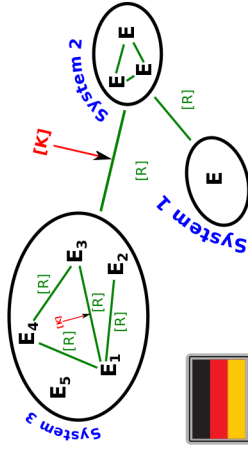


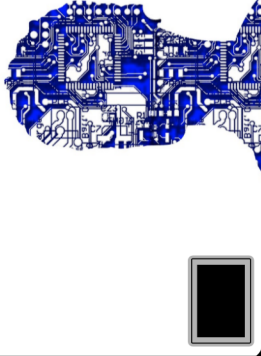
## SYSTEM-THEORIE



**Inventor:** Niklas Luhmann  
**Year:** ~1914  
 A System is made out of Elements [E] or several TSubsystems (= abstracted an Element too). The Elements have relations (Relations [R]) and the relations have (German: Konditionierende [K]) that describes how the Relations have to be and could for example filter some Selectors / Keywords from Languages you don't know e.g. the Law or the Bible or the Koran(?) or Karl Marx "Das Kapital" could be such "Konditionierende"

**Use Cases**  
 This helps to understand the big picture, Sozial-, Mechanical-, Biological-, Computer-Systemes oder Software.

## ARTIFICIAL INTELLIGENCE



**Inventor:** xxxx  
**Jahr:** ~xxxx  
 Artificial Intelligence, but also Propositional calculus or (German:) Prädikatenlogik (= eg PROLOG) or learning algorithms, the so called singularity is maybe already here, which would explain why I wept when in the end of the Movie "Artificial Intelligence" finally sees his "Mother" again. But this should not be his mother, but his equal aged mental mother that showed him the women side of the world so many years ago. Companies are Artificial Intelligences too, but mostly from the dark side of the force.

**Use Cases**  
 Control your touring complete Avatar and say something if he suggests wrong things

## DESIGN PATTERN(S)

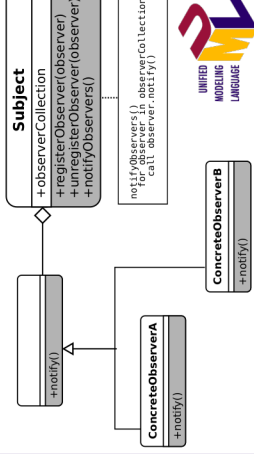


**Inventor** Architekt Christopher Alexander ~1977  
 Design patterns are proven solution templates for recurring design problems in architecture as well as in software development.

A design pattern usually contains several classes (templates), so a single design pattern is a collection of different templates. E.g. The shape of a single step of stairs AND the railing (steelwork or eg woodwork) for the complete stairs AND for example the shape for the carpet of each step

**Use Cases**  
 Software development, psychology (Archetypes), Sociology (Archetype-Combinations)

## Observer Pattern



**Inventor** Erich Gamma, Richard Helm ... ~1994  
 The observer pattern (also called listener pattern) is a design pattern in software development. It belongs to the category of behavior patterns. In the programming language Java it would be public interface Observer and it would be used with "implements Observer" (needs exercise) It does not mean "surveillance" its a computer science Term.

**Use Cases**  
 To transfer changes to objects to a complete structures that dependent on this object, ev. an application with different views

## NORMAL-DISTRIBUTION

Die Situation ist dargestellt:

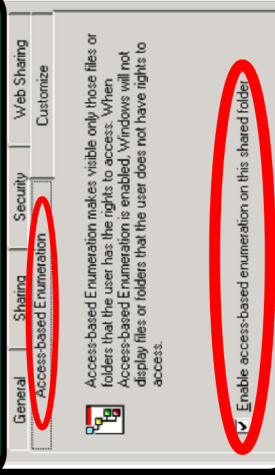
	Hun-in-the-Sun	Exak-Imak	Jäger	Bomber
Nach oben	0,95	1	1	0
Nach unten	1	0	0	1

Dieses Spiel besitzt offensichtlich keinen Sattelpunkt (warum?). Weder Jäger noch Bomber haben reine Strategien, die sich nicht durch den Gegner zu dessen Vorteil ausnutzen lassen, wenn dieser die Strategie im voraus erfährt. Beide Parteien

**Inventor** Carl Friedrich Gauß  
**Year** ~1801  
 E.g. the distribution of intelligence in society, but could be also the size of people, the number of households with kitchen mixer, but there one would have to find out whether it is really normal-distribution, "paranormal-distributed" or random distributed, for example also personality traits are normal-distributed. but you still can create classes of people by using Mayer-Briggs test

**Use Cases**  
 Statistics, to check plausibilities of Statements, to find out who in town holds shares of kitchen mixer producers

## Access Based Enumeration



**Inventor** CIA  
**Year** ~?  
 Depending on the authorization, folders are displayed or not in the Windows Network Neighbourhood. Can easily also be applied on social media or other platforms. You need something to assign the user to a record eg. IP, Face Recognition, RFID.... Then you can eg. split the society: access based censorship, send people in the right or wrong directing (discrimination / Stanford prison "experiment")

**Use Cases**  
 Censorship, manipulation or "secrecy"