

Computational Thinking und COOL- Informatics an der AAU Klagenfurt

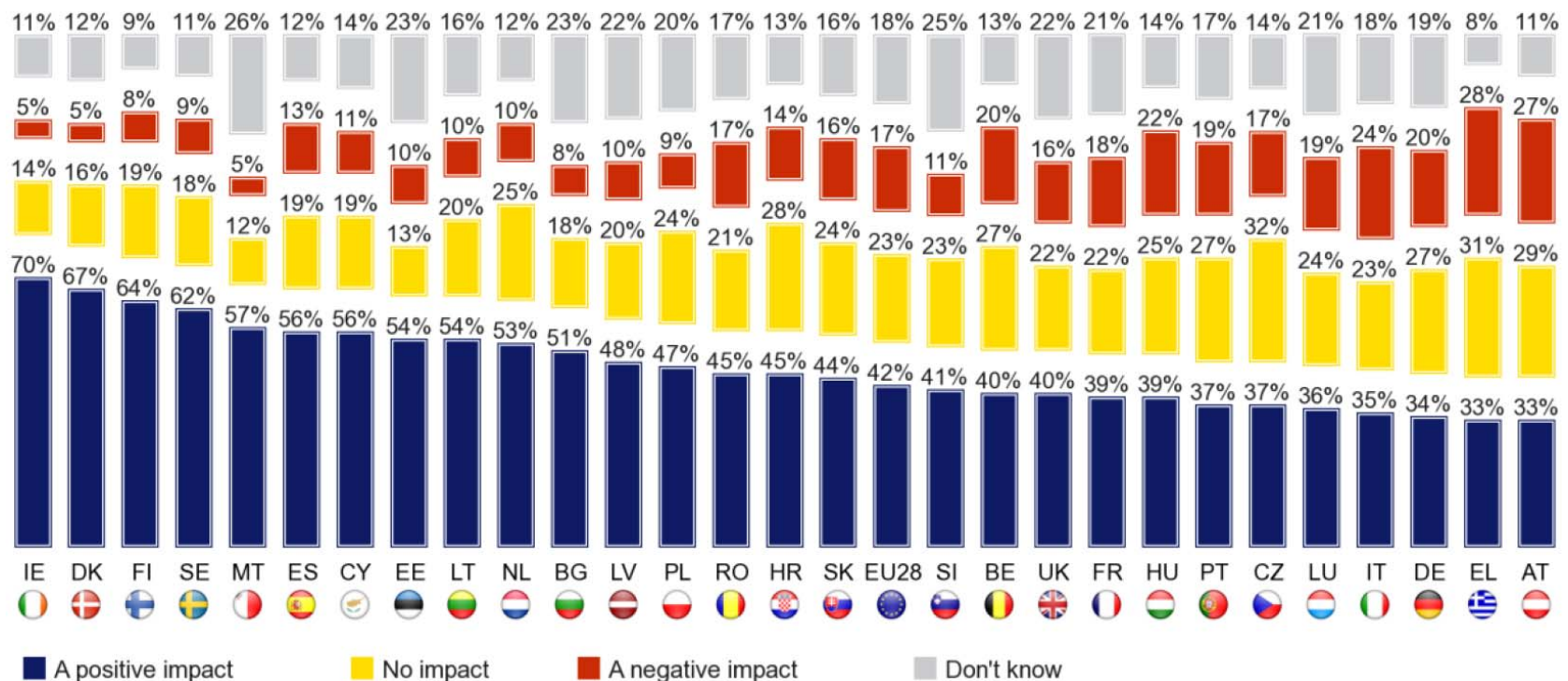
IMST-Tagung 2016, PH Kärnten, Fachdidaktiktag,
28. September 2016 / Andreas Bollin, Marianne Rohrer



FAKULTÄT FÜR TECHNISCHE WISSENSCHAFTEN

Ein Blick in die Zukunft

- In nur **11 von 29 Ländern** glaubt die Mehrheit der Bevölkerung, dass Forschung und technische Innovationen einen **positiven Einfluss** haben! [European Commission. SPECIAL EUROBAROMETER 419, Public Perceptions of Science, Research and Innovation, page 40, Oct. 2014]



CT an der AAU

- Teil (I) – Motivation
 - Informatik
 - Begriffsdefinitionen
- Teil (II) – Das WAS
 - Computational Thinking (CT)
 - Curricula und deren Grenzen



Uni für die Kleinen,
AAU Klagenfurt, 19.4.2013

- Teil (III) – Das WIE
 - COOL – Informatics
 - Prinzipien
- Teil (IV) Beispiele
- Teil (V) Ausblick

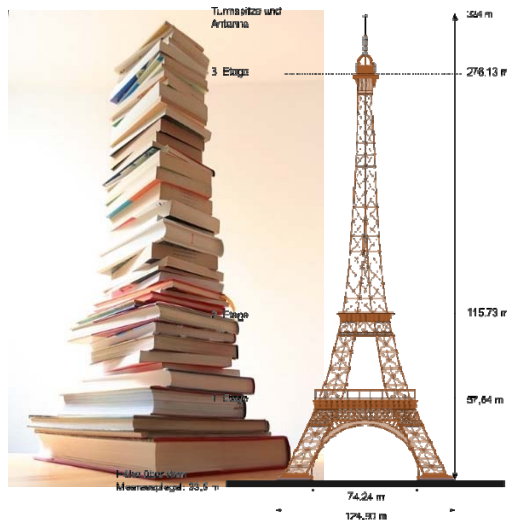


Uni für die Kleinen,
AAU Klagenfurt, 19.4.2013

Motivation (1/3)

- **Um was geht es eigentlich?**

- Technologie ist **überall**
- Wichtiger **Wirtschaftsfaktor**
- Natürlich auch eine Portion **Hype / Mystery**



- **Komplexität** steigt beständig
 - Voyager ... 3 KLOC (1977),
 - Mars Rover 160 KLOC (2003),
 - ISS ... 5 MLOC (2009),
 - Boeing 787 ... 6.4 MLOC (2011)
 - GMC ... 100 MLOC (2012)

➔ **Was** können wir tun um die Gesellschaft auf aktuelle und zukünftige Herausforderungen / Probleme / Möglichkeiten vorzubereiten?

Motivation (2/3)

- **Informatik** ([Computer | Computing] Science):
 Teil von **MINT** (Mathematik, Informatik, Naturwissenschaft und Technik)

Def. (1): "Duden Informatik", 1993.

Def. (2)(3)(4): Einführung in die Informatik, Bollin/Eder/Hitz, 2015.

(1) Informatik: Wissenschaft der systematischen Verarbeitung von **Information** – besonders der automatischen Verarbeitung von Information mit Hilfe von **Computer**

(3) Wissen:

- (a) dient der Wissensvermehrung,
- (b) ist in **Sprache** gefasst
- (c) ist an einen physischen Träger gebunden

(2) Information:

- (a) Statisch/Objektbezogen: **Wissen** das in passende Form gebracht ist
- (b) Prozessorientiert: Ablauf oder Ergebnis der Übermittlung oder Aneignung von **Wissen** (→ Bedeutung)

(4) Computer:

Er dient der maschinellen Verarbeitung von Information (zu diesem Zweck in geeigneter Form abgelegt). Zur Verarbeitung befolgt er detaillierte Anweisungen (Programm), die er automatisch interpretiert

- Ein COMPUTER ist eine SPRACH-INTERPRETATIONS-MASCHINE
- Informatik ist mehr als nur Programmierung, ICT bzw. Medienkompetenz

Motivation (3/3)

- Es geht uns daher um **Sprachen, der Verwendung von Sprachen und Information.**
 - Von Interesse sind damit z.B. nicht ganz spezielle Anwendungen sondern das Gemeinsame der Anwendungen. Durch die „passende“ Art zu denken können wir nämlich
 - Probleme und Muster verstehen
 - Planen lernen und so
 - Neues und Nützliches schaffen



<http://blogfarm15.mb.uni-flensburg.de/lisasophie/wp-content/uploads/sites/71/2016/01/Kulturtechniken.jpg>

➔ **4th Cultural-Technology**
(J. Wing [1])

➔ Prägt unter anderem die Projekte
an der AAU Klagenfurt

[1] Computational Thinking. J. Wing. CACM 49,3 (2006)

Computational Thinking (Das WAS) (1/4)

- Es gibt eine “klassische Definition”:

Computational Thinking (CT) is the **thought processes** involved in formulating problems and their solutions so that the solutions are represented in a form that can be effectively carried out by an information-processing agent. [2]

- Aber ... mehrere “Schulen”, die CT verschieden sehen:
 - im breiten Sinn (Menge an Problemlösefähigkeiten [3]),
 - im engen Sinn (starker Codierungsbezug [4]),
 - die goldene Mitte (Codieren NACH spielerischer Erfahrung mit CT [5])

[2] Gallenbacher. Abenteuer Informatik. Hands-on exhibits for learning about computational thinking. WiPCSE'12, Germany (2012).

[3] Dierbach,C., Hochheiser,H., Collins,S., Jerome, G, Ariza,C., Kellever,T., Kelinsasser,W., Dehlinger, J., Siddharth, K. A model for piloting pathways for computational thinking in general education. SIGCSE '11. ACM, 257-262 NY (2011).

[4] Seiter,L., Foreman,B. and Carroll,J. Modeling the Learning Progressions of Computational Thinking of Primary Grade Students. 9th international ACM conference on International computing education research. 59-66 NY (2013).

[5] As cited in: E. Cole. On Pre-requisite Skills for Universal Computational Thinking Education. ICER'15, Omaha, USA (2015).

Computational Thinking (Das WAS) (2/4)

- Duncan und Bell analysierten versch. Curricula [6]:

Themes	England Key Stage 1 (students from 5-7 years old)	Australia F-2 (students from 5-8 years old)	CSTA L1:3 (students from 5 to 8 years old)
Algorithms	Understand what algorithms are	Sorting and patterns	Logical problems
Programming	Create and debug simple programs; relationship with algorithms	Step-by-step proce- dures; sequences; instructions, robotic toys	Purpose of software; turtle instructions
Data representation		Patterns and symbols; pixels and file size	How 0s and 1s represent information
Digital devices & infrastructure		HW & SW components, features of a device, data transfer	Use input/output devices

[6] C. Duncan, T. Bell. Pilot Computer Science and Programming Course for Primary School Students. WiPSCE '15, UK (2015)

Computational Thinking (Das WAS) (3/4)

- Duncan und Bell analysierten versch. Curricula [6] (Forts.):

Themes	England Key Stage 1 (students from 5-7 years old)	Australia F-2 (students from 5-8 years old)	CSTA L1:3 (students from 5 to 8 years old)
Digital devices and infrastructure	Create, organize, store, manipulate and retrieve content	Capture and manipulate data e.g. photo; downloading information; collect, organize, present data; use common Software	Writing tools; Organizing information (e.g. sorting); gather and communicate information; age appropriate research; use and create multimedia; concept mapping
Humans and computer	Recognize use beyond school; use safely and respect	Ergonomics, digital devices in everyday life; collaboration; ethical and safe use	Work collaboratively; careers that use computing; legal and ethical behavior

[6] C. Duncan, T. Bell. Pilot Computer Science and Programming Course for Primary School Students. WiPSCE '15, UK (2015)

Computational Thinking (Das WAS) (4/4)

- Aber, um von CT zu profitieren [1, p. 35] sind auch folgende Punkte wichtig:
 - „Conceptualization, not programming“
→ **Modellierung und Abstraktion** auf unterschiedlichen Ebenen
 - „A way, human, not computers think“
→ **Problemlösen**, mit/**OHNE** einen Rechner
 - „Complements/combines mathematical/engineering thinking“
→ **Sprachbeherrschung** nicht vernachlässigen
 - „For everyone, everywhere“
→ **Integration** in verschiedene Unterrichtsfächer, eine Vielzahl an Themen betreffend, sodass es ein integraler Teil menschlichen Bestrebens wird

COOL Informatics (Das WIE) (1/4)

- Projekte wie “Informatik – ein Kinderspiel”, “Informatik erLeben” [7], “Computer Science Unplugged” [8] motivierten uns 2013
 - unsere **eigene** “**Informatik-Werkstatt**” zu starten, und
 - unseren **eigenen** Lehransatz zu definieren (“COOL Informatics”) [9]



[7] R.T. Mittermeir, E. Bischof, K. Hodnigg. Showing core-concepts of informatics to kids and their teachers. In Teaching Fundamentals Concepts of Informatics. Springer Berlin Heidelberg, 143-154 (2010)

[8] T. Bell, L. Lambert, D. Marghitu. CS unplugged, outreach and CS kinesthetic activities. In Proceedings of the 43rd ACM technical symposium on Computer Science Education (2012)

[9] B. Sabitzer. A Neurodidactical Approach to Cooperative and Cross-curricular Open Learning: COOL Informatics. Habilitation thesis. Alpen-Adria-Universität Klagenfurt (2014)

COOL Informatics (Das WIE) (2/4)

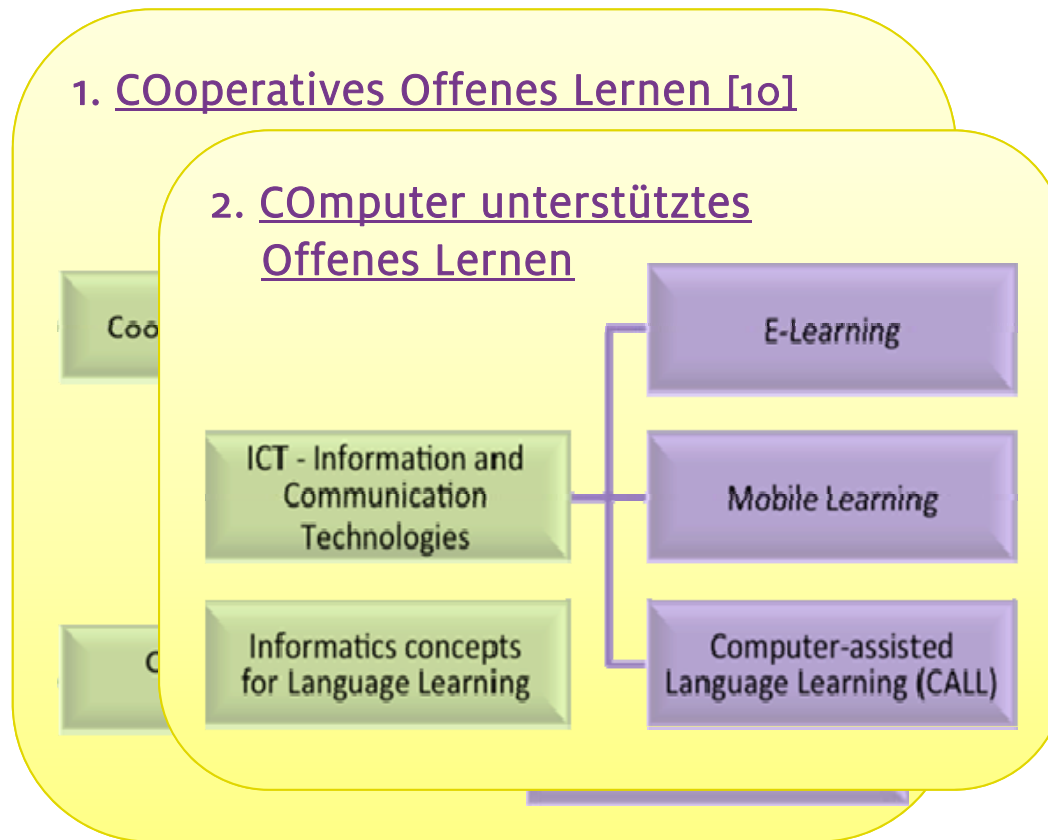
- “COOL Informatics” ist ein Akronym für:



[10] B. Greimel-Fuhrmann. Was ist und was kann Cooperatives Offenes! Lernen (COOL)?! Neues Unterrichtskonzept. Charakteristika, Chancen und Risiken. wissenplus 3-06/07 (2006)

COOL Informatics (Das WIE) (2/4)

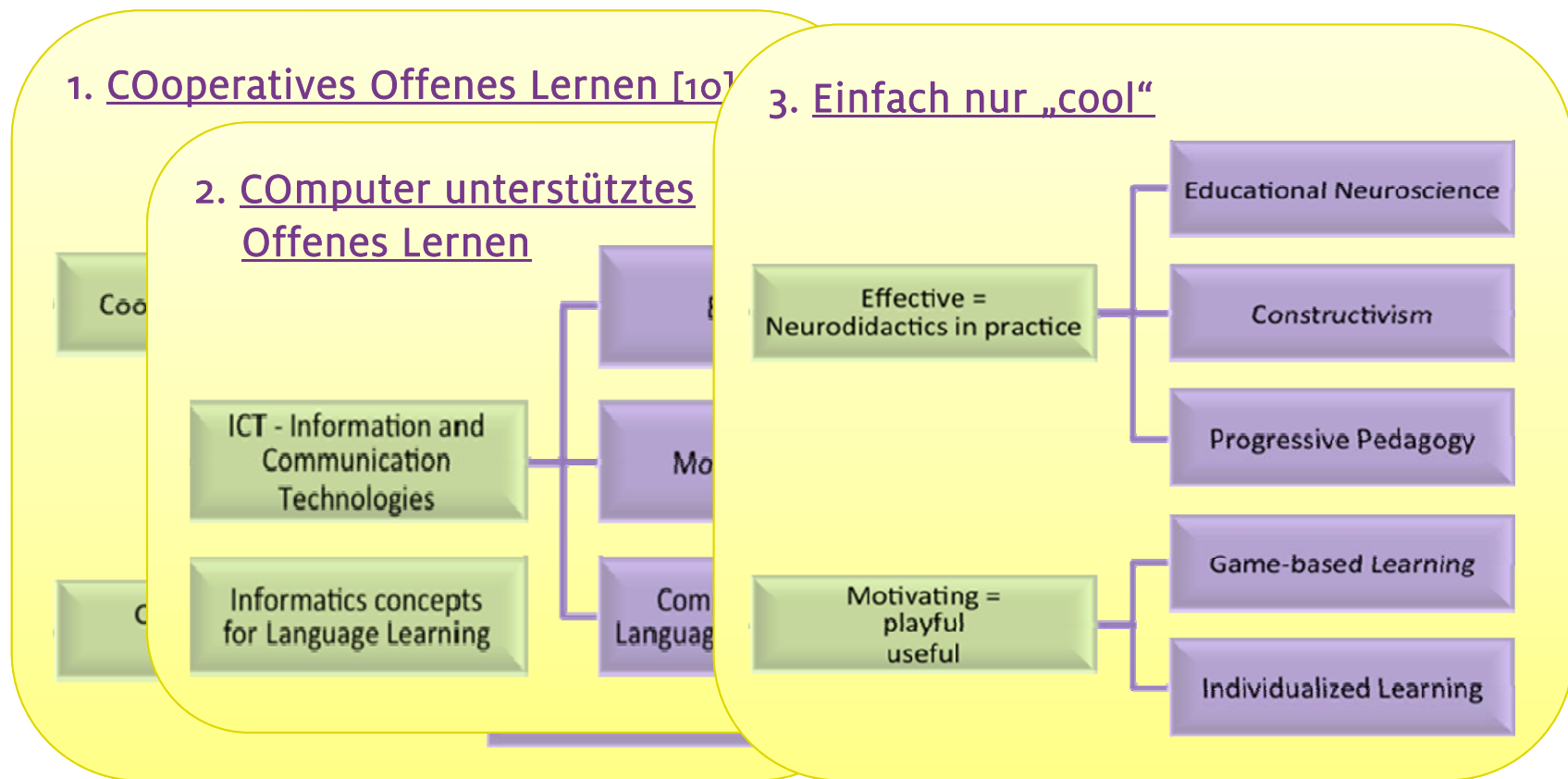
- “COOL Informatics” ist ein Akronym für:



[10] B. Greimel-Fuhrmann. Was ist und was kann Cooperatives Offenes! Lernen (COOL)?! Neues Unterrichtskonzept. Charakteristika, Chancen und Risiken. wissenplus 3-06/07 (2006)

COOL Informatics (Das WIE) (2/4)

- “COOL Informatics” ist ein Akronym für:



[10] B. Greimel-Fuhrmann. Was ist und was kann Cooperatives Offenes! Lernen (COOL)?! Neues Unterrichtskonzept. Charakteristika, Chancen und Risiken. wissenplus 3-06/07 (2006)

COOL Informatics (Das WIE) (3/4)

- “COOL Informatics” basiert auf 4 Prinzipien [9, p.26]

Principle	Teaching & Learning Method	Neurodidactical Basis
Discovery	Solution-based learning (worked examples); Step-by-step instructions + tasks Observational learning; Video tutorials; Hands-on, Minds-on Learning with all senses	Pattern recognition Mirror neurons Individual learning rhythm modality / multimedia effect
Cooperation	Team and group work; Peer tutoring and –teaching; Pair programming; Cross-curricular learning; Project-based learning	“A joy (=knowledge) shared is a joy (=knowledge) doubled.” Recall = re-storage in long-term memory Integrating individual needs, talents and competences as well as practical relevance

[9] B. Sabitzer. A Neurodidactical Approach to Cooperative and Cross-curricular Open Learning: COOL Informatics. Habilitation thesis. Alpen-Adria-Universität Klagenfurt (2014)

COOL Informatics (Das WIE) (4/4)

- “COOL Informatics” basiert auf 4 Prinzipien [9, p.26] (Forts.)

Principle	Teaching & Learning Method	Neurodidactical Basis
Individuality	Competence-based learning; Questioning; Self-organized learning with compulsory and optional tasks	Connecting new information to previous knowledge, Considering individual interests, needs, tasks, methods and learning rhythm
Activity	Hands-on, Minds-on; Learning by doing; Learning by animation, simulation and playing; Learning by playing and designing games (creative learning)	Knowledge must be newly created (constructed) by each learner (= constructivism) Learning is an active process (=progressive education)

[9] B. Sabitzer. A Neurodidactical Approach to Cooperative and Cross-curricular Open Learning: COOL Informatics. Habilitation thesis. Alpen-Adria-Universität Klagenfurt (2014)

Beispiele (1/5)

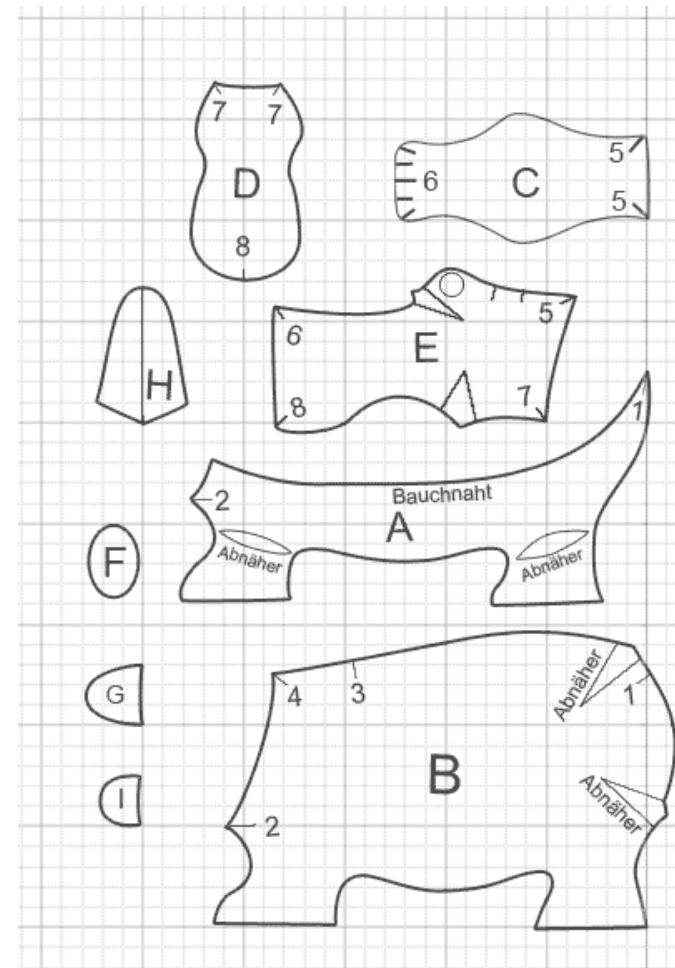
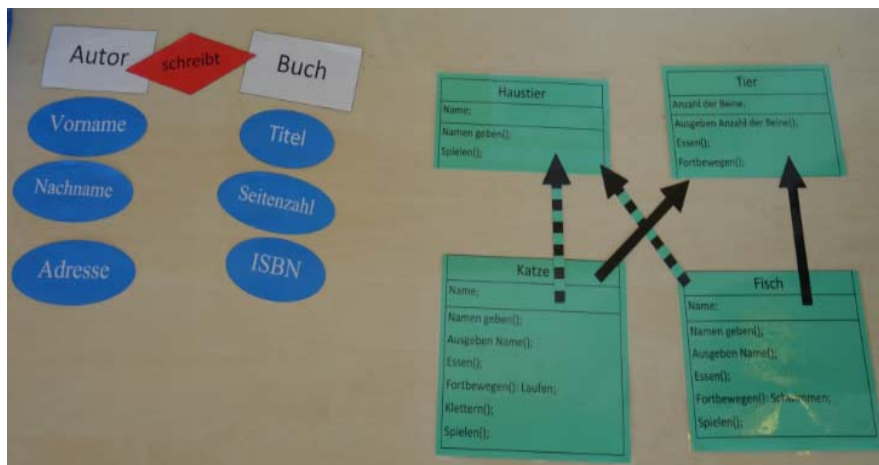
- **[Mobile] Informatik-Werkstatt** an der AAU-Klagenfurt
 - Mehr als 2000 TeilnehmerInnen (von jeder Altersstufe an) seit 2013
 - Behandelt “Standard” Themen wie
 - Boolean Algebra / Logik
 - Binärzahlen
 - Codierung und Verschlüsselung
 - Internet und Netzwerke
 - Computer Systeme, Hardware
 - Informationsverarbeitung

Beispiele (1/5)

	Besucherinnen und Besucher	Praktikantinnen und Praktikanten	Studentinnen und Studenten
Mobile Werkstatt 2013/14	480		8
Mobile Werkstatt 2014/15	450	4	9
Sommer Werkstatt 07/2014	76	7	
Sommer Werkstatt 07/2015	312	13	
Sommer Werkstatt 07/2016	163	21	
Informatik-Camp 07/2016	29		
Jahres-Werkstatt 2015/2016	495		9
Gesamt 10/2013 bis 07/2016	2005	45	26

Beispiele (2/5)

- Die Werkstatt **fokussiert** auf:
 - Angebote **un|plugged**
 - **Modellierung**, Abstraktion
 - **Kommunikation**, Teams
 - **Lehrplanübergreifende Projekte** (e.g. Schnittmuster, Sprach-Unterricht)



Beispiele (3/5)

- **CT_VS** – ein Projekt gestartet Ende 2015 mit dem BMBF
- Wo findet man CT im Lehrplan der Primarstufe ?
 - Analyse der informatischen Wissensgebiete (Basierend auf dem ACM/IEEE CSC 2013)
 - Analyse internationaler Curricula
 - Analyse des VS-Lehrplans → **Leitfaden für Lehrer**

	A	B	C	D	E	F	G	H	
1	Lehrplan-Integration								
2	Stufe	Fach	Erfahrungs- und Lernbereich	Thema	Beschreibung/Aktivitäten	Informatik	Aktivitäten	IT-Unterstützung?	Konzept
3	Vorschule	Sachbegegnung	Gemeinschaft	Orientierung im neuen Lebensraum	Orientierung	Algorithmen	mit verbundenen Augen gehen auf Anweisung, "gegenseitig Programmieren"	Nein	Kennenlerne Beschreibbar braucht um Modellierung werden.
4	Vorschule	Sachbegegnung	Natur	Begegnung mit Tieren und Pflanzen um Uns	Gemeinsame Merkmale und Unterschiede finden (zB Tiere mit Fell; Tiere, die schwimmen können usw.)	Modellierung	zuordnen von Gegenständen zu Klassen, Klassendiagramme	Nein	Kennenlerne sie Ereignis Informatik c Kennenlerne
5	Vorschule	Sachbegegnung	Natur	Naturerscheinungen	Beobachtungen und einfache Versuche machen Tageszeiten (Tag, Nacht) Himmelskörper Wetter Jahreszeiten	Aussagenlogik	Arbeitsblätter	Nein	Kennenlerne dass das Ve welche ernei zum Thema
6	Vorschule	Sachbegegnung	Kind und Gesundheit	Abbau übertriebener Ängste	Verhalten in Situationen die Kindern Angst machen	Algorithmen	"Anleitung" zur Bewältigung solcher Situationen selbst erstellen -> Aktivitätsdiagramm	Nein	
7	Vorschule	Sachbegegnung	Raum	Orientierung auf dem Schulweg	Orientierungsgesichtspunkte finden und beschreiben, Wege beschreiben	Algorithmen	Wegbeschreibung, Orientierungsspiele	Nein	
8	Vorschule	Sachbegegnung	Raum	Raubegriffe	rechts, links, etc Selbst hergestellte Modelle verwenden (Einrichtung der Klasse planen;	Algorithmen	Orientierungsspiele, Beebots etc	Ja	Grundlegende Beebots.
9	Vorschule	Sachbegegnung	Raum	Arbeiten mit Modelle	Den Tagesablauf gliedern (am Morgen	Modellierung	Märchen in Modelle verpacken, Haus bauen (Kennenlernen des Klassendiagramms)	Nein	Spielerisch c

Beispiele (3/5)

- **CT_VS** – ein Projekt gestartet Ende 2015 mit dem BMPE

- **Wo**
 - **A** **Fach** Sachbegegnung
 - **A** **Erfahrungs-/ Lernbereich** Gemeinschaft
 - **A** **Thema** Orientierung im neuen Lebensraum
 - **A** **Beschreibung / Aktivität** Orientierung
 - **A** **Informatik** Algorithmen (Kategorie: AL/BA)

Stufe	Vorschule				Lehrer	IT-Unterstützung?	Konzept		
1	Lehrplan-Integration	Aktivität(en) mit verbundenen Augen gehen auf Anweisung, Vorbereitung: "gegenseitig Programmieren"							
2	Stufe	ICT Bedarf							
3	Vorschule	Konzept Kennenlernen, dass unter einem Algorithmus eine möglichst genaue Beschreibung verstanden wird und dass man dies auch braucht um gewünschte Aktionen korrekt auszuführen. (Weiterführung: Kann später auch mit Modellierung verknüpft werden, indem z.B. Aktivitätsdiagramme erstellt werden.)				Augen gehen auf Anweisung, programmieren"	Nein	Kennenlerne Beschreibur braucht um Modellierung werden.	
4	Vorschule					iständen zu Klassen,	Nein	Kennenlerne sie Ereignis Informatik c Kennenlerne	
5	Vorschule						Nein	Kennenlerne dass das Ve welche ernei zum Thema	
6	Vorschule	Sachbegegnung	Kind und Gesundheit	Abbau übertriebener Ängste	Verhalten in Situationen die Kindern Angst machen	Algorithmen	"Anleitung" zur Bewältigung solcher Situationen selbst erstellen -> Aktivitätsdiagramm	Nein	
7	Vorschule	Sachbegegnung	Raum	Orientierung auf dem Schulweg	Orientierungsgesichtspunkte finden und beschreiben, Wege beschreiben	Algorithmen	Wegbeschreibung, Orientierungsspiele	Nein	
8	Vorschule	Sachbegegnung	Raum	Raubegriffe	rechts, links, etc	Algorithmen	Orientierungsspiele, Beebots etc	Ja	Grundlegend: Beebots.
9	Vorschule	Sachbegegnung	Raum	Arbeiten mit Modelle	Selbst hergestellte Modelle verwenden (Einrichtung der Klasse planen; Plan Tagesablauf gliedern (am Morgen	Modellierung	Märchen in Modelle verpacken, Haus bauen (Kennenlernen des Klassendiagramms)	Nein	Spielerisch c

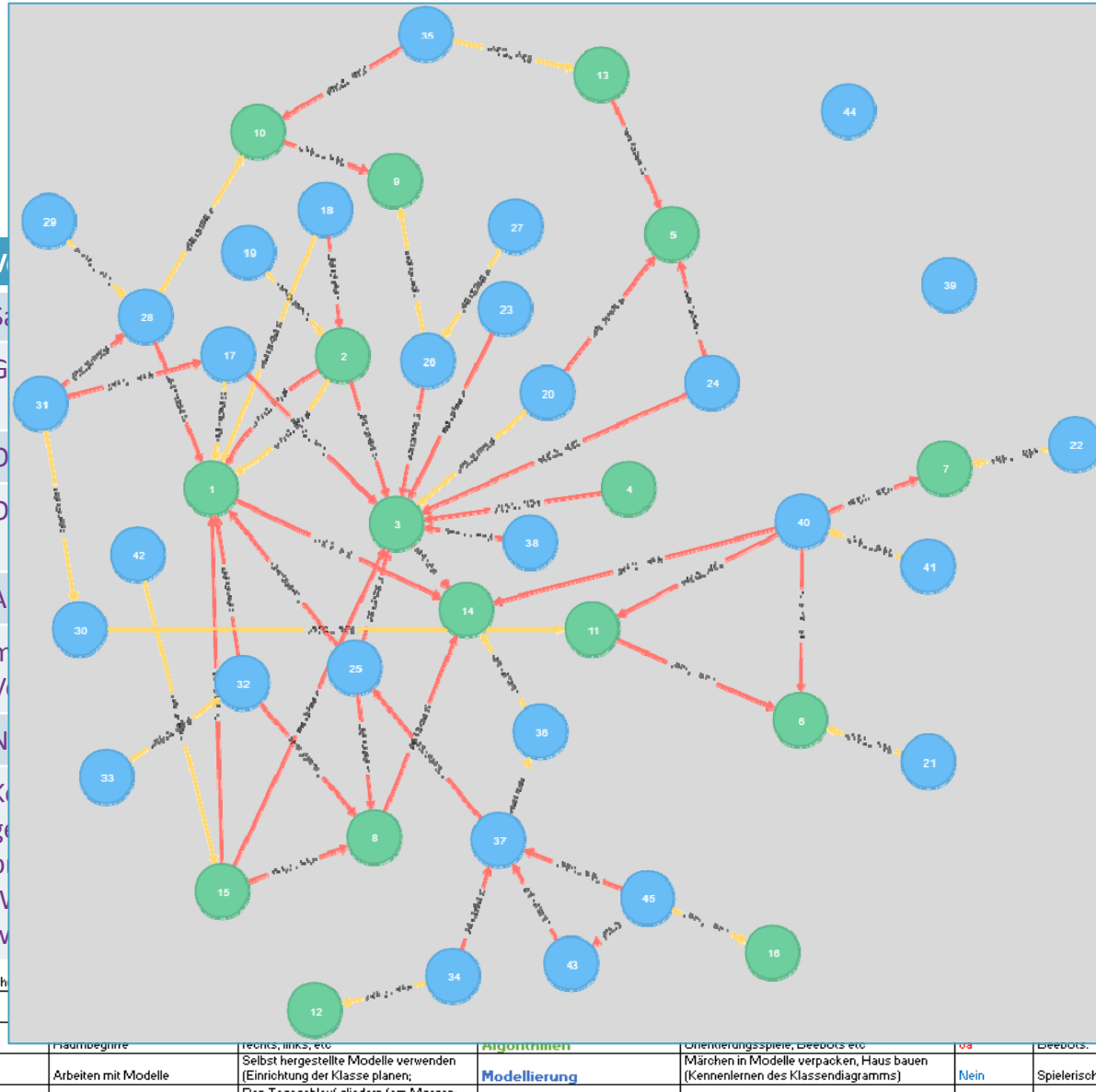
Beispiele (3/5)

- **CT_VS** – ein BMPE

- **Wo**

- **A** Erfahrungs-/ Lernbereich
- **(I** Thema
- **A** Beschreibung / Aktivität
- **A** Informatik

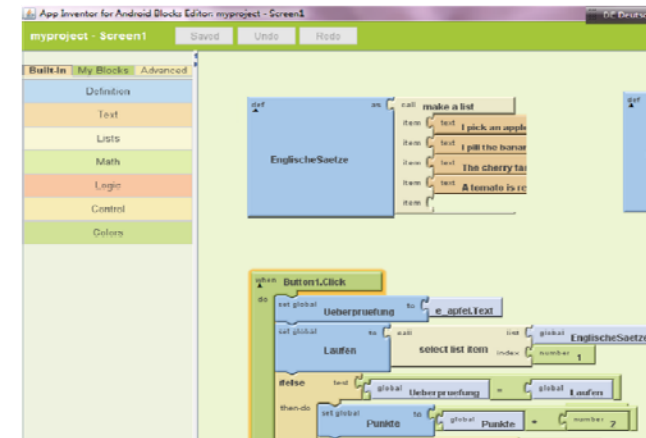
Stufe	V
Fach	Sa
Erfahrungs-/ Lernbereich	G
Thema	O
Beschreibung / Aktivität	O
Informatik	A
Aktivität(en)	m
	V
ICT Bedarf	N
Konzept	K
	g
	b
	(V
	w
1 Lehrplan-Integration	
2 Stufe	
3 Vorschule	
4 Vorschule	
5 Vorschule	
6 Vorschule	Sachbegegnung Kind und Gesundheit
7 Vorschule	Sachbegegnung Raum
8 Vorschule	Sachbegegnung Raum
9 Vorschule	Sachbegegnung Raum



Raumbegegnung	techns, inks, etc	Algorithmen	Orientierungsspiele, Beebors etc	ja	Beebors.
Arbeiten mit Modelle	Selbst hergestellte Modelle verwenden (Einrichtung der Klasse planen; Plan Tauschlauf gliedern (am Morgen	Modellierung	Märchen in Modelle verpacken, Haus bauen (Kennenlernen des Klassendiagramms)	Nein	Spielerisch c

Beispiele (4/5)

- **SW Engineering für Nicht-InformatikerInnen** [11]
Implementierung einer Sprachlern-App
- Berufsbildende Schule (8 Mädchen, Ende Sek II als "SW Ingenieure") und eine AHS (25 SchülerInnen Sek I als "Kunden").
- Verwendung des "App-Inventors" von MIT, aber der Fokus lag auf:
 - **Modellierung** und **Problem-Analyse**
 - **Kommunikations-Fähigkeiten**



MIT App-Inventor, 2015,
<http://appinventor.mit.edu>

[11] A. Bollin, S. Pasterk, P. Antonitsch, B. Sabitzer. Software Engineering in Primary and Secondary Schools – Informatics Education is More Than Programming. IEEE CSEE&T 2016. Dallas, USE (2016)

Beispiele (5/5)

- **Kompetenzen (ACM/IEEE SE2004 Knowledge Areas)**

CMP	Computing Essentials (172h): CS foundations, construction technologies & tools, formal methods	VAV	Software V&V (42h): reviews, testing, HCI testing & evaluation , problem analysis
FND	Mathematical & Engineering Fundamentals (89h): logic, measures, discrete math, statistics, economics	MGT	Software Management (19h): planning, personnel, control & configuration management
MAA	Modeling and Analysis (53h): modeling foundations , specification & validation of requirements	PRO	Software Process (13h): process concepts and implementations
PRF	Professional Practice (35h): group dynamics, psychology, professionalism and ethics , comm. skills, professionalism and ethics	QUA	Software Quality (16): quality concepts, culture, standards, and processes
DES	Software Design (45h): concepts, strategies, architectures , HCI design	EVL	Software Evolution (10h): evolution processes and activities

Zusammenfassung und Ausblick

- Wir leben in einer **sozioökonomischen Welt** und **Wechsel** ist unausweichlich. Aber ...
 - Es gibt **eine Menge an Dingen** die wir tun können um die Gesellschaft auf die Anforderungen vorzubereiten.
 - **CT** gehört dazu.
 - Die Basis ist die Vermittlung auf **COOLe Art und Weise**.
- ... um folgende Zukunft (hoffentlich) zu vermeiden ;)

"In the future, airplanes will be flown by a dog and a pilot. And the dog's job will be to make sure that if the pilot tries to touch any of the buttons, the dog bites him."

(Scott Adams / Author of „Dilbert“)

➔ Showcase (Mag. Marianne Rohrer):
Informatikwerkstatt

<http://informatikwerkstatt.jimdo.com>

Fragen:

Andreas.Bollin@aau.at