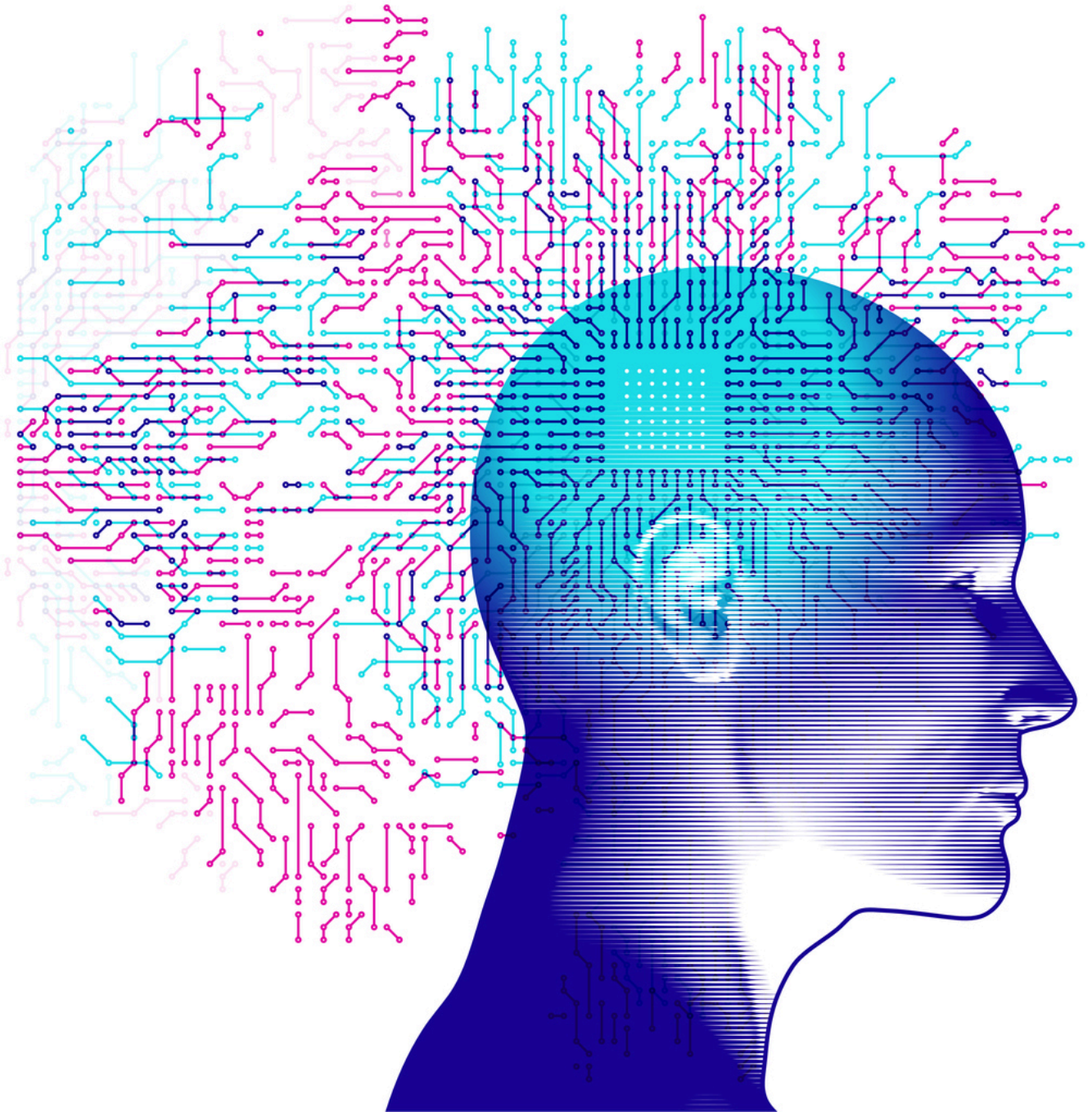


# Warum Interdisziplinarität die Grundlage für Computational Thinking ist

Autor : SocietyByte

Datum : 8. Juli 2019



**Das einzige Lebewesen, das denken kann, ist der Mensch. Doch bald übernehmen Computer das Denken für uns - oder doch nicht? Zu diesem Thema hat uns Alexander Repenning, Informatikprofessor der FHNW und Referent an unserer Transform-**

**Konferenz am 12. und 13. September, ein paar Fragen beantwortet.**

**Was verstehen Sie unter Computational Thinking?**



Prof. Dr. Alexander Repenning

Das Denken *mit* dem Computer. Also nicht denken *wie* ein Computer oder *über* Computer, sondern den Computer als Denkinstrument benutzen. Zum Beispiel mit Simulationen, die der Mensch programmiert hat, kann der Computer helfen, die Konsequenzen des menschlichen Denkens sichtbar zu machen.

**Wie wichtig ist Computational Thinking – heute und in Zukunft?**

Computational Thinking ist eine Kernkompetenz der 21st Century Workforce. Ohne Computational Thinking wird die Digitale Revolution die Enkel der Industriellen Revolution fressen. Das will heissen, dass die Hochspezialisierung, welche geprägt wurde in der Industriellen Revolution, und welche immer noch das Konzept von Fachgebieten in Schulen bestimmt, die heutige Gesellschaft extrem verletzlich macht bezüglich Wegrationalisierung durch Roboter oder Künstliche Intelligenz. Ein wichtiger Aspekt von Computational Thinking ist die Unterstützung von Interdisziplinarität. Leute müssen die Zusammenhänge der Disziplinen verstehen und nutzen können.

**In welchem Alter sollte der Unterricht in Computational Thinking beginnen?**

Wenn Kinder selber interessiert sind, können sie schon in der 1. Klasse, oder noch früher, anfangen einfache Programme zu schreiben. Aber gleichzeitig muss man aufpassen, wenn man Kinder schon zu früh zwingt zum Programmieren. Gut gemeinte "motivierende" Aktivitäten wie das Programmieren von Robotern, die irgendwelchen Linien nachfahren sollen, kippen dann verblüffend schnell mal in "are we there yet?" Situationen. Die Computational-Thinking-Konzepte, die man beispielsweise Viertklässlern beibringen möchte, um Spiele und Simulationen zu programmieren, benötigen keine Vorbereitung im Kindergarten oder bis zur 3. Klasse. Hier ist weniger mehr.

### **Wie erlernt man am einfachsten Computational Thinking? Wie unterrichtet man es am besten?**

Es braucht eine Kombination von pädagogischen Konzepten, die nicht nur IKEA-artig Projekte Step-by-Step nachbauen und Computational-Thinking-Tools, die den Programmierprozess sowie den Kreativitätsprozess unterstützen. Lehrpersonen sowie Schülerinnen und Schüler sollen Computational Thinking als iterativen Prozess der ständig Abstraktion, Automation und Analyse wiederholen, um Lösungen zu finden. Ganz wichtig ist es, projektorientiert vorzugehen, aber gleichzeitig nötige Abstraktionen, wie zum Beispiel Computational-Thinking-Patterns zu thematisieren.

### **Kann man als IT-ferner Mensch die Fähigkeit zum Computational Thinking im fortgeschrittenen Alter noch erlernen, beziehungsweise verbessern?**

Ja, auf jeden Fall. Wichtig ist aber zuerst ein Interessengebiet zu finden und dann Computational Thinking als Ansatz dieses Gebiet besser zu verstehen. Danach ist es wichtig, geeignete Computational-Thinking-Tools zu finden.

### **Gibt es einen Unterschied zwischen Buben und Mädchen, was das Talent zum Computational Thinking betrifft? Was sind ihre praktischen Erfahrungen?**

Wir fokussieren uns hauptsächlich auf Game Design. Aus der Kompetenzperspektive und Motivationsperspektive gibt es kaum Unterschiede. Allerdings wenn man ein fertiges Spiel anschaut kann man oft erraten ob es von Buben oder Mädchen kreiert wurde.

---

## **Zur Person**

[Alexander Repenning](#) ist Professor für Informatikausbildung an der FHNW und Informatik-Professor an der University of Colorado. Er leitet die internationale Scalable Game Design Initiative. Repenning ist ein Pionier der blockbasierten Programmierung (alias Drag and Drop-Programmierung). Er arbeitete in der Forschung und Entwicklung bei Asea Brown Boveri, Xerox PARC, Apple Computer und Hewlett Packard. Repenning ist der Schöpfer der AgentSheets und AgentCubes Simulations- und Spielinformations-Tools.

Alexander Repenning referiert an der Transform-Konferenz am 13. September zum Thema "Computational Thinking for All".

---

## Transform-Konferenz

Die Konferenz "TRANSFORM - Digital Skills for the Transformation of Disciplines, Business, and Government" wird vom [Institut Public Sector Transformation](#) und [Institute Digital Enabling](#) in Zusammenarbeit mit nationalen und internationalen Partnern als Forum zur Diskussion von Herausforderungen rund um die digitale Transformation mitorganisiert. Die [Schweizerische Informatikgesellschaft](#) ist Veranstaltungspartner.

Das Programm der Konferenz und das Anmeldeformular finden Sie [hier](#).