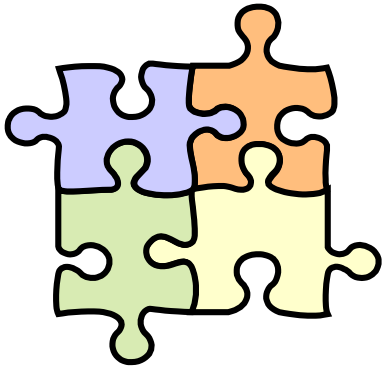
ASE
Applied Software Engineering

TU Graz
Graz University of Technology



Requirements Engineering

- AK Softwaretechnologie 1-

Univ.-Prof. DI Dr. Alexander Felfernig
email: alexander.felfernig@ist.tugraz.at

Institute for Software Technology, Inffeldgasse 16b/2
8010 Graz, Austria

Contact

Graz University of Technology

Lecture

Univ.-Prof. DI Dr. Alexander Felfernig
alexander.felfernig@ist.tugraz.at

Group Decision Tool OpenReq

DI Ralph Samer
rsamer@ist.tugraz.at

Ad Personam

Graz University of Technology

- Professor of Applied Software Engineering at Graz University of Technology since 2009
- Computer Science Studies at University of Klagenfurt
- Engaged in many commercial software projects and two company formations
- Commercial projects in financial services and recommender systems
- Research projects at different levels (European Union, FFG Bridge, FFG Basic, FWF)

Goals of the Course

Graz University of Technology

- Basic Requirements Engineering (RE) Methods
- Recommendation Methods in RE
- Requirements Prioritization
- Decision Support & Decision Biases
- Example Decision Environment (OpenReq)
- Exercises to better understand concepts

Who participates?

Graz University of Technology

Company, position, experiences with requirements engineering (RE).

Agenda

Graz University of Technology

Day	Time	Location	Content
TUE, Oct. 2nd, 2018	-	-	Briefing / Organizational Issues
MI, Nov. 7th, 2018	9:00-18:00	Seminarraum IST (IC02062)	Lecture + Exercises
THU, Jan. 17th, 2019	09:00-13:00	HS FSI 2 (FSEG070)	Invited Talk & Exam

Lecture Topics

Graz University of Technology

Topic	Day	Lecture + Exercises (h)	Time
Introduction/Motivation	WED (07.11.)	0.5 + 0.5	9:00 – 10:00h
Modeling Requirements	WED (07.11.)	0.5 + 0.5	10:00 – 11:00h
Recommender Systems in RE	WED (07.11.)	0.5 + 0.5	11:00 – 12:00h
Requirements Prioritization	WED (07.11.)	0.25 + 0.25	12:30 –13:00h
Variability Modeling	WED (07.11.)	0.5 + 0.5	13:00 – 14:00h
Decision Biases	WED (07.11.)	0.5 + 0.5	14:00 – 15:00h
A Decision Support Environment	WED (07.11.)	0.5 + 0.5	15:00 – 16:00h
Result Presentation (Dec. Support)	WED (07.11.)	1.0	16:00 – 16:45h
Invited Talk & Exam	THU (17.1.)		9:00 – 13:00h

Starting References

Graz University of Technology

1. G. Ninaus, A. Felfernig, M. Stettinger, S. Reiterer, G. Leitner, L. Weninger, and W. Schanil. *IntelliReq: Intelligent Techniques for Software Requirements Engineering*, European Conference on Artificial Intelligence, Prestigious Applications of Intelligent Systems (PAIS), pp. 1161-1166, 2014.
2. A. Felfernig, G. Ninaus, H. Grabner, F. Reinfrank, L. Weninger, D. Pagano, and W. Maalej, *An Overview of Recommender Systems in Requirements Engineering*, Managing Requirements Knowledge Book, Springer Verlag, Berlin Heidelberg, pp. 315-332, 2013.
3. D. Greer and G. Ruhe. *Software release planning: an evolutionary and iterative approach*, Information and Software Technology 46:243-253, 2004.
4. A. Dansh, R. Ahmad, M. Saybani, and A. Tahir. *Companies Approaches in Software Release Planning – Based on Multiple Case Studies*, Journal of Software, 5(2):471-478, 2012.
5. D. Firesmith. *Prioritizing Requirements*, Journal of Object Technology, 3(8):35-47.
6. K. Czarnecki, P. Grünbacher, R. Rabiser, K. Schmid, and A. Wasowski. *Cool Features and Tough Decisions: A Comparison of Variability Modeling Approaches*, Vamos 2012, pp. 173-182, 2012.
7. A. Felfernig, D. Benavides, J. Galindo, and F. Reinfrank. *Towards Anomaly Explanation in Feature Models, Workshop on Configuration*, Vienna, Austria, pp. 117-124, 2013.
8. G. Walia and J. Carver. *A systematic literature review to identify and classify software requirement errors*, Information and Software Technology, 51:1087-1109, 2009.
9. F. Paetsch, A. Eberlein, and F. Maurer. *Requirements Engineering and Agile Software Development*, 12th IEEE Workshops Enabling Technolgoies. pp. 308-313, 2003.

Group Work & Exam

Graz University of Technology

1. Form „groups“ of 1-3 persons and select three papers from the afore mentioned list („starting references“) – store this information (group and selected papers) under https://www.eventhelpr.com/event/YuORoVlyASuaB2Mr/participate
2. Your group will be assigned to **one** (the selected topic) of your three selected papers (this assignment will be performed by the lecturers and you will be informed about your selected topic as soon as this decision has been made). Use this paper (the selected topic) as starting point for a **10-page summary** of related work which includes at least 15 additional references.
3. No group overlaps of topics are possible, i.e., in some cases the topic assignment for the group will differ from the specified preferences
4. Send the **10-page summary** to (strict deadline: Jan. 10th): alexander.felfernig@ist.tugraz.at. The template for the summary can be found here: https://www.eventhelpr.com/event/YuORoVly/public
5. The exam at the end of the course will be related to the lecture topics.

Evaluation

Graz University of Technology

Weighting:

50% 10-page summary (written by group), 50% exam

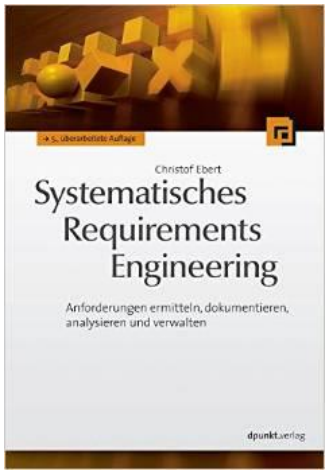
Grades:

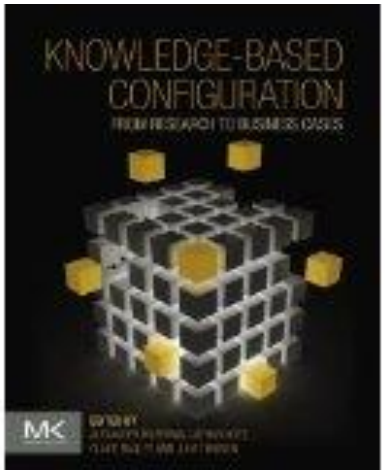
87.51 – 100.0 %:	„sehr gut“
75.01 – 87.5%:	„gut“
62.51 – 75.0:	„befriedigend“
50.0 – 62.50:	„genügend“
< 50.0%:	„nicht genügend“

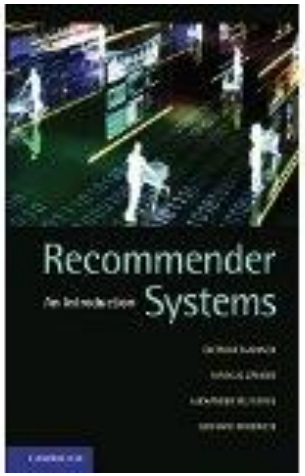
Course Material

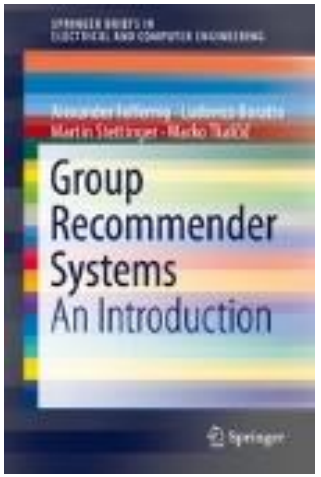
Graz University of Technology

- Lecture Slides, SW Documentation, References
- Books
  - C. Ebert, Systematisches Requirements Engineering, dpunkt, 2014.
  - A. Felfernig, L. Hotz, C. Bagley, and J. Tiihonen, Knowledge-Based Configuration, Morgan Kaufmann, 2014.
  - D. Jannach, M. Zanker, A. Felfernig, and G. Friedrich, Recommender Systems, Cambridge University Press, 2010.
  - A. Felfernig, L. Boratto, M. Stettinger, M. Tkalcic, Group Recommender Systems, Springer, 2018.









Further References

Graz University of Technology

- D. Benavides, S. Segura, A. Cortés: Automated analysis of feature models 20 years later: A literature review. Inf. Syst. 35(6): 615-636, 2010.
- V. Chandola, A. Banerjee, and V. Kumar: Anomaly Detection: A Survey, ACM Computing Surveys, 41(3):15, 2009.
- C. Ebert, Systematisches Requirements Engineering, dpunkt, (2014).
- N. E´en and N. Sörensson: Translating Pseudo-Boolean Constraints into SAT, Journal on Satisfiability, Boolean Modeling and Computation 2:1-25, 2006.
- A. Felfernig, G. Ninaus, H. Grabner, F. Reinfrank, L. Weninger, D. Pagano, and W. Maalej, An overview of recommender systems in requirements engineering, in Managing Requirements Knowledge, pp. 315–332, (2013).
- A. Felfernig and G. Ninaus. Group Recommendation Algorithms for Requirements Prioritization, ICSE 2012 Workshop on Recommender Systems for Software Engineering (RSSE 2012), Zürich, Switzerland, pp. 1–4, (2012).
- A. Felfernig, W. Maalej, M. Mandl, F. Ricci, and M. Schubert, Recommendation and Decision Technologies For Requirements Engineering, ICSE 2010 Workshop on Recommender Systems in Software Engineering, Cape Town, South Africa, pp. 1-5, (2010).
- H. Hofmann and F. Lehner, Requirements Engineering as a success factor in software projects, IEEE Software,18(4):58–66, (2001).

Further References

Graz University of Technology

- D. Leffingwell, Calculating the return on investment from more effective requirements management, American Programmer, 10(4):13–16, (1997).
- S. Lim, D. Quercia, and A. Finkelstein, Stakenet: Using social net- works to analyse the stakeholders of large-scale software projects, 32nd ACM/IEEE Intl. Conf. on Software Eng., pp. 295–304, Cape Town, South Africa, (2010).
- W. Maalej, Z. Kurtanovic, and A. Felfernig, What Stakeholders Need to Know About Requirements, 4th IEEE Intl. Workshop on Empirical Requirements Engineering, Karlskrona, Sweden, (2014).
- G. Ninaus, F. Reinfrank, A. Felfernig, and M. Stettinger, Content-based Recommendation Techniques In Requirements Engineering, 1st Intl. Workshop on AI For Requirements Engineering, Karlskrona, Sweden, (2014a).
- G. Ninaus, A. Felfernig, M. Stettinger, S. Reiterer, G. Leitner, L. Weninger, and W. Schanil. IntelliReq: Intelligent Techniques for Software Requirements Engineering, European Conference on Artificial Intelligence, Prestigious Applications of Intelligent Systems (PAIS), pp. 1161-1166, (2014b).
- T. Simon, J. Streit, and M. Pizka “Practically Relevant Quality Criteria for requirements documents”, itestra GmbH, Ludwigstr. 35, 86916 Kaufering Germany.
- M. Stettinger, A. Felfernig, G. Leitner, S. Reiterer, and M. Jeran. Counteracting Serial Position Effects in the CHOICLA Group Decision Support Environment, 20th ACM Conference on Intelligent User Interfaces (IUI2015), pp. 148-157, Atlanta, Georgia, USA, (2015).
- M. Stettinger, A. Felfernig, G. Leitner, and S. Reiterer. Counteracting Anchoring Effects in Group Decision Making, 23rd Conference on User Modeling, Adaptation, and Personalization (UMAP'15) (UMAP Best Papers 2015), Dublin, Ireland, LNCS 9146, pp. 118-130, 2015.

Graz University of Technology

Requirements Engineering: Motivation

Motivation

Tue zuerst das <u>Notwendige</u>, dann das <u>Mögliche</u>, und plötzlich schaffst du das <u>Unmögliche</u>. Franz von Assisi (1182 - 1226) ☞ Giovanni Bernardone, kath. Heiliger, Stifter des Ordens der Franziskaner.

[engl.: first do the necessary, then the possible, and suddenly you will achieve the impossible]

Motivation

Graz University of Technology

Alice: „Könntest du mir bitte sagen, welchen Weg ich von hier aus nehmen soll?“ [Could you tell me, which way I should choose?]

Katze: „Das hängt vor allem davon ab, wohin du gehen willst.“ [It depends on where you want to go]

Alice: „Ich weiß es nicht …“ [I do not know]

Katze: „Dann ist es egal, wohin du gehst.“ [then it does not matter]

[Dialog from „Alice in Wonderland“, Lewis Carroll, 1865]

☞Requirements Engineering hilft, Ziele zu klären und zu festigen. [Ebert 2014]

[Requirements Engineering helps to identify and consolidate goals]

Motivation

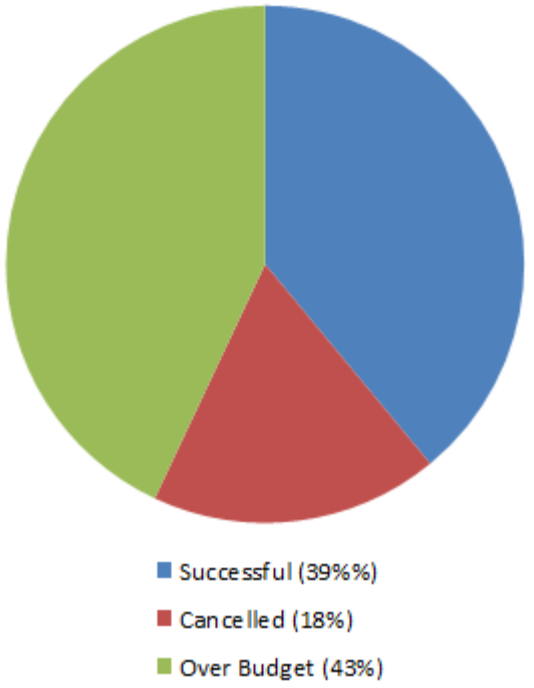
Graz University of Technology

Project: Writing equipment for space missions (standard pens do not work due to missing gravitation). **A** developed the „**Space Pen**“ from metal for a broad temperature scope, **R** applied a pencil… **complex requirements do not need complex solutions**. But: non-functional requirements should be taken into account! [Ebert 2014]

Motivation

Graz University of Technology

- 39% of projects completed successfully.
- 18% were cancelled.
- 43% completed but were over budget.
- Average of efforts dedicated to RE: 2-5%.
- But: erroneous requirements have a worse effect.
- Projects cancelled due to, for example:
  - no clear initial requirements.
  - implicit/missing requirements.
  - faulty/inconsistent requirements.
  - no change management.



[Ebert 2014]

Motivation

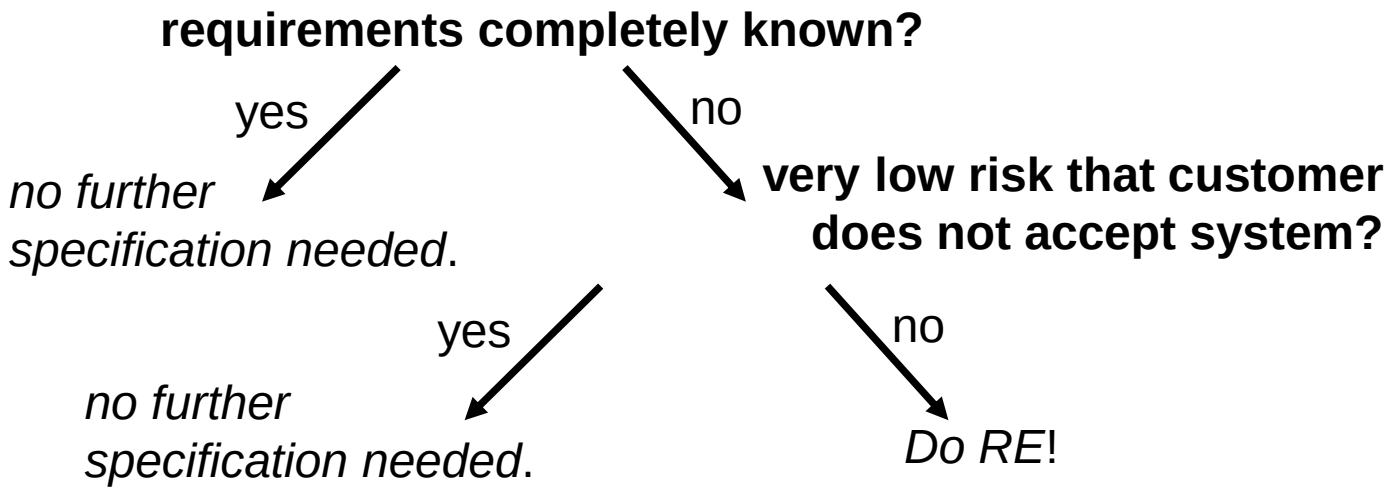
Graz University of Technology

- Requirements Engineering (RE) is one of the most critical phases in software development [Leffingwell 1997].
- Low-quality requirements are a major reason for the failure of a software project [Hofmann and Lehner 2001].
- Existing RE tools focus on definition and cataloguing tasks but fail to provide, for example, quality and dependency information [Maalej et al. 2014].
- Integration of recommendation & decision technologies into RE processes [Felfernig et al. 2010].

Requirements Engineering (RE)?

Graz University of Technology

- Requirements Engineering (RE) is related to the specification and administration of requirements with the overall goal of risk minimization: to not implement a system which is not useful and acceptable for the customer.



Early Requirements Engineering

Graz University of Technology

- Early requirements engineering is concerned with the analysis of the operational environment where a software system will eventually function.

A. Fuxman and L. Liu and M. Pistore and M. Roveri and J. Mylopoulos Specifying and Analyzing Early Requirements: Some Experimental Results. RE Conference 2003, IEEE, pp. 105—14, 2003.

- Early requirements engineering should involve identification of the differences and similarities that exist among the mental models of the stakeholders.

B. Burgemeestre and J. Liu and J. Hulstijn and Y. Tan. Early Requriements Engineering for e-Customs Decision Support: Assessing Overlap in Mental Models, CAISE 2009, pp. 31—36, 2009.

When/Why RE?

Graz University of Technology

- Problematic approaches, for example …
  - “not needed”
  - “we do not have the time”
  - “it is expensive”
- Better approach …
  - What do we have to do in order to reduce the risk of failure to an acceptable degree?
  - The overall RE efforts should be inverse to the degree of the willingness to take risks!

Indirect Savings

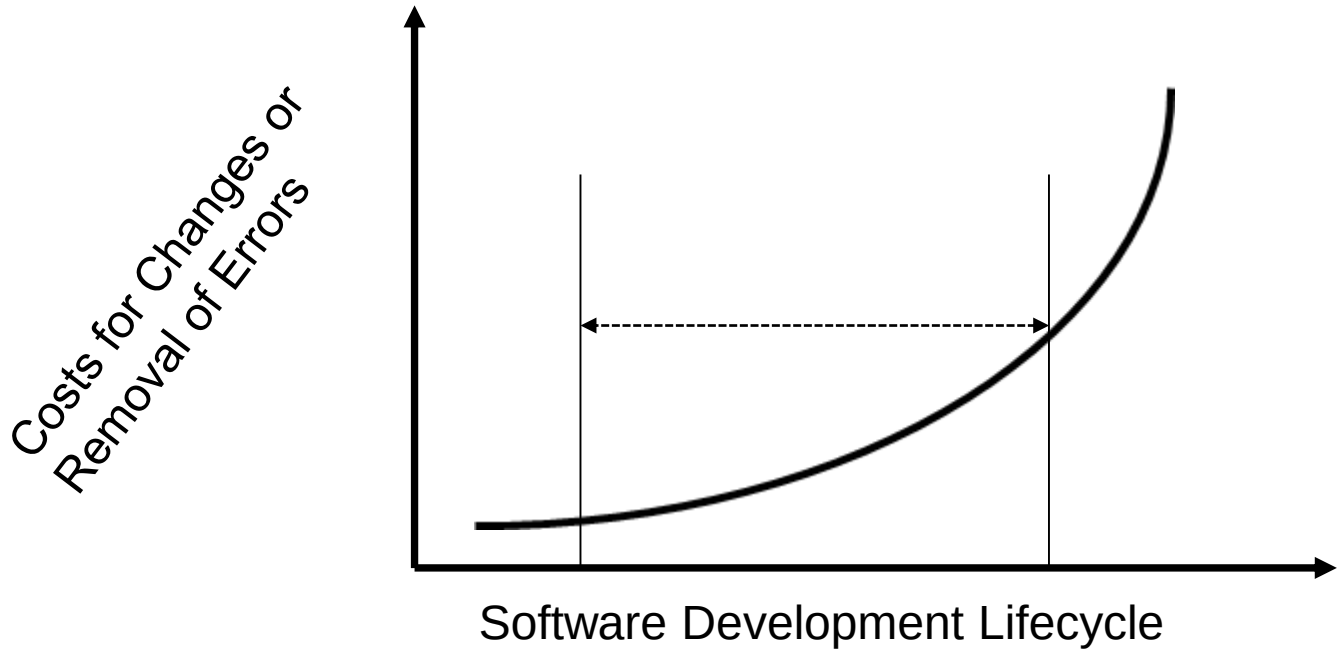
Graz University of Technology

- **Value orientation**: reduced development and maintenance costs (only relevant functions).
- **Quality**: reduced costs for correcting faulty software (due to low-quality RE).
- **Lower costs**: directly related to RE (double efforts ☞ -40% life cycle costs).

[Ebert 2014]

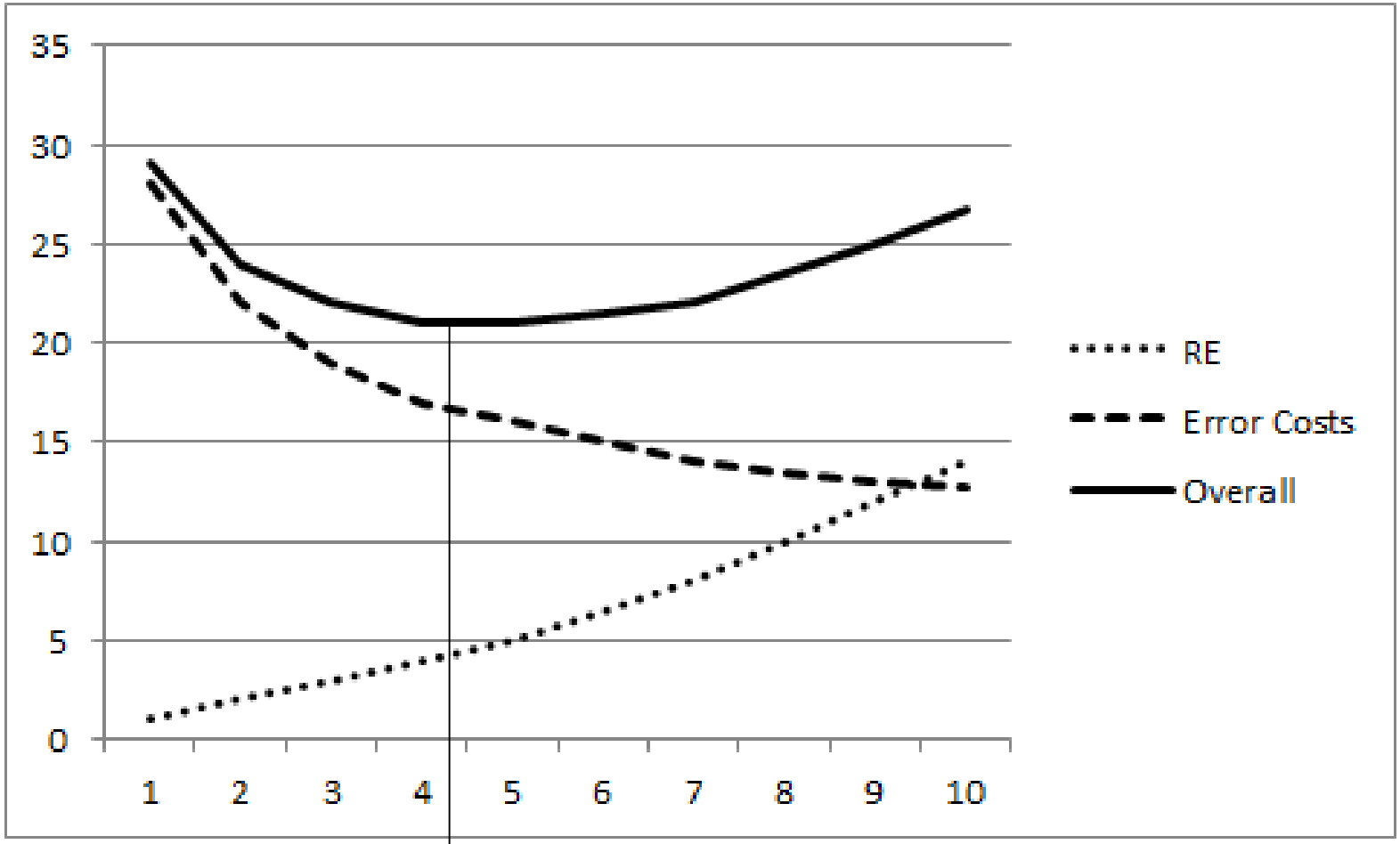
Development Costs

Graz University of Technology



RE Investments

Graz University of Technology



Point of optimal profitability

Exercises (1)

Graz University of Technology

Selbsttest. Ergebnis ist ein Stärken/Schwächenprofil für Ihre Projekte (= Startpunkt für Verbesserungen). Nehmen Sie aktuelle (relevante) Projekte für Ihre Antworten. Antworten beziehen sich auf die aktuelle Situation und nicht den Sollzustand. Beantworten Sie Fragen aus der Sicht des Projektmanagers. Die Fragen im Test sollten aufgrund der folgenden Punkteskala bewertet werden:

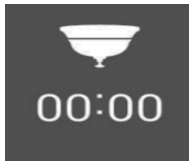
- 3 Punkte: Ja, vollständig, da bin ich mir sicher.
- 2 Punkte: Davon gehe ich aus und habe es bereits gesehen.
- 1 Punkt: Fragwürdig, das kann ich mir nicht richtig vorstellen.
- 0 Punkte: Nein, gar nicht.

[Ebert 2014]

Graz University of Technology

Modeling Requirements

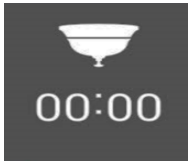
Conference Timer: Business Case



Graz University of Technology

- Functionality not included in competitor products.
- Needed in most of existing conference formats.
- Conference organizers articulate a strong need (less administrative overheads).
- Features help to improve the punctuality of speakers.

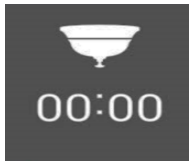
Conference Timer: Glossary



Graz University of Technology

ID	Term	Description
1	time	current local time displayed
2	schedule	assignment of conference paper presentations to time slots
3	conference	the conference
4	session	represents a presentation unit for a set of papers
5	speaker	presenter of a paper in a session but also an invited speaker
6	session chair	responsible for organizing a session
…	…	…

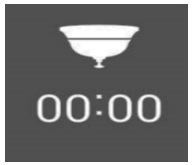
Conference Timer: Requirements List



Graz University of Technology

ID	Short Description
1	Show the current time
2	Set the time
3	Stopwatch
4	Define Moods
5	Timer: Define Schedule
6	Timer: Execute Schedule
7	Timer: Notify Speaker
8	Timer: Announce Next Speaker

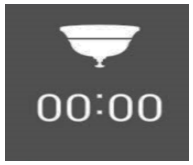
Analysis of Requirements



Graz University of Technology

ID	Short Description	Priority	Effort	Risk
1	Show the current time	1	low	low
2	Set the time	1	low	low
3	Stopwatch	2	medium	low
4	Define Moods	2	medium	medium
5	Timer: Define Schedule	1	high	medium
6	Timer: Execute Schedule	1	medium	medium
7	Timer: Notify Speaker	2	medium	low
8	Timer: Announce Next Speaker	2	medium	low

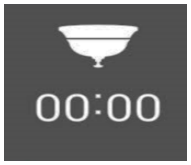
Release Planning



Graz University of Technology

ID	Short Description	Priority	Effort	Risk	Release
1	Show the current time	1	low	low	1.0
2	Set the time	1	low	low	1.0
3	Stopwatch	2	medium	low	1.0
4	Define Moods	2	medium	medium	2.0
5	Timer: Define Schedule	1	high	medium	1.0
6	Timer: Execute Schedule	1	medium	medium	1.0
7	Timer: Notify Speaker	2	medium	low	2.0
8	Timer: Announce Next Speaker	2	medium	low	2.0

Individual Requirement



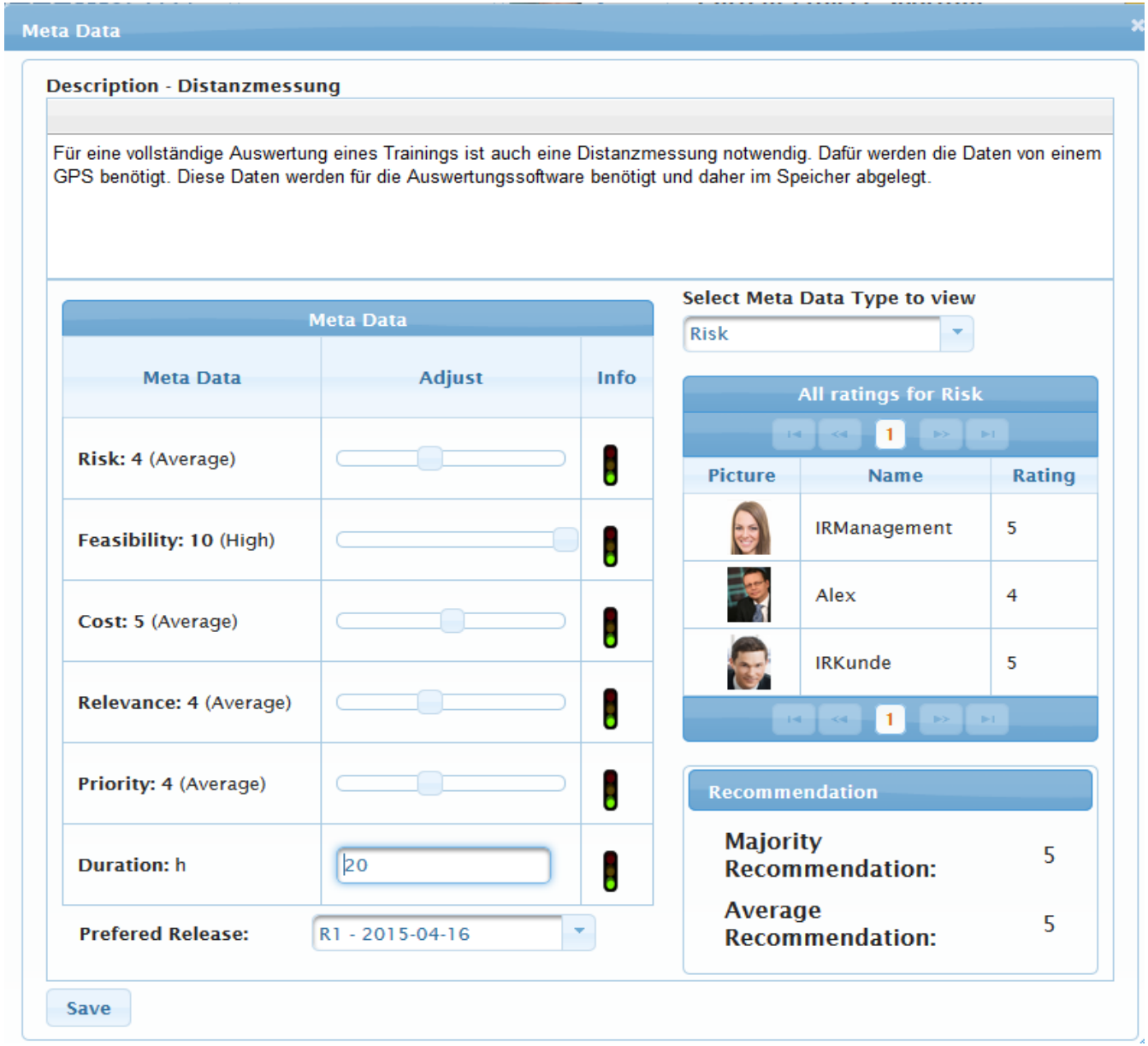
Graz University of Technology

ID	3	Priority	Effort	Risk	Release
Short Description	Stopwatch	2	medium	low	1.0
Description	The stopwatch of the conference timer can be activated via the main menu. After being activated, the watch is automatically initialized with 00:00:00:00 (hours:minutes:seconds:ms). The timekeeping can be activated (start button) and stopped (stop button). Pressing the start button again, continues timekeeping. Pressing the reset button starts a new session.				

UML Use Case: specific type of requirement description ☞ also includes corresponding scenario descriptions (examples of interacting with the user interface and corresponding user interface „mockups“.)

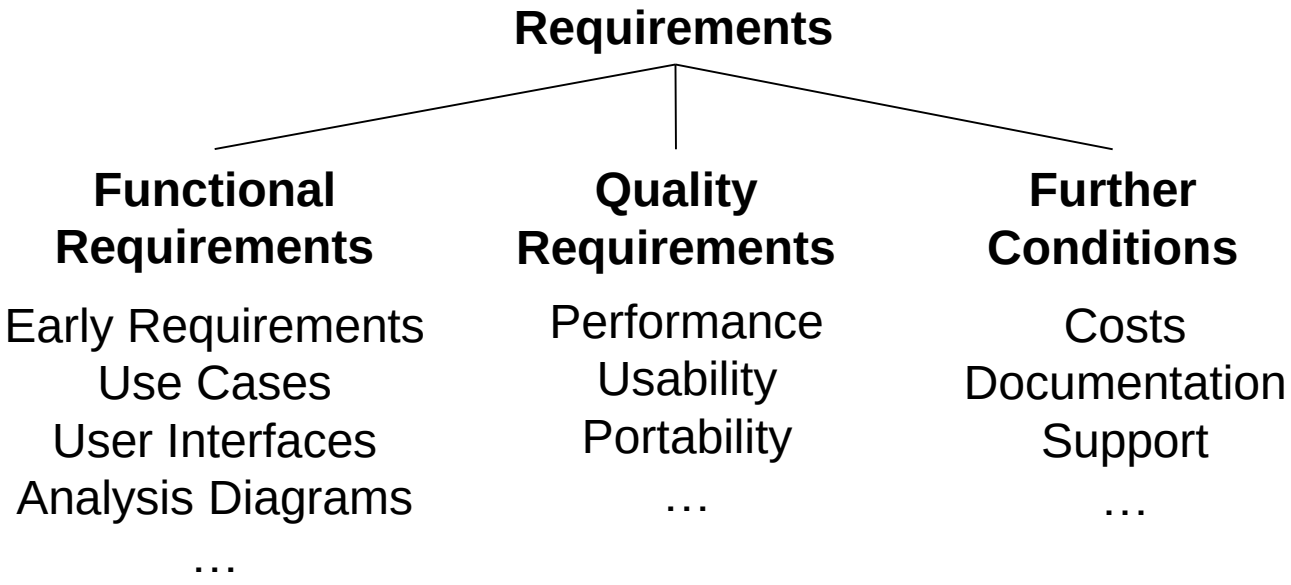
Quality Criteria: completeness, consistency, no trivialities, information “out of scope”, solution thinking, “complicating” details. [Simon et al.]

Requirement in INTELLIREQ



Requirements

A requirement describes what a customer or user expects from a product in terms of conditions, properties, goals, benefits.



[Ebert 2014]

Use Cases

Graz University of Technology

use cases are exploited **to scope the functionality** of the system

they define **which functions will be included**

on a level of granularity s.t. we can be sure that **we did not miss important aspects**

elements of a use case model: actors, use cases (and their relationships ☞ **Use Case Diagram**), documents, prototype screens

Use Cases

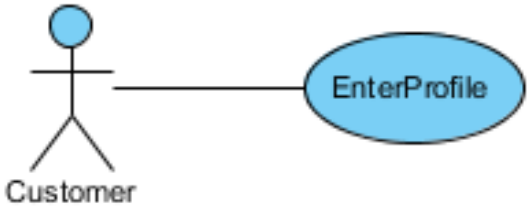
Graz University of Technology

use case

a “**procedure**” by which an actor uses the system

collection of scenarios

activated by an **actor**



actor

outside the system under consideration

linked to some use case(s)

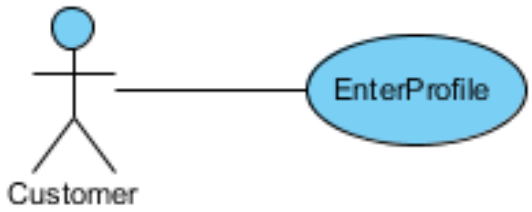
person, **system**, other external **entities**

activates the use case

Example Use Cases

Graz University of Technology

Enter Profile



Enter New Order

Apply For Course

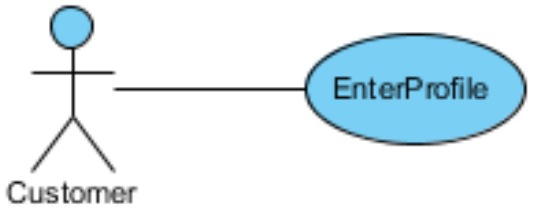
Enter New Course

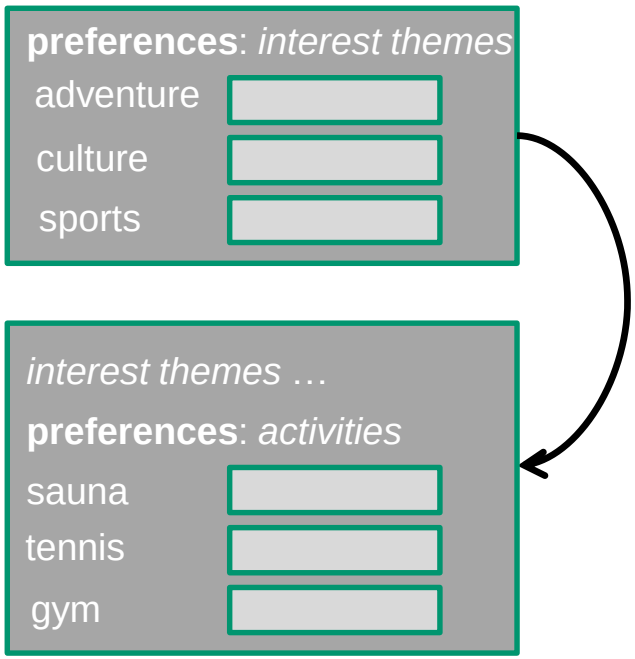
…

One Scenario of "Enter Profile"

Graz University of Technology

1. the **user** activates "EnterProfile"
2. the **system** asks for preferences regarding interest themes
3. the **user** enters his/her preferences
4. the **system** displays a summary of the user input
5. the **system** asks for preferences regarding activities
6. the **user** enters his/her preferences
7. the **system** displays a complete summary of the user input
8. the **user** confirms the preferences



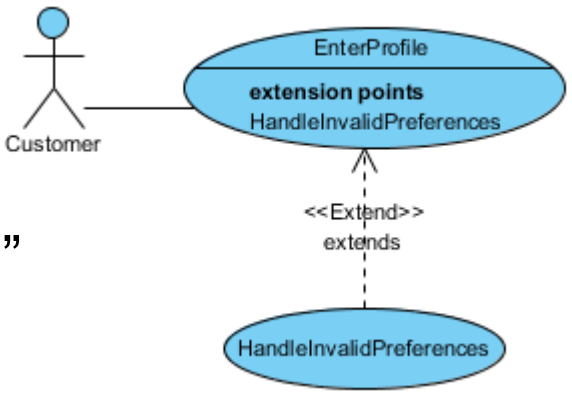


Relationships Between Use Cases

Graz University of Technology

- **Generalization**: variation on normal behavior
  [e.g., *Order Via Internet/Phone* →**spezialisation** *Order*]
- **Include**: common behavior (reuse)
  [e.g., *Book Hotel* →**includes** *Login*]
- **Extend**: additional actions performed
  [e.g., *Handle Invalid Preferences* →**extends** *Enter Profile*]

1. if at **3.** or **6.** the sum of the user preferences is not 100%
2. the system displays the message "sum(preferences) must be 100%"
3. the use case restarts with 3. (respectively 6.)



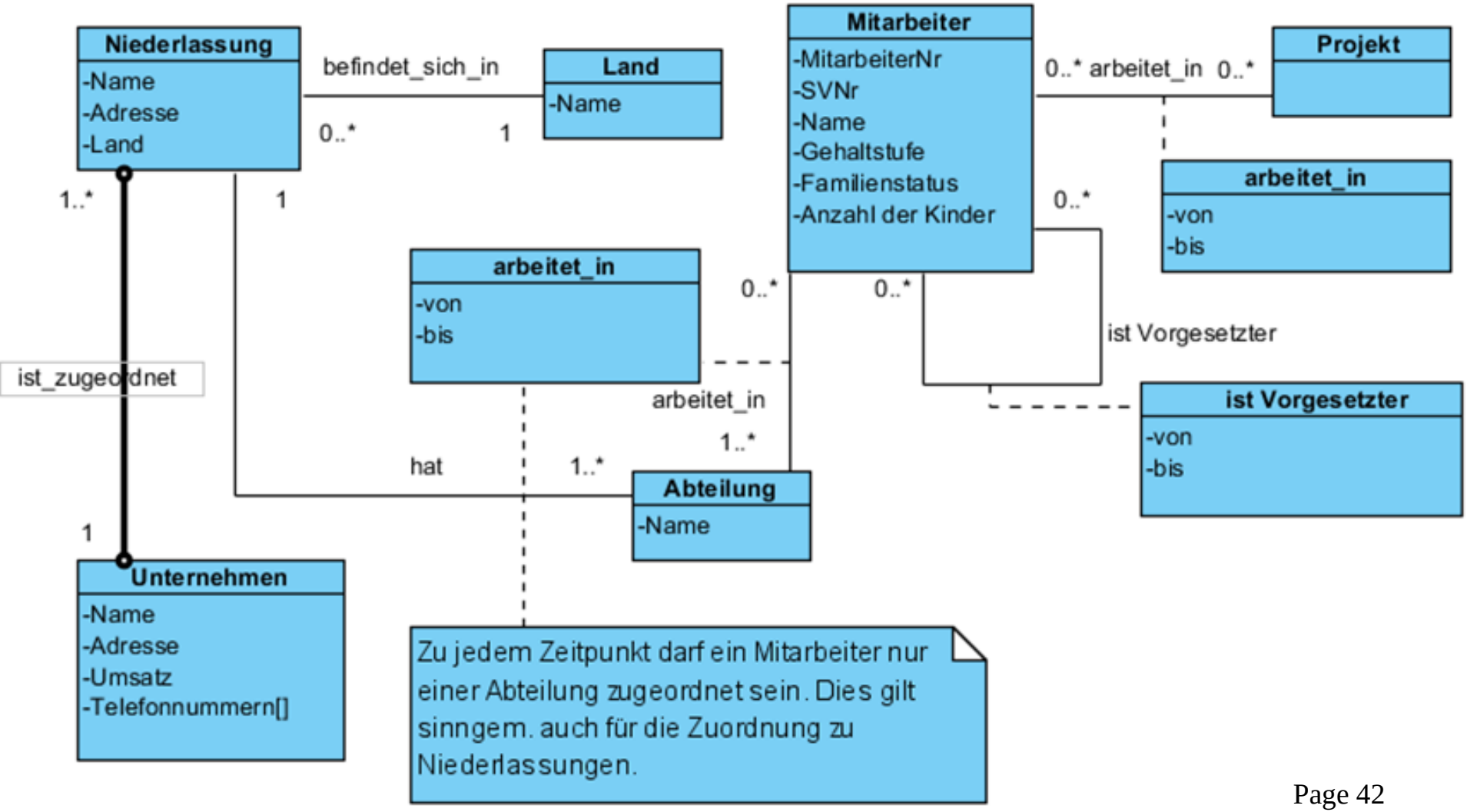
Domain Description

Graz University of Technology

Ein Unternehmen hat einen Namen, eine Adresse, und einen durchschnittlichen jährlichen Umsatz. Weiters hat ein Unternehmen mehrere Telefonnummern unter denen die Hotline erreichbar ist. In einem Unternehmen sind Mitarbeiter beschäftigt, die durch die Mitarbeiternummer, Sozialversicherungsnummer, Namen, Gehaltsstufe, Familienstatus und Anzahl Kinder charakterisiert sind. Jeder Mitarbeiter ist genau einer Abteilung zugeordnet. Mitarbeiter können Vorgesetzte von anderen Mitarbeitern sein, andererseits können Mitarbeiter auch Vorgesetzte haben. Mitarbeiter arbeiten in Projekten, wobei das von/bis Datum der Projektmitarbeit für die interne Zeiterfassung von Relevanz ist. Unternehmen haben verschiedene Niederlassungen und ein Mitarbeiter ist immer genau einer Niederlassung zugeordnet (kann zu unterschiedlichen Zeitpunkten aber auch unterschiedlichen Niederlassungen zugeordnet sein). Niederlassungen werden durch ihren Namen, ihre Adresse und das Land charakterisiert in dem sie sich befinden.

UML Class Diagram

Graz University of Technology



Requirements Document: Basic Elements

Graz University of Technology

- Business Case
- Glossary
- Requirements
- UML Class Diagram (Analysis)

Exercises (2)

Graz University of Technology

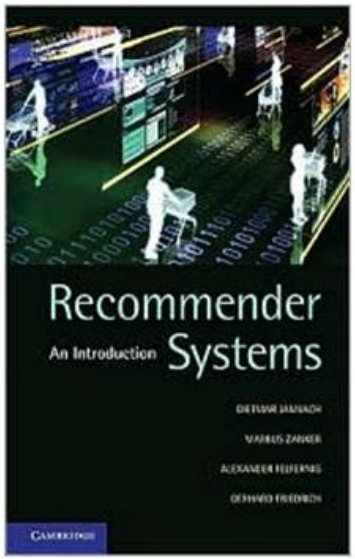
- **Group formation**: currently, N participants are registered for the course. Form teams, where each team should include >= 2 persons.
- **Working Example**: think about alternative domains to develop a software for (simple domain ~ 15 requirements): propose four domains (e.g., conference timer, computer watch, …), **no decision should be made**!

Graz University of Technology

Recommender Systems in Requirements Engineering

Recommender Systems

Graz University of Technology



Recommender Systems: An Introduction

by Dietmar Jannach, Markus Zanker, Alexander Felfernig, Gerhard Friedrich

AVERAGE CUSTOMER RATING:

(Be the first to review)

Gefällt mir Registrieren, um sehen zu können, was deinen Freunden gefällt.

FORMAT:
Hardcover

NOOKbook (eBook) - not available

Tell the publisher you want this in NOOKbook format

NEW FROM BN.COM

~~$65.00~~ List Price

$52.00 Online Price

(You Save 20%)

Add to Cart

NEW & USED FROM OUR

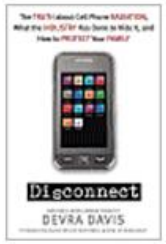
New starting at **$56.46** (You S

Used starting at **$51.98** (You S

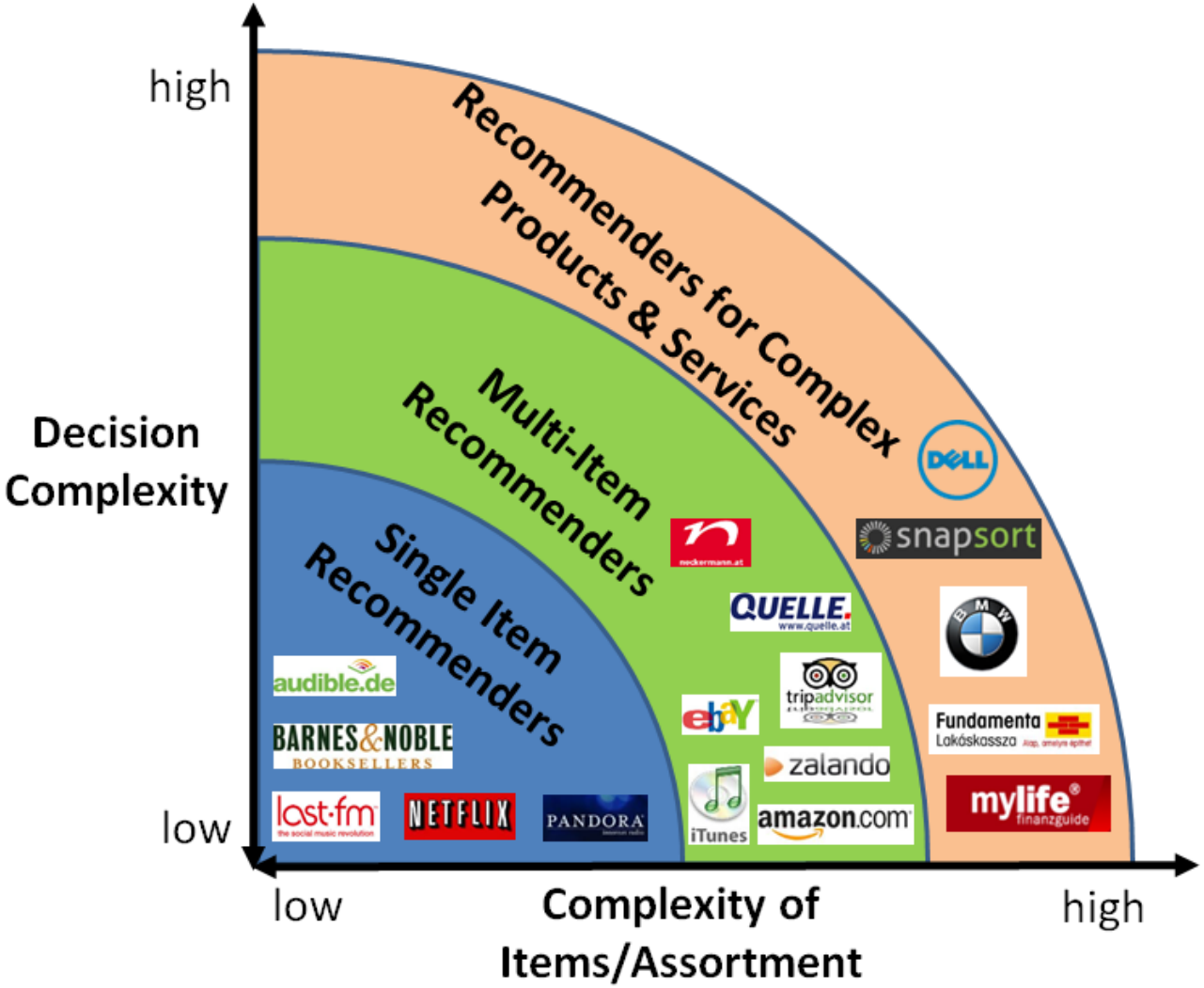
See All Prices

Table of Contents

Customers who bought this also bought

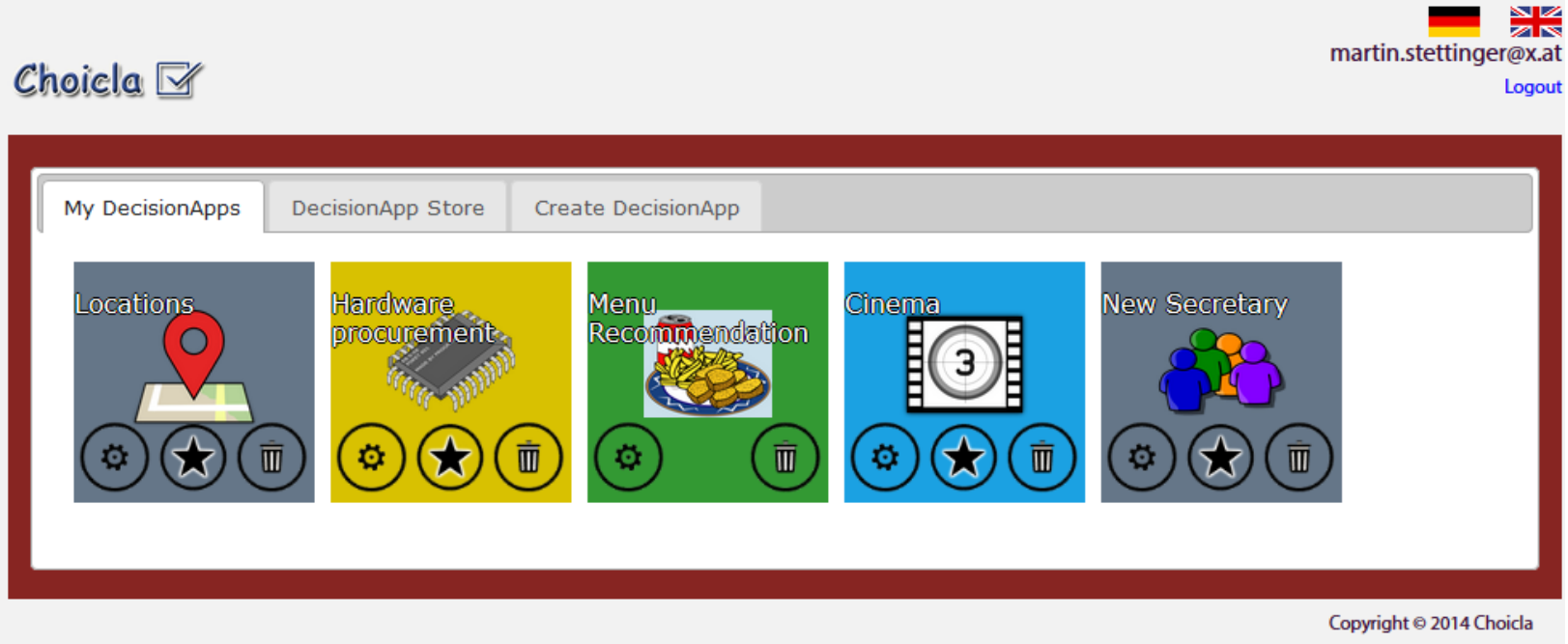


Recommender Systems



CHOICLA

Graz University of Technology



- Decision about new employees, idea selection, prioritizing requirements, …
- Modeling environment for decision apps

INTELLIREQ

FFG Project INTELLIREQ

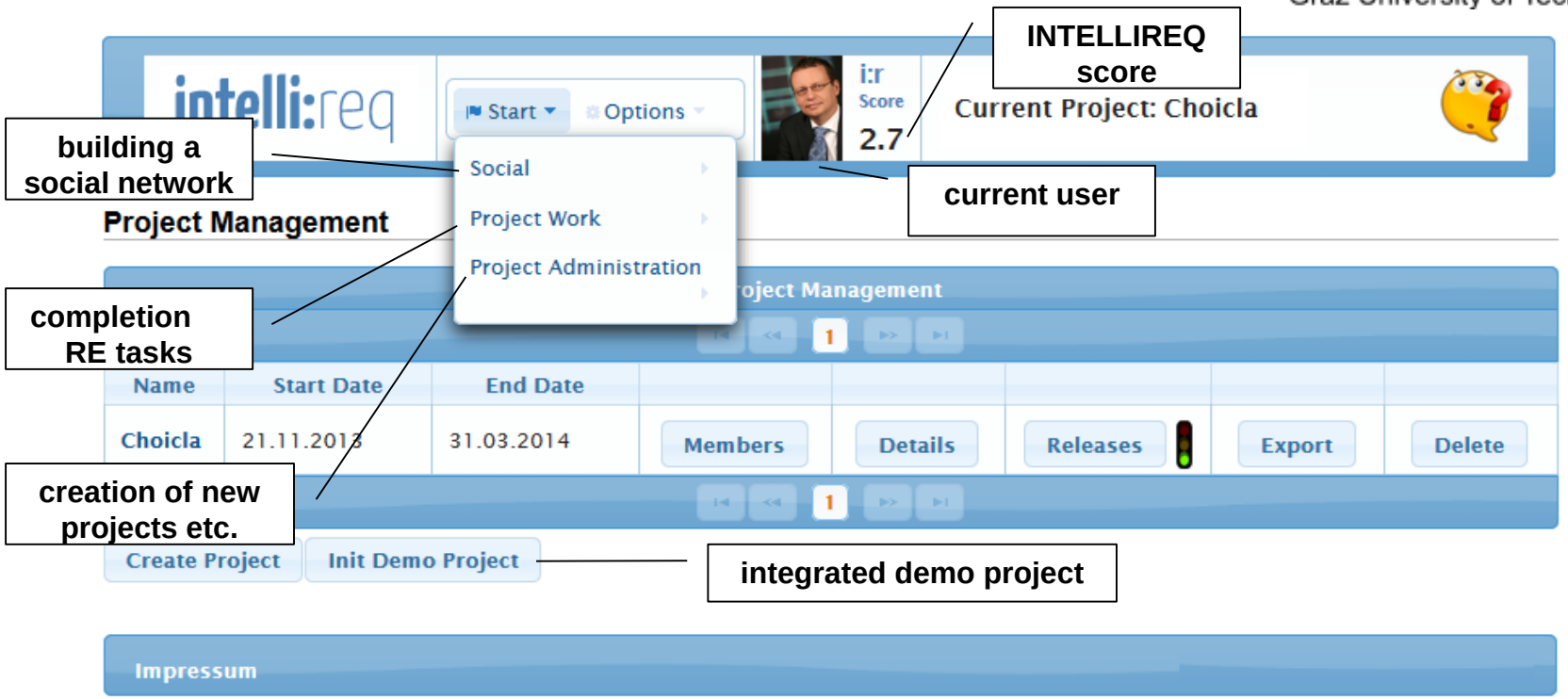
Recommender Technologies

- Group Recommendation
- Content-based Filtering

Application Scenarios

- Requirements Reuse
- Group Decision Making
- Automated Dependency Detection

INTELLIREQ



INTELLIREQ users:

- research partners of TU Graz
- students @ TU Graz

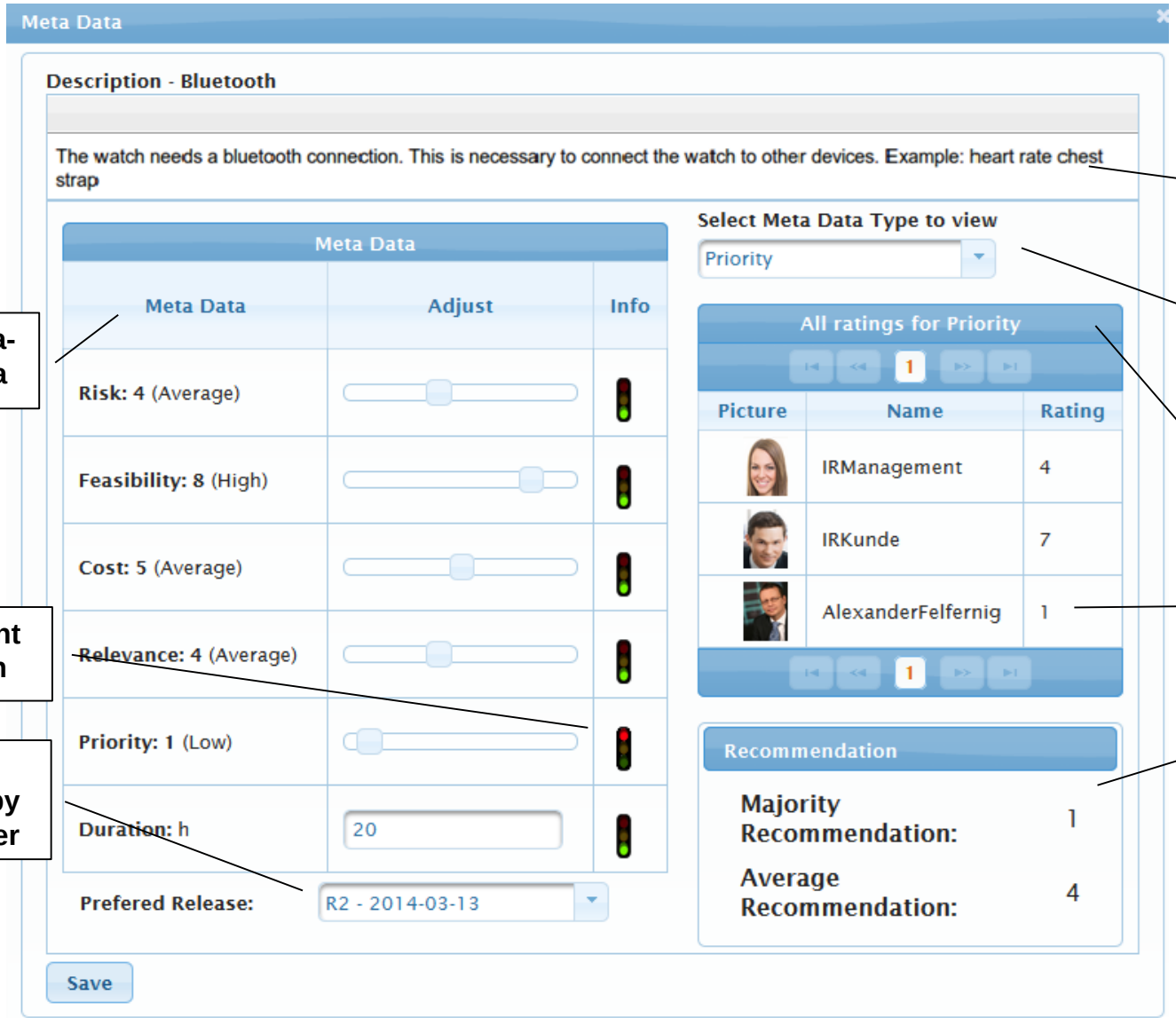
Example Scenarios

Graz University of Technology

- **Group-based evaluation** of requirements (w.r.t. metadata such as cost, duration, and priority).
- **Quality assurance** (e.g., detecting and resolving inconsistent evaluations).
- **Identification of dependencies** (e.g., A incompatible with B, A requires B).
- **Planning of releases** (assignment of requirements to releases).
- **Understanding** requirements (e.g., other users took a look at A after having analyzed B ☞ future work).

Requirements Definition

Graz University of Technology



meta-data

inconsistent evaluation

release preferred by current user

textual description

selected property

property evaluation by stakeholders

current user (stakeholder)

group recommendation

Group Decision Heuristics

Graz University of Technology

$$MAJ(s) = maxarg_{(d \in \{1..5\})}(\#(\bigcup_{u \in Users} eval(u,s) = d))$$

Majority Voting (MAJ) determines the majority of votings (d) for a specific solution s where eval(u,s) denotes the rating for solution s defined by user u.

$$GDIS(s) = minarg_{(d \in \{1..5\})}(\sum_{u \in Users} |eval(u,s) - d|)$$

Group Distance (GDIS) returns the value d as group recommendation which causes the lowest overall change of the individual user preferences.

Combination of individual group heuristics possible [Felfernig and Ninaus 2012].

Dependency Detection

Graz University of Technology

$$sim(r_a, r_b) = \frac{2 * |keywords(r_a) \cap keywords(r_b)|}{|keywords(r_a)| + |keywords(r_b)|}$$

Content-based similarity = predicted strength of dependency between requirements A and B (see also [Ninaus et al. 2014b]**). (Dice coeff.)**

Evaluation Results

Graz University of Technology

- Content-based recommendations alleviate search of reusable requirements.
- Recommendations increase amount of information exchange and discussions.
- Traffic lights trigger „need for completion“ and reduce quality assurance related time efforts.
- Dependency recommendations reduce time efforts related to dependency management.

[Ninaus et al. 2014b]

Literature Overview

[Felfernig et al. 2013]

Graz University of Technology

Activity	Scenario S_i	Recommendation Approach	
elicitation & definition	recommending stakeholders (S_1)	social network analysis	Lim et al. [32]
		content-based filtering	Castro et al. [8] Cleland-Huang et al. [40]
	recommending requirements (S_2)	content-based filtering	Dumitru et al. [12]
		social network analysis	Lim and Finkelstein [31]
		collaborative filtering	Castro et al. [8] Cleland-Huang et al. [40]
quality assurance	managing feature requests (S_3)	clustering	Cleland-Huang et al. [9]
		machine learning	Fitzgerald et al. [21]
	consistency management (S_4)	knowledge-based recommendation	Felfernig et al. [19]
	dependency detection (S_5)	clustering	Cleland-Huang et al. [9]
negotiation & planning	triage (S_6)	clustering utility theory	Duan et al. [11]
	release planning (S_7)	group recommendation utility theory	Felfernig et al. [20]

Content-based Filtering

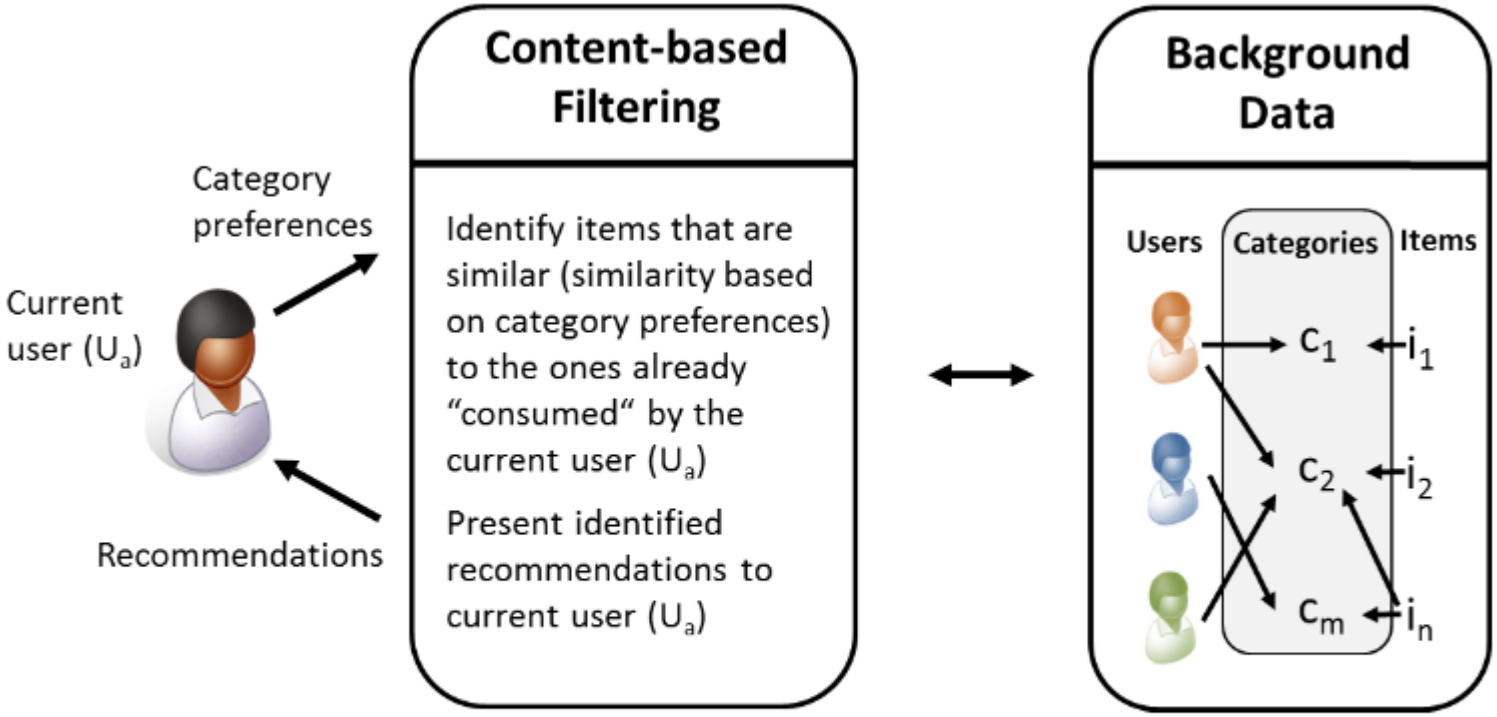


Fig. 2: Content-based filtering (CBF) dataflow: users are rating items and receive recommendations of items similar to those that have received a good evaluation from the current user (U_a).

Ken Follet („Eye of the Needle“) ☞ *Ken Follet („The Pillars of the Earth“)*

Example: Content-based Filtering

Graz University of Technology

requirement	category	planned release	efforts (person days)	description
r_1	database	1	150	store component configuration in DB
r_2	user interface	2	60	user interface with online help available
r_3	database	1	300	separate tier for DB independence
r_4	user interface	1	30	user interface with corporate identity

requirement	extracted keywords
r_1	store component configuration DB
r_2	user interface help
r_3	tier DB independence
r_4	user interface corporate

$$sim(s,r) = \frac{|keywords(s) \cap keywords(r)|}{|keywords(s) \cup keywords(r)|}$$

r_i: requirements
s: requirement (or user profile)

Collaborative Filtering

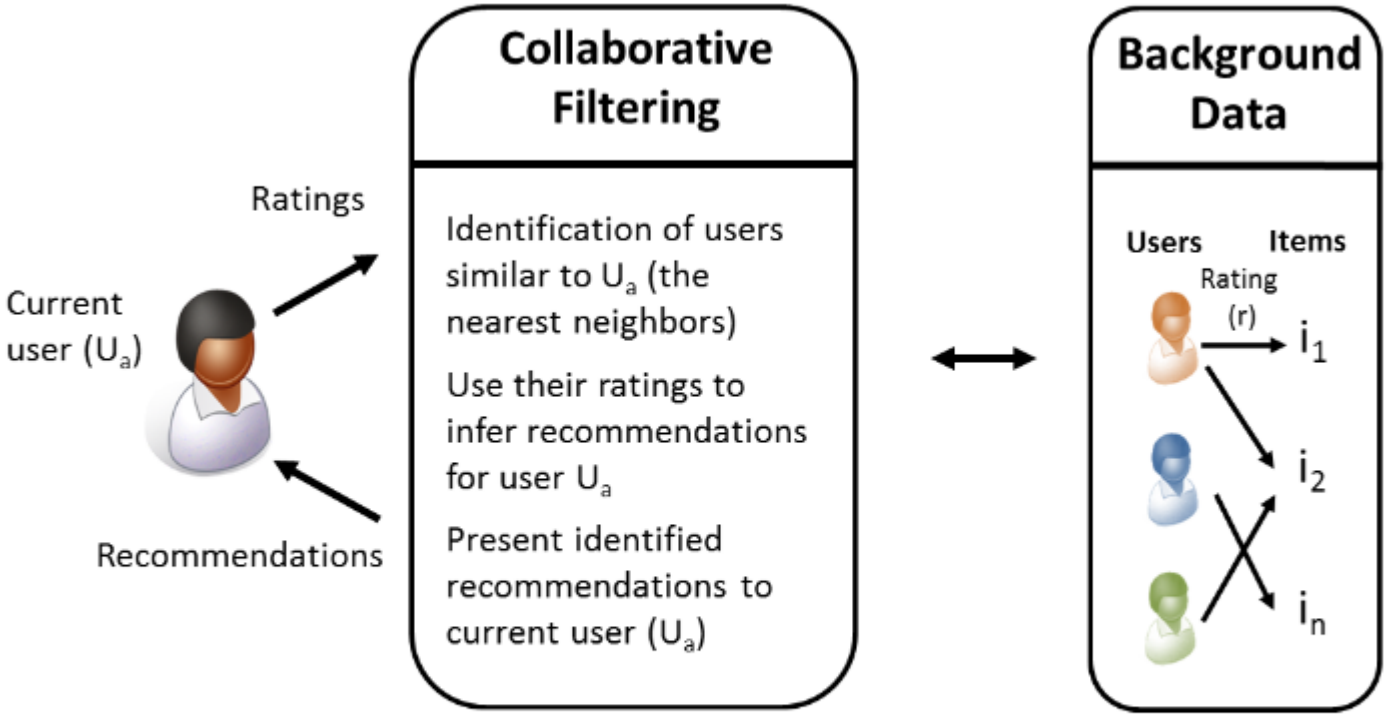


Fig. 1: Collaborative filtering (CF) dataflow: users are rating items and receive recommendations for items based on the ratings of users with a similar rating behavior – the nearest neighbors (NN).

(„New Zealand Travel Guide“) ☞ („Lord of the Rings Blu-ray“)

Example: Collaborative Filtering

Graz University of Technology

	r_1	r_2	r_3	r_4
s_1	1	0	1	0
s_2	1	0	1	1
s_3	1	1	0	1

r_i: requirements
s_i: stakeholder (user)
{1,0}: did analyze (or not)

Example:
current stakeholder (user): s_c
s_c analyzed: r_1 and r_2 (r_1=1,r_2=1)
nearest neighbor(s_c): s_3
recommendation: r_4

Knowledge-based Recommendation

Graz University of Technology

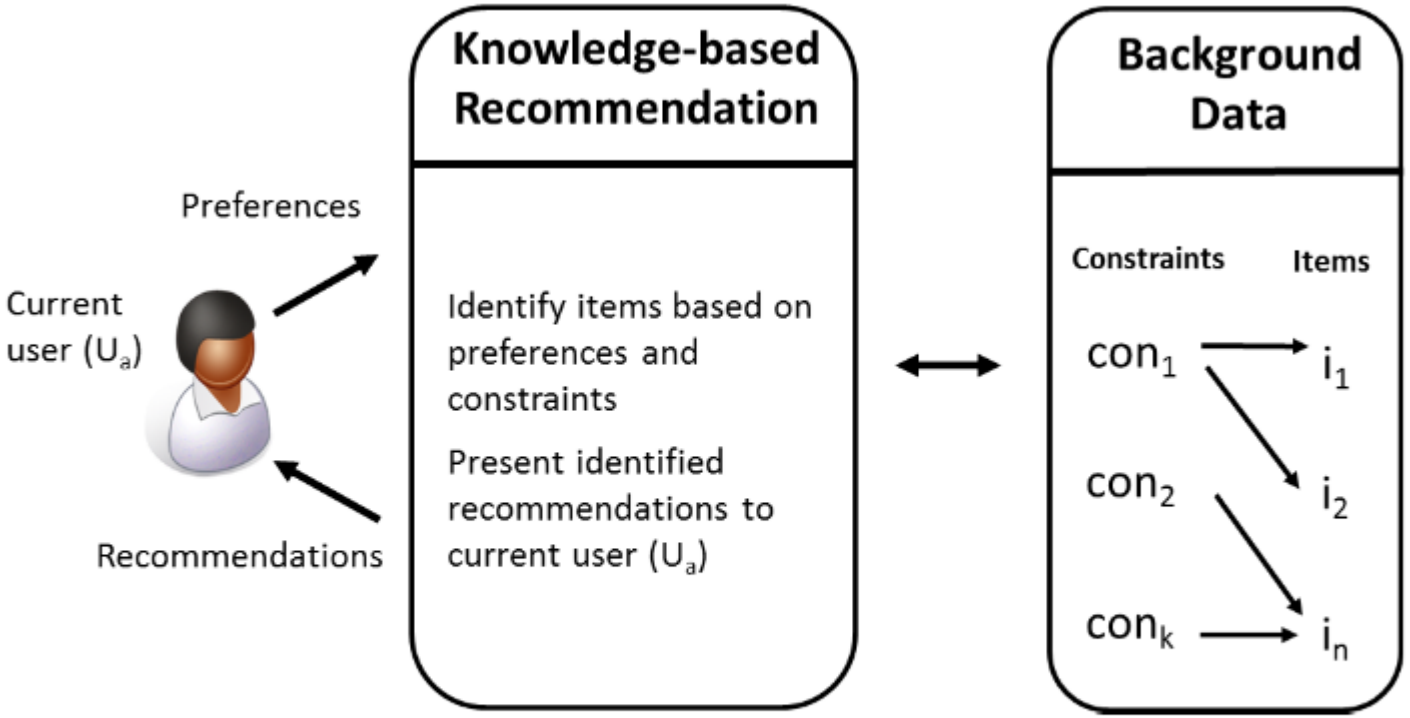


Fig. 3: Knowledge-based recommendation (KBR) dataflow: users are entering their preferences and receive recommendations based on the interpretation of a set of rules (constraints).

Constraint: Very simple domain ☞ choose „Waterfall Model“.

Example: Group Recommendation

Graz University of Technology

requirement: r	s_1	s_2	s_3	s_4
quality	medium	medium	medium	high
effort (person days)	10	7	14	8
decision	accept	revision	accept	accept

r: requirement
s_i: stakeholder

Majority voting: (quality:medium, effort:>=10, accept)

Exercises (3)

Graz University of Technology

- Provide an example of the application of each of the following algorithms in the Requirements Engineering context (also sketch the determination of recommendations):
  - Content-based Filtering
  - Collaborative Filtering
  - Group Recommendation

Graz University of Technology

Requirements Prioritization

Motivation

Graz University of Technology

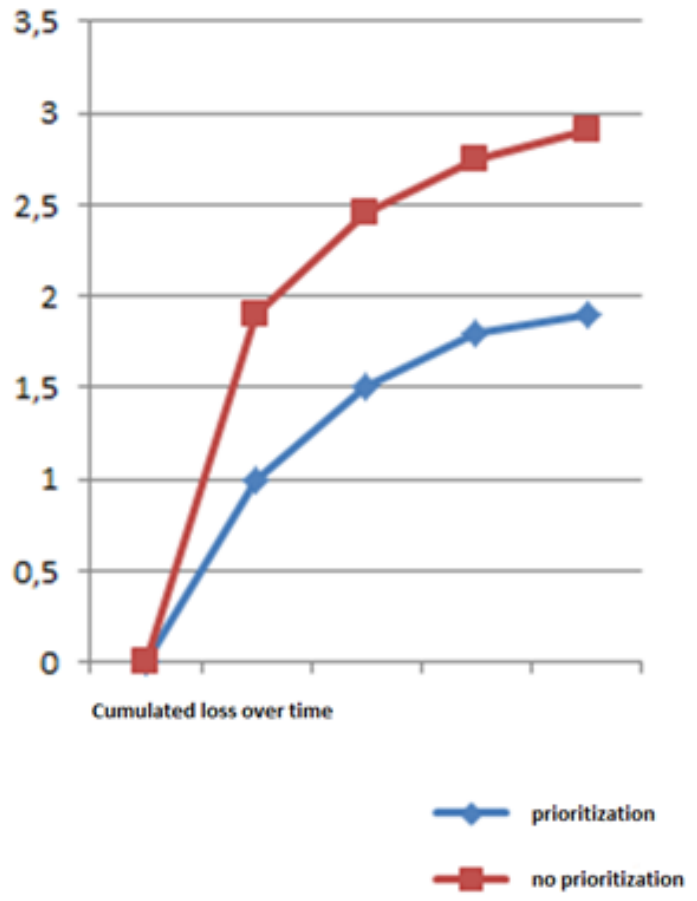
- “Hundreds of requirements”
- “Implement everything!”
- But: limited resources
  - budget, personnel, time

- Which requirements should be implemented?
  - stakeholders have different opinions
  - requirements have to be prioritized

Why Prioritization?

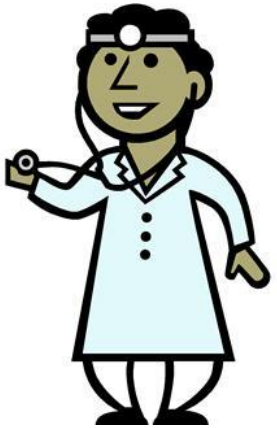
Graz University of Technology

- Focusing Resources √
- More efficient „going live“ √
- Increased integration of stakeholders √
- Key requirements have highest priority √
- Lower risk of project failure due to focus on the most important requirements √



Prioritization Principle

Graz University of Technology



- “Triage” in desasters
  - No survival (-)
  - Survival does not depend on treatment (~)
  - Survival only with treatment (+)
- “Requirements Triage”
  - Not in the next release (-)
  - Optional for the next release (~)
  - „Must“ for the next release (+)

„Open Prioritization“

Graz University of Technology

Prioritization decisions by …

customer + developers + management

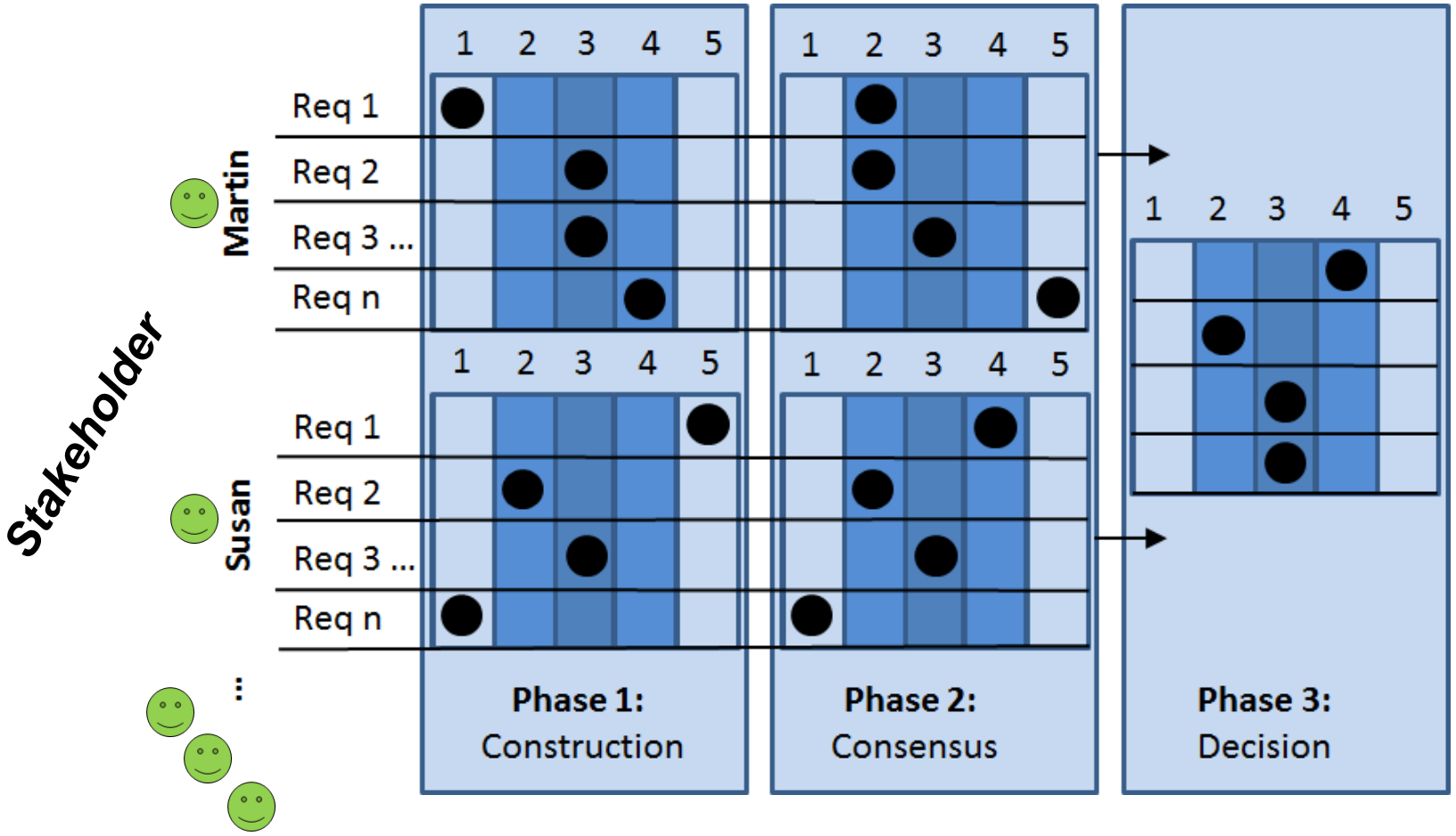
Without customers: “only simple requirements will be implemented”

Without developers: “Exaggerated effort estimation”

Without Management: “additional resources are assumed to be included”

Group Decisions in RE

Graz University of Technology



Group Based Prioritization

Graz University of Technology

Requirements

Stakeholder

	+	-	+	-	+	~	-	+	+
	+	~	+	-	~	-	-	+	~
	+	~	~	-	+	-	~	~	~
	~	~	+	~	+	-	-	+	~
	+	~	+	-	+	-	-	+	~

Example: „Majority Voting“

Group Based Prioritization

Graz University of Technology

Requirements

Stakeholder									
	5	1	5	2	5	3	1	4	5
	4	2	5	1	3	1	2	5	4
	5	3	5	1	5	2	3	3	3
	3	3	4	3	5	2	1	5	3
	4	2	5	2	5	2	2	4	4

Example: „Average Value“ regarding voting

Group-based MAUT

Graz University of Technology

MAUT: Multi Attribute Utility Theory

Interest dimensions, for example:

profit, effort, risk

Properties of requirements:

id, name, description

Requirements
(e.g., regarding a computer watch):

id	name	description
1	show time	…
2	set time	…
3	stopwatch	…
4	conference timer	…
5	moods	…

Group-based MAUT

Graz University of Technology

A must?

Increases profits?

Acceptable efforts?

Not risky?

Evaluation of requirements by individual stakeholders (users)

id	profit	effort	risk
1	-	-	-
2	-	-	-
3	3	3	1
4	8	8	7
5	9	8	10

Group-based MAUT

Graz University of Technology

stakeholder (user=1)

id	profit	effort	risk
1	-	-	-
2	-	-	-
3	3	8	10
4	8	4	3
5	9	2	1

stakeholder (user=2)

id	profit	effort	risk
1	-	-	-
2	-	-	-
3	2	7	9
4	6	3	3
5	9	3	2

$$val(id, dim) = \frac{\sum_{user=1}^{m} eval(id, user, dim)}{m}$$

id	profit	effort	risk
1	-	-	-
2	-	-	-
3	2,5	7,5	9,5
4	7	3,5	3
5	9	2,5	1,5

A must?

Increases profits?

Acceptable efforts?

Not risky?

Group-based MAUT

Graz University of Technology

Example importance distribution (d):

Increases profits? → d=20%

Acceptable efforts? → d=65%

Not risky? → d=15%

$$\boldsymbol{utility(id)} \quad \sum_{dim=1}^{n} \mathrm{val}(id, dim) * importance(dim)$$

id	profit	d	effort	d	risk	d	utility
1	-	-	-	-	-	-	-
2	-	-	-	-	-	-	-
3	2,5	0.2	7,5	0.65	9,5	0.15	**6,8**
4	7	0.2	3,5	0.65	3	0.15	4,125
5	9	0.2	2,5	0.65	1,5	0.15	3,65

Exercises (4)

Graz University of Technology

- Define 5 requirements for an application you select on your own.
- Prioritize these requirements using MAUT (you can use the MAUT dimensions presented in the lecture).

Graz University of Technology

Variability Modeling

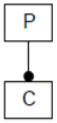
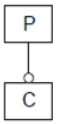
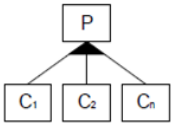
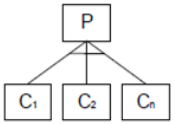
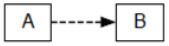
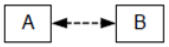
Feature Models

Graz University of Technology

- Modeling the variability of software
- Graphical notation
- “Feature” as basic concept
- Constraints between concepts reduce the set of possible “configurations”
- Formal definition of concepts
- Automated reasoning about models

Semantics

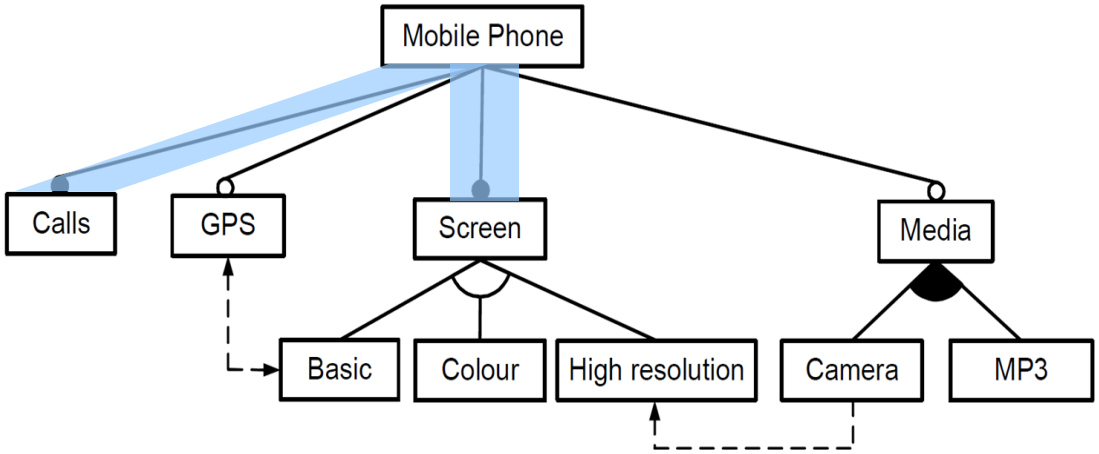
Graz University of Technology

Relationship		PL Mapping	Mobile Phone Example
MANDATORY		$P \leftrightarrow C$	MobilePhone $\leftrightarrow$ Calls MobilePhone $\leftrightarrow$ Screen
OPTIONAL		$C \rightarrow P$	GPS $\rightarrow$ MobilePhone Media $\rightarrow$ MobilePhone
OR		$P \leftrightarrow (C_1 \vee C_2 \vee \ldots \vee C_n)$	Media $\leftrightarrow$ (Camera $\vee$ MP3)
ALTERNATIVE		$(C_1 \leftrightarrow (\neg C_2 \wedge \ldots \wedge \neg C_n \wedge P)) \wedge$ $(C_2 \leftrightarrow (\neg C_1 \wedge \ldots \wedge \neg C_n \wedge P)) \wedge$ $(C_n \leftrightarrow (\neg C_1 \wedge \neg C_2 \wedge \ldots \wedge \neg C_{n-1} \wedge P))$	(Basic $\leftrightarrow$ ($\neg$Color $\wedge$ $\neg$Highresolution $\wedge$ Screen)) $\wedge$ (Color $\leftrightarrow$ ($\neg$Basic $\wedge$ $\neg$Highresolution $\wedge$ Screen)) $\wedge$ (Highresolution $\leftrightarrow$ ($\neg$Basic $\wedge$ $\neg$Color $\wedge$ Screen))
IMPLIES		$A \rightarrow B$	Camera $\rightarrow$ Highresolution
EXCLUDES		$\neg(A \wedge B)$	$\neg$(GPS $\wedge$ Basic)

[Benavides et al. 2010]

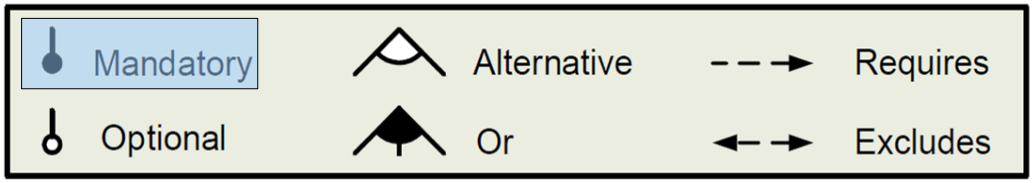
„Mandatory“ Relationships

Graz University of Technology



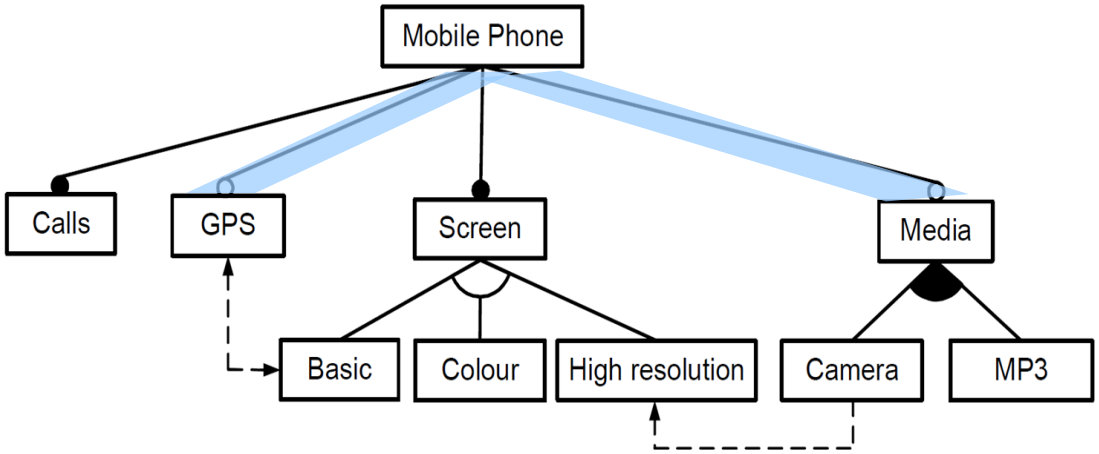
- A ↔ B
- MobilePhone ↔ Calls
- MobilePhone ↔ Screen

[Benavides et al. 2010]

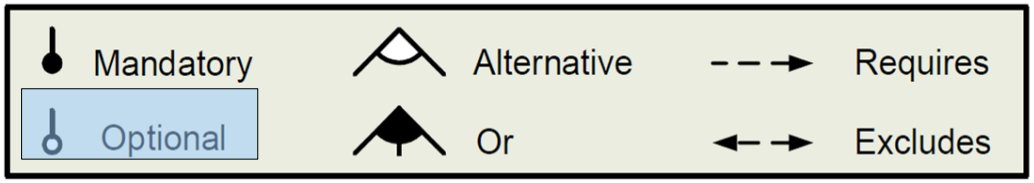


„Optional“ Relationships: Semantics

Graz University of Technology

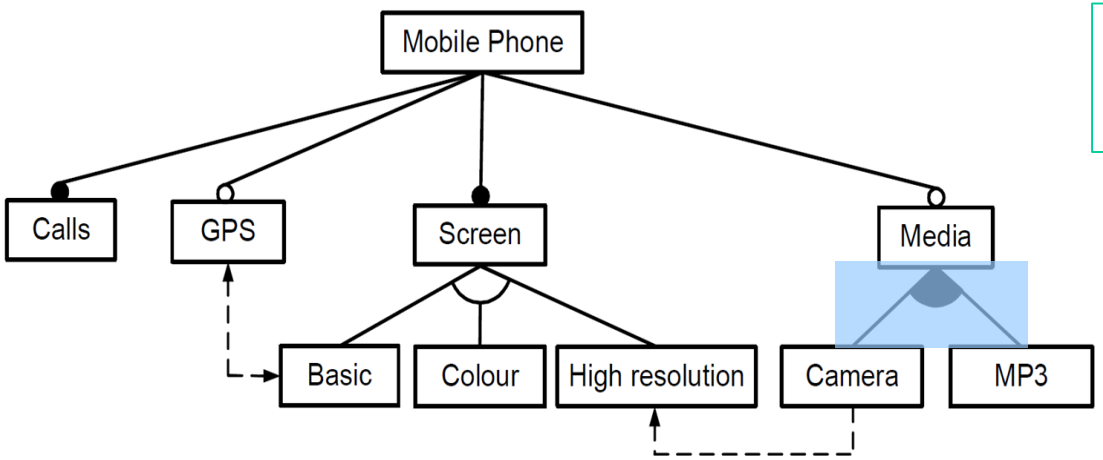


- B → A
- GPS → MobilePhone
- Media → MobilePhone

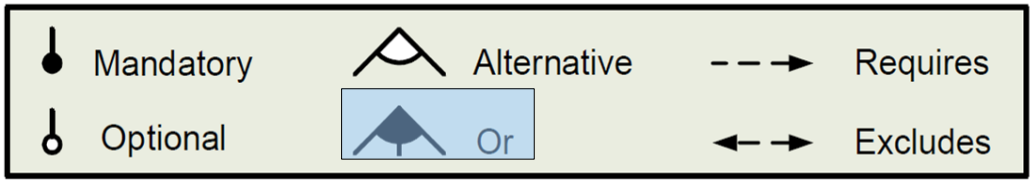


„OR“ Relationships: Semantics

Graz University of Technology

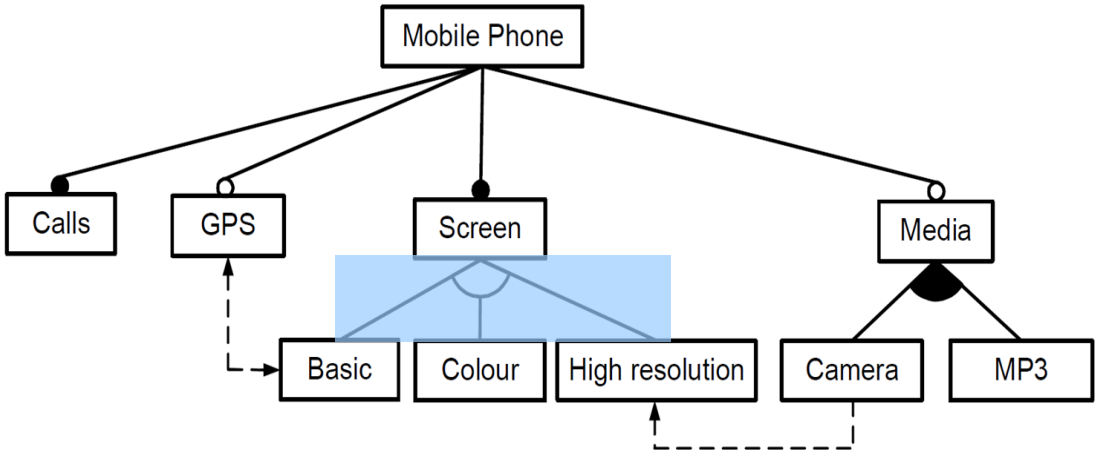


- $A \leftrightarrow (B1 \vee B2 \vee \ldots \vee Bn)$
- $Media \leftrightarrow (Camera \vee MP3)$

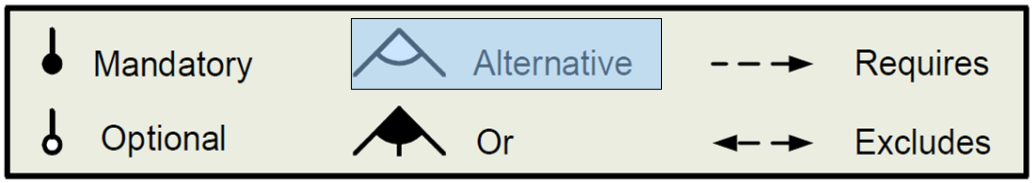


„Alternative“ Relationships: Semantics

Graz University of Technology

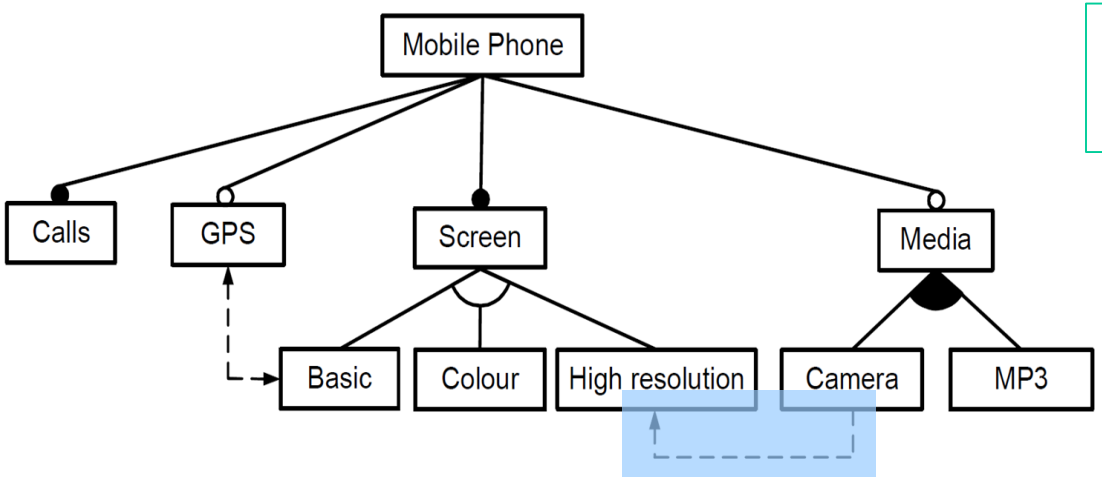


- $(B1 \leftrightarrow (\neg B2 \wedge \ldots \wedge \neg Bn \wedge A)) \wedge$
  $\ldots \wedge$
  $(Bn \leftrightarrow (\neg B1 \wedge \ldots \wedge \neg Bn\text{-}1 \wedge A))$
- $(Basic \leftrightarrow (\neg Colour \wedge \neg HighRes \wedge Screen)) \wedge$
  $(Colour \leftrightarrow (\neg Basic \wedge \neg HighRes \wedge Screen)) \wedge$
  $(HighRes \leftrightarrow (\neg Colour \wedge \neg Basic \wedge Screen))$

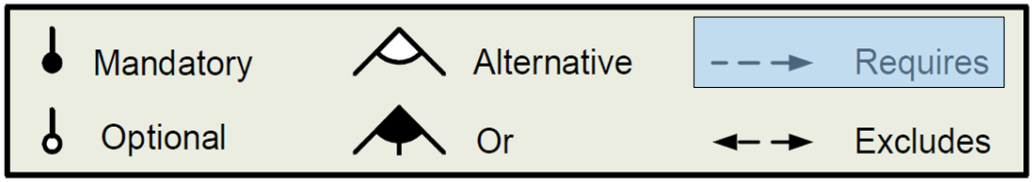


„Requires“ Relationships: Semantics

Graz University of Technology

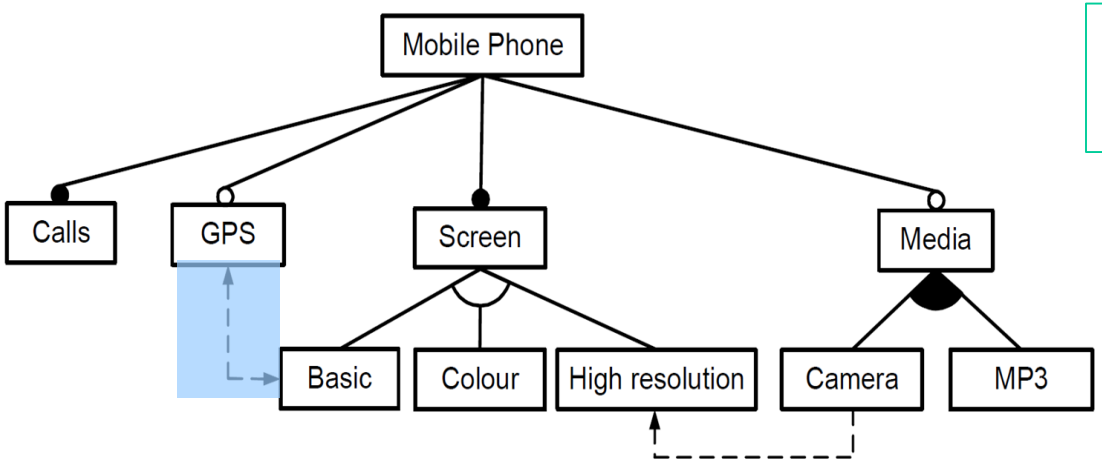


- A → B
- Camera → HighRes

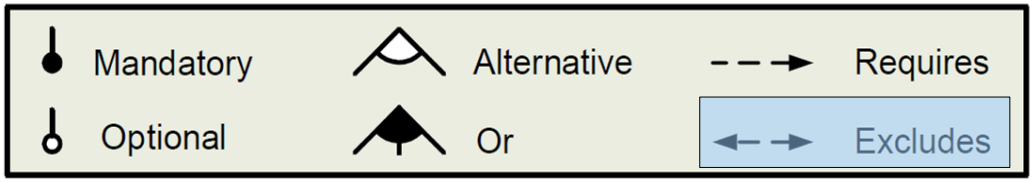


„Excludes“ Relationships: Semantics

Graz University of Technology



- $\neg(A \wedge B)$
- $\neg(GPS \wedge Basic)$



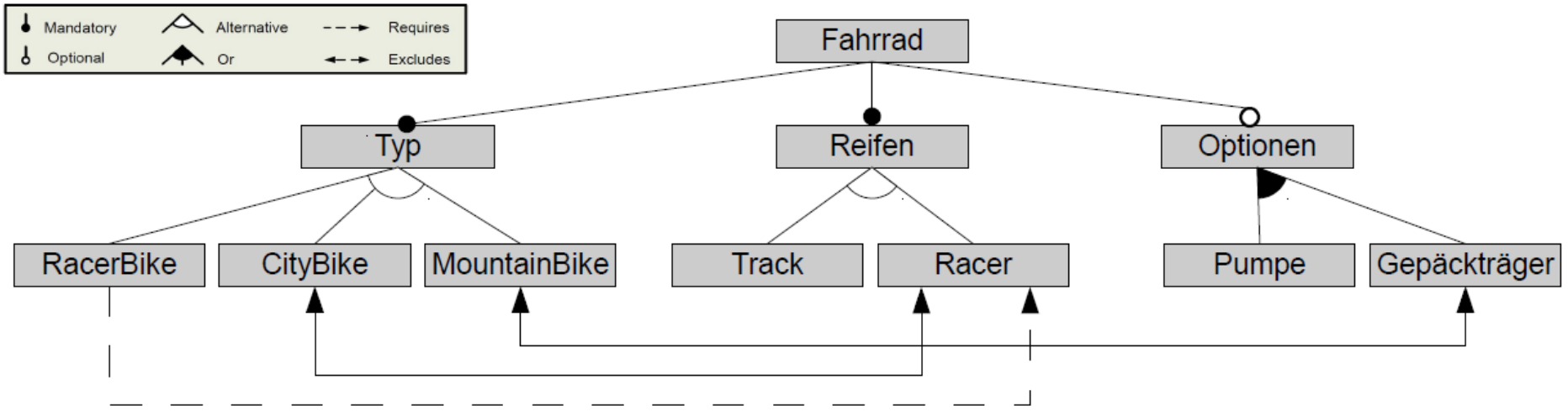
Integrated Knowledge Base

Graz University of Technology

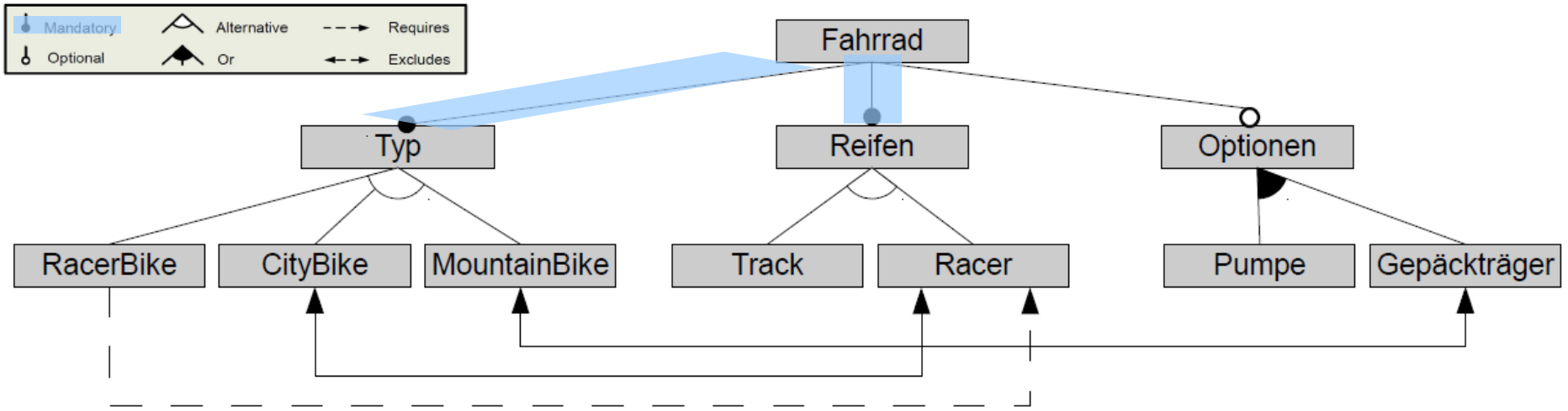
- c_1: MobilePhone $\leftrightarrow$ Calls
- c_2: MobilePhone $\leftrightarrow$ Screen
- c_3: GPS $\rightarrow$ MobilePhone
- c_4: Media $\rightarrow$ MobilePhone
- c_5: Media $\leftrightarrow$ (Camera $\vee$ MP3)
- c_6: (Basic $\leftrightarrow$ (¬Colour $\wedge$ ¬HighRes $\wedge$ Screen)) $\wedge$
  (Colour $\leftrightarrow$ (¬Basic $\wedge$ ¬HighRes $\wedge$ Screen)) $\wedge$
  (HighRes $\leftrightarrow$ (¬Colour $\wedge$ ¬Basic $\wedge$ Screen))
- c_7: Camera $\rightarrow$ HighRes
- c_8: ¬(GPS $\wedge$ Basic)

Next Example: Translation = ?

Graz University of Technology

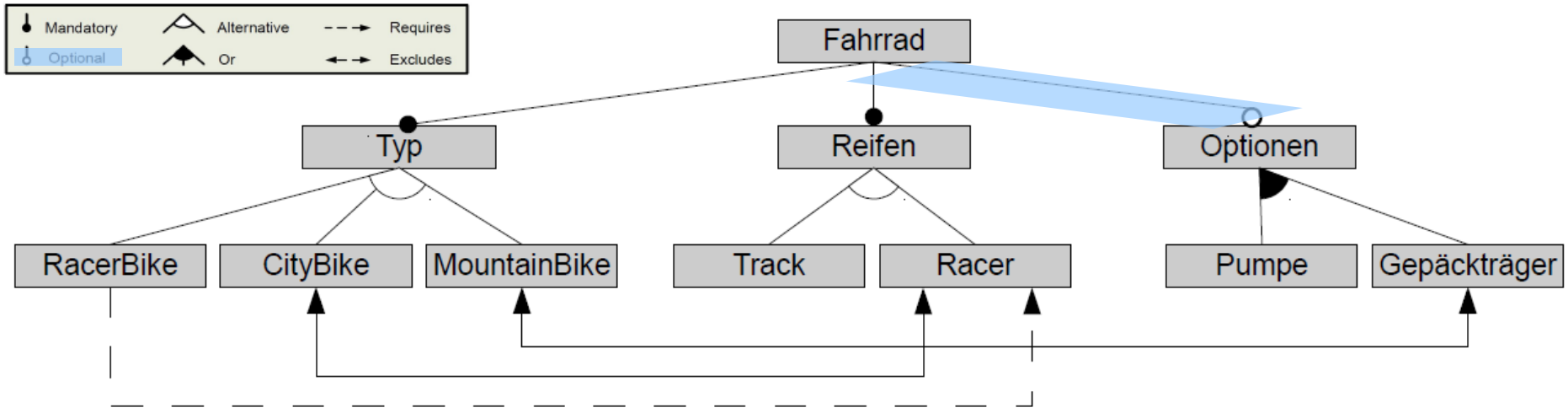


„Mandatory“



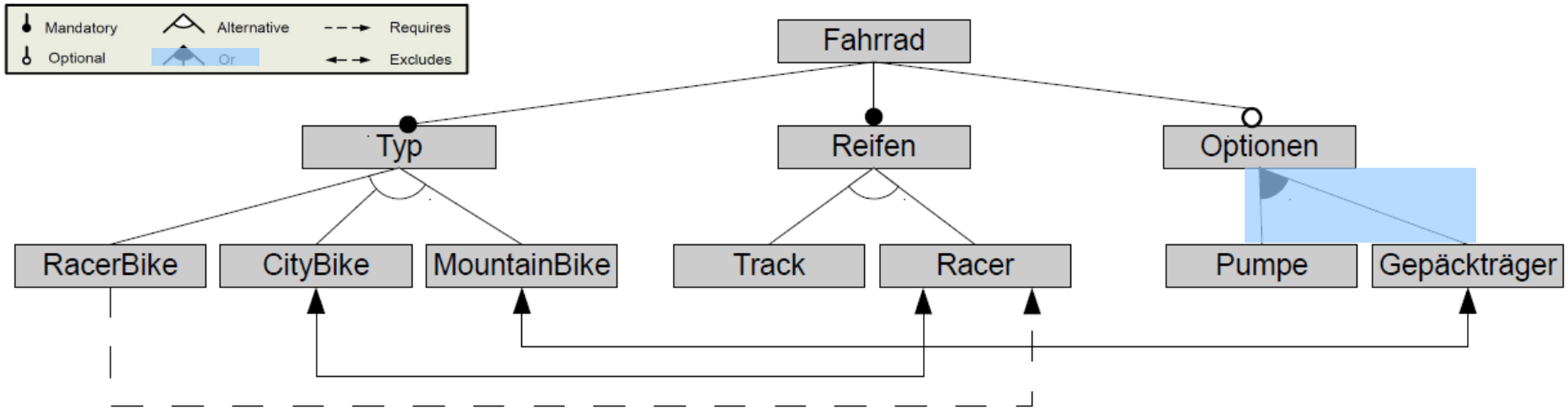
- Fahrrad ↔ Typ
- Fahrrad ↔ Reifen

„Optional“



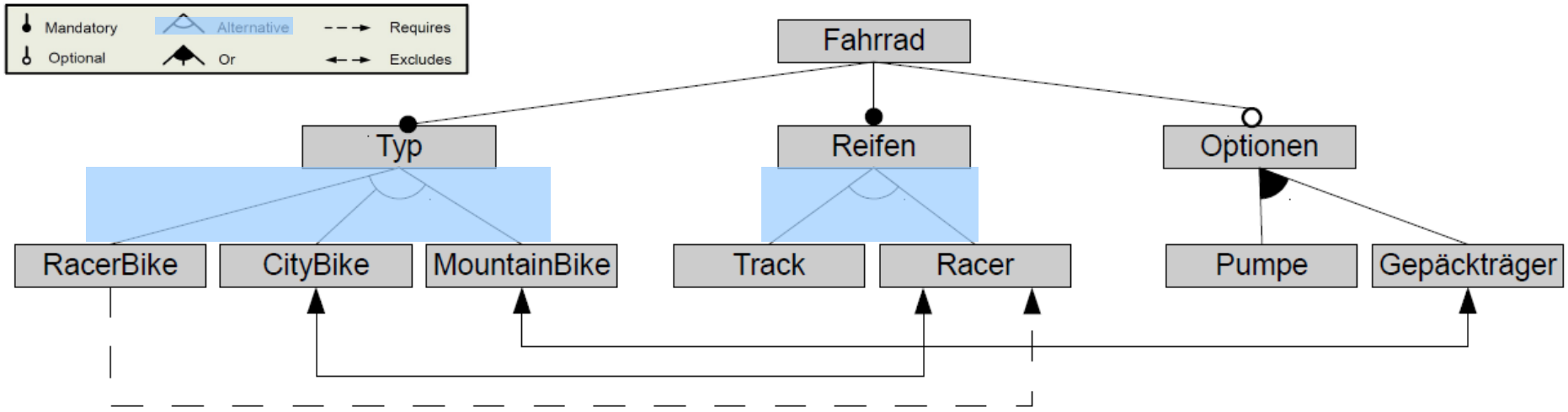
- Optionen → Fahrrad

„OR“



- Optionen $\leftrightarrow$ (Pumpe $\vee$ Gepäckträger)

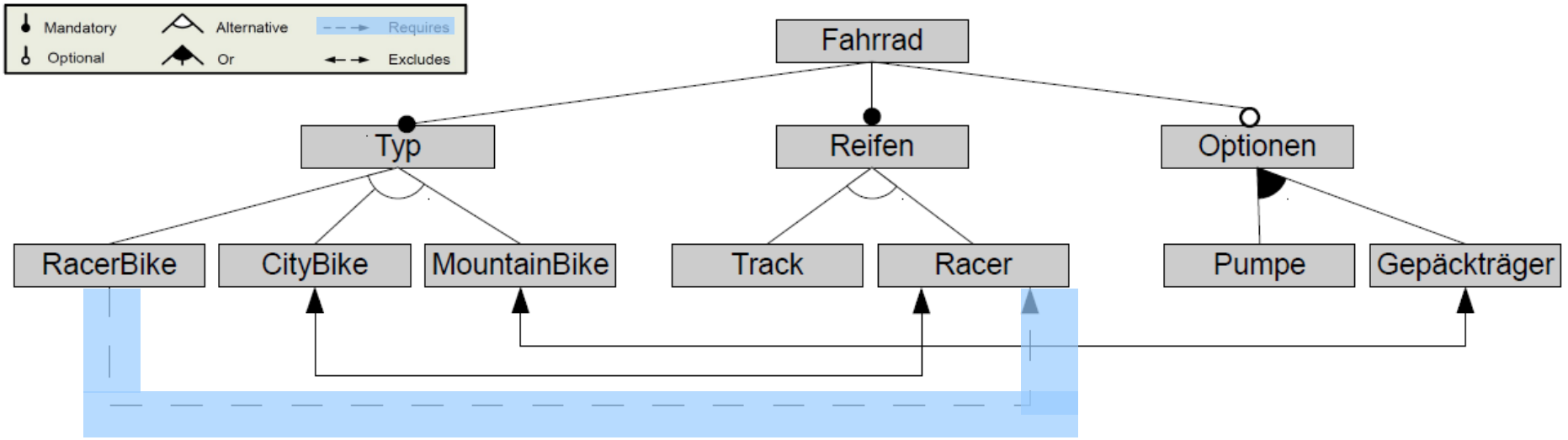
„Alternative“



- $(\text{RacerBike} \leftrightarrow (\neg\text{CityBike} \wedge \neg\text{MountainBike} \wedge \text{Typ})) \wedge$
  $(\text{CityBike} \leftrightarrow (\neg\text{RacerBike} \wedge \neg\text{MountainBike} \wedge \text{Typ})) \wedge$
  $(\text{MountainBike} \leftrightarrow (\neg\text{RacerBike} \wedge \neg\text{CityBike} \wedge \text{Typ}))$

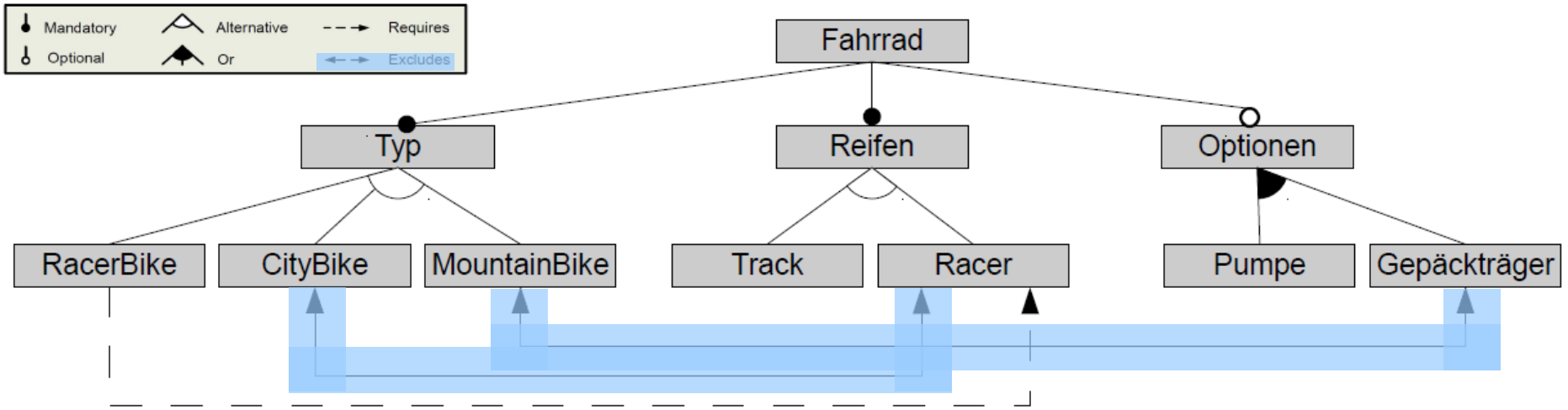
- $(\text{Track} \leftrightarrow (\neg\text{Racer} \wedge \text{Reifen})) \wedge$
  $(\text{Racer} \leftrightarrow (\neg\text{Track} \wedge \text{Reifen}))$

„Requires“



- RacerBike → Racer

„Excludes“



- ¬(CityBike ∧ Racer)
- ¬(MountainBike ∧ Gepäckträger)

Integrated Knowledge Base

Graz University of Technology

- c_1: Fahrrad $\leftrightarrow$ Typ
- c_2: Fahrrad $\leftrightarrow$ Reifen
- c_3: Optionen $\rightarrow$ Fahrrad
- c_4: Optionen $\leftrightarrow$ (Pumpe $\vee$ Gepäckträger)
- c_5: (RacerBike $\leftrightarrow$ (¬CityBike $\wedge$ ¬MountainBike $\wedge$ Typ)) $\wedge$
  (CityBike $\leftrightarrow$ (¬RacerBike $\wedge$ ¬MountainBike $\wedge$ Typ)) $\wedge$
  (MountainBike $\leftrightarrow$ (¬RacerBike $\wedge$ ¬CityBike $\wedge$ Typ))
- c_6: (Track $\leftrightarrow$ (¬Racer $\wedge$ Reifen)) $\wedge$
  (Racer $\leftrightarrow$ (¬Track $\wedge$ Reifen))
- c_7: RacerBike $\rightarrow$ Racer
- c_8: ¬(CityBike $\wedge$ Racer)
- c_9: ¬(MountainBike $\wedge$ Gepäckträger)

CHOICLA Feature Model

Graz University of Technology

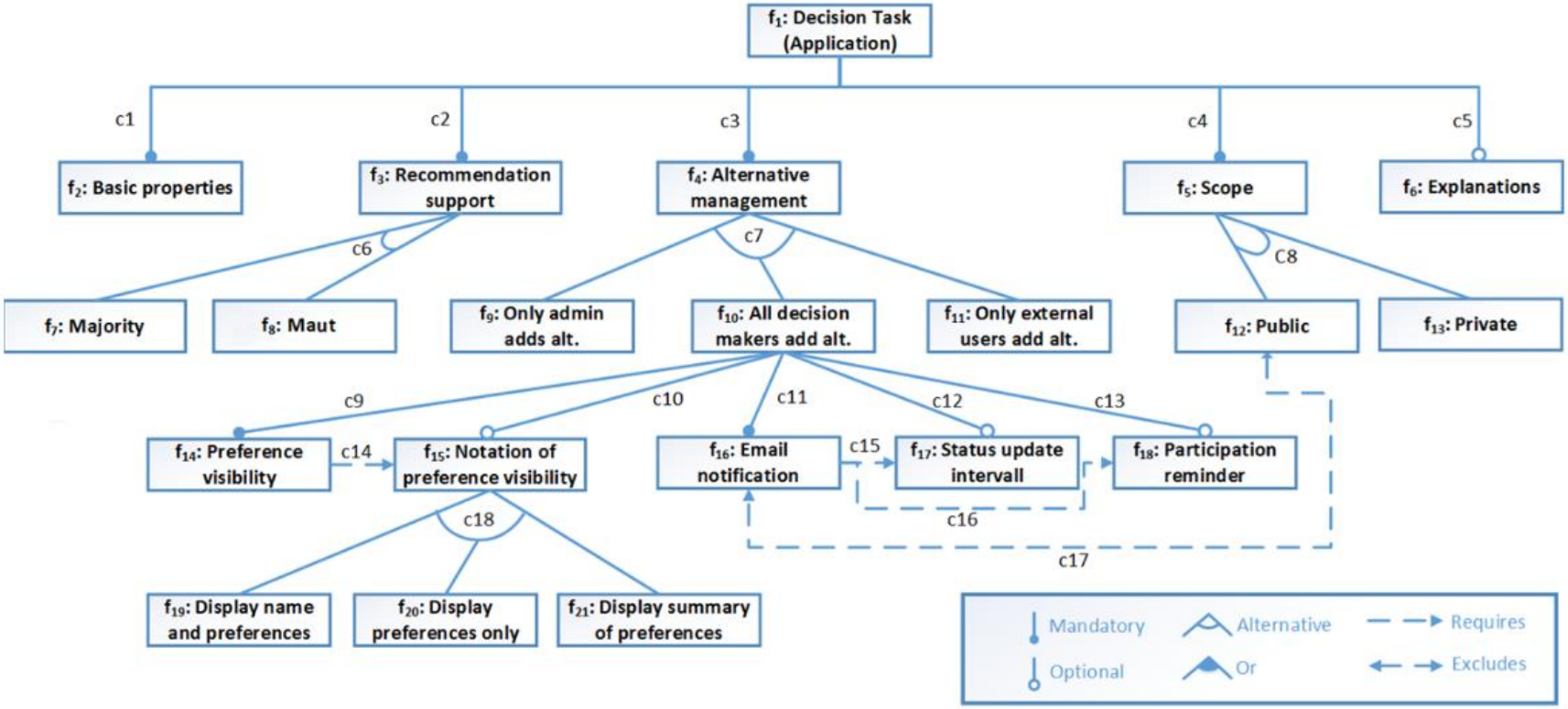


Figure 1. Fragment of the CHOICLA feature model. f_i is used as abbreviation for the individual features, for example, f_3 is the short notation for feature *Recommendation Support*.

[Stettinger et al. 2015]

CHOICLA
Application Requirements

Graz University of Technology

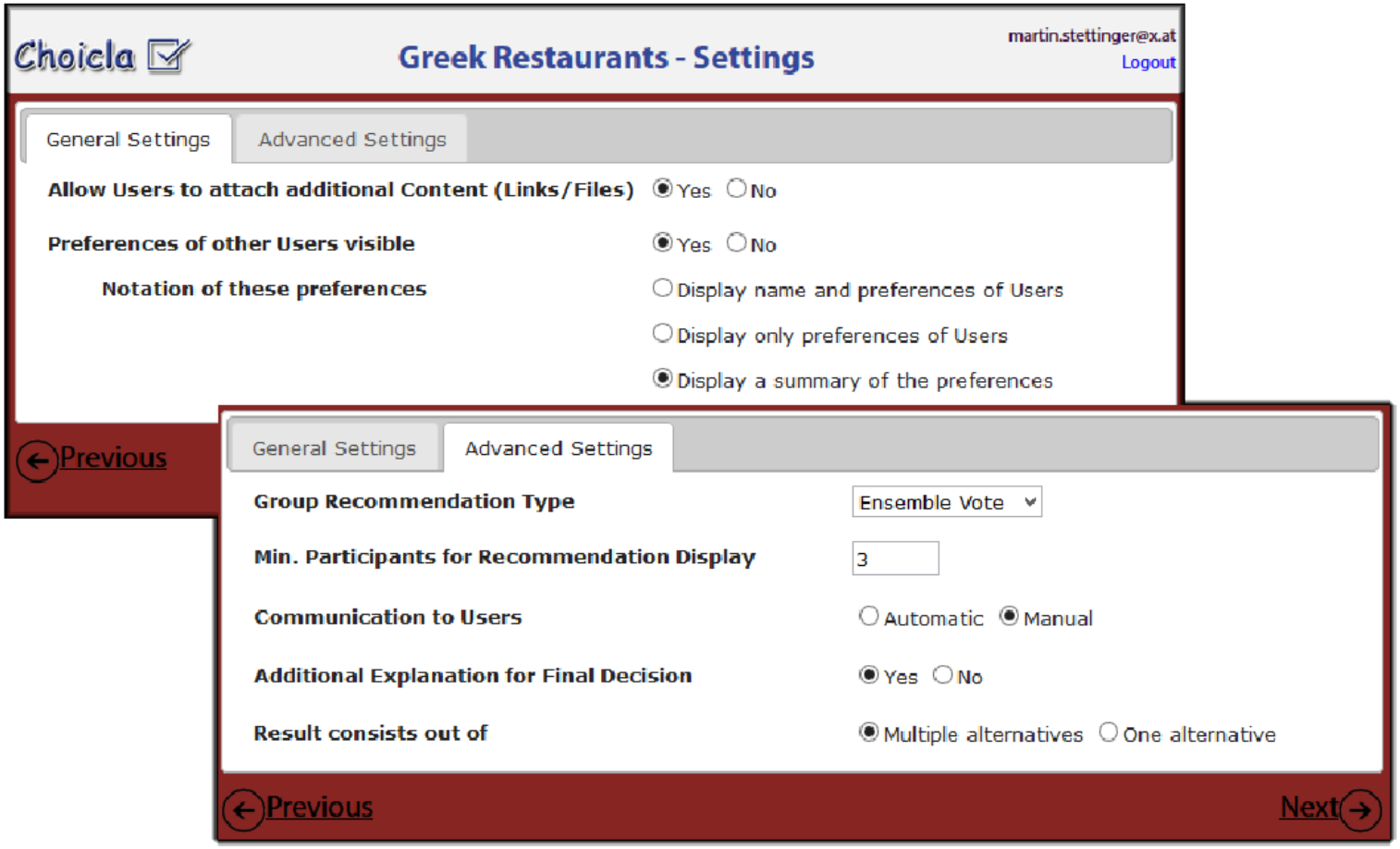


Figure 2. Example set of parameters which can be selected when creating a new CHOICLA decision app.

[Stettinger et al. 2015]

Anomalies in Feature Models

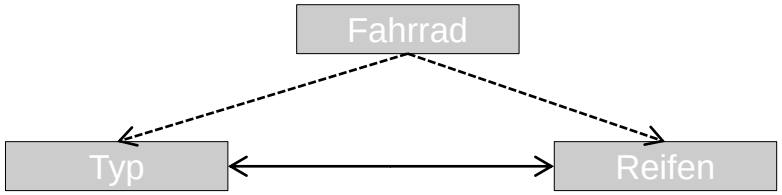
Graz University of Technology

- Anomalies are patterns that do not conform to a well-defined notion of normal behavior
  [Chandola et al. 2009]
- Negative Impacts:
  - Maintainability ☹
  - Readability ☹
  - Stability ☹
- Types of Anomalies
  - **Inconsistencies**
  - Redundancies

Example: Inconsistencies

Graz University of Technology

- c_0: Fahrrad
- c_1: Fahrrad $\rightarrow$ Typ
- c_2: Fahrrad $\rightarrow$ Reifen
- c_3: $\neg$(Typ $\wedge$ Reifen)

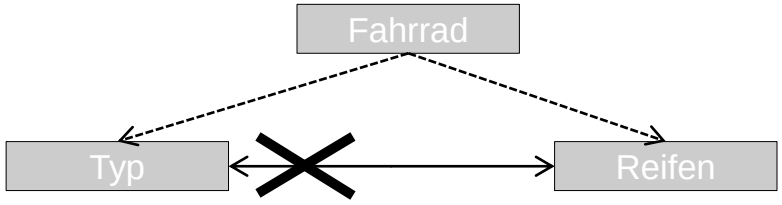


Inconsistency: $\neg\exists$ solution for $\{c_0, c_1, c_2, c_3\}$

Example: Resolving Inconsistencies

Graz University of Technology

- c_0: Fahrrad
- c_1: Fahrrad → Typ
- c_2: Fahrrad → Reifen
- ~~c_3: ¬(Typ ∧ Reifen)~~



Minimal Diagnosis (D):

$D \subseteq \{c_1,c_2,c_3\}$: consistent($\{c_0,c_1,c_2,c_3\} - D$) and $\neg\exists D'$: $D' \subset D$.

Minimal diagnoses: $\{c_3\}$, … ?

Exercises (5)

Graz University of Technology

a. Design a feature model

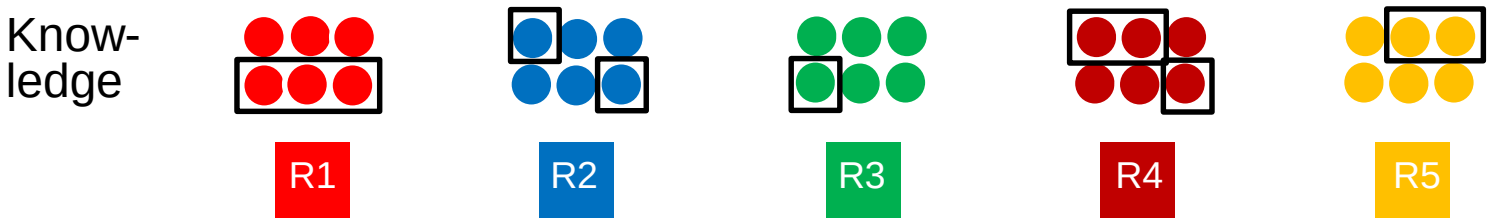
b. Translate the feature model into a corresponding logical representation

Graz University of Technology

Decision Biases

„Hidden Profiles“ (1)

Graz University of Technology



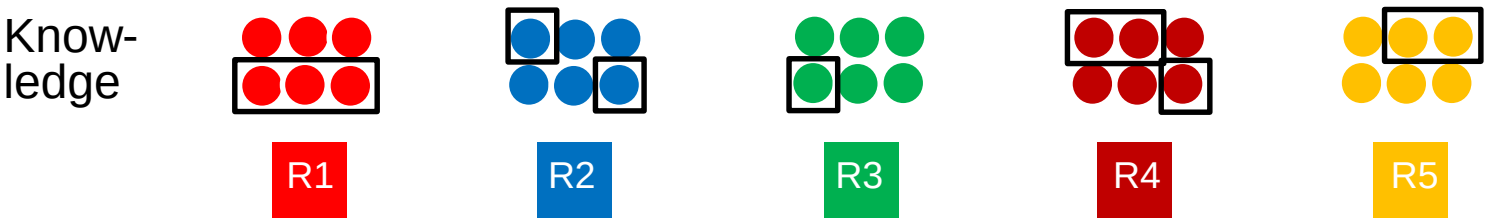
Stakeholder/ Requirement				Public (Preferences known)	Public (Preferences ¬ known)

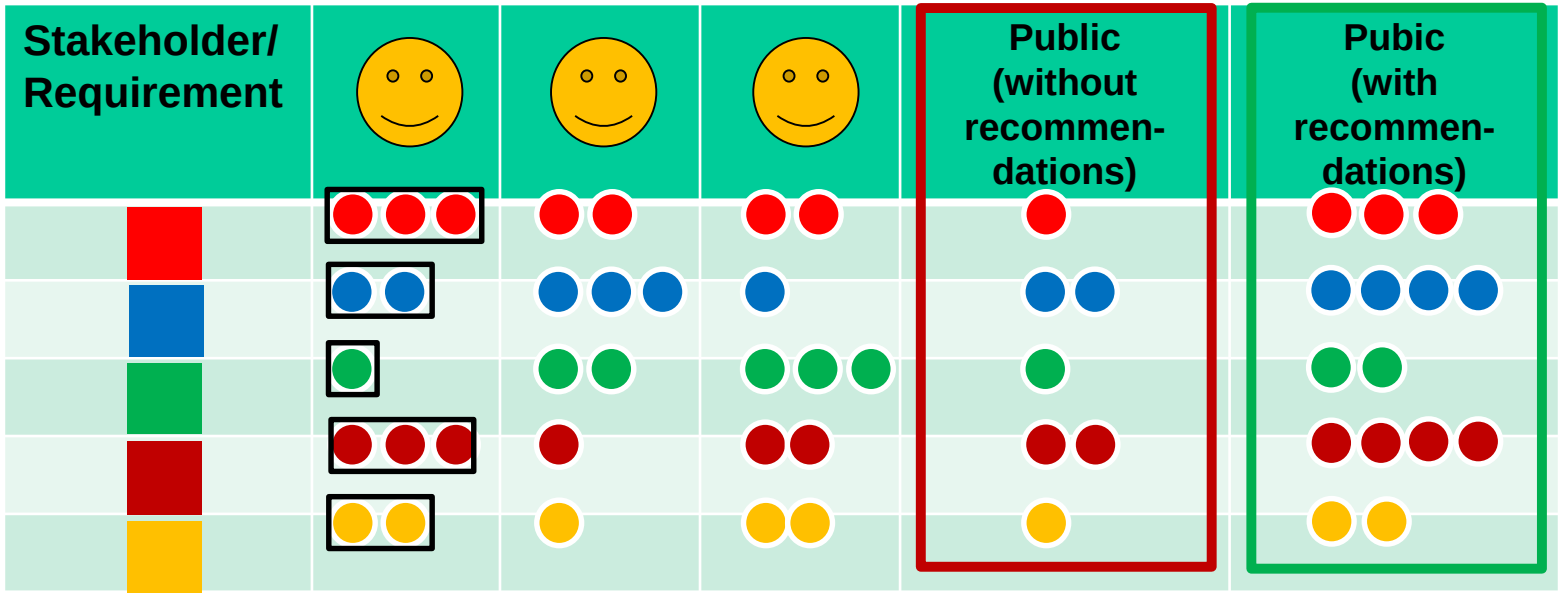
☹ **Stakeholder preferences known: --Knowledge interchange**

☺ **Stakeholder preferences not known: ++Knowledge interchange**

„Hidden Profiles“ (2)

Graz University of Technology



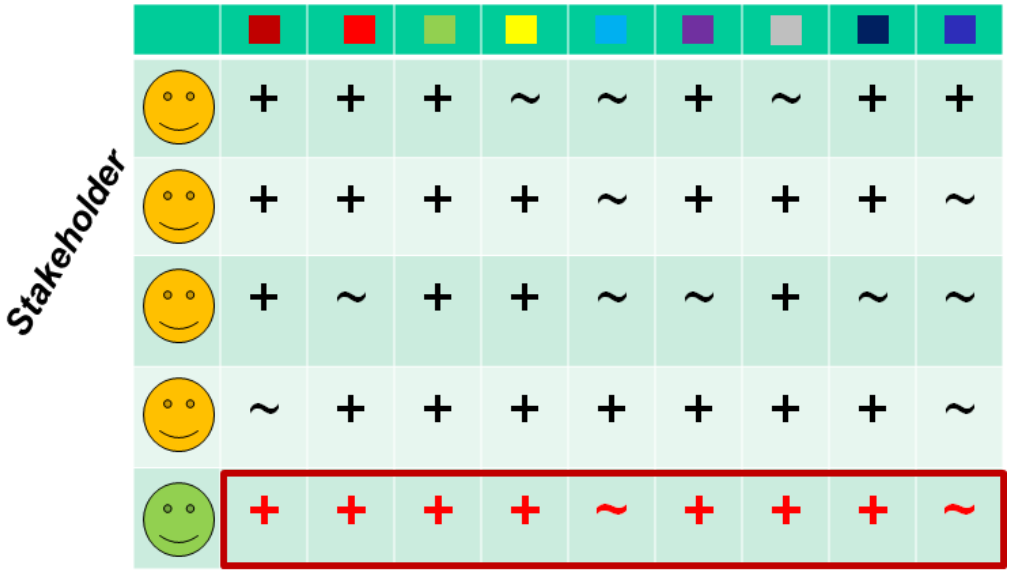


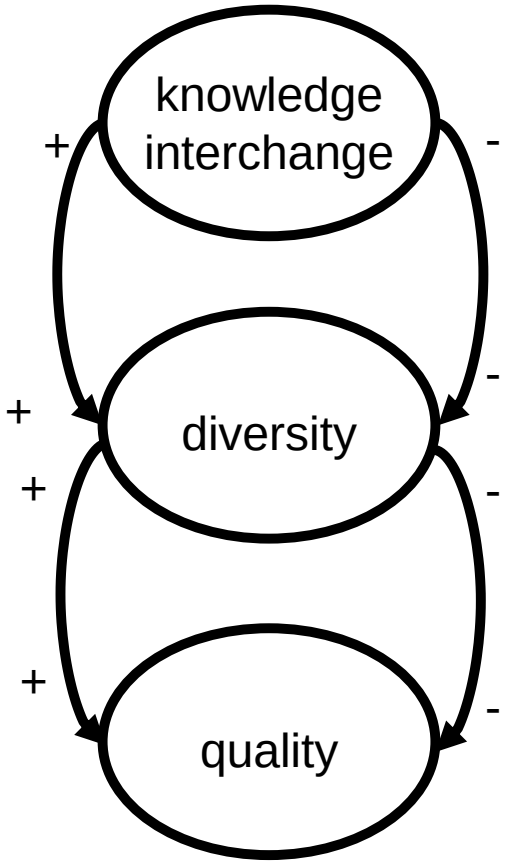
☹ **No group recommendations: --Knowledge interchange**

☺ **Group recommendations: ++Knowledge interchange**

Need Them All

Graz University of Technology





☹ --Knowledge interchange: --Diversity

☹ --Diversity: --Quality of requirements

☺ ++Knowledge interchange: ++Diversity

☺ ++Diversity: ++Quality of requirements

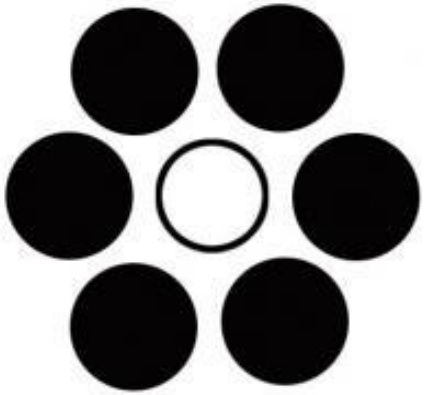
Context Effects (1)

Graz University of Technology

- A decision is always made depending on the context in which item alternatives are presented
- For example, completely inferior item alternatives can trigger significant changes in choice behaviors
- Example context effects are discussed in the following

Short Note: Ebbinghaus Effect

Graz University of Technology



- Illusion of relative size perception
- Triggered by context in which objects are shown
- Commonalities with context effects

Context Effects (2)

Graz University of Technology

Release Alternatives

Dimensions

Risk	5	3
Profit	120.000	50.000

	√	-	-	√
	-	√	√	-

50% - 50%

Dimensions

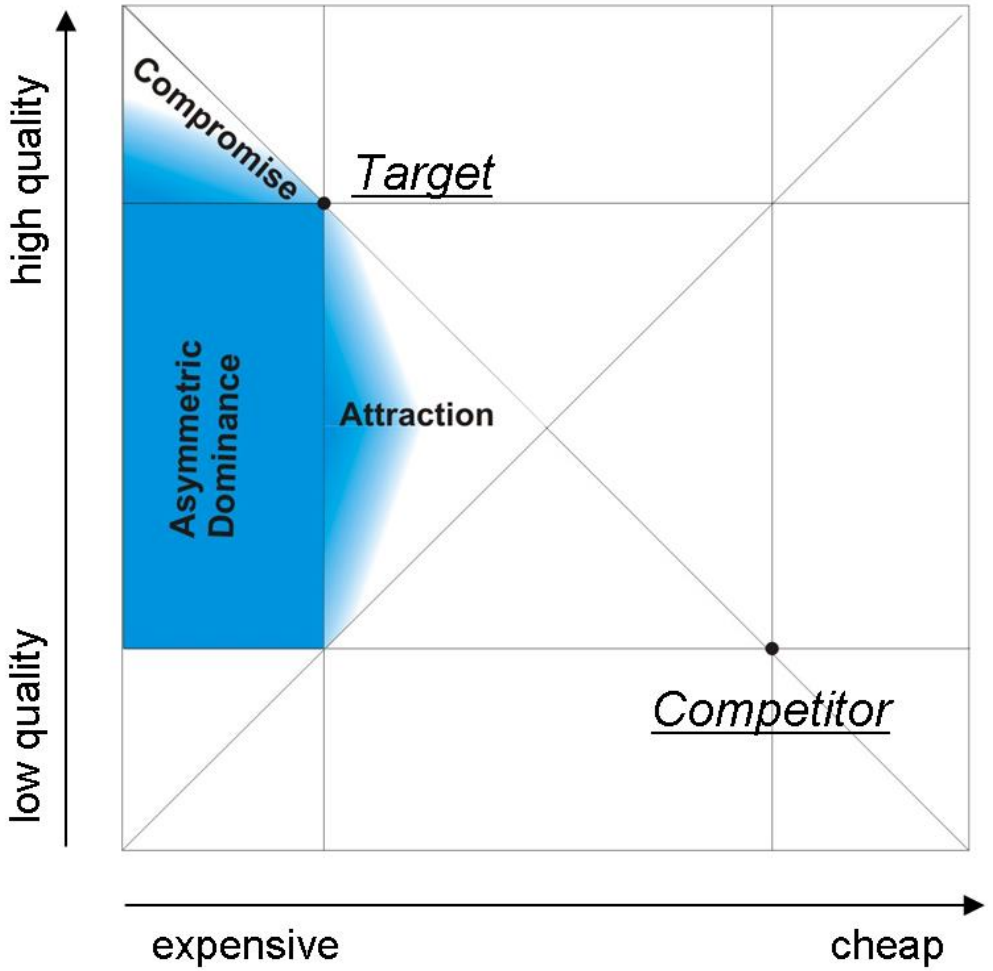
Risk	5	3	7
Profit	120.000	50.000	100.000

	√	√	-	√
	-	-	√	-
	-	-	-	-

☹ **Asymmetric Dominance:**
dominates .

75% - 25%

Context Effects: Overview



- **Compromise** : Target (T) is a compromise to decoy item D (T is **less expensive** and has **slightly lower quality**)
- **Asymmetric Dominance**: T dominates D (T is **cheaper** and has a **higher quality**)
- **Attraction**: T is more attractive than D (T is **slightly more expensive** but has a **higher quality**)

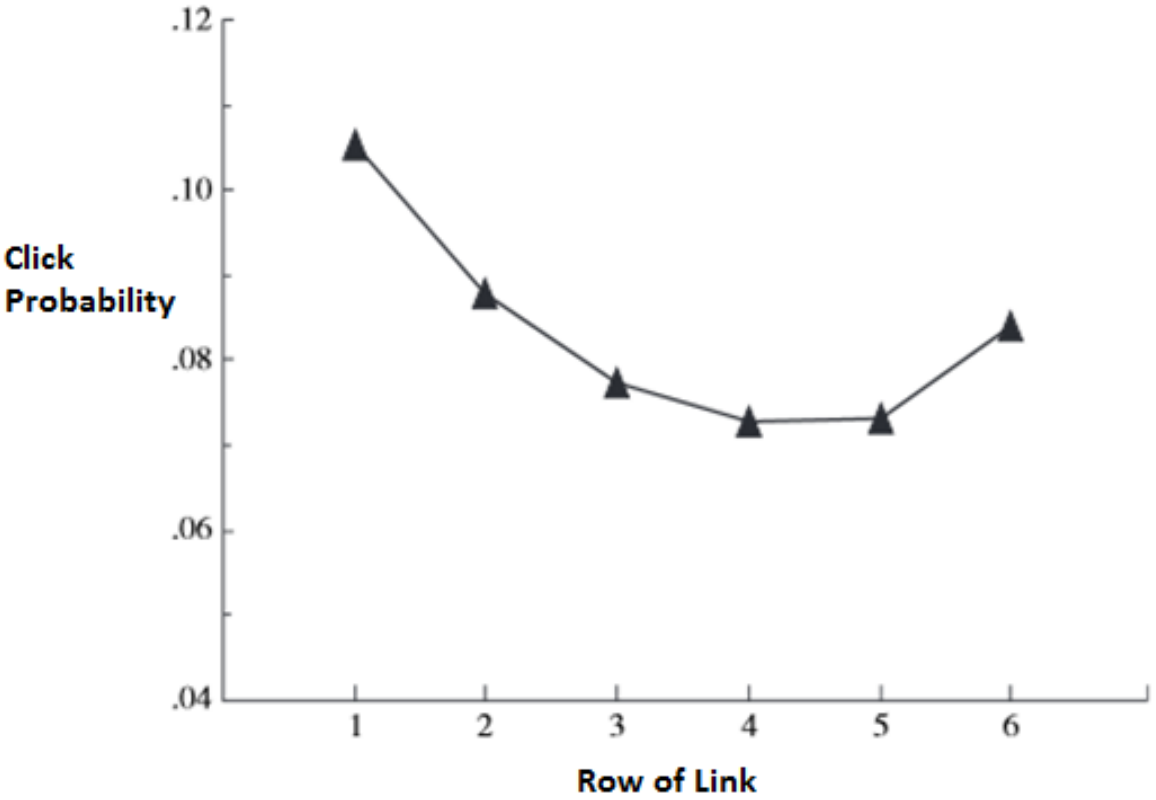
Primacy/Recency Effects

Graz University of Technology

- Describe situations in which items presented at the beginning and at the end of a list are evaluated significantly more often than others
- Typically, users are not interested in evaluating large lists to identify those that best fit their preferences
- The same phenomenon exists as well in the context of web search scenarios

Primacy/Recency in Web Links Selection

Graz University of Technology



- Primacy effect
- Efficacy of the first link
- But also recency
- Tendency to click links at the end

Primacy/Recency in Evaluation of Texts

Graz University of Technology

Negative salient description (Zeus)
Upper price segment. Food quality does not justify the price. Breathtaking view over the city center. Easily accessible via the highway, plenty of parking opportunities. Terrific ambiente with an open-view kitchen. Side dishes and sauces can be freely customized. Friendly service. Elegant furniture and a good place for relaxing. Personel leaves the impression of unkemptness. Poor cleanliness.

Table 5. Negative salient description (= *Description 1*) of a restaurant.

Positive salient description (Zeus)
Breathtaking view over the city center. Easily accessible via the highway, plenty of parking opportunities. Terrific ambiente with an open-view kitchen. Side dishes and sauces can be freely customized. Friendly service. Upper price segment. Food quality does not justify the price. Personel leaves the impression of unkemptness. Poor cleanliness. Elegant furniture and a good place for relaxing.

Table 4. Positive salient description (= *Description 2*) of a restaurant.

- Task was to evaluate alternatives
- 1st group: 5 star rating
- 2nd group: MAUT
- Evaluation difference between negative and positive salient was significantly lower with MAUT

[Stettinger et al. 2015]

Anchoring Effects

Graz University of Technology

Anchoring effects cause decisions which are influenced by the group member who first articulated his/her preferences.

Example from RE follows …

Anchoring Effects

Graz University of Technology

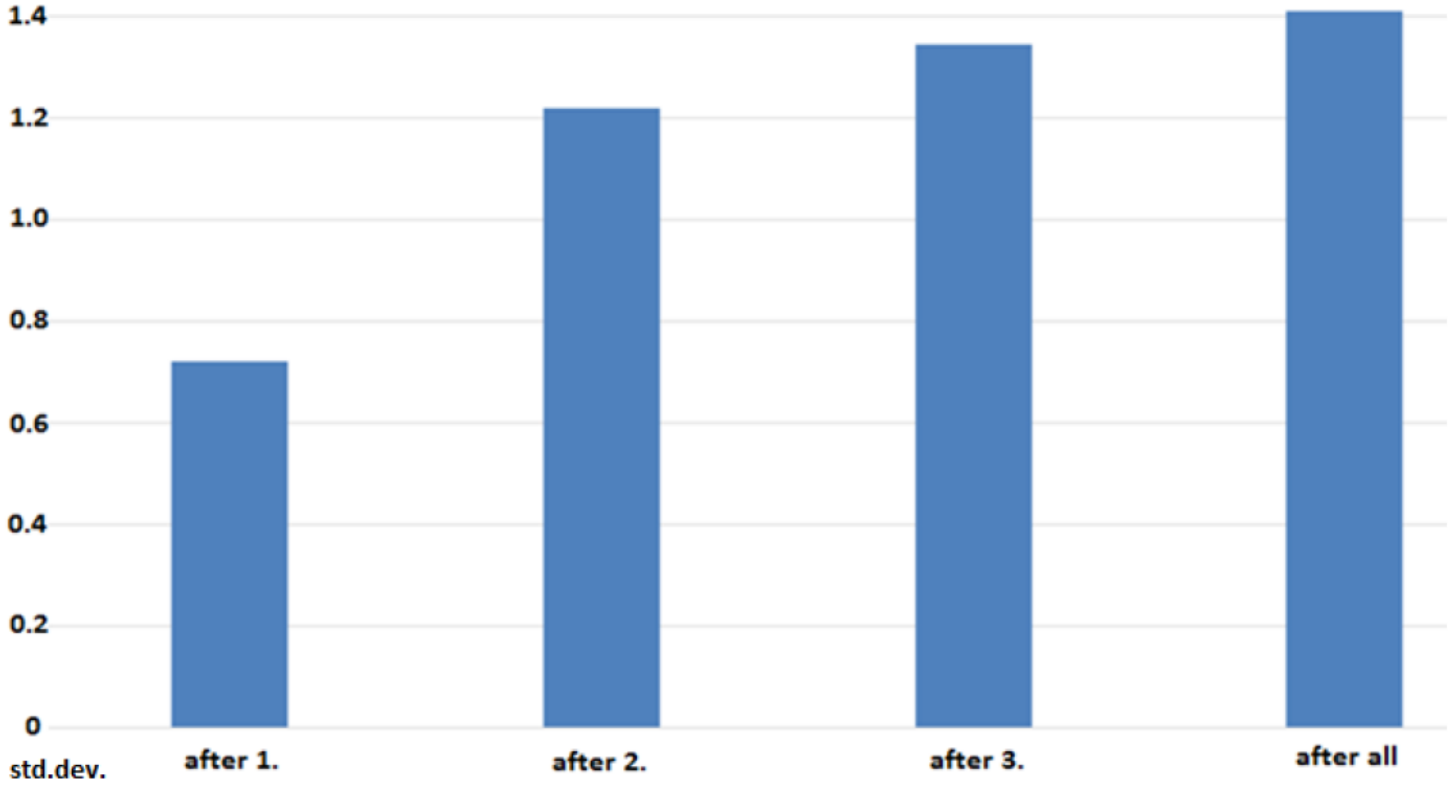


Fig. 6. Standard deviations of user ratings of alternatives (requirements) depending on the time of preference disclosure (after 1..3, or all users articulated their preferences).

[Stettinger et al. 2015b]

Anchoring Effects

Graz University of Technology

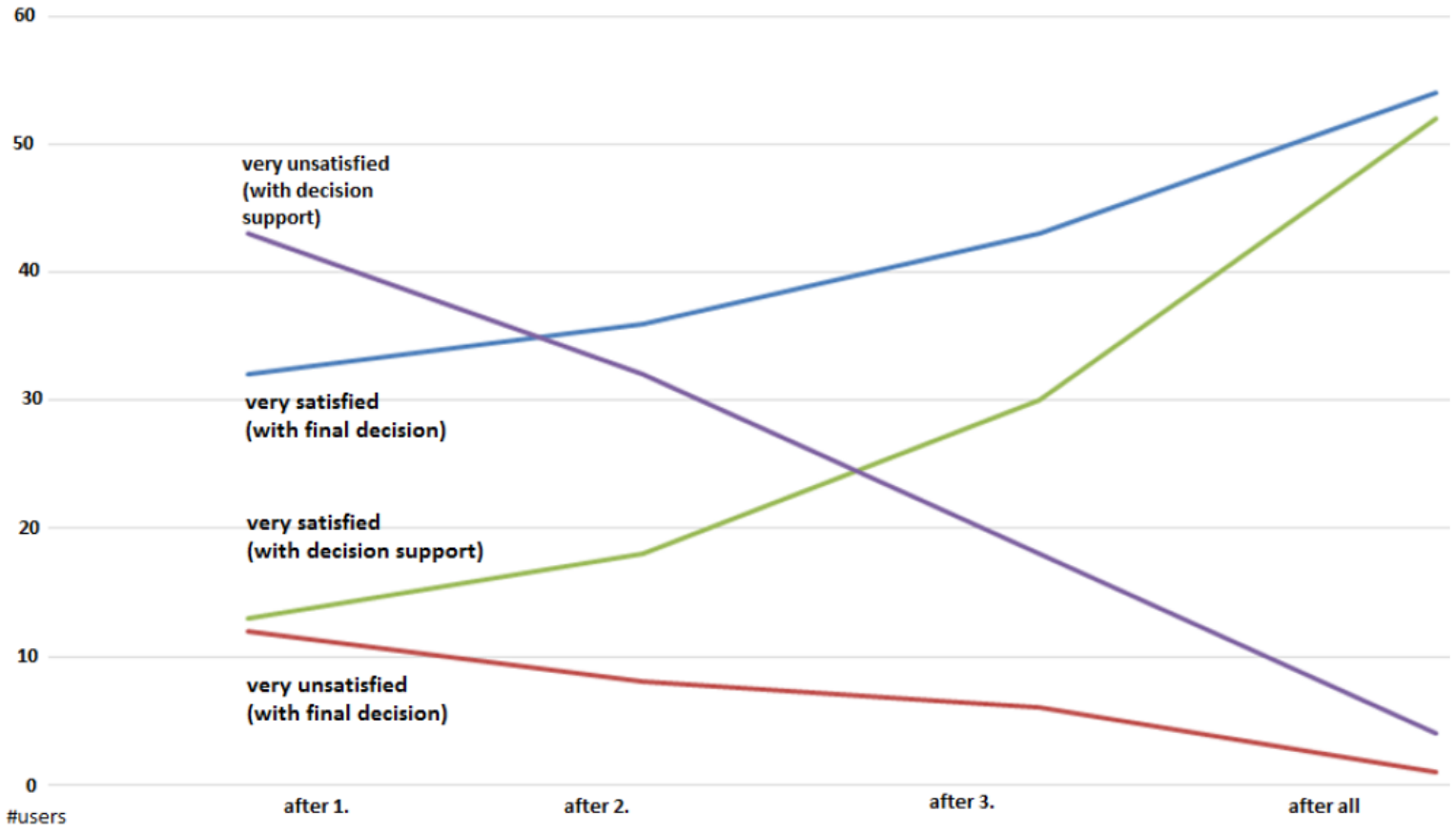


Fig. 7. *Satisfaction with final group decision* and *perceived degree of decision support* depending on the time of preference disclosure (after 1..3, or all users articulated their preferences).

[Stettinger et al. 2015b]

Anchoring Effects

Graz University of Technology

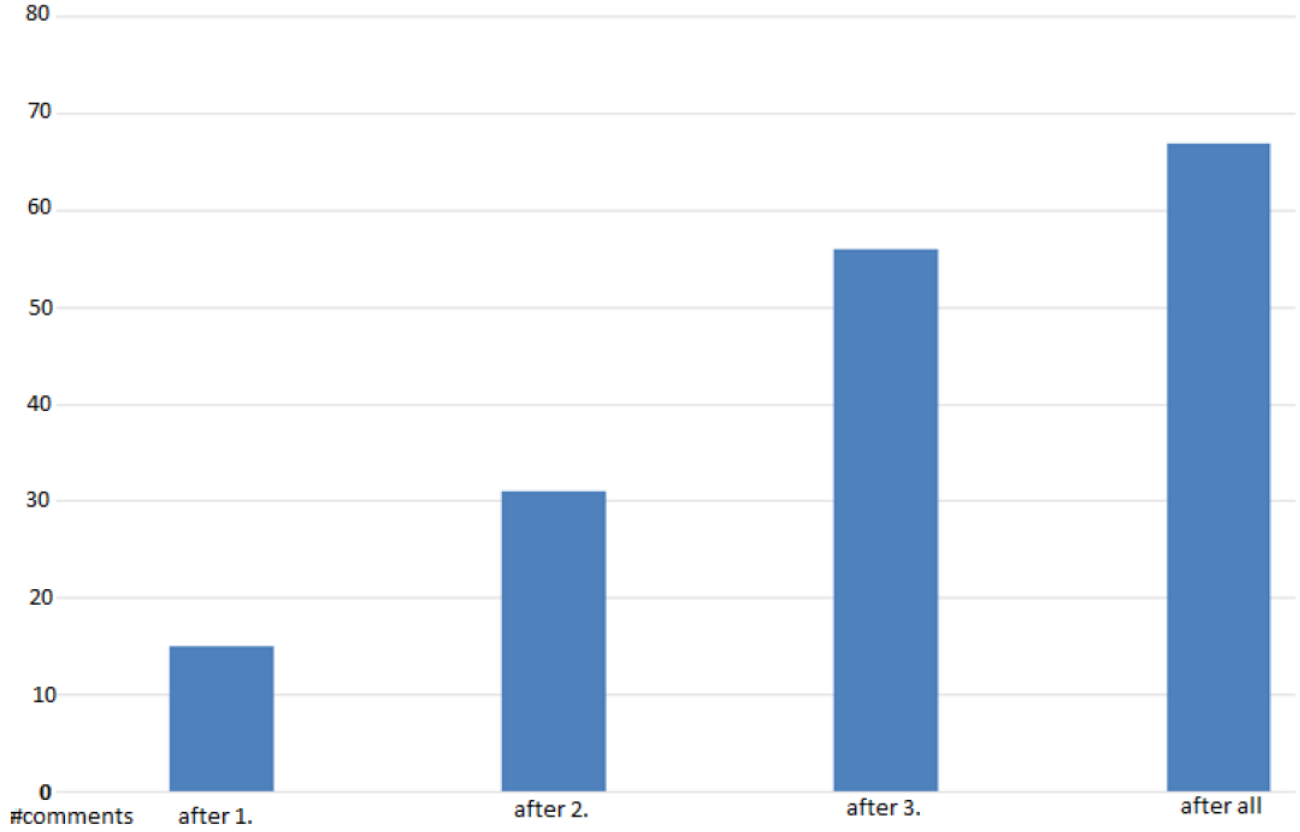


Fig. 8. Number of comments in the CHOICLA discussion forum depending on the time of preference disclosure (after 1..3, or all users articulated their preferences).

[Stettinger et al. 2015b]

Exercises (6)

Graz University of Technology

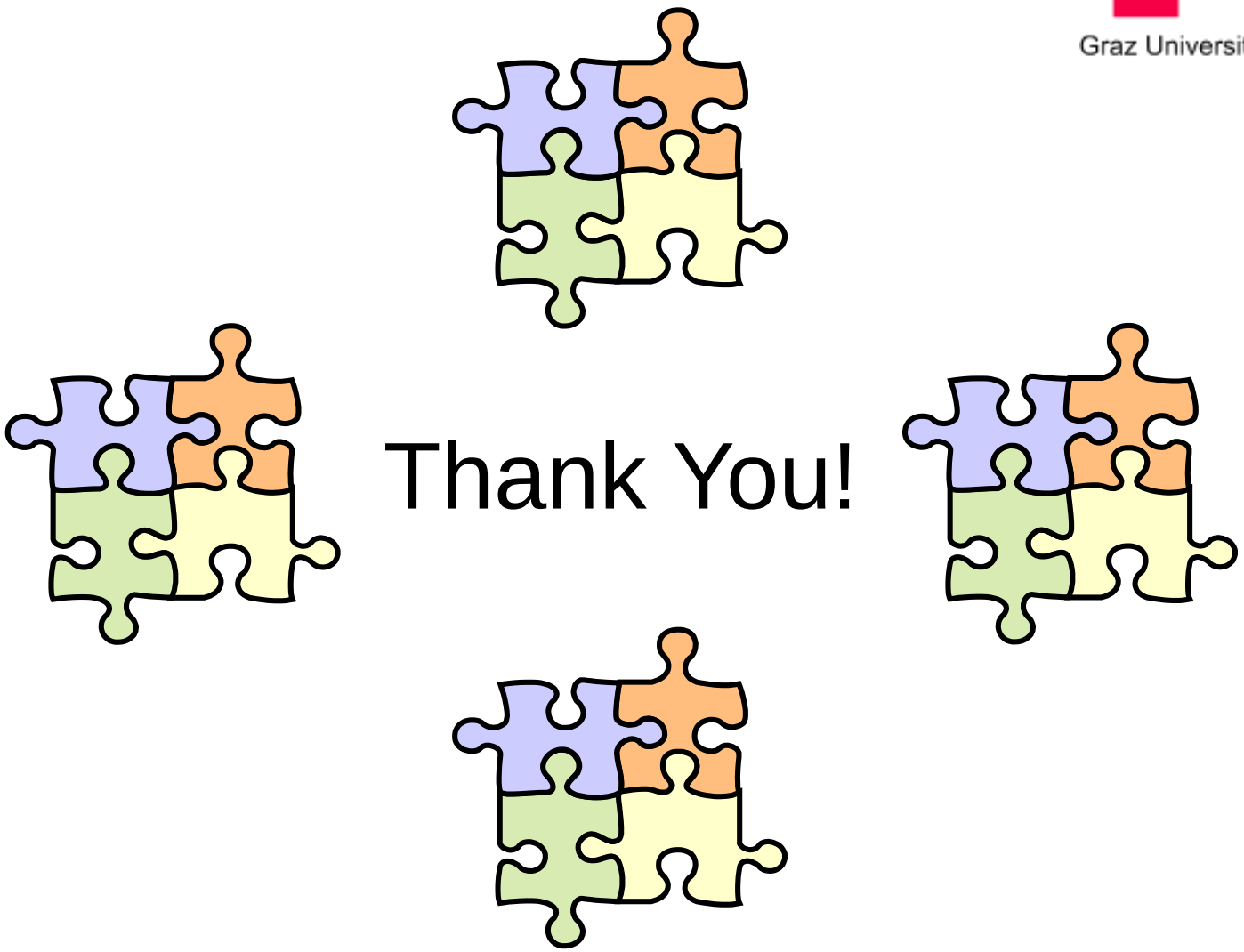
- Explain the term “hidden profiles”.
- Provide one example of a context effect (Asymmetric Dominance Effect).
- Provide an example of a negative and a positive salient description of a person.

Graz University of Technology

OpenReq Environment

innosensr.com

Graz University of Technology



Thank You!

References

Graz University of Technology

- D. Benavides, S. Segura, A. Cortés: Automated analysis of feature models 20 years later: A literature review. Inf. Syst. 35(6): 615-636, 2010.
- V. Chandola, A. Banerjee, and V. Kumar: Anomaly Detection: A Survey, ACM Computing Surveys, 41(3):15, 2009.
- C. Ebert, Systematisches Requirements Engineering, dpunkt, (2014).
- N. E´en and N. Sörensson: Translating Pseudo-Boolean Constraints into SAT, Journal on Satisfiability, Boolean Modeling and Computation 2:1-25, 2006.
- A. Felfernig, G. Ninaus, H. Grabner, F. Reinfrank, L. Weninger, D. Pagano, and W. Maalej, An overview of recommender systems in requirements engineering, in Managing Requirements Knowledge, pp. 315–332, (2013).
- A. Felfernig and G. Ninaus. Group Recommendation Algorithms for Requirements Prioritization, ICSE 2012 Workshop on Recommender Systems for Software Engineering (RSSE 2012), Zürich, Switzerland, pp. 1–4, (2012).
- A. Felfernig, W. Maalej, M. Mandl, F. Ricci, and M. Schubert, Recommendation and Decision Technologies For Requirements Engineering, ICSE 2010 Workshop on Recommender Systems in Software Engineering, Cape Town, South Africa, pp. 1-5, (2010).
- H. Hofmann and F. Lehner, Requirements Engineering as a success factor in software projects, IEEE Software,18(4):58–66, (2001).

References

Graz University of Technology

- D. Leffingwell, Calculating the return on investment from more effective requirements management, American Programmer, 10(4):13–16, (1997).
- S. Lim, D. Quercia, and A. Finkelstein, Stakenet: Using social net- works to analyse the stakeholders of large-scale software projects, 32nd ACM/IEEE Intl. Conf. on Software Eng., pp. 295–304, Cape Town, South Africa, (2010).
- W. Maalej, Z. Kurtanovic, and A. Felfernig, What Stakeholders Need to Know About Requirements, 4th IEEE Intl. Workshop on Empirical Requirements Engineering, Karlskrona, Sweden, (2014).
- G. Ninaus, F. Reinfrank, A. Felfernig, and M. Stettinger, Content-based Recommendation Techniques In Requirements Engineering, 1st Intl. Workshop on AI For Requirements Engineering, Karlskrona, Sweden, (2014a).
- G. Ninaus, A. Felfernig, M. Stettinger, S. Reiterer, G. Leitner, L. Weninger, and W. Schanil. IntelliReq: Intelligent Techniques for Software Requirements Engineering, European Conference on Artificial Intelligence, Prestigious Applications of Intelligent Systems (PAIS), pp. 1161-1166, (2014b).
- T. Simon, J. Streit, and M. Pizka “Practically Relevant Quality Criteria for requirements documents”, itestra GmbH, Ludwigstr. 35, 86916 Kaufering Germany.
- M. Stettinger, A. Felfernig, G. Leitner, S. Reiterer, and M. Jeran. Counteracting Serial Position Effects in the CHOICLA Group Decision Support Environment, 20th ACM Conference on Intelligent User Interfaces (IUI2015), pp. 148-157, Atlanta, Georgia, USA, (2015).
- M. Stettinger, A. Felfernig, G. Leitner, and S. Reiterer, Counteracting Anchoring Effects in Group Decision Making, UMAP 2015, to appear, Dublin Ireland, (2015b).