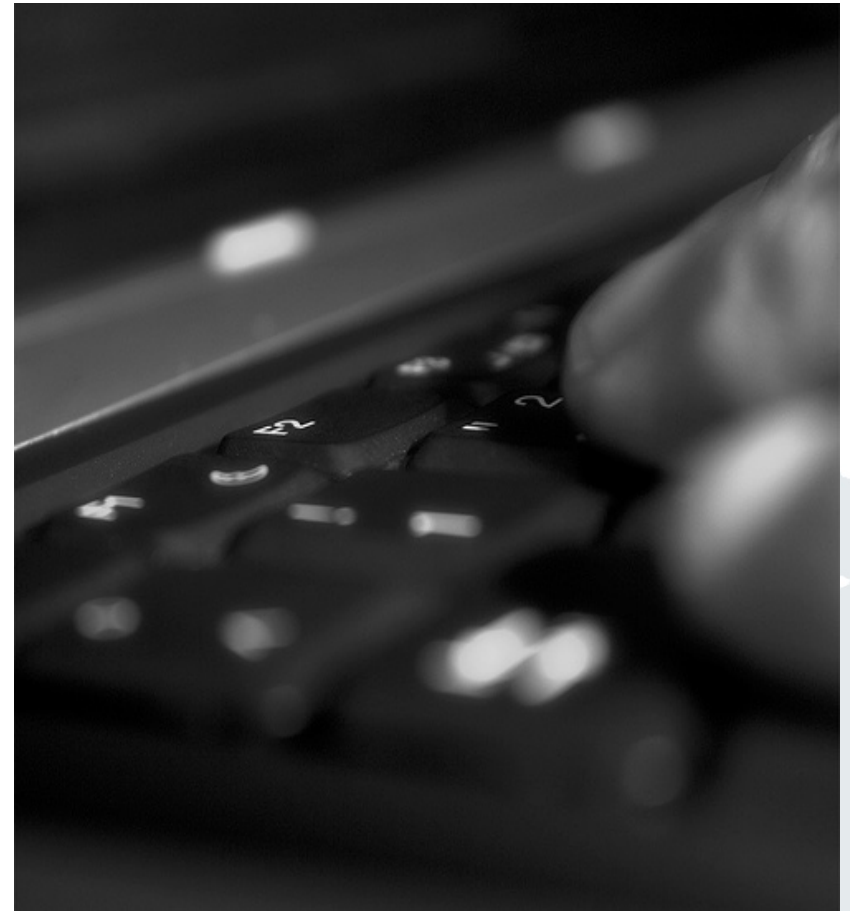


Q&A Session  
Business Informatics 2 (PWIN)

Prof. Dr. Kai Rannenberg  
[www.m-chair.de](http://www.m-chair.de)

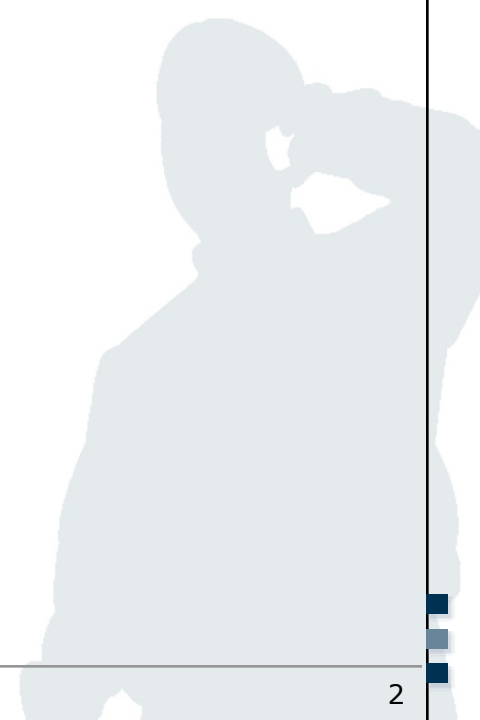


Jenser (Flickr.com)

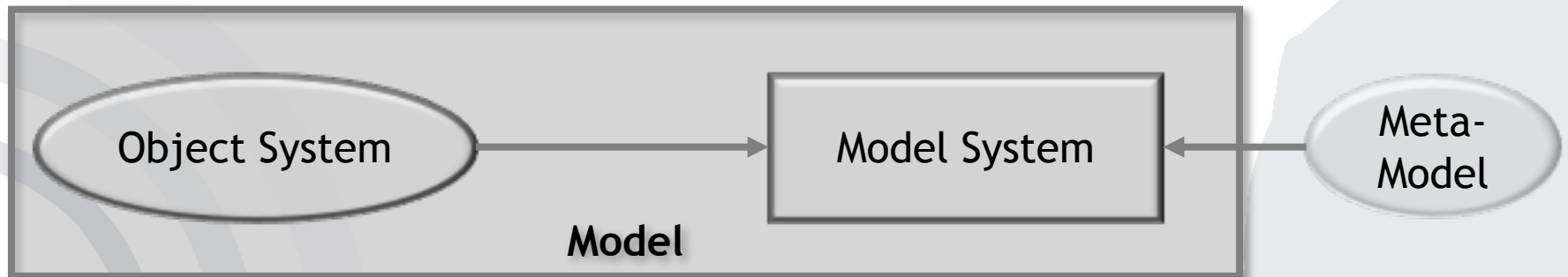
## Lecture 2 Information Systems I (Slide 33)

### Frage:

- Was ist ein Beispiel für ein Meta Model und wie genau definiert es die Sprache für das Modell?



- **Consistent correspondence of structure and behaviour** between model and original required
- Consequently, the modelling function is to be a **homomorph** transformation
- The “language” for the definition of a model system is provided by a meta-model.



# What is a Model?

- A model is a representation of a the real world with the following properties
  - **Representation:** A model is always representation of natural or artificial objects, which themselves can be models.
  - **Abstraction:** Models are typically an excerpt of reality.
  - **Pragmatism:** The contents of a model are relativised through the following questions: For whom? Why? For what?

- *Example:*
  - Building vs. Build Plan



- Rule declaration for the elements in a DTD:

<!ELEMENT flirt	(name, telephone, email, city, dates)>	
<!ELEMENT name	(&#PCDATA)>	← Text
<!ELEMENT telephone	(mobile   home   work)+>	
<!ELEMENT mobile	(&#PCDATA)>	↑ Selection list
<!ELEMENT home	(&#PCDATA)>	
<!ELEMENT work	(&#PCDATA)>	
<!ELEMENT email	(&#PCDATA)>	
<!ELEMENT city	(&#PCDATA)>	
<!ELEMENT dates	(firstdate, lastdate)>	
<!ELEMENT firstdate	(&#PCDATA)>	
<!ELEMENT lastdate	(&#PCDATA)>	

## Lecture 5 Communication Systems I - Layer-based Communication (Slide 10)

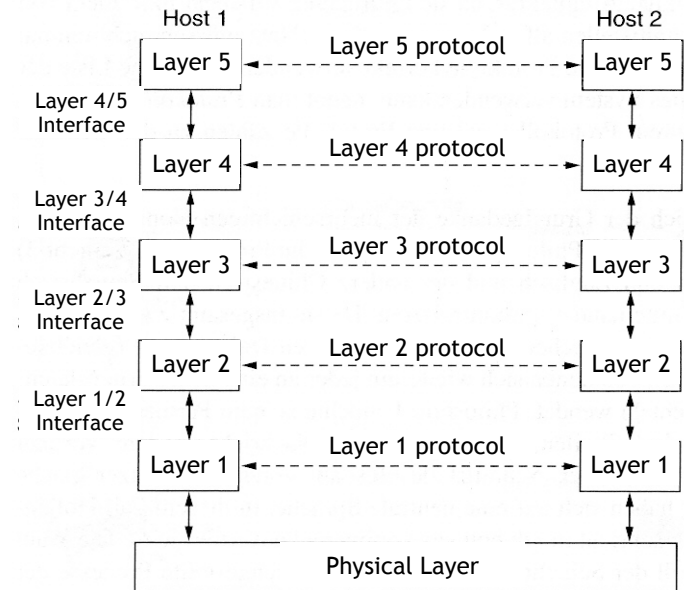
### Frage:

- Was ist damit gemeint, dass der Informationsfluss durch Schnittstellen minimiert werden soll?

- When choosing the relevant functions, protocols should be respected which were previously standardised on an international level.
- Boundaries between layers should be chosen in a way that information flow through interfaces is minimised.
- The number of layers should be sufficiently big to not have different functions grouped inside one layer, but also small enough to not be left with an unmanageable model or architecture.

# 5-layer Model

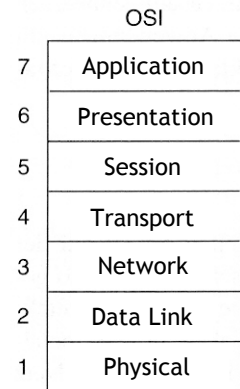
- The figure shows 5 layers.
- Communication inside one layer uses the respective **protocol**.
  - In telecommunication, a communication **protocol** is a system of rules that allow two or more entities of a communications system to transmit information via any kind of variation of a physical quantity. The protocol defines the rules syntax, semantics and synchronization of communication and possible error recovery methods (Wikipedia).
- **No direct data communication** from layer n of one host to the same layer n of another host.
- Each layer sends data and control messages to the layer below until the lowermost layer was reached.
- Located below layer 1 is the physical transmission medium which is used for the communication.





# OSI Reference Model

- The OSI reference model consists of **seven** layers:
  - **Layers 1-4** are **transport-oriented** layers as they focus on the transportation of bits.
  - **Layers 5-7** are **application-oriented** layers as they support the use of data transport applications.
  - The following basic ideas have lead to the 7 layers:
    - Whenever a new **level of abstraction** is necessary, a new layer has to be introduced
    - Each layer should deal with one well-defined task or function.



## Lecture 5 Communication Systems I - Layer-based Communication (Slide 13)

### Frage:

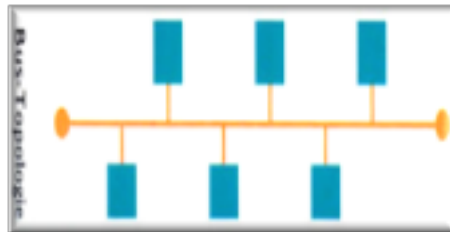
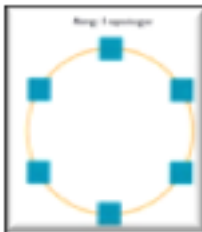
- Sind hier mit Topologien die Topologien wie Bus- oder Sterntopologie gemeint?

- The **network layer** takes care of the transmission of packets **from the source to the destination**. This may also include *passing through* network segments on the way between two routers.
  - The main task of this layer is **routing**.
    - Network Layer needs to know about the topology of the communication network in order to be able to choose an adequate path for each packet
    - A path must be chosen carefully in order to avoid capacity overload or congestion of (sub)networks.

- Wired networks
  - Use different types of conductors to build a network
    - Coaxial cable, twisted pair wire, optical fibre, ...



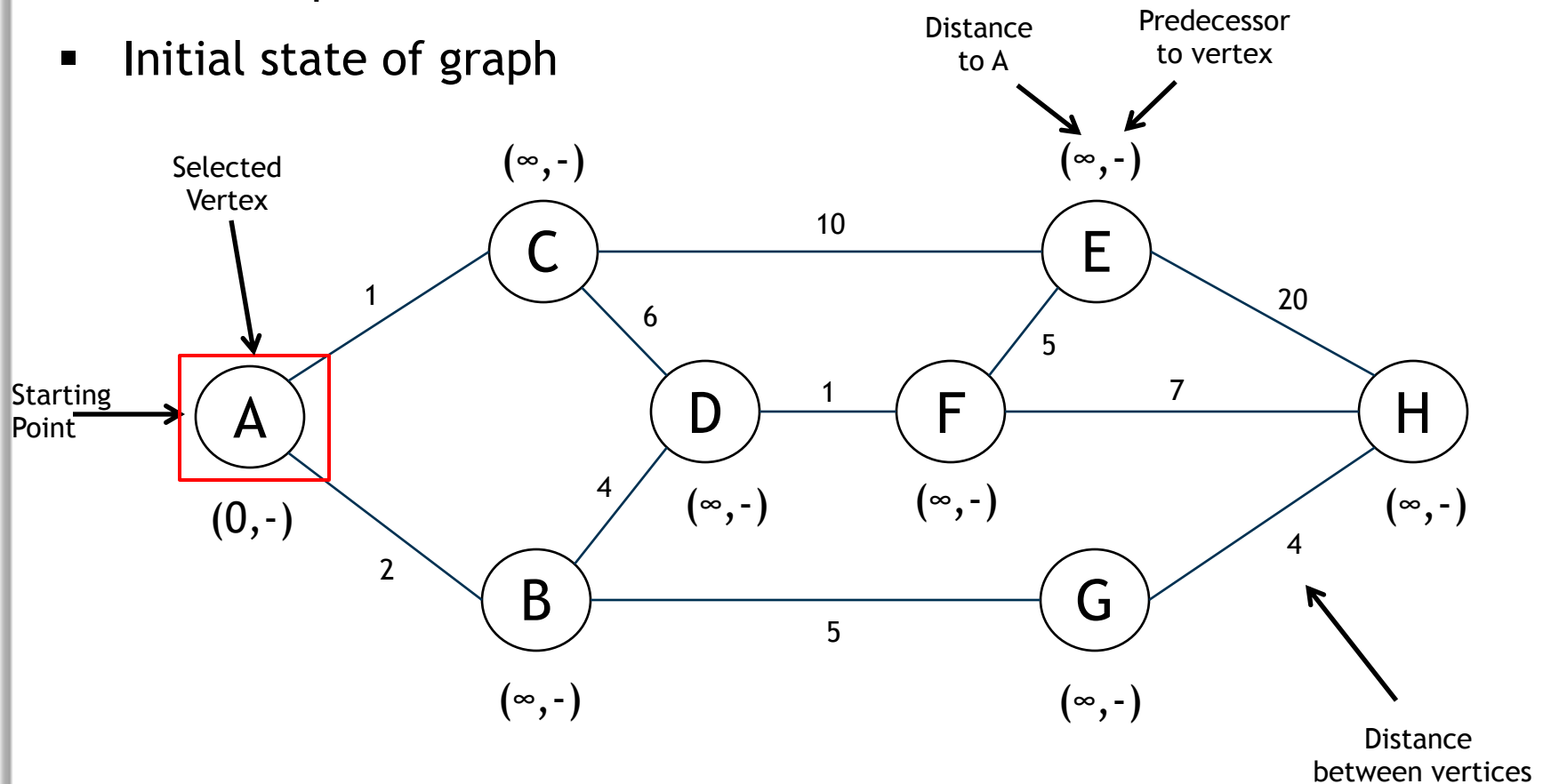
- Make use of different topologies
  - E.g. ring, bus, star topology



- Provide different transfer rates
  - 10 / 100 / 1000 Mbit / 1 Gbit / 10 Gbit and more...

# Layer 3: Network Layer Using Dijkstra Algorithm

- Shortest path from A to H?
- Initial state of graph



- Definition of wiring and signalling standards for local area networks (LAN)
- Developed in the beginning of the 1970s by Xerox and from 1980 continued by DEC, Intel and Xerox (DIX group)
- Implementation of an access method called Carrier Sense Multiple Access/Collision Detection (CSMA/CD), an enhancement of the Aloha concept, which was previously developed at the University of Hawaii
- Thus, Ethernet is a product name by Xerox, DEC and Intel, in which the product itself had not been compatible to the IEEE specification 802.3 for years.

Lecture 5 Communication Systems I - Layer-based Communication (Slides 12 and 14)

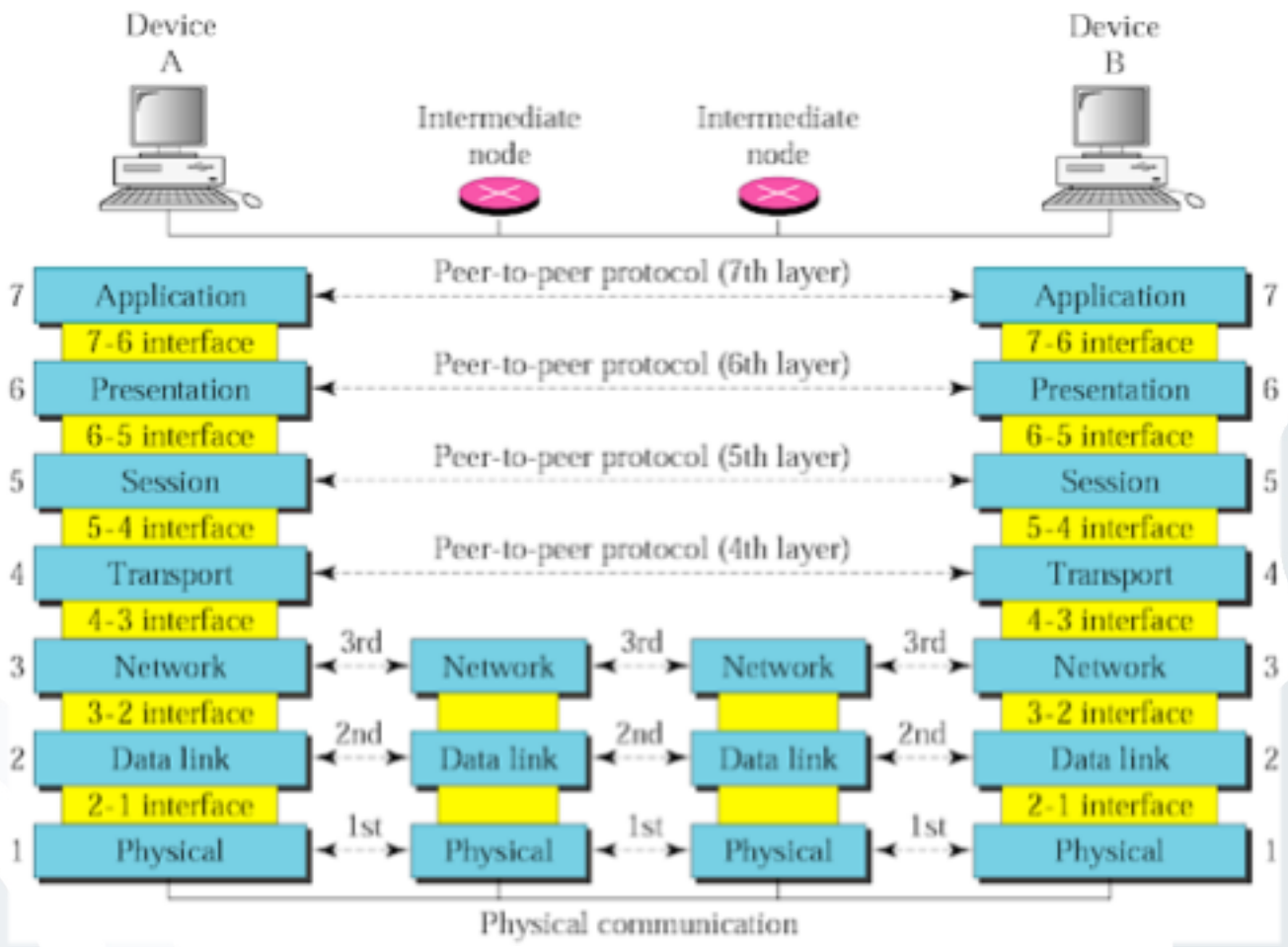
**Frage:**

- Findet die Segmentierung bzw. Aufteilung in kleine Pakete auf der 2. oder 4. Schicht statt?

- The **data link layer** contains **algorithms for efficient and reliable communication** between neighbouring communicating parties.
  - Tasks are:
    - Error control
    - Flow control
    - Error detection and correction
  - This can be assured by:
    - Segmentation; allocation of data to small packets to be passed on sequentially
    - Confirmation; every packet which was properly received is confirmed by the communicating party which receives it
    - ...



- The **transport layer** is the core of the protocol hierarchy. It ensures reliable and cost-effective transport of data from the source to the destination, independent from physical networks (**logical end-to-end connection**). Cost-effective often means the shortest path.
- Tasks include:
  - Logical set-up of end-to-end connections
  - Receiving data from the session layer (layer 5) and dividing it into smaller segments, handing these segments over to the network layer (layer 3), keeping track and ensuring that all of them are received by the receiving party.

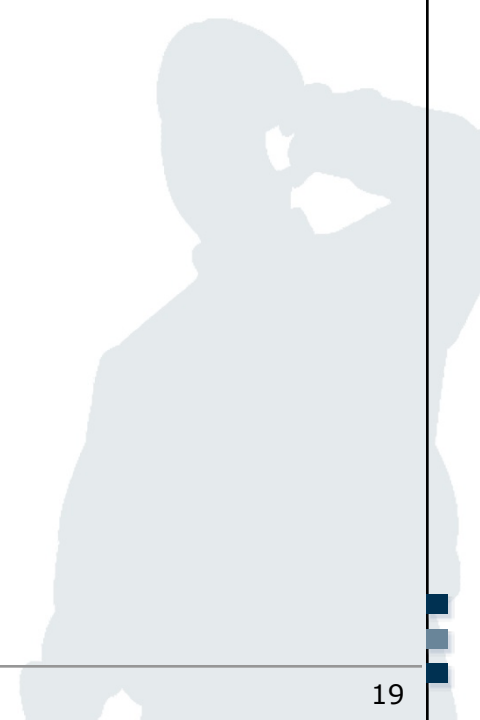


Source: <http://studyregular.blogspot.com/2016/10/computer-networks-topic-osi-model.html>

## Lecture 7 Management of ICT Projects

### Frage:

- Können Sie nochmal genauer erklären, was wir unter ICT Projekten verstehen?



Project characteristics according to DIN 69 901 “Project Management, terms”:

- Unique conditions
- Well defined goals and objectives
- Fixed start date and end date
- Limitations in terms of time, finance, staffing, etc.
- Differentiation from other projects
- Project-specific organisation

# Particular Characteristics of ICT Projects

- Most of the time complex and innovative tasks
- Typical duration between 6 and 12 months
- Low degree of freedom (e.g. regarding costs and dates)
- Time and cost risk are very high and hard to measure for complex and innovative project types.
- Often many employees from different departments are involved:  
→ Competition for resources within an enterprise
- Constantly changing technology
- Integration of interfaces to other Information Systems
- Consideration of existing legacy systems
- ...

- Development of customised Information Systems
- Selection, configuration, and introduction of standard software (e.g. ERP systems or security software)
- Planning and installation of ICT infrastructure
- ICT Projects for business process optimisation
- Outsourcing of ICT services or parts of the ICT infrastructure
- ...

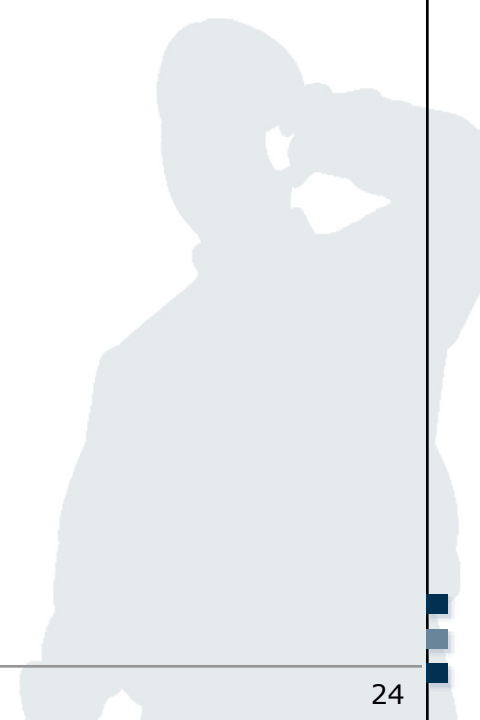
# Critical Success Factors for ICT Projects

- Clearly defined objectives and strategy
- Competent and motivated project members
- Commitment from company executives
- Efficient governance
- Realistic time and resource planning
- Integration of end users into the development process
- Risk management
- Efficient software development system and infrastructure
- ...

## Lecture 7 Management of ICT Projects (Slide 13)

### Frage:

- Was hat Wannacry(pt) mit ICT Projekten zu tun?





- Ransomware is a malware software which encrypts the victim's files and requests a payment to decrypt the files.
- WannaCry(pt) is a ransomware software which makes use of vulnerabilities of outdated Windows XP systems.
- Microsoft still provides Windows XP updates but not for free. Therefore, it is an economic cost/benefit decision to update or not to update a system.
- In case of the U.K. National Health Service it was a political decision taken at the highest level of the government not to spend money to keep receiving security updates.
- Security incidents can often be explained by an economic/political perspective on information security.

Source: Danezis (2017)

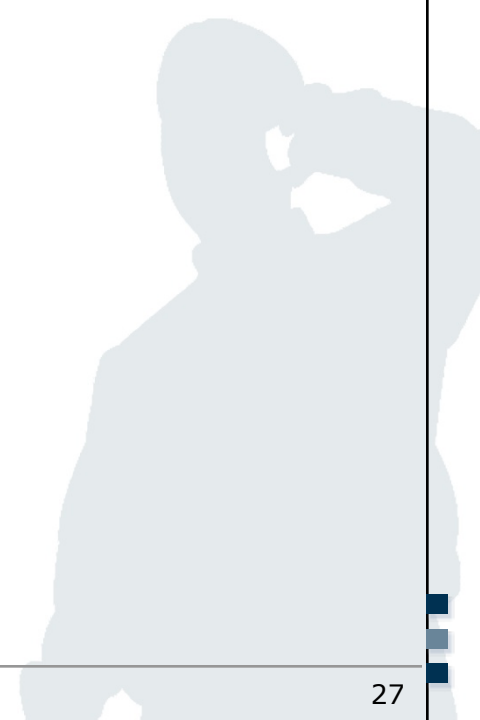
# Critical Success Factors for ICT Projects

- Clearly defined objectives and strategy
- Competent and motivated project members
- Commitment from company executives
- Efficient governance
- Realistic time and resource planning
- Integration of end users into the development process
- Risk management
- Efficient software development system and infrastructure
- ...

## Lecture 8 ICS Development I - Software Engineering

### Frage:

- Wo liegt der Unterschied zwischen Software Engineering und ICT Projekt Management?



# Why Do Software Development Projects Fail?



How the customer explained it



How the project leader understood it



How the analyst designed it



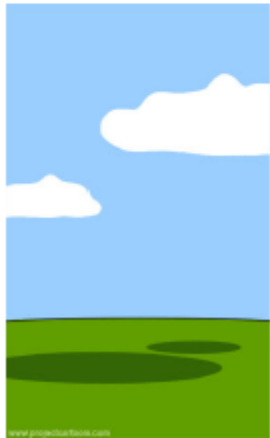
How the programmer wrote it



What the beta testers received



How the business consultant described it



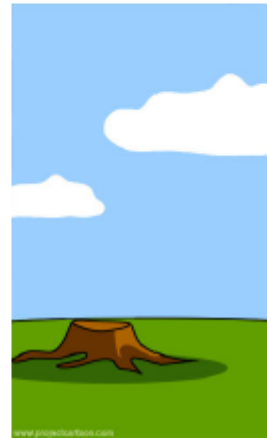
How the project was documented



What operations installed



How the customer was billed



How it was supported

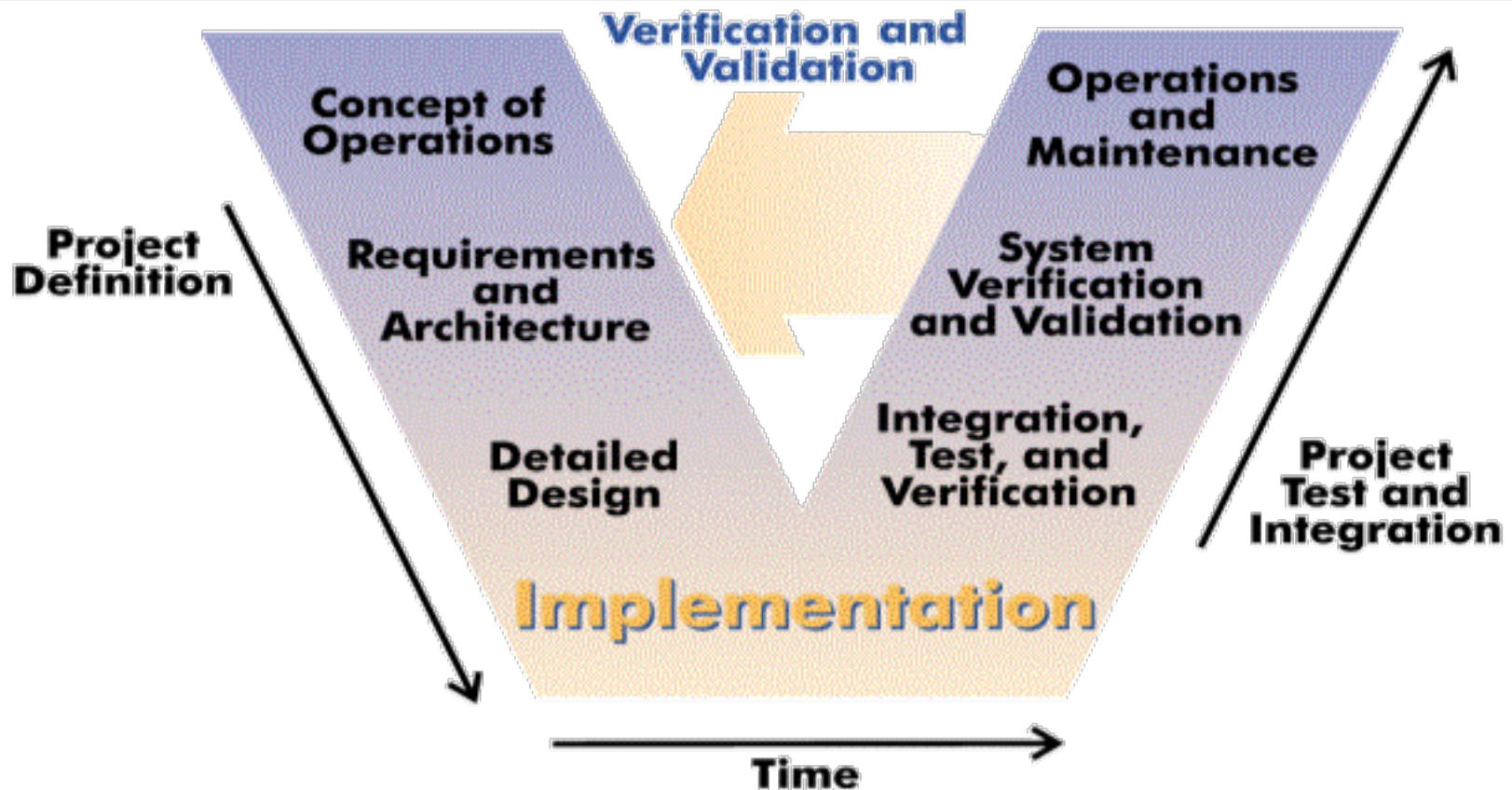


iSwing

What marketing advertised



What the customer really needed

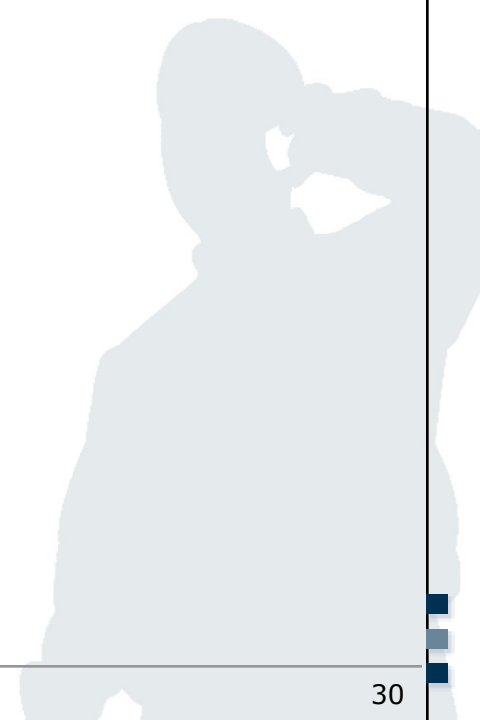


- Horizontal lines denote the information flow between activities at the same abstraction level.
- First proposal in 1979

## Lecture 10 ICS Development III - Markup Languages (Slide 25)

### Frage:

- Können Sie anhand eines Beispiels die Verwendung von „ANY“ und „EMPTY“ erklären?



- Content (in elements):

EMPTY	Empty element
ANY	Any content
	Selection list
,	Sequence
()	Grouping
(#PCDATA)	<i>Parsed Character Data</i> (mixed data)

- Cardinalities (for elements):

	empty: exactly one value is necessary
+	At least one value
?	None or one value
*	None or multiple values

```
<?xml version="1.0"?>
<Root>
  <Eintrag>
    <Leer></Leer>
    <Beliebig>String
  </Beliebig>
</Eintrag>
<Eintrag>
  <Leer/>
  <Beliebig>
    <NeuerTag1>String
  </NeuerTag1>
    <NeuerTag2>String
  </NeuerTag2>
  </Beliebig>
</Eintrag>
<Eintrag>
  <Leer/>
  <Beliebig></Beliebig>
</Eintrag>
</Root>
```

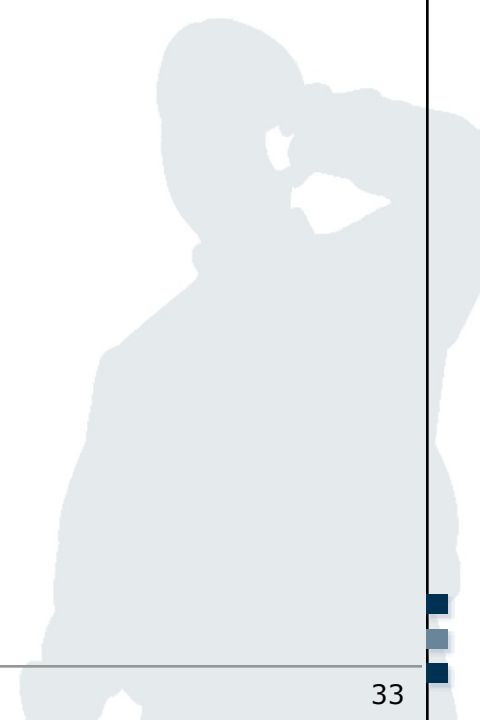
```
<!DOCTYPE Student [
  <!ELEMENT Root Eintrag*>
  <!ELEMENT Eintrag (Leer,
  Beliebig)>
  <!ELEMENT Leer EMPTY>
  <!ELEMENT Beliebig ANY>
]>
```



## Lecture 11 Database Management I - Databases & Data-oriented Modelling (Slides 42-44)

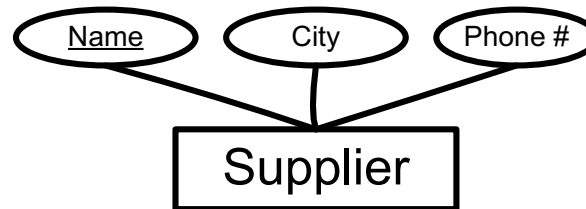
### Frage:

- Was bedeutet auf diesen Folien jeweils der erste Stichpunkt?



- The relation type with its corresponding attributes is derived from the entity type.

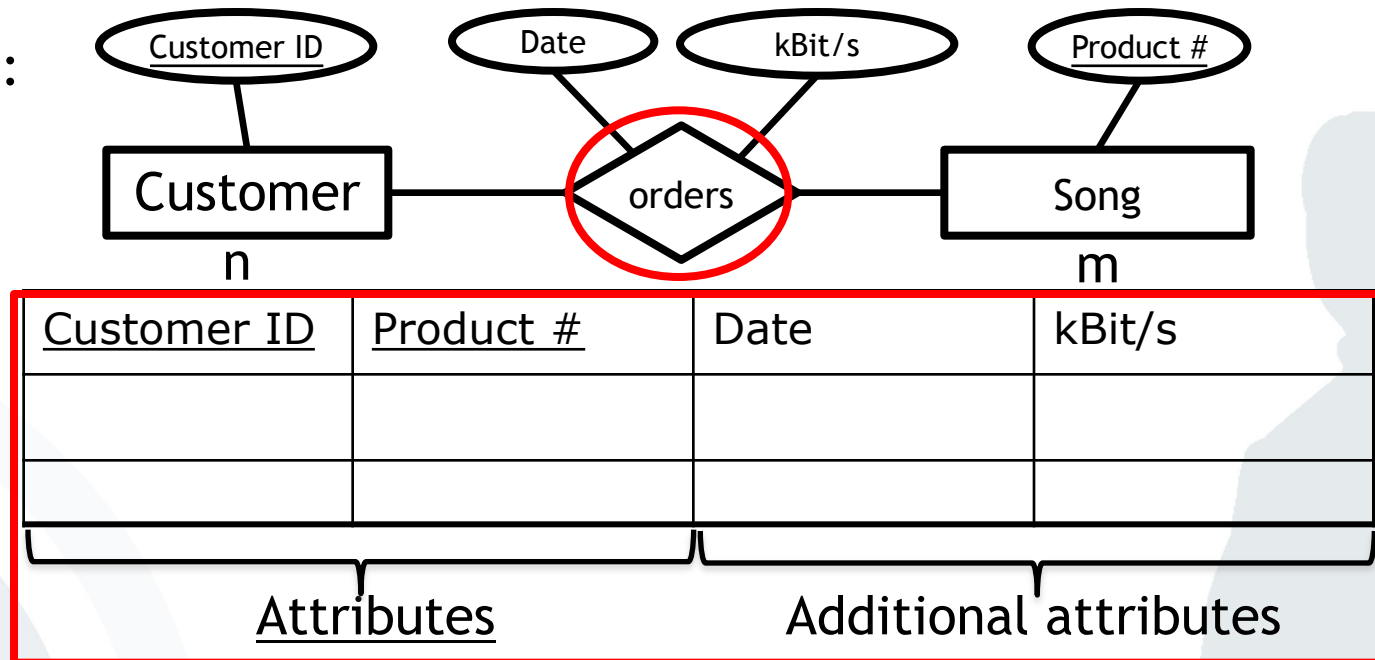
Example:



<u>Name</u>	City	Phone#

- An n:m-relationship type induces an additional relation-type.
- The relation contains
  - primary keys of involved entity types as attributes
  - and additional attributes of the relation types

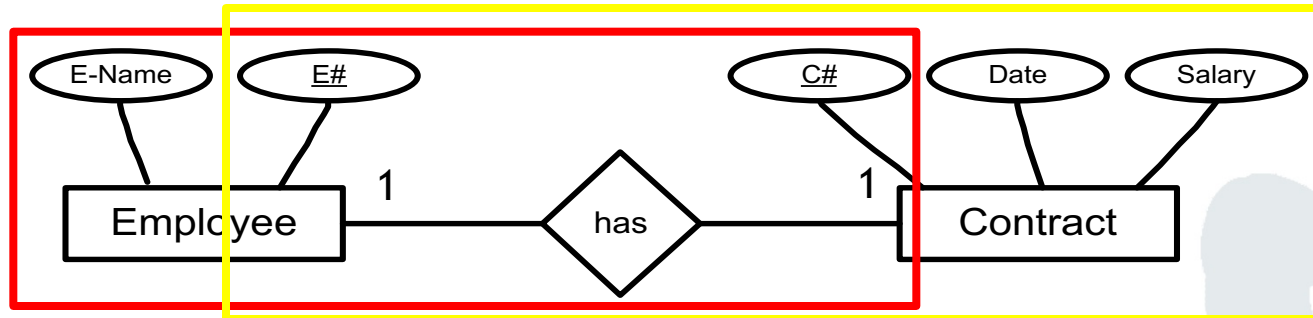
Example:



Note: In order to reflect the complete ER-Model above, two more relations (Customer (Customer ID) and Song (Product #)) are required. The relation above connects both Customer and Song entities.

- A 1:1 relationship type does NOT become a relation on its own.
- The information is to be “attached” to one of the involved entity types.

Example:



Alternative 1:

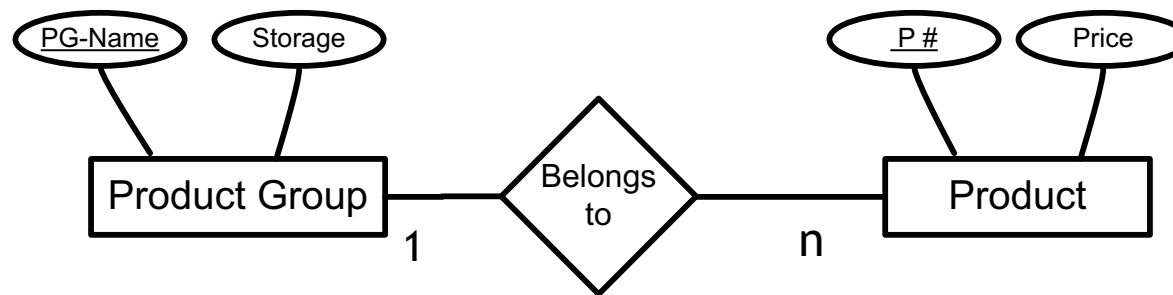
<u>E#</u>	E-Name	C#

Alternative 2:

<u>C#</u>	Date	Salary	E#

- A 1:n relationship type does NOT become a relation on its own.
- The information is to be “attached” to that relation that corresponds to the entity type with the n-signed edge.

Example:

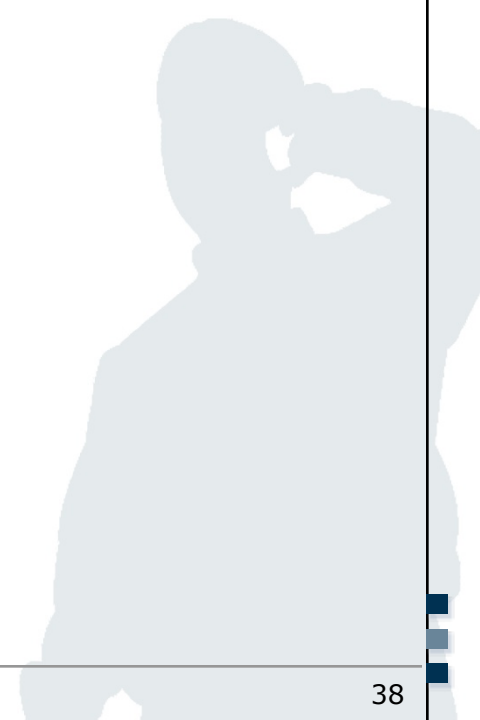


<u>P #</u>	Price	PG-Name

## Exercise 4 Management of IT Projects & Software Engineering

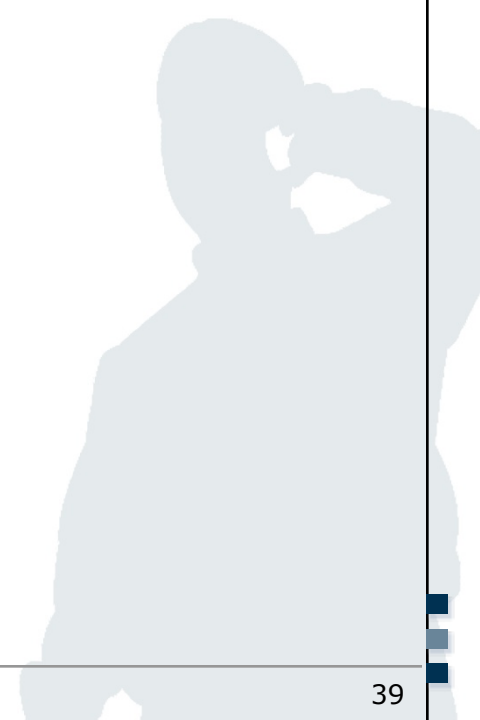
### Frage:

- Inwiefern sollen wir die Software Development Process Models anwenden können?



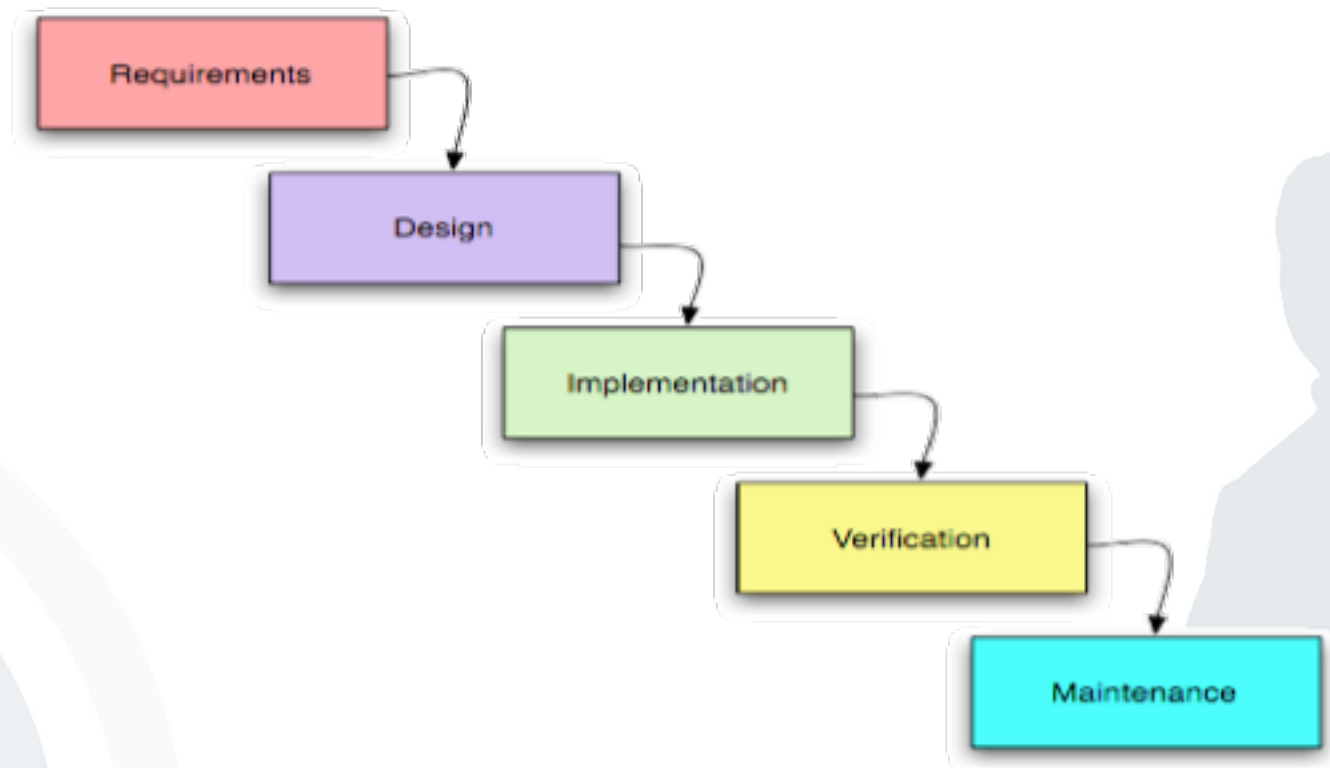
# Exercise 4: Software Development Process Models

- Name and describe three software development process models and select one of them for the InstaMatch Service. Motivate your choice.



## Waterfall model

- First described by Royce in 1970
- There seem to be at least as many versions as there are authorities - perhaps more



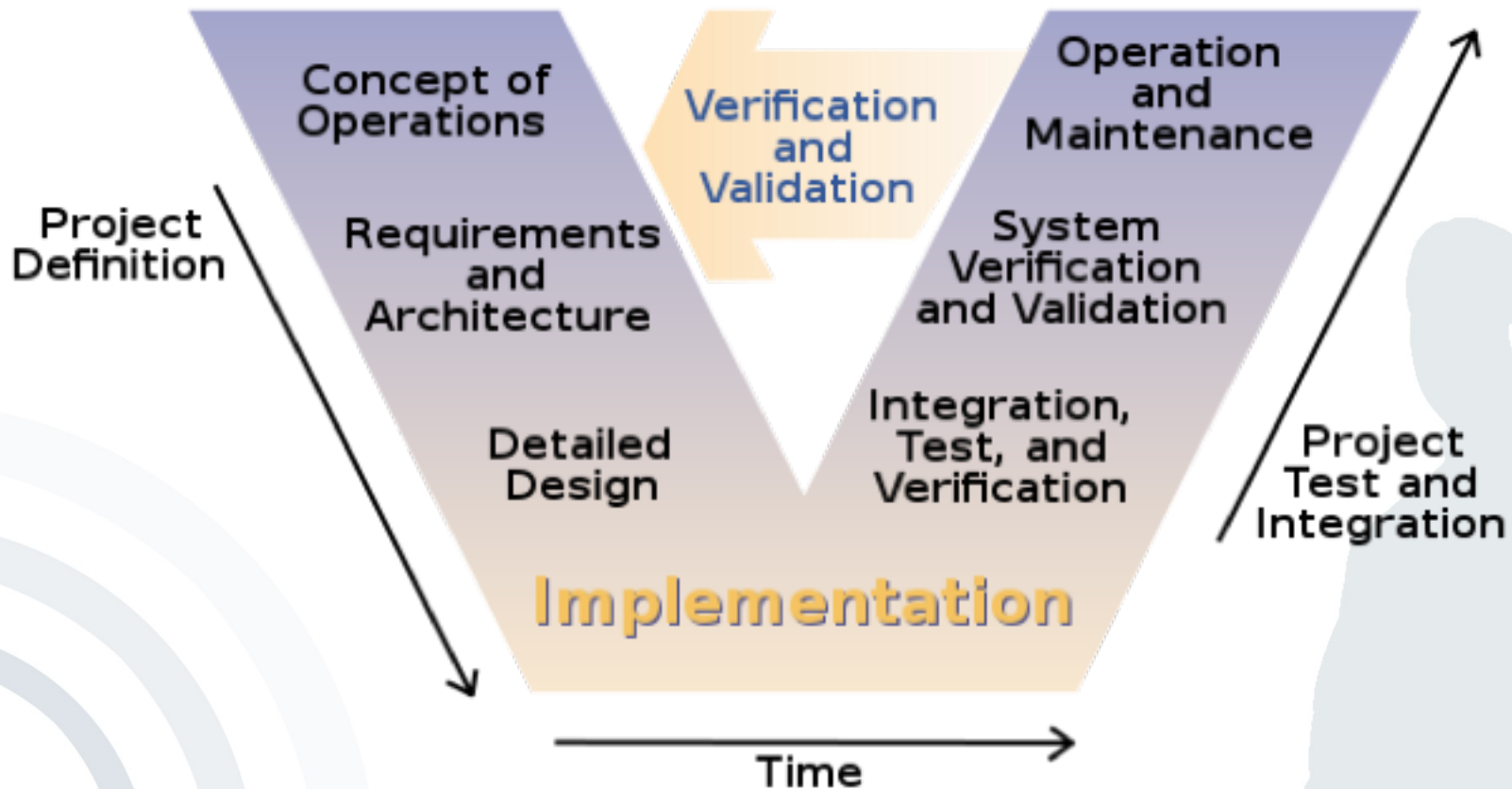


# Exercise 4: Software Development Process Models

- One or more documents are produced after each phase and “signed off”.
- Points to note:
  - “Water does not flow up”.
    - it is difficult to change artifact produced in the previous phase.
  - This model should be used only when the requirements are well understood.
  - Reflects engineering practice.
  - Simple management model.

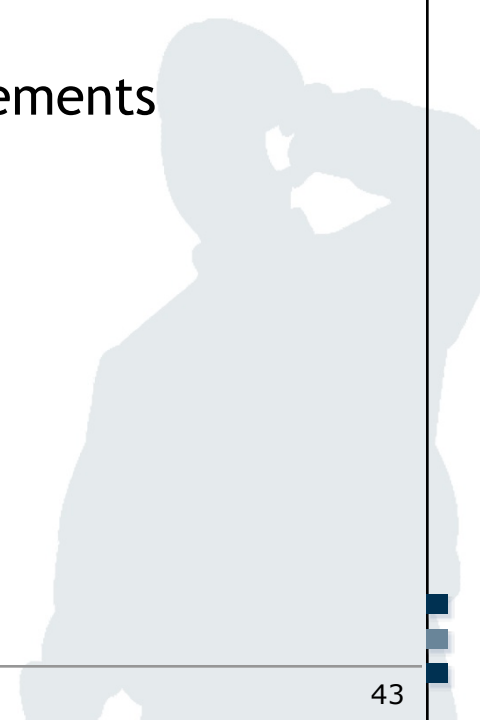
## V-Model

- Horizontal lines denote the information flow between activities at the same abstraction level.



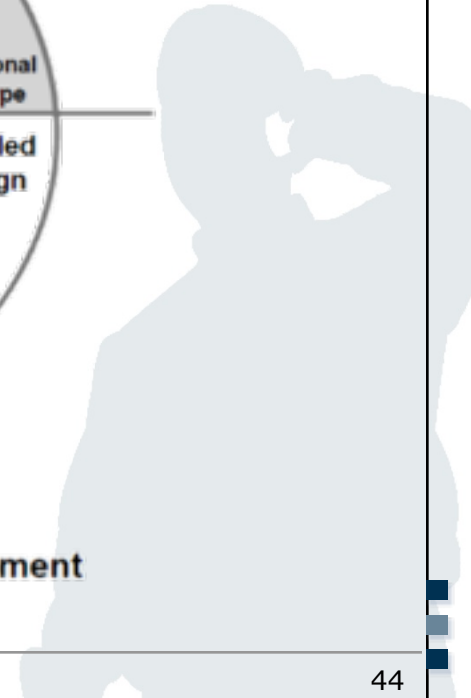
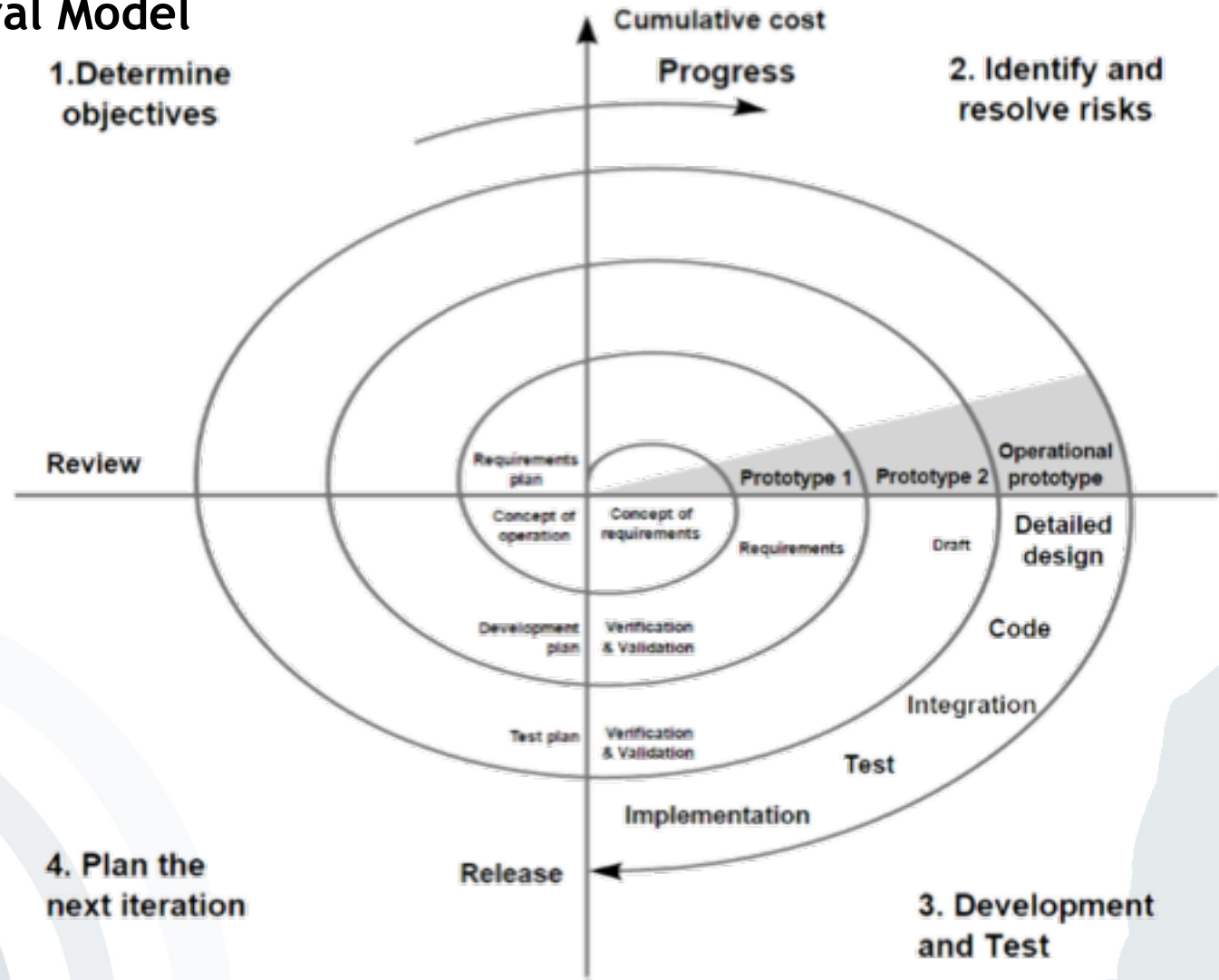
# Exercise 4: Software Development Process Models

- Similar to pure waterfall model but makes explicit the dependency between development and verification activities.
- The left half of the V represents *development* and the right half *system validation*.
- Note the requirements specification includes requirements elicitation and analysis.



# Exercise 4: Software Development Process Models

## Spiral Model



# Exercise 4: Software Development Process Models

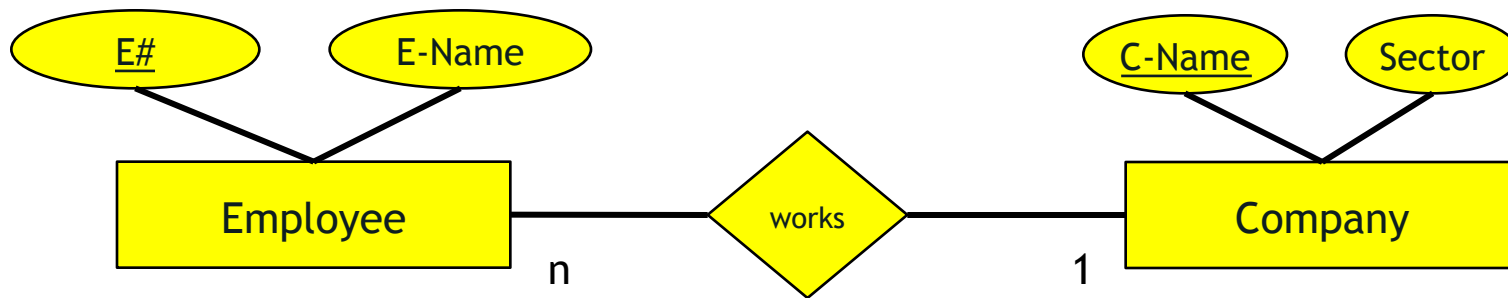
- Basic Concept
  - Develop an initial implementation, demonstrate it to user, get feedback and refine it until an adequate system has been produced.
- Advantages
  - Estimates for budget, schedule, etc. become more realistic as work progresses
- Disadvantages
  - Requires expertise in risk evaluation and mitigation
  - Appropriate only for large systems

- Process model proposal for the InstaMatch<sup>®</sup> Service: V-Model
- Motivation
  - The InstaMatch<sup>®</sup> system is a complex system. The V-model was designed for complex systems.
  - The V-model makes explicit the dependency between development and validation and allows to jump back to earlier development phases.

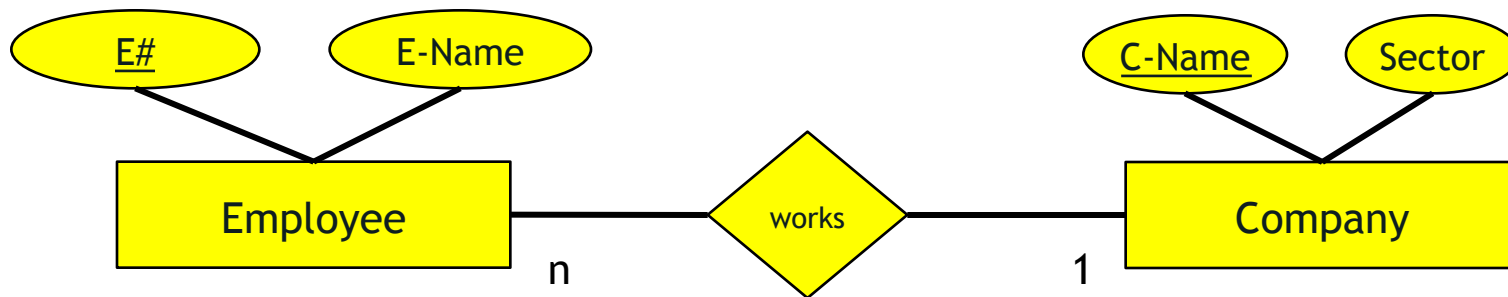
## Exercise 6 Databases & Data-oriented Modelling

### Frage:

- In der zweiten Aufgabe hatten wir nur dann Attribute für die Relation, wenn wir eine  $n:m$ -Relation hatten. In dem Fall gibt es ja dann auch Primary Keys. Um dann die Aufgabe „Derive relations from an ERM“ zu erfüllen, müssen wir bei  $n:m$  Matrizen dann drei Tabellen erstellen. Ist das soweit richtig? -> **JA**
- Könnte es aber auch einen Fall geben, in der wir eine  $1:n$  Relation und trotzdem Attribute für die Relation haben? Wie sollten in diesem Fall die Tabellen, die wir erstellen, aussehen und wie viele sollten es sein?





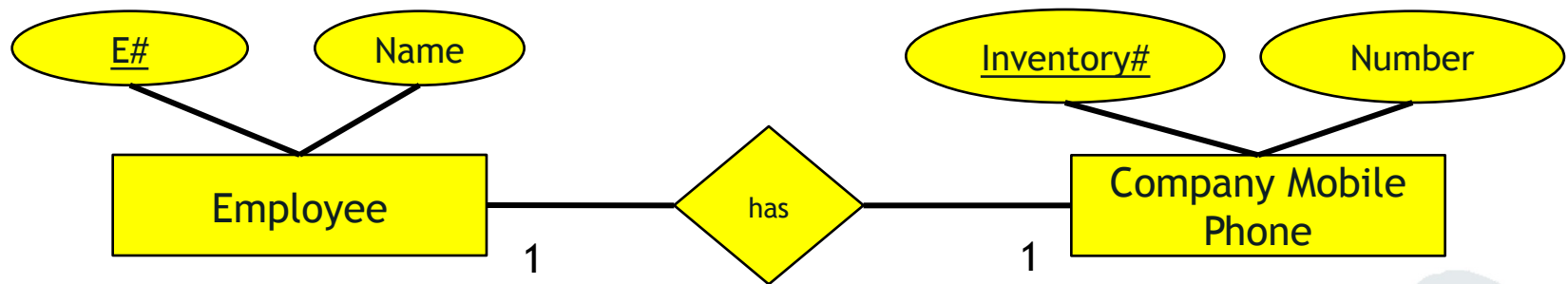


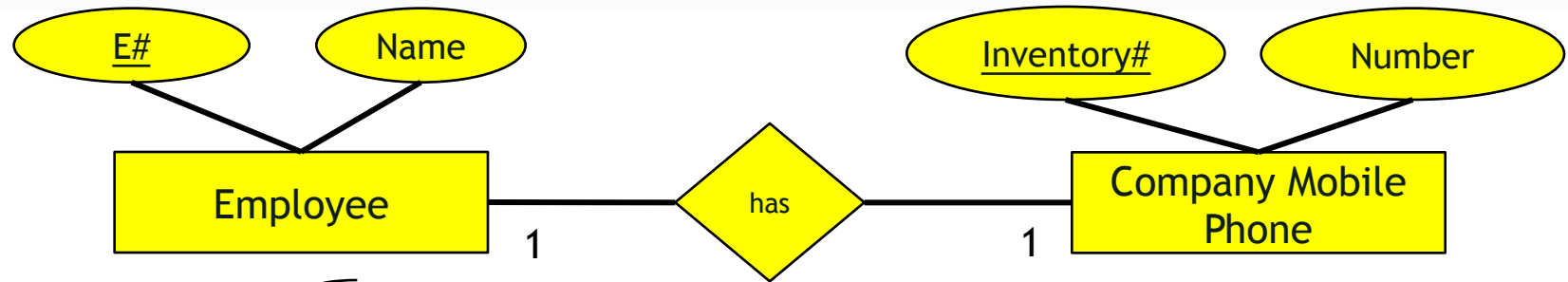
Employee:

<u>E#</u>	E-Name	C-Name
...	...	...

Company:

<u>C-Name</u>	Sector
...	...





Alternative 1  
↑  
or  
↓  
Alternative 2

Company Mobile Phone:

<u>Inventory#</u>	Number	E#
...	...	...

Employee:

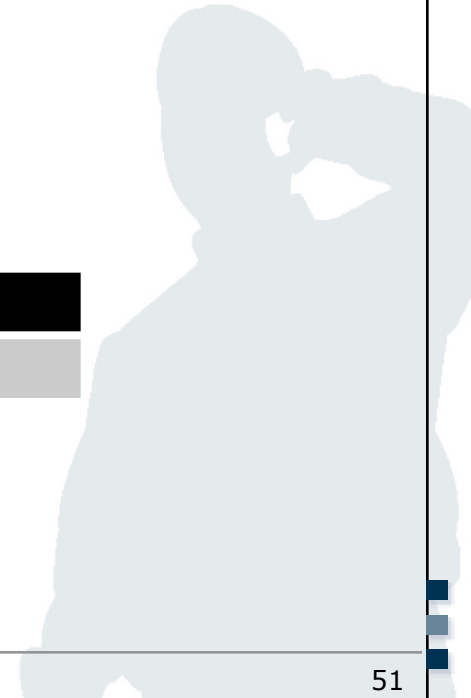
<u>E#</u>	Name
...	...

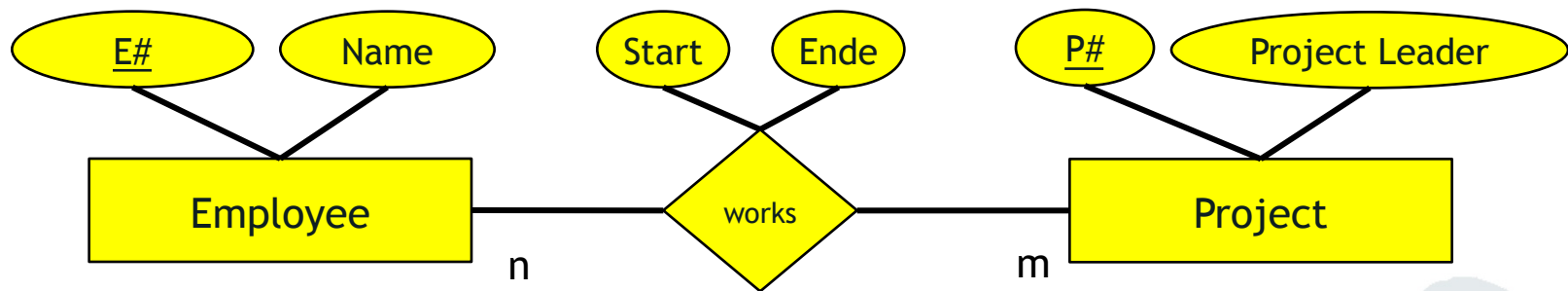
Employee:

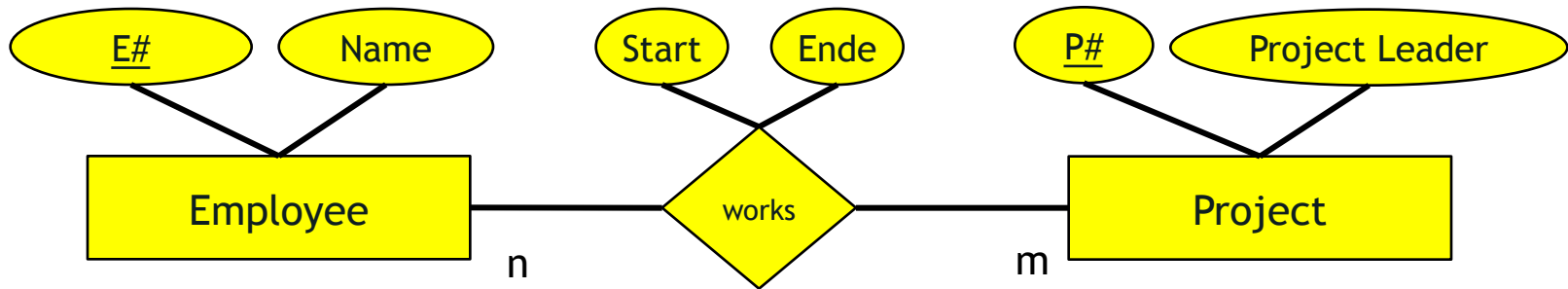
<u>E#</u>	Name	Inventory#
...	...	...

Company Mobile Phone:

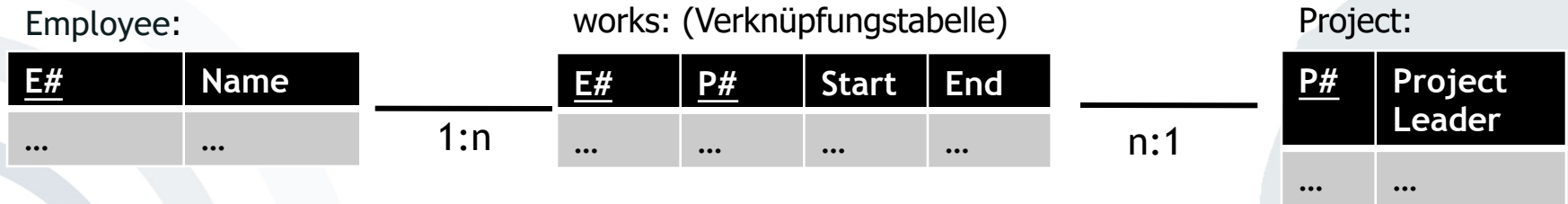
<u>Inventory#</u>	Number
...	...







Eine n:m Beziehung kann in zwei 1:n Beziehungen zerlegt werden, wofür eine Verknüpfungstabelle benötigt wird.  
Hierfür werden die Primärschlüssel der beiden Tabellen benötigt



# Open Questions?

