



# Schulinformatik

Unterricht gestalten

Univ.-Prof. MMag. Dr. Barbara Sabitzer

MINT-Didaktik & COOL Lab

Johannes Kepler Universität Linz

<https://www.jku.at/schule/cool-lab/>

[barbara.sabitzer@jku.at](mailto:barbara.sabitzer@jku.at)





# Lehren & Lernen

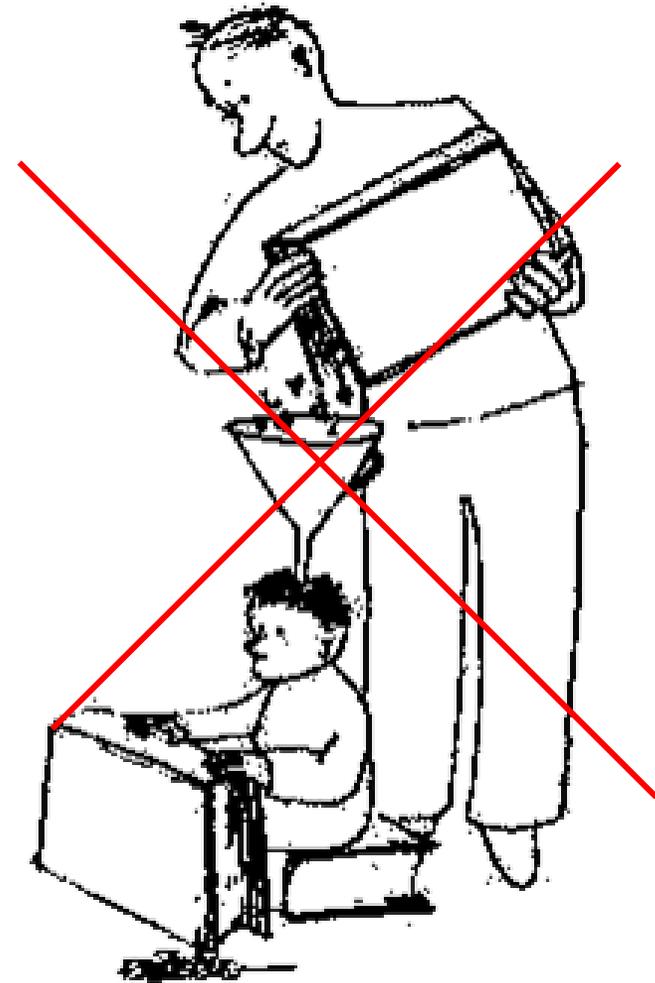
Vom Gehirn aus betrachtet

# Das Wichtigste zuerst

Wissen kann **nicht**  
**übertragen** oder **vermittelt** werden  
– es muss vom Lernenden  
**neu geschaffen** werden.

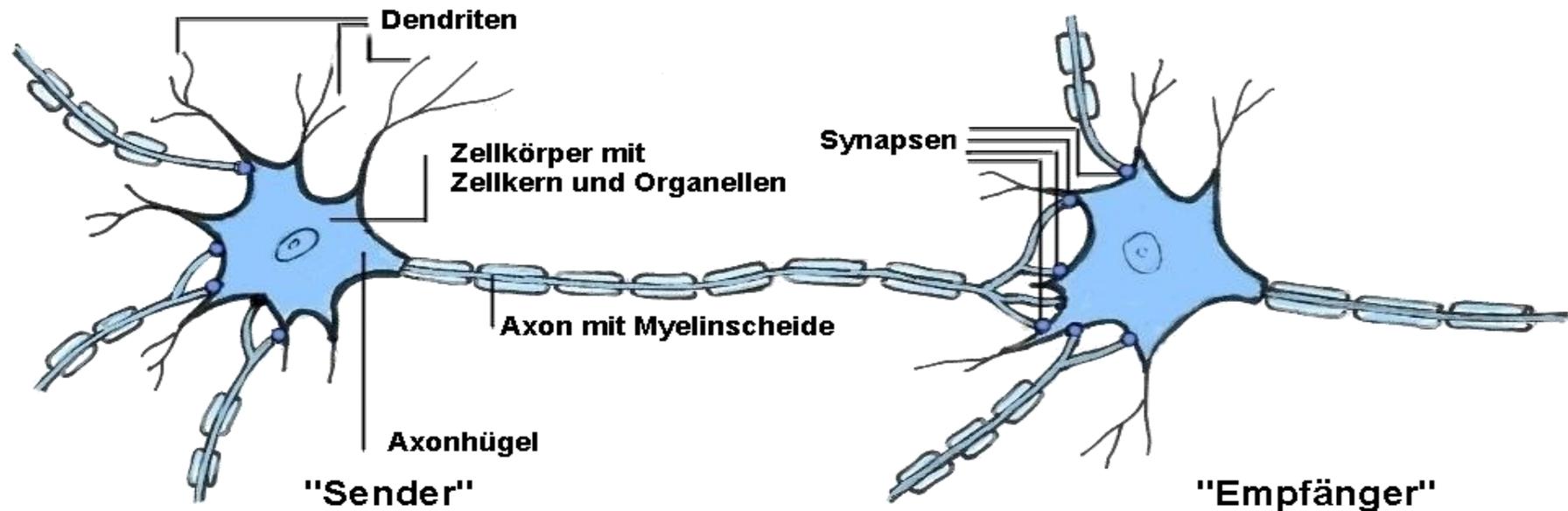


Lernen ist immer ein  
**aktiver** und **selbstgesteuerter**  
Prozess

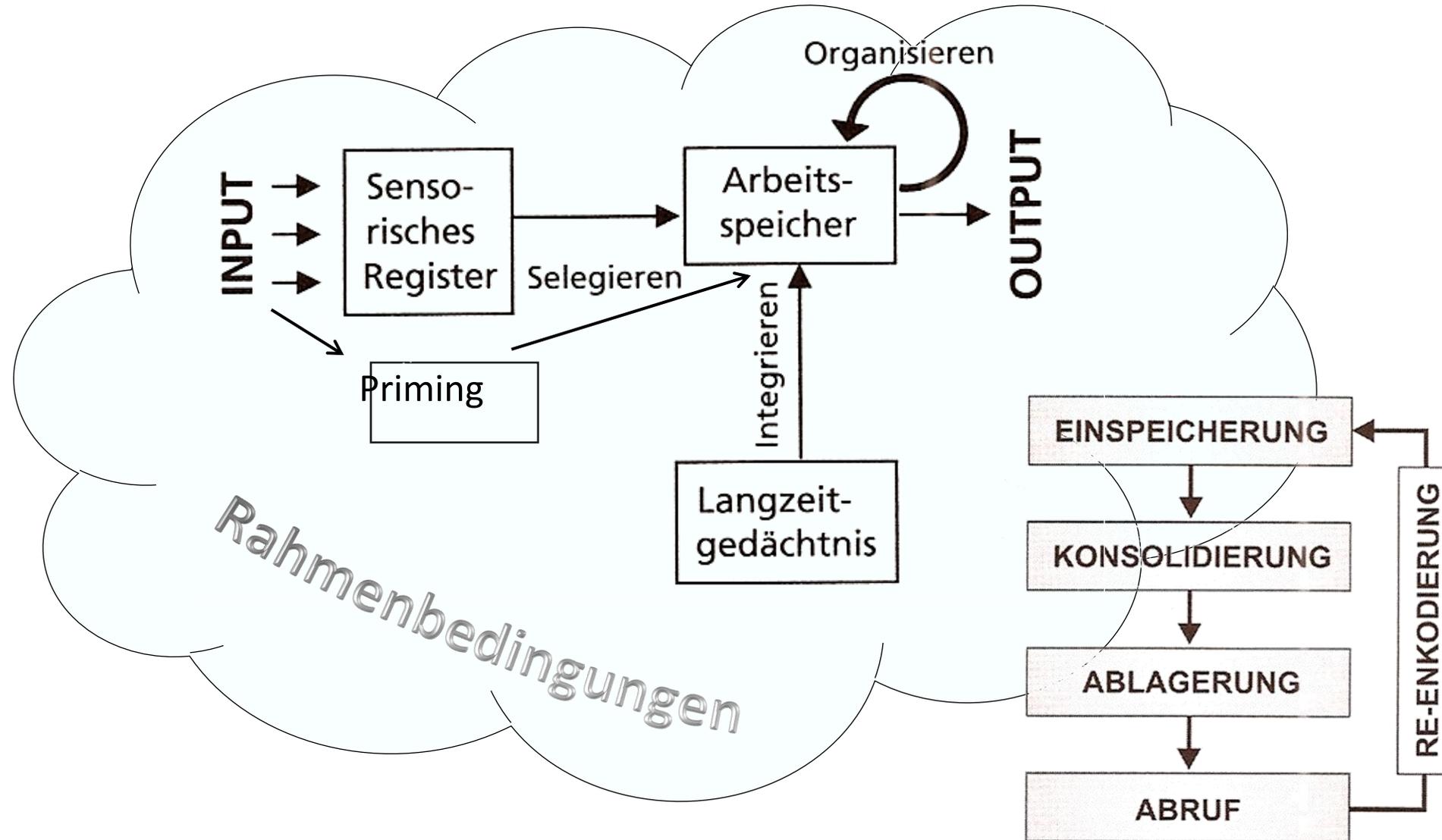


# Lernen

- Elektro-chemischer Vorgang
- verändert die Struktur des Gehirns
- Übertragung von Informationen in Form von elektrischen und chemischen Signalen
- Bildung und Verstärkung von Synapsen



# Gedächtnis



# Neurodidaktik – Das Wichtigste

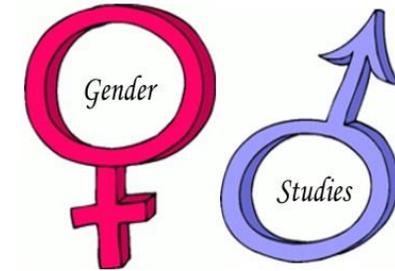
- **Automatische Mustererkennung** & Regelextraktion → Entdecken
- Das Gehirn ist ein Spiegel. → Imitation
- Doppelt hält besser! → Multimediaeffekt
- „Lernen im Schlaf“ → **Konsolidierung**
- Lernen MUSS Sinn machen → Bezug, Nutzen
- 4 Augen sehen mehr als 2 → **Kooperation**
- Neugier + **Entdecken** = Dopamin → Belohnung

# Gehirngerecht Lehren und Lernen (1/2)

## (1) Biologische Faktoren beachten (Alter, Geschlecht, Hormone, Neurotransmitter)

### → Wahlmöglichkeiten bieten

- Themen
- Lehr- und Lernmethoden
- Art und Formulierung der Aufgaben
- Lernrhythmus
- Sozialform etc.



## (2) Rahmenbedingungen gestalten (LV-Organisation, Material, Kooperation, Ort, Zeit)

### → Offener Unterricht

- COOL
- EVA-Lernen
- Lernende in Entscheidungen einbeziehen

## (3) Individuelle Faktoren beeinflussen

(Interesse, Motivation, Emotionen, Aufmerksamkeit etc.)

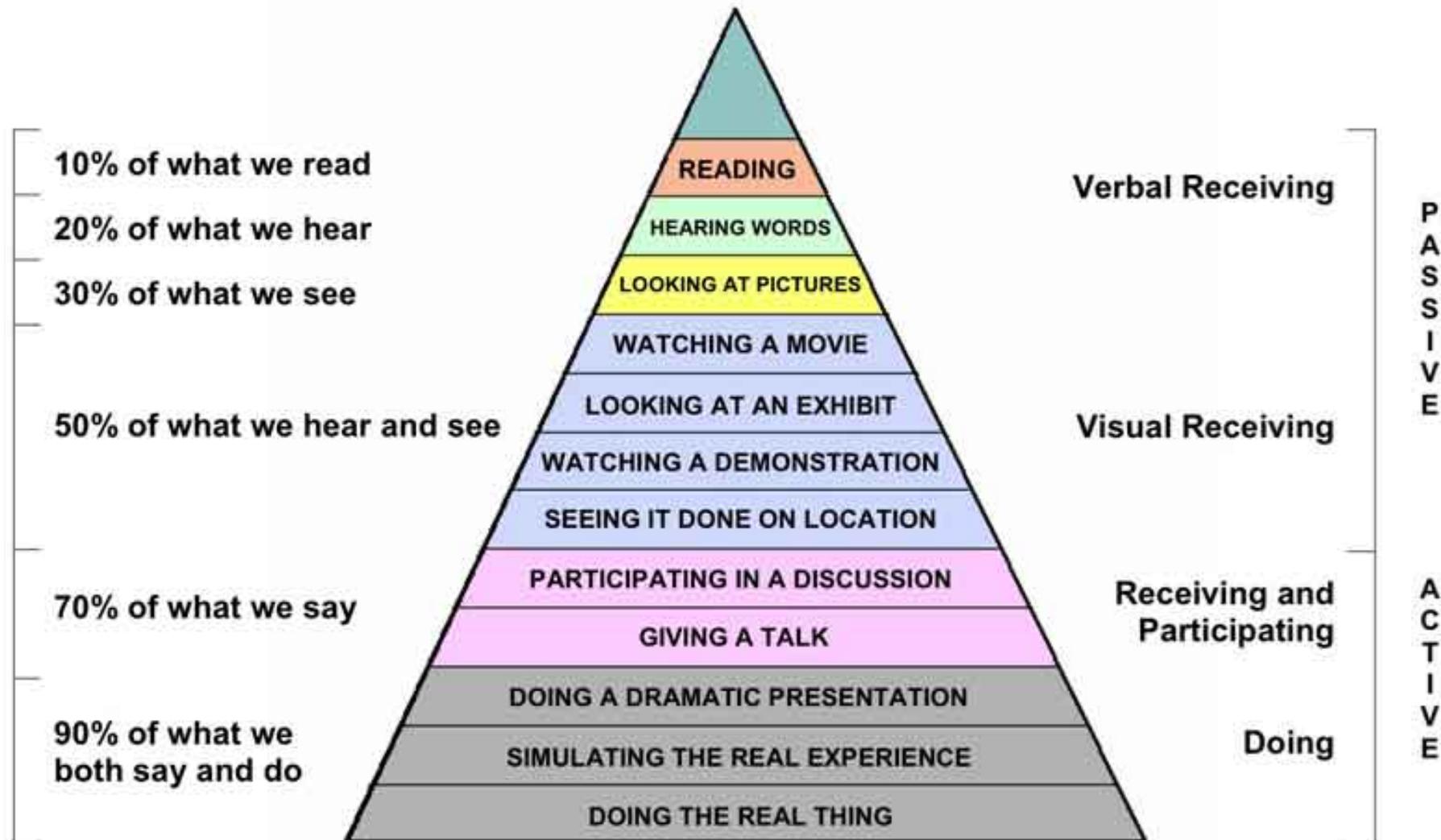
### → Wahlfreiheit

# Gehirngerecht Lehren und Lernen (2/2)

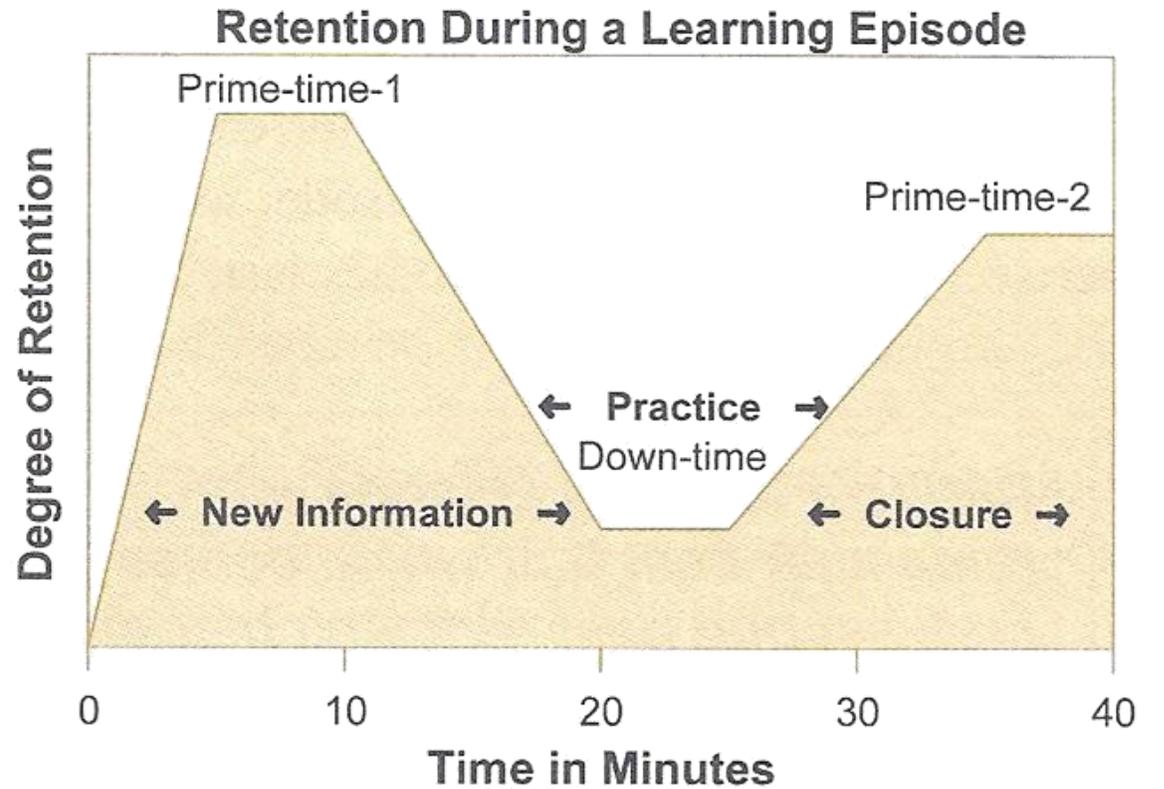
## **(4) Gehirn- und Gedächtnisfunktionen unterstützen → kognitive Lerneffekte nutzen**

- Automatische Mustererkennung – Regelextraktion
  - Entdeckendes Lernen, Schritt-für-Schritt-Aufgaben, Musterlösungen
- Konsolidierung – „Lernen im Schlaf“
  - Pausen, Schlaf
- Spiegelneuronen - Imitation
  - Lernen am Modell, Schritt-für-Schritt-Aufgaben
- Arbeitsgedächtnis
  - Lernstrategien, Kognitive Belastung reduzieren
- Effekte
  - Priming, Primacy-Recency, Multimedia
- Priming
  - Modellierung, Concept Maps, MindMaps
- Abruf = Neueinspeicherung
  - Lernen durch Lehren, Gruppenpuzzle, Pair-Programming

# Was wir uns am besten merken

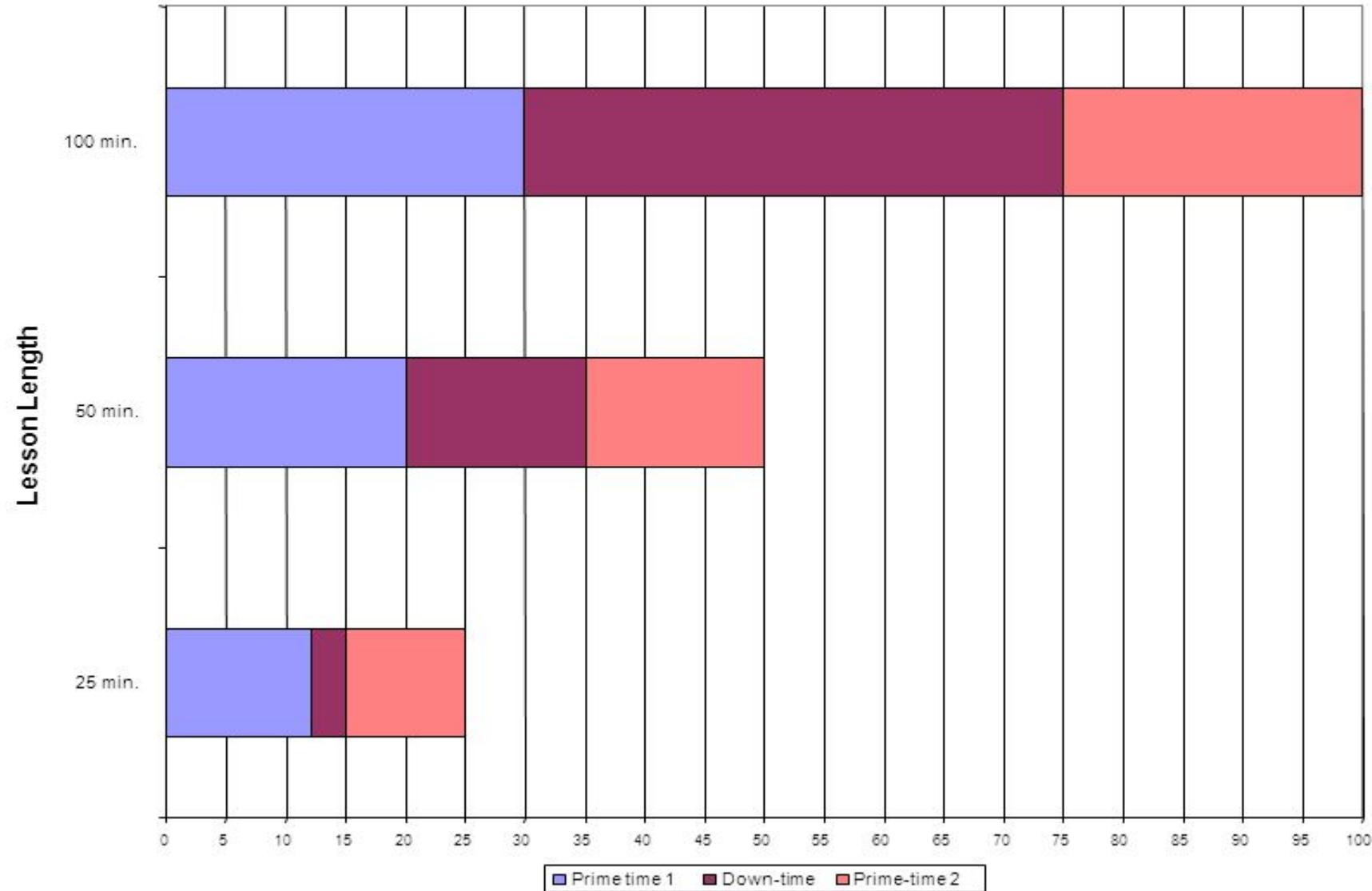


# Unterricht planen

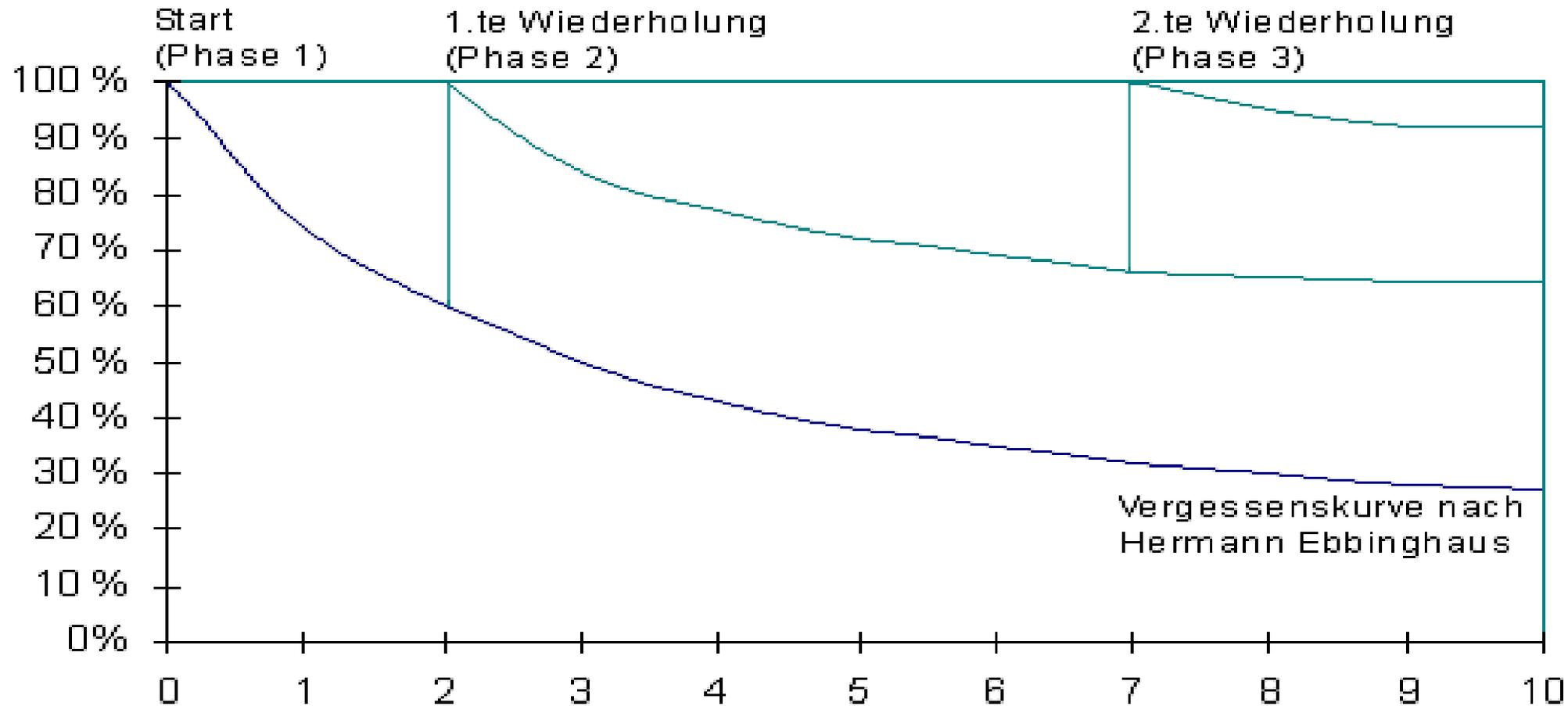


Integrating consolidation phases in classroom (Sousa, 2006)

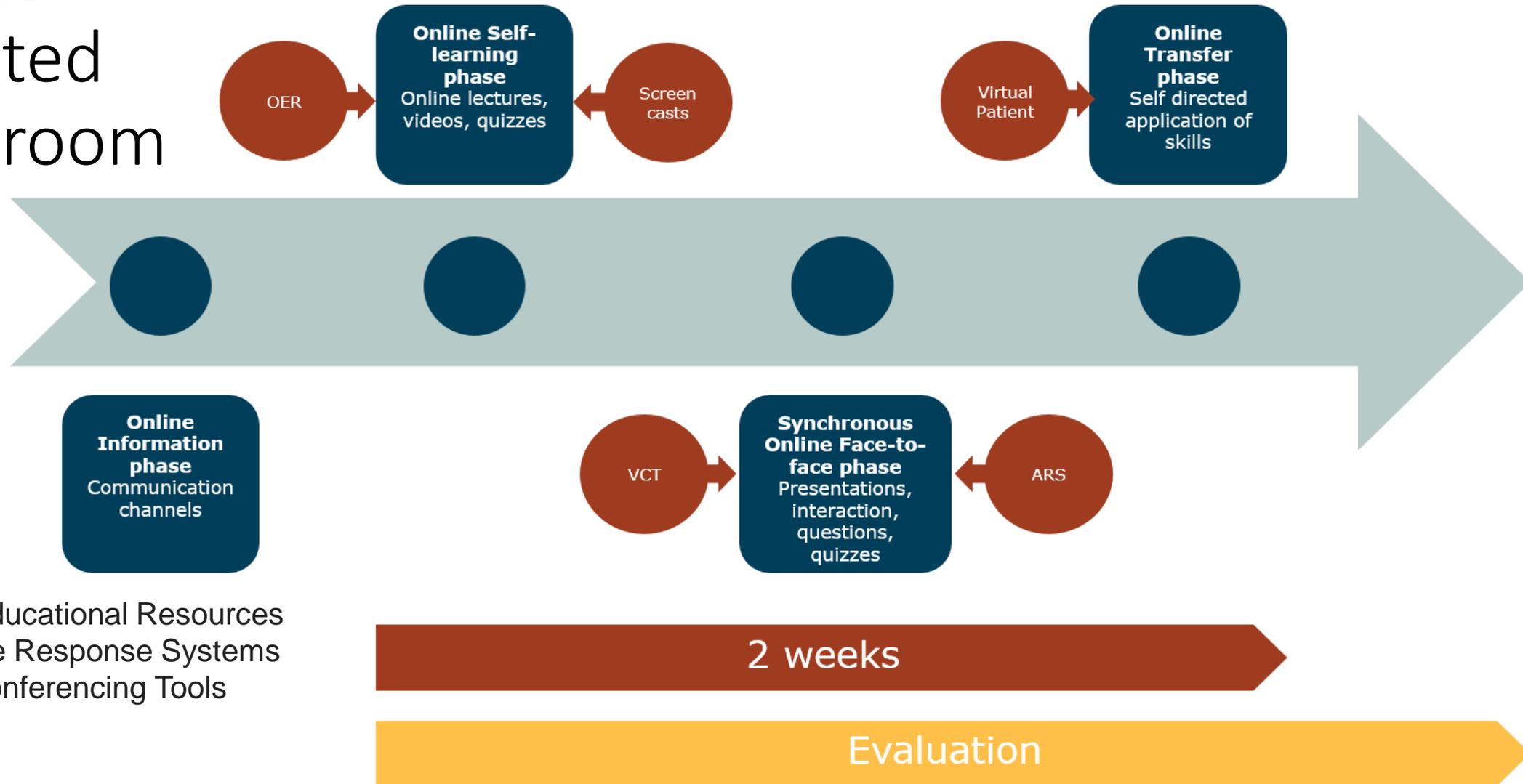
# Approximate Ratio of Prime-time to down-time in different length learning episodes



# Vergessenskurve



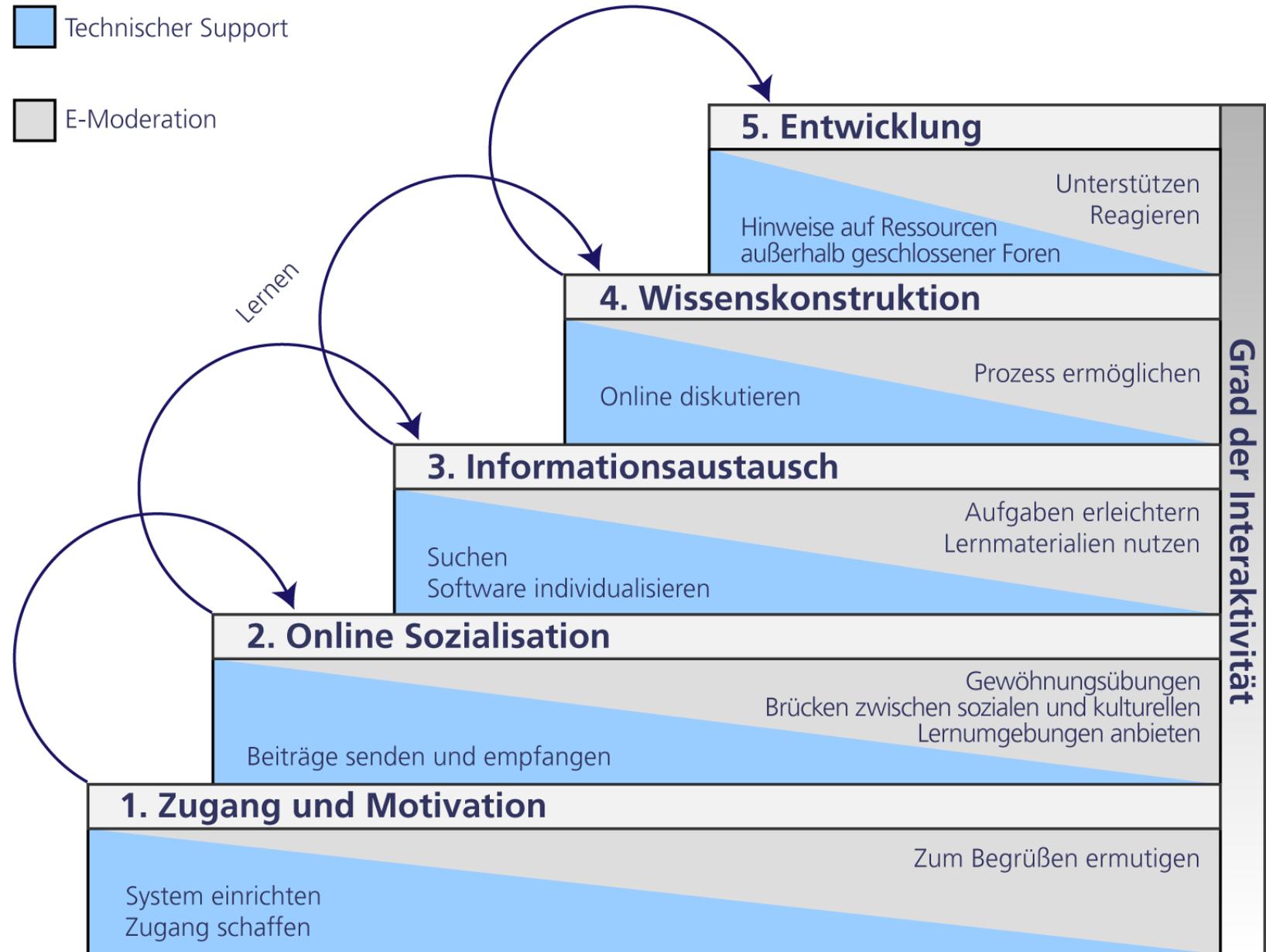
# Online Inverted Classroom



OER – Open Educational Resources  
ARS – Audience Response Systems  
VCT – Video Conferencing Tools

Tolks, D., Romeike, B. F., Ehlers, J., Kuhn, S., Kleinsorgen, C., Huber, J., ... & Hege, I. (2020). The online inverted classroom model (oICM). A blueprint to adapt the inverted classroom to an online learning setting in medical and health education. *MedEdPublish*, 9. From: [https://www.mededpublish.org/manuscriptfiles/3118/Abbildung\\_oICM\\_neu.png](https://www.mededpublish.org/manuscriptfiles/3118/Abbildung_oICM_neu.png)

# Online-Lehre Erfolgsfaktoren 5-Stufen-Modell von Gilly Salmon



# Salmon's five-stage model

## An example of a course designed according to Salmon's five-stage model

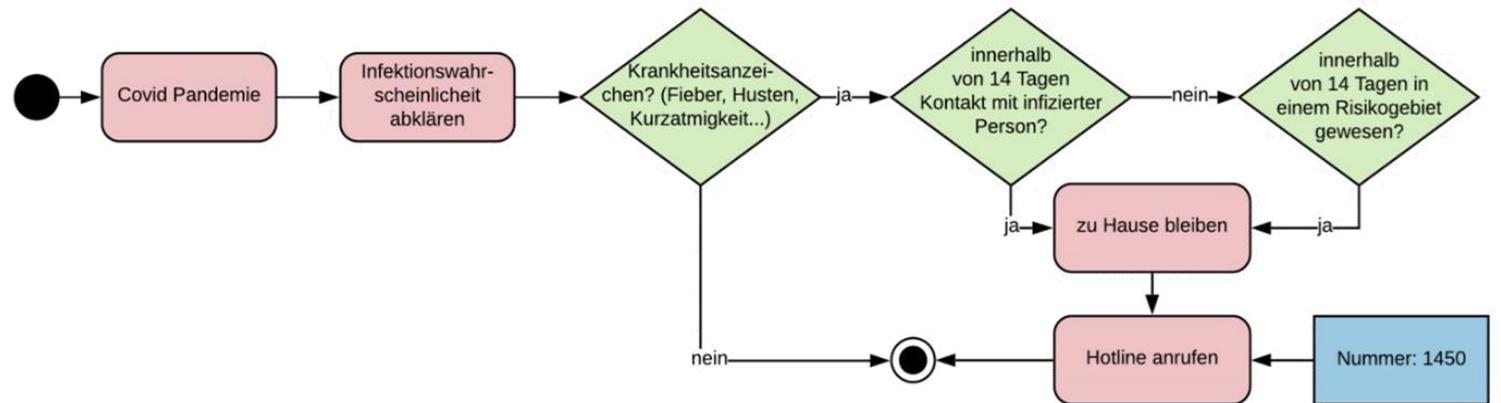
The students are postgraduate professionals, studying a blended course parttime.

1. *Access and motivation*: Explore the virtual learning environment (VLE) and the course site.
2. *Online socialization*: Get into groups and share tips and advice on how to work collaboratively online.
3. *Information exchange*: Work together in groups to compile an annotated bibliography on one of four topic areas.
4. *Knowledge construction*: Work together in groups to identify an online course or materials of common interest and review them from the perspective of an educational theory or theories.
5. *Development*: Work on your individual assignment, searching the literature and producing an outline plan, and a draft.

(Provided by Sandra Windeatt, University of Northumbria at Newcastle)

Digitaler Ansatz	Beispiele für enthaltene digitale Lehrelemente (Tools)
Mobile Technologien, i.d.R. für den interaktiven Gebrauch in Präsenzveranstaltungen	Tablets z.B. mit fachspezifischer Software, Smartphones, grafikfähige Taschenrechner
Online-Kursprogramme	MOOCs (Massive Open Online Courses), geschlossene und zielgruppenorientierte Kursangebote wie Virtual Universities
Learning Objects (LOs) und Digital Libraries, also Repositorien, in denen selbst entwickelte oder verbreitete LOs zusammengefasst und gespeichert sind	<p>LOs:</p> <ul style="list-style-type: none"> <li>• Online-lectures, Lehrvideos, VideoDemo/Audio-Folien-Präsentation (jeweils selbst entwickelt oder von Video-Portalen wie Youtube usw.), Konferenzvorträge,</li> <li>• Apps zur Nutzung bei Aufgabenbearbeitungen, für Games, Voting, Tests, Umfragen, Social-Media-Portfolios, Bar codes usw.</li> <li>• Literatur-Dateien, z.B. als pdf, etwa über URLs bereit gestellt</li> <li>• Software mit Aufgaben und Übungen einschliesslich Simulationen (z.B. online oder auf Tablets angeboten)</li> </ul>
Technologische Umgebungen für kollaboratives Lernen	<p>LMS – Learning Management Systems wie z.B. Moodle, mit Komponenten wie</p> <ul style="list-style-type: none"> <li>• Digital libraries</li> <li>• Angeboten für kollaborative Arbeit wie Foren, Studierendenordner, E-Mails usw.</li> </ul>
Digitalisierte Lehrkonzepte	<p>z.B. E-Lectures (also digital angebotene Lehre, insbesondere Lehrvideos einschliesslich interaktiver digitaler Elemente) und Blended-Learning-Format, z.B. als Kombination aus</p> <ul style="list-style-type: none"> <li>• E-Learning-Angeboten</li> <li>• Präsenzveranstaltungen</li> <li>• Forums-/E-Mail-Austausch</li> <li>• Webinaren, d.h. online angebotenen Seminaren, die Präsentations- und Interaktionsformate einschliessen wie Video, Audio, Gruppenraum, Chat, Whiteboard, Votes, Befragungen usw.</li> </ul>

Tab. 1.: 5 Ansätze für die digitalisierte Hochschullehre.



# Lehr-Lern-Methoden & - Strategien



**Prinzip der Methodenkompetenz**

## **Klassische Methoden:**

Unterrichtsstunden mit

- o Frontalunterricht/Präsentationen
- o Fragend-entwickelnde Methode

mit stark begrenzter Eigenständigkeit:

- o Einzelarbeit
- o Partnerarbeit
- o Gruppenarbeit

## **Handlungsorientierte Methoden/ eher »große Methoden«:**

- o Anchored Instruction
- o Biografiearbeit
- o Briefmethode
- o Cognitive Apprenticeship
- o E-Learning
- o Erkundung
- o Experiment
- o Fallstudien
- o Freiarbeit
- o Gruppen-Experten-Rallye
- o Gruppen-Wettkampf-Rallye
- o Kooperatives Lernen
- o Leittexte
- o Metaleernen (nach Reich)
- o Moderation / Metaplan\*
- o Offener Unterricht
- o Planspiel
- o Portfolio
- o Problem Based Learning
- o Projektarbeit
- o Rollenspiele
- o Referate
- o Situiertes Lernen
- o Storyline (Methode Glasgow)
- o Stationenlernen

- o weitere

und weitere Methoden oft als Mischformen mit anderen

## **Eher Techniken/ »kleine Methoden«**

- o Arbeitsateliers
  - o Blitzlicht
  - o Brainstorming
  - o Clustering
  - o Concept Learning
  - o Einstiege/Ausstiege
  - o Erzählung
  - o Fantasiereise
  - o Gespräch
  - o Inklusive Kommunikation
  - o multilinguale Gruppen
  - o Konstruktives Wissensspiel
  - o Korrespondenz
  - o Memory
  - o Metakognition
  - o Metaplan
  - o Mnemotechniken
  - o Mindmapping
  - o Open Space
  - o Organizer
  - o Placemat-Methode
  - o Postkorbmethode
  - o Quiz und Rätsel
  - o Subjektives Kartografieren
  - o Szenario-Methode
  - o Szenische Interpretation
  - o Sozialstudie
  - o Tagebuchmethode
  - o Techniken (einige ausgewählte)
  - o Wandzeitung
  - o Web-Tools
  - o Wochenplan
- o weitere

## **Werkstattarbeit:**

- o Computerwerkstatt
- o Werkstattunterricht (Reichen)
- o Zukunftswerkstatt

und weitere

Unterricht gestalten

## **Demokratie im Kleinen:**

- o Communities of Practice
- o Kinderparlament
- o Klassenrat
- o Demokratie im Kleinen

und weitere

## **Lernarrangements:**

- o Auftragsmethode
- o Juniorfirma
- o Lerninseln
- o Nachhaltige Schülerfirmen

und weitere

## **Öffentlichkeitsarbeit:**

- o Aufführungen
- o Ausstellungen
- o Internetpräsentation
- o Klassen-/Schulzeitung

und weitere

- o Feedback
- o Psychodrama
- o Reflecting teams
- o Reframing
- o Skulpturen
- o Systemaufstellungen
- o Szenisches Spiel
- o Teamentaching
- o Zirkuläres Fragen

und weitere

## **Außenkontakte/ Erlebnisse:**

- o Erlebnispädagogik
- o Feste und Feiern
- o Wandertag

und weitere

## **Systemische Benotung**

## **Supervision**

## **Evaluation**



**Prinzip der Methodenvielfalt**



**Prinzip der Methodeninterdependenz**

# Entdeckendes Lernen: Mustererkennung

- Tutorials
- Vorzeigen
- (interaktive) Videos ca. 5-7 min.
- Schritt-für-Schrittaufgaben
- Worked Examples
- Musterbeispiele & -lösungen
- Aufgabe + Lösung
- Begleitende Fragen und/oder Hinweise

```
public class sum {  
    public static void main (String[] args) {  
        int a = 5, b = 3;  
        int c = a + b;  
        System.out.println("a = " + a);  
        System.out.println("b = " + b);  
        System.out.println("c = " + c);  
    }  
}
```

# Mustererkennung

```
public class sum {  
    public static void main (String[] args) {  
        int a = 5, b = 3;  
        int c = a + b;  
        System.out.println("a    = " + a);  
        System.out.println("b    = " + b);  
        System.out.println("c    = " + c);  
    }  
}
```

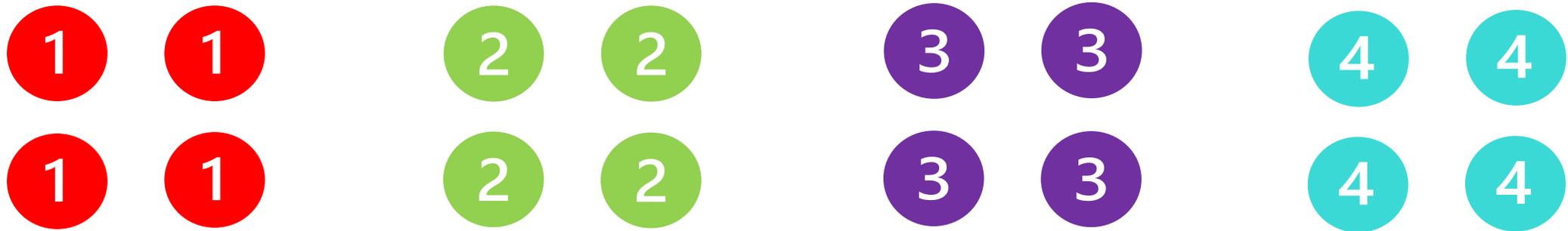
```
public class mult {  
    public static void main (String[] args) {  
        int a = 5, b = 3;  
        int c = a * b;  
        System.out.println("a    = " + a);  
        System.out.println("b    = " + b);  
        System.out.println("c    = " + c);  
    }  
}
```

# Kooperatives Lernen: Lernen durch Lehren

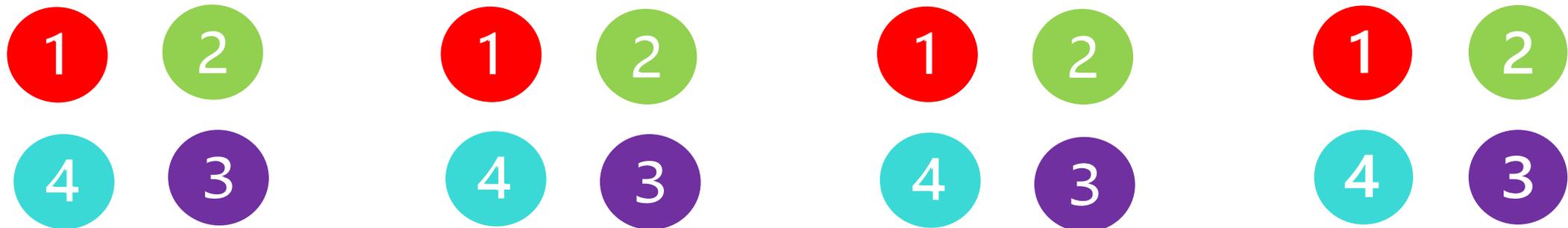
- Lernen durch Lehren
- Peer Learning
- Peer Teaching
- Talente-Tausch
- Gruppenpuzzle
- Buddy-System
- Questioning
- Pair Programming
- Project-based Learning
- Teamarbeit / Gruppenarbeit
- ...
- Abruf aus dem Gedächtnis
  - Neustart des Gedächtnisprozesses
  - Vertiefung (Brand & Markowitsch, 2009)
- Peer Tutoring (Bowman-Perrott et al., 2013)
  - Metastudie über 26 Einzelstudien
  - 980 Lernende von 1. – 12. Schulstufe
  - Mittlere bis große Effektstärke

# Gruppenpuzzle

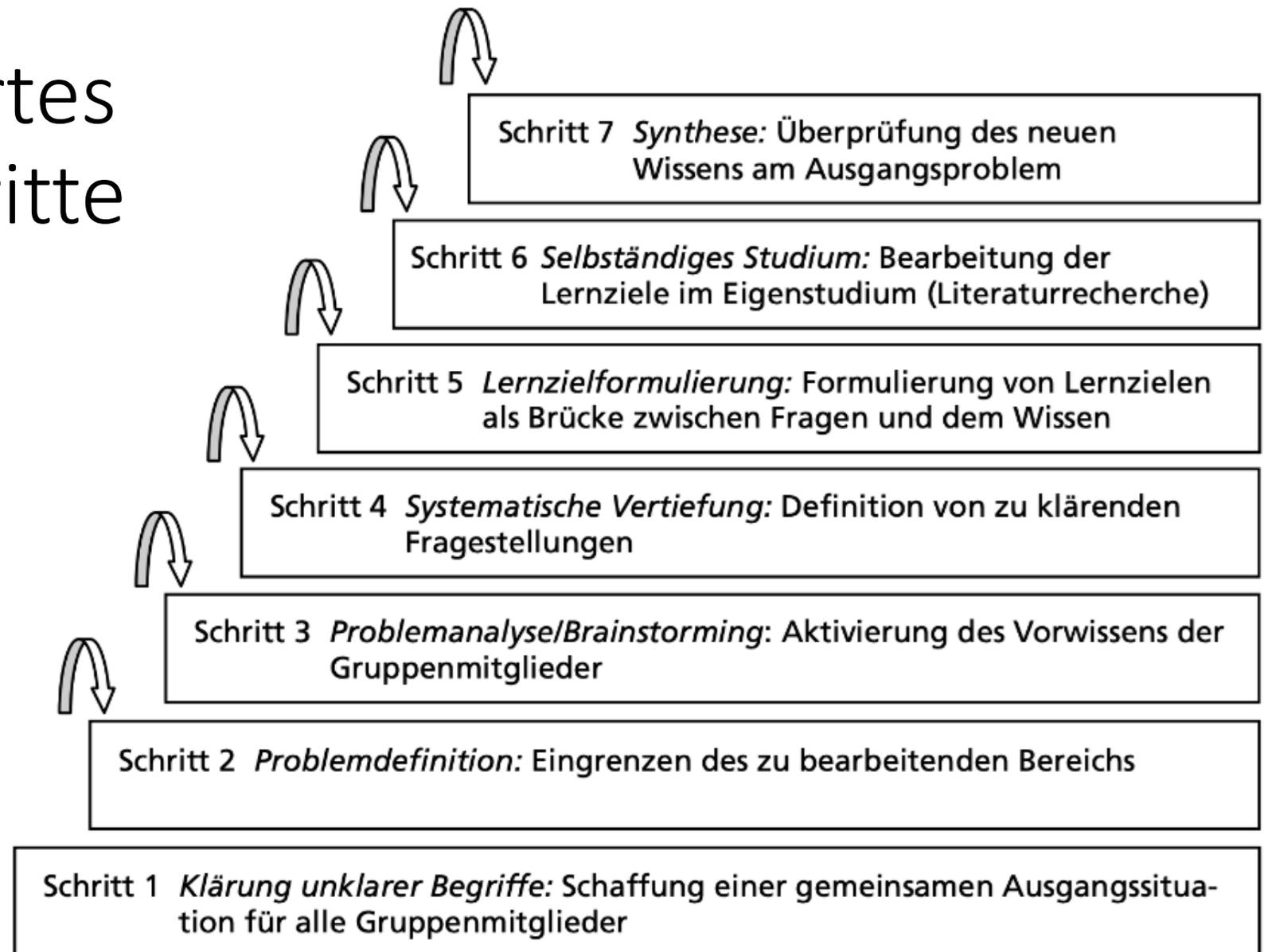
1. Runde: Expertengruppen – Wissen erarbeiten



2. Runde: Gemischte Gruppen – Wissen austauschen

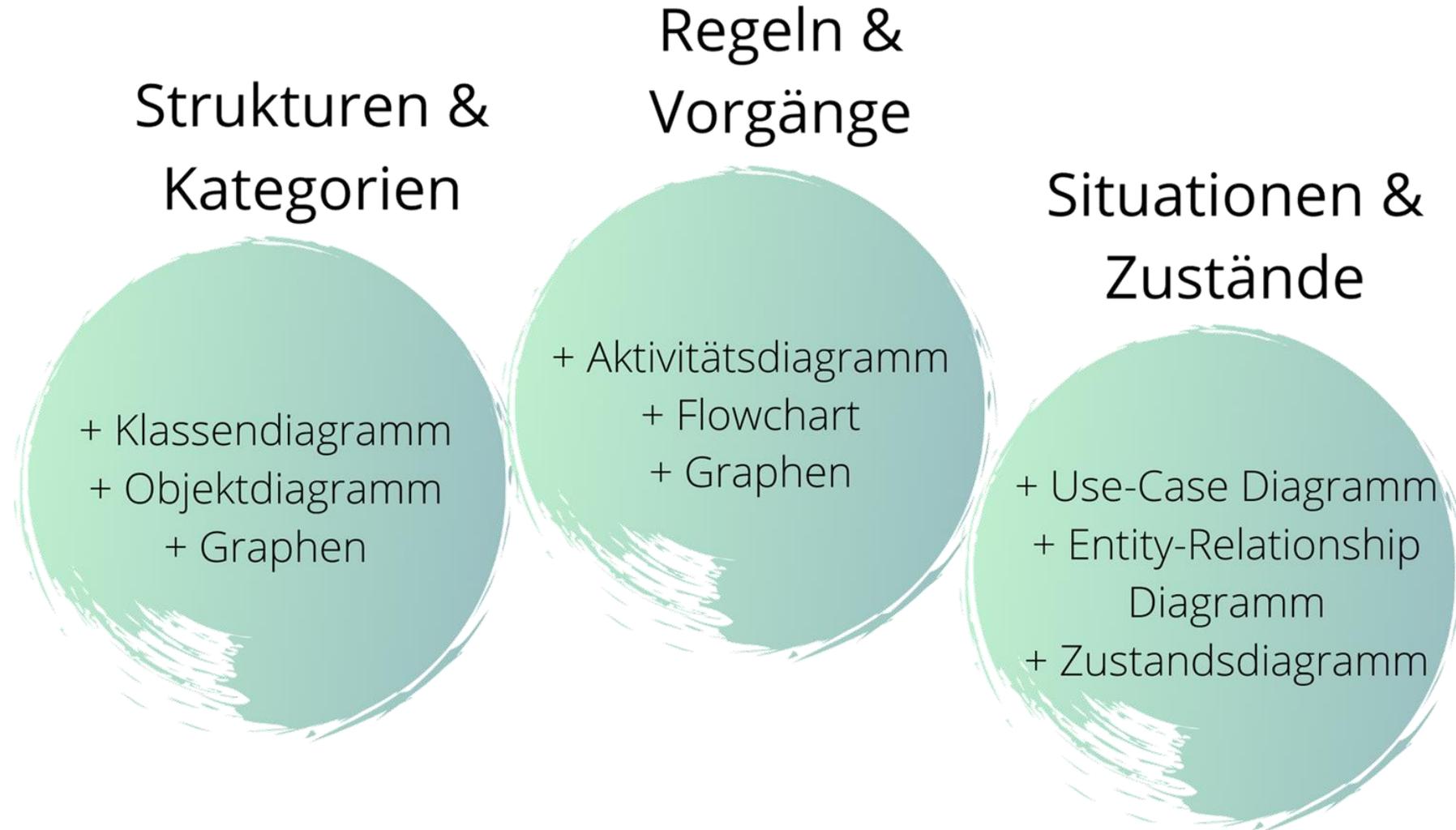


# Problembasiertes Lernen: 7 Schritte



Schmidt, H. G. (1983). Problem-based learning: rationale and description. *Medical Education*, 17(1), 11–16.

# Modellierung als Lernstrategie



01



# KATEGORIEN MIT OBJEKT- & KLASSENDIAGRAMMEN ERSTELLEN

Musterlösung

02



# SCHRITT FÜR SCHRITT ZUR LÖSUNG MIT AKTIVITÄTSDIAGRAMMEN

Beispiel

abstrakt



Klasse

konkret



Objekt

Virusfamilie

+ Aussehen  
+ Kategorie

+ verursacht Krankheitssymptome

Coronavirus (CoV)

+ Krone (Corona)  
+ SARS-Virus (severe acute respiratory syndrome)

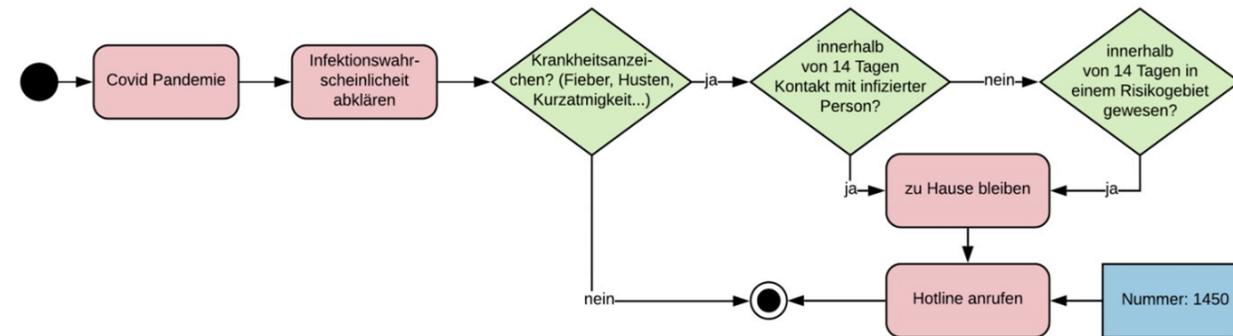
+ verursacht Husten, Fieber, Lungenentzündung, Übelkeit, Durchfall, Gliederschmerzen, Schüttelfrost, Lungenschmerzen...

Klassen-/Objektname

Attribut (Eigenschaften, Charakteristika)

Methode (was bewirkt es?)

Was tue ich, wenn ich vielleicht infiziert bin?



Corona-Unterrichtspaket für LehrerInnen

# Think aloud Protokolle

- Besonders geeignet für
  - Problemlöseprozesse
  - Regeln und Workflows
  - Learning by teaching
- Varianten & Anwendung von Think aloud Protokollen
  - Lehrmaterial (z.B. also Audio im Video, Musterlösungen)
  - Dialog / Frage-Antwort
  - Problemlösestrategie
  - Lern- & Merkstrategie
  - Aufdecken von Denkfehlern

# Aus der Praxis

## Konzepte & Beispiele

### The Concept of COOL Informatics



#### 1. Discovery

**Teaching and learning methods:**  
Solution-based learning  
Step-by-step instructions & tasks  
Video tutorials  
Observational learning  
Learning with all senses

**Neurodidactical base:**  
Pattern recognition  
Mirror neurons  
Individual learning rhythm  
Modality/multimedia effect



#### 2. Individuality

**Teaching and learning methods:**  
Competence-based learning  
Questioning  
Self-organized learning with compulsory and optional tasks

**Neurodidactical base:**  
Connecting new information to previous knowledge.  
Considering individual interests, needs, tasks, methods, learning rhythm



#### 3. Cooperation

**Teaching and learning methods:**  
Team and group work  
Peer tutoring and teaching  
Pair programming  
Cross-curricular learning  
Project-based learning

**Neurodidactical base:**  
"A joy (= knowledge) shared is a joy (= knowledge) doubled."  
Recall = re-storage in memory  
Integrating individual needs, talents, competences, practical relevance



#### 4. Activity

**Teaching and learning methods:**  
Hands-on, mind-on  
Learning by doing  
Learning by animation, simulation by playing and designing games (creative learning)

**Neurodidactical base:**  
Knowledge must be created (constructed) by each learner (= constructivism)  
Learning is an active process (= progressive education, e.g. Montessori)

# Brain-based Programming

## Lesson Design für Übungen

- Gruppen nach Kompetenzniveau
  - Profis = Peer-Tutoren, Peer-Teacher
  - Amateure = teilweise Peer-Tutoren, Buddies
  - Einsteiger
- LVA-Phasen (nicht unbedingt in dieser Reihenfolge)
  - Fragerunde (ca. 10 min): Vorwissen berücksichtigen, Abruf = Neueinspeicherung
  - Entdecken (10-20 min): Mustererkennung, Lernrhythmus (Konsolidierung)
  - Labor (Pair-Programming): Practice makes perfect, Abruf = Neueinspeicherung
  - (Vortrag/Erklärung/Erklärvideo: bei Bedarf max. 20 min)

# Brain-based Programming Task Design

- Entdeckendes Lernen - Mustererkennung
  - Leseecken (Programmcode, Puzzle, Lückentext, Fehlersuche)
  - Schritt-für-Schritt-Anleitungen mit Lösungen
  - Aufgabenstellungen mit Musterlösung
- Kompetenzorientiertes Üben – Vorwissen
  - Miniübungen
- Programmieren – Selbständiges Üben
  - Kleine, vollständige Programme
  - Teile bzw. Unterprogramme für ein Semesterthema

# The Concept of COOL Informatics



## 1. Discovery

### Teaching and learning methods:

Solution-based learning  
Step-by-step instructions & tasks  
Video tutorials  
Observational learning  
Learning with all senses

### Neurodidactical base:

Pattern recognition  
Mirror neurons  
Individual learning rhythm  
Modality/multimedia effect



## 2. Individuality

### Teaching and learning methods:

Competence-based learning  
Questioning  
Self-organized learning with compulsory and optional tasks

### Neurodidactical base:

Connecting new information to previous knowledge.  
Considering individual interests, needs, tasks, methods, learning rhythm



## 3. Cooperation

### Teaching and learning methods:

Team and group work  
Peer tutoring and teaching  
Pair programming  
Cross-curricular learning  
Project-based learning

### Neurodidactical base:

"A joy (= knowledge) shared is a joy (= knowledge) doubled."  
Recall = re-storage in memory  
Integrating individual needs, talents, competences,  
Practical relevance



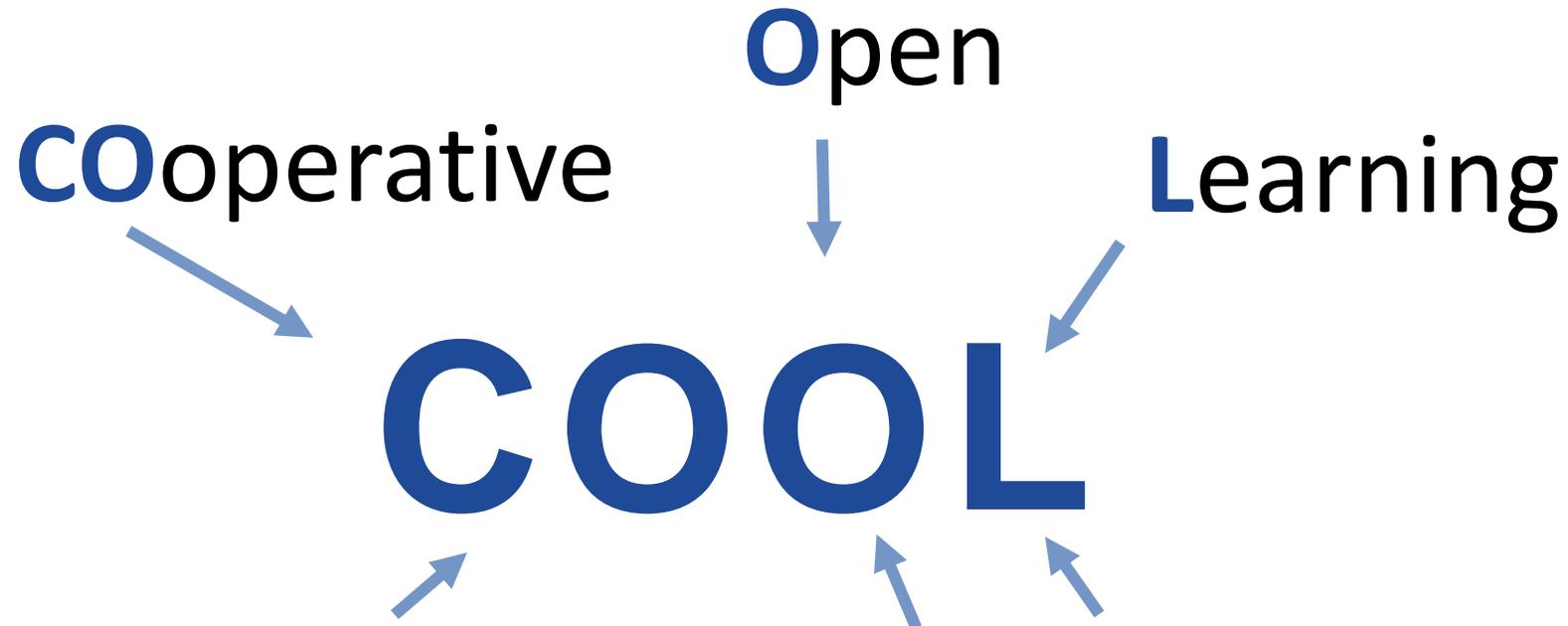
## 4. Activity

### Teaching and learning methods:

Hands-on, mind-on  
Learning by doing  
Learning by animation, simulation by playing and designing games (creative learning)

### Neurodidactical base:

Knowledge must be created (constructed) by each learner (= constructivism)  
Learning is an active process (= progressive education, e.g. Montessori)



**CO**mputer-supported **O**pen **L**earning

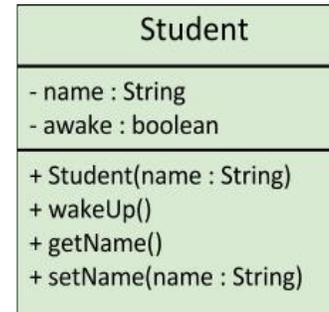
**CO**mputer-Science-supported

**C**ross-curricular **O**nline **O**pen **L**earning

**„cool“** motivierend, spielerisch, kreativ

# COOL: *Discovery* Lesecke

```
public class Student {  
  
    private String name;  
    private boolean awake;  
  
    public Student (String name) {  
        this.name = name;  
        this.awake = false;  
    }  
  
    public String getName() {  
        return this.name;  
    }  
  
    public void setName (String name) {  
        this.name = name;  
    }  
  
    public void wakeUp() {  
        this.awake = true;  
    }  
}
```



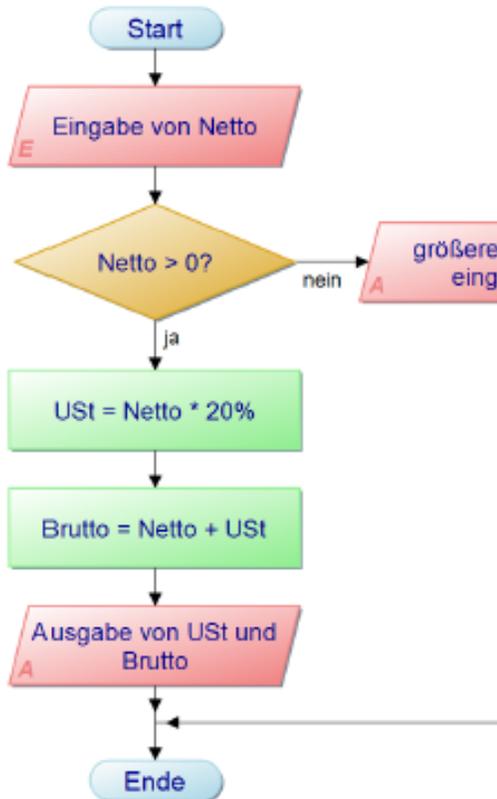
```
public class Studententag {  
  
    public static void main (String[] args) {  
        Student object 1 = new Student ("Jami e");  
        Student andy = new Student ("Andreas");  
        Student randomName = new Student ("Naomi");  
  
        object 1.wakeUp();  
        andy.wakeUp();  
  
        randomName.setName ("Melanie");  
        System.out.println(randomName.getName());  
    }  
}
```

## 8. Fragen und Aufgaben zur Lesecke 1

1. Lesen Sie die obigen Beispiele für Klassen und schreiben Sie einen „Schwindelzettel“ mit den wesentlichen Informationen zu Klassen und Objekten.
2. Markieren Sie den Konstruktor in der Klasse `Student`. Welche Parameter bekommt er, welche Variablen setzt er? Welche Methoden hat die Klasse `Student`? Was leisten sie?
3. Wie viele Objekte werden in der `main`-Methode (Klasse `Studententag`) erstellt? Wie heißen die Objekte?
4. Was wird an den Objekten verändert? Was wird schließlich auf der Konsole ausgegeben?

## A 1 Lesecke 1: Darstellung von Algorithmen

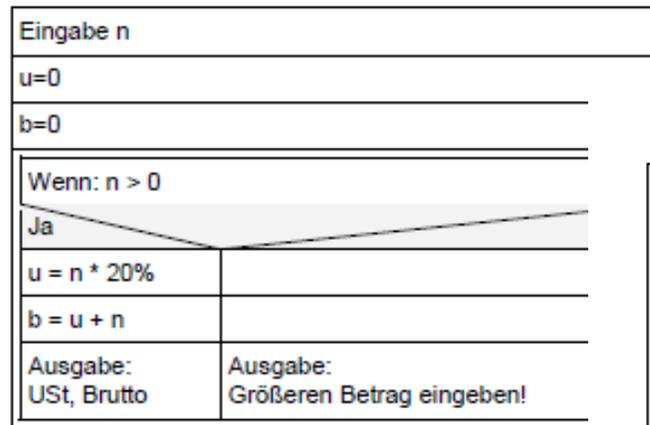
Bruttobetrag - Flussdiagramm



### Bruttobetrag – Algorithmus in natürlicher Sprache

- 1 Frage nach dem Nettobetrag.
- 2 Gib einen Nettobetrag ein.
- 3 Wenn der Nettobetrag größer 0 ist, berechne 20% Umsatzsteuer vom Nettobetrag und den Bruttobetrag (Summe von Netto und USt).
- 4 Gib die USt und den Bruttopreis am Bildschirm aus.
- 5 Wenn der Nettobetrag nicht größer 0 ist, frage nach einem größeren Betrag.

Bruttobetrag - Nassi-Shneiderman-Diagramm



Bruttobetrag – Java Code

```

public class Brutto2 {
    static Scanner in = new Scanner(System.in);

    public static void main(String[] args) {
        System.out.println("Bitte geben Sie den Nettobetrag ein: ");
        double n = in.nextDouble();
        double u=0, b=0;
        if (n <= 0)
            System.out.println("Geben Sie einen größeren Betrag ein!");
        else {
            u = n*20/100;
            b = n + u;
            System.out.println("Die USt beträgt: " + u);
            System.out.println("Der Bruttopreis beträgt: " + b);
        }
    }
}
  
```

```

Bitte geben Sie den Nettobetrag ein: 10
Die USt beträgt: 2.0
Der Bruttopreis beträgt: 12.0
  
```

```

Bitte geben Sie den Nettobetrag ein: 0
Geben Sie einen größeren Betrag ein!
  
```

# Lesecke

Die Lesecke zeigt anhand des ein-fachen Beispiels der Berechnung von Umsatzsteuer und Bruttobetrag den Weg von der Problemstellung zum Programm. Lesen Sie erst den Algorithmus in natürlicher Sprache und ordnen Sie die einzelnen Schritte den jeweiligen Entsprechungen im Fluss-diagramm, im Nassi-Shneidermann-Diagramm und im Programmcode zu (z.B. durch Nummerieren oder farbliches Hervorheben).

# 1.Schritt-für-Schritt: Eindimensionale Arrays

Das Programm soll überprüfen, ob in einem Supermarkt ein spezielles Lebensmittel noch lagernd ist. Dazu benötigt man zuerst ein Objekt der Klasse Scanner, um das Lebensmittel einlesen zu können:

```
public class Arrays {  
    static Scanner sc = new Scanner(System.in);
```

Als nächstes wird in der `main` – Methode ein `String[] supermarket` erstellt und mit einigen Lebensmitteln (Strings) befüllt. Außerdem erstellt man eine String Variable, in der man das eingelesene Lebensmittel speichert:

```
public static void main(String[] args) {  
    String[] supermarket = new String[] {"Brot", "Nudeln", "Milch", "Kaffee", "Zucker"};  
    String lm = "";
```

Nun soll das Lebensmittel eingegeben werden:

```
System.out.println("Bitte Lebensmittel eingeben: ");  
lm = sc.next();
```

Nun erstellt man eine Variable für die `for`-Schleife, damit man später noch darauf zugreifen kann:

# COOL: *Individuality*

- Einbringen und Berücksichtigen von individuellen
  - Interessen – Themenwahl
  - Talenten – Peer Teaching & Tutoring
  - Aufgaben und Problemstellungen – Lernende entwickeln selbst
  - Lernmethoden, Lernstilen & Lernrhythmus – Offener Unterricht
  - Rollen – Lernende, Lehrende, EntwicklerInnen
  - Alltagsbezogene Themen und Ziele
  - Gender & Diversität
- Wahlmöglichkeiten!

## COOL: *Cooperation*

---

- Pair Programming
- Peer Learning
- Peer Tutoring
- Learning by Teaching
- Buddy-Prinzip
- Project-based Learning
- Questioning
- Fächer-, Klassen-,  
Institutionenübergreifend



## COOL: *Activity*

- „coole“ Aktivitäten
- Wissen erarbeiten
- Entwickeln & Gestalten
- Handlungsorientierung
- Bewegung & Animation
- Learning by Doing
- Kreativität & Aktivität
- Modellierung u.a. Lernstrategien
- Spielen & Spiele entwickeln



# Logik → Hands-on mit Kärtchen & Magneten

Maxi's birthday is in carnival. He invites some of his friends for a birthday party. The children, who come to the party, wear a mask **OR** bring a birthday present. All of them who are wearing a mask **OR** bring a gift get a piece of the cake. The others only drink milk.

- How many pieces of cake and how many cups of chocolate are needed?
- How many pieces are needed when only children who wear a mask **AND** bring a gift eat a cake.

✓ = yes = 1  
X = no = 0



# Boolesche Algebra

## True or false? Truth Tables: OR

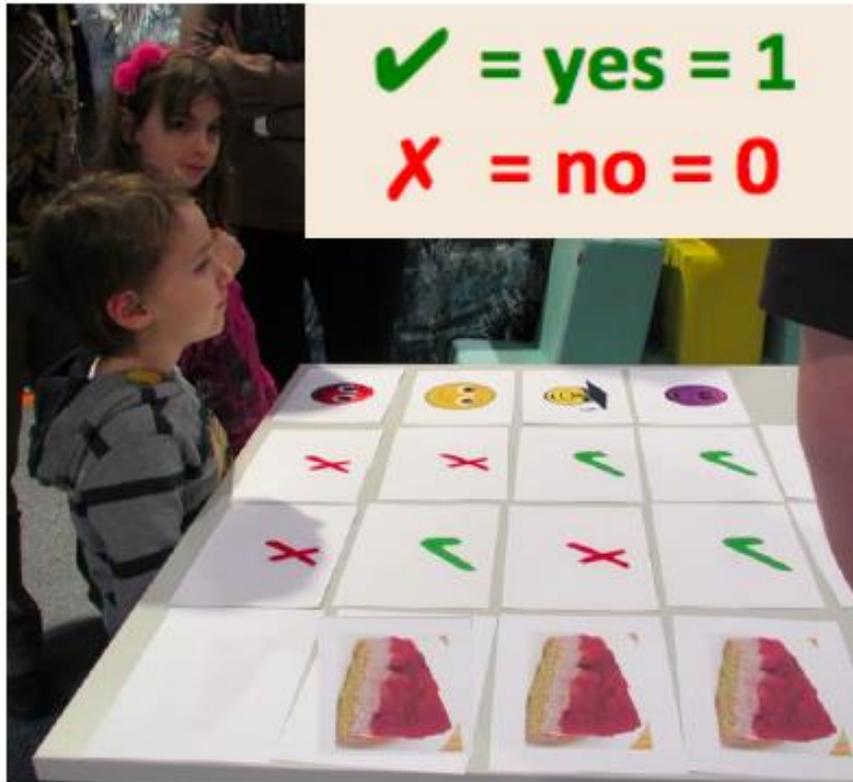


Fig. 1 Truth table for “mask OR gift”

	Mask	Gift	Mask OR Gift
	1	1	1
	1	0	1
	0	1	1
	0	0	0

Fig. 2 Truth table with binary numbers

# Boolesche Algebra & Schaltkreise: OR

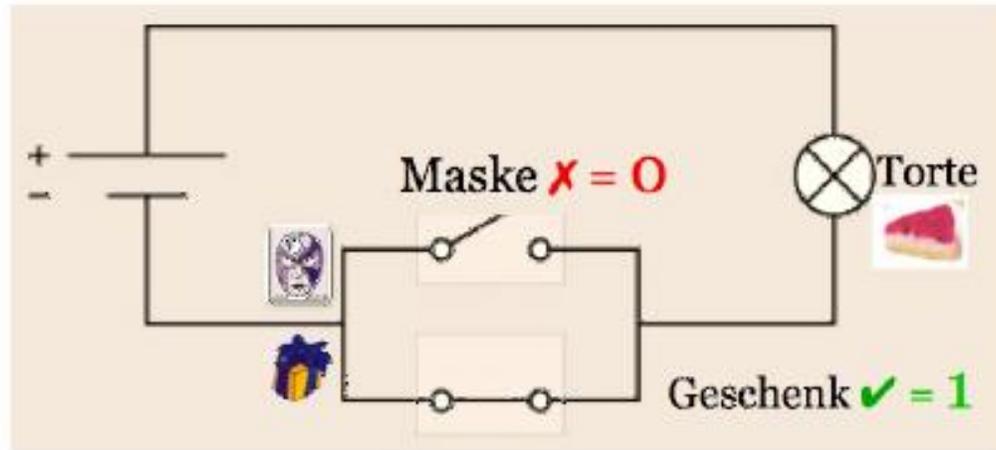


Fig. 3 Digital circuit based on truth table in Fig. 2

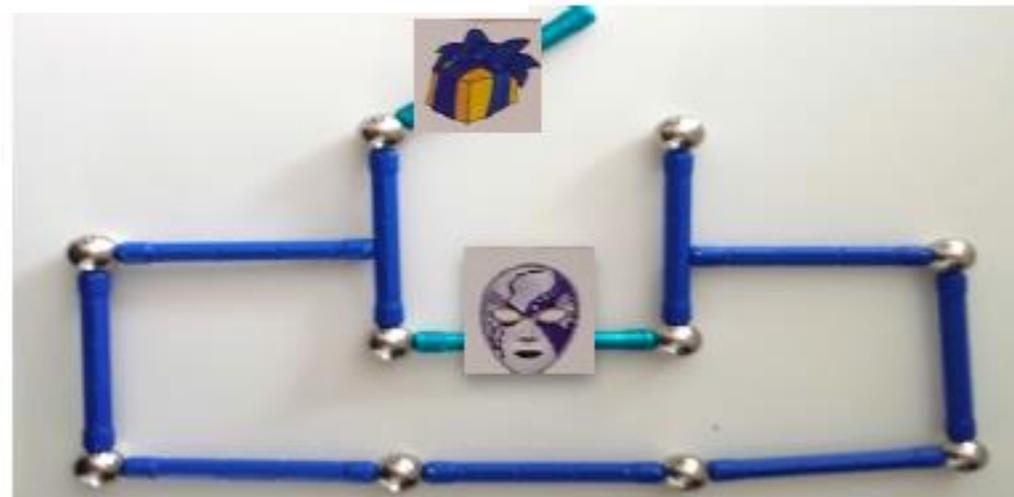


Fig. 4 Digital circuit with magnetic toys

# Beurteilung

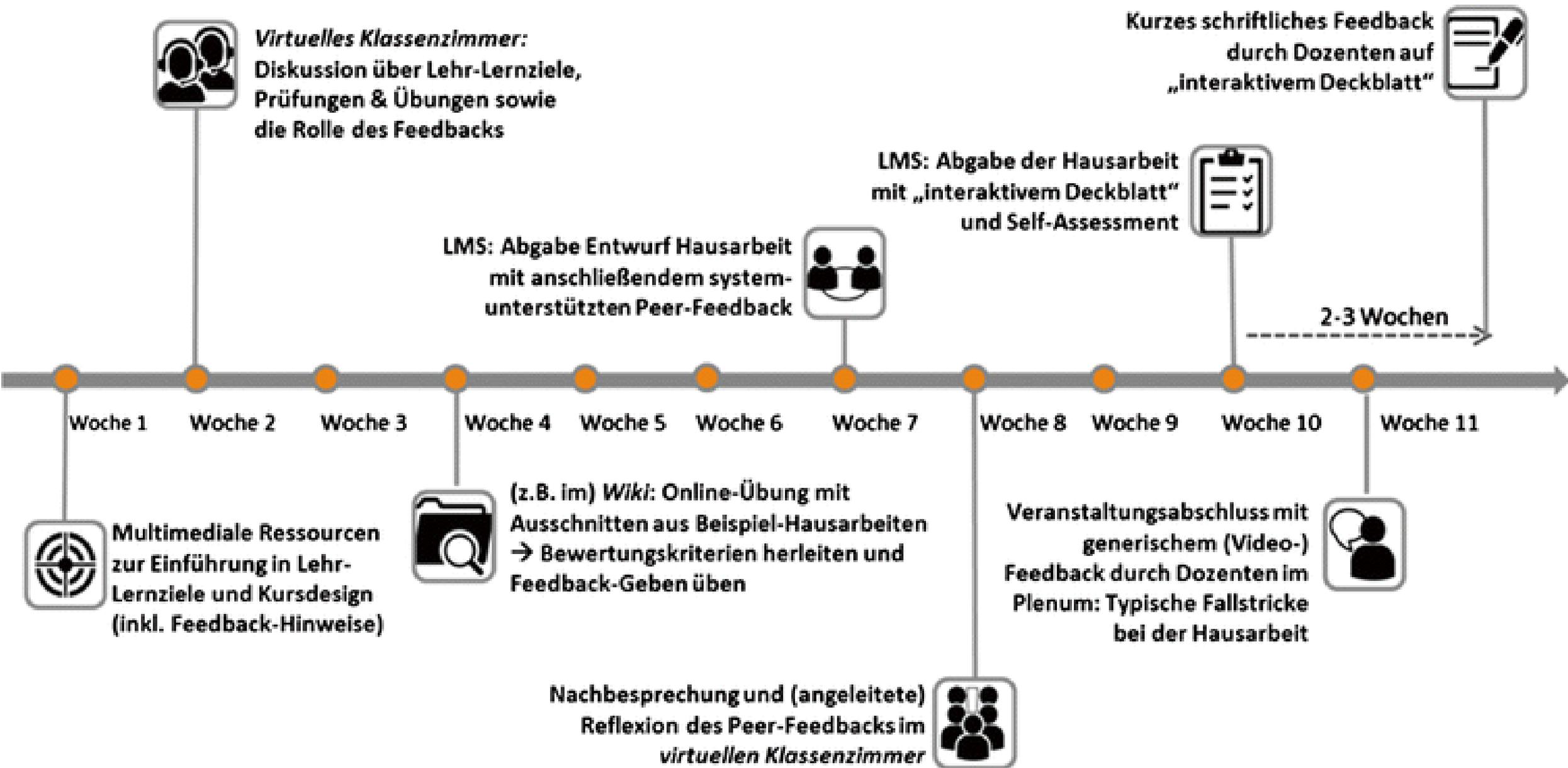


# Beurteilung

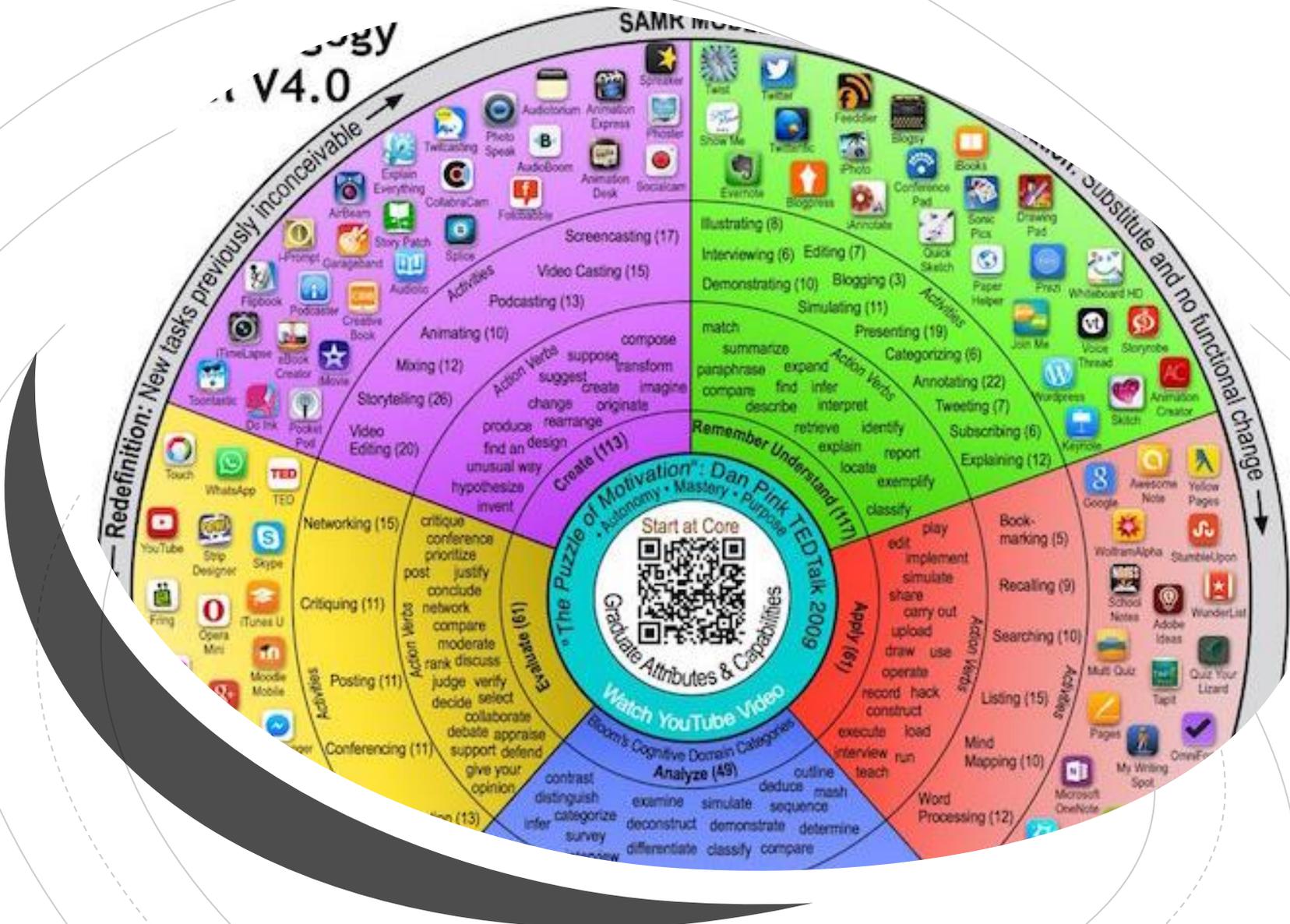
- Alternative Beurteilungsformen
  - Self-assessment
  - Peer assessment
  - Co-assessment
- Wie
  - Punktesystem & Badges
  - Pflicht- und Wahlaufgaben
  - Projekte / Gruppenarbeiten
- Tools
  - Moodle
  - Socrative
- Webquest
- Videoprüfung
- Hausarbeit
- E-Klausur
- Peer-Assessment
- Badges
- E-Portfolio
- Fallstudien
- Elektronische Aufgaben
- (e-teaching.org)

Lernziel	E-Assessment-Methode
Faktenwissen	E-Test (mit automatisch auswertbaren Aufgaben, meist Multiple Choice)
Begriffliche Zusammenhänge	Erstellen von Mindmaps
Historische Daten	<a href="#">Erstellen von Online-Zeitleisten, z.B. mit Xtimeline</a>
Selbstkontrolle des Lernfortschritts	Online-Quiz
Reflexion der Lernstrategie	<a href="#">E-Portfolio, Lerntagebuch</a>
Teamfähigkeiten	<a href="#">Online-Projektgruppen, Kollaboration per Wiki, Peer-Assessment</a>
Konzeptionelles Verständnis	<a href="#">Interaktive Module (Simulation, virtuelles Labor, Webquest)</a>
Problemlösefähigkeiten	Online-Rollenspiel, Fallstudien, problembasierte Szenarien
Kommunikation, Rhetorik	Diskussionsforen, Weblogs, Online-Vorträge
Kreativität, technische Kompetenz	Produktion von Podcasts, Videos, Skripts etc.

Tabelle 1: Formen des E-Assessment



Exemplarische Feedback-Strategie für ein Online-Seminar.  
 (adaptiert; verwendete Icons designed by Freepik) (Hartung, 2017)



# Tools & Plattformen



# Tools

## Videos erstellen

- Loom: <https://www.loom.com/>
- Renderforest:  
<https://www.renderforest.com/de/>
- Powtoon: <https://www.powtoon.com/>

## Zusammenarbeit & Kommunikation

- Miro  
<https://miro.com/online-whiteboard/>
- LiveBoard: <https://liveboard.online/>
- Trello Projektmanagement:  
<https://trello.com/>
- **Slack** Kommunikation: <https://slack.com/>

## Plattformen

- **GeoGebra**: <https://www.geogebra.org/>
- Moodle: <https://moodle.ph-ooe.at/course/view.php?id=1943>
- W3schools (Informatik): <https://www.w3schools.com/>

## Aufgaben & Tests erstellen (Interaktives)

- Textpuzzles online  
<https://www.classtools.net/dragdrop/>
- Learning Apps <https://learningapps.org/>
- H5P <https://h5p.org/>
- Kahoot <https://kahoot.com/>
- **Mentimeter** (Umfragen)  
<https://www.mentimeter.com/>
- Socrative (Tests) <https://www.socrative.com/>

# Textpuzzle

- [https://www.classtools.net/dragdrop/202009\\_DTUALL](https://www.classtools.net/dragdrop/202009_DTUALL)



New Edit Share

## Arrays1

Drag and Drop the items into the correct order. You can also [create your own](#) | [view a sample](#).

```
int[] durchschnitt = new int[10];
System.out.println();
int gefunden = 0;
for (int j = 0; j < 10; j++) {
for (int i = 0; i < 10; i++) {
}
System.out.print(durchschnitt[i] + " ");
public static void main(String[] args) {
for (int i = 0; i < 10; i++) {
int[] zahlen1 = { 12, 23, 45, 67, 55, 77, 89, 99, 43, 22 };
if(zahlen1[i] == zahlen2[j]) {
}
}
}
zahlen2[i] = sc.nextInt();
}
durchschnitt[gefunden++] = zahlen1[i];
System.out.print("Geben Sie Zahl " + (i + 1) + " von 10 ein: ");
static Scanner sc = new Scanner(System.in);
for(int i=0; i < gefunden; i++) {
public class Array1 {
```

# Links für Informatik

- Unterrichtsmaterialien Informatik: <https://www.swisseduc.ch/informatik/>
- Videochannel Simple Club Informatik : <https://www.youtube.com/c/TheSimpleInformatics/>
- Lernpfad Java: <https://unterrichten.zum.de/wiki/Java>
- Kollaboratives Programmieren: <https://repl.it/>