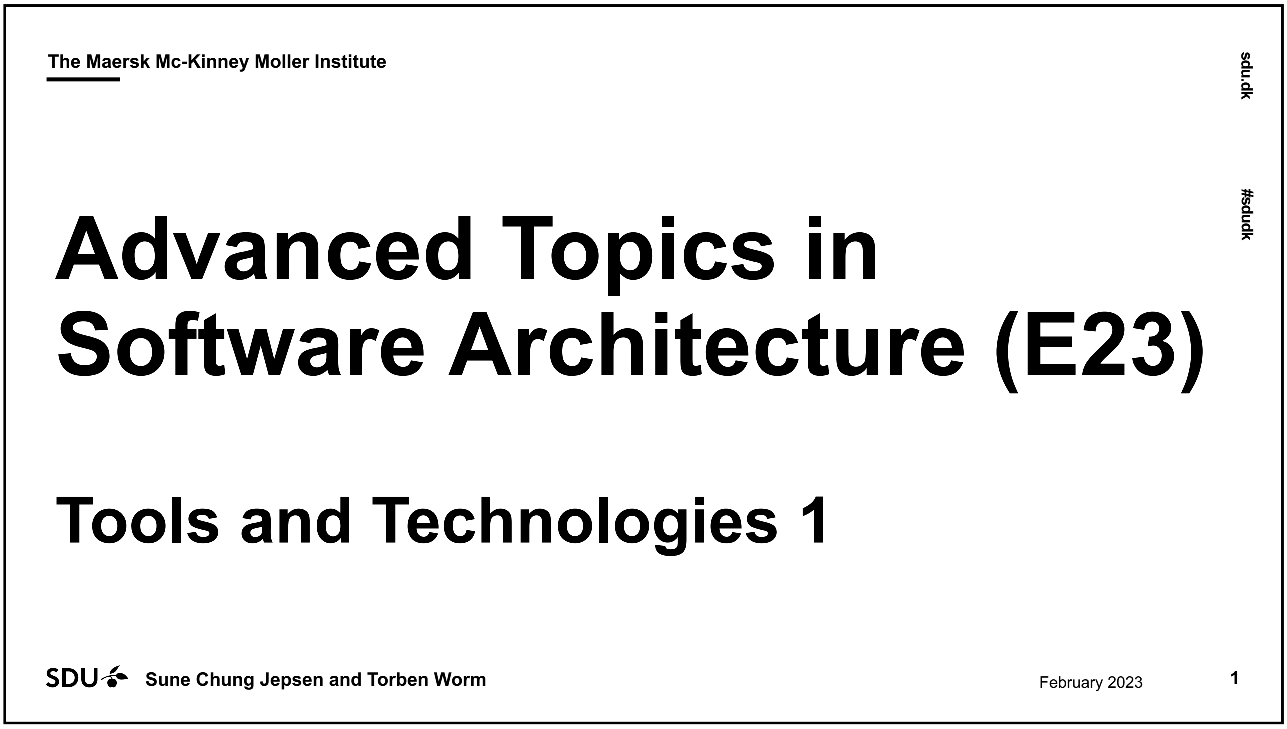


The Maersk Mc-Kinney Moller Institute

sdu.dk

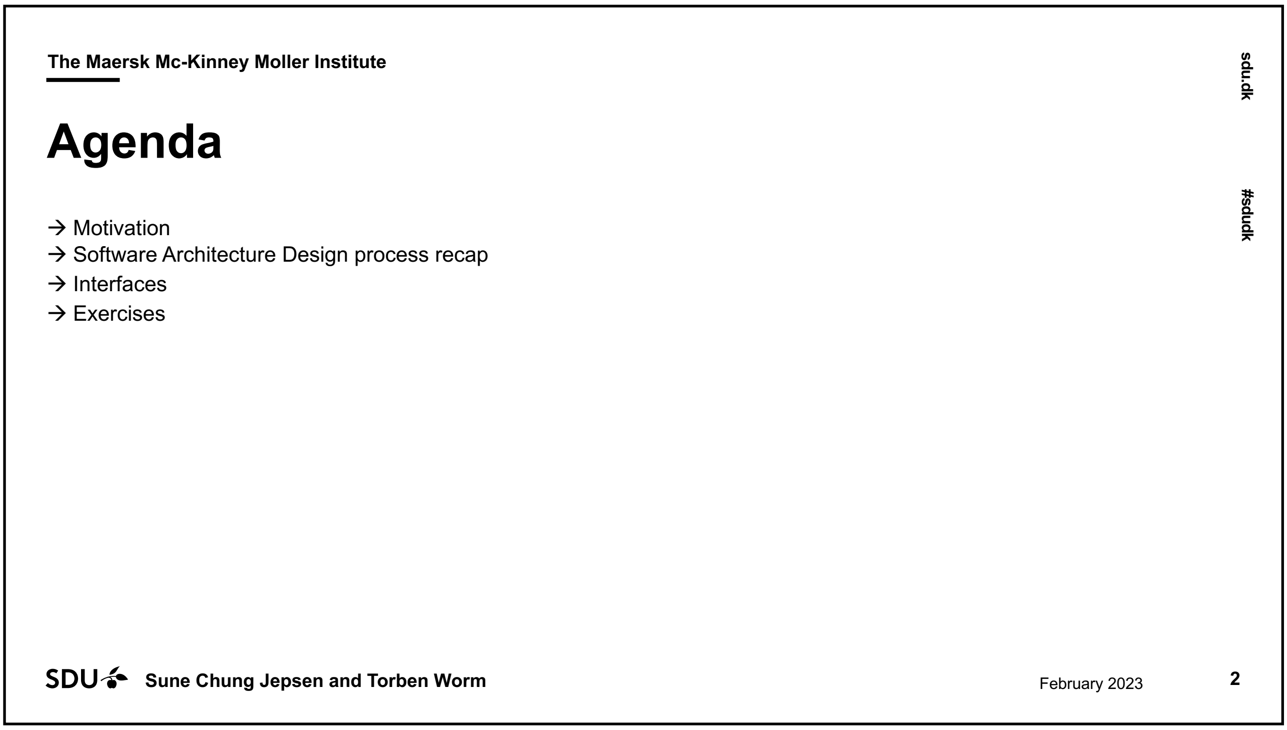
#sdudk

Advanced Topics in Software Architecture (E23)

Tools and Technologies 1

SDU Sune Chung Jepsen and Torben Worm

February 2023



The Maersk Mc-Kinney Moller Institute

sdu.dk

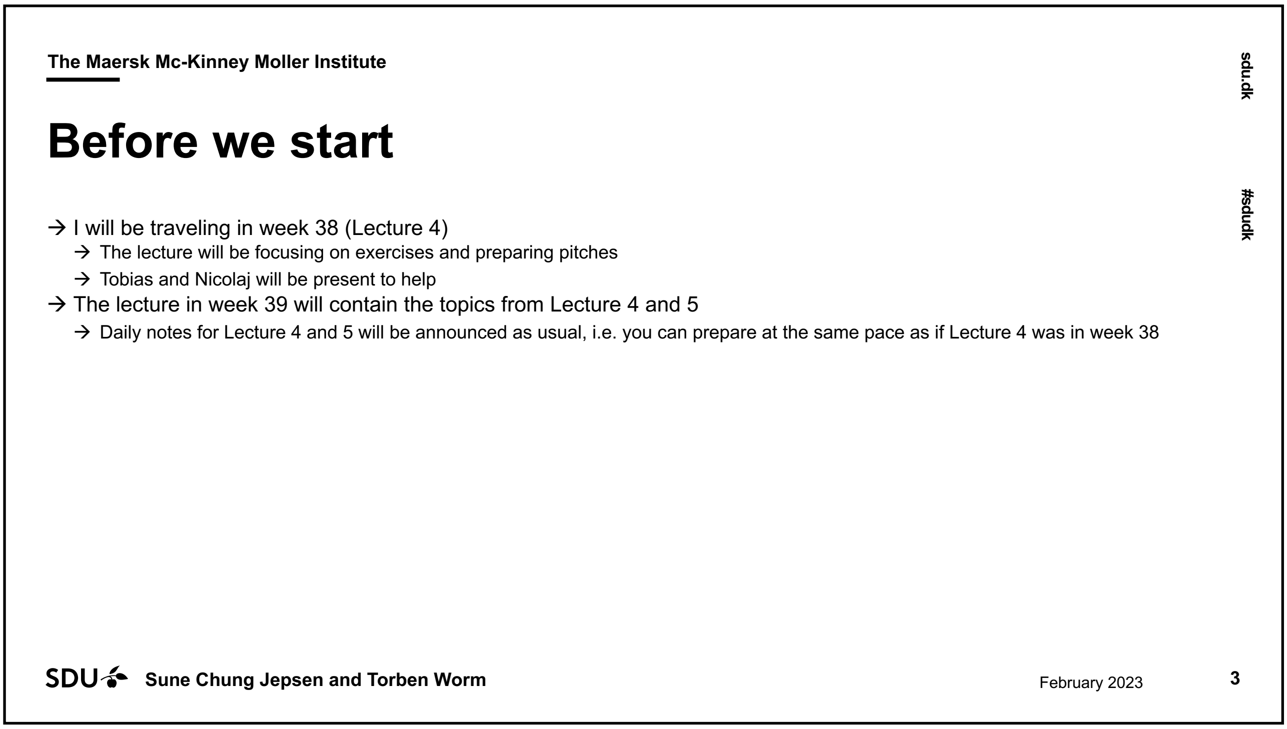
#sdudk

Agenda

→ Motivation
→ Software Architecture Design process recap
→ Interfaces
→ Exercises

SDU Sune Chung Jepsen and Torben Worm

February 2023



The Maersk Mc-Kinney Moller Institute

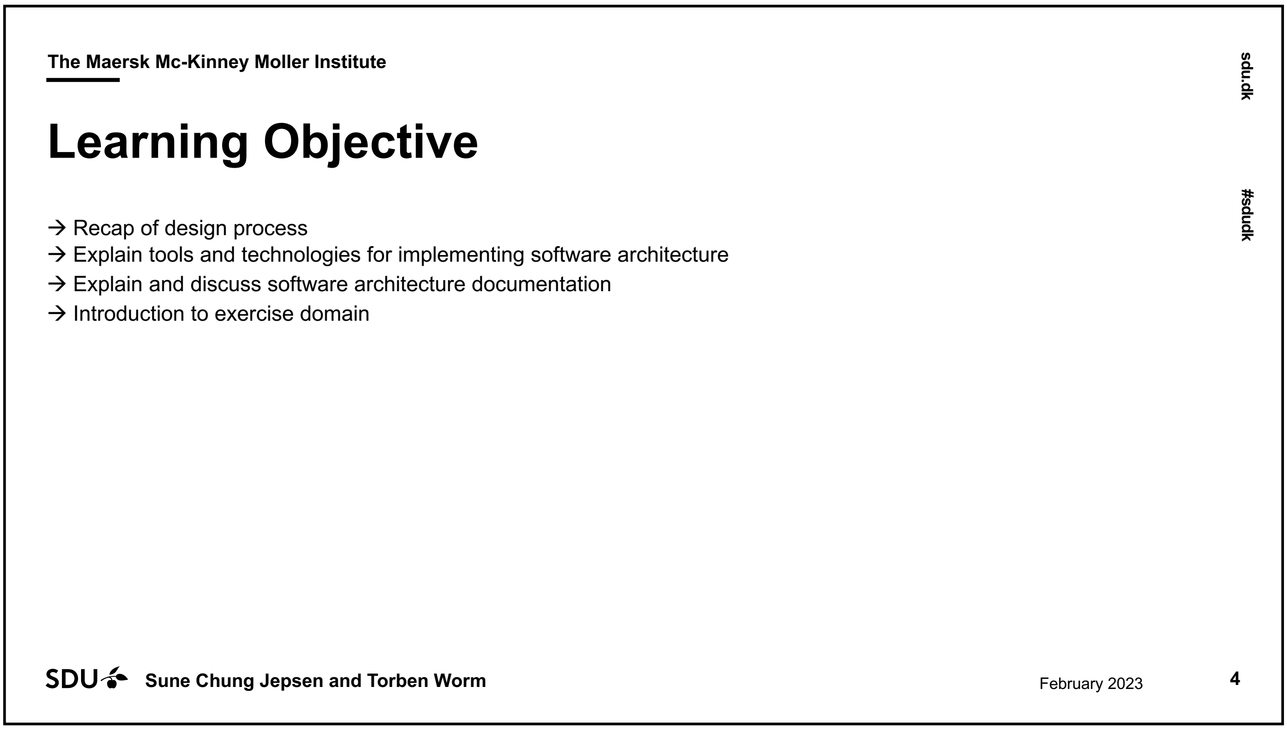
Before we start

- → I will be traveling in week 38 (Lecture 4)
  - → The lecture will be focusing on exercises and preparing pitches
  - → Tobias and Nicolaj will be present to help
- → The lecture in week 39 will contain the topics from Lecture 4 and 5
  - → Daily notes for Lecture 4 and 5 will be announced as usual, i.e. you can prepare at the same pace as if Lecture 4 was in week 38

sdu.dk #sdudk

SDU Sune Chung Jepsen and Torben Worm February 2023 3

3



The Maersk Mc-Kinney Moller Institute

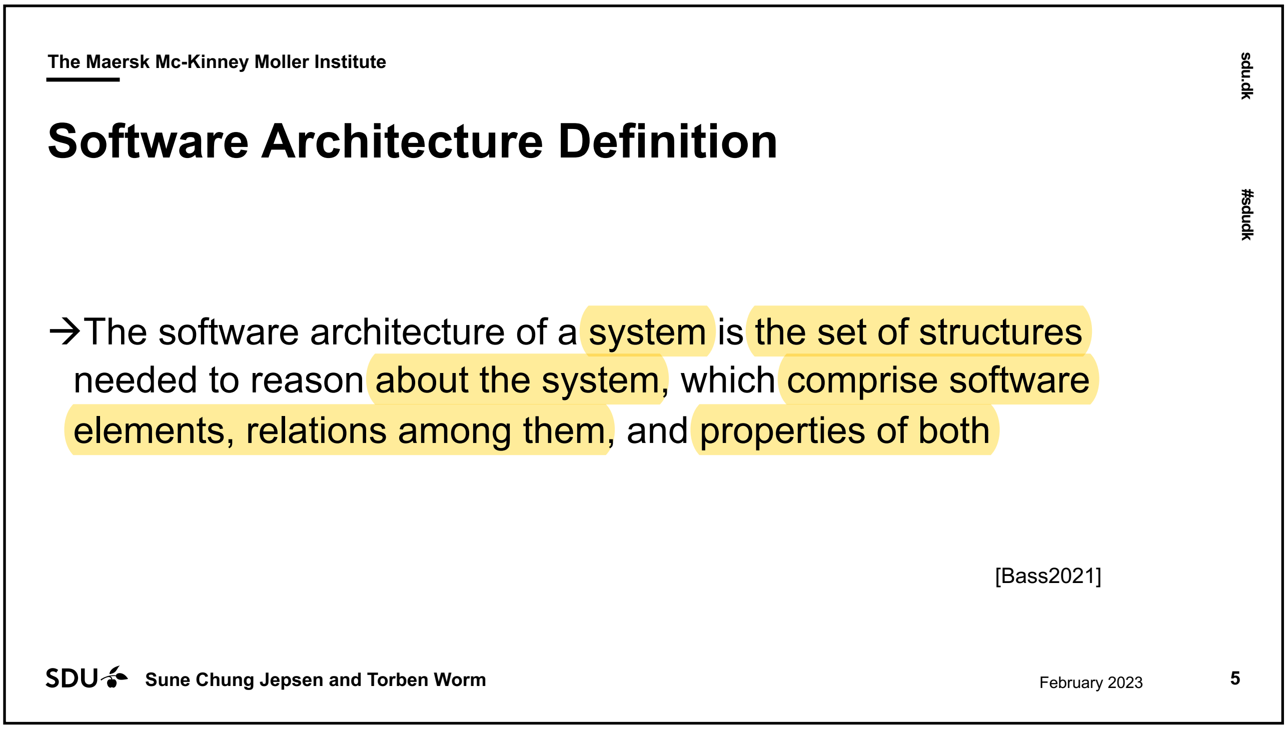
Learning Objective

- → Recap of design process
- → Explain tools and technologies for implementing software architecture
- → Explain and discuss software architecture documentation
- → Introduction to exercise domain

sdu.dk #sdudk

SDU Sune Chung Jepsen and Torben Worm February 2023 4

4



The Maersk Mc-Kinney Moller Institute

Software Architecture Definition

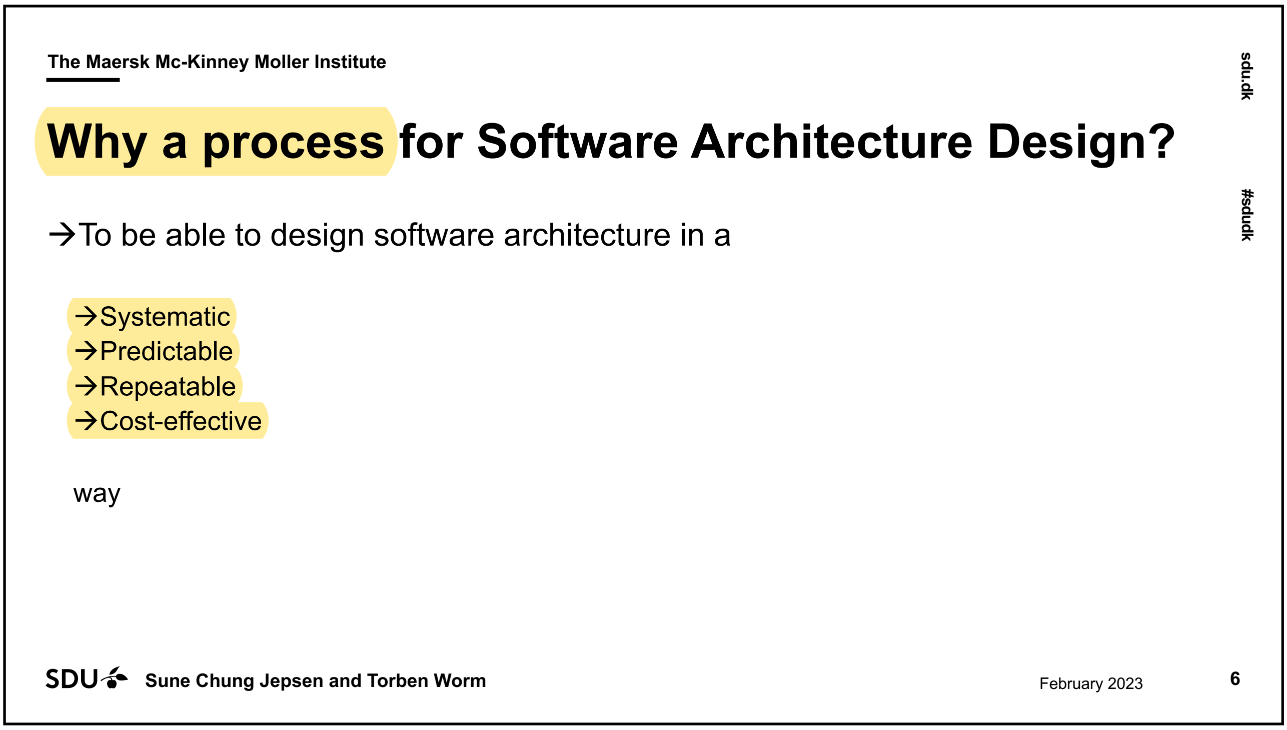
→The software architecture of a system is the set of structures needed to reason about the system, which comprise software elements, relations among them, and properties of both

[Bass2021]

sdu.dk #sdudk

SDU Sune Chung Jepsen and Torben Worm February 2023 5

5



The Maersk Mc-Kinney Moller Institute

Why a process for Software Architecture Design?

→To be able to design software architecture in a

→Systematic
→Predictable
→Repeatable
→Cost-effective

way

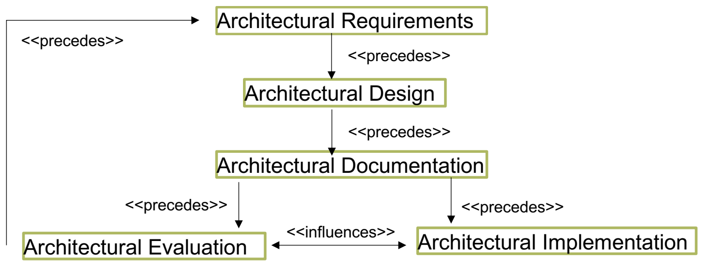
sdu.dk #sdudk

SDU Sune Chung Jepsen and Torben Worm February 2023 6

6

The Maersk Mc-Kinney Moller Institute

Software Architecture Life-cycle Activites



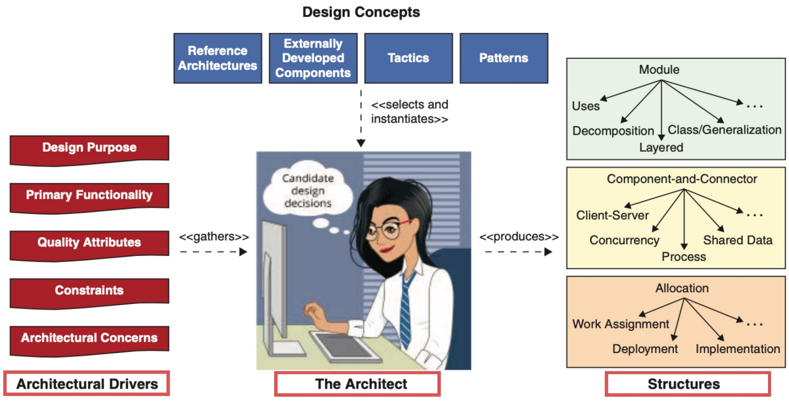
sdu.dk #sdudk

SDU Sune Chung Jepsen and Torben Worm February 2023 7

7

The Maersk Mc-Kinney Moller Institute

Architecture Design Activity

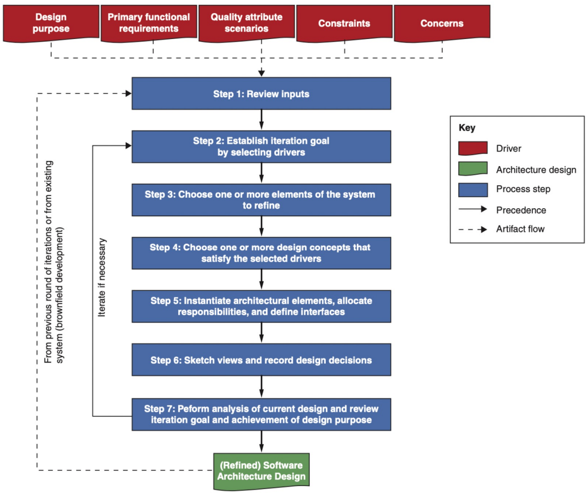


sdu.dk #sdudk

SDU Sune Chung Jepsen and Torben Worm February 2023 8

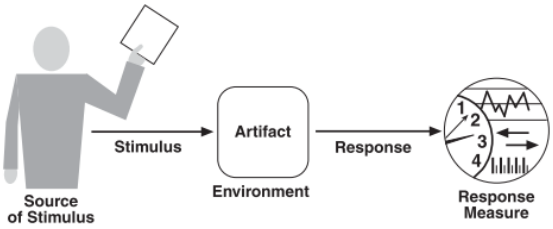
8

Steps and Artifacts of ADD


Design purpose
Primary functional requirements
Quality attribute scenarios
Constraints
Concerns
Step 1: Review inputs
Step 2: Establish iteration goal by selecting drivers
Step 3: Choose one or more elements of the system to refine
Step 4: Choose one or more design concepts that satisfy the selected drivers
Step 5: Instantiate architectural elements, allocate responsibilities, and define interfaces
Step 6: Sketch views and record design decisions
Step 7: Peform analysis of current design and review iteration goal and achievement of design purpose
(Refined) Software Architecture Design
From previous round of iterations or from existing system (brownfield development)
Iterate if necessary
Key
Driver
Architecture design
Process step
Precedence
Artifact flow

- Design purpose:
  - To create arhitectural prototype(S)
- Primary FR
  - For you to determine
  - Primary use cases
- QAS
  - See exercise – and add new if needed
- Constraints
- Concerns

Quality Attribute Scenario



→ Quality Attribute Workshop (QAW)

→ Mission Thread Workshop
  → Systems of Systems

→ Utility Tree

→ Prioritization
  → Importance (L/M/H)
    → Ranked by the customer
  → Technical Risk (L/M/H)
    → Ranked by the architect

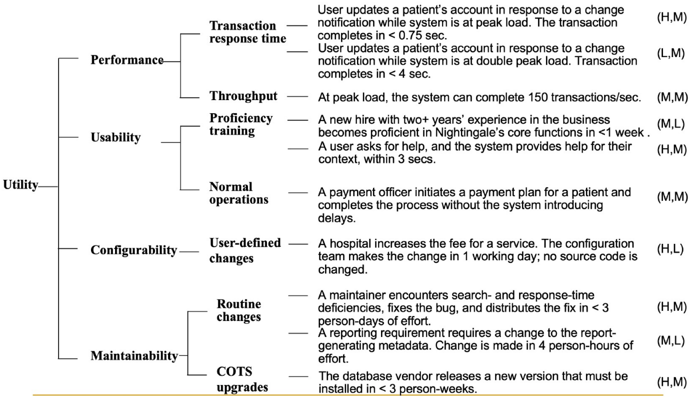
  → Select (H,H), (H,M), (M,H)

The Maersk Mc-Kinney Moller Institute

sdu.dk

#sdudk

Utility tree



Utility			
Performance	Transaction response time	User updates a patient's account in response to a change notification while system is at peak load. The transaction completes in < 0.75 sec.	(H,M)
		User updates a patient's account in response to a change notification while system is at double peak load. Transaction completes in < 4 sec.	(L,M)
	Throughput	At peak load, the system can complete 150 transactions/sec.	(M,M)
Usability	Proficiency training	A new hire with two+ years' experience in the business becomes proficient in Nightingale's core functions in <1 week .	(M,L)
		A user asks for help, and the system provides help for their context, within 3 secs.	(H,M)
	Normal operations	A payment officer initiates a payment plan for a patient and completes the process without the system introducing delays.	(M,M)
Configurability	User-defined changes	A hospital increases the fee for a service. The configuration team makes the change in 1 working day; no source code is changed.	(H,L)
Maintainability	Routine changes	A maintainer encounters search- and response-time deficiencies, fixes the bug, and distributes the fix in < 3 person-days of effort.	(H,M)
		A reporting requirement requires a change to the report-generating metadata. Change is made in 4 person-hours of effort.	(M,L)
	COTS upgrades	The database vendor releases a new version that must be installed in < 3 person-weeks.	(H,M)

SDU Sune Chung Jepsen and Torben Worm

February 2023

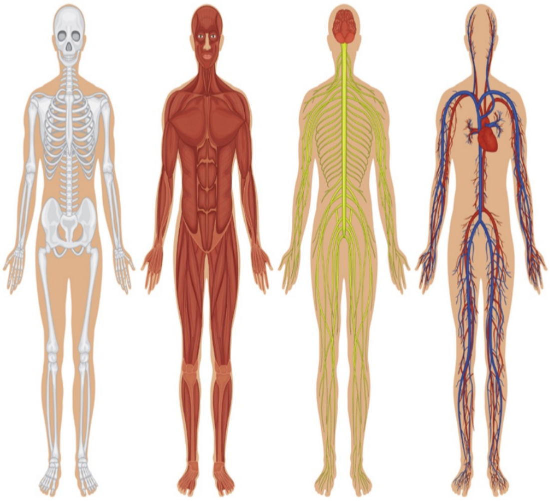
The Maersk Mc-Kinney Moller Institute

sdu.dk

#sdudk

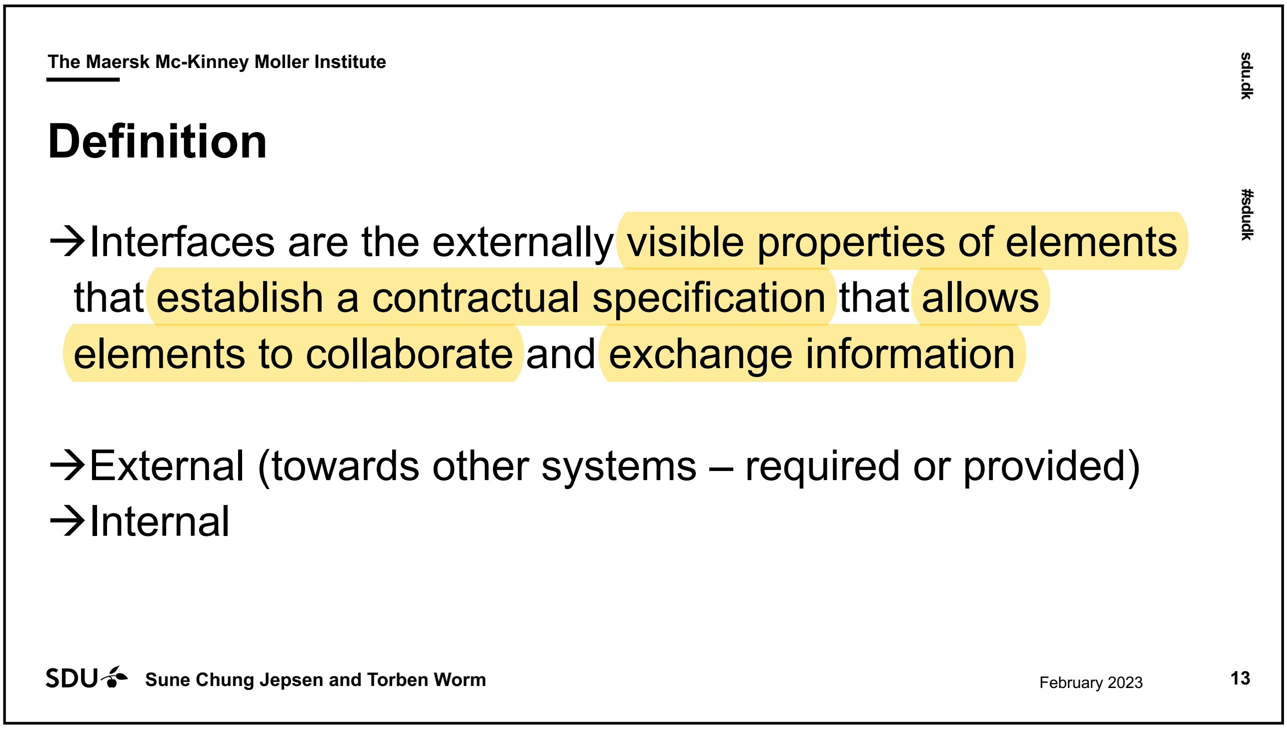
Motivation

→ Systems and subsystem
→ Parallel processing
→ Communication
→ Systems grows
→ Becomes increasingly complex
→ Decoupling allows
  → Adding
  → Removing
  → Maintain
→ Integrity of the system



SDU Sune Chung Jepsen and Torben Worm

February 2023



The Maersk Mc-Kinney Moller Institute

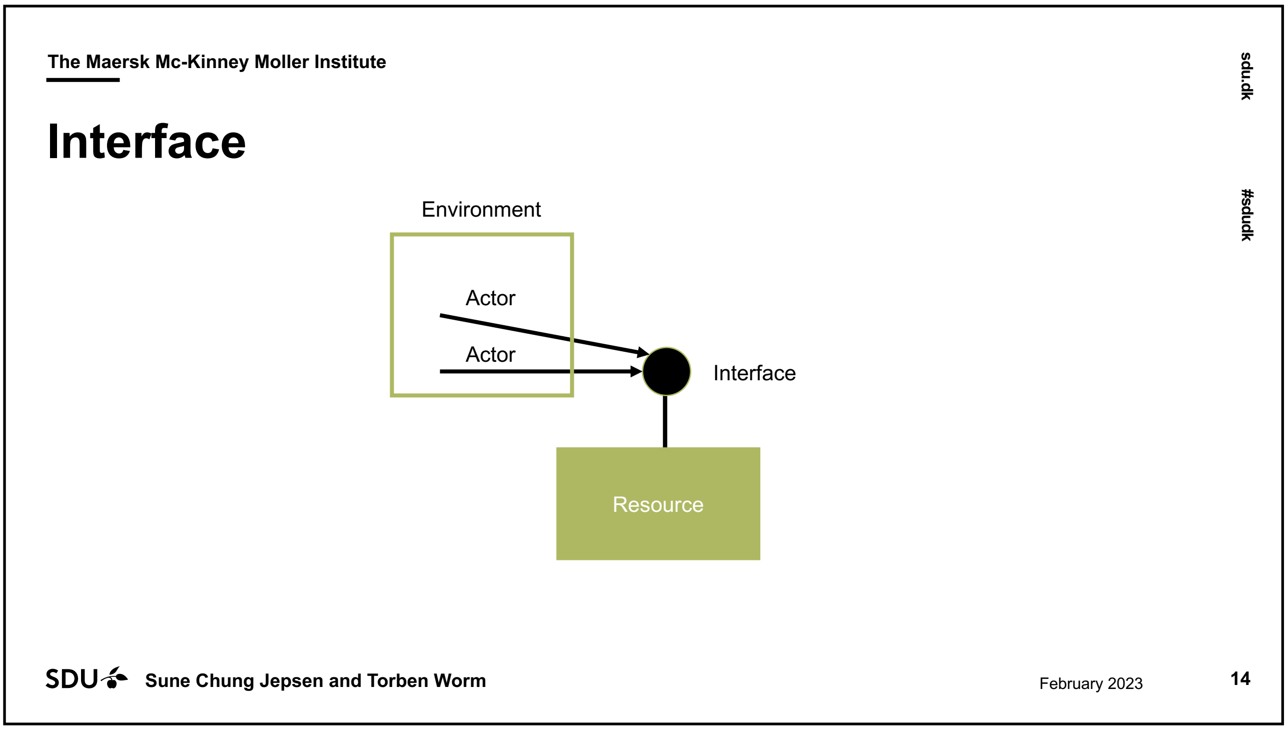
sdu.dk #sdudk

Definition

→Interfaces are the externally visible properties of elements that establish a contractual specification that allows elements to collaborate and exchange information

→External (towards other systems – required or provided)
→Internal

13



The Maersk Mc-Kinney Moller Institute

sdu.dk #sdudk

Interface

Environment

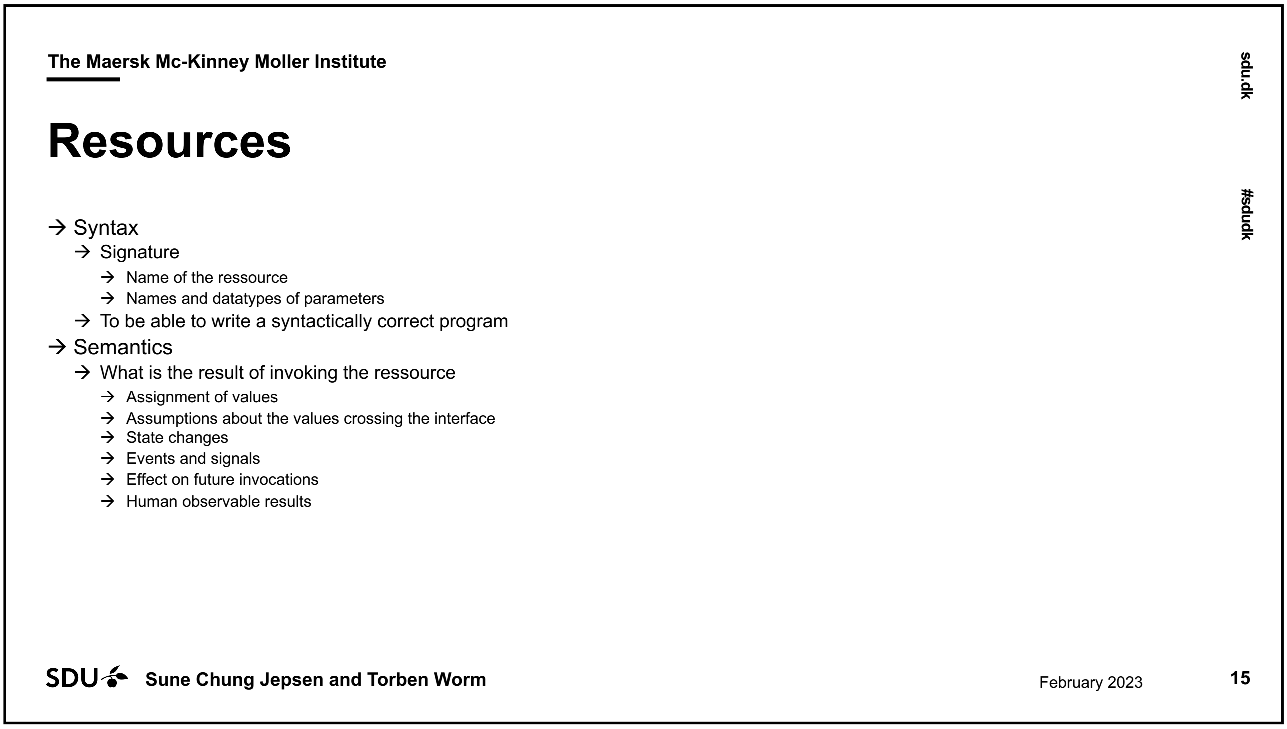
Actor

Actor

Interface

Resource

14



The Maersk Mc-Kinney Moller Institute

sdu.dk

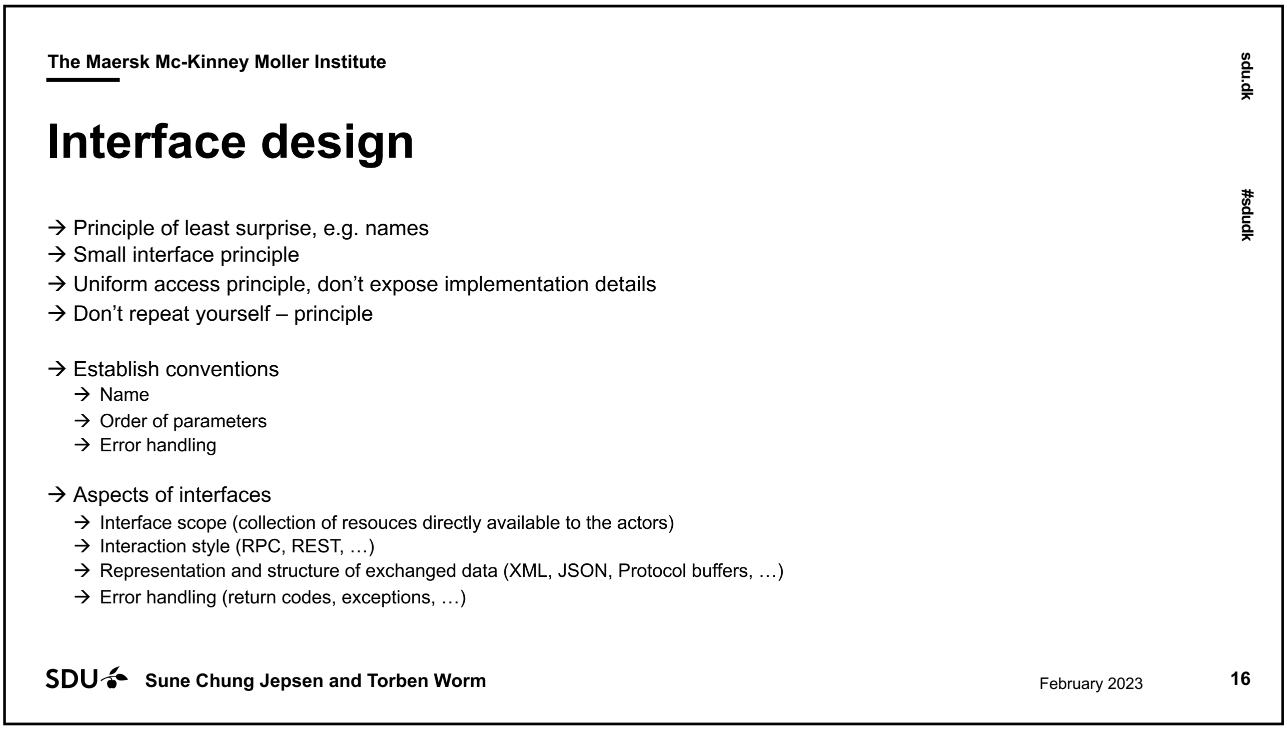
#sdudk

Resources

- → Syntax
  - → Signature
    - → Name of the ressource
    - → Names and datatypes of parameters
  - → To be able to write a syntactically correct program
- → Semantics
  - → What is the result of invoking the ressource
    - → Assignment of values
    - → Assumptions about the values crossing the interface
    - → State changes
    - → Events and signals
    - → Effect on future invocations
    - → Human observable results

SDU Sune Chung Jepsen and Torben Worm February 2023 15

15



The Maersk Mc-Kinney Moller Institute

sdu.dk

#sdudk

Interface design

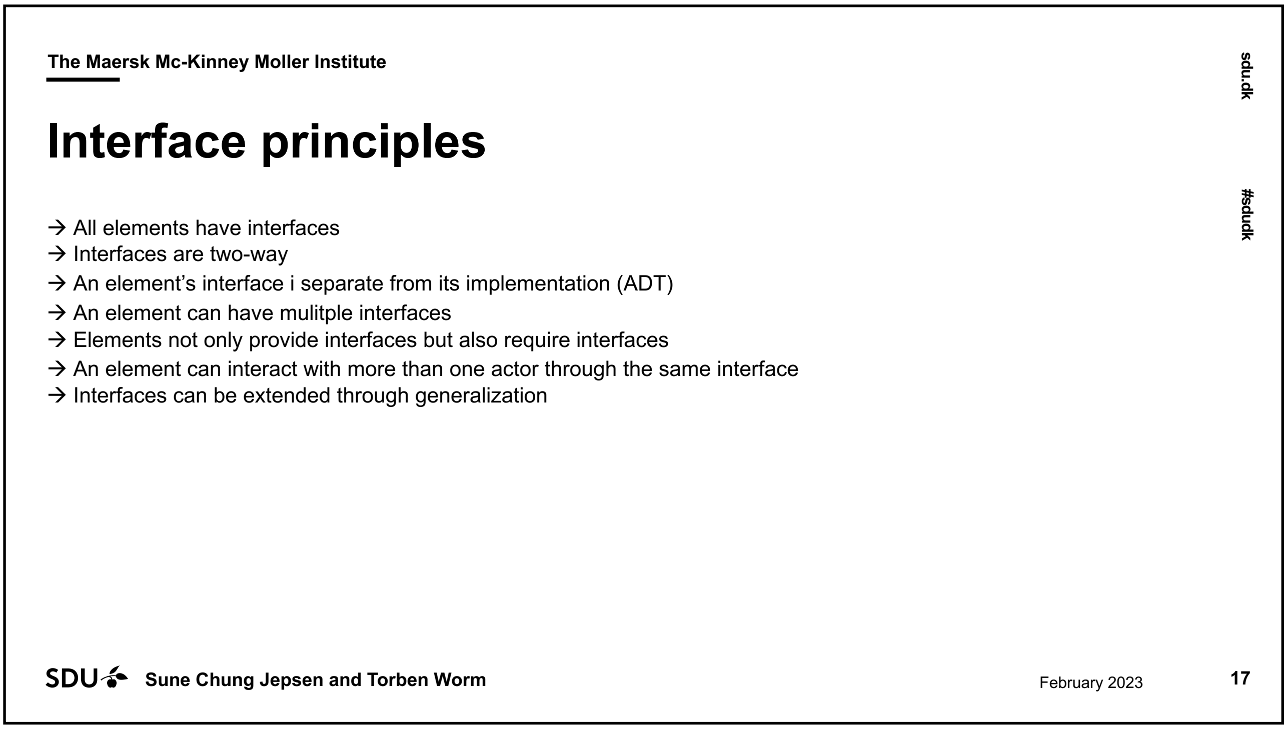
- → Principle of least surprise, e.g. names
- → Small interface principle
- → Uniform access principle, don’t expose implementation details
- → Don’t repeat yourself – principle

- → Establish conventions
  - → Name
  - → Order of parameters
  - → Error handling

- → Aspects of interfaces
  - → Interface scope (collection of resouces directly available to the actors)
  - → Interaction style (RPC, REST, …)
  - → Representation and structure of exchanged data (XML, JSON, Protocol buffers, …)
  - → Error handling (return codes, exceptions, …)

SDU Sune Chung Jepsen and Torben Worm February 2023 16

16



The Maersk Mc-Kinney Moller Institute

sdu.dk

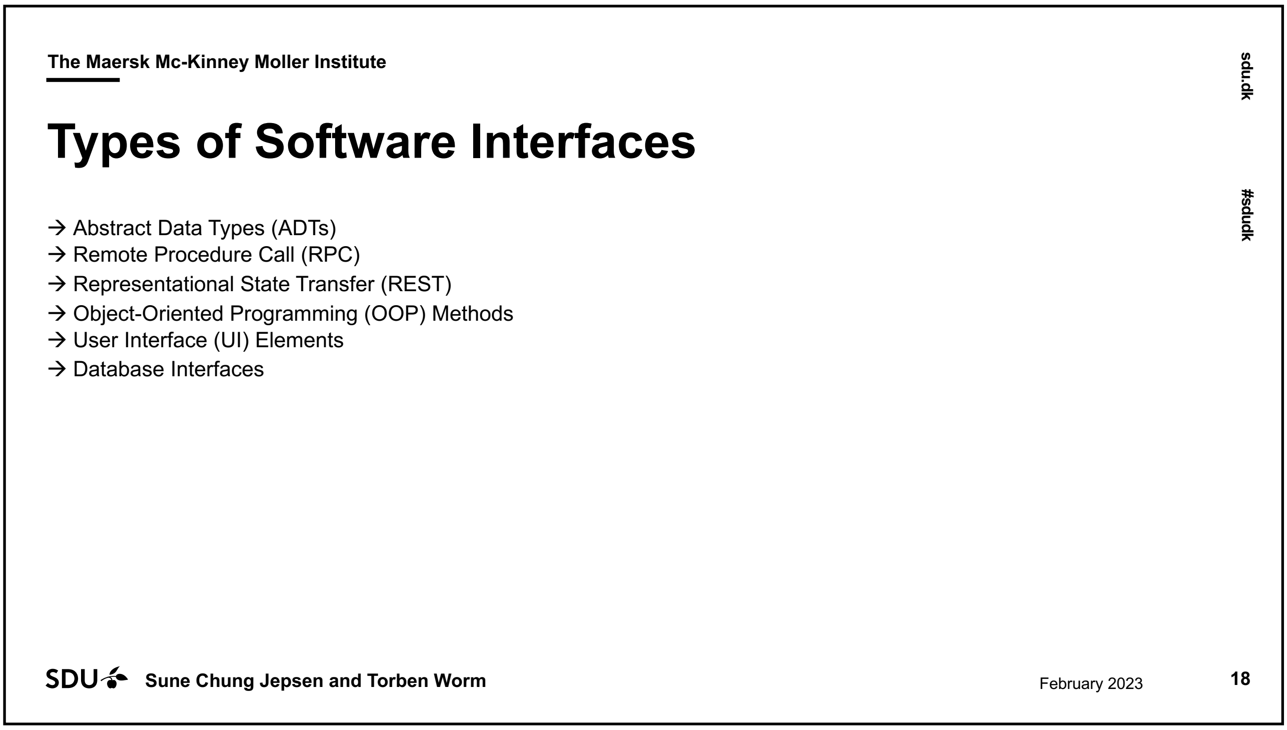
#sdudk

Interface principles

→ All elements have interfaces
→ Interfaces are two-way
→ An element's interface i separate from its implementation (ADT)
→ An element can have mulitple interfaces
→ Elements not only provide interfaces but also require interfaces
→ An element can interact with more than one actor through the same interface
→ Interfaces can be extended through generalization

SDU Sune Chung Jepsen and Torben Worm February 2023 17

17



The Maersk Mc-Kinney Moller Institute

sdu.dk

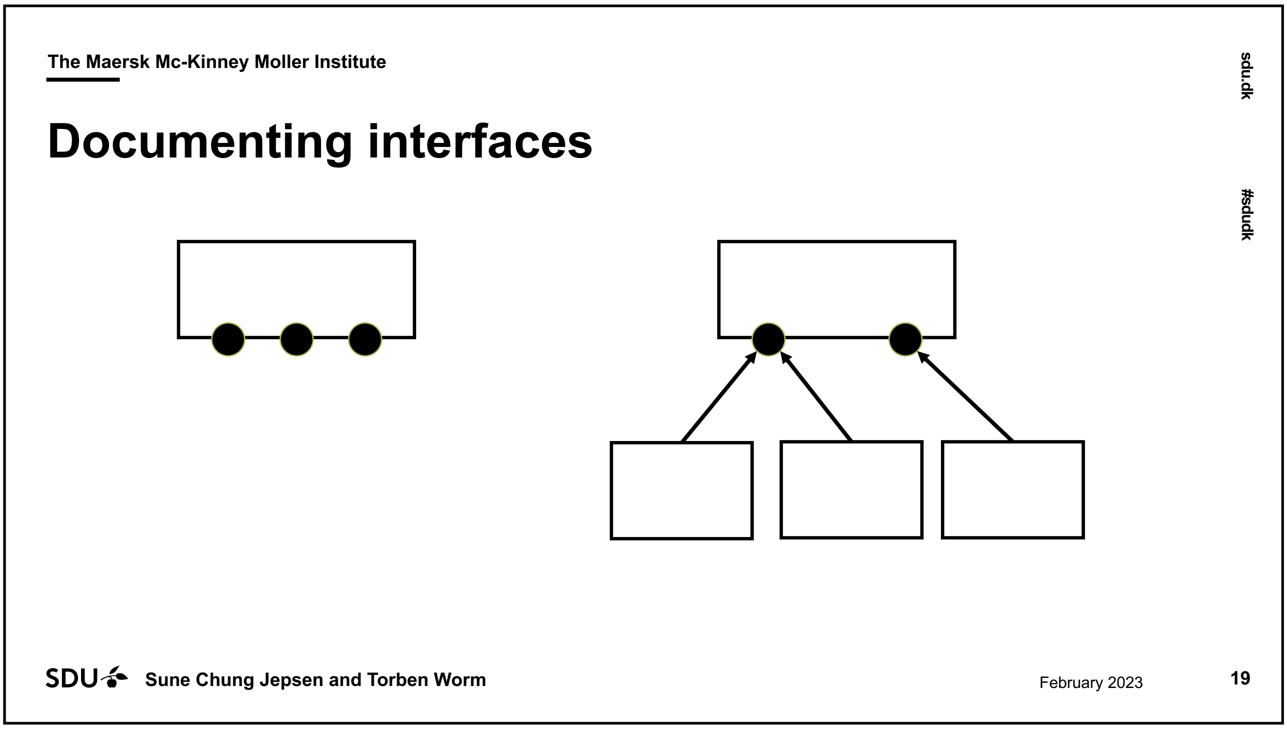
#sdudk

Types of Software Interfaces

→ Abstract Data Types (ADTs)
→ Remote Procedure Call (RPC)
→ Representational State Transfer (REST)
→ Object-Oriented Programming (OOP) Methods
→ User Interface (UI) Elements
→ Database Interfaces

SDU Sune Chung Jepsen and Torben Worm February 2023 18

18



The Maersk Mc-Kinney Moller Institute

sdu.dk

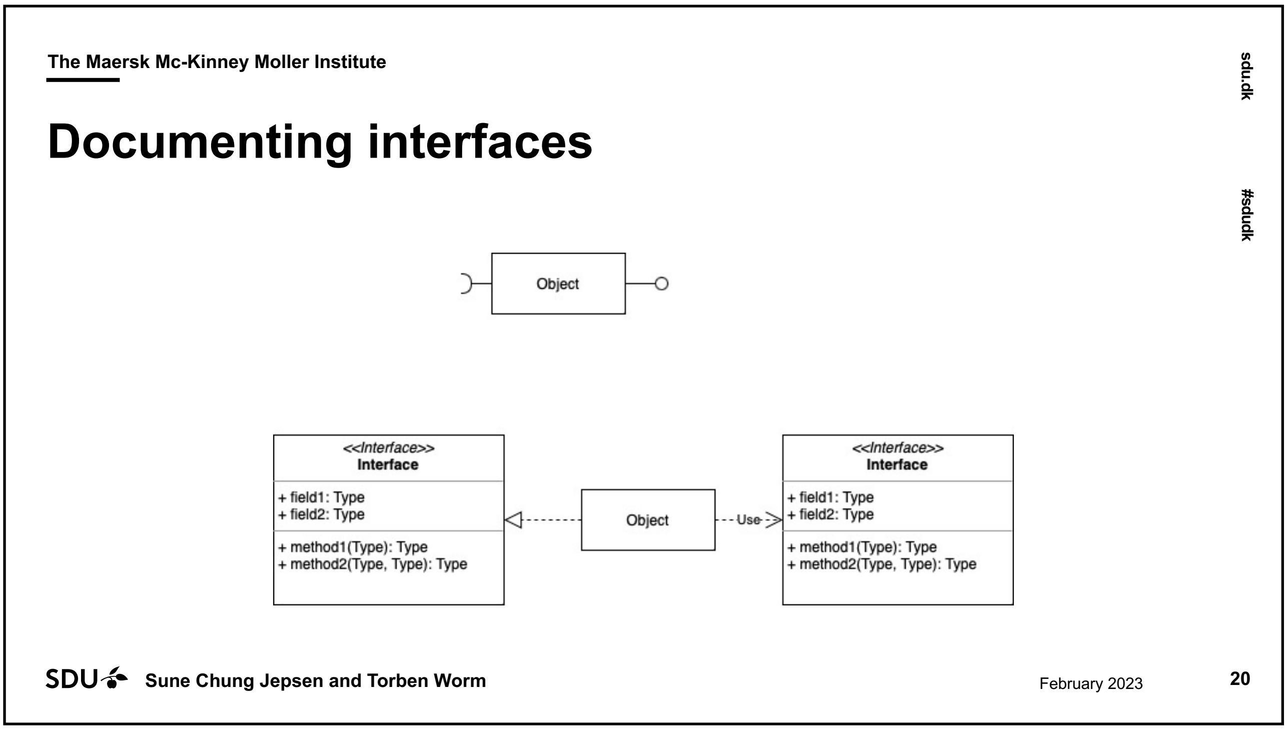
#sdudk

Documenting interfaces

SDU Sune Chung Jepsen and Torben Worm

February 2023

19



The Maersk Mc-Kinney Moller Institute

sdu.dk

#sdudk

Documenting interfaces

SDU Sune Chung Jepsen and Torben Worm

February 2023

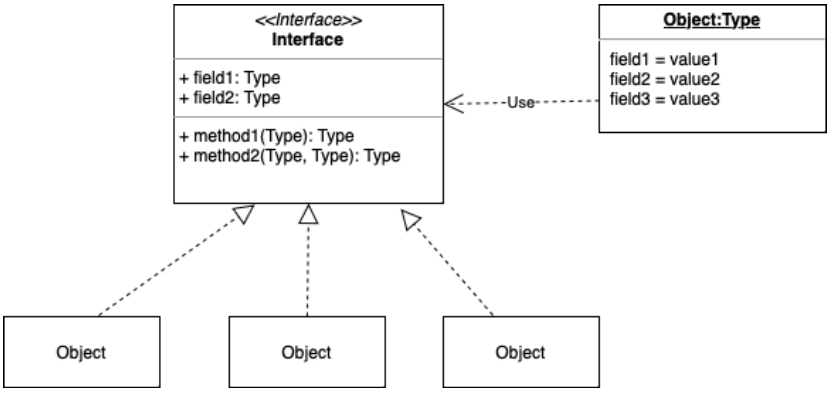
20

The Maersk Mc-Kinney Moller Institute

sdu.dk

#sdudk

Documenting interfaces



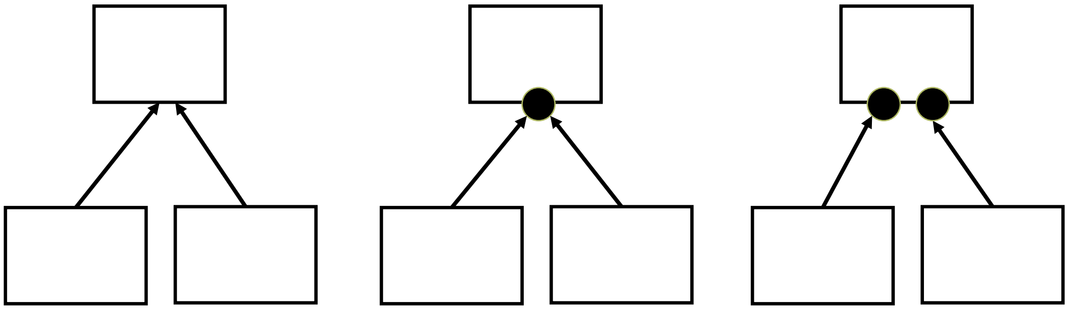
22

The Maersk Mc-Kinney Moller Institute

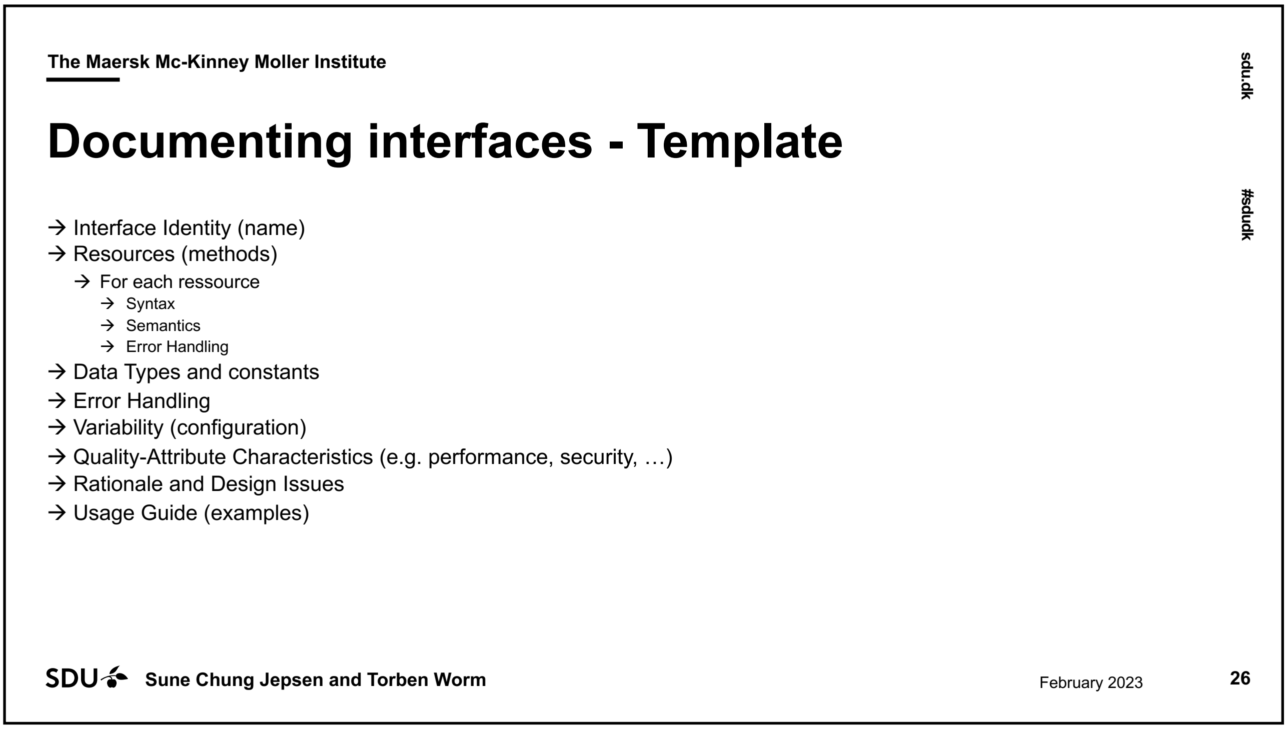
sdu.dk

#sdudk

Documenting interfaces



24



The Maersk Mc-Kinney Moller Institute

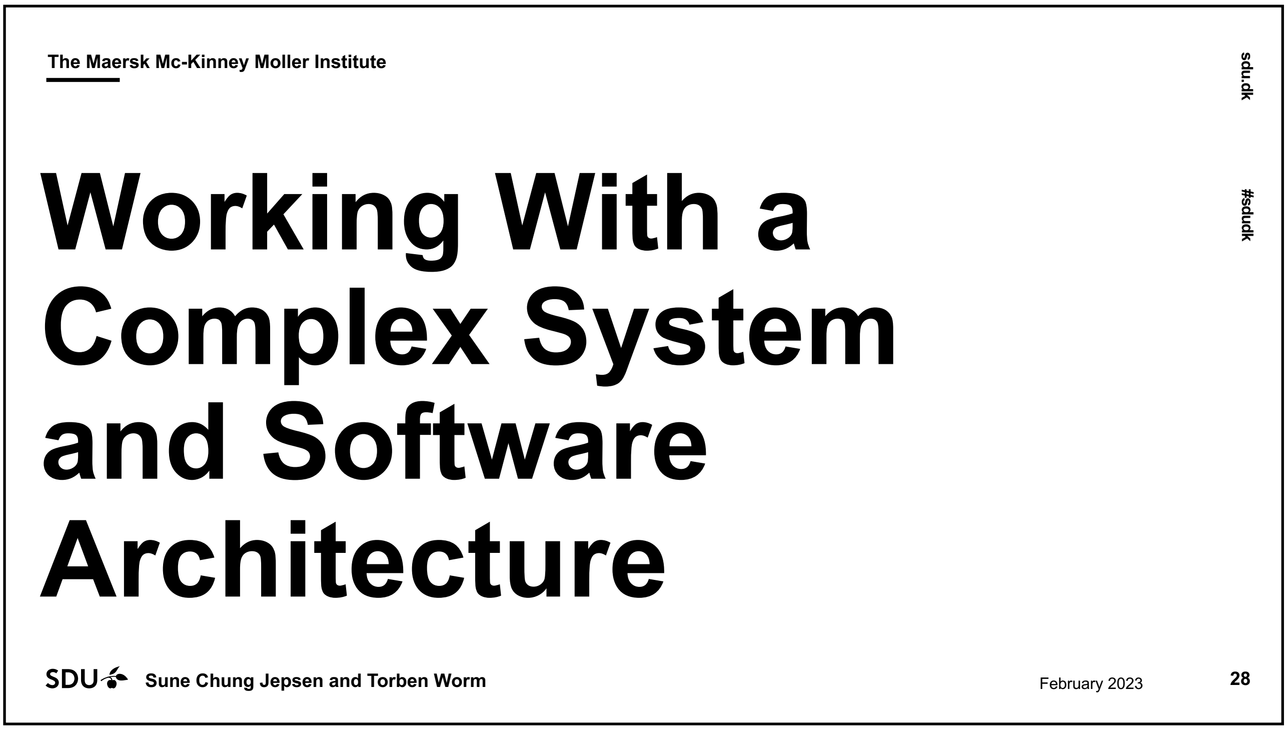
Documenting interfaces - Template

- → Interface Identity (name)
- → Resources (methods)
  - → For each ressource
    - → Syntax
    - → Semantics
    - → Error Handling
- → Data Types and constants
- → Error Handling
- → Variability (configuration)
- → Quality-Attribute Characteristics (e.g. performance, security, …)
- → Rationale and Design Issues
- → Usage Guide (examples)

sdu.dk #sdudk

SDU Sune Chung Jepsen and Torben Worm February 2023 26

26



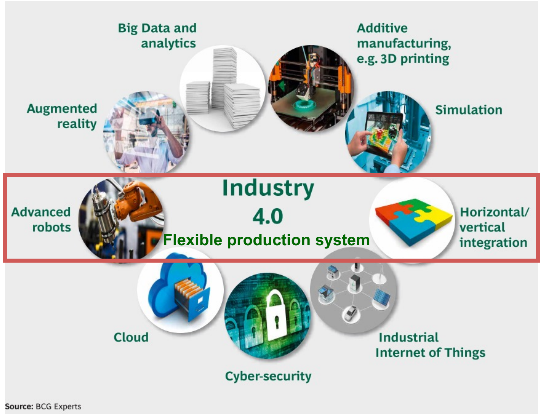
The Maersk Mc-Kinney Moller Institute

Working With a Complex System and Software Architecture

sdu.dk #sdudk

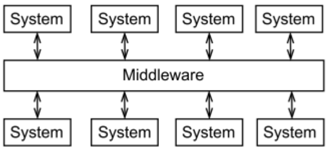
SDU Sune Chung Jepsen and Torben Worm February 2023 28

28


Big Data and analytics
Additive manufacturing, e.g. 3D printing
Augmented reality
Simulation
Advanced robots
Industry 4.0
Flexible production system
Horizontal/ vertical integration
Cloud
Industrial Internet of Things
Cyber-security

Source: BCG Experts

Middleware software architecture



The Maersk Mc-Kinney Moller Institute

sdu.dk

#sdudk

Industrial 4.0 and Challenges

→ Increasing connectivity

→ 30-50% reduction of total machine downtime [19]

→ Due to increasing connectivity

  → Security threats

  → Scalability

  → Software infrastructure

→ A way to address the software infrastructure challenge

  → **Middleware**

[19] McKinsey & Company. Industry 4.0 How to navigate digitization of the manufacturing sector. Tech. rep. McKinsey & Company, 2015, p. 62 (cit. on pp. 4, 54).

SDU Sune Chung Jepsen February 2023

29

The Maersk Mc-Kinney Moller Institute

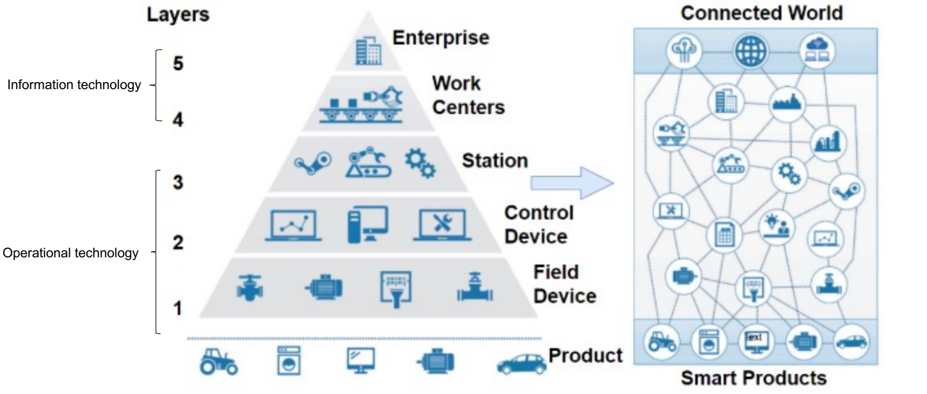
sdu.dk

#sdudk

Flexible Production System

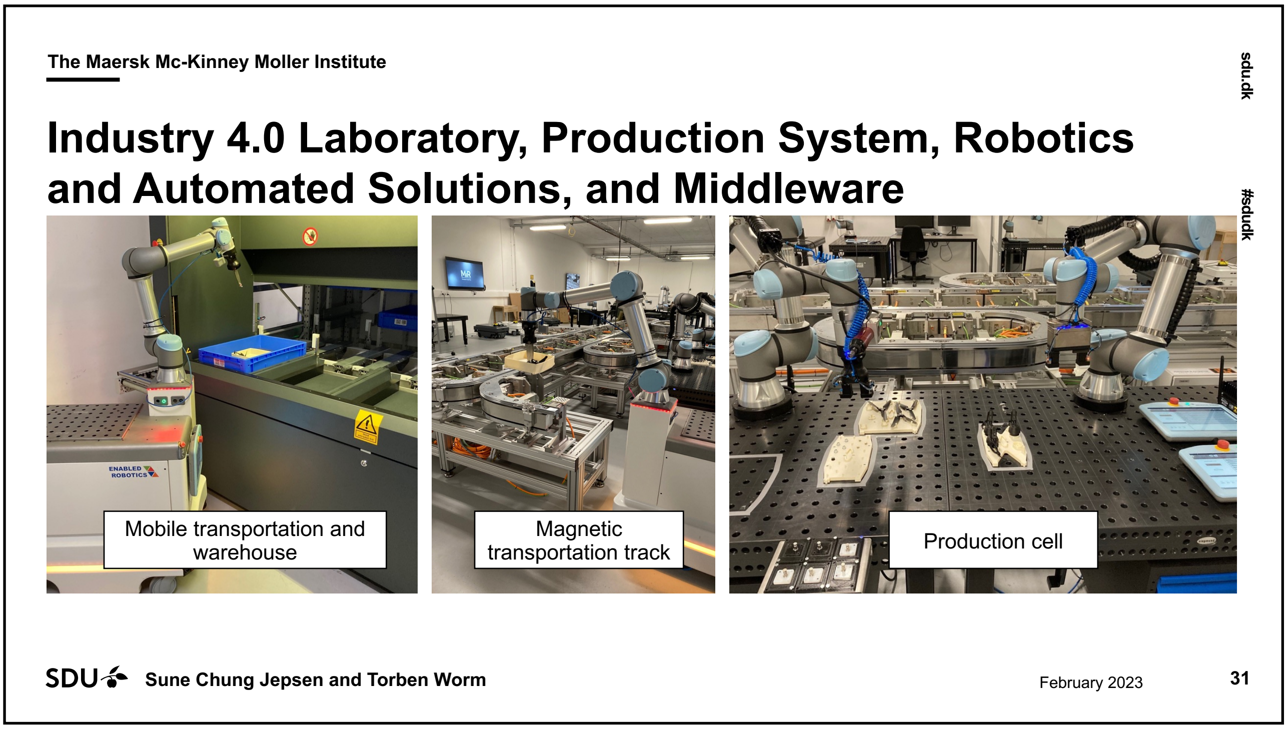
Traditional production system

Industry 4.0 production system



SDU Sune Chung Jepsen February 2023 30

30



The Maersk Mc-Kinney Moller Institute

sdu.dk

#sduck

Industry 4.0 Laboratory, Production System, Robotics and Automated Solutions, and Middleware

Mobile transportation and warehouse

Magnetic transportation track

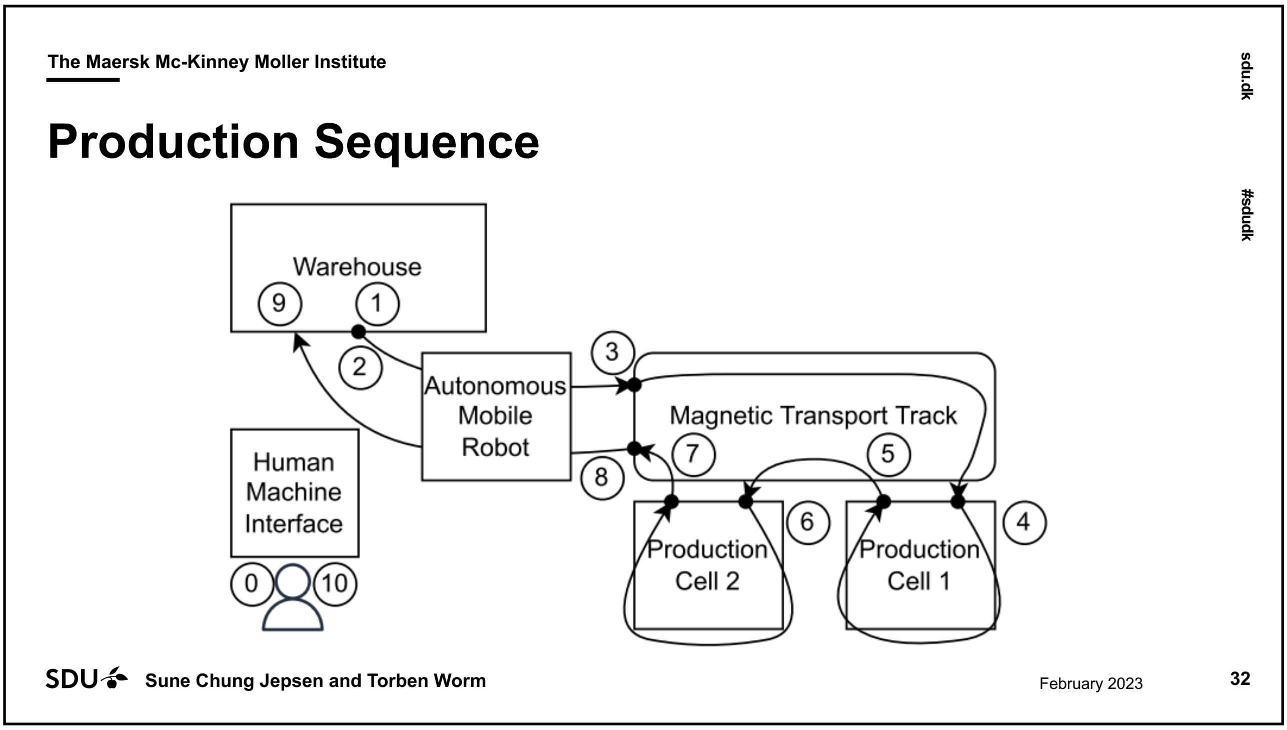
Production cell

SDU Sune Chung Jepsen and Torben Worm

February 2023

31

31



The Maersk Mc-Kinney Moller Institute

sdu.dk

#sduck

Production Sequence

Warehouse

9

1

2

Autonomous Mobile Robot

3

Magnetic Transport Track

7

5

8

Human Machine Interface

0

10

Production Cell 2

6

Production Cell 1

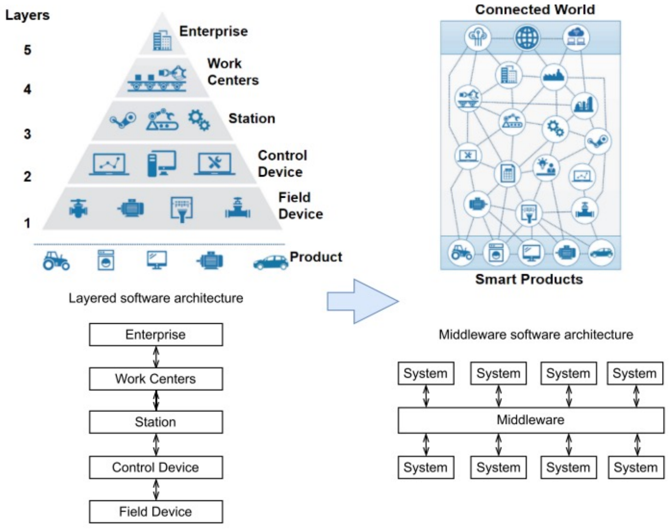
4

SDU Sune Chung Jepsen and Torben Worm

February 2023

32

32



The Maersk Mc-Kinney Moller Institute

sdu.dk

#sdudk

Software Architecture's Role in Middleware

→ Designing involves software engineering

→ A discipline is software architecture

  → Quality attributes scenarios and tactics

→ Software architecture is expensive to change later

The Maersk Mc-Kinney Moller Institute

sdu.dk

#sdudk

Flexible Production Processes

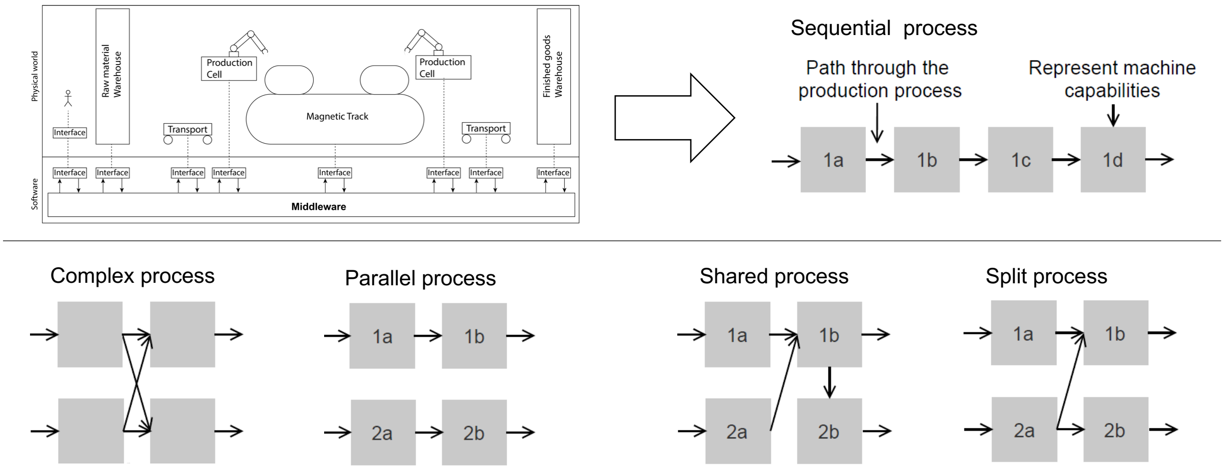
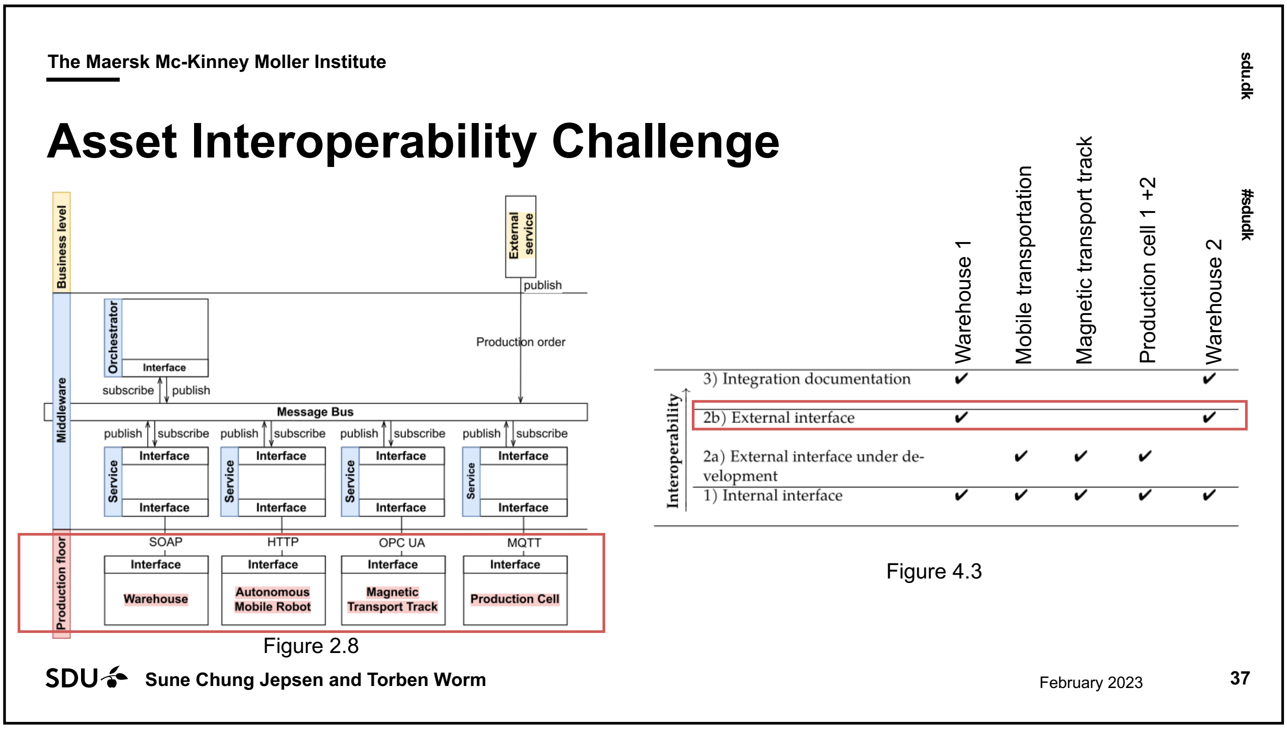


Figure 3.2



The Maersk Mc-Kinney Moller Institute

sdu.dk

#sdudk

Asset Interoperability Challenge

Business level

External service

publish

Middleware

Orchestrator

Interface

subscribe publish

Production order

Message Bus

publish subscribe publish subscribe publish subscribe publish subscribe

Service Interface Interface

Service Interface Interface

Service Interface Interface

Service Interface Interface

Production floor

SOAP HTTP OPC UA MQTT

Interface Warehouse

Interface Autonomous Mobile Robot

Interface Magnetic Transport Track

Interface Production Cell

Figure 2.8

		Warehouse 1	Mobile transportation	Magnetic transport track	Production cell 1 +2	Warehouse 2
Interoperability	3) Integration documentation	✔				✔
	2b) External interface	✔				✔
	2a) External interface under development		✔	✔	✔	
	1) Internal interface	✔	✔	✔	✔	✔

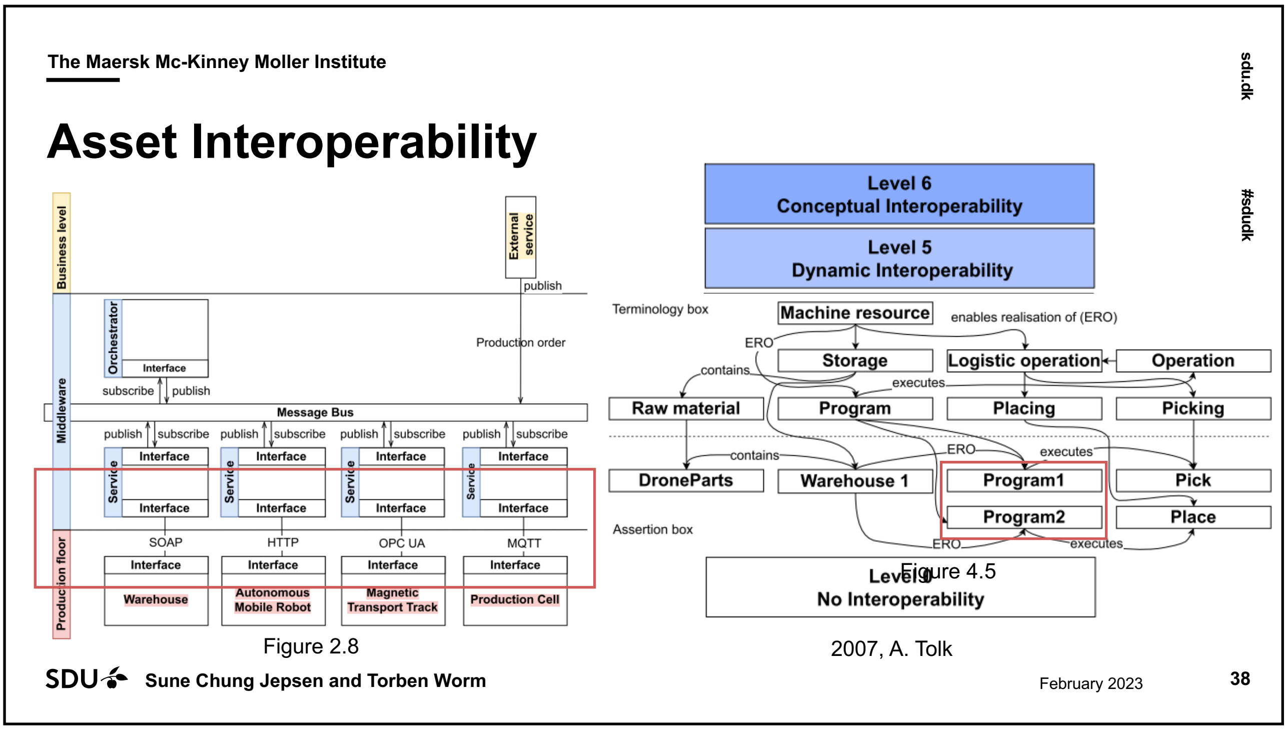
Figure 4.3

SDU Sune Chung Jepsen and Torben Worm

February 2023

37

37



The Maersk Mc-Kinney Moller Institute

sdu.dk

#sdudk

Asset Interoperability

Business level

External service

publish

Middleware

Orchestrator

Interface

subscribe publish

Production order

Message Bus

publish subscribe publish subscribe publish subscribe publish subscribe

Service Interface Interface

Service Interface Interface

Service Interface Interface

Service Interface Interface

Production floor

SOAP HTTP OPC UA MQTT

Interface Warehouse

Interface Autonomous Mobile Robot

Interface Magnetic Transport Track

Interface Production Cell

Figure 2.8

Level 6
Conceptual Interoperability

Level 5
Dynamic Interoperability

Terminology box

Machine resource

enables realisation of (ERO)

ERO

Storage

Logistic operation

Operation

contains

executes

Raw material

Program

Placing

Picking

contains

ERO

executes

DroneParts

Warehouse 1

Program1

Pick

Assertion box

Program2

Place

ERO

executes

Level 0
No Interoperability

Figure 4.5

2007, A. Tolk

SDU Sune Chung Jepsen and Torben Worm

February 2023

38

38

The Maersk Mc-Kinney Moller Institute

Middleware Interoperability

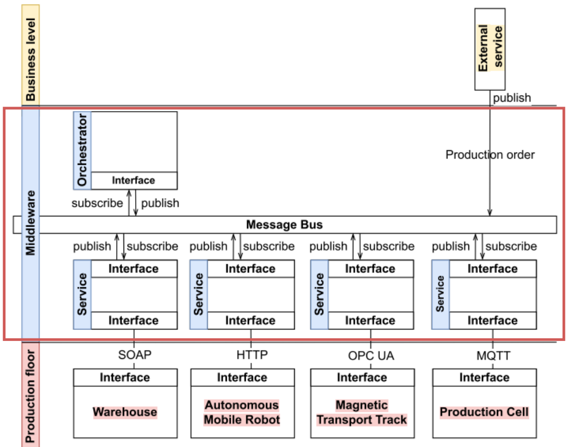
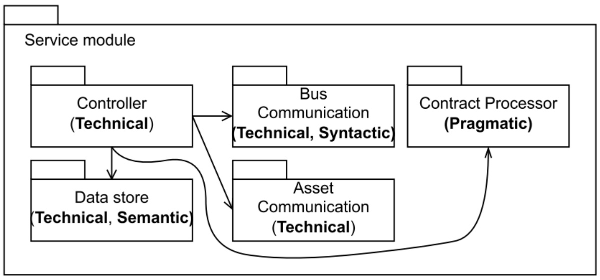


Figure 2.8


Service module
Controller (Technical)
Bus Communication (Technical, Syntactic)
Contract Processor (Pragmatic)
Data store (Technical, Semantic)
Asset Communication (Technical)

The Maersk Mc-Kinney Moller Institute

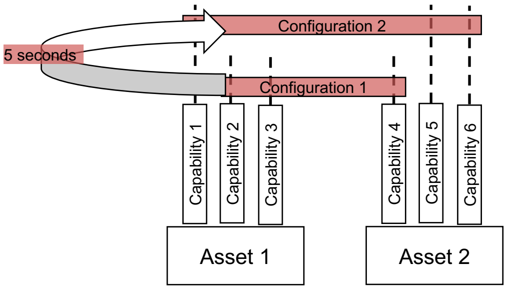
Reconfigurability Definition

Definition 5.1 (Capability) *A capability can be invoked on one or more assets.*

Definition 5.2 (Configuration) *A configuration is a set of capabilities.*

Definition 5.3 (Reconfigurability) *The ability to accommodate the necessary change in software structures to support transition between configurations given a set of constraints.*

Definition 5.4 (Constraint) *A constraint is requirement that must be satisfied by the system.*



Reconfiguration Template Analysis

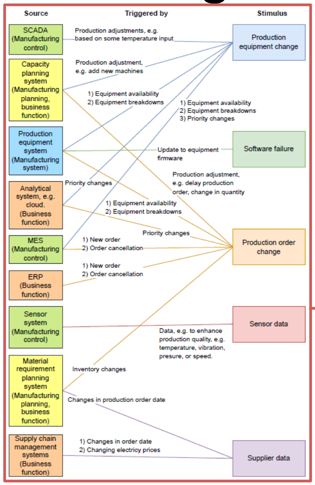

Source
Triggered by
Stimulus
SCADA (Manufacturing control)
Capacity planning system (Manufacturing planning, business function)
Production equipment system (Manufacturing system)
Analytical system, e.g. cloud. (Business function)
MES (Manufacturing control)
ERP (Business function)
Sensor system (Manufacturing control)
Material requirement planning system (Manufacturing planning, business function)
Supply chain management systems (Business function)
Production adjustments, e.g. based on some temperature input
Production adjustment, e.g. add new machines
1) Equipment availability 2) Equipment breakdowns
1) Equipment availability 2) Equipment breakdowns 3) Priority changes
Update to equipment firmware
Production adjustment, e.g. delay production order, change in quantity
Priority changes
1) Equipment availability 2) Equipment breakdowns
Priority changes
1) New order 2) Order cancellation
1) New order 2) Order cancellation
Data, e.g. to enhance production quality, e.g. temperature, vibration, presure, or speed.
Inventory changes
Changes in production order date
1) Changes in order date 2) Changing electricy prices
Production equipment change
Software failure
Production order change
Sensor data
Supplier data

Figure 5.7

	Communication		
Middleware	Request-reply	Point-to-point	Publish/ Subscribe
Service-oriented	✔	×	×
Agent-based	×	✔	×
Event-based	×	×	✔

Figure 2.2

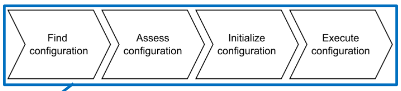


Figure 5.8

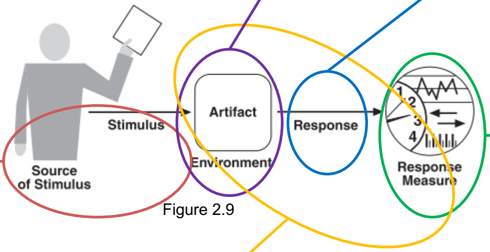


Figure 2.9

Definition 5.3 (Reconfigurability) *The ability to accommodate the necessary change in software structures to support transition between configurations given a set of constraints.*

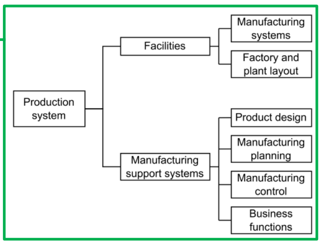


Figure 2.1

References

1. [Bass2021] L. Bass, P. Clements, R. Kazman, and an O'Reilly Media Company Safari.Software Architecture in Practice, 4th Edition. SEI series in software engineering. Addison-Wesley Professional, 2021