UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

Requirements Engineering & Management

Scenarios I

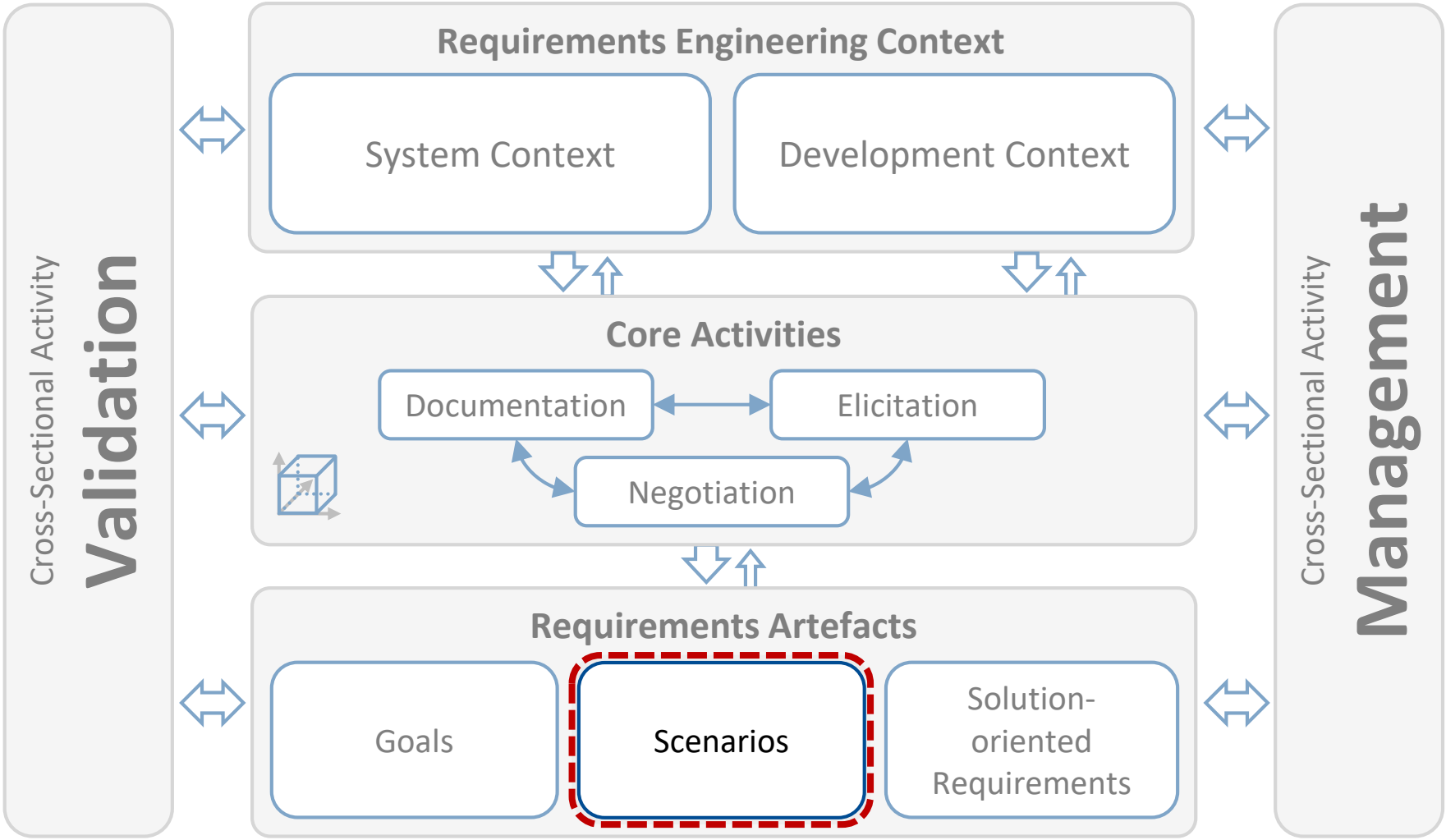
Prof. Dr. Klaus Pohl

PALUNO
The Ruhr Institute for Software Technology

Agenda

1. Fundamentals of Scenarios
2. Scenario Types
3. Documenting Scenarios
4. Benefits of using Scenarios

Framework for Requirements Engineering



1. Fundamentals of Scenarios

Scenario

"[…] A **Scenario** is a **short story** about **people**, their **activities** and the **contexts** in which those activities take place that are relevant to the technology in question. […]"

[Alexander, Maiden 2004, ch. 11, p. 213]

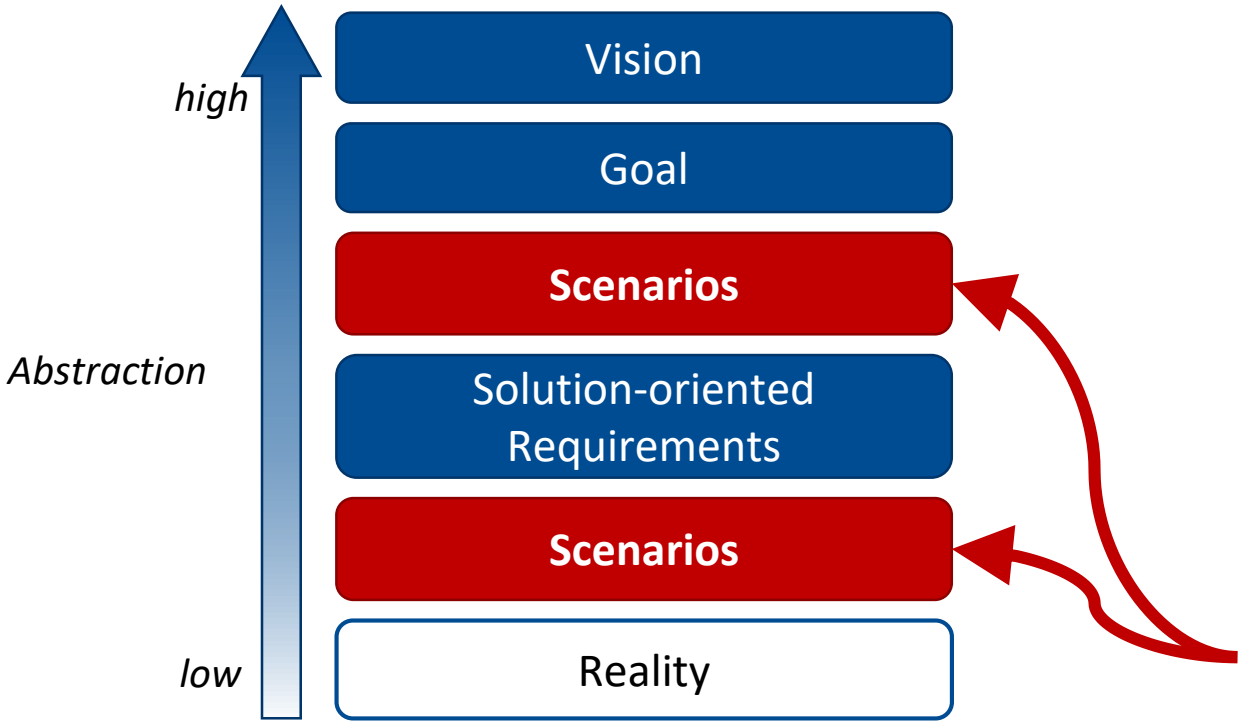
Definition: Scenario

D A scenario describes a **concrete example** of **satisfying** or **failing to satisfy** a **goal** (or set of goals).

- A scenario thereby provides **more detail** about one or several **goals**.
- A scenario typically defines a **sequence of interaction steps** executed to satisfy a goal and relates these interaction steps to the **system context**.

Scenario as Middle Level Abstractions

- Scenarios are **intermediary abstractions**:
  - Between reality and solution-oriented requirements
  - Between solution-oriented requirements and goals



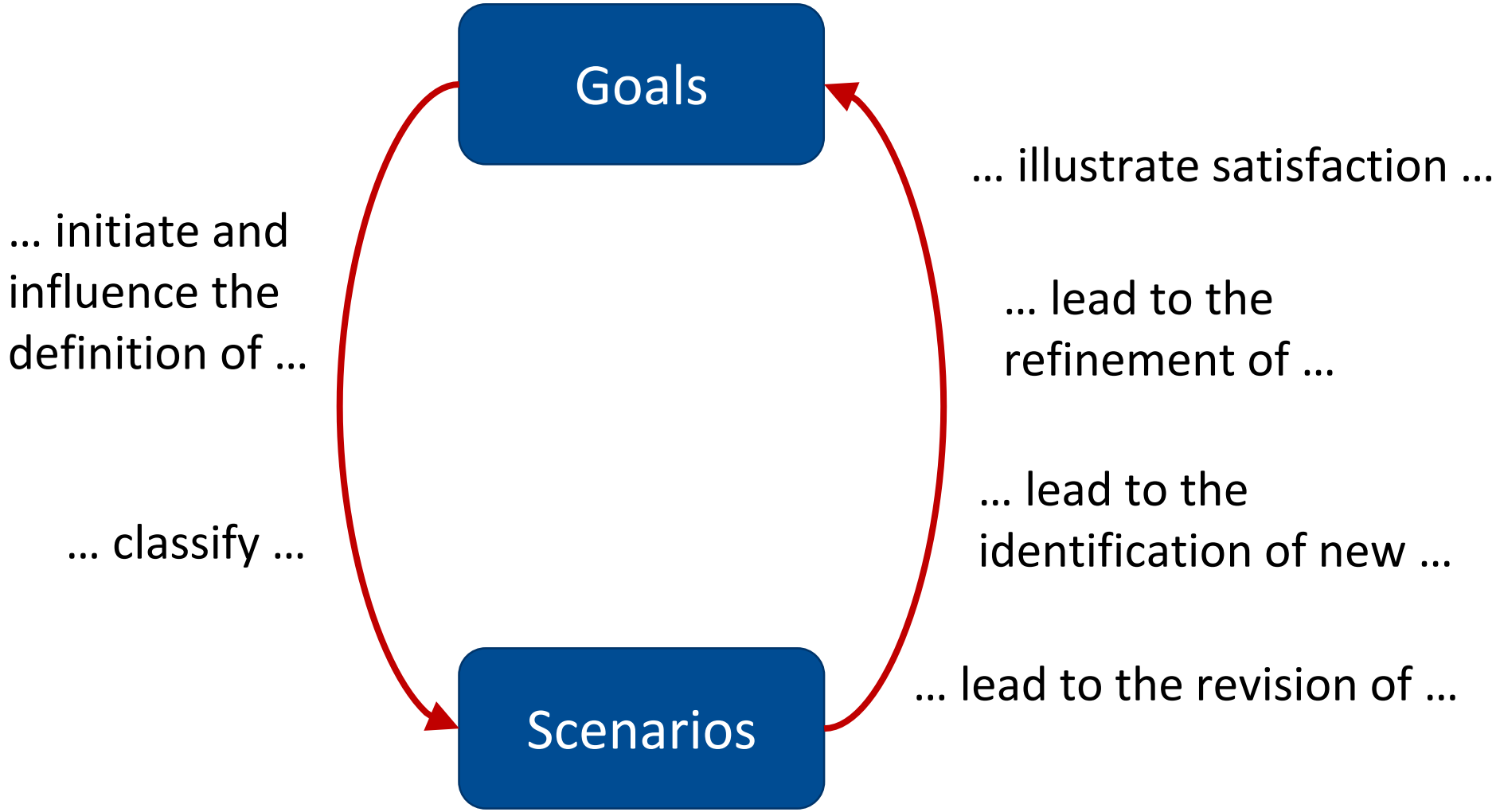
With scenarios you can **capture information** which is:

- very **close to reality** and/or
- **abstract** (almost on a goal or requirements level).
- and any **mixture of those**

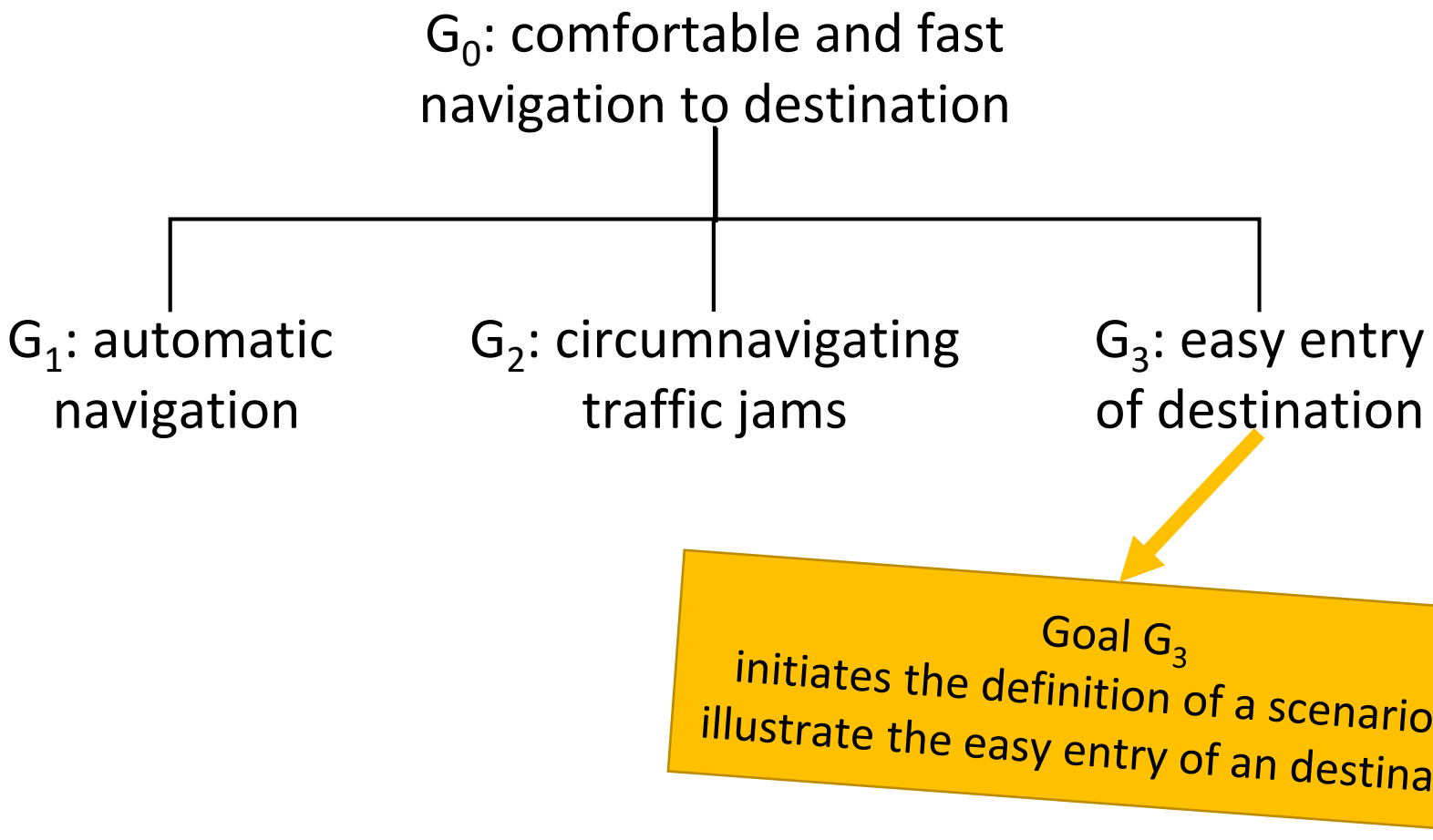
Scenarios in Requirements Engineering

- Focus on **system usage**
- Describe **system application** (old or new) and not the system properties
- Explain **consequences** for users, organization, ...; discuss "**what if**"
- Document **alternatives**, **design rationale** and **causal relationships**
- Lead to the consideration of **user experience**
- Help to detect **quality requirements**
- Document system **behaviour** in the case of an **error**

Interdependencies between Goals and Scenarios



Scenario Definition for a Goal (1)

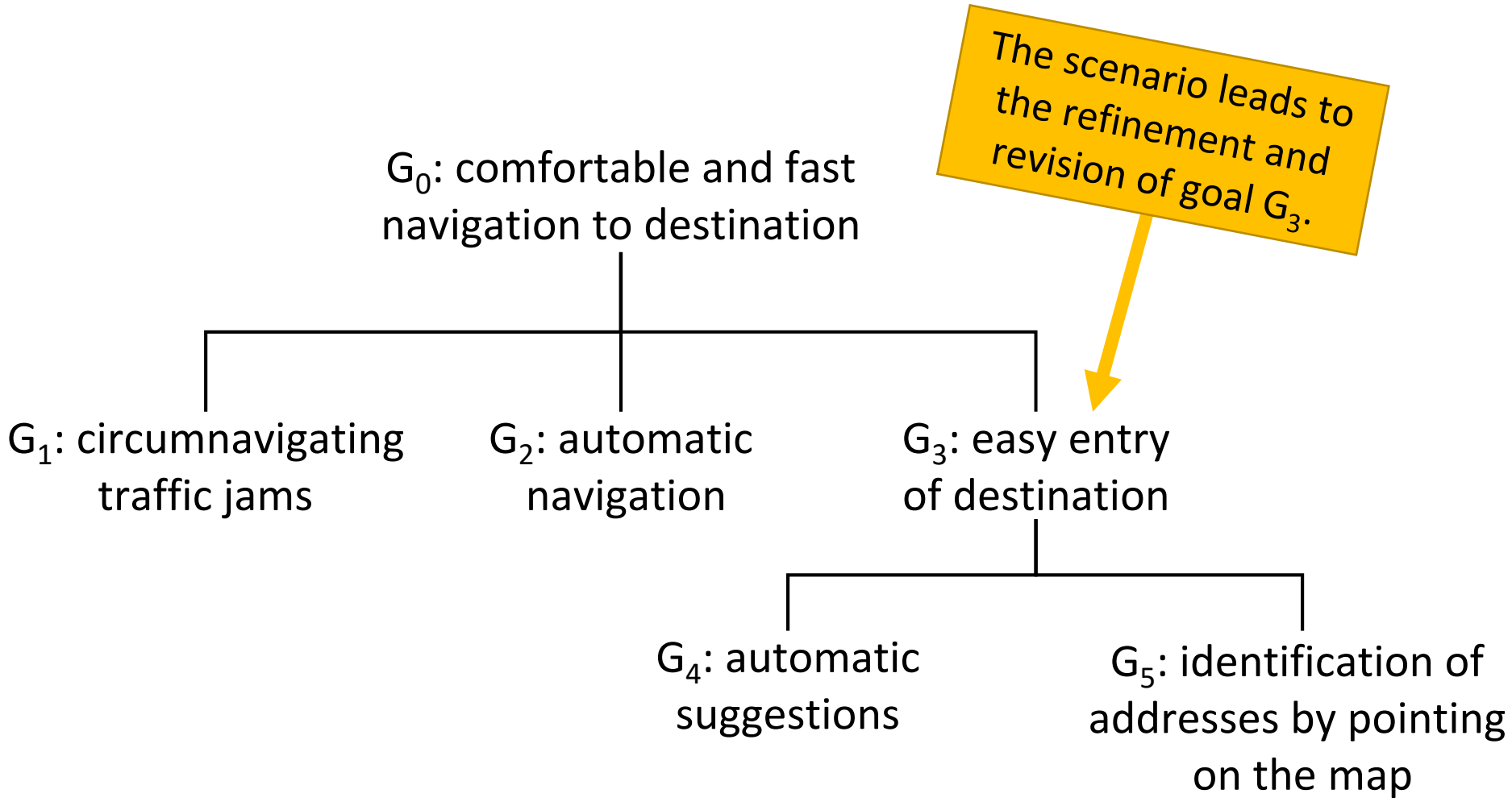


Scenario Definition for a Goal (2)

No.		Section	Content
ID	1.2	Name	Entry of destination
Management	2.1	Actors	Driver
Scenario Definition	4.3	Associated goals	Easy entry of destination
	4.6	Precondition	Navigation system activated
	4.7	Postcondition	Entered destination
	4.8	Result	Driver entered the destination
	4.9	Scenario steps	1. Driver select “navigation to destination” 2. Navigation system asks if the driver wants to select a destination on the map, from the address book or if he likes to enter an address 3. Driver selects the address book option 4. Navigation system provides the entries of the address book of the mobile phone 5. Driver selects an address

Scenario Definition for a Goal (3)

E



2. Documenting Scenarios using Natural Language

Narrative vs. Structured Text vs. Diagram

Narrative

The bank customer inserts their card into the ATM, enters their PIN and chooses the amount they want to withdraw. The ATM returns the card and the money.

Structured Text

User	ATM
1. Users inserts card	
2. User enters PIN	
3. User choose amount	
	4. ATM returns card
	5. ATM issues money

The documentation of scenarios using graphical modelling languages and more formal languages see following lectures.

Narrative Scenario Documentation

Narrative scenarios describe interaction sequences of the intended system behaviour

- As a "**story**"
- In an **unstructured** fashion
- At different **levels of abstraction**

Narrative scenario descriptions can be aided by

- **Documentation guidelines**
- **Reference templates**

Narrative Documentation

The “Send Me a Pizza” app lets users order a pizza from a nearby pizza place using their smartphone. Dave chooses the pizza place with the highest user-rating within a three-mile radius from the dorm. He chooses the three-mile radius to make sure the pizza is still hot when it arrives. Because the app can send the delivery location automatically, users now only have to decide if they want to pay by credit card via the app or in cash when the order arrives.

Narrative Documentation

Instead of having secretaries **photocopy incoming materials** for manual **distribution** and **filling** in several local archives, the entry of material into the hypermedia (e.g., by scanning) should imply **automatic notification** to personnel subscribing to that type of material. This procedure requires **more printers**, but probably **less photocopying** for enabling people to get hard copies quickly. **Photocopies** of certain materials may be made for a few persons who have to print anyway.
However, I do not think that Jane is going to enter the material into the hypermedia.

- **Various alternatives** are shown
- **Argumentation** for alternatives
- Contains **implicit knowledge**

Rules for Narrative Scenario Description (1)

Language and Grammar

- Use present tense.
- Use active voice.
- Use the subject-predicate-object (SPO) sentence structure.
- Avoid modal verbs.

Structure Rules

- Use short sentences.
- Describe each interaction in a separate sentence.
- Number each scenario step.

Rules for Narrative Scenario Description (2)

Content

- Only one interaction sequence per scenario.
- Describe scenarios from the view of third person.
- Explicitly name the actors involved.
- Explicitly state the goal of the scenario.
- Focus on illustrating how the goal is satisfied or not satisfied by the scenario.

Structured Text (1)

- **Improvement of comprehensibility and readability** compared to narrative scenarios.
- **Scenario steps** are typically **enumerated**.
- **Interaction sequences** are typically documented in a **tabular form**.
- **Attributes** of scenarios are typically documented using **scenario templates**.

Structured Text (2)

Scenario “Navigate to destination”	
Driver	**Navigation system**
1. Activates the navigation system.	
	2. Determines the current position.
	3. Asks for the destination.
4. Enters the destination.	
	5. Identifies the relevant part of the map.
	6. Displays a map of the target area.
	7. Asks for the routing options.
8. Enters the routing options.	
	9. Calculates the route.
	10. Displays that the route has been calculated.
	11. Creates a list of waypoints.
	12. Displays the next waypoint.

Template for Scenario Documentation (1)

No.		Section	Content / Explanation
ID	1.1	Identifier	Unique identifier of the scenario.
	1.2	Name	Unique name of the scenario.
Management	2.1	Author	Names of the authors who have worked on the scenario description.
	2.2	Version	Current version number of the documentation of the scenario.
	2.3	Change history	List of the changes applied to the documentation of the scenario.
	2.4	Priority	Indication of the importance of the described scenario according to the prioritisation technique used.
	2.5	Criticality	Criticality of the scenario, e.g. for the overall success of the system.
Context	3.1	Source	Denomination of the sources from which the scenario originates.
	3.2	Responsible stakeholder	The stakeholders responsible for the scenario.

Template for Scenario Documentation (2)

No.		Section	Content / Explanation
Scenario Definition	4.1	Scenario type	Classification of the scenario based on the scenario types .
	4.2	Short description	Concise description of the scenario (approximately ¼ page).
	4.3	Associated goal	Goals which shall be satisfied by executing the scenario including identifiers pointing to the associated goal definitions.
	4.4	Primary actor	Indication of the primary actor (the actor who benefits the most from the execution of the scenario). Typically, the primary actor initiates the execution of the scenario.
	4.5	Other actors	Determination of all other actors involved in the scenario.
	4.6	Precondition	A list of necessary prerequisites that need to be fulfilled before the execution of the scenario can be initiated.
	4.7	Postcondition	A list of conditions that hold after execution of the scenario.
	4.8	Results	Description of the outputs that are created during execution of the scenario.
	4.9	Scenario steps	Detailed interaction sequence of the scenario: narrative / sequentially numbered / tabular
	4.10	Quality requirements	Cross references to quality requirements.

Template for Scenario Documentation (3)

No.		Section	Content / Explanation
Relationships	5.1	Goals	Relationships of the scenario to other goals than mentioned in “Associated goal(s)”.
	5.2	Use cases	Relationships of the scenario to use cases.
	5.3	Scenarios	Relationships of the scenario to other scenarios (e.g. main, alternative scenario).
	5.4	Solution-oriented requirements	Relationships of the scenario to solution-oriented requirements.
	5.5	Other artefacts	Relationships of the scenario to other artefacts.
Miscellaneous	6.1	Supplementary information	Additional information regarding the scenario.
	6.2	Open issues	A list of notes regarding the documentation of the scenario. As soon as the scenario is documented completely this row has to be empty.

Template-based Scenario Documentation

No.		Section	Content
ID	1.2	Name	Buy goods
Scenario definition	4.3	Associated goals	Received goods
	4.4	Primary actors	Customer, Cashier
	4.6	Precondition	Cashier ready; POS system opened
	4.7	Postcondition	POS ready
	4.8	Result	Customer paid goods and left
	4.9	Scenario steps	1. Customer puts goods on belt 2. Cashier scans goods with the scanner 3. POS system sums up amount 4. Cashier calls final amount 5. Customer pays appropriate 6. Customer packs and leaves

Note: For the complete template for scenario documentation, please refer to the previous slides.

Narrative vs. Structured Natural Language

- **Narrative** - **Advantages**
  - Comprehensible to all stakeholders.
  - Can be communicated at different abstraction levels.
  - Narrative scenarios should be used at an early stage to elicit rich content.
    - Descriptive, exploratory, and explanatory elements, as well as alternative and exceptional steps can be documented together with structural, functional, behavioural, and quality aspects.
- **Narrative** - **Disadvantages**
  - Informal and unstructured.
  - Not suitable for later development phases.
- **Structured Text** - **Advantages**
  - Comprehensibility and readability can be significantly improved.
- **Structured Text** - **Disadvantages**
  - Should not be used at a too early stage.

3. Scenario Types

Current vs. Desired State (1)

Two general types of scenarios:

- **Current-state scenario** (indicative scenario).

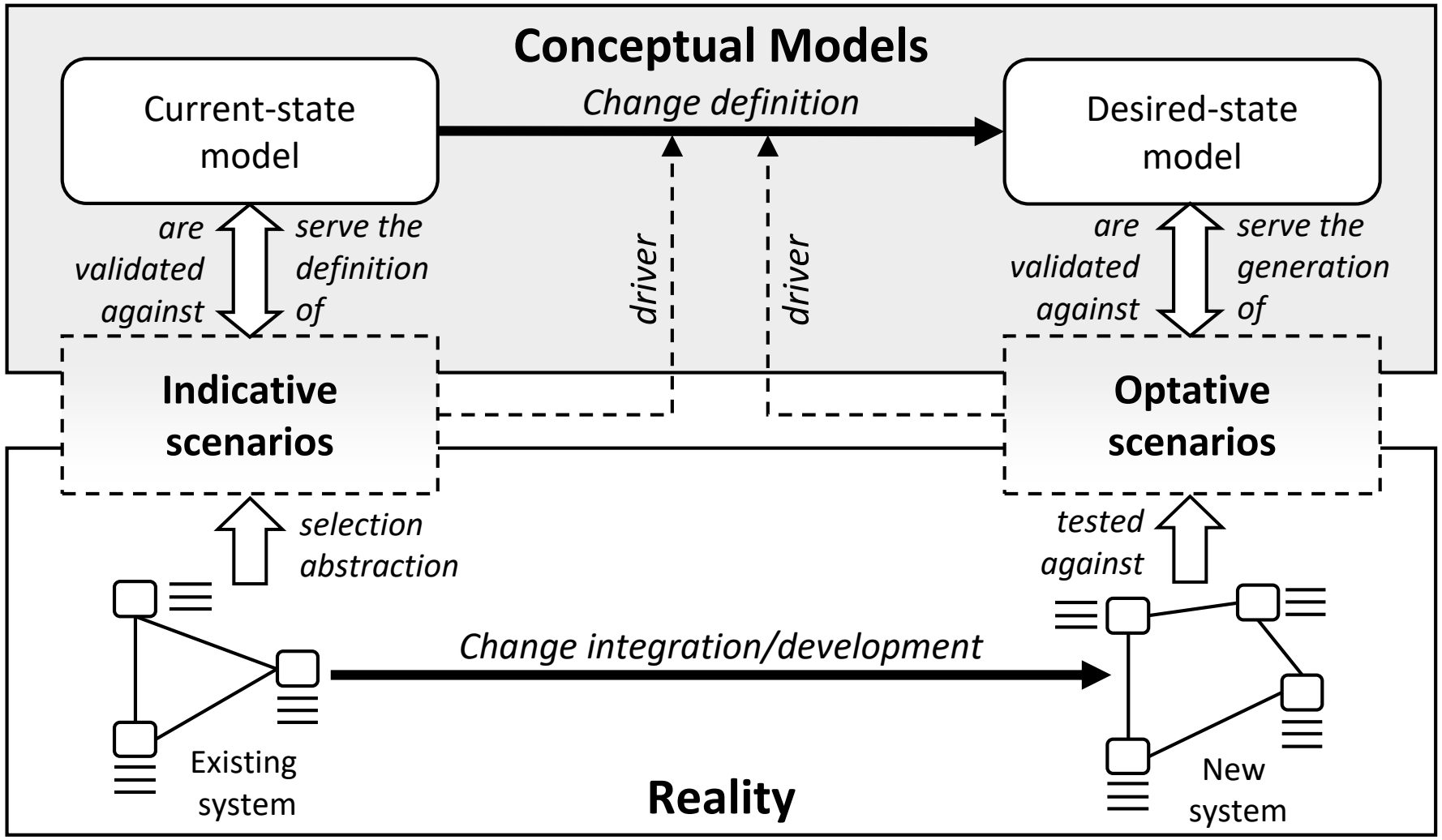
  Describes the **current reality** including the current system usage, quality requirements as well as shortcomings of the current system respectively the system usage.

- **Desired-state scenario** (optative scenario).

  Describes the **intended future reality**, including the intended system usage, quality requirements.

- Indicative and optative scenarios are **important drivers for change definition**.

Current vs. Desired State (2)



Based on [Haumer et al. 1999]

Scenarios at Different Levels of Abstraction (1)

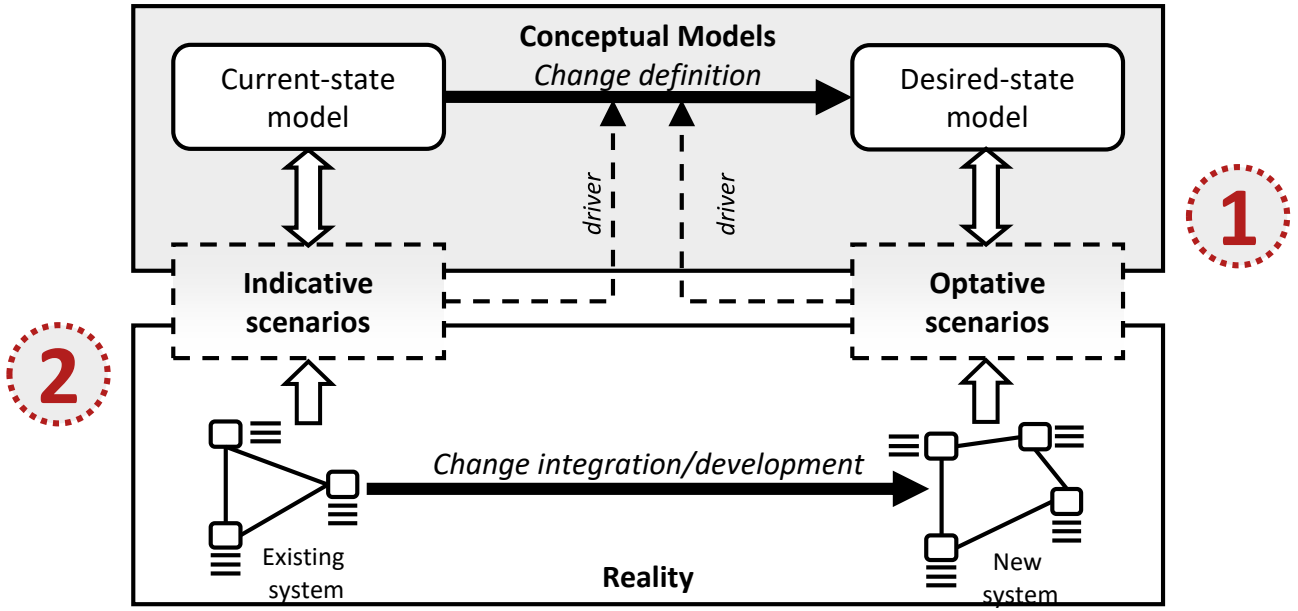
- **Abstractions** used in optative or indicative scenarios:
  - Can **vary** between **very concrete** and **very abstract** (almost conceptual model)
  - Even within one scenario!
- Choice of **abstract** statements in the scenario for aspects which are **well understood**
- Choice of more **concrete** statements in the scenario for aspects which are **not well understood** or not agreed

Scenarios at Different Levels of Abstraction (2)

Examples of scenario excerpts at different level of abstraction:

(1) “the mobile phone offers a flexible display and activates the hologram function.”

(2) “Sam enters the pin-code 342 to access the room 1-012”



Instance vs. Type Level (1)

- **Instance** level scenario (theoretical)

  The complete scenario uses instance level information and avoids type information, e.g., it describes a (descriptive or optative) **sequence of concrete interactions** (e.g., entering of pin “3451”) between **concrete actors** (e.g. Paul and the ATM machine No.45-123)

- **Type** scenario

  The scenario uses type level information and avoids the use of concrete instances. For example, the scenario describes a (descriptive or optative) **sequence of (type level) interactions** (e.g., entering a pin) between actors (e.g. “User” and “ATM Machine”).

Instance vs. Type Level (2)

- Typically: **Mixed use** of instance and type level
- Information at the instance level is an indicator for
  - **Content not completely understood** to avoid errors resulting from early abstractions
  - Conflicting or potentially **conflicting content**
  - Implicit knowledge associated with the instances used
- Information at the type level is an indicator for
  - **Well understood** content, i.e. abstraction of concrete instances to the type level is easy and understandable by most stakeholders

Instance vs. Type Scenario

The driver wants to drive to a destination using the navigation system. Carl wants to drive to Union Street in Plymouth. Carl uses the navigation system of his VW Golf with license number “E-IS-12”. He enters the destination presses the key “calculate route”. The system calculates the route from the current position of the car to the destination Union Street in Plymouth.

Type level

Instance level

Positive vs. Negative Scenario (1)

- Regarding goal satisfaction, a scenario can be
  - **Positive:** Positive scenario documents a sequence of interactions leading to the **satisfaction of a goal**
  - **Negative:** Negative scenario documents a sequence of interactions **failing to satisfy a goal**

- Positive and negative scenarios **complement each other**

Positive vs. Negative Scenario (2)

Negative scenario may be **allowed** or **forbidden**

Chris inserts her bank card into the slot of the ATM (automated teller machine). Chris enters her personal identification number and the amount to withdraw. The amount to withdraw exceeds her balance…

- **Allowed, negative scenario**

  The ATM informs Chris that withdrawing the desired amount is not possible because the amount exceeds her balance.

- **Forbidden, negative scenario**

  The ATM dispenses the desired amount but cannot charge Chris's account.

Main vs. Alternative vs. Exception Scenario (1)

- **Main** scenario
  - Most common sequence of **interactions for satisfying a goal**
- **Alternative** scenario
  - Sequence of interactions that can be executed **instead of main scenario**
  - Results in **satisfaction of the goals** associated with the main scenario
- **Exception** scenario
  - Sequence of interactions executed **instead of interactions documented in the main scenario**
  - **Only** executed when **special events occur** that prohibit goal fulfilment
  - Describes an interaction sequence for **exception handling**

Main vs. Alternative vs. Exception Scenario (2)

Excerpt of the main scenario:

Step 11: …

Step 12: The driver chooses a destination by pointing on an electronic map.

Step 13: …

Excerpt of an alternative scenario:

Step 11: …

Step 12a: The driver chooses a destination from a list of destinations.

Step 13: …

Main vs. Alternative vs. Exception Scenario (3)

Excerpt of the main scenario:

Step 12:	…
Step 13:	The system confirms the successful entry of the destination.
Step 14:	The system informs the driver about the successful calculation of the route to the destination.
Step 15:	…

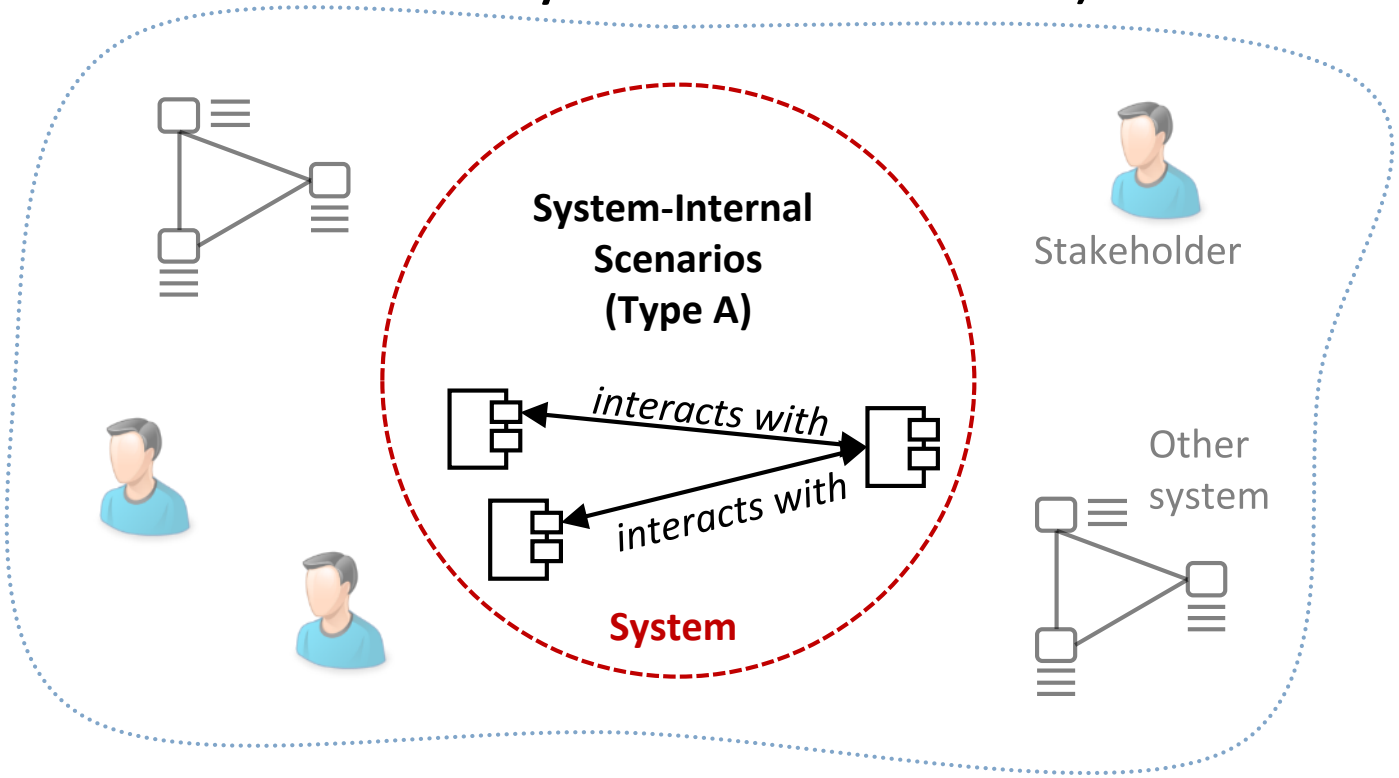
Excerpt of an exception scenario:

Step 12:	…
Step 13a1:	The navigation system is not able to calculate a valid route to the destination.
Step 13a2:	The system informs the driver that navigation is not possible.

System-Internal vs. Interaction vs. Contextual (1)

System-Internal scenarios (Type A)

Focus exclusively on **system-internal interactions**
(interactions that occur within the system boundaries)



System-Internal vs. Interaction vs. Contextual (2)

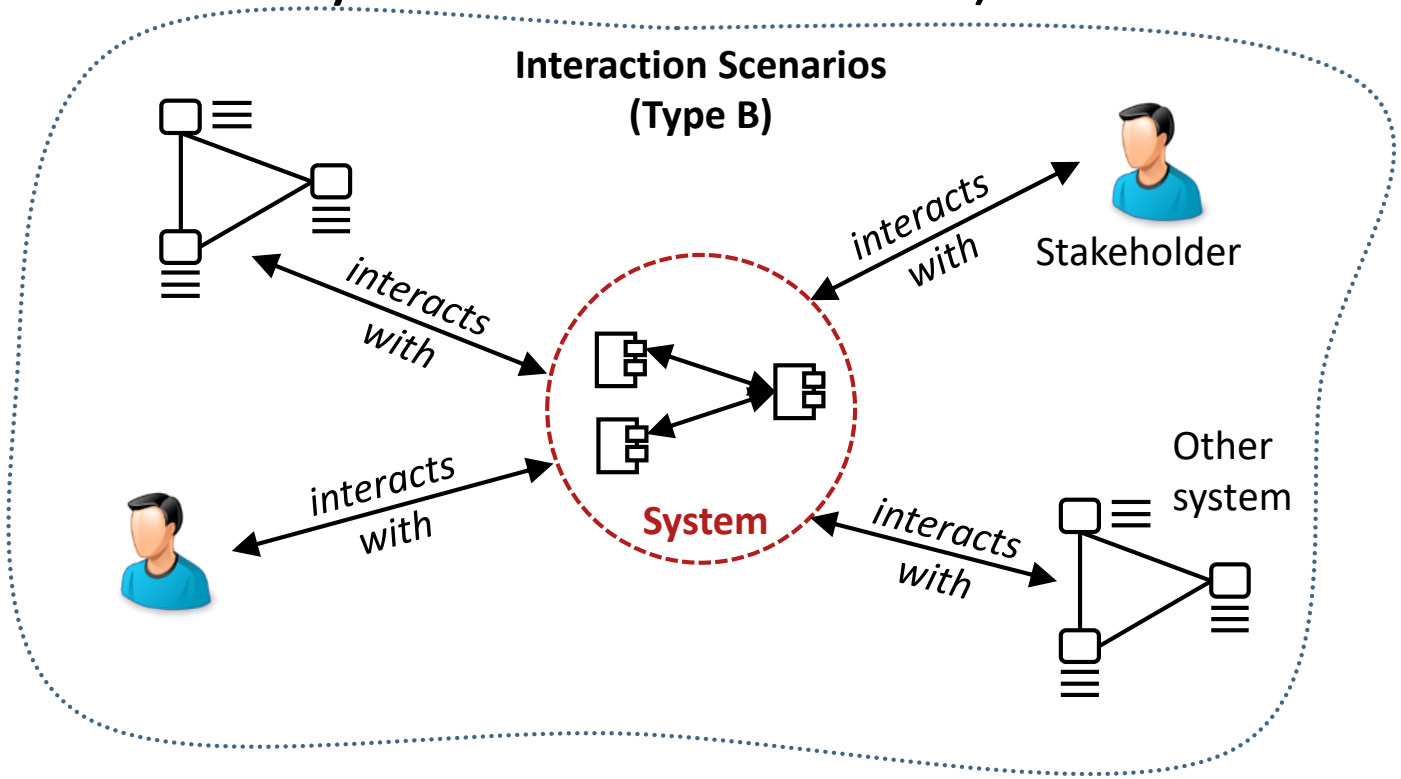
System-Internal scenario (Type A)

The component “navigation control” requests the GPS coordinates from the “localization” component. The “localization” component provides the coordinates to the “navigation control”. The component “navigation control” invokes the component “display control” and passes on the current position and the destination.
The component “screen input” transmits the route parameters to the component “navigation control”. The “navigation control” component calculates the final route.

System-Internal vs. Interaction vs. Contextual (3)

Interaction scenarios (Type B)

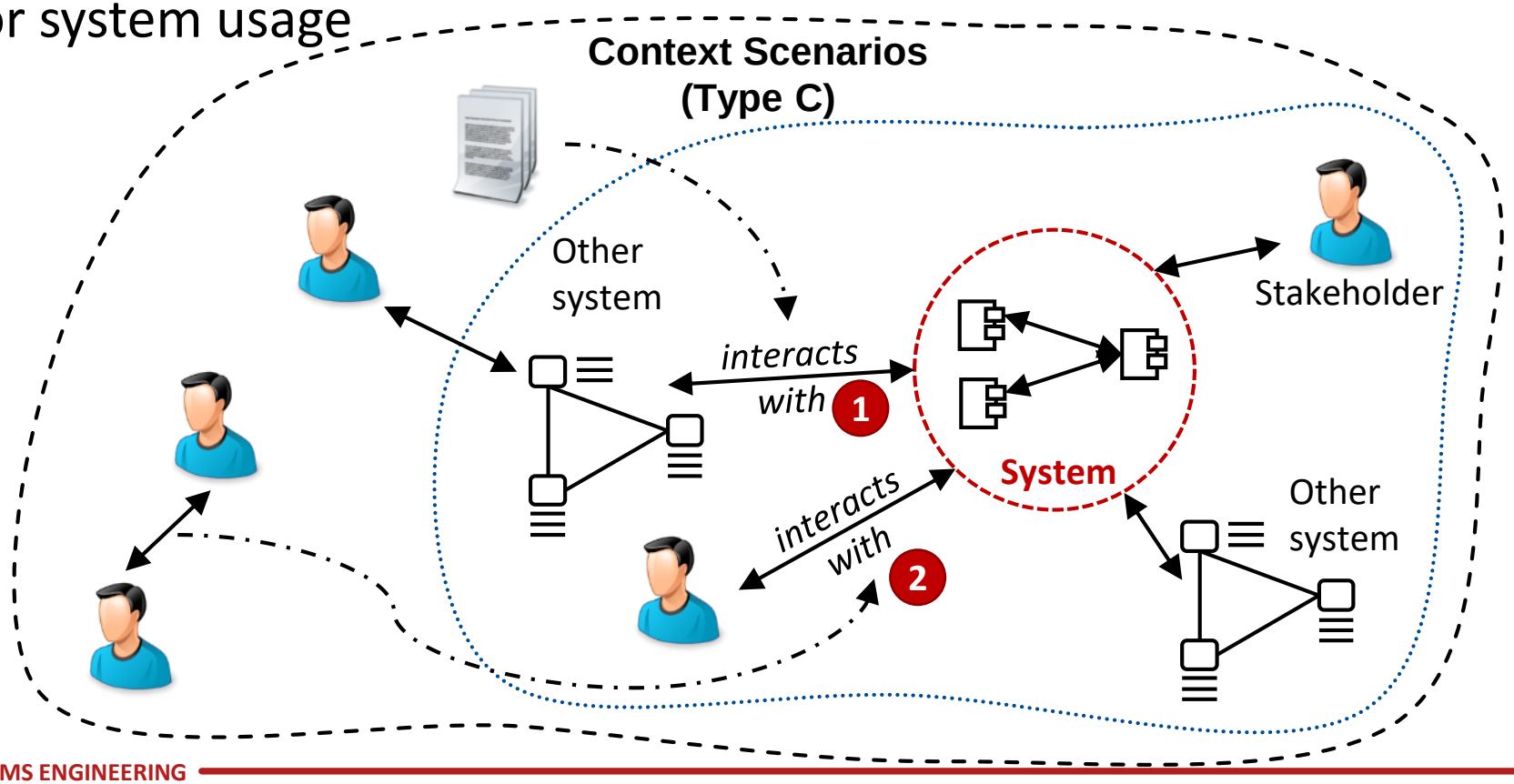
Document sequences **of interactions between the system and its actors** (persons and systems in the context)



System-Internal vs. Interaction vs. Contextual (4)

Contextual scenarios (Type C)

- Additionally to interaction (B) scenarios, contextual (C) scenarios document **interactions between actors/systems in the system's context** when relevant for system usage



System-Internal vs. Interaction vs. Contextual (5)

Contextual Scenario (Type C)

The driver enters a destination not covered by the maps shipped with the navigation system. The system thus displays an error message.

→ Additional context information/assumptions:

- Navigation system is not connected to the Internet to download new maps
- Navigation system has an interface to access external data sources

The driver opens the “route planning” app on his mobile phone. The driver enters the starting point as well as the destination and shortest route as routing parameters. The route planning app calculates the route. It displays the route on the mobile phone. The driver establishes a Bluetooth connection to the navigation system and transfers the map and route data provided by the routing-system. The navigation system updates the map and route and starts routing.

Misuse Scenario

A **misuse** scenario documents a sequence of interaction in which a **hostile actor** uses the system **against the stakeholders' intention**.

- Especially relevant for safety- and security-critical systems

During heavy rain, Carl intentionally cuts in with high speed right behind an another car to test the vehicle's safety system, i.e. to test, how its car performs an emergency braking. Due to the wet conditions and the sudden change of direction caused by cutting in behind the other car, aquaplaning occurs. As a result car is skidding into the crash barriers..

Descriptive vs. Exploratory vs. Explanatory (1)

- **Descriptive** scenarios
  - Are used to **refine goals and requirements** such as processes, interactions or workflows
  - Are typically developed jointly by a **group of stakeholders** who thereby **identify and capture** required **functions**, relevant **events**, or **new stakeholders.**
  - Illustrate the **meaning of goals and requirements**.
  - Used to elicit and document **detail innovative ideas** / innovative solutions.
- **Exploratory** scenarios are used to explore and evaluate possible, alternative solutions.
  - Typically describes a set of **different/alternative realizations** (and their **effects**) to be explored.
  - Support **decision making** (find possible decision criteria, when unknown)
  - Help to make **different viewpoint** of stakeholders explicit.

Descriptive vs. Exploratory vs. Explanatory (2)

- **Explanatory** scenarios justify and explain interactions.
  - They provide **background information** and **rationales** for particular interaction sequences.
  - They can **include alternative views** of different stakeholders.
  - They are used to explain **complex facts** and additional value of the system to uninvolved persons.

4. Benefits of using Scenarios

Elicitation Activity

- Scenarios are a **good** starting point for eliciting requirements.
- Scenarios **foster communication** between the stakeholders: Scenarios **put requirements into context** and include **concrete examples** of system usage.
- Scenarios **refine goals**: By defining scenarios, exiting goals are refined, new (sub-)goals are introduced, and alternative ways for satisfying/dissatisfying goals are identified.
- Scenarios **support the identification** of requirements sources.
- The use of concrete instances (instead of types) is an **indicator for hidden**, not explicit **knowledge**, important for RE.
- Scenarios explain **stakeholder intentions**.
- Scenarios are a good starting point for developing **solution-oriented requirements**.

Negotiation Activity

- Scenarios are **easy to comprehend** and thus support all types of negotiation activities.
- Scenarios help **uncover conflicts**.
- Creating scenarios to **illustrate detected conflicts** helps understand the conflict and in particular its causes.
- Scenarios can be used to **express alternative solutions** in a way that is understandable to all stakeholders.
- Using scenarios in conflict resolutions facilitates **reaching a consensus**.

Documentation Activity

- Scenarios support **structuring** requirements **documents**, e.g., using scenarios to **define views** on a requirements document.
- Scenarios providing **rich context information** for solution-oriented requirements.
- Scenarios **embed** solution-oriented requirements **into a usage context**:

  Each individual requirement can be **traced** to at least one scenario.
- Relating scenarios and glossary terms **puts glossary terms into context** and **facilitates comprehension** of the terms.

Validation Activity

- Scenarios facilitate **stakeholder involvement** in requirements validation (as they are easy to understand).
- Scenarios facilitate the **validation of solution-oriented** requirements by **putting** those **requirements in context**.
- Scenarios support the **identification of irrelevant requirements**:

  If a requirement is not related to a scenario, it might be irrelevant.
- Scenarios facilitate the consideration of **context information** during the validation.
- Scenarios, in combination with **prototyping**, facilitate **the detection of shortcomings** in the current specification.

Management Activity

- Prioritization of scenarios can be used as the basis for **prioritizing solution-oriented requirements**.
- Scenarios **support traceability** and **consistency** between goals and requirements: Scenarios act as bridge between **goals** and **solution-oriented requirements**.
- The context information documented in scenarios, support, among others, **change analysis** and **change management**.
- The definition of **traceability scenarios** facilitates the communication of **required trace recording** to the development team.

Summary

- A scenario describes a concrete example of satisfying or failing to satisfy a goal (or set of goals). A scenario:
  - Provides more detail about one or several goals.
  - Typically defines a sequence of interactions steps executed to satisfy a goal and relates these interaction steps to the system context.
- Several scenario types exist:
  - Current vs. desired state
  - Instance vs. type
  - Internal vs. interaction vs. context
  - Main vs. alternative vs. exception
  - Descriptive vs. exploratory vs. explanatory
- Scenarios support the all RE activities: Documentation, elicitation, negotiation, validation and management

Literature (1)

[Alexander and Maiden 2004] I. Alexander, N. Maiden (Eds.): Scenarios, Stories, Use Cases – Through the system Development Life-Cycle. Wiley, Chichester, 2004.

[Carroll, 1995] John M. Carroll. The scenario perspective on system development. In: John M. Carroll. Scenario-Based Design: Envisioning Work and Technology in System Development. Wiley, New York, 1995. 1-17

[Caroll 2000] J.M. Caroll (Ed.): Making Use – Scenario-Based Design of Human Computer Interactions. MIT Press, Cambridge, 2000.

[Cockburn 2001] A. Cockburn: Writing effective Use Cases. Addison-Wesley, Bosten, 2001.

[Haumer et al. 1999] P. Haumer, P. Heymans, M. Jarke, K. Pohl: Bridging the Gap between Past and Future RE – A Scenario-Based Approach. In: Proceedings of the 4th IEEE International Symposium on Requirements Engineering (RE'99), IEEE computer Society Press, Los Alamitos, 1999, pp. 66-73.

Literature (2)

[Jacobson et al. 1992] I. Jacobson, M. Christerson, P. Jonsson, G. Oevergaard: Object-Oriented Software Engineering – A Use Case Driven Approach. Addison-Wesley, Reading, 1992.

[Kyng,1995] Morton Kyng. Creating Contexts for Design. In: John M. Carroll. Scenario-Based Design: Envisioning Work and Technology in System Development. Wiley, New York, 1995. 85-107

[Pohl 2010] K. Pohl: Requirements Engineering - Fundamentals, Principles and Techniques. 1st edition, Springer, 2010.

[Weidenhaupt et al. 1998] K. Weidenhaupt, K. Pohl, M. Jarke, P. Haumer: Scenario Usage in System Development – A Report on Current Practice. IEEE Software, Vol. 15, No. 2, IEEE Press, Los Alamitos, pp. 22-45.

Literature for Further Reading

[Haumer et al. 1998] P. Haumer, K. Pohl, K. Weidenhaupt: Requirements Elicitation and Validation with Real World Scenes. IEEE Transactions on Software Engineering, Vol. 24, No. 12, 1998, pp. 1036-1054.

[Rolland et al. 1998a] C. Rolland, C. Ben Achour, C. Cauvet, J. Raylt, A. Sutcliffe, N. Maiden, M. Jarke, P. Haumer, K. Pohl, E. Dubois, P. Heymans: A Proposal for a Scenario Classification Framework. Requirements Engineering Journal, Vol. 3, No. 1, Springer, Berlin, Heidelberg, 1998, pp. 23-47.

[Rolland et al. 1998b] C. Rolland, C. Souveyet, C. Ben Achour: Guiding Goal Modelling Using Scenarios. IEEE Transactions on Software Engineering, Vol. 24, No 1., 1998, pp. 1055-1071.

[Sutcliffe et al. 1998] A. Sutcliffe, N. Maiden, S. Minocha, M. Darrel: Supporting scenario-Based Requirements Engineering. IEEE Transactions on Software Engineering, Vol. 24, No. 12, 1998, pp. 1072-1088.

Image References

[1] Licensed by http://www.iconshock.com/

[2] Provided by Microsoft Office

Legend

Definition

Example

UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

Requirements Engineering & Management

Vielen Dank für Ihre Aufmerksamkeit

PALUNO
The Ruhr Institute for Software Technology