbare Hürde beim Einsatz der LEGO Mindstorms ist der hohe Preis von aktuell 350 € für den EV3, welcher neben dem programmierbaren Brick, drei interaktiven Servomotoren, drei Sensoren (Farb-, Berührungs-, Infrarotsensor mit Fernbedienung) und entsprechenden Legoteilen zum Bau der vorgeschlagenen Modelle enthält.

### **Nötige Vorerfahrungen und Vorkenntnisse der Lernbegleitung**

Zum Einsatz der LEGO Mindstorms sollte die Lernbegleitung selbst über gute Grundlagenkenntnisse im Bereich der Programmierung verfügen. Sowohl in der grafischen wie auch textuellen Variante kommen sehr schnell Programmierkonstrukte wie Schleifen und Bedingungen zum Einsatz. Bei Verwendung textueller Programmiersprachen muss darüber hinaus auch die Syntax von der Lernbegleitung beherrscht werden, da sonst die Fehlersuche in den Schülerprogrammen kaum erfolgen kann.

### **Nötige Vorerfahrungen und Vorkenntnisse der Kinder**

Prinzipiell (gerade bei Verwendung der grafischen Programmiersprache) sind grundlegende Kenntnisse im Umgang mit klassischer Software ausreichend, da die Lernenden alle Programmierkonstrukte und auch die Syntax einer Programmiersprache während der Lerneinheit mit den LEGO Mindstorms erlernen können. Empfehlenswert, besonders bei der Verwendung einer textuellen Programmiersprache, ist allerdings, dass die Kinder bereits erste Schritte in der Programmie-

rung anhand eines der anderen Informatiksysteme erworben haben, um den Einstieg bzw. Übergang zu den LEGO Mindstorms zu erleichtern.

### **Lernziele**

Lernziele können mit Hilfe der LEGO Mindstorms sowohl im Bereich der Konstruktion eines Roboters (Technik) als auch im Bereich der Programmierung (Informatik) erworben werden.

Im Bereich der Roboterkonstruktion können technische Aspekte wie z. B. die Übersetzung mittels unterschiedlich großer Zahnräder oder auch das Kurvenfahren durch unterschiedliche Geschwindigkeiten auf beiden Antriebsachsen thematisiert werden.

Für die Programmierung sind im Bereich der imperativen Programmierung keine Grenzen gesetzt. Die Schülerinnen und Schüler können alle wichtigen grundlegenden Programmierkonstrukte mit Hilfe des Roboters erlernen. Auch wird ein besonderer Fokus auf den Umgang mit Sensorwerten gelegt.

### **Bezug zu den Zieldimensionen informatischer Bildung**

Im Fokus auch dieses Informatiksystems stehen die *Inhaltsbereiche* „I4 Informatiksysteme“ sowie „I2 Algorithmen & Programmierung“ und je nach Ausgestaltung der Lerneinheit auch „I5 Informatik, Mensch und Gesellschaft“. Wie beim LEGO WeDo kann mittels der LEGO Mindstorms ein Informatiksystem von den Lernenden selbst gestaltet (konstruiert und programmiert) werden. Durch die umfangreicheren Möglichkeiten können zudem komplexere Algorithmen (z. B. das automatisierte Folgen einer schwarzen Linie oder auch das Herausfinden aus einem unbekanntem Labyrinth) implementiert werden. Inwiefern auch gesellschaftliche Auswirkungen der Robotik thematisiert werden, liegt vorrangig an der Ausgestaltung der Lerneinheit, kann aber für das Thema Robotik hervorragend integriert werden. Aufgrund der zahlreichen Sensoren kann das Thema „I1 Information & Daten“ in einer Lerneinheit mit den LEGO Mindstorms aufgegriffen werden. Insgesamt werden die gleichen Zieldimensionen wie für den LEGO WeDo auf einem höheren Niveau angestrebt.

Bei den *Prozessbereichen* wird mit den LEGO Mindstorms vorrangig „P1 Modellieren & Implementieren“ fokussiert, wobei der Schwerpunkt in den meisten Umsetzungen auf dem Implementieren liegt. Das reine Explorieren hierbei ist weit schwieriger als bei LEGO WeDo bzw. bedarf eines höheren Grades an Anleitung. Inwiefern die Bereiche „P2 Begründen & Bewerten“, „P4 Kommunizieren & Kooperieren“ und „P5 Darstellen und Interpretieren“ zusätzlich vertieft werden, ist abhängig von der Ausgestaltung der Lerneinheit. Häufig arbeiten die Kinder und Jugendlichen zu zweit mit einem Roboter oder müssen sich im Team auf einen Algorithmus zur Lösung eines Problems (z. B. Weg aus dem Labyrinth) einigen,

wodurch auch Kompetenzen dieser Prozessbereiche erworben bzw. ausgebaut werden.

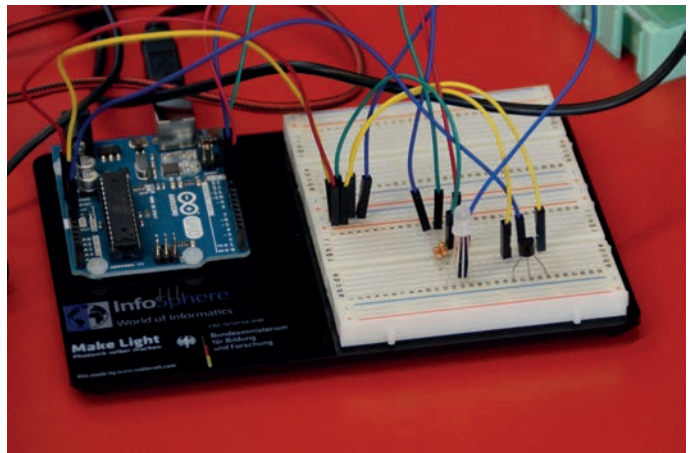
## 2.10 Arduino-Mikrocontroller mit Ardublock

### Überblick

Ardublock<sup>61</sup> ist – analog wie Scratch oder App Inventor – eine grafische Programmierumgebung, die am Computer genutzt wird. Die Besonderheit an Ardublock ist, dass nicht innerhalb einer Mikrowelt Objekte auf dem Bildschirm gesteuert werden, sondern ein Arduino-Mikrocontroller<sup>62</sup> und daran angeschlossene Aktoren (z. B. Motoren) und Sensoren (z. B. ein Lichtsensor) programmiert werden. Eben solche Kombinationen sind auch mit anderen Mikrocontrollern (wie z. B. dem Raspberry Pi) und auch anderen Programmierumgebungen (wie Scratch) umsetzbar. Der Arduino-Mikrocontroller und die beliebigen elektrischen Bauteile, die an diesen angeschlossen werden können, stellen dabei das flexibelste der hier beschriebenen Informatiksysteme dar, welches nach Belieben erweitert werden kann.

Ein überschaubarer Bausatz, der auch von Unterstufenschülerinnen und -schülern zusammengesetzt (gelötet) werden kann, sind die *Little Robot Friends* der Firma Aesthetec Studio. Diese umfassen wenige Bauteile und bieten dennoch dank Mikrofon, Lautsprecher und LEDs vielfältige und kreative Programmiermöglichkeiten. Spannend ist hierbei insbesondere die Kommunikation/Vernetzung mehrerer *Little Robot Friends* untereinander.

Eine Alternative sind *Little Bits* der Firma littleBits Electronics Inc. Mit dem Basisset können hauptsächlich Kompetenzen im Bereich der Elektronik (Physik, Elektrotechnik) erworben werden, indem elektrische Komponenten so zusammengesteckt werden, dass z. B. eine LED oder ein Signalgeber mittels Taster gesteuert werden kann. Ein Arduino-Erweiterungsset ermöglicht die Programmierung dieser



61 <http://blog.ardublock.com>

62 <https://www.arduino.cc>

Schaltungen. Die Programmierung des Arduino kann beispielsweise über ArduBlock passieren.

### **Zielgruppe**

Bei der Kombination der Software ArduBlock mit dem Arduino hängt die Zielgruppe stark von der Auswahl der Bauteile und von der Komplexität der Aufgaben ab. Erste Experimente mit ein paar LEDs und einem Motor (insbesondere mit den kindgerechten Bauteilen der *Little Bits*) sind bereits in der Grundschulzeit umsetzbar. Das eigentliche Ziel der selbstständigen Planung und Umsetzung eigener Projekte mit elektronischen Bauteilen und deren Programmierung entspricht allerdings eher der Kompetenzstufe von Schülerinnen und Schülern der Mittelstufe (ab Klasse 7 oder 8).

### **Nutzerfreundlichkeit**

Die Software Arduino mit der Erweiterung ArduBlock muss im Gegensatz zu den anderen Beispielen installiert werden. Dies hat den Nachteil des Mehraufwands, welcher je nach Verwaltungssystem der Einrichtung nur durch externe Administratoren leistbar ist. Gleichzeitig hat eine Offline-Software den Vorteil, dass sie, unabhängig von der Internetverbindung, stabil läuft und auch außerhalb der WLAN-Abdeckung genutzt werden kann.

Weitergehend ist ArduBlock ähnlich nutzerfreundlich wie Scratch, so dass die Lernenden schnell Erfolge sehen und nicht durch Syntaxfehler einer textuellen Programmiersprache belastet werden.

Die zugehörige Hardware ist sehr unterschiedlich und variiert von kindgerechten *Little Bits* bis zu klassischen Sensoren und Aktoren aus dem Elektronikhandel.

### **Nötige Vorerfahrungen und Vorkenntnisse der Lernbegleitung**

Dieses Lernkonzept erfordert einige Vorkenntnisse sowohl im Bereich der Elektrotechnik/Physik als auch der Informatik. Die Lernbegleitung muss die physikalischen Grundlagen elektrischer Schaltungen verinnerlicht haben, die Funktionsweise der verwendeten Bauteile kennen und diese auch zielgruppengerecht vermitteln können. Weiter müssen fundierte Programmierkenntnisse (mit dem Fokus auf die Besonderheiten von Mikrocontrollern) vorhanden sein. Insgesamt ist für diesen Ansatz die – im Vergleich zu den anderen Ansätzen – höchste Kompetenz auf Seiten der Lernbegleitung erforderlich.

### **Nötige Vorerfahrungen und Vorkenntnisse der Kinder**

Ein Lernkonzept mittels Arduino ist eine spannende Verbindung von Technik und Informatik, wobei in beiden Gebieten Vorkenntnisse bei den Lernenden vorhanden sein sollten, um diese nicht zu überfordern. So sollten physikalische Grund-



lagen (z. B. zu Stromkreisen) ebenso vorausgesetzt werden können wie einfache Algorithmen (z. B. die Arbeitsweise einer Ampelanlage) nachvollzogen und selbst nachgebaut werden können.

### **Lernziele**

Durch die Kombination aus einer Programmierumgebung (ArduBlock oder auch Arduino) und dem Arduino-Mikrocontroller kann eine umfassende MINT-Kompetenz abgedeckt werden. Technische und informatische Lernziele stehen dabei im Fokus, da die Funktionsweise verschiedener elektrischer Bauteile verstanden und diese sinnvoll in einer Schaltung kombiniert werden müssen, bevor der Mikrocontroller, der diese Schaltung steuert, programmiert werden kann. Aufgrund der beliebigen Komplexität der Bauteile und der zu entwickelnden Algorithmen, sind auch die Lernziele sehr variabel.

### **Bezug zu den Zieldimensionen informatischer Bildung**

Mit dem Konzept der Kombination von Hard- und Software lassen sich nahezu alle *Inhaltsbereiche* und *Prozesse* abdecken, wobei die Verteilung auf die verschiedenen Bereiche von der Umsetzung der Lerneinheit abhängig ist. So bedarf es beispielsweise für den Inhaltsbereich „13 Sprachen & Automaten“ einem expliziten Fokus auf automatengesteuerte Programmierung.

### 3 Empfehlungen

Abschließen möchten wir mit einer Empfehlung, wie die oben analysierten Informatiksysteme sinnvoll bei der Zielgruppe der Kinder in der Kita bzw. im Grundschulbereich eingesetzt werden können. So unterschiedlich Kinder, Schulen, Horte und auch die Lernbegleitungen sind, so verschieden sind auch die dazugehörigen Empfehlungen.

Die 10 oben analysierten Informatiksysteme haben unterschiedliche Schwerpunkte, auf welche Zieldimensionen informatischer Bildung sie abzielen und auf welches Vorwissen, auf Seiten der Lernbegleitung und auch der Kinder, sie aufbauen. Insgesamt ist aber allen Systemen gemein, dass sie vorrangig die Inhaltsbereiche „I2 Algorithmen & Programmierung“ und „I4 Informatiksysteme“ thematisieren und einen Fokus auf die Prozessbereiche „P0 Anwenden & Explorieren“ sowie „P1 Modellieren & Implementieren“ legen. Dazu wurden die 10 Systeme in drei Komplexitätsstufen unterteilt: Systeme mit geringer, mittlerer und höherer Komplexität.

In die Stufe „*Systeme mit geringer Komplexität*“ fallen die Systeme:

- Cubetto
- Bee-Bot
- KIBO
- Ozobot ohne visuelle Sprache

Diese Systeme können in der Regel autonom verwendet werden, das bedeutet, es wird kein weiteres technisches System (Computer, Tablet o. Ä.) benötigt. Alle vier eignen sich sehr gut für den Einsatz im Kita-Bereich bzw. zum Erstkontakt in der Grundschule und sprechen vor allen Dingen den Prozessbereich „P0 Anwenden und Explorieren“ an. Speziell der Cubetto kommt nahezu ohne Anleitung aus, da die Holzsteine die Bewegungsrichtung vorgeben und der Roboter die eingesteckten Befehle abarbeitet. Soweit kann dieses System von jeder Lernbegleitung unabhängig ihrer informatischen Vorbildung eingesetzt werden. Möchte man aber auch die zusätzliche Methode (grüner Baustein) einsetzen, bedarf es eines Grundverständnisses dieses Programmierkonstrukts. Beim Einsatz des Bee-Bot müssen die Kinder in der Lage sein, sich ihr (über die Tasten eingegebenes) Programm zu merken, oder die Lernbegleitung unterstützt sie z. B. durch externes Aufmalen

der Befehle auf Papier. Die beiden weiteren Systeme KIBO und Ozobot erhöhen die Komplexität durch den Einsatz von Sensoren bzw. einen höheren Umfang an Programmiermöglichkeiten.

Insgesamt weisen diese vier Systeme eine sehr geringe Einstiegshürde auf und benötigen nur eine geringe informatische Kompetenz der Lernbegleitung und keinerlei Vorwissen auf Seiten der Kinder. Cubetto und Bee-Bot sind für den kurzzeitigen Einsatz geeignet, da sie praktisch keine Einstiegshürde haben, aber auch nach wenigen Stunden an die Grenze ihrer Funktionen kommen. Wenn eine langfristige Beschäftigung mit dem System geplant ist, eignen sich eher der KIBO und der Ozobot.

Nachdem ein Erstkontakt stattgefunden hat oder auch beim Einstieg mit älteren Kindern, ist der Einsatz eines Systems mit mittlerer Komplexität zu empfehlen.

In die Stufe der „*Systeme mit mittlerer Komplexität*“ gehören:

- ScratchJr
- Dash & Dot
- WeDo
- Makey Makey
- Ozobot mit visueller Sprache (Besonderheit hier, dieses System kann die komplette Bandbreite der Komplexität abdecken)

Zum Einsatz dieser Systeme benötigt die Lernbegleitung weitere technische Ausstattung in Form von Tablets (Smartphones) oder Computern. Dies bedingt direkt eine andere Vorbildung auf Seiten der Lernbegleitung, da sie die Kinder im Umgang mit einem klassischen, also nicht speziell für junge Kinder entwickelten System unterstützen muss. Für den Kita-Bereich eignen sich Systeme, die über eine App auf einem Tablet gesteuert werden am besten, da die Bedienung eines Tablets aufgrund der Touch-Gesten wesentlich intuitiver ist als die Nutzung eines Computers (oder Laptops) mit Tastatur und Maus. Somit sind hier Dash & Dot und ScratchJR naheliegende Beispiele. Für den Einsatz der Systeme WeDo und Makey Makey spricht, dass auch das System selbst anpassbar ist. Daher kann hier der Inhaltsbereich „I4 Informatiksysteme“ auf einem anderen Niveau behandelt werden. Vor dem Einsatz dieser Systeme sollten die Kinder den generellen Umgang

mit einem Computer (bzw. Laptop) kennen, sie sollten z. B. wissen, wie sie Dateien öffnen und speichern.

Aufgrund der Verwendung einer Programmiersprache (bei allen Beispielen in dieser Kategorie) benötigt die Lernbegleitung je nach System erste oder auch weitergehende informatische Kompetenzen im Bereich der Programmierung. Grundlegende Programmierkonstrukte (z. B. Schleifen, Variablen und Verzweigungen) müssen der Lernbegleitung vor Einsatz der Systeme bekannt sein.

Systeme der dritten Kategorie eignen sich in erster Linie zur Vertiefung oder zum Einsatz bei älteren Lerngruppen in weiterführenden Schulen bzw. kleinen Gruppen mit hohem Betreuungsschlüssel, z. B. in AGs. Für den Kita-Bereich werden diese Systeme nicht empfohlen.

Zu den „*Systemen mit höherer Komplexität*“ zählen:

- Scratch
- LEGO Mindstorms
- Arduino-Mikrocontroller mit ArduBlock

Bei diesen Systemen kommt eine vollständige (grafische oder auch textuelle) Programmiersprache zum Einsatz. Auch wenn bei einer grafischen Programmiersprache (Puzzlebausteine statt Texteingabe) nicht die Syntax erlernt werden muss, so benötigt die Lernbegleitung ein höheres Fachwissen, sowohl im Bereich Programmierung als auch im Bereich Informatiksysteme. Da die Systeme dieser Kategorie aufgrund ihrer höheren Komplexität wesentlich umfangreichere Möglichkeiten mitbringen, kann hier eine langfristige und tiefgehende Auseinandersetzung mit den verschiedenen Zieldimensionen stattfinden. Insbesondere kann der Bereich „I2 Algorithmen und Programmierung“ mittels dieser Systeme umfassend behandelt werden.

## 4 Fazit

Aus der Analyse der hier vorgestellten Informatiksysteme ergibt sich die Empfehlung, im Kita- und Grundschulbereich mit einem System geringer Komplexität zu beginnen, um sowohl der Lernbegleitung als auch den Kindern durch eine geringe Einstiegshürde Selbstvertrauen im Umgang mit Informatiksystemen zu geben und erste wichtige Grundlagen im Umgang zu festigen. Anschließend können und sollten, je nach Altersstruktur und Vorwissen der Kinder und Lernbegleitung sowie Ausstattung des Lernortes, Informatiksysteme mit mittlerer und/oder höherer Komplexität eingesetzt werden, um die angelegten Kompetenzen spiralförmig zu vertiefen. Schwerpunkte des Kompetenzerwerbs mit den hier vorgestellten Informatiksystemen beziehen sich auf die Inhaltsbereiche „I2 Algorithmen & Programmierung“ und „I4 Informatiksysteme“ sowie die Prozessbereiche „P0 Anwenden & Explorieren“ und „P1 Modellieren & Implementieren“. Weitere Inhalts- und Prozessbereiche können natürlich auch mit abgedeckt werden. Die Verwendung der vorgestellten Informatiksysteme ist hier allerdings nicht zwingend notwendig.

So stellen Informatiksysteme eine von vielen Möglichkeiten dar, wie informatische Inhalte in der frühen Bildung vermittelt werden können. Weitere Möglichkeiten sind beispielsweise Ansätze ganz ohne Informatiksysteme, sog. Unplugged-Ansätze. Auch sie sind für eine frühe informatische Bildung gut geeignet.

Informatische Bildung ist nötig, um ein Grundlagenverständnis über Informatiksysteme zu erlangen, die in unserem Alltag eine immer größere Rolle spielen. Für eine Teilhabe an der von Digitalisierung geprägten heutigen Lebenswelt ist es elementar, dass alle Kinder die Chance auf eine frühzeitige Bildung im Bereich der Informatik erhalten.

Auch wenn es eine Vielzahl an Ansätzen zur Vermittlung von Informatik gibt, halten wir die nachhaltige Beschäftigung mit Informatiksystemen, die auf unterschiedliche Zieldimensionen abzielen (vom reinen Erkunden bis hin zur Entwicklung eigener Algorithmen) für unabdingbar. Zwar gibt es zurzeit keine Studien, die dies belegen, wir gehen aber davon aus, dass Kinder nur zu aktiven Mitgestalterinnen und Mitgestaltern unserer heutigen und vor allem zukünftigen digitalen wie auch realen Welt heranwachsen können, wenn im Kontext informatischer Bildung auch Informatiksysteme verwendet werden. Ansonsten fehlen den Kindern häufig der Anwendungsbezug bzw. der konkrete Kontext der erlernten informatischen Konzepte.

# Fazit und Ausblick – Wie die Stiftung „Haus der kleinen Forscher“ mit den Erkenntnissen umgeht

Stiftung Haus der kleinen Forscher



- 1 Empfehlungen aus der Expertise als Grundlage für die (Weiter-)Entwicklung der Stiftungsangebote
- 2 Ausblick und weitere wissenschaftliche Begleitung

## 1 Empfehlungen aus den Fachbeiträgen als Grundlage für die (Weiter-)Entwicklung der Stiftungsangebote

Alle Stiftungsangebote basieren auf fachlich fundierten Zieldimensionen für Kinder und pädagogische Fach- und Lehrkräfte im jeweiligen Themenbereich. Diese dienen der Stiftung als Orientierungsgrundlage für ihre inhaltlichen Angebote und spezifizieren, welche Ziele mit bestimmten Stiftungsangeboten erreicht werden sollen. Darüber hinaus bildet das jeweilige Modell der Zieldimensionen die theoretische Grundlage für die wissenschaftliche Begleitung und dessen empirische Überprüfung.

Bisher wurden die Zieldimensionen naturwissenschaftlicher Bildung im Elementar- und Primarbereich (vgl. Band 5 dieser Schriftenreihe, Stiftung Haus der kleinen Forscher, 2013), früher technischer Bildung (vgl. Band 7, Stiftung Haus der kleinen Forscher, 2015a) sowie früher mathematischer Bildung (vgl. Band 8, Stiftung Haus der kleinen Forscher, 2017a) erarbeitet und publiziert. Der vorliegende Band beinhaltet nun die Zieldimensionen informatischer Bildung im Elementar- und Primarbereich und die daraus resultierenden Empfehlungen für die inhaltliche Entwicklung der Stiftungsangebote zum Themenbereich „Informatik“.

Im Folgenden wird beschrieben, wie die Stiftung „Haus der kleinen Forscher“ die Empfehlungen der Fachexpertinnen und -experten aufgreift und umsetzt, um ein erstes Angebot im Bereich informatischer Bildung für drei- bis zehnjährige Kinder bzw. die begleitenden pädagogischen Fach- und Lehrkräfte in Kita, Hort und Grundschule zu entwickeln. Dabei wurde besonders darauf geachtet, Praxisnähe herzustellen und die Pädagoginnen und Pädagogen in ihrer Rolle als Lernbegleitung in der Umsetzung informatischer Bildungsinhalte zu stärken.

Folgende Zieldimensionen informatischer Bildung verfolgt die Stiftung auf Ebene der Kinder (vgl. Empfehlungen Kapitel 3 und Abbildung 26):

- Motivation, Interesse und Selbstwirksamkeit im Umgang mit Informatik
- Informatische Prozessbereiche
- Informatische Inhaltsbereiche

Auf Ebene der pädagogischen Fach- und Lehrkräfte stellt die Stiftung die folgenden von Bergner et al. empfohlenen Zieldimensionen in den Vordergrund (vgl. Empfehlungen Kapitel 4 und Abbildung 35):

- Motivation, Interesse und Selbstwirksamkeit bezogen auf informatische Bildung
- Informatische Prozessbereiche
- Informatische Inhaltsbereiche
- Informatikdidaktische Kompetenzen
- Einstellungen, Haltungen und Rollenverständnis in Bezug auf die Gestaltung informatischer Bildung

Alle inhaltlichen Formate der Stiftung zielen darauf ab, die Entwicklung von Kindern im Kita- und Grundschulalter in den entsprechenden Zieldimensionen zu stärken. Die meisten Stiftungsangebote unterstützen zunächst die pädagogischen Fach- und Lehrkräfte, die dann die Mädchen und Jungen in den Bildungseinrichtungen in ihrer Auseinandersetzung mit der Thematik begleiten und kindliche Lern- und Entwicklungsprozesse fördern. Die Stiftung bietet dabei einen praxisnahen Ansatz, der es den Pädagoginnen und Pädagogen ermöglicht, ihre Kenntnisse und Kompetenzen zu erweitern und in der alltäglichen Arbeit mit den Kindern einzusetzen.

Das Ziel der frühen informatischen Bildung ist es, den Kindern informatische Grunderfahrungen zu ermöglichen und ein informatisches Grundlagenverständnis zu entwickeln bzw. zu fördern. Im Folgenden sind die einzelnen Zieldimensionen und ihre konkrete Umsetzung in den Angeboten der Stiftung für Fach- und Lehrkräfte sowie Kinder ausführlich beschrieben.

### **1.1 Motivation, Interesse und Selbstwirksamkeit im Umgang mit Informatik**

Die erste von Bergner et al. empfohlene Zieldimension lautet „Motivation, Interesse und Selbstwirksamkeit im Umgang mit Informatik“. Sie gilt für Kinder und pädagogische Fach- und Lehrkräfte gleichermaßen und unterstreicht die Bedeutung von Begeisterung, Neugier und Interesse als wesentlichem Schlüssel für einen positiven Zugang zur Informatik. Dazu gehören:

- Interesse an Informatik(systemen)
- Motivation im Umgang mit informatischen Fragestellungen



- Selbstwirksamkeitserwartung im Umgang mit Informatik
- für Fachkräfte zusätzlich Motivation sowie Selbstwirksamkeitserwartung in Bezug auf die Gestaltung informatischer Bildung

Im Hinblick auf die motivationalen und emotionalen Aspekte ist nicht auszuschließen, dass es im Bereich der Informatik eine Diskrepanz zwischen den pädagogischen Fach- und Lehrkräften einerseits und den Kindern andererseits gibt. Kinder sind, zumindest was die Nutzung digitaler Geräte betrifft, meist sehr motiviert und interessiert. Sie werden dabei bislang vor allem durch das eigene Elternhaus geprägt, wobei Kinder aus bildungsnahen Familien häufiger mit Vorbehalten und Ängsten konfrontiert werden, was die Nutzung dieser Geräte betrifft (Chaudron, 2015; vgl. auch Bergner et al. in diesem Band, Abschnitt 2.1.1). Ähnliche Vorbehalte können auch bei Pädagoginnen und Pädagogen auftreten, wenn es um die Nutzung von digitalen Geräten in den Einrichtungen und die Auseinandersetzung mit informatischen Themen und Fragestellungen geht. Fehlende oder nicht ausreichende Kompetenzen im Bereich der informatischen Bildung können zu Unsicherheiten im Umgang mit der Thematik und somit auch zu einer Vermeidung führen.

Nach einer repräsentativen Umfrage der Stiftung „Haus der kleinen Forscher“ befürworten jedoch 75 % der befragten Fachkräfte das Erlernen eines verantwortungsvollen Umgangs mit digitalen Geräten in der Kita (Stiftung Haus der kleinen Forscher, 2017c). Die große Mehrheit der pädagogischen Fachkräfte sieht einen verantwortungsvollen Einsatz digitaler Geräte also durchaus positiv.

Die pädagogischen Fach- und Lehrkräfte dienen zudem als Vorbild für die Kinder, was fraglos auch für das Erleben von und die Einstellung zu Informatik gilt. Die Kinder profitieren hierbei von einem vorgelebten Interesse der pädagogischen Bezugsperson und finden Ansprechpartnerinnen und -partner für ihre informatikbezogenen Fragen. Ein offener, angstfreier Umgang mit Informatik ermöglicht den Kindern, ein Interesse für das Thema aufzubauen und langfristig aufrechtzuerhalten.

Daher steht in der Umsetzung dieser Zieldimension eine offene und angstfreie Haltung von Kindern wie auch von pädagogischen Fach- und Lehrkräften gegenüber Informatik im Vordergrund. Die Pädagoginnen und Pädagogen sollen Freude an der Gestaltung informatischer Bildung im pädagogischen Alltag entwickeln und motiviert werden, zusammen mit den Kindern informatischen Fragestellungen nachzugehen.

### *Umsetzung dieser Zieldimension in den Angeboten der Stiftung*

Für die Stiftung „Haus der kleinen Forscher“ stehen der Spaß und die Freude am Entdecken und Verstehen unserer Welt im Vordergrund. Die Auseinandersetzung mit alltäglichen Fragen zu Natur und Technik soll die Neugier, Lern- und Denkfriede der Mädchen und Jungen fördern. Auch im Bereich der Informatik sollen sich die Kinder als kompetent und selbstwirksam in ihrem Alltag und in einer zunehmend digitalisierten Welt erleben.

Bei der Umsetzung der Zieldimension „Motivation, Interesse und Selbstwirksamkeit“ ist es wichtig, die Kinder wie auch die pädagogischen Fach- und Lehrkräfte an die Auseinandersetzung mit informatischen Fragestellungen heranzuführen und ihnen dabei positive Grunderfahrungen zu ermöglichen. Ein Ziel dabei ist, sie für den gestalterischen und problemlösenden Charakter der Informatik zu begeistern und keinen rein rezeptiven Umgang mit digitalen Geräten zu fördern. Insbesondere die Pädagoginnen und Pädagogen sollen erfahren, dass sich informatische Themen nicht nur auf Computer, Smartphone etc. beziehen, sondern vielfältig in ihrem eigenen Alltag und dem der Kinder vorkommen. Gewinnstrategien entwickeln, Handlungsschritte ausführen oder Geheimnisse wahren sind nur einige Situationen, in denen sich informatische Lerngelegenheiten erkennen lassen. Das Erkennen solcher Situationen wird somit einfacher und eventuelle Vorbehalte gegenüber der Informatik abgebaut.

Die Fortbildungen, die die Stiftung konzipiert und die in den regionalen Netzwerken durch die Trainerinnen und Trainer bundesweit durchgeführt werden, zielen in ihrer Gestaltung stets darauf ab, den pädagogischen Fach- und Lehrkräften einen positiven Zugang zur Thematik zu ermöglichen und eine offene, angstfreie Haltung zu entwickeln. Im Bereich der informatischen Bildung hat die Stiftung ein Fortbildungsangebot zum Thema „Informatik entdecken – mit und ohne Computer“ entwickelt.



Um eventuellen Vorbehalten gegenüber der informatischen Bildung im Elementar- und Primarbereich zu begegnen und die Begeisterung und das Interesse an der Thematik zu stärken, verfolgt die Stiftung bei der Konzeption der Informatik-Angebote vor allem den Zugang ohne Einsatz von digitalen Geräten („unplugged“; vgl. Bergner et al. in diesem Band, Abschnitt 2.3.1). Außerdem ist die Umsetzung der

informatischen Bildung somit unabhängig von den finanziellen Möglichkeiten der pädagogischen Einrichtungen. Wichtige Grundlagen der Informatik lassen sich mit Papier und Stift, mit Alltagsmaterialien oder mit reinem Körpereinsatz erfahren. Die Fortbildung „Informatik entdecken – mit und ohne Computer“ ist deshalb so angelegt, dass Einrichtungen keine digitalen Geräte benötigen, um eigene Lerngelegenheiten zu gestalten. Auch ohne den Einsatz eines Computers kann gemeinsam mit den Mädchen und Jungen z. B. erforscht werden, wie Computer Zahlen sortieren, wie man Nachrichten verschlüsselt oder warum digitale Bilder aus Pixeln bestehen. Dieser Unplugged-Ansatz ist zum einen gut für das entdeckende und forschende Lernen mit jungen Kindern geeignet, zum anderen baut er Hemmschwellen bei den pädagogischen Fach- und Lehrkräften ab. Wie bei den anderen Bildungsbereichen der Stiftung auch kann die Förderung informatischer Kompetenzen spielerisch im Alltag geschehen.

Die Stiftung nutzt jedoch auch zwei weitere Zugänge zur Informatik, den softwarebasierten Einstieg (vgl. Bergner et al., Abschnitt 2.3.2) sowie physisch erfahrbare Einstiege in die Programmierung (Robotik; vgl. Bergner et al., Abschnitt 2.3.3). Die Expertengruppe geht davon aus, dass diese Ansätze besonders bei den Kindern sehr motivationsfördernd sind, da sie eine direkte Rückmeldung zu ihrer Arbeit bekommen. Für den softwarebasierten Zugang existieren grafische Programmierumgebungen mit geringer Komplexität, die auch ohne Lesekenntnisse und viel Vorerfahrung auf Seiten der Kinder genutzt und bedient werden können. Die Kinder lernen, das Programmieren als kreatives Werkzeug zu nutzen und ein eigenes „Produkt“ zu erschaffen (z. B. den Roboter Cubetto um einen Stuhl fahren lassen und diese Aufgabe mit der Bedienung des Programmierbretts lösen oder mit der Programmiersprache Scratch die Bewegungen von Figuren auf dem Display durch das Zusammensetzen grafischer Puzzleteile steuern.) Kindgerechte Robotersysteme ermöglichen das Programmieren in der physisch erfahrbaren Welt. Die Kinder können direkt mit solchen Informatiksystemen interagieren und bekommen in der Interaktion eine unmittelbare Rückmeldung. Unabhängig vom gewählten Zugang werden die Mädchen und Jungen durch die informatische Bildung in ihrer Problemlösekompetenz gestärkt und erfahren durch das Schaffen eigener Produkte Selbstwirksamkeit.

In der Fortbildung bekommen die Teilnehmerinnen und Teilnehmer konkrete Praxisideen, die auch den Bezug der Informatik zum Alltag verdeutlichen. Ebenso sind zahlreiche anschauliche Ideen, die den Einstieg erleichtern, in den pädagogischen Materialien wie z. B. den Entdeckungskarten sowie in der Themenbrochure (Stiftung Haus der kleinen Forscher, 2017b) enthalten. Die pädagogischen Fach- und Lehrkräfte sollen so lernen, informatische Inhalte in Alltagssituationen zu erkennen. Dieser Praxisbezug stärkt sowohl die Motivation als auch ihr Selbstvertrauen. Die Praxisbeispiele sind außerdem so gewählt, dass sie für die Kin-

der und ihre Lernbegleitung herausfordernd, aber auch erfolgreich zu bewältigen sind, um ihre Freude an der Auseinandersetzung mit informatischen Frage- oder Problemstellungen zu wecken (vgl. Bergner et al., Abschnitt 3.2.1).

Die von der Stiftung durchgeführten Pilotfortbildungen mit pädagogischen Fach- und Lehrkräften haben gezeigt, dass nach der Teilnahme an der Informatik-Fortbildung die Motivation steigt, sich weiter mit dem Thema zu beschäftigen, und dass die Pädagoginnen und Pädagogen sich darauf freuen, informatische Bildung mit Kindern umzusetzen. Diese positive Einstellung zum Thema Informatik nimmt wiederum einen motivierenden Einfluss auf die Kinder.

## 1.2 Informatische Prozessbereiche

Die Zieldimension „Informatische Prozessbereiche“ wird von Bergner et al. sowohl auf Ebene der Kinder als auch auf Ebene der pädagogischen Fach- und Lehrkräfte als bedeutender Kompetenzbereich beschrieben. Die Prozessbereiche beschreiben demnach, auf welche Art und Weise die Kinder mit fachlichen Inhalten umgehen sollen (Abschnitt 3.3). Bei der Herleitung der einzelnen prozessbezogenen Kompetenzen orientierte sich die Expertengruppe an den von der Gesellschaft für Informatik vorgeschlagenen Standards für die Sekundarstufe I (GI – Gesellschaft für Informatik e. V., 2008) sowie existierender internationaler Curricula der frühen informatischen Bildung. Diese Bildungsstandards wurden um den Prozessbereich ‚Interagieren und Explorieren‘ erweitert, um die Bedeutung für den spielerisch erkundenden Umgang mit Informatiksystemen im Kita- und Grundschulbereich zu betonen (vgl. Bergner et al., Abschnitt 2.5.2). Somit empfehlen die Expertinnen und Experten die folgenden Prozessbereiche:

- Interagieren und Explorieren
- Modellieren und Implementieren
- Begründen und Bewerten
- Strukturieren und Vernetzen
- Kommunizieren und Kooperieren
- Darstellen und Interpretieren

Die Autorinnen und Autoren der Expertise betonen, dass die einzelnen Prozesse immer in Verknüpfung mit einem oder mehreren Inhaltsbereichen (siehe folgen-

der Abschnitt 1.3) erarbeitet und angewandt werden sollten. Prinzipiell können dabei alle Prozessbereiche mit allen Inhaltsbereichen verknüpft werden. Jedoch gibt es Kombinationen, die sinnvoller erscheinen als andere, vor allem im Hinblick auf das Alter der Kinder. Zudem haben viele Praxisbeispiele oft auch Bezüge zu mehreren Prozess- und Inhaltsbereichen, die somit gleichzeitig, wenn auch zu einem unterschiedlichen Grad, thematisiert werden können.

In der Expertise wird Informatik als konstruierende Wissenschaft beschrieben, d. h. die Entwicklung eines Informatiksystems unterliegt einem Konstruktionsprozess (Bergner et al., in diesem Band, Abschnitt 1.3). Dabei werden zyklische Modelle genutzt, um neue Produkte zu entwickeln oder bestehende anzupassen und weiterzuentwickeln. In der informatischen Bildung sollten laut der Expertengruppe dabei vor allem Prozesse der Planung und der Gestaltung im Mittelpunkt stehen. Ein entsprechender Gestaltungskreis für die Konstruktion von digitalen Artefakten wurde von Bergner et al. entwickelt und vorgeschlagen (Abschnitt 1.5.2). Diesem Gestaltungsprozess könnte vor allem in der frühen informatischen Bildung das Explorieren von Informatiksystemen vorausgehen. Dies ist ebenfalls ein zyklischer Prozess, bei dem anhand der Funktionen und dem Einsatzzweck die einzelnen Bestandteile und die Wirkmechanismen des Systems erkundet werden können. Ein entsprechender Explorationskreis wurde ebenfalls von der Expertengruppe vorgeschlagen (Abschnitt 1.5.1). Sowohl der Gestaltungs- als auch der Explorationskreis unterstreichen den prozesshaften Charakter der Informatik.

#### *Umsetzung dieser Zieldimension in den Angeboten der Stiftung*

In der Umsetzung der Zieldimension „Informatische Prozessbereiche“ verfolgt die Stiftung das Ziel, sowohl pädagogische Fach- und Lehrkräfte als auch Kinder mit dem prozesshaften informatischen Vorgehen vertraut zu machen. Sie sollen die zyklischen Arbeitsweisen kennen- und anwenden lernen. Im Mittelpunkt steht dabei die Vermittlung allgemeiner Denk- und Problemlösefähigkeiten, womit die informatische Bildung wiederum einen Beitrag zur Allgemeinbildung leisten kann.

Für alle von der Expertengruppe vorgeschlagenen Prozessbereiche wurden von der Stiftung Praxisideen entwickelt. Diese werden exemplarisch in der Fortbildung „Informatik entdecken – mit und ohne Computer“ thematisiert. Weitere Ideen für die informatische Bildung im Alltag mit den Kindern finden die Pädagoginnen und Pädagogen auf den Entdeckungskarten und der Themenbroschüre im dazugehörigen Materialpaket (Stiftung Haus der kleinen Forscher, 2017b). Der in der Expertise hergeleitete und ergänzte Prozessbereich „Interagieren und Explorieren“ nimmt im Themenangebot der Stiftung eine eher untergeordnete Rolle ein. Das hängt damit zusammen, dass sich die Stiftung entschieden hat, hauptsächlich Praxisideen zu entwickeln, die ohne digitale Geräte auskommen und stattdessen mit Alltagsmaterialien umsetzbar sind. Nichtsdestotrotz sieht auch

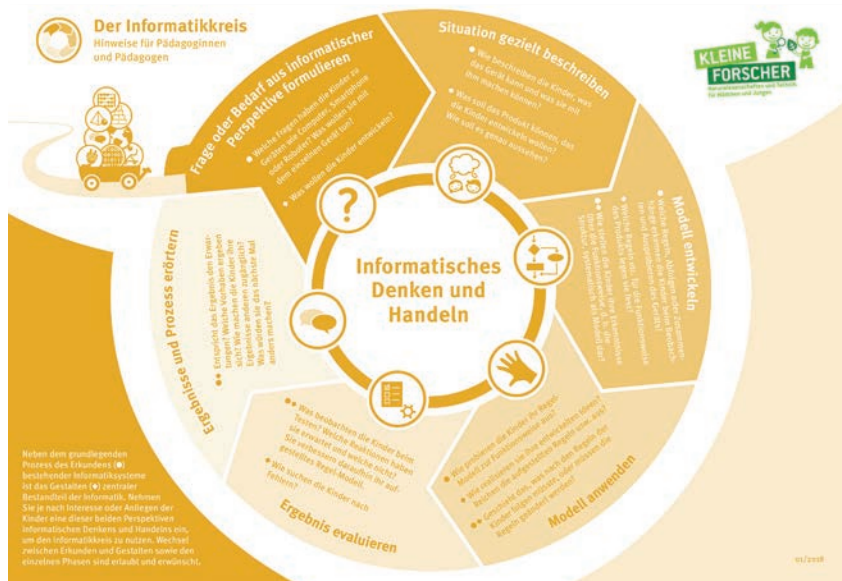
die Stiftung diesen Prozessbereich als wichtige Kompetenz an und verweist bei den Entdeckungskarten auf den jeweiligen Bezug zu Informatiksystemen, der in einigen Fällen auch das Erkunden dieses Systems beinhaltet. Je nach technischer Ausstattung der Einrichtungen kann dieser Prozessbereich also ebenfalls umgesetzt werden. Des Weiteren finden sich in der Themenbroschüre (vgl. Abbildung 46) entsprechende Bezüge wie beispielsweise im Kapitel „Roboter – vom Stauen zum Steuern“. Auf Basis der Fachempfehlung von Bergner und Müller (Kapitel Fachempfehlung Informatiksysteme in diesem Band) wurden entsprechende altersgerechte Systeme ausgewählt und vorgestellt. Bei diesen Informatiksystemen geht dem eigentlichen Programmieren das Entdecken und Ausprobieren der Robotersysteme voraus. Wie schon in der Expertise beschrieben, sind die einzelnen Praxisideen nicht immer eindeutig einem Prozessbereich zuzuordnen, sondern tangieren mehrere Bereiche gleichzeitig.



**Abbildung 46.** Titelseite und Inhaltsverzeichnis der Themenbroschüre „Informatik entdecken – mit und ohne Computer“ (Stiftung Haus der kleinen Forscher, 2017b)

Um die informatikbezogene Prozessorientierung zu verdeutlichen, wurde – analog zur Methode „Forschungskreis“, die in der frühen naturwissenschaftlichen Bildung Einsatz findet (vgl. Pädagogischer Ansatz der Stiftung in Stiftung Haus der kleinen Forscher, 2015b), sowie zum „Mathematikkreis“ (vgl. Stiftung Haus der kleinen Forscher, 2016) und zum „Technikkreis“ (vgl. Stiftung Haus der kleinen Forscher, 2017d) – zusammen mit Fachexpertinnen und -experten der „Informatikkreis“ entwickelt (vgl. Abbildung 46).

Der „Informatikkreis“ ist angelehnt am Explorations- und Gestaltungsprozess der Expertise und adaptiert die einzelnen Phasen für die Bedürfnisse der pädagogischen Fach- und Lehrkräfte im Elementar- und Primarbereich (siehe Abbildung 47). Mit Hilfe des Kreises soll der Prozess des informatischen Vorgehens verdeutlicht und unterstützt werden. Er gliedert sich in sechs Phasen: (1) Frage oder Bedarf aus informatischer Perspektive formulieren, (2) Situation gezielt beschreiben, (3) Modell entwickeln, (4) Modell anwenden, (5) Ergebnis evaluieren sowie (6) Ergebnisse und Prozess erörtern. Somit beinhaltet der „Informatikkreis“ Phasen des konkreten Handelns und Phasen der Dokumentation und Reflexion. Bei der Anwendung des Kreises kann man sowohl die Perspektive des Erkundens bestehender Informatiksysteme als auch die Perspektive des Gestaltens eines neuen Produktes einnehmen und ebenso zwischen den einzelnen Phasen dieser Perspektiven wechseln.



**Abbildung 47.** Der „Informatikkreis“ bildet den Prozess des informatischen Vorgehens ab und beinhaltet sowohl die Perspektive des Erkundens als auch des Gestaltens

Es ist geplant, den „Informatikkreis“ im Bildungsangebot der Stiftung zu verankern und somit die pädagogischen Fach- und Lehrkräfte dabei zu unterstützen, mit Kindern über informatische Fragestellungen zu sprechen und ihnen beim Entwickeln eines eigenen Modells zu helfen.

### 1.3 Informatische Inhaltsbereiche

Die Zieldimension „Informatische Inhaltsbereiche“ wird von Bergner et al. ebenfalls sowohl auf Ebene der pädagogischen Fach- und Lehrkräfte als auch auf Ebene der Kinder priorisiert. Die Inhaltsbereiche charakterisieren die informatischen Inhalte, die von den Pädagoginnen und Pädagogen sowie den Kindern erworben werden sollen (Abschnitt 3.3). Wie bei den informatischen Prozessbereichen orientierten sich die Expertinnen und Experten bei der Ausgestaltung dieses Kompetenzbereiches an den vorgeschlagenen Standards für die Sekundarstufe I der Gesellschaft für Informatik (GI, 2008). Sie unterscheiden dabei die folgenden fünf Inhaltsbereiche:

- Information und Daten
- Algorithmen und Programmierung
- Sprachen und Automaten
- Informatiksysteme
- Informatik, Mensch und Gesellschaft

Die Orientierung an den GI-Standards für Kinder im Kita- und Grundschulalter ermöglicht über die gesamte Bildungskette hinweg eine Kohärenz, die die Voraussetzung dafür schafft, aufbauend auf informatischen Alltags- und Spielerfahrungen im Elementar- und Primarbereich informatische Kompetenzen in den weiterführenden Schulen systematisch weiterzuentwickeln.

Wie bereits erwähnt, sollen die Inhaltsbereiche immer mit ein oder mehreren Prozessbereichen bzw. können mehrere Inhaltsbereiche mit ein und demselben Prozessbereich verknüpft werden. Besonders geeignet für den Elementar- und Primarbereich sind laut der Expertengruppe die folgenden Kombinationen aus Inhalts- und Prozessbereichen (vgl. Bergner et al., in diesem Band, Abschnitt 3.4.1):

- Modellieren und Implementieren von Algorithmen und Programmen
- Interaktion mit und Exploration von Informatiksystemen
- Darstellen und Interpretieren von Information und Daten



■ Nachdenken über und Bewerten des Zusammenhangs von Informatik, Mensch und Gesellschaft

Bei der Auswahl dieser Kompetenzbereiche durch die Expertengruppe spielten neben fachdidaktischen und lern- und entwicklungspsychologischen Aspekten u. a. auch der Alltagsbezug, das Interesse von Kindern sowie der Beitrag zur Allgemeinbildung eine Rolle. Die priorisierten Kombinationen aus Inhalts- und Prozessbereichen sollen neben den fachlich informatischen Kompetenzen auch allgemeine Basiskompetenzen und damit einhergehend auch die Allgemeinbildung fördern.

*Umsetzung dieser Zieldimension in den Angeboten der Stiftung*

Mit der Entwicklung eines ersten Angebotes zur frühen informatischen Bildung sind nun alle vier MINT-Fächer im Bildungsangebot der Stiftung vertreten. Der Bildungsbereich Informatik soll dabei im Stiftungsangebot perspektivisch gleichwertig zu den Bereichen der mathematischen, naturwissenschaftlichen und technischen Bildung vertreten sein.

Im Gegensatz zu den bisherigen inhaltlich stärker differenzierten Fortbildungsthemen der Mathematik, Naturwissenschaften und Technik (z. B. „Mathematik in Raum und Form entdecken“ vs. „Zahlen, Zählen, Rechnen – Mathematik entdecken“ mit dem Fokus auf Zahlen und Operationen) wurden bei der Entwicklung eines Stiftungsangebotes für die informatische Bildung alle von der Expertengruppe vorgeschlagenen Inhaltsbereiche für das erste Fortbildungsangebot und die dazugehörigen Praxisideen berücksichtigt. Dabei können wie bei den Prozessbereichen einzelne Praxisideen durchaus mehreren Inhaltsbereichen zugeordnet werden. Bei der Entwicklung der Ideen wurde sowohl die praktische Umsetzung mit Kindern im Kita- und Grundschulalter bedacht als auch die von Bergner et al. priorisierten Kombinationen aus Inhalts- und Prozessbereichen berücksichtigt.

Die Präsenzfortbildung „Informatik entdecken – mit und ohne Computer“ sowie das dazugehörige Materialpaket (Stiftung Haus der kleinen Forscher, 2017b) mit der Themenbroschüre, den Entdeckungskarten (Abbildung 48) sowie einem Wimmelbild (Abbildung 49) unterstützen die pädagogischen Fach- und Lehrkräfte dabei, das Thema „Informatik“ im Alltag mit den Kindern zu entdecken und hieran die Entwicklung informatischer Fähigkeiten der Mädchen und Jungen zu fördern. In der Fortbildung wird der Bezug zu alltäglichen Informatiksystemen wie der Ampelanlage, dem Smartphone oder der vollautomatischen Waschmaschine verdeutlicht, um den pädagogischen Fach- und Lehrkräften und somit auch den Kindern die Welt der Informatik sichtbar und auf anschauliche Weise erfahrbar zu machen. Zudem lernen die Pädagoginnen und Pädagogen in der Fortbildung die Inhaltsbereiche der Informatik kennen und entdecken zusammen die dazugehörigen Praxisideen.



INFORMATIK ENTDECKEN – MIT UND OHNE COMPUTER



### EINMAL ROBOTER SEIN Abläufe planen und steuern

**Wo begegnet es uns im Alltag?**

Roboter nehmen den Menschen Arbeiten ab. Um dies zu tun, werden sie von Computerprogrammen gesteuert. Kinder kennen Roboter aus Geschichten oder Filmen. Dort erscheinen sie als faszinierende malträchtige menschenähnliche Wesen, die mit leuchtenden Augen, blinkenden Knöpfen und maschinellen Lauten kommunizieren und sich eigenständig, aber ruckartig bewegen.

**Darum geht's**

Die Mädchen und Jungen beschäftigen sich mit Robotern und deren Steuerung. Die Kinder schäufeln sowohl in die Rolle der Programmierenden als auch in die der Roboter. Sie planen mit Richtungs-befehlen einen Weg auf einem Raster und probieren als Roboter aus, ob sie auch am gewünschten Ziel ankommen.

**Das wird gebraucht**

- Materialien zum Roboter-Anzug-Basteln: Papprollen, Alufolie, Pappkartons etc.
- Malerkepp für drinnen, Straßenmalzeide und Seife für draußen
- Festeres Papier oder Klebezettel für die Befehlskarten
- Stifte, eventuell Schere
- Unterlage zum Legen der Befehlskarten



**WIE EIN ROBOTER (EINSTIMMUNG)**

Fragen Sie die Mädchen und Jungen, ob sie schon einmal einen Roboter gesehen haben. Wissen die Kinder, was ein Roboter ist? Was kann ein Roboter alles? Woher ist ein Roboter da? Besprechen Sie nicht den Mädchen und Jungen, dass Roboter Maschinen sind, die den Menschen bestimmte Arbeiten abnehmen. Was wünschen sich die Kinder von einem Roboter, der ihnen ganz allein gehört? Überlegen Sie gemeinsam mit den Mädchen und Jungen, wie Roboter eigentlich aussehen. Basteln Sie zusammen Roboter-Anzüge. Mit Papprollen über den Armen sind zum Beispiel nur bestimmte Bewegungen möglich. Wie bewegen sich die Kinder nun als Roboter oder wie sprechen sie miteinander? Welche Ideen haben die Mädchen und Jungen, wie der Roboter Informationen aufnimmt und was er dann tut? Erfinden und bauen Sie mit den Kindern „Sensoren“ für den Roboter, etwa zum Wahrnehmen von Geräuschen oder Lichtsignalen, und „Aktoren“ als Bewegungselemente, die die Reaktion des Roboters dann ausführen. Es kommt dabei nicht darauf an, dass diese Sensoren und Aktoren wirklich funktionieren – das ist mit Pappe, Klebeband und Schere kaum zu erreichen –, sondern darauf, dass die Mädchen und Jungen eigene Ideen entwickeln und sich überlegen, wie man sie umsetzen könnte.

---



INFORMATIK ENTDECKEN – MIT UND OHNE COMPUTER



**ROBOTER STEuern (AKTIVITÄT)**

Die Kinder steuern sich nun gegenseitig. Ein Kind ist dabei der Roboter und das andere gibt die Befehle. Dabei sollte es sich möglichst um kurze, knappe Anweisungen handeln, damit der Roboter sie versteht. Ist der Roboter immer, was er tun soll? Wie könnte das steuernde Kind seine Anweisungen präziser formulieren, so dass der Roboter sie fehlerfrei ausführen kann? Was darf man einem Roboter eigentlich befehlen bzw. was nicht und darf ein Roboter biese Dinge tun? Welche Regeln wollen die Mädchen und Jungen für das Steuern eines Roboters vereinbaren?

**Lernerfahrung:**  
Befehle, die eine Maschine ausführen soll, müssen eindeutig formuliert sein.



**ROBOTER LAUFT (AKTIVITÄT)**

Mit Malerkepp oder Kreide markieren Sie ein großes Raster auf dem Boden. Die Roboter-Kinder sollen nun von einem Start zu einem Zielfeld gelangen. Die Mädchen und Jungen überlegen sich dazu Befehle, die sie für bestimmte Bewegungen verwenden wollen, zum Beispiel: vorwärts, rechts, links, hinten, lachen. Auf festeres Papier oder kleine Klebezettel zeichnen die Kinder Symbole für diese Befehle. Mit diesen Karten wird nun ein „Programm“ erstellt, indem sie in einer bestimmten Reihenfolge angeordnet werden. Ein Roboter-Kind bekommt von einem Programmierer-Kind die Befehle der Roboter nach vorgegebener Reihenfolge zu dem Ziel? Was fällt den Mädchen und Jungen ein, um ihr „Programm“ zu verbessern? Wie lassen sich beispielsweise mehrere gleiche Symbole, die aufeinanderfolgen, zu einem Befehl zusammenfassen?

**Lernerfahrung:**  
Der Weg des Roboters lässt sich im Voraus planen. Einzelne Schritte können zu einem „Programm“ zusammengestellt werden. Durch Zeichen für Wiederholungen ist es möglich, die „Programme“ kürzer darzustellen.

**WISSENSWERTES FÜR INTERESSIERTE ERWACHSENE**

Schon seit langer Zeit ist es ein Traum der Menschen, intelligente Maschinen zu bauen. Die Robotik beschäftigt sich mit der Entwicklung und Steuerung solcher Roboter. Dabei werden Maschinen konzipiert, die mit Hilfe von Sensoren, Aktoren und Informationsverarbeitung Aufgaben übernehmen können. Diese Roboter werden mit von Menschen entwickelten Programmen gesteuert. Nicht nur Informatikerinnen und Informatiker setzen sich mit den Ideen der künstlichen Intelligenz auseinander. Denn neben den Aspekten, die sich mit den Funktionsweisen intelligenter Maschinen befassen, sind auch solche Fragen zu bedenken, die das Miteinander von intelligenten Maschinen, Mensch und Umwelt betreffen. – Was darf ein Roboter und was nicht?



Die Mädchen und Jungen programmieren den Weg des Roboters Roberto auf der Kinderwebsite [www.meine-forscherwelt.de](http://www.meine-forscherwelt.de) mit dem Spiel „Robbie-Roboter“.

**Abbildung 48.** Vorder- und Rückseite der Entdeckungskarte „Einmal Roboter sein. Abläufe planen und steuern“ (Stiftung Haus der kleinen Forscher, 2017b)

Neben der Präsenzfortbildung bietet die Stiftung „Haus der kleinen Forscher“ auch einen Online-Kurs „Informatik entdecken“, in dem die pädagogischen Fach- und Lehrkräfte ebenfalls die informatischen Inhaltsbereiche kennenlernen und Anregungen für die Praxis erhalten. Für Kinder im Grundschulalter besteht über Entdeckungskarten und die Kinder-Website [www.meine-forscherwelt.de](http://www.meine-forscherwelt.de) (derzeit „Fabios Flächen“, „Ronjas Roboter“) die Möglichkeit, eigenständig zu forschen und zu entdecken.



**Abbildung 49.** Wimmelbild zum Entdecken und Erkunden von Informatiksystemen im Alltag als Bildkarte oder Poster (Stiftung Haus der kleinen Forscher, 2017b)

## 1.4 Informatikdidaktische Kompetenzen

Als eine bedeutende Zieldimension auf Ebene der pädagogischen Fach- und Lehrkräfte empfehlen Bergner et al. die „informatikdidaktischen Kompetenzen“. Zusammen mit den prozess- und inhaltsbezogenen informatischen Kompetenzen bildet dieser Kompetenzbereich die professionelle Grundlage für eine wirksame informatische Bildung im Kita- und Grundschulbereich. Zu den fachdidaktischen Kompetenzen gehören:

- Informatikdidaktische Basiskompetenzen (Wissen über Ziele informatischer Bildung)

- Kompetenz zur Planung von informatischen Lernumgebungen und Lernsituationen
- Kontextuelle informatikdidaktische Handlungskompetenz
- Kompetenz zur Diagnose und Evaluation von informatischen Lernsituationen
- Pädagogisch-fachliche Kommunikationskompetenz mit Beteiligten

Das Ziel ist es, die pädagogischen Handlungsstrategien der Fach- und Lehrkräfte zu stärken. Das Wissen über die Ziele informatischer Bildung, verschiedene didaktische Methoden sowie die Gestaltung effektiver Lernumgebungen ist dabei von großer Bedeutung. Zudem ist es vor allem im Elementar- und Primarbereich wichtig, Alltags- und Spielsituationen mit informatischem Gehalt zu erkennen und zu nutzen, um informatische Lerngelegenheiten zu schaffen. Dabei sollten die pädagogischen Fach- und Lehrkräfte in der Lage sein, den Entwicklungsstand der einzelnen Kinder einzuschätzen und die Unterstützung entsprechend anzupassen.

#### *Umsetzung dieser Zieldimension in den Angeboten der Stiftung*

Die Stiftungsangebote zielen darauf ab, die pädagogischen Fach- und Lehrkräfte mit konkreten pädagogischen Handlungsansätzen vertraut zu machen, die sie dann nutzen können, um die Kinder in ihren Lernprozessen zu unterstützen. Zu Beginn der Präsenzfortbildung „Informatik entdecken – mit und ohne Computer“ steht zunächst das Entdecken von Informatik und deren Bedeutung im Alltag im Fokus. Die pädagogischen Fach- und Lehrkräfte lernen die Definition des Begriffs Informatik und die Ziele informatischer Bildung kennen. Auch in der dazugehörigen Themenbroschüre (Stiftung Haus der kleinen Forscher, 2017b) wird die Bedeutung der Informatik und der informatischen Bildung thematisiert und erläutert, wie man als Lernbegleitung informatische Bildung unterstützen kann.

Im weiteren Verlauf der Fortbildung können die Teilnehmerinnen und Teilnehmer in Übungsphasen konkrete Methoden ausprobieren und dabei Praxiserfahrung sammeln. In anschließenden Reflexionsphasen wird thematisiert, wie das Gelernte in der praktischen Arbeit mit den Kindern umgesetzt werden kann. Die Pädagoginnen und Pädagogen erhalten somit Impulse, informatische Lernmöglichkeiten in ihren pädagogischen Alltag zu integrieren. Der Wechsel aus Praxiserfahrung und Reflexionsphasen ermöglicht es den pädagogischen Fach- und Lehrkräften, ihre informatikdidaktischen Kompetenzen weiterzuentwickeln und zu vertiefen.

Um den informatischen Bezug im Alltag zu verdeutlichen, werden in der Fortbildung praktische Beispiele vorgestellt und angewendet. Die Pädagoginnen und Pädagogen sollen so das informatische Potenzial von Spiel- und Alltagssituationen einschätzen lernen und zu nutzen wissen. Die fachliche Kompetenz, effektive Lernumgebungen zu erkennen und zu gestalten, sehen die Fachexpertinnen und -experten als eine grundlegende Gelingensbedingung für eine wirkungsvolle frühe informatische Bildung (vgl. Bergner et al., in diesem Band, Abschnitt 4.4.2).

Im Rahmen der Fortbildung lernen die pädagogischen Fach- und Lehrkräfte auch die drei Zugänge zur informatischen Bildung kennen. Wie bereits beschrieben, sind die von der Stiftung entwickelten Praxisideen zumeist ohne digitale Geräte umsetzbar, um den Pädagoginnen und Pädagogen wie auch in den anderen Themen beim „Haus der kleinen Forscher“ die Möglichkeit zu bieten, mit Alltagsmaterialien zu arbeiten und ihnen den Einstieg in die informatische Bildung zu erleichtern. Daher liegt der Fokus der Fortbildung und der pädagogischen Materialien auf dem Zugang ohne Computer. Den pädagogischen Fach- und Lehrkräfte ist es aber zudem auch möglich, die zwei weiteren Zugänge zur informatischen Bildung kennenzulernen: den softwarebasierten Einstieg in die Programmierung und die Programmierung mit altersgerechten Robotersystemen. Dabei erhalten sie Informationen über die verschiedenen Chancen, Grenzen und Herausforderungen der drei Zugänge in ihrer Rolle als Lernbegleitung, aber auch in Bezug auf die Kinder. Je nach Präferenz und Ausstattung der Einrichtungen können die Pädagoginnen und Pädagogen zwischen den drei Zugängen wählen und für die praktische Arbeit mit den Kindern nutzen. Dabei steht das entdeckende und forschende Lernen immer im Vordergrund.

## 1.5 Einstellungen, Haltungen und Rollenverständnis in Bezug auf die Gestaltung informatischer Bildung

Die informatikbezogene Haltung der pädagogischen Lernbegleitung hat einen großen Einfluss auf die informatische Förderung in Kita, Hort und Grundschule. Daher erachten die Autorinnen und Autoren der Expertise auf Ebene der pädagogischen Fach- und Lehrkräfte die Zieldimension „Einstellungen, Haltungen und Rollenverständnis in Bezug auf die Gestaltung informatischer Bildung“ als sehr wichtig. Zu dieser Zieldimension gehören:

- Überzeugungen zum Wesen der Informatik
- Überzeugungen zum Stellenwert informatischer Bildung in Kita, Hort und Grundschule

- Überzeugungen zum Lehren und Lernen von Informatik
- professionelles Rollen- und Selbstverständnis

Eine positive Einstellung zum Fach Informatik ist die Voraussetzung für eine gelingende Auseinandersetzung mit informatischen Themen im Alltag mit den Kindern. Die Einstellungen, Haltungen und Rollenerwartungen pädagogischer Fach- und Lehrkräfte in Bezug auf die Gestaltung informatischer Bildung stehen im Zusammenhang mit der unter Abschnitt 1.1 beschriebene Zieldimension „Motivation, Interesse und Selbstwirksamkeit bezogen auf informatische Bildung“. Ängste, Bedenken und Vorbehalte der Pädagoginnen und Pädagogen gegenüber informatischen Themen und Fragestellungen sowie der Nutzung digitaler Geräte oder fehlende Kompetenzen in diesem Bereich können zu Unsicherheiten im Umgang mit informatischen Themen und bis zur Vermeidung dieser führen. Das Interesse an Informatik und die Selbstwirksamkeit pädagogischer Fach- und Lehrkräfte bezüglich informatischer Bildung sind von großer Bedeutung für die Entwicklung eines informatischen Interesses bei Kindern.

Im Gegensatz zu anderen Bildungsbereichen findet die informatische Bildung im Elementar- und Primarbereich bisher kaum Beachtung. Es fehlen allgemeingültige Standards, wodurch der Stellenwert der informatischen Bildung noch stärker abhängig ist von individuellen Überzeugungen. Ein Ziel sollte es daher sein, dass auch die pädagogischen Fach- und Lehrkräfte der frühen informatischen Bildung einen angemessenen Stellenwert beimessen. Ein entsprechendes Hintergrundwissen und die Entwicklung von fachdidaktischen Kompetenzen bei den Pädagoginnen und Pädagogen sind nach Bergner et al. Voraussetzung für die Entwicklung einer positiven Einstellung gegenüber Informatik und für die Unterstützung von kindlichen Lernprozessen in diesem Themenbereich im Alltag.



*Umsetzung dieser Zieldimension  
in den Angeboten der Stiftung*

Das Stiftungsangebot im Bereich Informatik zielt auf eine positive Grundhaltung bei den pädagogischen Fach- und Lehrkräften gegenüber früher informatischer Bildung ab. Die zahlreichen Praxisbeispiele in den Präsenz- und Online-Fortbildungsformaten sowie in den dazugehörigen Materialien verdeutlichen den

großen Alltagsbezug der Informatik und erleichtern es den Pädagoginnen und Pädagogen, entsprechende Lerngelegenheiten zu erkennen und zu nutzen. Das in der Fortbildung vermittelte Fach- und fachdidaktische Wissen ermöglicht es den pädagogischen Fach- und Lehrkräften, sich mit der Thematik vertraut zu machen und somit auch eine positive Haltung zu entwickeln. Ebenso ist das Stiftungsangebot dahingehend ausgerichtet, die Fach- und Lehrkräfte mit dem kreativen und problemlösenden Charakter der Informatik vertraut zu machen, der ebenso einen Beitrag zur Allgemeinbildung leistet. Die praktische Auseinandersetzung mit der frühen informatischen Bildung soll eine positive Einstellung zum Thema fördern und somit auch den Stellenwert der Informatik im Kanon der anderen Bildungsbereichen stärken.

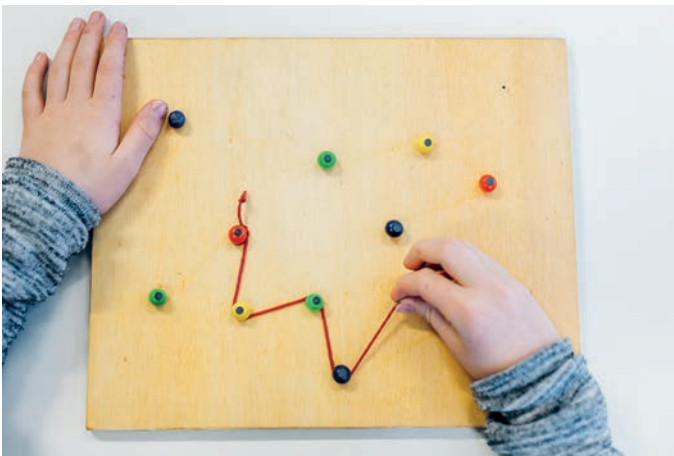
Bergner et al. empfehlen außerdem einen „kompetenten und kritisch-reflektierten Umgang mit digitalen Medien, insbesondere in der Funktion als Lernmedien der informatischen Bildung“ (Bergner et al. in diesem Band, Abschnitt 4.6, S. 210). Die Stiftung plant, ihre digitalen Lern-, Informations- und Serviceangebote für Pädagoginnen und Pädagogen erheblich auszubauen und regelmäßige Reflexionsanlässe zum Lernen mit digitalen Medien zu schaffen. Damit möchte sie pädagogische Fach- und Lehrkräfte bestmöglich in ihrem eigenen Lernen begleiten und sie dabei unterstützen, den Fragen und Herausforderungen, die die Digitalisierung in der Bildung für Kinder und Erwachsene mit sich bringt, kundig und reflektiert zu begegnen.

## 2 Ausblick und weitere wissenschaftliche Begleitung

Mit der Erweiterung des Stiftungsangebotes um den Themenbereich Informatik bietet die Stiftung mittlerweile Bildungsangebote in allen vier MINT-Domänen. Im Gegensatz zu den anderen Bildungsbereichen hat sich die Stiftung mit der informatischen Bildung im Elementar- und Primarbereich jedoch auf relatives Neuland begeben. Daher ist es hier besonders wichtig, die Umsetzung im praktischen Alltag mit den Kindern zu beobachten und Erkenntnisse zu sammeln, welche Konzepte mit Kindern im Kita- und Grundschulalter umsetzbar sind und wo Anpassungen im Bildungsangebot notwendig sind. Die Expertengruppe sieht die begleitende Evaluation der entwickelten Konzepte „als ein[en] wesentliche[n] Faktor für die erfolgreiche Implementierung früher informatischer Bildung in der Praxis“ (Bergner et al., in diesem Band, S. 251) an. Daher wäre auch im Bereich der informatischen Bildung eine wissenschaftliche Wirkungsforschung wünschenswert, um mehr Erkenntnisse zu den Wirkungen der Informatik-Angebote bei pädagogischen Fach- und Lehrkräften und Trainerinnen und Trainern sowie auch zu Wirkungen auf Ebene der Kinder zu sammeln.

Im Rahmen der Entwicklung der Informatik-Angebote hat die Stiftung verschiedene Pilotierungen und Erhebungen durchgeführt. Die von der Stiftung neu entwickelten Umsetzungsideen und Materialien wurden zunächst mit Piloteinrichtungen auf ihre Praxistauglichkeit getestet, bevor sie allen Einrichtungen zur Verfügung gestellt wurden. Auch die Präsenzfortbildung wurde mit ausgewählten Pädagoginnen und Pädagogen erprobt und evaluiert, um die Wirkungen zu untersuchen und Verbesserungen vornehmen zu können. Prä-post-Vergleiche mit den teilnehmenden Pädagoginnen und Pädagogen aus den Piloteinrichtungen

zeigten, dass nach Teilnahme an der Fortbildung der Stiftung zur informatischen Bildung (mit und ohne digitale Geräte) pädagogische Fach- und Lehrkräfte eine gestiegene Selbstwirksamkeitserwartung zur Umsetzung der Lerninhalte in der pädagogischen Praxis, ein höheres selbst eingeschätztes Wissen und höhere fachdidaktische Kompetenz zeigen. Darüber hinaus erkennen sie die Relevanz von Informatik





im Alltag im Vergleich zur Erhebung vor dem Besuch der Fortbildung nun deutlich stärker und gewinnen die Überzeugung, dass frühe informatische Bildung mit wie auch ohne Computer möglich ist.

Zudem werden fortlaufend im Rahmen von Monitoring-Maßnahmen die Rückmeldungen der an den Fortbildungen teilnehmenden pädagogischen Fach- und Lehrkräfte erhoben und ausgewertet. Dadurch erhält die Stiftung die Gelegenheit, auch über die Erprobungsphase hinweg, wichtige Erkenntnisse zur Umsetzung der Fortbildungen in der Fläche zu sammeln. Somit ist das Stiftungsangebot nicht nur wissenschaftlich fundiert, sondern profitiert auch von den Erfahrungen pädagogischer Fach- und Lehrkräfte aus der Praxis. Die Materialien werden regelmäßig überprüft und bei Bedarf überarbeitet und angepasst.

Um bereits frühzeitig einen Einblick in die Implementierung Informatischer Bildung in der Kita zu erhalten, plant die Stiftung zudem im Rahmen einer Evaluation zu untersuchen, inwieweit die pädagogischen Fach- und Lehrkräfte ihre Einstellungen und ihr Wissen durch die Teilnahme an der Fortbildung zur informatischen Bildung verändern bzw. erweitern und wie sie die Fortbildungsinhalte in der pädagogischen Praxis anwenden. Diese Evaluation soll 2018 in Kooperation mit einem Hamburger Kita-Träger stattfinden. Die Ergebnisse aus Monitoring und Evaluation zum Informatik-Angebot plant die Stiftung, im Monitoring-Bericht 2018/2019 zu veröffentlichen.

Begleitend zur inhaltlichen (Weiter-)Entwicklung der Angebote im Bereich Informatik wird die Stiftung auch weiterhin den fachlichen Austausch mit Expertinnen und Experten aus Wissenschaft und Praxis pflegen. Die Stiftungskonzepte werden regelmäßig auf Fachtagungen präsentiert und mit Fachleuten aus anderen Institutionen und Praxisinitiativen im Bereich Informatik diskutiert. Der Wissenschaftliche Beirat begleitet die Stiftung zu Forschungsfragen und zur fachlichen Fundierung des Stiftungsangebotes. Die Informatikdidaktik wird dort seit 2017 von Johannes Magenheim vertreten. Zudem fließen die Ergebnisse aus der kontinuierlichen Evaluation und dem Qualitätsmonitoring der verschiedenen Stiftungsangebote in Weiterentwicklungen innerhalb dieses Themenbereichs mit ein.

Eine wichtige Voraussetzung für eine qualitativ hochwertige Fortbildung der pädagogischen Fach- und Lehrkräfte bilden die bundesweit als Multiplikatorinnen und Multiplikatoren agierenden Trainerinnen und Trainer in der Initiative. Mit der Trainerakademie 2.0 hat die Stiftung „Haus der kleinen Forscher“ in 2016 eine dreijährige Offensive für gute frühe MINT-Bildung gestartet. Ziel des Projektes, welches von der aqtvator gGmbH gefördert wird, ist es, bundesweit die hohe Qualität der „Haus der kleinen Forscher“-Fortbildungen für pädagogische Fach- und Lehrkräfte zu sichern. Die Trainerakademie 2.0. soll die Ausbildung der Trainerinnen und Trainer in Zukunft noch stärker am individuellen Bedarf der einzelnen Weiterbildungsteilnehmerinnen und -teilnehmer ausrichten. Damit leistet dieses

Projekt im MINT-Bereich einen zentralen Beitrag zur Entwicklung der Lernbegleitung pädagogischer Fach- und Lehrkräfte. Im Rahmen der Trainerakademie 2.0 wird die Stiftung bis 2019 das bestehende Angebot mit neuen Qualifizierungsformaten erweitern und die individuellen Kompetenzen der Trainerinnen und Trainer noch gezielter unterstützen. Das gesamte Projekt baut auf Erfahrungen auf, die die Stiftung in den vergangenen Jahren im Bereich der Trainerqualifizierung für die frühe MINT-Bildung in Kitas, Horten und Grundschulen gesammelt hat. Das erweiterte Angebot für Trainerinnen und Trainer dient der Qualitätssicherung und -verbesserung und eröffnet neue Perspektiven für Deutschlands größte Bildungsinitiative auch im Bereich der frühen informatischen Bildung.

Zusammenfassend lässt sich festhalten, dass das Informatik-Angebot der Initiative „Haus der kleinen Forscher“ Bildungseinrichtungen im Elementar- und Primarbereich dabei unterstützen soll, sich als „Ort des forschenden Lernens“ auch im Bereich der frühen informatischen Bildung nachhaltig weiterzuentwickeln. Das Ziel ist es, förderliche Lernumgebungen für Kinder zu schaffen und sowohl bei Kindern als auch Pädagoginnen und Pädagogen eine positive Einstellung zu Informatik zu erreichen und ein Grundverständnis anzubahnen. Gemeinsam mit ihren Bezugspersonen, den pädagogischen Fach- und Lehrkräften, sollen die Kinder Freude am Entdecken und Verstehen ihrer Lebensumwelt haben.

# Literatur



## Einleitung –

Stiftung Haus der kleinen Forscher

- Anders, Y. & Ballaschk, I. (2014). Studie zur Untersuchung der Reliabilität und Validität des Zertifizierungsverfahrens der Stiftung „Haus der kleinen Forscher“. In Stiftung Haus der kleinen Forscher (Hrsg.), *Wissenschaftliche Untersuchungen zur Arbeit der Stiftung „Haus der kleinen Forscher“* (Band 6, S. 35–116). Schaffhausen: Schubi Lernmedien AG.
- DIVSI U9-Studie: Kinder in der digitalen Welt. (2015). DIVSI. Zugriff am 08.09.2017. Verfügbar unter: <https://www.divsi.de/publikationen/studien/divsi-u9-studie-kinder-der-digitalen-welt/>
- Eickelmann, B. (2015). Bildungsgerechtigkeit 4.0 – ICILS 2013: Grundlage für eine neue Debatte zur Bildungsgerechtigkeit. Zugriff am 08.09.2017. Verfügbar unter: <https://www.boell.de/de/2015/04/27/bildungsgerechtigkeit>
- GI – Gesellschaft für Informatik e. V. (2008). *Bildungsstandards Informatik für die Sekundarstufe I: Empfehlungen der Gesellschaft für Informatik e. V.* (Band 28). LOG IN.
- Stiftung Haus der kleinen Forscher (2013). *Wir lassen die Neugier in Kindern aufblühen. So wird Ihre Einrichtung ein „Haus der kleinen Forscher“*. Berlin: Stiftung Haus der kleinen Forscher. Verfügbar unter: [www.haus-der-kleinen-forscher.de](http://www.haus-der-kleinen-forscher.de)
- Stiftung Haus der kleinen Forscher (2015). *Pädagogischer Ansatz der Stiftung „Haus der kleinen Forscher“. Anregungen für die Lernbegleitung in Naturwissenschaften, Mathematik und Technik* (5. Auflage). Berlin: Stiftung Haus der kleinen Forscher. Verfügbar unter: [www.haus-der-kleinen-forscher.de](http://www.haus-der-kleinen-forscher.de)
- Stiftung Haus der kleinen Forscher (2017a). *Monitoring-Bericht 2016/2017 der Stiftung „Haus der kleinen Forscher“*. Berlin: Stiftung Haus der kleinen Forscher. Verfügbar unter: [www.haus-der-kleinen-forscher.de](http://www.haus-der-kleinen-forscher.de)
- Stiftung Haus der kleinen Forscher (Hrsg.) (2017b). *Zertifizierung für Kitas, Horte und Grundschulen. So wird Ihre Einrichtung ein „Haus der kleinen Forscher“*. Berlin: Stiftung Haus der kleinen Forscher.
- Stiftung Haus der kleinen Forscher (in Vorbereitung). *Wirkungen naturwissenschaftlicher Bildung im Elementarbereich. Wissenschaftliche Untersuchungen zur Arbeit der Stiftung „Haus der kleinen Forscher“* (Band 10). Opladen, Berlin, Toronto: Verlag Barbara Budrich.

## Zieldimensionen informatischer Bildung im Elementar- und Primarbereich –

Nadine Bergner, Hilde Köster, Johannes Magenheim, Kathrin Müller, Ralf Romeike, Ulrik Schroeder, Carsten Schulte

- Adams, J. C. & Webster, A. R. (2012). What do students learn about programming from game, music video, and storytelling projects? *Proceedings of the 43rd ACM Technical Symposium on Computer Science Education* (S. 643–648). ACM Press. doi:10.1145/2157136.2157319

- Albers, C., Magenheimer, J. & Meister, D. (Hrsg.) (2011). *Schule in der digitalen Welt: medienpädagogische Ansätze und Schulforschungsperspektiven* (Medienbildung und Gesellschaft) (1. Aufl.). Wiesbaden: VS Verlag für Sozialwissenschaften.
- Ambler, S. W. (1998). *Building object applications that work: your step-by-step handbook for developing robust systems with object technology* (Managing object technologies series). Cambridge, New York: Cambridge University Press; SIGS Books.
- Anders, Y., Hardy, I., Pauen, S., & Steffensky, M. (2013a). Zieldimensionen früher naturwissenschaftlicher Bildung im Kita-Alter und ihre Messung. In Stiftung Haus der kleinen Forscher (Hrsg.). *Wissenschaftliche Untersuchungen zur Arbeit der Stiftung „Haus der kleinen Forscher“*, (Band 5, S. 19–82). Schaffhausen: Schubi Lernmedien AG.
- Anders, Y., Hardy, I., Sodian, B., & Steffensky, M. (2013b). Zieldimension früher naturwissenschaftlicher Bildung im Grundschulalter und ihre Messung. In Stiftung Haus der kleinen Forscher (Hrsg.). *Wissenschaftliche Untersuchungen zur Arbeit der Stiftung „Haus der kleinen Forscher“*, (Band 5, S. 83–146). Schaffhausen: Schubi Lernmedien AG.
- Bagozzi, R. P. (2007). The legacy of the technology acceptance model and a proposal for a paradigm shift. *Journal of the Association for Information Systems* 8, 4, 244–254.
- Bell, T. (1999). A low-cost high-impact computer science show for family audiences (S. 10–16). Canberra, Australia: IEEE Comput. Soc. doi:10.1109/ACSC.2000.824374
- Bell, T., Rosamond, F. & Casey, N. (2012). Computer Science Unplugged and Related Projects in Math and Computer Science Popularization (Lecture Notes in Computer Science). In H.L. Bodlaender, R. Downey, F.V. Fomin & D. Marx (Hrsg.), *The Multivariate Algorithmic Revolution and Beyond* (S. 398–456). Springer Berlin Heidelberg. doi:10.1007/978-3-642-30891-8\_18
- Bell, T., Witten, I. H. & Fellows, M. (1998). Computer Science Unplugged ... off-line activities and games for all ages.
- Bender, E., Hubwieser, P., Schaper, N., Margaritis, M., Berges, M., Ohrndorf, L. et al. (2015). Towards a Competency Model for Teaching Computer Science. *Peabody Journal of Education*, 90 (4), 519–532. doi:10.1080/0161956X.2015.1068082
- Bender, E., Schaper, N., Caspersen, M. E., Margaritis, M. & Hubwieser, P. (2016). Identifying and formulating teachers' beliefs and motivational orientations for computer science teacher education. *Studies in Higher Education*, 41 (11), 1–16. doi:10.1080/03075079.2015.1004233
- Benz, C., Grüßing, M., Lorenz, J. H., Selter, C. & Wollring, B. (2017). Zieldimensionen mathematischer Bildung im Elementar- und Primarbereich. In Stiftung Haus der kleinen Forscher (Hrsg.), *Frühe mathematische Bildung – Ziele und Gelingensbedingungen für den Elementar- und Primarbereich. Wissenschaftliche Untersuchungen zur Arbeit der Stiftung „Haus der kleinen Forscher“* (Band 8, S. 32–177). Opladen, Berlin, Toronto: Verlag Barbara Budrich.
- Bergner, N. (2015). *Konzeption eines Informatik-Schülerlabors und Erforschung dessen Effekte auf das Bild der Informatik bei Kindern und Jugendlichen*. Aachen: Lehr- und Forschungsgebiet Informatik 9, RWTH Aachen University.
- Beutner, M., Kundisch, D., Magenheimer, J. & Zoyke, A. (2014). Support, Supervision, Feedback and Lectures Role in the use of the Classroom Response Systems PINGO. *Proceedings of World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education 2014* (S. 1210–1217).

- Biggs, J. B. & Collis, K. F. (1982). *Evaluating the quality of learning: the SOLO taxonomy (structure of the observed learning outcome)* (Educational psychology). New York: Academic Press.
- Blömeke, S. (Hrsg.) (2008). *Professionelle Kompetenz angehender Lehrerinnen und Lehrer: Wissen, Überzeugungen und Lerngelegenheiten deutscher Mathematikstudierender und -referendare; erste Ergebnisse zur Wirksamkeit der Lehrerbildung*. Münster: Waxmann.
- Blömeke, S., Kaiser, G. & Lehmann, R. (2011). Messung professioneller Kompetenz angehender Lehrkräfte: „Mathematics Teaching in the 21st Century“ und die IEA-Studie TEDS-M. (Fachdidaktische Forschungen). In H. Bayhuber, U. Harms, B. Muszynski, B. Ralle, M. Rothgangel, L.-H. Schön et al. (Hrsg.), *Empirische Fundierung in den Fachdidaktiken* (S. 9–26). Münster: Waxmann.
- Blumer, H. (1969). *Symbolic Interactionism: Perspective and Method*. Berkeley, Los Angeles: University of California Press.
- Boles, D. (2005). Spielerisches Erlernen der Programmierung mit dem Java-Hamster-Modell. *Capturing the interest of the uninterested* (S. 243–252). Gehalten auf der Lecture Notes in Informatics (LNI)-Proceedings 60, Wellington, New Zealand.
- Borowski, C. & Diethelm, I. (2009). Kinder auf dem Wege zur Informatik: Programmieren in der Grundschule. 13. *GI-Fachtagung – Informatik und Schule*. Gehalten auf der INFOS, Berlin: LOG IN.
- Borowski, C., Diethelm, I. & Mesaros, A.-M. (2010). Informatische Bildung im Sachunterricht in der Grundschule. Verfügbar unter: [www.widerstreit-sachunterricht.de](http://www.widerstreit-sachunterricht.de)
- Borowski, C., Diethelm, I. & Wilken, H. (2016). What children ask about computers, the Internet, robots, mobiles, games etc. (S. 72–75). ACM Press. doi:10.1145/2978249.2978259
- Börstler, J. & Schulte, C. (2005). Teaching object oriented modelling with CRC cards and role-playing games. *Proceedings WCCE* (Band 5). Verfügbar unter: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.437.3492&rep=rep1&type=pdf>
- Brabrand, C. & Dahl, B. (2009). Using the SOLO taxonomy to analyze competence progression of university science curricula. *Higher Education*, 58 (4), 531–549. doi:10.1007/s10734-009-9210-4
- Brandhofer, G. (2014). Ein Gegenstand „Digitale Medienbildung und Informatik“ – notwendige Bedingung für digitale Kompetenz? *R&E-SOURCE* (1).
- Brandt-Pook, H. & Kollmeier, R. (2008). *Softwareentwicklung kompakt und verständlich. Wie Softwaresysteme entstehen*. Wiesbaden: Vieweg + Teubner/GWV Fachverlage GmbH, Wiesbaden.
- Brauner, P. (2009). *Konzeption, Entwicklung und Analyse eines greifbaren Turtles in Hinblick auf die Steigerung der Computerselbstwirksamkeit von Schülerinnen und Schülern*. Diplomarbeit. Aachen: RWTH Aachen.
- Breier, N. (2004a). Informatik und die klassischen Naturwissenschaften – Partner oder Kontrahenten? Zugriff am 05.03.2018. Verfügbar unter: <https://www.ew.uni-hamburg.de/ueber-die-fakultaet/personen/breier/files/mnu-pdf.pdf>
- Breier, N. (2004b). Stand und Perspektive der informatischen Bildung. 1. *Fachtagung der GI-Fachgruppe Hamburger Informatiklehrerinnen und -lehrer*.
- Breiter, A., Welling, S. & Stolpmann, B. E. (2010). *Medienkompetenz in der Schule: Integration von Medien in den weiterführenden Schulen in Nordrhein-Westfalen* (Schriftenreihe Medienforschung der LfM). Berlin: Vistas Verl.

- Brennan, K. (2013). Learning Computing through Creating and Connecting. *Computer*, 46 (9), 52–59. doi:10.1109/MC.2013.229
- Brinda, T., Diethelm, I., Gemulla, R., Romeike, R., Schöning, J. & Schulte, C. (2016). Dagsstuhl-Erklärung: Bildung in der digitalen vernetzten Welt. GI – Gesellschaft für Informatik e. V. Zugriff am 29.08.2016. Verfügbar unter: <https://www.gi.de/aktuelles/meldungen/detailansicht/article/dagstuhl-erklaerung-bildung-in-der-digitalen-vernetzten-welt.html>
- Bröker, K., Kastens, U. & Magenheim, J. (2014). Competences of Undergraduate Computer Science Students. In T. Brinda, N. Reynolds & R. Romeike (Hrsg.), *Key Competencies in Informatics and ICT* (S. 77–96). Gehalten auf der KEYCIT 2014, Potsdam.
- Brown, N. C. C., Sentance, S., Crick, T. & Humphreys, S. (2014). Restart: The Resurgence of Computer Science in UK Schools. *ACM Transactions on Computing Education*, 14 (2), 1–22. doi:10.1145/2602484
- Buchholz, M., Saeli, M. & Schulte, C. (2013). PCK and reflection in computer science teacher education. *Proceedings of the 8th Workshop in Primary and Secondary Computing Education* (S. 8–16). ACM Press. doi:10.1145/2532748.2532752
- Bussmann, H. & Heymann, H. W. (1987). Computer und Allgemeinbildung. *Vierteljahrszeitschrift für Erziehung und Gesellschaft*, 27 (1), 2–39.
- Carlsen, W. (2002). Domains of Teacher Knowledge. In J. Gess-Newsome & N.G. Lederman (Hrsg.), *Examining Pedagogical Content Knowledge* (Band 6, S. 133–144). Dordrecht: Kluwer Academic Publishers. Zugriff am 18.01.2017. Verfügbar unter: [http://link.springer.com/10.1007/0-306-47217-1\\_5](http://link.springer.com/10.1007/0-306-47217-1_5)
- Caspersen, M. E. & Kolling, M. (2009). STREAM: A First Programming Process. *ACM Transactions on Computing Education*, 9 (1), 1–29. doi:10.1145/1513593.1513597
- Chaudron, S. (2015). *Young Children (0-8) and digital technology: a qualitative exploratory study across seven countries*. (European Commission, Joint Research Centre & Institute for the Protection and the Security of the Citizen, Hrsg.). Luxembourg: Publications Office. Zugriff am 17.01.2017. Verfügbar unter: <http://dx.publications.europa.eu/10.2788/00749>
- Claus, V. (1977). Informatik an der Schule: Begründungen und allgemeinbildender Kern (Schriftenreihe des Instituts für Didaktik der Mathematik der Universität Bielefeld). In H. Bauersfeld, M. Otte & H.G. Steiner (Hrsg.), *Informatik im Unterricht der Sekundarstufe II: Grundfragen, Probleme und Tendenzen mit Bezug auf allgemeinbildende und berufsqualifizierte Ausbildungsgänge* (Band 1, S. 19–33). Bielefeld.
- Claus, V. & Schwill, A. (Hrsg.) (2006). *Duden Informatik A – Z: Fachlexikon für Studium, Ausbildung und Beruf* (Duden) (4. Aufl. [Taschenbuchausg.]). Mannheim: Dudenverl.
- Computing at School Working Group. (2013). *Computing in the national curriculum: A guide for primary teachers*. Verfügbar unter: [www.computingatschool.org.uk/primary](http://www.computingatschool.org.uk/primary)
- Crutzen, C. K. M. (2000). Interaction, a world of differences. A vision on informatics from the perspective of gender studies. Open Universiteit Nederland.
- CSTA K-12 Computer Science Standards (2011). Verfügbar unter: <http://csta.acm.org/Curriculum/sub/K12Standards.html>
- Davis, E. (2008). Teaching Elementary Teachers' Ideas about Effective Science Teaching: A Longitudinal Study. *Proceedings of the 8th international conference on International conference for the learning sciences* (Band 1, S. 199–206).
- Department for Education. (2013). Statutory guidance – National curriculum in England: computing programmers of study.

- Deutscheschweizer Erziehungsdirektion. (2015). Lehrplan 21 zur Einführung in den Kantonen.
- Deutscheschweizer Erziehungsdirektoren-Konferenz (Hrsg.) (2016). *Lehrplan 21 – Medien und Informatik*. Luzern. Zugriff am 17.08.2016. Verfügbar unter: [http://v-ef.lehrplan.ch/container/V\\_EF\\_Modul\\_MI.pdf](http://v-ef.lehrplan.ch/container/V_EF_Modul_MI.pdf)
- Dickins, R., Nielsen, S., Barden, E. & Lamont, H. (2015). *Lift-the-flap computers and coding*. Usborne Publishing.
- DIVSI U9-Studie: Kinder in der digitalen Welt. (2015). *DIVSI*. Zugriff am 30.06.2015. Verfügbar unter: <https://www.divsi.de/publikationen/studien/divsi-u9-studie-kinder-der-digitalen-welt/>
- Döbeli Honegger, B. (2010). ICT im Hosensack – Informatik im Kopf? Gehalten auf der 25 Jahre Schulinformatik: Zukunft mit Herkunft, Österr. Computer-Ges.
- Döbeli Honegger, B. (2015). Digitale Medien im Lehrplan 21: Hoffnung oder Hydra? Zugriff am 05.03.2018. Verfügbar unter: <https://beat.doebe.li/talks/bern15/sldoo9.htm>
- Döbeli Honegger, B. (2016). *Mehr als 0 und 1: Schule in einer digitalisierten Welt*. hep der bildungsverlag.
- Dohmen, M., Magenheimer, J. & Engbring, D. (2009). Kreativer Einstieg in die Programmierung – Alice im Informatik Anfangsunterricht. *Informatische Bildung in Theorie und Praxis* (Band 13, S. 69–80). Gehalten auf der INFOS, LOG IN.
- Döhrmann, M., Kaiser, G. & Blömeke, S. (2010). Messung des mathematischen und mathematikdidaktischen Wissens: Theoretischer Rahmen und Teststruktur. In S. Blömeke, G. Kaiser & R. Lehmann (Hrsg.), *TEDS-M 2008 – Professionelle Kompetenz und Lerngelegenheiten angehender Primarstufenlehrkräfte im internationalen Vergleich*. Münster: Waxmann.
- Döhrmann, M., Kaiser, G. & Blömeke, S. (2012). The conceptualisation of mathematics competencies in the international teacher education study TEDS-M. *ZDM*, 44 (3), 325–340. doi:10.1007/s11858-012-0432-z
- Duncan, C. & Bell, T. (2015). A Pilot Computer Science and Programming Course for Primary School Students (S. 39–48). ACM Press. doi:10.1145/2818314.2818328
- Dworschak, M. (2015). Wie man Lehrer fernsteuert. *Der Spiegel*, 44/2015, 114–116.
- Eberle, F. (1996). *Didaktik der Informatik bzw. einer informations- und kommunikationstechnologischen Bildung auf der Sekundarstufe II: [Ziele und Inhalte, Bezug zu andern Fächern sowie unterrichtspraktische Handlungsempfehlungen]* (Pädagogik bei Sauerländer Schwerpunkt: Fachdidaktik, Berufsbildung und gymnasiale Bildung) (1. Aufl.). Aarau: Verl. für Berufsbildung Sauerländer.
- Eickel, J., Brauer, W., Claus, V., Deussen, P., Haake, W., Hosseus, W. et al. (1969). Zielsetzungen und Inhalte des Informatikunterrichts. In Im Auftrag des Fachausschusses „Ausbildung“ des Gesellschaft für Informatik (Hrsg.), *Zentralblatt für Didaktik der Mathematik* (Band 1, S. 35–43). Klett.
- Engbring, D. & Selke, H. (2013). Informatik und Gesellschaft als Gebiet der Informatik. *HDI 2012–Informatik für eine nachhaltige Zukunft: 5. Fachtagung Hochschuldidaktik der Informatik; 06.–07. November 2012, Universität Hamburg* (Band 5, S. 111–116). Gehalten auf der HDI Hochschuldidaktik der Informatik.
- Eulenberger, J. (2015). Die Persönlichkeitsmerkmale von Personen im Kontext des Lehrer\_innenberufs. Gehalten auf der DIW-SOEP – German Socio Economic Panel Study (SOEP).



- Fernandez, C. (2014). Knowledge Base for Teaching and Pedagogical Content Knowledge (PCK): Some useful Models and Implications for Teacher's Training. *Training Problems of education in the 21st century*, 60, 79–100.
- Fessakis, G. & Karakiza, T. (2011). Pedagogical beliefs and attitudes of computer science teachers in Greece. *Themes in Science & Technology Education*, 4 (2), 75–88.
- Feurzeig, W., Papert, S. A. & Lawler, B. (1970). Programming-languages as a conceptual framework for teaching mathematics. *Interactive Learning Environments*, 19 (5), 487–501. doi:10.1080/10494820903520040
- Fischer, G., Giaccardi, E., Ye, Y., Sutcliffe, A. G. & Mehandjiev, N. (2004). Meta-design: a manifesto for end-user development. *Communications of the ACM*, 47 (9), 33–38. doi:10.1145/1015864.1015884
- Flannery, L. P., Silverman, B., Kazakoff, E. R., Bers, M. U., Bontá, P. & Resnick, M. (2013). Designing ScratchJr: support for early childhood learning through computer programming (S. 1–10). ACM Press. doi:10.1145/2485760.2485785
- Forneck, H.-J. (1992). *Bildung im informationstechnischen Zeitalter: Untersuchung der fachdidaktischen Entwicklung der informationstechnischen Bildung* (1. Aufl.). Aarau: Sauerländer.
- Frank, H. & Meyer, I. (1974). Kybernetische Pädagogik Schriften 1958-1972. *Rechnerkunde: Elemente d. digitalen Nachrichtenverarbeitung u. ihrer Fachdidaktik* (Band 5). Stuttgart, Berlin, Köln, Mainz: Kohlhammer.
- Fuller, U., Riedesel, C., Thompson, E., Johnson, C. G., Ahoniemi, T., Cukierman, D. et al. (2007). *Developing a computer science-specific learning taxonomy* (S. 152–170). ACM Press. doi:10.1145/1345443.1345438
- Fulton, K. L. (1999). *How teachers' beliefs about teaching and learning are reflected in their use of technology: case study from urban middle schools*. Unpublished doctoral dissertation. University of Maryland.
- Gallenbacher, J. (2009). Abenteuer Informatik – „Abenteuer begreifen“ wörtlich gemeint. *INFOS*.
- Gander, W., Petit, A., Berry, G., Demo, B., Vahrenhold, J., McGettrick, A. et al. (2013). Informatics education: Europe cannot afford to miss the boat. *Report of the joint Informatics Europe & ACM Europe Working Group on Informatics Education*.
- Gesellschaft für Didaktik des Sachunterrichts (Hrsg.) (2013). *Perspektivrahmen Sachunterricht* (vollständig überarbeitete und erweiterte Ausgabe). Bad Heilbrunn: Verlag Julius Klinkhardt.
- GI – Gesellschaft für Informatik e. V. (2008). *Bildungsstandards Informatik für die Sekundarstufe I: Empfehlungen der Gesellschaft für Informatik e. V.* (Band 28). LOG IN.
- GI – Gesellschaft für Informatik e. V. (2016a). *Bildungsstandards Informatik für die Sekundarstufe II, Empfehlungen der Gesellschaft für Informatik e. V.* (Arbeitskreis „Bildungsstandards SII“). LOG IN.
- GI – Gesellschaft für Informatik e. V. (2016b). *Empfehlungen für Bachelor- und Masterprogramme im Studienfach Informatik an Hochschulen*. Bonn.
- Gibson, J. P. (2012). Teaching graph algorithms to children of all ages. In T. Lapidot, J. Gal-Ezer, M. E. Caspersen & O. Hazzan (Hrsg.), *Proceedings of the 17th ACM annual conference on Innovation and technology in computer science education* (S. 34–39). New York: ACM Press. Zugriff am 17.01.2017. Verfügbar unter: <http://dl.acm.org/citation.cfm?doid=2325296.2325308>

- Gil, J., Schwarz, B. B. & Asterhan, C. S. C. (2007). Intuitive moderation styles and beliefs of teachers in CSCL-based argumentation (S. 222–231). *Association for Computational Linguistics*. doi:10.3115/1599600.1599643
- Grafe, S. & Breiter, A. (2014). Modeling and Measuring Pedagogical Media Competencies of Pre-Service Teachers (M<sup>3</sup>K). In C. Kuhn, M. Toepper & O. Zlatkin-Troitschanskaia (Hrsg.), *Current International State and Future Perspectives on Competences Assessment in Higher Education* (S. 76–80). Gehalten auf der Report from the KoKoHs Affiliated Group Meeting at the AERA Conference, Philadelphia (USA).
- Gujberova, M. & Kalas, I. (2013). Designing productive gradations of tasks in primary programming education (S. 108–117). ACM Press. doi:10.1145/2532748.2532750
- Gutnick, A. L., Robb, M., Takeuchi, L., Kotler, J., Bernstein, L. & Levine, M. H. (2011). Always connected. Zugriff am 05.03.2018. Verfügbar unter: [http://www.joanganzcooneycenter.org/wp-content/uploads/2011/03/jgcc\\_alwaysconnected.pdf](http://www.joanganzcooneycenter.org/wp-content/uploads/2011/03/jgcc_alwaysconnected.pdf)
- Hartig, J. & Klieme, E. (2006). Kompetenz und Kompetenzdiagnostik. In K. Schweizer (Hrsg.), *Leistung und Leistungsdiagnostik* (S. 127–143). Berlin, Heidelberg: Springer-Verlag. Zugriff am 18.01.2017. Verfügbar unter: [http://link.springer.com/10.1007/3-540-33020-8\\_9](http://link.springer.com/10.1007/3-540-33020-8_9)
- Hartmann, S. & Schecker, H. (2005). Bietet Robotik Mädchen einen Zugang zu Informatik, Technik und Naturwissenschaft? – Evaluationsergebnisse zu dem Projekt „Roberta“. *Zeitschrift für Didaktik der Naturwissenschaften*, 11, 7–19.
- Hattie, J. (2009). *Visible learning: a synthesis of over 800 meta-analyses relating to achievement*. London, New York: Routledge.
- Hauser, B. & Rechsteiner, K. (2011). Frühe Mathematik: Geführtes Spiel oder Training? *4bis8*, 5, 28–30.
- Heimann, P., Otto, G. & Schulz, W. (1979). *Unterricht: Analyse und Planung* (Auswahl Reihe B) (10., unveränd. Aufl.). Hannover: Schroedel.
- Herczeg, M. (2009). *Software-Ergonomie: Theorien, Modelle und Kriterien für gebrauchstaugliche interaktive Computersysteme* (Lehrbuchreihe interaktive Medien) (3., vollst. überarb. und erw. Aufl.). München: Oldenbourg.
- Hettlage, R. & Steinlin, M. (2006). The Critical Incident Technique in Knowledge Management Related Contexts. Zürich: IngeniusPeoplesKnowledge.
- Heymann, H.-W. (1997). Allgemeinbildung als Aufgabe der Schule und als Maßstab für Fachunterricht. *Allgemeinbildung und Fachunterricht* (S. 7–17). Hamburg: Bergmann + Helbig.
- Hoppe, H. U. & Löthe, H. (1984). *Problemlösen und Programmieren mit LOGO. Ausgewählte Beispiele aus Mathematik und Informatik*. Wiesbaden: Vieweg+Teubner Verlag. Zugriff am 17.01.2017. Verfügbar unter: <http://nbn-resolving.de/urn:nbn:de:1111-20130418805>
- Hubwieser, P. (2007). *Didaktik der Informatik: Grundlagen, Konzepte, Beispiele; mit 68 Tabellen* (eXamen.press) (3., überarbeitete und erweiterte Auflage.). Berlin, Heidelberg: Springer.
- Hubwieser, P. & Broy, M. (1997). Ein neuer Ansatz für den Informatikunterricht am Gymnasium. *LOG IN*, (3/4), 42–47.
- Hubwieser, P., Magenheimer, J., Mühlhng, A. & Ruf, A. (2013). Towards a conceptualization of pedagogical content knowledge for computer science (S. 1–8). New York: ACM Press. doi:10.1145/2493394.2493395
- Hubwieser, P., Schubert, S., Armoni, M., Brinda, T., Dagiene, V., Diethelm, I. et al. (2011). Computer science/informatics in secondary education. *Proceedings of the 16th annual*

- conference reports on Innovation and technology in computer science education – working group reports* (S. 19–38). New York: ACM Press. doi:10.1145/2078856.2078859
- Humbert, L. & Puhlmann, H. (2004). Essential ingredients of literacy in informatics. *Informatics and Student Assessment. Concepts of emirical research and standardisation of measurement in the area of didactics* (Band 1, S. 65–76).
- Initiative D21 e. V. (2008). Bildung via Internet: Wie vernetzt sind unsere Kinder? Eine Sonderstudie im Rahmen des (N)Onliner Atlas. Zugriff am 05.03.2018. Verfügbar unter: [http://www.initiaved21.de/fileadmin/files/o8\\_NOA/FSC\\_Sonderstudie\\_72dpi.pdf](http://www.initiaved21.de/fileadmin/files/o8_NOA/FSC_Sonderstudie_72dpi.pdf)
- Isayama, D., Ishiyama, M., Relator, R. & Yamazaki, K. (2016). Computer Science Education for Primary and Lower Secondary School Students: Teaching the Concept of Automata. *ACM Transactions on Computing Education*, 17 (1), 1–28. doi:10.1145/2940331
- ISTE – The International Society for Technology in Education. (2008). Standards for Teachers. Zugriff am 05.03.2018. Verfügbar unter: <http://www.iste.org/standards/iste-standards/standards-for-students>
- Kafai, Y. B. & Burke, Q. (2014). *Connected code: why children need to learn programming* (The John D. and Catherine T. Macarthur foundation series on digital media and learning). Cambridge, Massachusetts: The MIT Press.
- Kay, A. & Goldberg, A. (1977). Personal Dynamic Media. *Computer*, 10 (3), 31–41. doi:10.1109/C-M.1977.217672
- Keil, R. (2012). Der Computer als Medium – Medien als Denkzeug des Geistes. In A. Knaut, C. Kühne, R. Rehak, S. Ullrich, C. Kurz & J. Pohle (Hrsg.), *Per Anhalter durch die Turing-Galaxis* (S. 147–219). Münster: Verl.-Haus Monsenstein und Vannerdat.
- Keil-Slawik, R. (1994). Softwareentwicklung: Die Gestaltung des Unsichtbaren: Kurzfassung in der Vorlesung Informatik und Gesellschaft.
- Kelleher, C. (2006, November). Motivating Programming: Using storytelling to make computer programming attractive to middle school girls. Dissertation. Pittsburgh: Carnegie Mellon University.
- Kelleher, C. (2015). Looking Glass (S. 271–271). Vortrag, gehalten auf der Proceedings of the 46th ACM Technical Symposium on Computer Science Education. New York: ACM Press. doi:10.1145/2676723.2691873
- Kelleher, C. & Pausch, R. (2005). Lowering the barriers to programming: A taxonomy of programming environments and languages for novice programmers. *ACM Computing Surveys*, 37 (2), 83–137. doi:10.1145/1089733.1089734
- Kind, A. (2015). Computing Attitudes: Will Teaching 2nd Grade Students Computer science Improve their Self-Efficacy and Attitude and Eliminate Gender Gaps?
- Klafki, W. (1993). Allgemeinbildung heute. *Pädagogische Welt* (Band 47, S. 28–33).
- Kleickmann, T. (2008). *Zusammenhänge fachspezifischer Vorstellungen von Grundschullehrkräften zum Lehren und Lernen mit Fortschritten von Schülerinnen und Schülern im konzeptuellen naturwissenschaftlichen Verständnis*. Münster: Westfälische Wilhelms-Universität Münster.
- Körber, B. & Peters, I.-R. (1988). Grundlagen einer Informatik der Didaktik. FU Berlin, Sommersemester.
- Kosack, W., Jeretin-Kopf, M. & Wiesmüller, C. (2015). Zieldimensionen technischer Bildung im Elementar- und Primarbereich. *Wissenschaftliche Untersuchungen zur Arbeit der Stiftung „Haus der kleinen Forscher“* (Band 7).

- Köster, H. (2006). *Freies Explorieren und Experimentieren: eine Untersuchung zur selbstbestimmten Gewinnung von Erfahrungen mit physikalischen Phänomenen im Sachunterricht* (Studien zum Physik- und Chemielernen). Berlin: Logos-Verl.
- Koubek, J., Schulte, C., Schulze, P. & Witten, H. (2009). Informatik im Kontext (InIK)-Ein integratives Unterrichtskonzept für den Informatikunterricht. *INFOS* (S. 268–279).
- Kramer, F. & Rabe-Kleberg, U. (2011). *Wissenschaftliche Untersuchungen zur Arbeit der Stiftung „Haus der kleinen Forscher“* (Band 2). Stiftung „Haus der kleinen Forscher“ (Hrsg.). Köln: Schubi-Verl.
- Krathwohl, D. R. (2002). A Revision of Bloom’s Taxonomy: An Overview. *Theory Into Practice*, 41 (4), 212–218. doi:10.1207/s15430421tip4104\_2
- Kultusministerkonferenz (2012). *Medienbildung in der Schule*.
- Kultusministerkonferenz (2015). *Ländergemeinsame inhaltliche Anforderungen für die Fachwissenschaften und Fachdidaktiken in der Lehrerbildung*.
- Kunter, M. & Baumert, J. (2011). *Professionelle Kompetenz von Lehrkräften. Ergebnisse des Forschungsprogramms COACTIV*. (W. Blum, U. Klusmann, S. Krauss & M. Neubrand, Hrsg.). Münster: Waxmann Verlag.
- Lambert, L. & Guiffre, H. (2009). Computer science outreach in an elementary school. *J. Comput. Small Coll*, 24, 118–124.
- Leonhardt, T. (2015). *Etablierung eines begabungsfördernden Lernumfelds für Mädchen im Bereich Informatik*. RWTH Aachen University.
- Levy, S. T. & Mioduser, D. (2008). Does it “want” or “was it programmed to...”? Kindergarten children’s explanations of an autonomous robot’s adaptive functioning. *International Journal of Technology and Design Education*, 18 (4), 337–359. doi:10.1007/s10798-007-9032-6
- Libow Martinez, S. & Stager, G. (2013). *Invent to learn: making, tinkering, and engineering in the classroom*. Torrance, Calif.: Constructing Modern Knowledge Press.
- Linck, B., Ohrndorf, L., Schubert, S., Stechert, P., Magenheim, J., Nelles, W. et al. (2013). Competence model for informatics modelling and system comprehension. *Global Engineering Education Conference (EDUCON’13)* (S. 85–93). IEEE. doi:10.1109/Edu-Con.2013.6530090
- Lindmeier, A. (2011). *Modeling and measuring knowledge and competencies of teachers: a threefold domain-specific structure model for mathematics* (Empirische Studien zur Didaktik der Mathematik). Münster: Waxmann.
- Liukas, L. (2015). *Hello Ruby: adventures in coding*. Macmillan.
- van Lück, W. (1986). *Informations- und kommunikationstechnische Grundbildung und der Computer als Medium im Fachunterricht*. Soest: Soester Verlagskontor.
- Maceli, M. & Atwood, M. E. (2011). From Human Crafters to Human Factors to Human Actors and Back Again: Bridging the Design Time – Use Time Divide. In M. F. Costabile, Y. Dittrich, G. Fischer & A. Piccinno (Hrsg.), *End-User Development* (Band 6654, S. 76–91). Berlin, Heidelberg: Springer Berlin Heidelberg. Zugriff am 16.01.2017. Verfügbar unter: [http://link.springer.com/10.1007/978-3-642-21530-8\\_8](http://link.springer.com/10.1007/978-3-642-21530-8_8)
- Magenheim, J. (2000). Informatiksysteme und Dekonstruktion als didaktische Kategorien: Theoretische Aspekte und unterrichtspraktische Implikationen einer systemorientierten Didaktik der Informatik. *GI-Tagung „Informatik – Ausbildung und Beruf 2000“*.
- Magenheim, J. (2008). Systemorientierte Didaktik der Informatik Sozio-technische Informatiksysteme als Unterrichtsgegenstand? (Proceedings). In U. Kortenkamp, H.-G. Weigand, T. Weth & Gesellschaft für Didaktik der Mathematik (Hrsg.), *Informatische*

- Ideen im Mathematikunterricht: Bericht über die 23. Arbeitstagung des Arbeitskreises „Mathematikunterricht und Informatik“ in der Gesellschaft für Didaktik der Mathematik e. V. vom 23. bis 25. September 2005 in Dillingen an der Donau* (S. 17–36). Hildesheim: Franzbecker.
- Magenheim, J., Nelles, W., Rhode, T., Schaper, N., Schubert, S. & Stechert, P. (2010). Competencies for informatics systems and modeling: Results of qualitative content analysis of expert interviews. *Education Engineering (EDUCON)* (S. 513–521). IEEE. doi:10.1109/EDUCON.2010.5492535
- Magenheim, J. & Schulte, C. (2006). Social, ethical and technical issues in informatics – An integrated approach. *Education and Information Technologies*, 11 (3–4), 319–339. doi:10.1007/s10639-006-9012-6
- Magenheim, J., Schulte, C. & Scheel, O. (2002). Informatics and Media Education. Designing a Curriculum for Media Education in Teacher Training with Regard to Basic Areas of Informatics. *Proceeding of ED-MEDIA* (S. 1200–1205). Gehalten auf der World conference on Educational Multimedia, Hypermedia & Telecommunications, Denver.
- Magnusson, S., Krajcik, J. & Borko, H. (1999). Nature, Sources, and Development of Pedagogical Content Knowledge for Science Teaching. In J. Gess-Newsome & N.G. Lederman (Hrsg.), *Examining Pedagogical Content Knowledge* (Band 6, S. 95–132). Dordrecht: Kluwer Academic Publishers. Zugriff am 18.01.2017. Verfügbar unter: [http://link.springer.com/10.1007/0-306-47217-1\\_4](http://link.springer.com/10.1007/0-306-47217-1_4)
- Mahr, B. (2009). Die Informatik und die Logik der Modelle. *Informatik-Spektrum*, 32 (3), 228–249. doi:10.1007/s00287-009-0340-y
- Mayerová, K. (2012). Pilot Activities: LEGO WeDo at Primary School. In M. Moro & D. Alimisis (Hrsg.), *Proceedings of 3rd International Workshop Teaching Robotics* (S. 32–39). Gehalten auf der Teaching with Robotics.
- Mayring, P. (2010). *Qualitative Inhaltsanalyse: Grundlagen und Techniken* (Beltz Pädagogik) (11., aktualisierte und überarb. Aufl.). Weinheim: Beltz.
- McKenney, S. & Voogt, J. (2010). Technology and young children: How 4–7 year olds perceive their own use of computers. *Computers in Human Behavior*, 26 (4), 656–664. doi:10.1016/j.chb.2010.01.002
- McNerney, T. (2004). From turtles to Tangible Programming Bricks: explorations in physical language design. *Personal and Ubiquitous Computing*, 8 (5). doi:10.1007/s00779-004-0295-6
- Medienwissenschaft Universität Bayreuth. (2014). Planspiel Datenschutz (2.0) – Wer weiß was über mich im Internet? Verfügbar unter: <http://medienwissenschaft.uni-bayreuth.de/informatik-im-kontext/index.php/entwuerfe/planspiel-datenschutz-2-0/>
- Meyer, J. H. F. & Land, R. (2005). Threshold concepts and troublesome knowledge (2): Epistemological considerations and a conceptual framework for teaching and learning. *Higher Education*, 49 (3), 373–388. doi:10.1007/s10734-004-6779-5
- Mishra, P. & Koehler, M. J. (2006). Technological pedagogical content knowledge: A framework for teacher knowledge. *Teachers College Record*, 108 (6), 1017–1054.
- MIT (2011). Crickets. Verfügbar unter: <http://llk.media.mit.edu/projects/cricket>
- Mittermeir, R. (2010). Informatikunterricht zur Vermittlung allgemeiner Bildungswerte. *OCG Schriftenreihe* (Band 271, S. 54–73). Gehalten auf der 25 Jahre Schulinformatik: Zukunft mit Herkunft, Wien: Österr. Computer-Ges.

- Modrow, E. (2010). Informatik als technisches Fach. *springerprofessional.de, LOG IN* (163/164), 38–42.
- Morreale, P. & Joiner, D. (2011). Reaching future computer scientists. *Communications of the ACM*, 54 (4), 121–124. doi:10.1145/1924421.1924448
- MPFS – Medienpädagogischer Forschungsverbund Südwest. (2014a). *MPFS – Medienpädagogischer Forschungsverbund Südwest : KIM-Studie 2014*. Zugriff am 30.06.2015. Verfügbar unter: <http://www.mpfs.de/index.php?id=646>
- MPFS – Medienpädagogischer Forschungsverbund Südwest. (2014b). *miniKIM 2014. Kleinkinder und Medien*. Zugriff am 14.10.2015. Verfügbar unter: [http://www.mpfs.de/fileadmin/miniKIM/2014/miniKIM\\_2014.pdf](http://www.mpfs.de/fileadmin/miniKIM/2014/miniKIM_2014.pdf)
- MPFS – Medienpädagogischer Forschungsverbund Südwest. (2015). *JIM-Studie 2015 – Jugend, Information, (Multi-)Media*. Stuttgart.
- Müller, K. (2015). What do we expect from graduates in CS? First results of a survey at university and company as part of a methodology for developing a competence model. “A New Culture of Learning: Computing and next Generations” (S. 192–201). Gehalten auf der IFIP TC3 Working Conference, Vilnius, Lithuania.
- Myers, B. A. (1990). Taxonomies of visual programming and program visualization. *Journal of Visual Languages & Computing*, 1 (1), 97–123. doi:10.1016/S1045-926X(05)80036-9
- National Research Council (U.S.) & Committee for the Workshops on Computational Thinking. (2010). *Report of a workshop on the scope and nature of computational thinking*. Washington, D.C.: National Academies Press. Zugriff am 16.01.2017. Verfügbar unter: <http://public.eblib.com/choice/publicfullrecord.aspx?p=3378614>
- National Research Council (U.S.) & Committee for the Workshops on Computational Thinking. (2011). *Report of a workshop of pedagogical aspects of computational thinking*. Washington, D.C.: National Academies Press.
- Naur, P. (1985). Programming as theory building. *Microprocessing and microprogramming*, 15 (5), 253–261.
- NCTM – National Council of Teachers of Mathematics. (1989). Curriculum and evaluation standards for school mathematics.
- NCTM – National Council of Teachers of Mathematics. (1991). Professional standards for teaching mathematics.
- NCTM – National Council of Teachers of Mathematics. (1995). Assessment standards for school mathematics.
- NCTM – National Council of Teachers of Mathematics. (2000). Principles and Standards for School Mathematics.
- Neugebauer, J., Magenheimer, J., Ohrndorf, L., Schaper, N. & Schubert, S. (2015). Defining Proficiency Levels of High School Students in Computer Science by an Empirical Task Analysis Results of the MoKoM Project. In A. Brodnik & J. Vahrenhold (Hrsg.), *Informatics in Schools. Curricula, Competences, and Competitions* (Band 9378, S. 45–56). Cham: Springer International Publishing. Zugriff am 18.01.2017. Verfügbar unter: [http://link.springer.com/10.1007/978-3-319-25396-1\\_5](http://link.springer.com/10.1007/978-3-319-25396-1_5)
- Ni, L. (2011). Building professional identity as computer science teachers: supporting secondary computer science teachers through reflection and community building (S. 143). ACM Press. doi:10.1145/2016911.2016942
- Ni, L. & Guzdial, M. (2015). Prepare and Support Computer Science (CS) Teachers: Understanding CS Teachers’ Professional Identity.

- OECD. (2005). The Definition and Selection of Key Competencies. Executive Summary.
- Palaiologou, I. (2016). Children under five and digital technologies: implications for early years pedagogy. *European Early Childhood Education Research Journal*, 24 (1), 5–24. doi:10.1080/1350293X.2014.929876
- Papert, S. (1980). *Mindstorms: children, computers, and powerful ideas*. New York: Basic Books.
- Papert, S. (1982). *Mindstorms: Kinder, Computer und Neues Lernen*. Basel: Birkhäuser Verlag.
- Papert, S. (1987). Microworlds: transforming education. In Lawler, R. W., & Yazdani, M. (Hrsg.), *Artificial intelligence and education: Learning environments and tutoring systems* (Band 1, S. 79–94). Norwood, NJ: Ablex Publishing.
- Papert, S. (1998). *Die vernetzte Familie: Kinder und Computer*. Stuttgart: Kreuz.
- Papert, S. & Harel, I. (1991). Situating Constructionism. *Constructionism*. Norwood, NJ: Ablex Publishing.
- Pauen, S. & Pahnke, J. (2008). Mathematische Kompetenzen im Kindergarten. Evaluation der Effekte einer Kurzzeitintervention. *Empirische Pädagogik*, 22, 193–208.
- Perlman, R. (1976). *Using computer technology to provide a creative learning environment for preschool children*. MIT Cambridge: Logo Memo.
- Petersen, U., Theidig, G., Bördig, J., Leimbach, T. & Flintrop, B. (2007). Abschlussbericht Roberta.
- Plowman, L., Stevenson, O., Stephen, C. & McPake, J. (2012). Preschool children's learning with technology at home. *Computers & Education*, 59 (1), 30–37. doi:10.1016/j.compedu.2011.11.014
- Pohl, W., Kranzdorf, K. & Hein, H.-W. (2007). Einstieg Informatik – Aktivitäten und Erfahrungen. In INFOS (S. 253–264).
- Portelance, D. J., Strawhacker, A. L. & Bers, M. U. (2016). Constructing the ScratchJr programming language in the early childhood classroom. *International Journal of Technology and Design Education*, 26 (4), 489–504. doi:10.1007/s10798-015-9325-0
- Rechenberg, P. (2010). Was ist Informatik. *25 Jahre Schulinformatik* (S. 46–53). Österr. Computer-Ges.
- Reichert, R., Nievergelt, J. & Hartmann, W. (2005). *Programmieren mit Kara: ein spielerischer Zugang zur Informatik*. Berlin [u. a.]: Springer. Zugriff am 17.01.2017. Verfügbar unter: <http://dx.doi.org/10.1007/b138643>
- Resnick, M. (2007). Sowing the Seeds for a More Creative Society. *Proc. Learning & Leading with Technology*.
- Resnick, M. (2013). Lifelong Kindergarten. Cultures of Creativity. LEGO Education.
- Resnick, M. & Rusk, N. (1996). The Computer Clubhouse: Preparing for life in a digital world. *IBM Systems Journal*, 35 (3.4), 431–439. doi:10.1147/sj.353.0431
- Resnick, M. & Silverman, B. (2005). Some reflections on designing construction kits for kids (S. 117–122). ACM Press. doi:10.1145/1109540.1109556
- Resnick, M., Silverman, B., Kafai, Y., Maloney, J., Monroy-Hernández, A., Rusk, N. et al. (2009). Scratch: programming for all. *Communications of the ACM*, 52 (11), 60. doi:10.1145/1592761.1592779
- Richter, T. & Naumann, J. (2010). Inventar zur Computerbildung (INCOBI-R) – Eine revidierte Fassung des Inventars zur Computerbildung (INCOBI-R).
- Richter, T., Naumann, J. & Groeben, N. (2001). Das Inventar zur Computerbildung (INCOB): Ein Instrument zur Erfassung von Computer Literacy und computerbezogenen Einstel-

- lungen bei Studierenden der Geistes- und Sozialwissenschaften. *Psychologie in Erziehung und Unterricht*, 48 (1), 1–13.
- Riese, J. (2009). *Professionelles Wissen und professionelle Handlungskompetenz von (angehenden) Physiklehrkräften* (Studien zum Physik- und Chemielernen). Berlin: Logos-Verl.
- Rogozhkina, I. & Kushnirenko, A. (2011). PikoMir: teaching programming concepts to pre-schoolers with a new tutorial environment. *Procedia – Social and Behavioral Sciences*, 28, 601–605. doi:10.1016/j.sbspro.2011.11.114
- Romeike, R. (2008). *Kreativität im Informatikunterricht*. Potsdam: Universität Potsdam.
- Romeike, R. & Reichert, D. (2011). PicoCrickets als Zugang zur Informatik in der Grundschule. Gehalten auf der INFOS.
- Ropohl, G. (1999). *Allgemeine Technologie: eine Systemtheorie der Technik* (2. Aufl.). München, Wien: Hanser.
- Rosen, L. D., Lim, A. F., Felt, J., Carrier, L. M., Cheever, N. A., Lara-Ruiz, J. M. et al. (2014). Media and technology use predicts ill-being among children, preteens and teenagers independent of the negative health impacts of exercise and eating habits. *Computers in Human Behavior*, 35, 364–375. doi:10.1016/j.chb.2014.01.036
- Rozenszajn, R. & Yarden, A. (2014). Expansion of Biology Teachers' Pedagogical Content Knowledge (PCK) During a Long-Term Professional Development Program. *Research in Science Education*, 44 (1), 189–213. doi:10.1007/s11165-013-9378-6
- Ruf, A., Mühlhling, A. & Hubwieser, P. (2014). Scratch vs. Karel: impact on learning outcomes and motivation (S. 50–59). ACM Press. doi:10.1145/2670757.2670772
- Rushkoff, D. (2010). *Program or be programmed: ten commands for a digital age*. New York: Or Books. Zugriff am 17.01.2017. Verfügbar unter: <https://www.overdrive.com/search?q=45508BA6-FoA8-4158-8C56-784BA8856F7D>
- Sachser, N. (2004). Neugier, Spiel und Lernen: Verhaltensbiologische Anmerkungen zur Kindheit. *Zeitschrift für Pädagogik* 50 (4), 475–486.
- Saeli, M. (2012). *Teaching Programming for Secondary School: A Pedagogical Content Knowledge Based Approach*. Eindhoven: Technische Universiteit Eindhoven.
- Scheffe, P. (1999). Softwaretechnik und Erkenntnistheorie. *Informatik-Spektrum*, 22 (2), 122–135. doi:10.1007/s002870050131
- Schiefele, U. (2009). Motivation. In E. Wild & J. Möller (Hrsg.), *Pädagogische Psychologie* (S. 151–177). Berlin, Heidelberg: Springer Berlin Heidelberg. Zugriff am 18.01.2017. Verfügbar unter: [http://link.springer.com/10.1007/978-3-540-88573-3\\_7](http://link.springer.com/10.1007/978-3-540-88573-3_7)
- Schneider, W., Körkel, J. & Weinert, F. E. (1989). Domain-specific knowledge and memory performance: A comparison of high- and low-aptitude children. *Journal of Educational Psychology*, 81 (3), 306–312. doi:10.1037/0022-0663.81.3.306
- Schubert, S. & Schwill, A. (2011). *Didaktik der Informatik* (2. Auflage.). Heidelberg: Spektrum, Akademischer Verlag.
- Schulte, C. (2001). Vom Modellieren zum Gestalten: Objektorientierung als Impuls für einen neuen Informatikunterricht? *informatica didacta, Zeitschrift für fachdidaktische Grundlagen der Informatik*, 3.
- Schulte, C. (2003). *Lehr-Lernprozesse im Informatik-Anfangsunterricht theoriegeleitete Entwicklung und Evaluation eines Unterrichtskonzepts zur Objektorientierung in der Sekundarstufe II*. S.l.: s.n. Zugriff am 29.08.2016. Verfügbar unter: <http://ubdata.uni-paderborn.de/ediss/17/2003/schulte/disserta.pdf>



- Schulte, C. (2008a). Interesse wecken und Grundkenntnisse vermitteln. 3. *Münsteraner Workshop zur Schulinformatik*. Münster.
- Schulte, C. (2008b). Duality Reconstruction – Teaching Digital Artifacts from a Socio-technical Perspective. In R. T. Mittermeir & M. M. Sysło (Hrsg.), *Informatics Education – Supporting Computational Thinking* (Band 5090, S. 110–121). Berlin, Heidelberg: Springer. Zugriff am 17.01.2017. Verfügbar unter: [http://link.springer.com/10.1007/978-3-540-69924-8\\_10](http://link.springer.com/10.1007/978-3-540-69924-8_10)
- Schulte, C. (2009). Dualitätsrekonstruktion als Hilfsmittel zur Entwicklung und Planung von Informatikunterricht. *Zukunft braucht Herkunft: 25 Jahre „INFOS – Informatik und Schule“* (S. 355–366). Gehalten auf der INFOS, Berlin.
- Schulte, C. (2012). Uncovering structure behind function: the experiment as teaching method in computer science education (S. 40–47). New York: ACM Press. doi:10.1145/2481449.2481460
- Schulte, C. (2013). Reflections on the role of programming in primary and secondary computing education (S. 17–24). New York: ACM Press. doi:10.1145/2532748.2532754
- Schulte, C. & Bennedsen, J. (2006). What do teachers teach in introductory programming? (S. 17–28). Gehalten auf der ICER'06, New York: ACM Press. doi:10.1145/1151588.1151593
- Schulte, C. & Knobelsdorf, M. (2011). Medien nutzen, Medien gestalten – eine qualitative Analyse der Computernutzung. In C. Albers, J. Magenheim & D. M. Meister (Hrsg.), *Schule in der digitalen Welt* (S. 97–115). Wiesbaden: VS Verlag für Sozialwissenschaften. Zugriff am 19.01.2017. Verfügbar unter: [http://link.springer.com/10.1007/978-3-531-92850-0\\_5](http://link.springer.com/10.1007/978-3-531-92850-0_5)
- Schulz, W. (1997). Die lehrtheoretische Didaktik. In H. Gudjons (Hrsg.), *Didaktische Theorien* (S. 35–56). Hamburg: Bergmann + Helbig.
- Schwarzkopf, H. & Zolg, M. (1997). Kann der Computer denken? Gespräche mit Kindern. *Die Grundschulzeitschrift*, 11 (108), 44–46.
- Schwill, A. (1993). Fundamentale Ideen der Informatik. *Zentralblatt für Didaktik der Mathematik*, 25 (1), 20–31.
- Schwill, A. (1995). Fundamentale Ideen in Mathematik und Informatik. *Bericht über die 12. Arbeitstagung des AK „Mathematikunterricht und Informatik“ der GDM* (S. 18–25).
- Schwill, A. (2001). Ab wann kann man mit Kindern Informatik machen. *INFOS 2001-9. GI-Fachtagung Informatik und SchuleGI-Edition*, 13–30.
- Seel, N. M. (2003). *Psychologie des Lernens: Lehrbuch für Pädagogen und Psychologen ; mit 12 Tabellen und zahlreichen Übungsaufgaben* (UTB Pädagogik, Psychologie) (2., aktualisierte und erw. Aufl.). München: Reinhardt.
- Senatsverwaltung für Bildung, Jugend und Wissenschaft Berlin (Hrsg.) (2014). *Berliner Bildungsprogramm für Kitas und Kindertagespflege* (aktualisierte Neuaufl.). Weimar: Verlag das Netz.
- Shinners-Kennedy, D. & Fincher, S. A. (2013). Identifying threshold concepts: from dead end to a new direction (S. 9). New York: ACM Press. doi:10.1145/2493394.2493396
- Shulman, L. S. (1986). Those Who Understand: Knowledge Growth. *Teaching Educational*, 15 (2), 4–14.
- Shulman, L. S. (1987). Knowledge and teaching: Foundations of the new reform. *Harvard Educational Review*, 57 (1), 1–21.
- Siemens, G. (2008). Connectivism: A Learning Theory of the Digital Age 2008. *International Journal of Instructional Technology & Distance Learning*.

- Sodian, B. (2002). Die Entwicklung des bereichsspezifischen Wissens. In R. Oerter & L. Montada (Hrsg.), *Entwicklungspsychologie*. Weinheim: Psychologie Verlags Union.
- Sodian, B., Koerber, S. & Thoermer, C. (2006). Zur Entwicklung des naturwissenschaftlichen Denkens im Vor- und Grundschulalter. In P. Nentwig & S. Schanze (Hrsg.), *Es ist nie zu früh. Naturwissenschaftliche Bildung in jungen Jahren* (S. 11–20). Münster: Waxmann.
- Spiro, R. J., Feltovich, P. J., Jacobson, M. J. & Coulson, R. L. (1991). Cognitive flexibility, constructivism and hypertext: Random access instruction for advanced knowledge acquisition in ill-structured domains. *Educational Technology*, 24–33.
- Stechert, P. (2009). *Fachdidaktische Diskussion von Informatiksystemen und der Kompetenzentwicklung im Informatikunterricht* (Commentarii informaticae didacticae (CID)) (Band 2).
- Steinbuch, K. (1957). Automatische Informationsverarbeitung. SEG-Nachrichten (Technische Mitteilungen der Standard Elektrik Gruppe) – Firmenzeitschrift, 4, 171.
- Stern, E. (2002). Wie abstrakt lernt das Grundschulkind? In H. Petillon (Hrsg.), *Individuelles und soziales Lernen in der Grundschule* (S. 27–42). Wiesbaden: VS Verlag für Sozialwissenschaften. doi:10.1007/978-3-322-99278-9\_2
- Stiftung Haus der kleinen Forscher (2013a). *Kommst du mit die Zeit entdecken? Ideen zum Forschen und Staunen rund um das Phänomen „Zeit“*. Berlin: Stiftung Haus der kleinen Forscher.
- Stiftung Haus der kleinen Forscher (Hrsg.) (2013b). *Wissenschaftliche Untersuchungen zur Arbeit der Stiftung „Haus der kleinen Forscher“* (Band 5). Schaffhausen: Schubi Lernmedien AG. Verfügbar unter: [www.haus-der-kleinen-forscher.de](http://www.haus-der-kleinen-forscher.de)
- Stiftung Haus der kleinen Forscher (2014). *Kannst du mich verstehen? Die Vielfalt der Kommunikation erkunden und erforschen*. Berlin: Stiftung Haus der kleinen Forscher.
- Stiftung Haus der kleinen Forscher (2015). *Pädagogischer Ansatz der Stiftung „Haus der kleinen Forscher“. Anregungen für die Lernbegleitung in Naturwissenschaften, Mathematik und Technik* (5. Auflage). Berlin: Stiftung Haus der kleinen Forscher.
- Taub, R., Ben-Ari, M. & Armoni, M. (2009). The effect of CS unplugged on middle-school students' views of CS. *ACM SIGCSE Bulletin*, 41(3), 99–103. doi:10.1145/1562877.1562912
- Tedre, M. & Apiola, M. (2013). Three computing Traditions in school computing education. The Royal Society. (2012). Shut down or restart? The way forward for computing in UK schools. The Royal Society.
- The White House (2016). Fact Sheet: President Obama Announces Computer Science For All Initiative. Zugriff am 05.03.2018. Verfügbar unter: <https://www.whitehouse.gov/the-press-office/2016/01/30/fact-sheet-president-obama-announces-computer-science-all-initiative-0>
- Tillmann, A., Fleischer, S. & Hugger, K.-U. (Hrsg.) (2014). *Handbuch Kinder und Medien* (Digitale Kultur und Kommunikation). Wiesbaden: Springer VS.
- UNESCO (2012). UNESCO ICT Competency Framework for Teachers. Zugriff am 17.04.2018. Verfügbar unter: <http://unesdoc.unesco.org/images/0021/002134/213475E.pdf>
- van Merriënboer, J. J. G. & Krammer, H. P. M. (1987). Instructional strategies and tactics for the design of introductory computer programming courses in high school. *Instructional Science*, 16 (3), 251–285. doi:10.1007/BF00120253
- Vekiri, I. & Chronaki, A. (2008). Gender issues in technology use: Perceived social support, computer self-efficacy and value beliefs, and computer use beyond school. *Computers & Education*, 51 (3), 1392–1404. doi:10.1016/j.compedu.2008.01.003

- Walter-Herrmann, J. & Büching, C. (Hrsg.) (2013). *FabLab of machines, makers and inventors* (Cultural and media studies). Bielefeld: transcript.
- Wegner, P. (1997). Why interaction is more powerful than algorithms. *Communications of the ACM*, 40 (5), 80–91. doi:10.1145/253769.253801
- Weinert, F. E. (2001). Concept of Competence: A conceptual clarification. In D. S. Rychen & L. H. Salganik (Hrsg.), *Defining and Selecting Key Competencies*. Seattle: Hogrefe and Huber.
- Weintrop, D. & Wilensky, U. (2015). Using Commutative Assessments to Compare Conceptual Understanding in Blocks-based and Text-based Programs (S. 101–110). New York: ACM Press. doi:10.1145/2787622.2787721
- Wenger, E. (1998). Communities of Practice. Learning as a social system.
- Wiener, N. (1948). *Kybernetik – Regelung und Nachrichtenübertragung in Lebewesen und in der Maschine*. Düsseldorf: Econ-Verlag Düsseldorf.
- Wiesner, A. (2008). *Lerngruppe – Logisches Denken*. München: Ars-Ed.
- Wiesner, B. & Brinda, T. (2007). Erfahrungen bei der Vermittlung algorithmischer Grundstrukturen im Informatikunterricht der Realschule mit einem Robotersystem. *Didaktik der Informatik in Theorie und Praxis*, 113–124.
- Wikipedia (2007). Endlicher Automat. *Wikipedia*. Zugriff am 15.05.2018. Verfügbar unter: [https://de.wikipedia.org/w/index.php?title=Endlicher\\_Automat&stable=o&redirect=no#/media/File:Finite\\_state\\_machine\\_example\\_with\\_comments.svg](https://de.wikipedia.org/w/index.php?title=Endlicher_Automat&stable=o&redirect=no#/media/File:Finite_state_machine_example_with_comments.svg)
- Wikipedia (2011). Petri-Netz. *Wikipedia*. Zugriff am 15.05.2018. Verfügbar unter: [https://commons.wikimedia.org/wiki/File:Seasons\\_1.svg](https://commons.wikimedia.org/wiki/File:Seasons_1.svg)
- Wikipedia (2014). Artefakt. *Wikipedia*. Zugriff am 4.5.2016. Verfügbar unter: <https://de.wikipedia.org/w/index.php?title=Artefakt&oldid=136435468>
- Wikipedia (2015). UML Sequenzdiagramm. *Wikipedia*. Zugriff am 15.05.2018. Verfügbar unter: <https://commons.wikimedia.org/wiki/File:UmlSequenzdiagramm-2.svg>
- Wikipedia (2016). Softwarequalität. *Wikipedia*. Zugriff am 15.05.2018. Verfügbar unter: <https://de.wikipedia.org/wiki/Softwarequalit%C3%A4t>
- Williams, J. & Lockley, J. (2012). Using CoRes to Develop the Pedagogical Content Knowledge (PCK) of Early Career Science and Technology Teachers. *Journal of Technology Education*, 24 (1). doi:10.21061/jte.v24i1.a.3
- Wilson, C. (2015). Hour of code – a record year for computer science. *ACM Inroads*, 6 (1), 22–22. doi:10.1145/2723168
- Wilson, C., Sudal, L. A., Stephenson, C. & Stehlik, M. (2010). *Running on Empty: The Failure to Teach Computer Science in the Digital Age*. New York: Association for Computing Machinery and the Computer Science Teachers Association.
- Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, 49 (3), 33–35. doi:10.1145/1118178.1118215
- Wing, J. M. (2008). Computational thinking and thinking about computing. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 366 (1881), 3717–3725. doi:10.1098/rsta.2008.0118
- WYSIWYG (2016). *Wikipedia*. Zugriff am 30.08.2016. Verfügbar unter: <https://de.wikipedia.org/w/index.php?title=WYSIWYG&oldid=157393007>
- Xie, L., Antle, A. N. & Motamedi, N. (2008). Are tangibles more fun?: comparing children's enjoyment and engagement using physical, graphical and tangible user interfaces (S. 191). New York: ACM Press. doi:10.1145/1347390.1347433

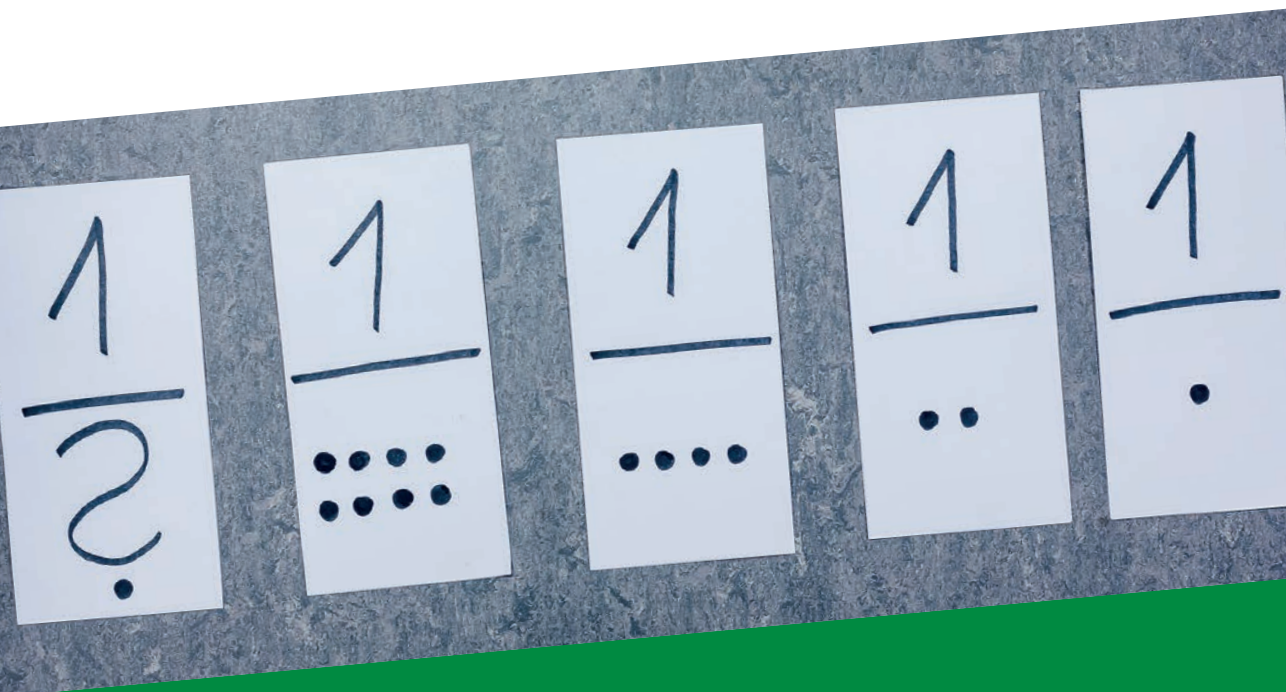
- Yardi, S. & Bruckman, A. (2007). What is computing?: bridging the gap between teenagers' perceptions and graduate students' experiences (S. 39). New York: ACM Press. doi:10.1145/1288580.1288586
- Ziegenbalg, J. (1985). *Programmieren lernen mit Logo*. München: Hanser.
- Zolg, M. (2006). Mut zur Technik! *Weltwissenschaft Sachunterricht*, (4), 4–5.

## Fazit und Ausblick –

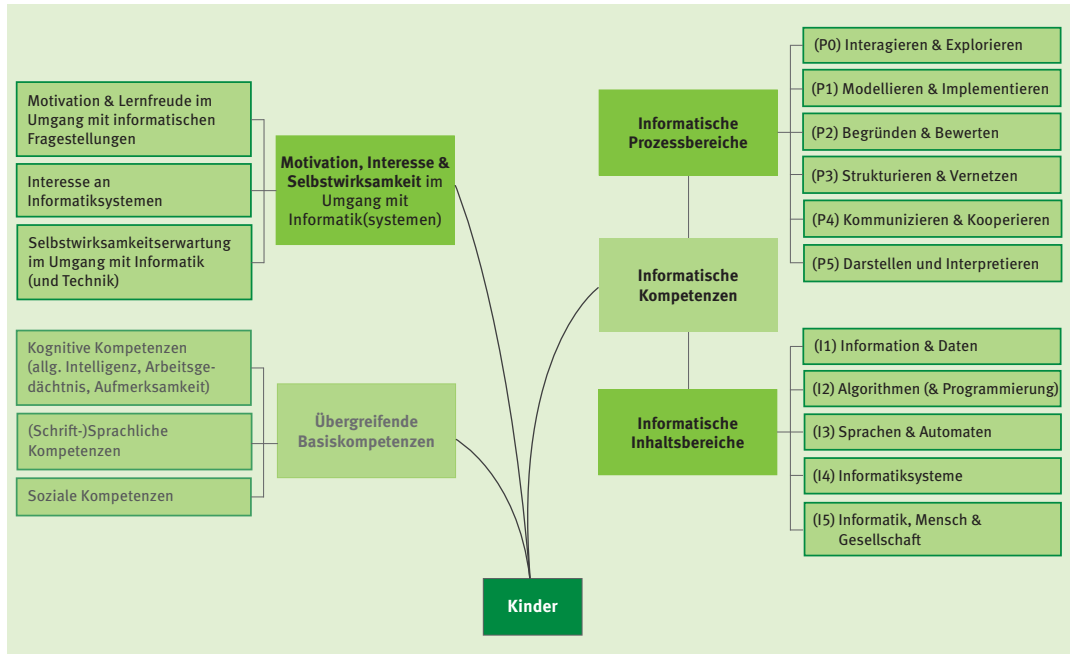
### Stiftung Haus der kleinen Forscher

- Chaudron, S. (2015). *Young Children (0–8) and digital technology: a qualitative exploratory study across seven countries*. European Commission, Joint Research Centre & Institute for the Protection and the Security of the Citizen (Hrsg.). Luxembourg: Publications Office. Zugriff am 17.01.2017. Verfügbar unter: <http://dx.publications.europa.eu/10.2788/00749>
- Stiftung Haus der kleinen Forscher (Hrsg.) (2013). *Wissenschaftliche Untersuchungen zur Arbeit der Stiftung „Haus der kleinen Forscher“* (Band 5). Schaffhausen: Schubi Lernmedien AG. Verfügbar unter: [www.haus-der-kleinen-forscher.de](http://www.haus-der-kleinen-forscher.de)
- Stiftung Haus der kleinen Forscher (Hrsg.) (2015a). *Wissenschaftliche Untersuchungen zur Arbeit der Stiftung „Haus der kleinen Forscher“* (Band 7). Schaffhausen: Schubi Lernmedien AG. Verfügbar unter: [www.haus-der-kleinen-forscher.de](http://www.haus-der-kleinen-forscher.de)
- Stiftung Haus der kleinen Forscher (2015b). *Pädagogischer Ansatz der Stiftung „Haus der kleinen Forscher“. Anregungen für die Lernbegleitung in Naturwissenschaften, Mathematik und Technik* (5. Auflage). Berlin: Stiftung Haus der kleinen Forscher. Verfügbar unter: [www.haus-der-kleinen-forscher.de](http://www.haus-der-kleinen-forscher.de)
- Stiftung Haus der kleinen Forscher (2016). *Zahlen, Zählen, Rechnen – Mathematik entdecken. Mathematisches Denken von Kita- und Grundschulkindern unterstützen*. Berlin: Stiftung Haus der kleinen Forscher. Verfügbar unter: [www.haus-der-kleinen-forscher.de](http://www.haus-der-kleinen-forscher.de)
- Stiftung Haus der kleinen Forscher (Hrsg.) (2017a). *Frühe mathematische Bildung – Ziele und Gelingensbedingungen für den Elementar- und Primarbereich. Wissenschaftliche Untersuchungen zur Arbeit der Stiftung „Haus der kleinen Forscher“* (Band 8). Opladen, Berlin, Toronto: Verlag Barbara Budrich. Verfügbar unter: [www.haus-der-kleinen-forscher.de](http://www.haus-der-kleinen-forscher.de)
- Stiftung Haus der kleinen Forscher (2017b). *Materialpaket Informatik entdecken – mit und ohne Computer*. Berlin: Stiftung Haus der kleinen Forscher. Verfügbar unter [www.haus-der-kleinen-forscher.de](http://www.haus-der-kleinen-forscher.de)
- Stiftung Haus der kleinen Forscher (2017c). *Umfrage unter Erzieherinnen und Erziehern zur Mediennutzung. Wie nutzen Erzieherinnen und Erzieher digitale Geräte in Kitas?* Berlin: Stiftung Haus der kleinen Forscher. Verfügbar unter [www.haus-der-kleinen-forscher.de](http://www.haus-der-kleinen-forscher.de)
- Stiftung Haus der kleinen Forscher (2017d). *Technikkreis*. Berlin: Stiftung Haus der kleinen Forscher.

# Anhang

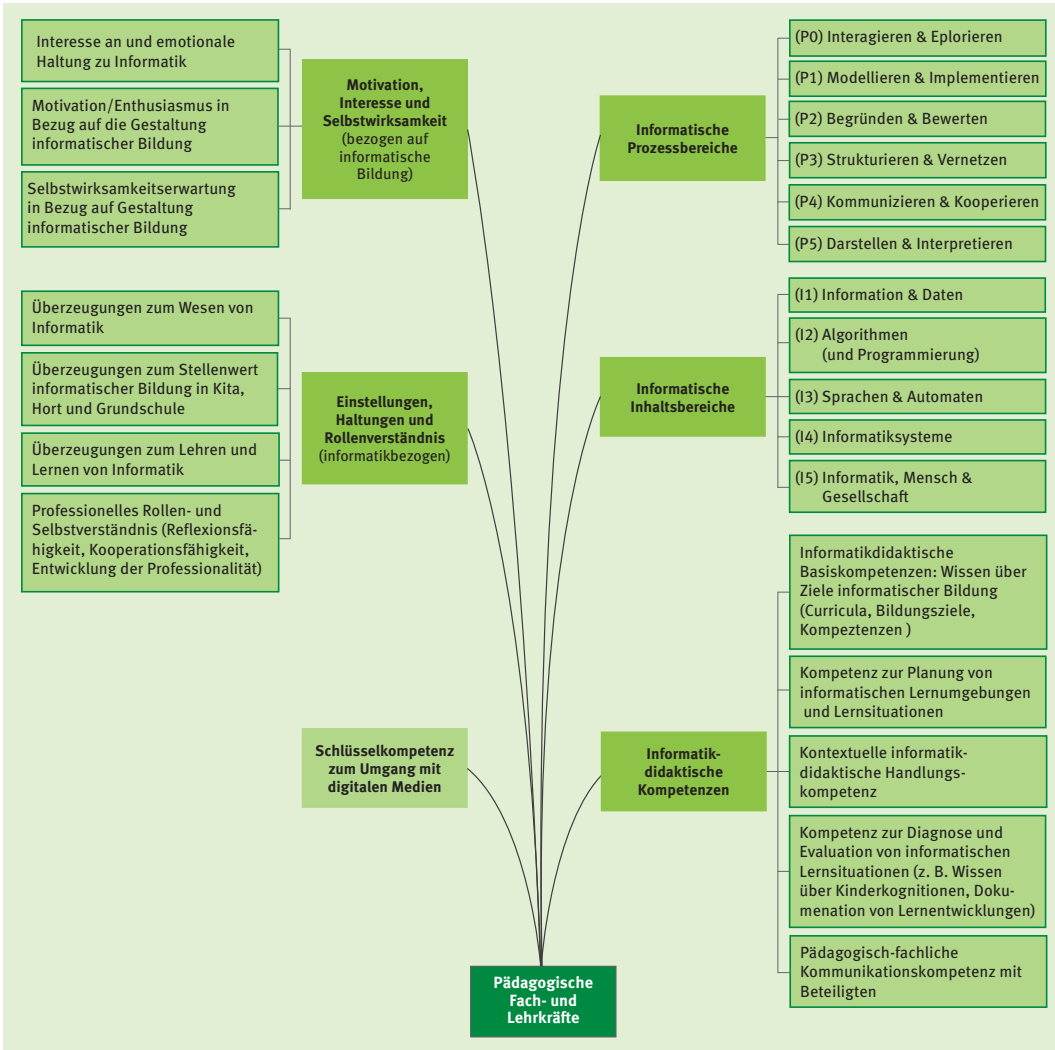


## Anhang I: Zieldimensionen informatischer Bildung auf Ebene der Kita- und Grundschul Kinder



**Quelle:** Stiftung Haus der kleinen Forscher, nach Bergner, N. et al. (2018). Zieldimensionen informatischer Bildung im Elementar- und Primarbereich. In: Haus der kleinen Forscher (Hrsg.), *Frühe informatische Bildung – Ziele und Gelingensbedingungen für den Elementar- und Primarbereich*. Wissenschaftliche Untersuchungen zur Arbeit der Stiftung „Haus der kleinen Forscher“ (Band 9).

## Anhang II: Zieldimensionen informatischer Bildung auf Ebene der pädagogischen Fach- und Lehrkräfte



**Quelle:** Stiftung Haus der kleinen Forscher, nach Bergner, N. et al. (2018). Zieldimensionen informatischer Bildung im Elementar- und Primarbereich. In: *Haus der kleinen Forscher (Hrsg.), Frühe informatische Bildung – Ziele und Gelingensbedingungen für den Elementar- und Primarbereich*. Wissenschaftliche Untersuchungen zur Arbeit der Stiftung „Haus der kleinen Forscher“ (Band 9).

## Bildquellenverzeichnis

Seite 18, 29, 34, 38, 59, 85, 101, 109, 146, 153, 181, 189, 203, 242, 250, 302, 306,  
318, 320, 324, 342: © Christoph Wehrer/Stiftung Haus der kleinen Forscher

Seite 51, 75, 134, 268, 290: © Stiftung Haus der kleinen Forscher

Seite 272, 286, 288, 292 295: © InfoSphere-Team

Seite 275, 277, 280, 284: © Kathrin Müller



## Über die Stiftung „Haus der kleinen Forscher“

Die gemeinnützige Stiftung „Haus der kleinen Forscher“ engagiert sich für gute frühe Bildung in den Bereichen Mathematik, Informatik, Naturwissenschaften und Technik (MINT) – mit dem Ziel, Mädchen und Jungen stark für die Zukunft zu machen und zu nachhaltigem Handeln zu befähigen. Gemeinsam mit ihren Netzwerkpartnern vor Ort bietet die Stiftung bundesweit ein Bildungsprogramm an, das pädagogische Fach- und Lehrkräfte dabei unterstützt, Kinder im Kita- und Grundschulalter qualifiziert beim Entdecken, Forschen und Lernen zu begleiten. Das „Haus der kleinen Forscher“ verbessert Bildungschancen, fördert Interesse am MINT-Bereich und professionalisiert dafür pädagogisches Personal. Partner der Stiftung sind die Helmholtz-Gemeinschaft, die Siemens Stiftung, die Dietmar Hopp Stiftung und die Deutsche Telekom Stiftung. Gefördert wird sie vom Bundesministerium für Bildung und Forschung.

### Vision und Mission der Stiftung „Haus der kleinen Forscher“

#### **Vision der Stiftung „Haus der kleinen Forscher“:**

#### **Fragen – Forschen – Zukunft gestalten**

Alle Kinder in Deutschland erleben Bildungsorte, in denen sie ihren eigenen Fragen nachgehen und forschend die Welt entdecken können. Solche „Häuser der kleinen Forscher“ machen Mädchen und Jungen stark für die Zukunft. Sie befähigen Kinder, selbstbestimmt zu denken und verantwortungsvoll zu handeln.

Technologisierung und Digitalisierung sowie Folgen des Klimawandels und der sozialen Ungleichheit beeinflussen zunehmend unseren Alltag. Wir tragen dazu bei, dass sich Menschen in unserer schnell verändernden Welt orientieren können und offen für Neues bleiben.

Die alltägliche Auseinandersetzung mit Natur und Technik fördert Neugier, Lern- und Denkfreude der Mädchen und Jungen. Wir sehen frühe Bildung als Schlüssel, um den Herausforderungen einer komplexen Welt erfolgreich begegnen zu können.

**Mission der Stiftung „Haus der kleinen Forscher“:**

Die Stiftung „Haus der kleinen Forscher“ ...

- befördert eine fragend-forschende Haltung bei Kindern,
- gibt Mädchen und Jungen schon in jungen Jahren die Chance, eigene Talente und Potenziale in den Bereichen Naturwissenschaften, Technik, Mathematik und Informatik zu entdecken
- und legt den Grundstein für einen reflektierten Umgang mit technologischen und gesellschaftlichen Veränderungen im Sinne einer nachhaltigen Entwicklung.

Gemeinsam mit ihren Bezugspersonen erleben die Kinder Spaß und Freude am Entdecken und Verstehen dieser Welt. Kinder gestalten Bildungsprozesse aktiv mit und erleben sich dadurch als kompetent und selbstwirksam in ihrem Alltag. Beim forschenden Lernen können Kinder Problemlösekompetenzen entwickeln, eigene Antworten finden und Selbstvertrauen spüren („Ich kann!“) – Erfahrungen und Fähigkeiten, die weit über die Kindheit hinaus für die Persönlichkeitsentwicklung und die spätere Berufsbiographie von Bedeutung sind.

In einem praxisnahen und qualitativ hochwertigen Professionalisierungsansatz unterstützt die Stiftung pädagogische Fach- und Lehrkräfte dabei, Kinder im Alter bis 10 Jahren beim Entdecken, Forschen und Lernen zu begleiten. Über vielfältige Fortbildungsangebote erleben Fach- und Lehrkräfte die Faszination eigenen Forschens für sich selbst. Sie erweitern ihre Kenntnisse und pädagogischen Kompetenzen und setzen sie in ihrer alltäglichen Arbeit mit Kindern um.

Die Initiative unterstützt Bildungseinrichtungen darin, sich als „Ort des forschenden Lernens“ nachhaltig weiterzuentwickeln und in diesem Sinn als „Haus der kleinen Forscher“ förderliche Lernumgebungen für Kinder zu schaffen.

## Bisher erschienen in der Wissenschaftlichen Schriftenreihe der Stiftung „Haus der kleinen Forscher“



### Band 1 (2011)

*Dagmar Berwanger, Petra Evanschitzky, Elke Heller, Christa Preissing, Ursula Rabe-Kleberg, Franziska Schulze, Anna Spindler*

Der erste Band der Schriftenreihe „Wissenschaftliche Untersuchungen zur Arbeit der Stiftung ‚Haus der kleinen Forscher‘“ stellt vier wissenschaftliche Expertisen aus den Jahren 2009 und 2010 vor, die von renommierten Wissenschaftlerinnen und Wissenschaftlern aus dem Bereich der frühkindlichen Pädagogik verfasst wurden. Die vorliegenden wissenschaftlichen Beiträge reflektieren den pädagogischen Ansatz und das Multiplikatorenmodell der Stiftung vor dem Hintergrund der eigenen Zielvorstellungen, analysieren die Umsetzungspraxis in den Netzwerken und Kitas und zeigen weitere Entwicklungsmöglichkeiten auf.



### Band 2 (2011)

*Franziska Kramer, Ursula Rabe-Kleberg*

Die Studie von Ursula Rabe-Kleberg und Franziska Kramer bildet eine logische Konsequenz und Ergänzung zur ersten Untersuchung der Autorinnen „Erzieherinnen und ihre Haltung zu Naturwissenschaft und Technik für Jungen und Mädchen“, die in Band 1 dieser Schriftenreihe veröffentlicht wurde (siehe oben). Die Autorinnen untersuchen die Gestaltung der Lernprozesse durch Erzieherinnen im Detail. Mit hoher Präzision und Sensibilität im Umgang mit den Möglichkeiten qualitativer Sozialforschung gelingt es den Autorinnen dabei, ko-konstruktive Augenblicke des gemeinsamen Lernens in Kitas einzufangen und intensiv im Hinblick auf wichtige Einflussvariablen zu reflektieren.

**Band 3** (2012)

*Michael Fritz, Gabriele Grieshop, Katrin Hille, Maren Lau, Martin Winter*

Im dritten Band werden zwei Studien vorgestellt, die sich mit der Rolle der Bedeutung der Trainerinnen und Trainer in der Initiative „Haus der kleinen Forscher“ aus jeweils unterschiedlichen Perspektiven beschäftigen: In der Studie von Maren Lau, Michael Fritz und Katrin Hille (ZNL) stehen das Rollen- und Selbstverständnis der Trainerinnen und Trainer sowie ihr subjektives Kompetenempfinden im Mittelpunkt. In der Untersuchung von Gabriele Grieshop und Martin Winter (Institut für Didaktik der Mathematik und des Sachunterrichts (IFD), Universität Vechta) wird – im Rahmen einer vorwiegend formativen Implementierungsevaluation am Beispiel Mathematik – die Beteiligung der Trainerinnen und Trainer an der Konzept- und Materialentwicklung von Angeboten der Stiftung „Haus der kleinen Forscher“ betrachtet.

**Band 4** (2012)

*Salman Ansari, Susanna Jeschonek, Janna Pahnke, Sabina Pauen*

Band 4 enthält vier Expertisen, die basierend auf aktuellen entwicklungspsychologischen Erkenntnissen Empfehlungen für die Entwicklung weiterer naturwissenschaftlicher, technischer und mathematischer Themenschwerpunkte der Stiftung „Haus der kleinen Forscher“ aussprechen, auf mögliche Stolpersteine hinweisen und Vorschläge für die Praxis aufzeigen. Die Expertise von Janna Pahnke und Sabina Pauen gibt einen Überblick über die Entwicklung des mathematischen und naturwissenschaftlichen Denkens und Wissens in der frühen Kindheit und zieht Schlussfolgerungen für eine darauf aufbauende frühe Bildung in diesen Bereichen. Susanna Jeschoneks Expertisen behandeln die Entwicklung des kindlichen Verständnisses der Bereiche ‚Magnetismus‘ und ‚Akustik‘ und geben Empfehlungen für die Aufbereitung von Bildungsangeboten zu diesen Themenschwerpunkten in der Praxis. In der Expertise von Salman Ansari stehen Prozesse des Lehrens und Lernens aus der Sicht der kognitiven Wissenschaften im Mittelpunkt. Ansari geht auf verschiedene Konzepte und Angebote der Stiftung „Haus der kleinen Forscher“ ein, spricht Empfehlungen für die Weiterentwicklung dieser Themen aus und verdeutlicht dies anhand von konkreten Beispielen für die praktische Umsetzung.





### Band 5 (2013)

*Yvonne Anders, Ilonca Hardy, Sabina Pauen, Jörg Ramseger, Beate Sodian, Mirjam Steffensky*

Der fünfte Band stellt Ziele naturwissenschaftlicher Bildung für Kinder und pädagogische Fach- und Lehrkräfte sowie prozessbezogene Qualitätskriterien für den naturwissenschaftlichen Unterricht im Elementar- und Primarbereich in den Fokus. Yvonne Anders, Ilonca Hardy, Sabina Pauen, Beate Sodian und Mirjam Steffensky spezifizieren in ihren Expertisen pädagogisch-inhaltliche Zieldimensionen naturwissenschaftlicher Bildung im Kita- und Grundschulalter. Neben einer theoretischen Fundierung verschiedener Zielbereiche werden Instrumente für deren Messung aufgeführt. Jörg Ramseger formuliert in seiner Expertise zehn Qualitätskriterien für den naturwissenschaftlichen Unterricht. Diese prozessbezogenen Kriterien können pädagogische Fach- und Lehrkräfte bei der Unterrichtsplanung sowie bei der Selbstevaluation naturwissenschaftlicher Angebote im Elementar- und Primarbereich unterstützen.



### Band 6 (2014)

*Yvonne Anders, Itala Ballaschk, Wolfgang Tietze*

Im sechsten Band der Schriftenreihe mit einem Geleitwort von Wolfgang Tietze berichten Yvonne Anders und Itala Ballaschk die Ergebnisse ihrer Studie zur Untersuchung der Reliabilität und Validität des Zertifizierungsverfahrens der Stiftung „Haus der kleinen Forscher“, mit dem sich Bildungseinrichtungen nach bestimmten Qualitätskriterien als „Haus der kleinen Forscher“ zertifizieren lassen können. Insgesamt konnte in der Studie das Potenzial des Verfahrens für die Messung der naturwissenschaftsbezogenen Bildungsqualität in pädagogischen Einrichtungen belegt und Ansatzpunkte für weitere Optimierungen aufgezeigt werden.

**Band 7** (2015)

*Gabriele Graube, Maja Jeretin-Kopf, Walter Kosack, Ingelore Mammes, Ortwin Renn, Christian Wiesmüller*

Der siebte Band der Reihe mit Geleitwort von Ortwin Renn fokussiert Ziele und Konzepte technischer Bildung im Elementar- und Primarbereich. Walter Kosack, Maja Jeretin-Kopf und Christian Wiesmüller spezifizieren in ihrer Expertise pädagogisch-inhaltliche Zieldimensionen technischer Bildung im Kita- und Grundschulalter. Neben einer theoretischen Fundierung verschiedener Zielbereiche werden Instrumente für deren Messung aufgeführt. Die Autoren stellen in zwei Berichten die Ergebnisse empirischer Studien dar. Zum einen wurde der Einfluss verschiedener technikdidaktischer Materialsysteme auf die kindliche Motivation, problemlösendes Denken und technische Kreativität, und zum anderen der Einfluss verschiedener technikdidaktischer Methoden auf die kindliche Motivation sowie technikspezifische Denk- und Handlungsweisen untersucht. Gabriele Graube und Ingelore Mammes beschreiben in ihrem Beitrag ein didaktisches Konzept zur Unterstützung des professionellen Handelns pädagogischer Fach- und Lehrkräfte bei der Begleitung kindlicher Bildungsprozesse in ihrer Auseinandersetzung mit Natur und Technik.

**Band 8** (2017)

*Christiane Benz, Meike Grüßing, Jens Holger Lorenz, Kristina Reiss, Christoph Selter, Bernd Wollring*

Der achte Band der Reihe mit einem Geleitwort von Kristina Reiss stellt die Ziele und Gelingensbedingungen mathematischer Bildung im Elementar- und Primarbereich in den Fokus. Christiane Benz, Meike Grüßing, Jens Holger Lorenz, Christoph Selter und Bernd Wollring spezifizieren in ihrer Expertise pädagogisch-inhaltliche Zieldimensionen mathematischer Bildung im Kita- und Grundschulalter. Neben einer theoretischen Fundierung verschiedener Zielbereiche werden Instrumente für deren Messung aufgeführt. Des Weiteren erörtern die Autorinnen und Autoren Gelingensbedingungen für eine effektive und wirkungsvolle frühe mathematische Bildung in der Praxis. Sie geben zudem Empfehlungen für die Weiterentwicklung der Stiftungsangebote und die wissenschaftliche Begleitung der Stiftungsarbeit im Bereich Mathematik. Das Schlusskapitel des Bandes beschreibt die Umsetzung dieser fachlichen Empfehlungen in den inhaltlichen Angeboten der Stiftung „Haus der kleinen Forscher“.



## Frühe informatische Bildung – Ziele und Gelingensbedingungen für den Elementar- und Primarbereich

Im Rahmen der Schriftenreihe „Wissenschaftliche Untersuchungen zur Arbeit der Stiftung ‚Haus der kleinen Forscher‘“ werden regelmäßig wissenschaftliche Beiträge von renommierten Expertinnen und Experten aus dem Bereich der frühen Bildung veröffentlicht. Diese Schriftenreihe dient einem fachlichen Dialog zwischen Stiftung, Wissenschaft und Praxis, mit dem Ziel, allen Kitas, Horten und Grundschulen in Deutschland fundierte Unterstützung für ihren Bildungsauftrag zu geben.

Der vorliegende neunte Band der Reihe mit einem dialogischen Geleitwort von Peter Hubwieser und Johannes Magenheim stellt die Ziele und Gelingensbedingungen informatischer Bildung im Elementar- und Primarbereich in den Fokus.

Nadine Bergner, Hilde Köster, Johannes Magenheim, Kathrin Müller, Ralf Romeike, Ulrik Schroeder und Carsten Schulte spezifizieren in ihrer Expertise pädagogisch-inhaltliche Zieldimensionen informatischer Bildung im Kita- und Grundschulalter. Neben einer theoretischen Fundierung verschiedener Zielbereiche erörtern die Autorinnen und Autoren Gelingensbedingungen für eine effektive und wirkungsvolle frühe informatische Bildung in der Praxis. Sie geben zudem Empfehlungen für die Entwicklung der Stiftungsangebote und die wissenschaftliche Begleitung der Stiftungsarbeit im Bereich Informatik. Nadine Bergner und Kathrin Müller beschreiben in ihrer Fachempfehlung eine Auswahl an Informatiksystemen für Kinder im Kita- und Grundschulalter und geben anhand fachlicher Kriterien Empfehlungen für besonders geeignete Informatiksysteme und deren Verwendung im Elementar- und Primarbereich.

Das Schlusskapitel des Bandes beschreibt die Umsetzung dieser fachlichen Empfehlungen in den inhaltlichen Angeboten der Stiftung „Haus der kleinen Forscher“ – mit und ohne Computer.

ISBN 978-3-8474-2107-8



[www.budrich-verlag.de](http://www.budrich-verlag.de)  
[www.haus-der-kleinen-forscher.de](http://www.haus-der-kleinen-forscher.de)