Karlsruhe Institute of Technology

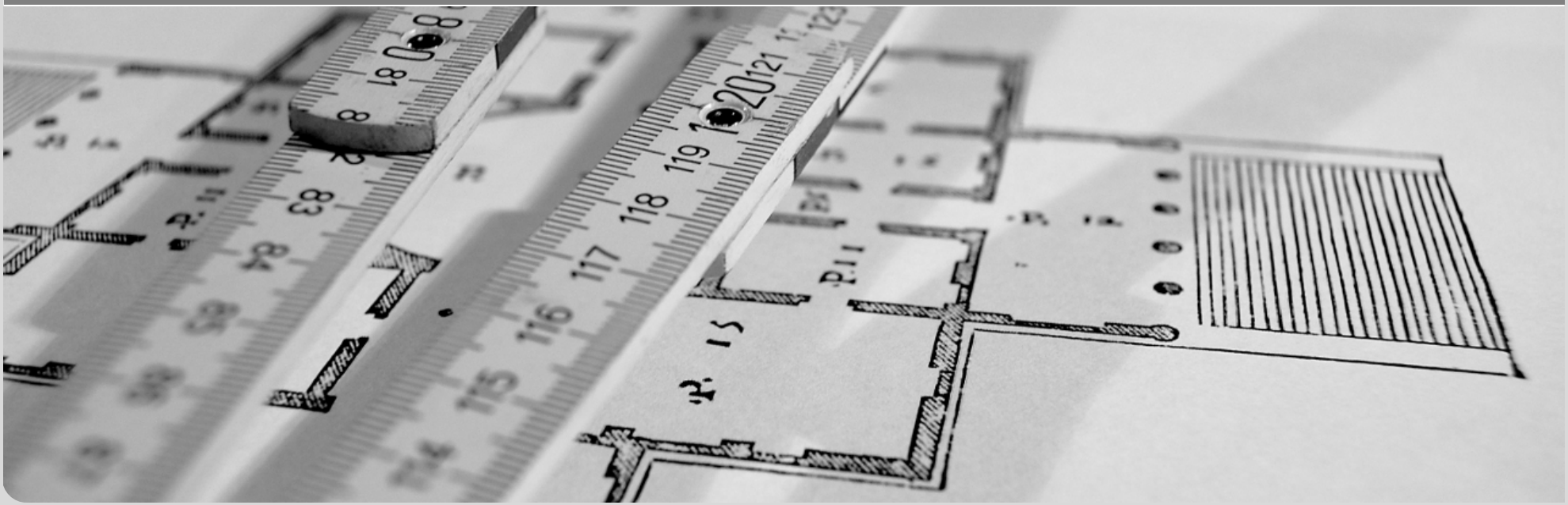
Software Design and Quality

Access Control Verification in Software Systems
Bachelor's thesis

Reviewer: Prof. Dr. Ralf H. Reussner, Jun.-Prof. Dr.-Ing. Anne Koziolek

Julian Hinrichs | October 5, 2018

CHAIR FOR SOFTWARE DESIGN AND QUALITY



Introduction

- Architectural security analyses.
  - Adapt the system model in an early design stage.
  - Avoid inconsistency between the security documentation and the system model.
- Different approaches: UMLSec (Jürjens 2002), Data-based privacy analysis(DBPA) (Seifermann 2016), etc.
- The evaluation of DBPA approaches is not carried out formally, but through case studies.
- It is not trivial to create case studies.
- Goal: support the creation of case studies to evaluate privacy defined by access rights.

Introduction

- Architectural security analyses.
  - Adapt the system model in an early design stage.
  - Avoid inconsistency between the security documentation and the system model.
- Different approaches: UMLSec (Jürjens 2002), Data-based privacy analysis(DBPA) (Seifermann 2016), etc.
- The evaluation of DBPA approaches is not carried out formally, but through case studies.
- It is not trivial to create case studies.
- Goal: support the creation of case studies to evaluate privacy defined by access rights.

Introduction

- Architectural security analyses.
  - Adapt the system model in an early design stage.
  - Avoid inconsistency between the security documentation and the system model.
- Different approaches: UMLSec (Jürjens 2002), Data-based privacy analysis(DBPA) (Seifermann 2016), etc.
- The evaluation of DBPA approaches is not carried out formally, but through case studies.
- It is not trivial to create case studies.
- Goal: support the creation of case studies to evaluate privacy defined by access rights.

Introduction

- Architectural security analyses.
  - Adapt the system model in an early design stage.
  - Avoid inconsistency between the security documentation and the system model.
- Different approaches: UMLSec (Jürjens 2002), Data-based privacy analysis(DBPA) (Seifermann 2016), etc.
- The evaluation of DBPA approaches is not carried out formally, but through case studies.
- It is not trivial to create case studies.
- Goal: support the creation of case studies to evaluate privacy defined by access rights.

Introduction

- Architectural security analyses.
  - Adapt the system model in an early design stage.
  - Avoid inconsistency between the security documentation and the system model.
- Different approaches: UMLSec (Jürjens 2002), Data-based privacy analysis(DBPA) (Seifermann 2016), etc.
- The evaluation of DBPA approaches is not carried out formally, but through case studies.
- It is not trivial to create case studies.
- Goal: support the creation of case studies to evaluate privacy defined by access rights.

Related work

- Case studies are already used in software engineering (Runeson and Höst 2009).
  - General purpose of a case study.
  - General process for creating a case study.
- Requirements for privacy: Non-influence (Oheimb 2004).

Related publication: Evered and Bögeholz 2004

- Definition of access rights in component-based systems.
- Example case study for a smaller scope.
- Measurement for good access rights.

Related work

- Case studies are already used in software engineering (Runeson and Höst 2009).
  - General purpose of a case study.
  - General process for creating a case study.
- Requirements for privacy: Non-influence (Oheimb 2004).

Related publication: Evered and Bögeholz 2004

- Definition of access rights in component-based systems.
- Example case study for a smaller scope.
- Measurement for good access rights.

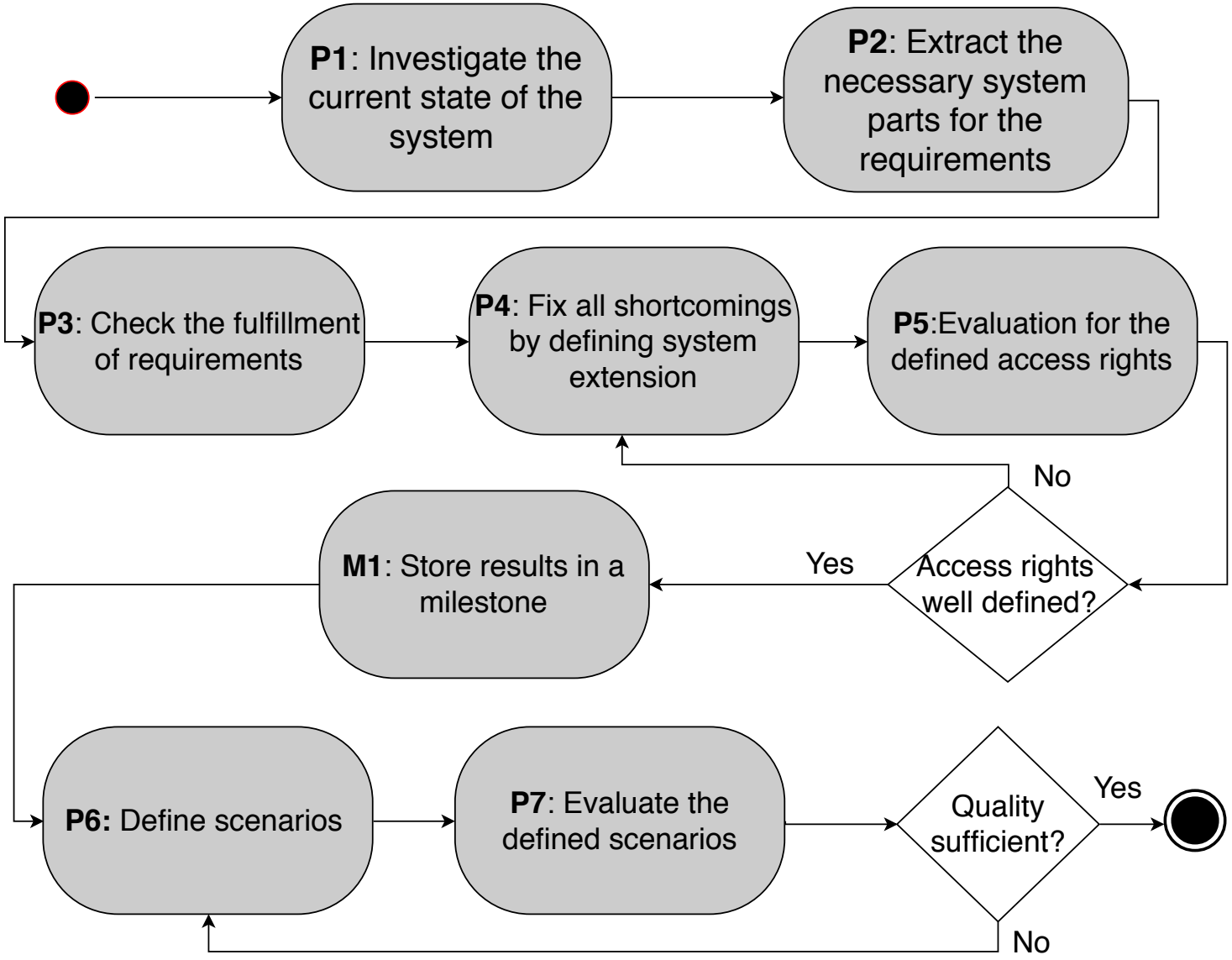
Related work

- Case studies are already used in software engineering (Runeson and Höst 2009).
  - General purpose of a case study.
  - General process for creating a case study.
- Requirements for privacy: Non-influence (Oheimb 2004).

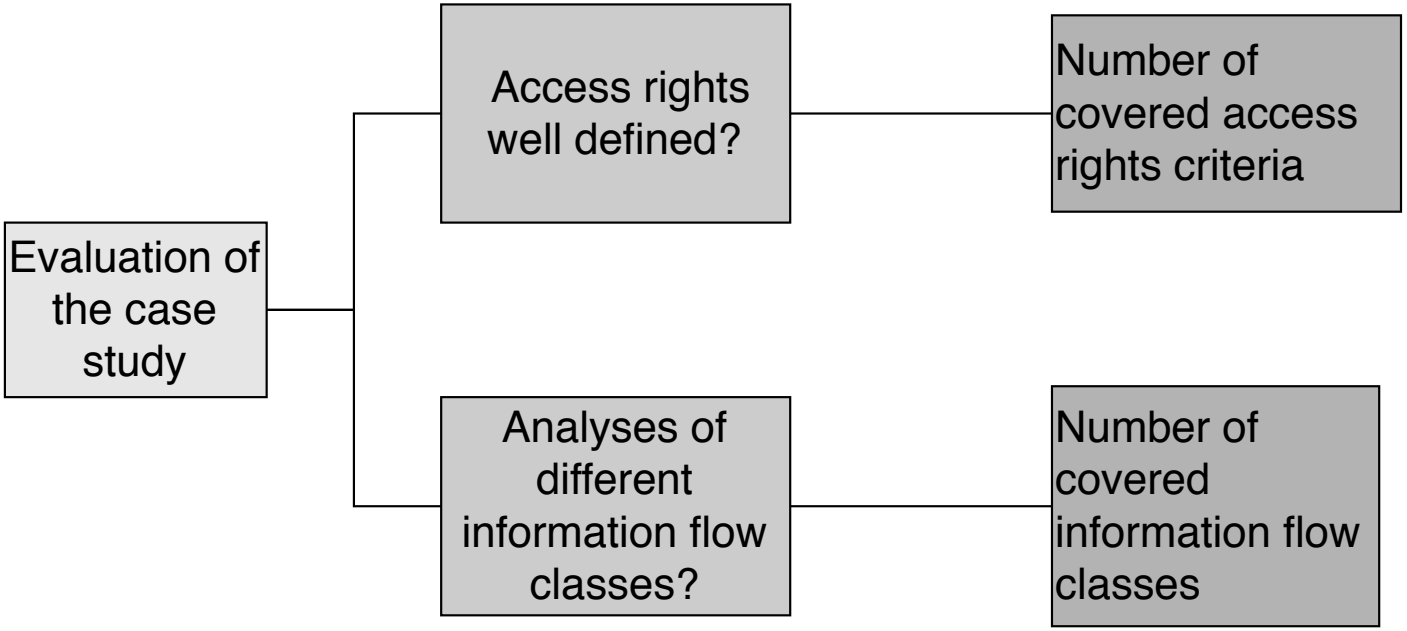
Related publication: Evered and Bögeholz 2004

- Definition of access rights in component-based systems.
- Example case study for a smaller scope.
- Measurement for good access rights.

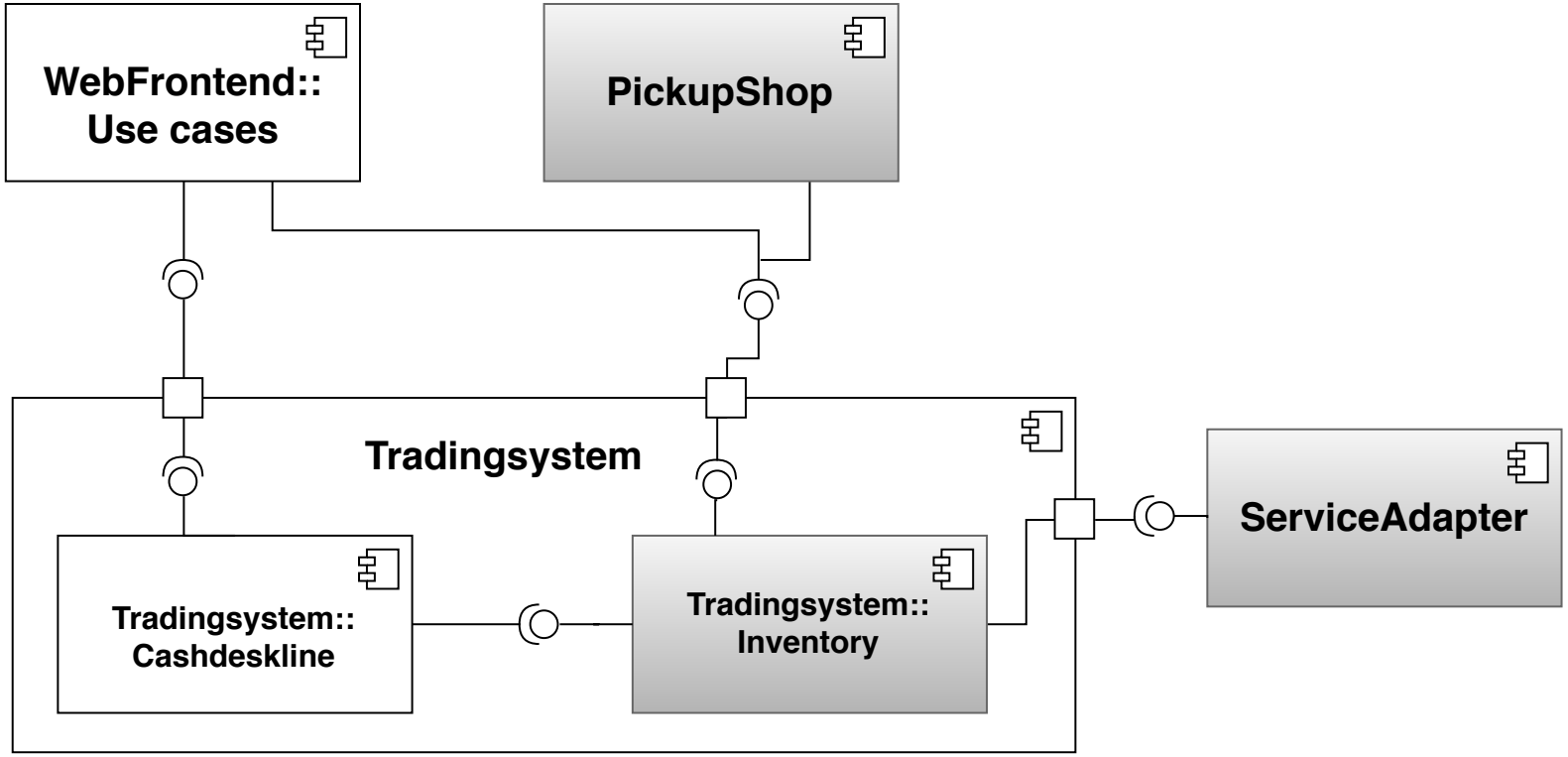
Procedure Overview



Evaluation of the case study



P1: Investigate the current state of CoCoME



P2: Requirements for privacy-considering case study

Requirements	
R1	Component-based system
R2	Definition of use cases
R3	Security relevant data
R4	Definition of user roles
R5	Definition of access rights
R6	Definition of the type of data processing in the components

Procedure P2 -P4
Requirements R1-R4

- ✓: documented, ▲: defined, ●: generated/ derived
- R1: Component based system. ✓
- R2: 13 use cases are defined in the documentation. ✓
- R3: Security relevant data. ▲
  - Four different classes for the data in CoCoME.
  - The security relevance for each class was measured according to Breier 2014.
  - Account data: security relevant
  - Customer data: security relevant
  - System data: security relevant
  - P&S data: security relevant in composition with one of the other classes.
- R4: 6 roles are defined in the documentation. ✓— ▲

Procedure P2 -P4
Requirements R1-R4

- ✓: documented, ▲: defined, ●: generated/ derived
- R1: Component based system. ✓
- R2: 13 use cases are defined in the documentation. ✓
- R3: Security relevant data. ▲
  - Four different classes for the data in CoCoME.
  - The security relevance for each class was measured according to Breier 2014.
  - Account data: security relevant
  - Customer data: security relevant
  - System data: security relevant
  - P&S data: security relevant in composition with one of the other classes.
- R4: 6 roles are defined in the documentation. ✓— ▲

Procedure P2 -P4
Requirements R1-R4

- ✓: documented, ▲: defined, ●: generated/ derived
- R1: Component based system. ✓
- R2: 13 use cases are defined in the documentation. ✓
- R3: Security relevant data. ▲
  - Four different classes for the data in CoCoME.
  - The security relevance for each class was measured according to Breier 2014.
  - Account data: security relevant
  - Customer data: security relevant
  - System data: security relevant
  - P&S data: security relevant in composition with one of the other classes.
- R4: 6 roles are defined in the documentation. ✓— ▲

Procedure P2 -P4
Requirements R1-R4

- ✓: documented, ▲: defined, ●: generated/ derived
- R1: Component based system. ✓
- R2: 13 use cases are defined in the documentation. ✓
- R3: Security relevant data. ▲
  - Four different classes for the data in CoCoME.
  - The security relevance for each class was measured according to Breier 2014.
  - Account data: security relevant
  - Customer data: security relevant
  - System data: security relevant
  - P&S data: security relevant in composition with one of the other classes.
- R4: 6 roles are defined in the documentation. ✓— ▲

Procedure P2 -P4
Requirements R1-R4

- ✓: documented, ▲: defined, ●: generated/ derived
- R1: Component based system. ✓
- R2: 13 use cases are defined in the documentation. ✓
- R3: Security relevant data. ▲
  - Four different classes for the data in CoCoME.
  - The security relevance for each class was measured according to Breier 2014.
  - Account data: security relevant
  - Customer data: security relevant
  - System data: security relevant
  - P&S data: security relevant in composition with one of the other classes.
- R4: 6 roles are defined in the documentation. ✓— ▲

Procedure P2-P4
R5: access rights ●

- Finer grained, high level form derived from (Evered and Bögeholz 2004).
- Access control matrix (ACM)
  - Level 1: **fullAccess**
  - Level 2: **AccessToUsedData**
  - Level 3: **AccessToOwnData**
  - Level 4: **default**

Roles	Webfrontend		TS:Inventory	
StockManager	Customer data	4	Customer data	4
	Account data	3	Account data	3
	P&S data	2	P&S data	2
	System data	4	System data	4

Procedure P2-P4
R5: access rights ●

- Finer grained, high level form derived from (Evered and Bögeholz 2004).
- Access control matrix (ACM)
  - Level 1: **fullAccess**
  - Level 2: **AccessToUsedData**
  - Level 3: **AccessToOwnData**
  - Level 4: **default**

Roles	Webfrontend		TS:Inventory	
StockManager	Customer data	4	Customer data	4
	Account data	3	Account data	3
	P&S data	2	P&S data	2
	System data	4	System data	4

Procedure P2-P4
R6: types of data processing in the system ●

- We identified four categories of data processing in CoCoME.
  - Transmission of data
  - alternation of data
  - relational algebra
  - I/O
- Operations matrix(OpM)

Components	customer	account	P&S	system
Webfrontend	transmit	transmit	I/O, transmit	n/a

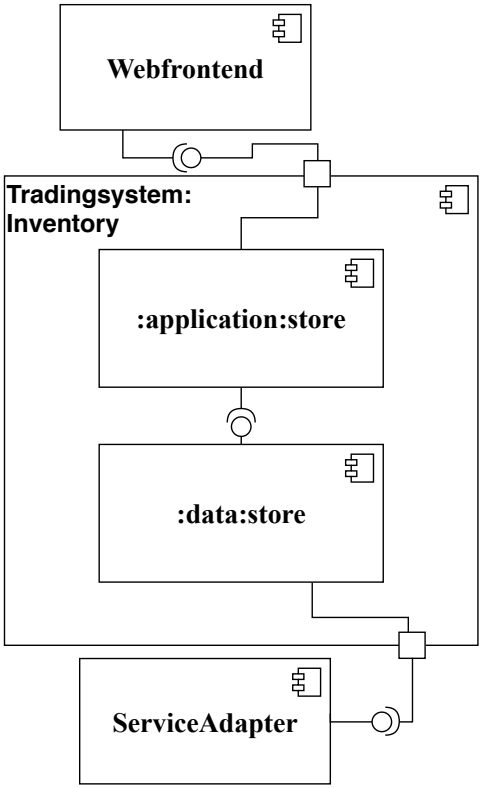
Procedure P2-P4
R6: types of data processing in the system ●

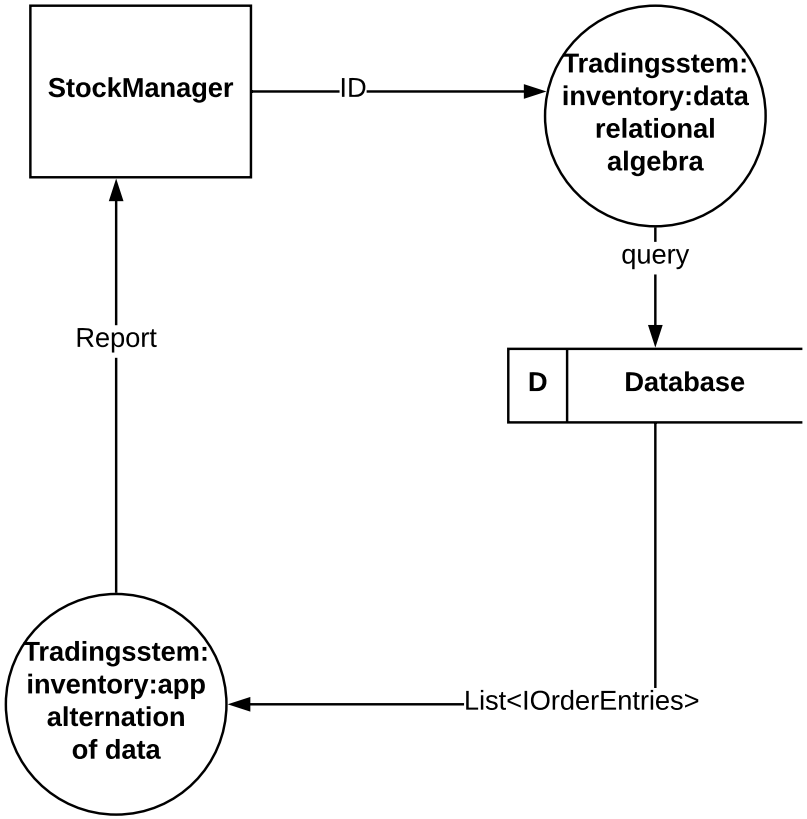
- We identified four categories of data processing in CoCoME.
  - Transmission of data
  - alternation of data
  - relational algebra
  - I/O
- Operations matrix(OpM)

Components	customer	account	P&S	system
Webfrontend	transmit	transmit	I/O, transmit	n/a

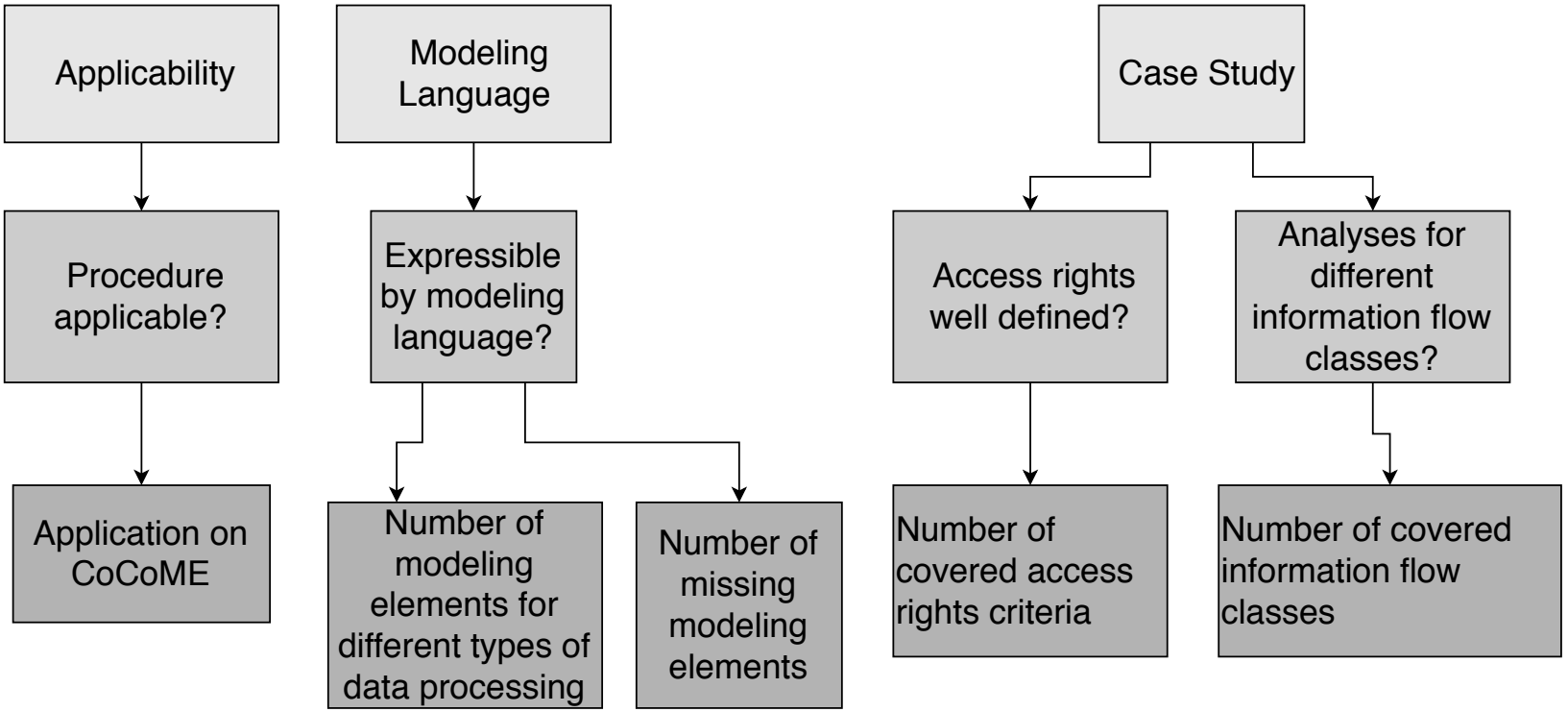
Procedure P6: Definition of a scenario

- Scenario: StockManager requests a report for the purchased products of a customer.





Goal-Question-Metric plan



Evaluation for the quality of the access rights

- Evered and Bögeholz defined seven criteria to measure the quality of access rights.

Access rights		fulfilled
Specification	Aspect-oriented	✓
	Positive	✓
	Need-to-know	✓
Comprehensibility	Clear	?
	Concise	?
Implementation	Fundamental	n/a
	Efficient	n/a

Evaluation of covered information flow classes

- Problem statement: Non-influence = non-interference + non-leakage (Oheimb 2004).
  - Non-interference: High data inputs in the program flow have no effect on low data outputs.
  - Non-leakage: Unobservable if certain actions have taken place.

Data flow	fulfilled?
Illegal information flow	✓
Information flow from high to low	✓
Direct information flow between roles	✗
No observable information flow	✗

Evaluation of covered information flow classes

- Problem statement: Non-influence = non-interference + non-leakage (Oheimb 2004).
  - Non-interference: High data inputs in the program flow have no effect on low data outputs.
  - Non-leakage: Unobservable if certain actions have taken place.

Data flow	fulfilled?
Illegal information flow	✓
Information flow from high to low	✓
Direct information flow between roles	✗
No observable information flow	✗

Threats to validity

Internal Validity	External Validity	Construct Validity	Conclusion Validity
II, III	I	II	III

- I: Not applied to various systems.
- II: Not all access rights criteria were checked.
- III: Not all information flow classes are covered.

Future work

- Evaluation of the procedure:
  - Create a case study for the complete CoCoME system.
  - Apply the procedure to other systems (e.g Travelsystem (Katkalov et al. 2013)) and create further case studies.
- Case study
  - Short term work
    - Evaluate the criteria *concise* and *clear*.
    - Define additional scenarios to cover all information flow classes.
  - Long term work
    - Evaluate the criteria *fundamental* and *efficient*.
    - Definition of further information flow classes other than non-influence.
    - Using the case study for evaluating a data based privacy analysis.

PIBA

- Problem
  - Usable case studies for evaluating data-based privacy analysis (DBPA) are difficult to create.
- Idea
  - Introduce a method for creating usable case studies for DBPA approaches.
- Benefit
  - Comparability for different privacy analysis approaches.
- Actions
  - Create a method for the creation of case studies.
  - Apply the method to a system.
  - Evaluate the created case study.

References I

Jakub Breier. “Asset Valuation Method for Dependent Entities”. In: *J. Internet Serv. Inf. Secur.* 4.3 (2014), pp. 72–81. URL: http://isyou.info/jisis/vol4/no3/jisis-2014-vol4-no3-05.pdf.

Mark Evered and Serge Bögeholz. “A Case Study in Access Control Requirements for a Health Information System”. In: *ACSW Frontiers 2004, 2004 ACSW Workshops - the Australasian Information Security Workshop (AISW2004), the Australasian Workshop on Data Mining and Web Intelligence (DMWI2004), and the Australasian Workshop on Software Internationalisation (AWSI2004) . Dunedin, New Zealand, January 2004*. 2004, pp. 53–61. URL: http://crpit.com/confpapers/CRPITV32Evered.pdf.

References II

Jan Jürjens. “UMLsec: Extending UML for Secure Systems Development”. In: *UML 2002 - The Unified Modeling Language, 5th International Conference, Dresden, Germany, September 30 - October 4, 2002, Proceedings*. 2002, pp. 412–425. DOI: 10.1007/3-540-45800-X_32. URL: https://doi.org/10.1007/3-540-45800-X_32.

Kuzman Katkalov et al. “Model-Driven Development of Information Flow-Secure Systems with IFlow”. In: *International Conference on Social Computing, SocialCom 2013, SocialCom/PASSAT/BigData/EconCom/BioMedCom 2013, Washington, DC, USA, 8-14 September, 2013*. 2013, pp. 51–56. DOI: 10.1109/SocialCom.2013.14. URL: https://doi.org/10.1109/SocialCom.2013.14.

References III

David von Oheimb. “Information Flow Control Revisited: Noninfluence = Noninterference + Nonleakage”. In: *Computer Security - ESORICS 2004, 9th European Symposium on Research Computer Security, Sophia Antipolis, France, September 13-15, 2004, Proceedings*. 2004, pp. 225–243. DOI: 10.1007/978-3-540-30108-0_14. URL: https://doi.org/10.1007/978-3-540-30108-0%5C_14.

Per Runeson and Martin Höst. “Guidelines for conducting and reporting case study research in software engineering”. In: *Empirical Software Engineering* 14.2 (2009), pp. 131–164. DOI: 10.1007/s10664-008-9102-8. URL: https://doi.org/10.1007/s10664-008-9102-8.

References IV

Stephan Seifermann. “Architectural Data Flow Analysis”. In: *13th Working IEEE/IFIP Conference on Software Architecture, WICSA 2016, Venice, Italy, April 5-8, 2016*. 2016, pp. 270–271. DOI: 10.1109/WICSA.2016.49. URL: https://doi.org/10.1109/WICSA.2016.49.

Evaluation Modeling language

Meta model	possible ?
relational algebra	yes
I/O operations	yes
Transmission of data	yes
Change of access rights	yes
Alternation of data	yes
ACM in system model	no

Operations matrix complete

Types of data processing	customer	account	P& S	system
Webfrontend	transmit	transmit	I/O transmit	n/a
PickupShop	transmit	transmit	I/O, transmit	n/a
Tradingsystem: inventory:app	change transmit	change transmit	change	n/a
Tradingsystem: inventory:data	rel. algebra operations	rel. algebra operations	rel. algebra operations	change
Tradingssystem: cashdeskline	change transmit	non-existent	change transmit	n/a

References

Definition of the value of an asset

- Different assets in system are related to each other.
- The assets are categorized in different levels. The value of an asset to the system is decreasing with descending numbers.
- A higher level is more crucial to protect for the system than the lower levels.
- In CoCoME:
  - Level 1: Customer and account data
  - Level 2: System and P& S data

Conclusion of the Procedure

- In the current state, we would argue it depends on the use of the resulting case study.
- Conclusion of the procedure
  - Access rights:
    - Concluded, further fulfillment of the criteria were not possible due to time constraints.
  - Information flow classes
    - If the covered information flow classes are sufficient for the intended use use of the case study
- No Conclusion of the procedure
  - Information flow classes are not covered yet.