EuroTelematik

>> we find your better way

Concept of an Integration Platform for the Support of Distributed Engineering Processes

„System Level Design in the MEDEA+ Project SpeAC“

2nd Open SpeAC Workshop

February 20, 2004

MEDEA+

SpeAC

MEDEA+ A508

Gerd Lutz, Mirsad Delic, Philipp Boysen

Euro Telematik AG, Ulm, Germany

bmb+f

EuroTelematik
>> we find your better way

Outline

- Distributed Engineering Processes
  - A Project Managers worst Nightmare
  - Requirements for an Integration Platform
  - SPEM -> Modeling Processes
  - XMI -> XML Metadata Interchange
- System Modeling & Code Generation
  - SysML
  - Model Driven Architecture (MDA)

Business Activity Euro Telematik Samples Aerospace

EuroTelematik
>> we find your better way

GCAS-2000 / CDTI-2000
GCAS Flight Companion
(own products)

3FMS / MA-AFAS (R&D Projects)
Onboard Taxi Management
Customer: EC
Partners: Airbus, Thales, Smiths, BAE Systems, etc.

MIDS Software (Engineering)
Information management MRCA Tornado
Customer: EADS

AWRH II (R&D Project)
Mission Management All Weather SAR Helicopter
Customer: BMWi
Partners: Eurocopter, Aerodata, etc.

NLES Testbed (Engineering)
Test System Development for Satellite Communication Ground Station within the EGNOS program (ESA)
Customer: Astrium

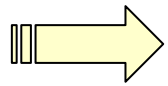
A Project Manager‘s worst Nightmare

EuroTelematik
>> we find your better way

- ESA in the mid 1970s
- Support of a mission, the launch of which was imminent, but project was late
- Staff involved were often brilliant engineers
- However not used to work in project environment with rigorous cost and schedule constraints
- Idea to reduce requirements, only proceed with those essential for the launch of the satellite
- ***There were no written requirements***
- Some, maybe, could be retrieved from meeting minutes.
- The rest were in the minds of the project engineers

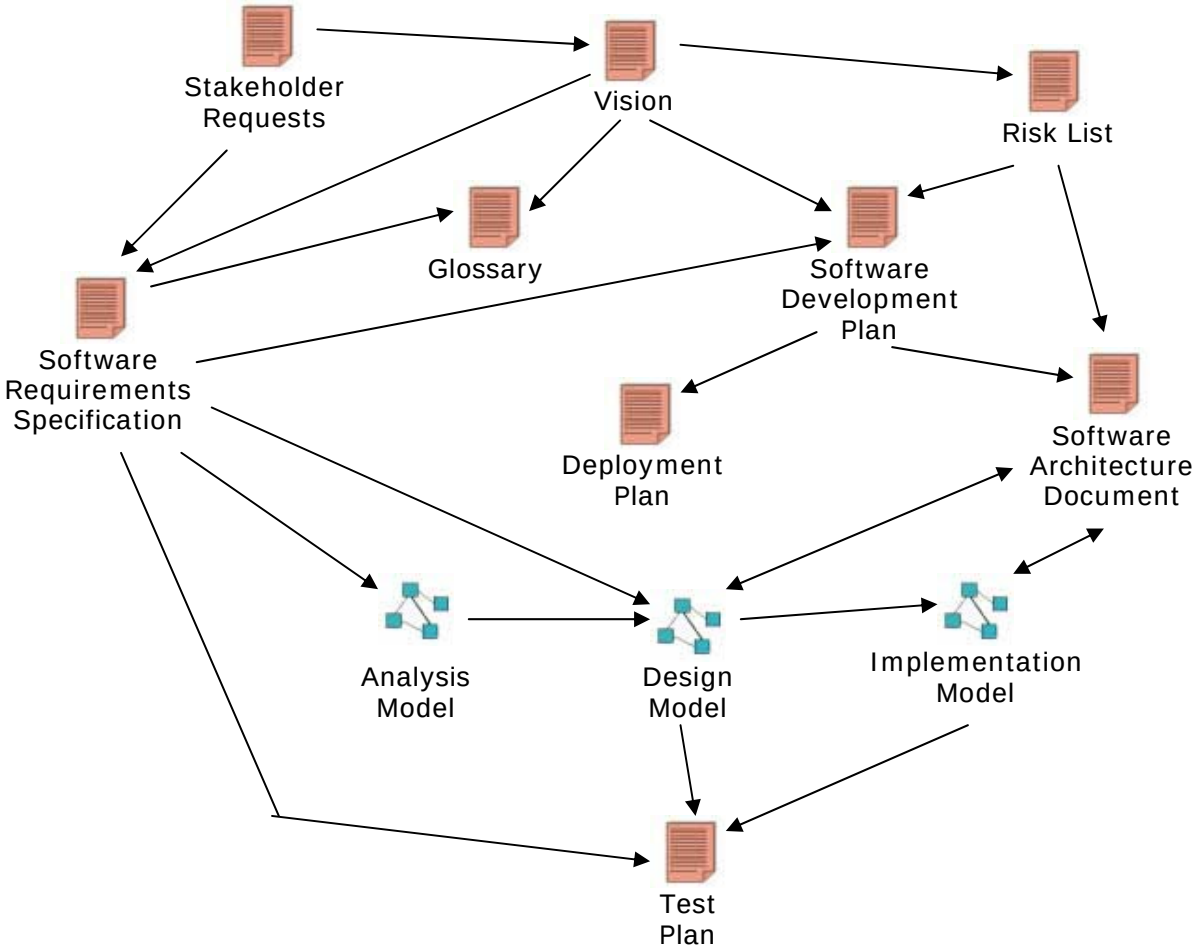
Requirements for Integration Platform

EuroTelematik
>> we find your better way

- Web based platform
- Secure connections possible
- Supporting distributed development processes
- Guidance through development process 
- Configuration management
- Connecting tools via standardized data format
- Supporting metrics

Artifacts of an exemplary Development Process

EuroTelematik
>> we find your better way



EuroTelematik
>> we find your better way

Exemplary Artifact: Vision

Vision	The **Vision** is a general vision of the core project's requirements, and provides the contractual basis for the more detailed technical requirements.
Worker:	System Analyst
Template:	Microsoft Word Template HTML Template
Examples:	Course Registration System
More Information:	Checkpoints: Vision Checkpoint: Stakeholder Requests Checkpoints: Requirements Attributes Artifact: Requirements Management Plan
Purpose Brief Outline Timing Responsibility Additional Information Annotated Outline (hyperlinks into HTML template in a new window)	

EuroTelematik

>> we find your better way

Template: Vision

Vision

1. Introduction

*[The purpose of this document is to collect, analyze, and define high-level needs and features of the <<System Name>>. It focuses on the capabilities needed by the stakeholders, and the target users, and **why** these needs exist. The details of how the <<System Name>> fulfils these needs are detailed in the use-case and supplementary specifications.]*

*[The introduction of the **Vision** document should provide an overview of the entire document. It should include the purpose, scope, definitions, acronyms, abbreviations, references, and overview of this **Vision** document.]*

1.1 Purpose

*[Specify the purpose of this **Vision** document.]*

1.2 Scope

*[A brief description of the scope of this **Vision** document; what Project(s) it is associated with, and anything else that is affected or influenced by this document.]*

1.3 Definitions, Acronyms and Abbreviations

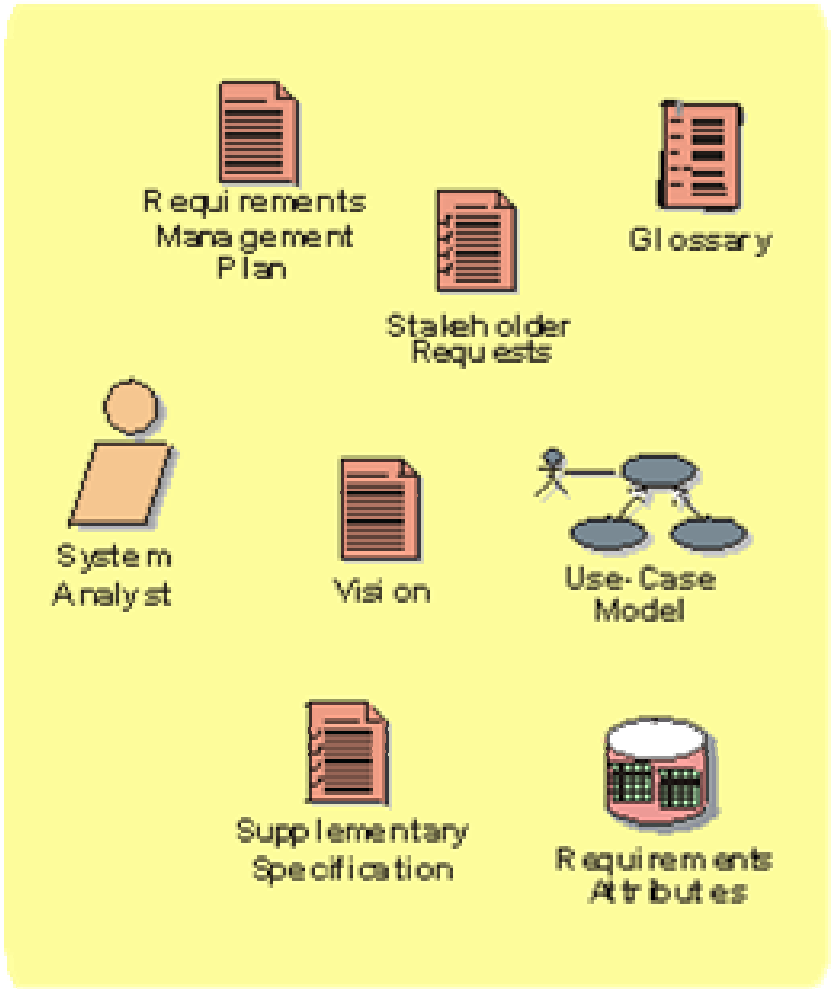
*[This subsection should provide the definitions of all terms, acronyms, and abbreviations required to properly interpret the **Vision** document. This information may be provided by reference to the project Glossary.]*

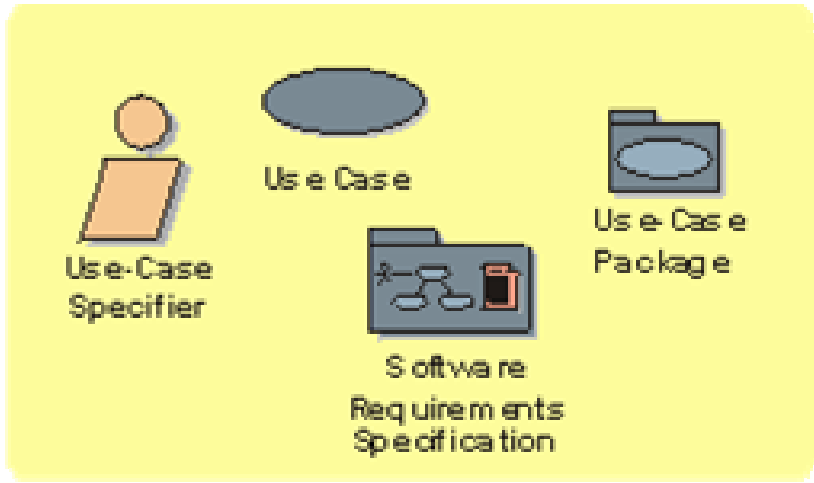
1.4 References

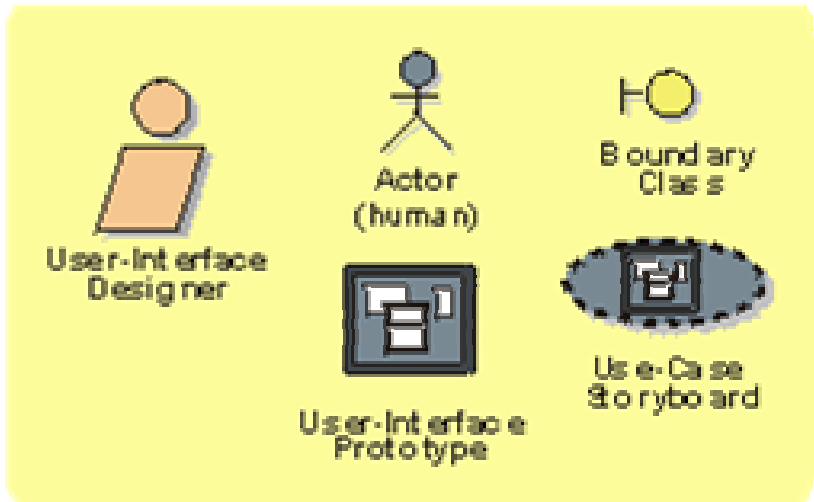
*[This subsection should provide a complete list of all documents referenced elsewhere in the **Vision** document. Each document should be identified by title, report number (if applicable), date, and publishing organization. Specify the sources from which the references can be obtained. This information may be provided by reference to an appendix or to another document.]*

1.5 Overview

Workers & according Artifacts







SPEM - Software Process Engineering Metamodel

EuroTelematik
>> we find your better way

- Necessity of supporting the definition of development processes
- Object-oriented approach
- UML as a graphical notation

- **SPEM**: Software Process Engineering Metamodel
- Specification is structured as a UML profile
- Metamodel for defining processes and their components
- Model serializable to XML code -> XMI
- Hosted by the Object Management Group (OMG)

EuroTelematik
>> we find your better way

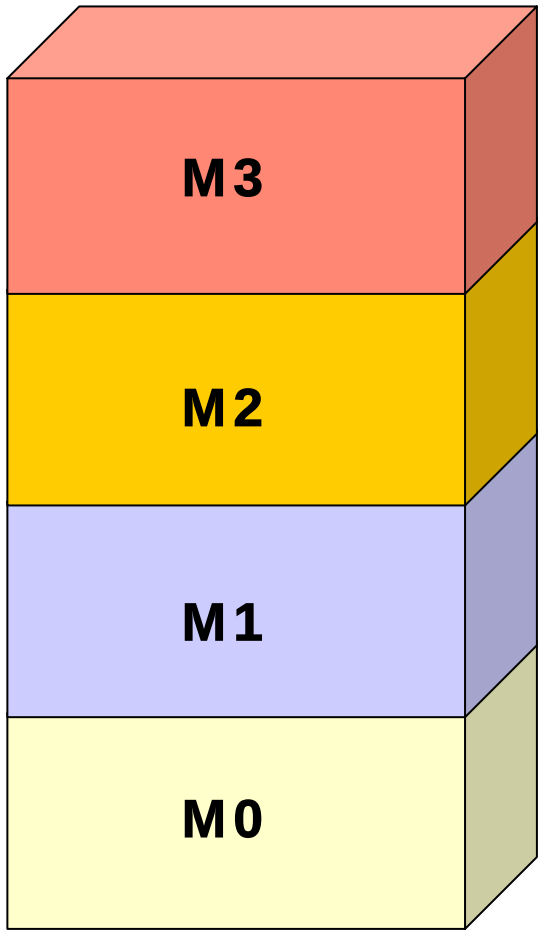
Levels of Modeling

MetaObject Facility
MOF

Process Metamodel

Process Model

Performing Process



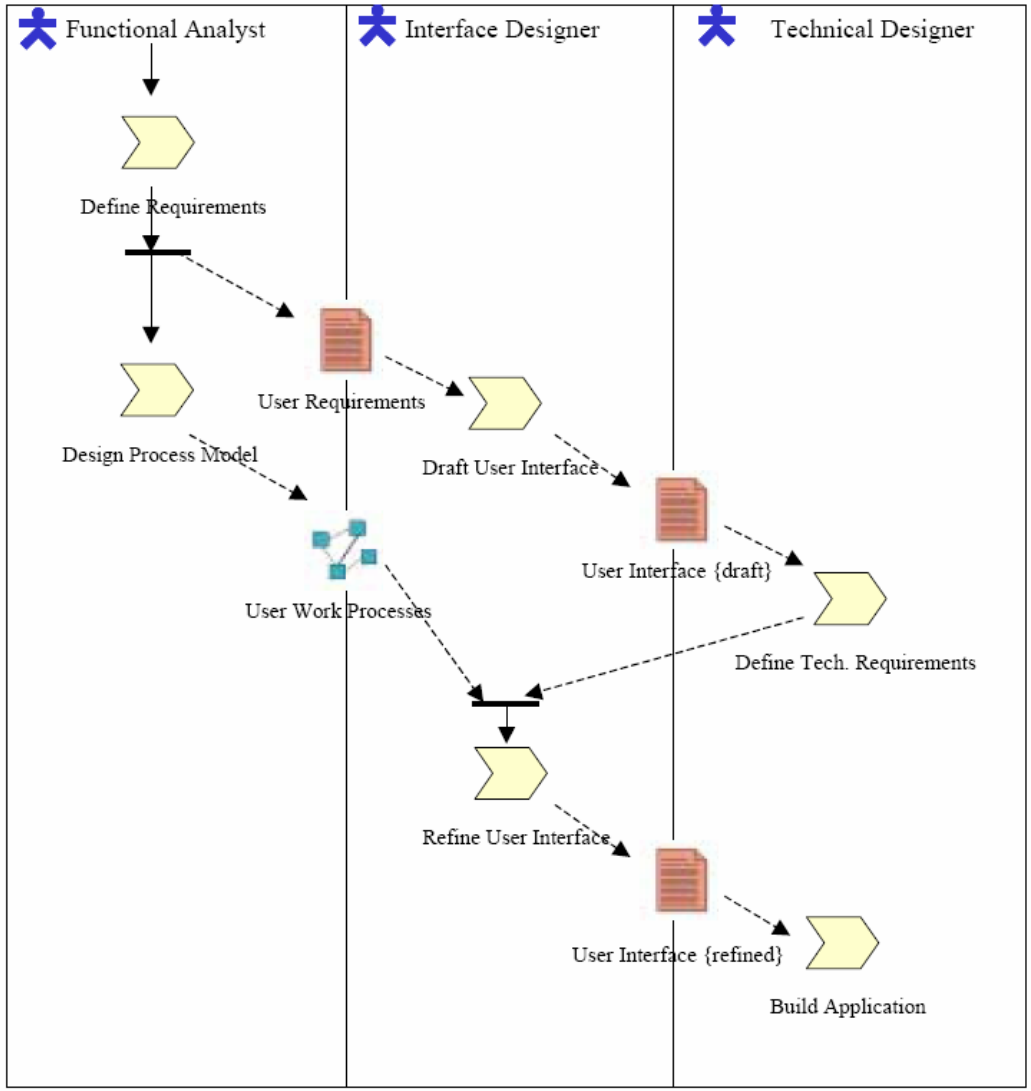
Definition of syntax for the meta-model layer

Template for M1,
UML profile, MOF metamodel

Definition of the process
e.g. RUP, ESA-ECSS, …
Also specific customaziation

Process as already enacted on a given project

Example of Activity Diagram

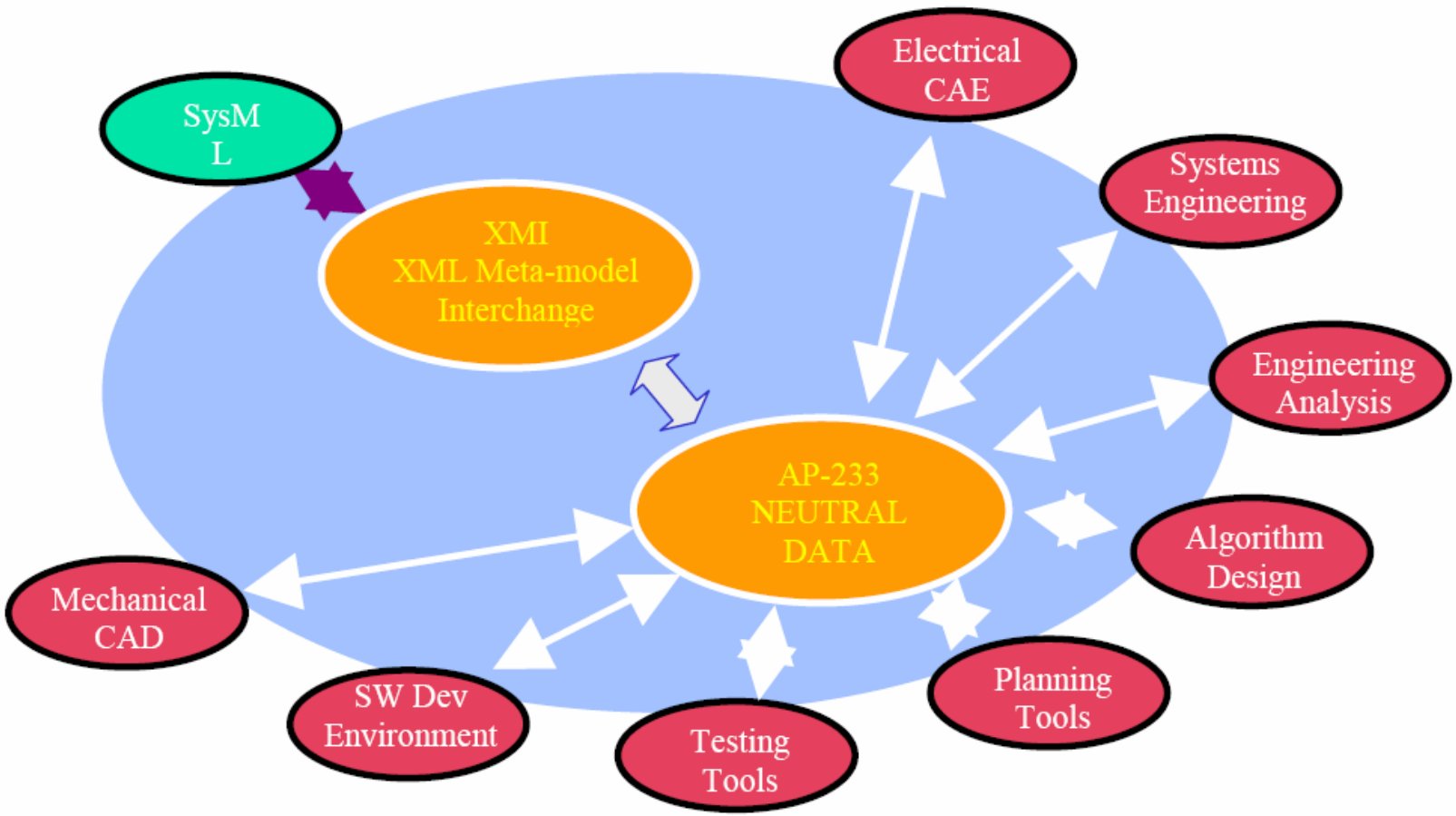


EuroTelematik
>> we find your better way

XMI
(XML Metadata Interchange)

- XMI is a widely used interchange format for sharing objects using XML
- XMI includes standard mechanisms to link objects within the same file or across files
- Validation of XMI documents using DTDs and Schemas
- XMI has two major components:
  - XML DTD production rules
  - XML document production rules
- Create SPEM Document Type Definitions
- Transfer process models based on SPEM as XML documents

SysML Alignment

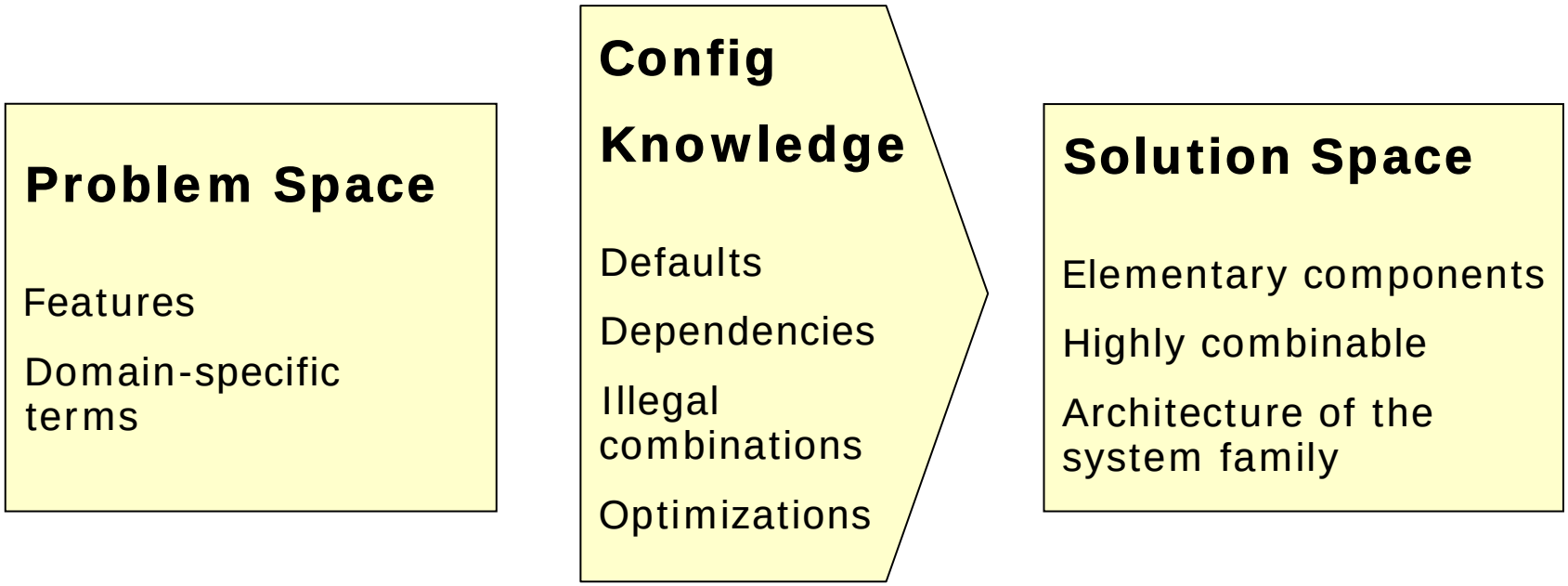


Model Driven Architecture & Code Generation

EuroTelematik
>> we find your better way

- Different lifecycles of domain specific terms and technical terms
  - Technical domains are changing fast
  - Domain specific knowledge is rather long-lived
- Segmentation of a system into functional and non-functional parts
  - Functional parts: domain specific terms
  - Non-functional parts: realization of technical requirements
    - Transactions, concurrency, scheduling, priority control, …
- Code generation: systematical concretion of an abstract specification

Transformation Process

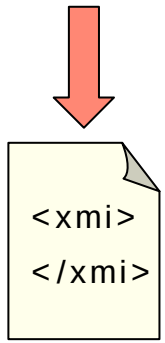


EuroTelematik
>> we find your better way

Code Generation

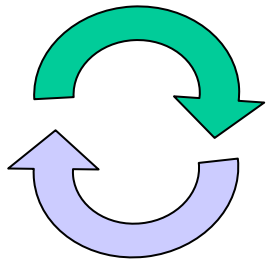
CASE Tool

(e.g. Rose, Together, Poseidon, …)

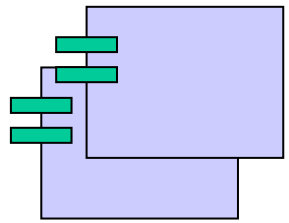


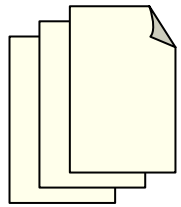
SysML Model

Code Generator



Code Components





Templates

e.g. Velocity Template Language

Thanks for your Attention.

Questions? Remarks?