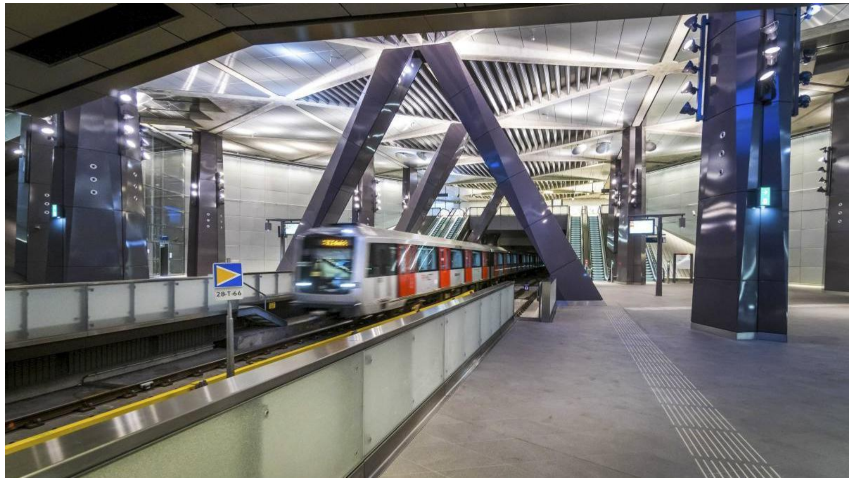
Modelling Large Infrastructure Systems

THE COLLECTIVE
the systems integration company

Who am i:

- Jan de Liefde
- Systems engineer in the infrastructure and industry domain
- Focus on Model Based (Systems) Engineering
- Jan.de.liefde@thecollective.si

THE COLLECTIVE
the systems integration company



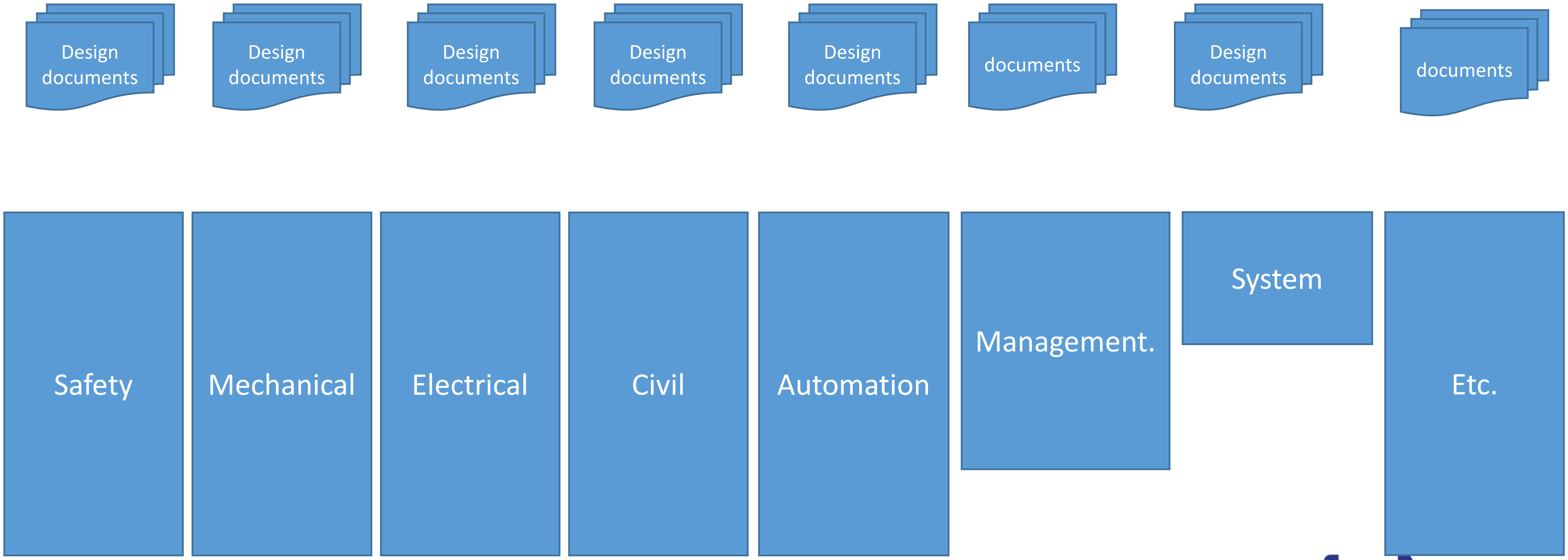
[TC] THE COLLECTIVE
the systems integration company

(Large) infrasystems: some characteristics

- High political and public interest (always in the spotlight)
- Long life-cycle (>80 years)
- Civil, Mechanical, Electrical, Industrial automation, etc.
- Direct or indirect impact on the environment (pollution, view, fine dust, etc.)
- Impact from the environment (salt water, wind, temperature, etc.)
- Strong focus on safety (zero casualties) and availability (24/7)
- Many different suppliers

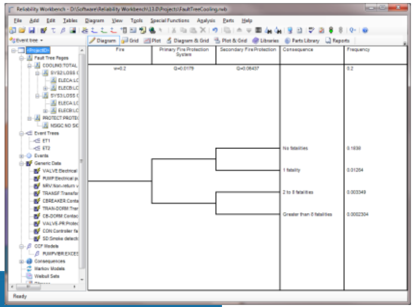
[TC] THE COLLECTIVE
the systems integration company

Today



Design documents
Design documents
Design documents
Design documents
Design documents
documents
Design documents
documents
Safety
Mechanical
Electrical
Civil
Automation
Management.
System
Etc.

Document centric

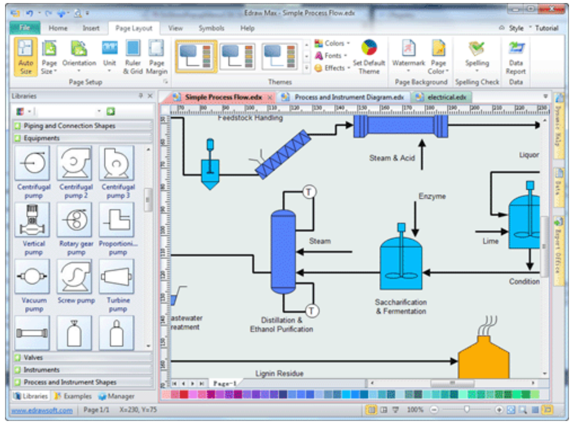


ENTERPRISE ARCHITECT
SPARX SYSTEMS

Microsoft Word

Visio

© Can Stock Photo - csp11947542



Microsoft Access

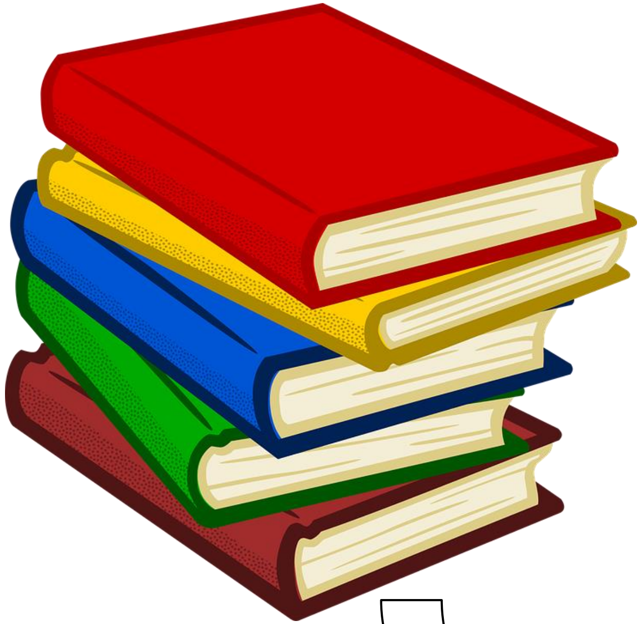
[TC] THE COLLECTIVE
the systems integration company

Today issues

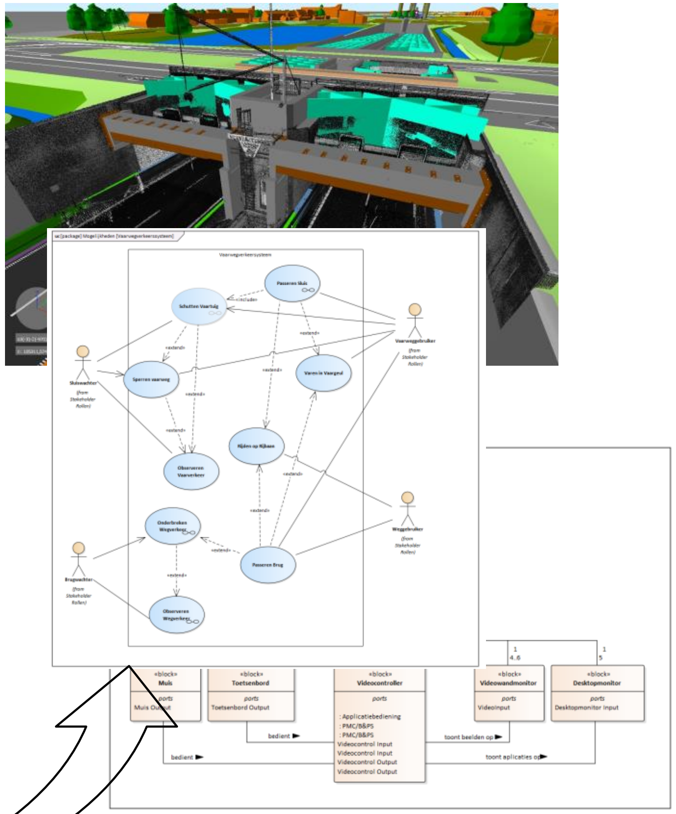
- Information duplication
  - In multiple documents same kind of information
  - Information loss during changes
- Access to information
  - Searching within documents
  - Exchange of information by using exports and copy/paste
- Lack of traceability
- Different not connected data sources
- Lack insights in the Impact of changes
- Discipline based

[TC] THE COLLECTIVE
the systems integration company

Document centric



Model centric



[TC] THE COLLECTIVE
the systems integration company

Modelling: Today issues

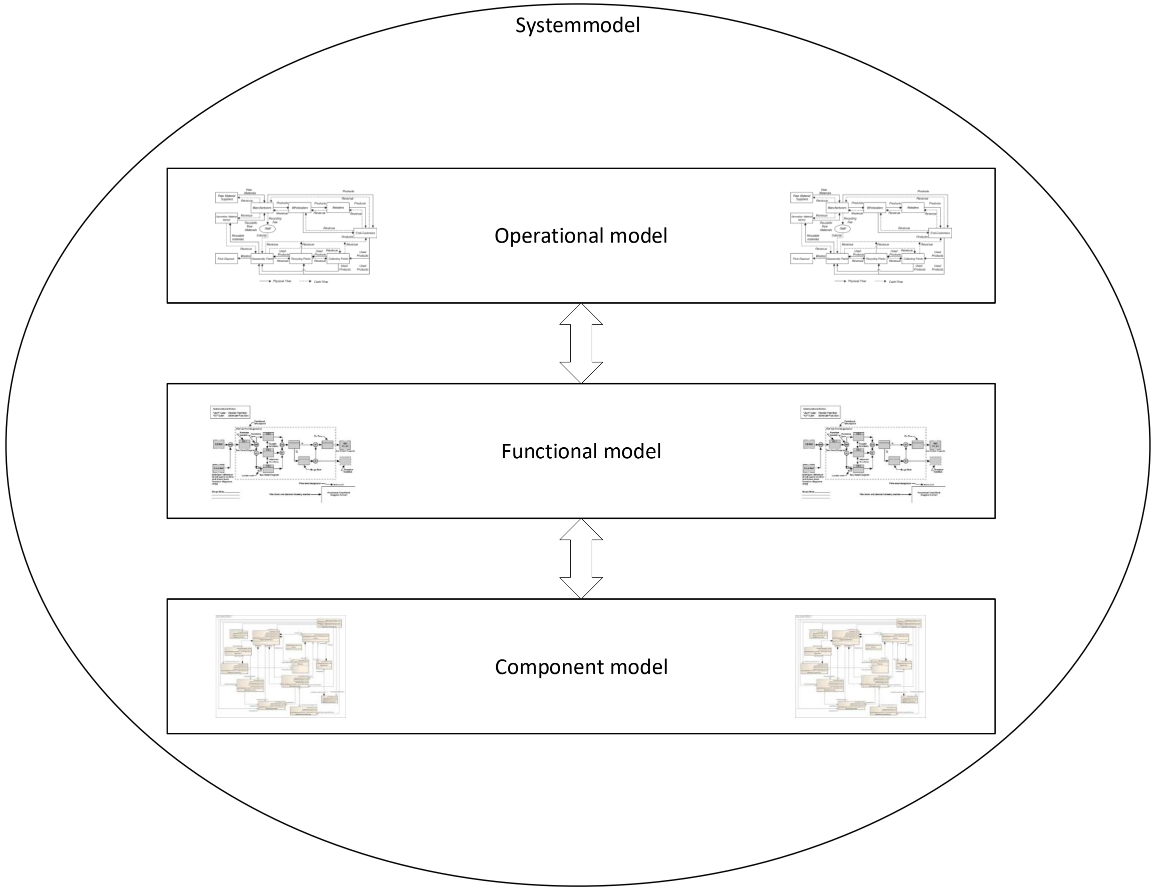
- High level design
  - Focus mainly on producing documents
  - Focus mainly on functions
- Objects partly modeled (no constraints, properties, etc.)
- Lack of traceability
- No distinction between problem and solution domain (usage is not the same as functionality/solution)
- Language not readable for domain engineers

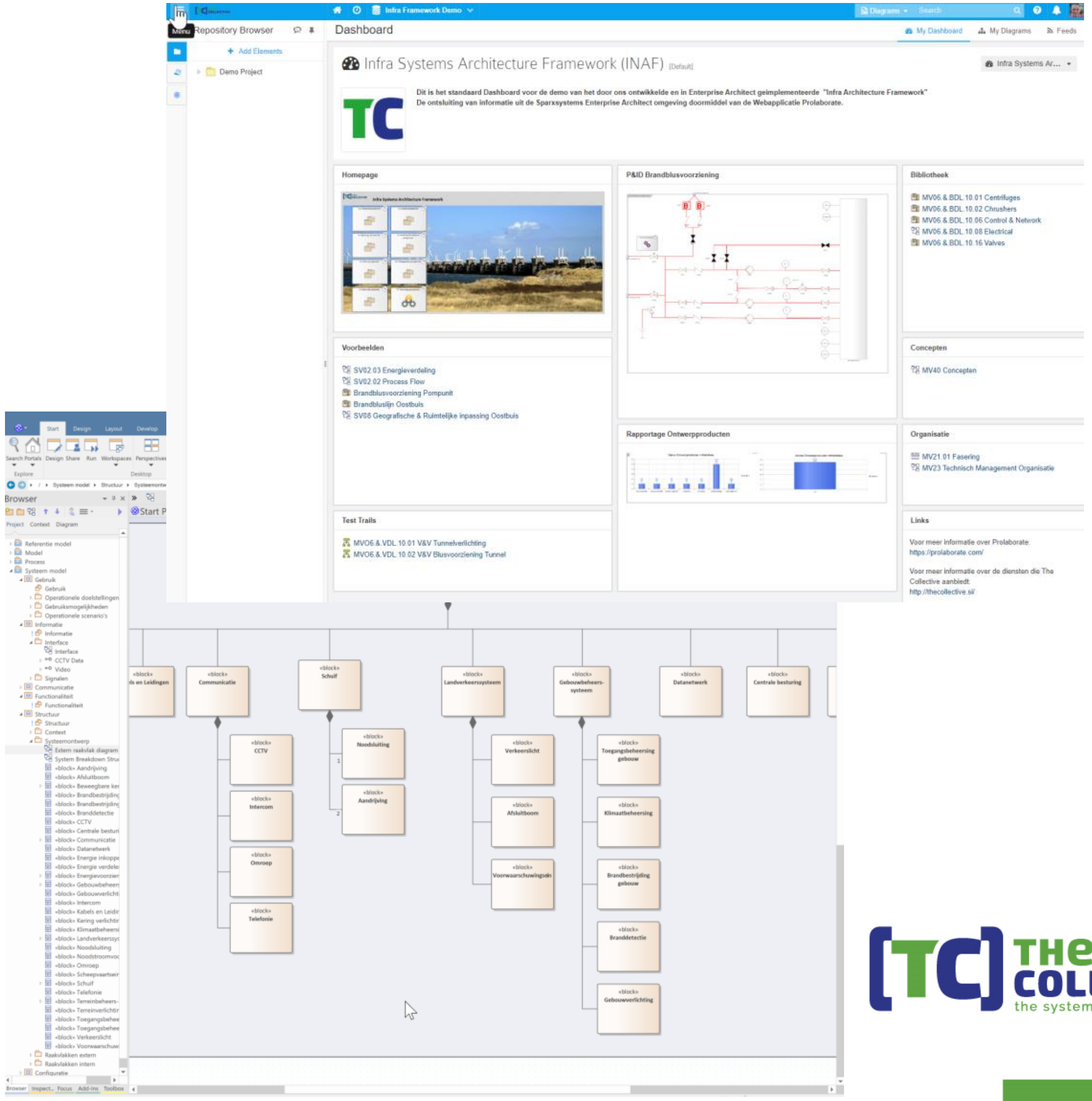
What do we need?

Working together in and
from a “single source of truth”

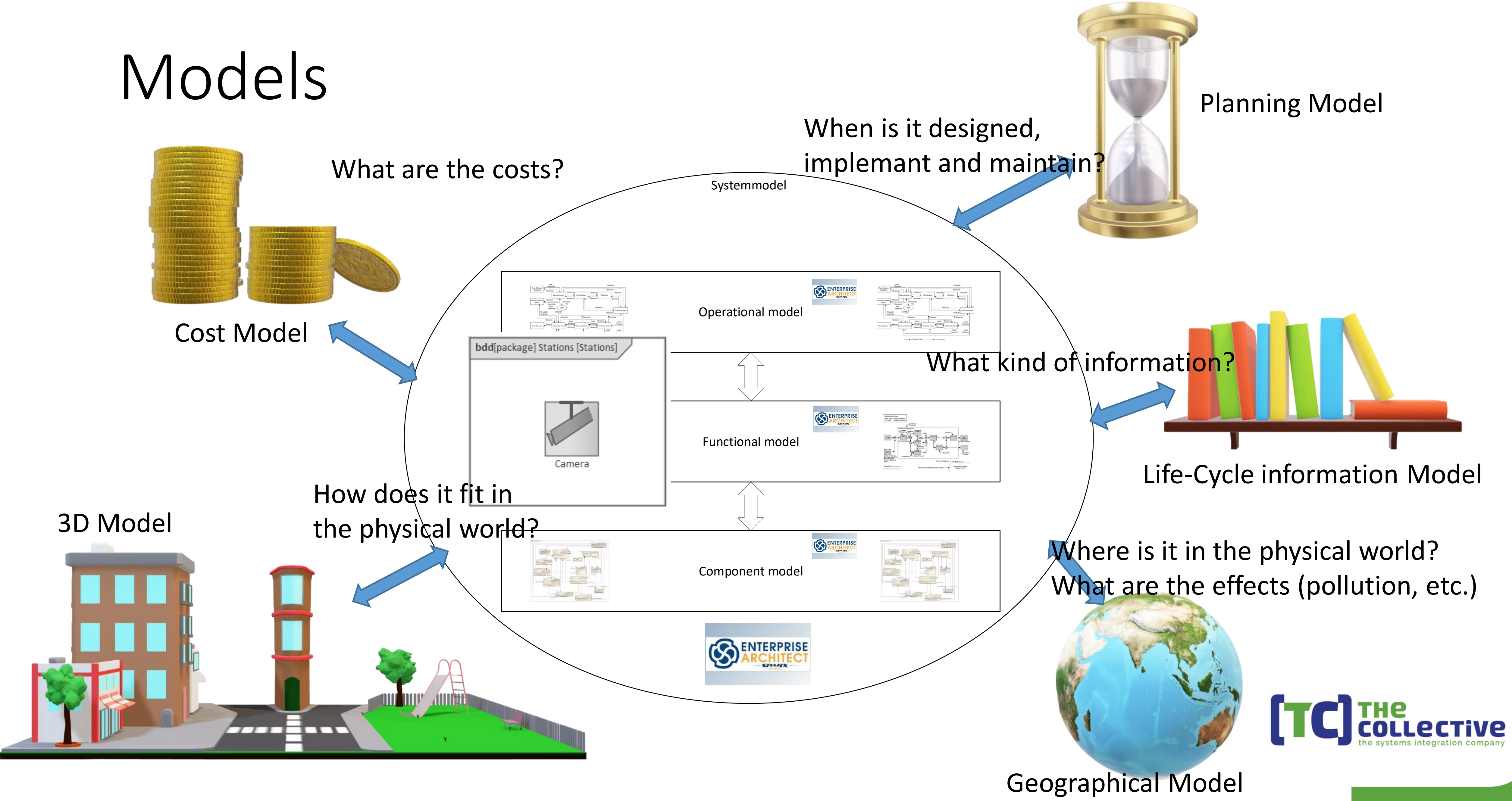
THE COLLECTIVE
the systems integration company

How: A model and a multi-user workspace!

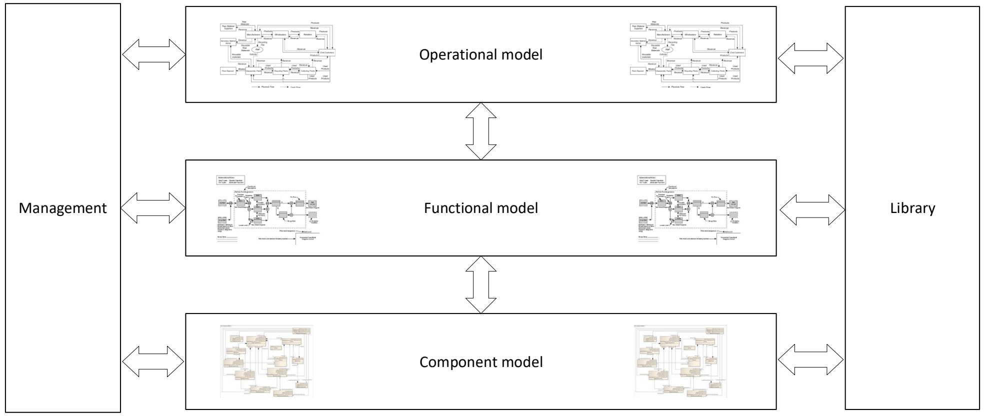




Models

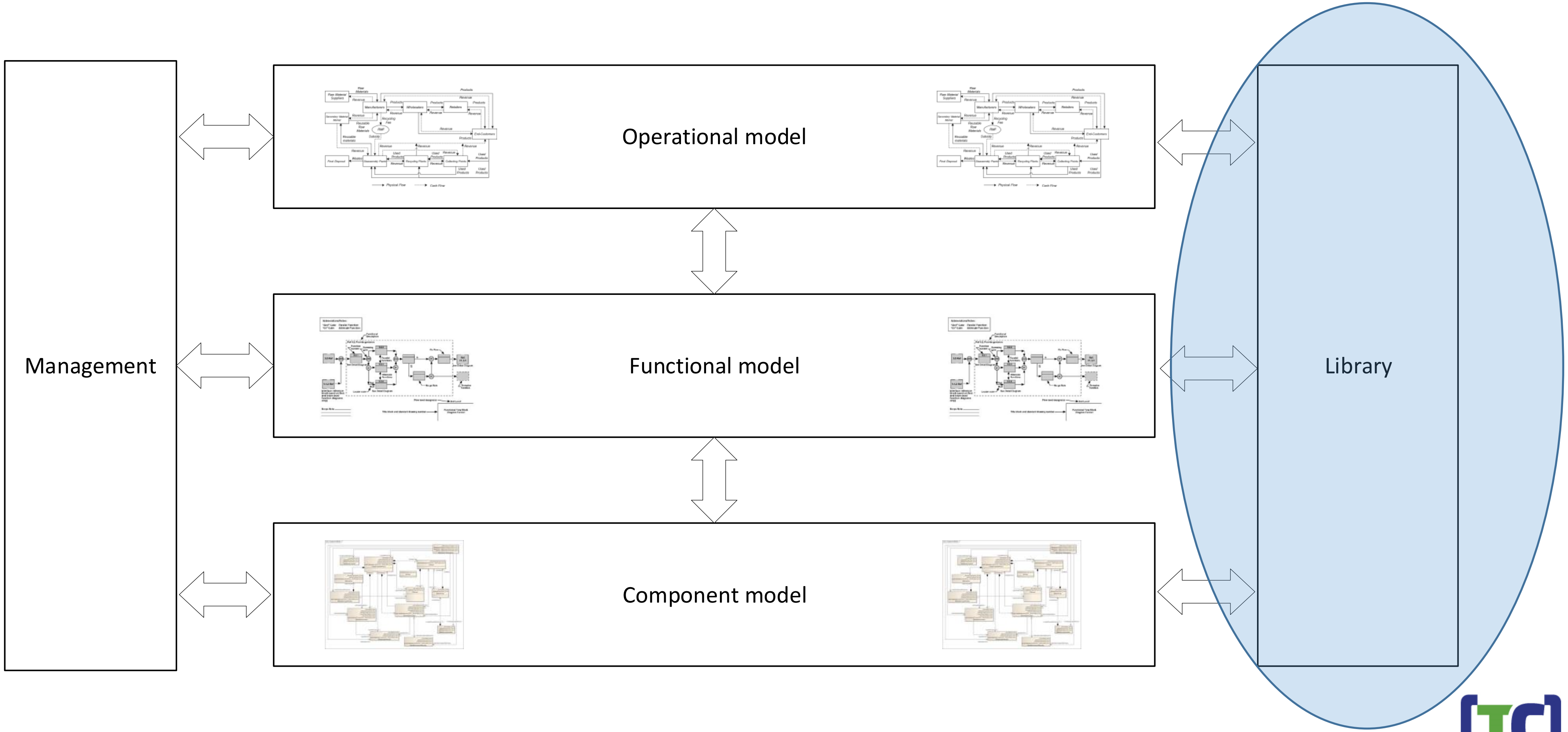

Cost Model
What are the costs?
When is it designed,
implemant and maintain?
Planning Model
Systemmodel
Operational model
bdd[package] Stations [Stations]
Camera
Functional model
Component model
ENTERPRISE ARCHITECT
What kind of information?
Life-Cycle information Model
3D Model
How does it fit in
the physical world?
Where is it in the physical world?
What are the effects (pollution, etc.)
Geographical Model
[TC] THE COLLECTIVE
the systems integration company

Model



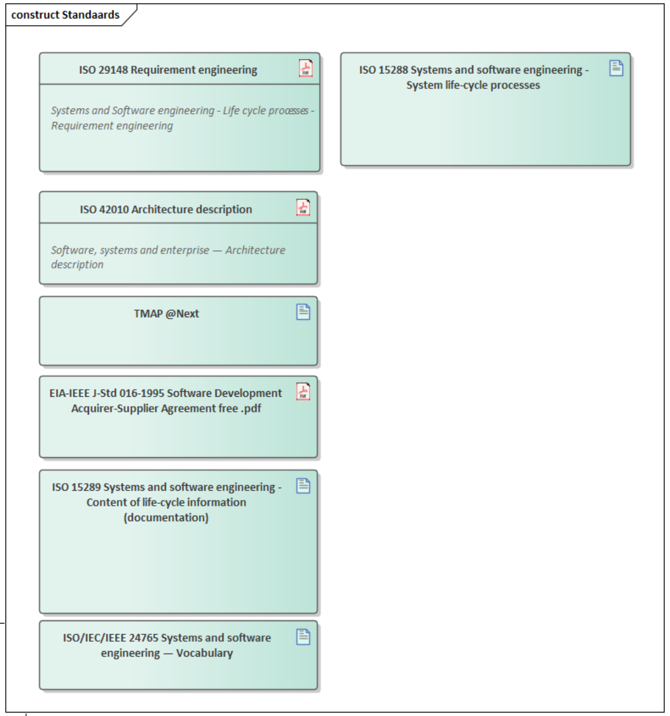
- Every model should have a goal:
  - The goal of the System model is to be a “single source of truth”
  - The goal of the Library model is to support the reuse of of signals, value types, etc.
  - The goal of the management model is to support the project management processes (for example, planning informatie- and configuratiemanagement
  - The goal of the operational model is to define the intended usage of the system
  - The goal of the functional model is to define the necessary functionality and systemstructure that is needed to support the intended usage
  - The goal of the component model is to define the technical solution

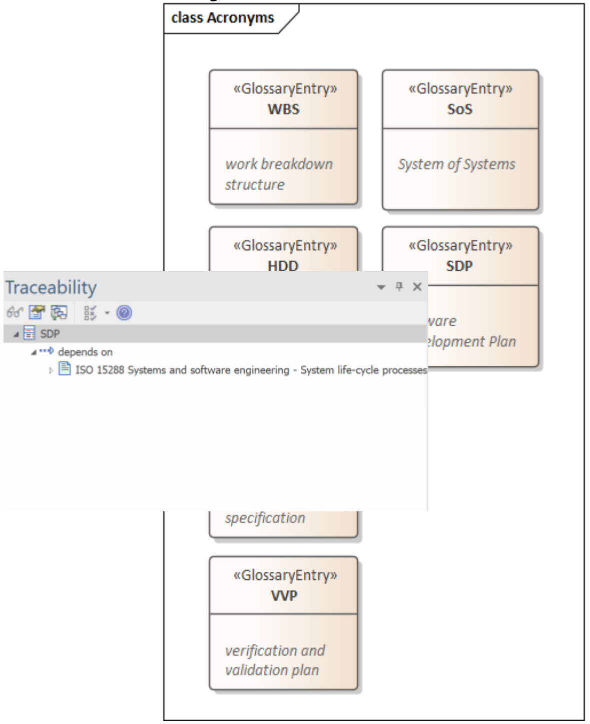
Model

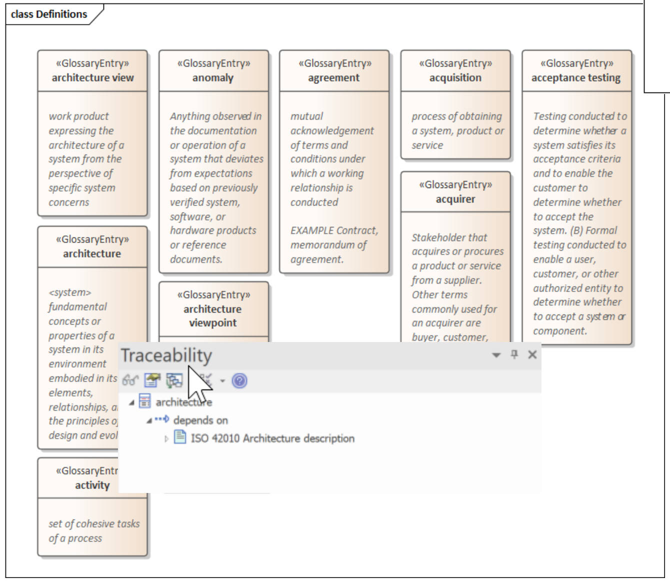

Management
Operational model
Functional model
Component model
Library

Library

- Use Standards
- Define Glossary
- Define Acronyms

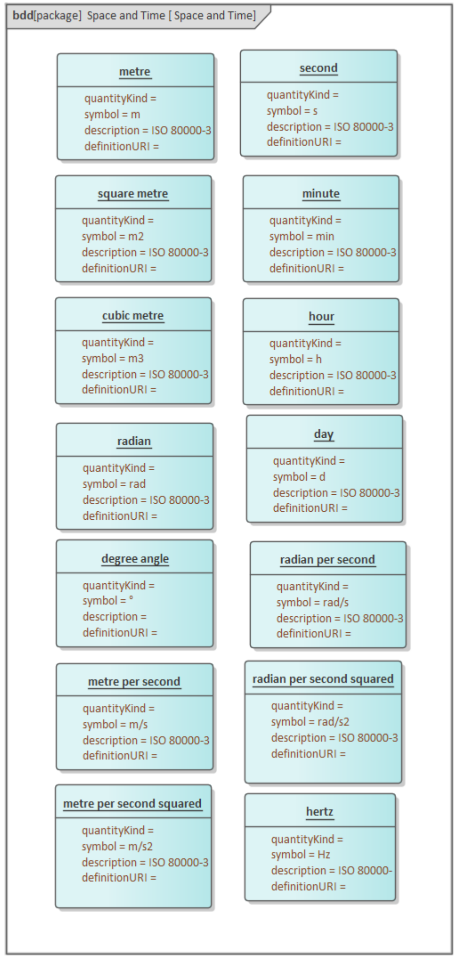


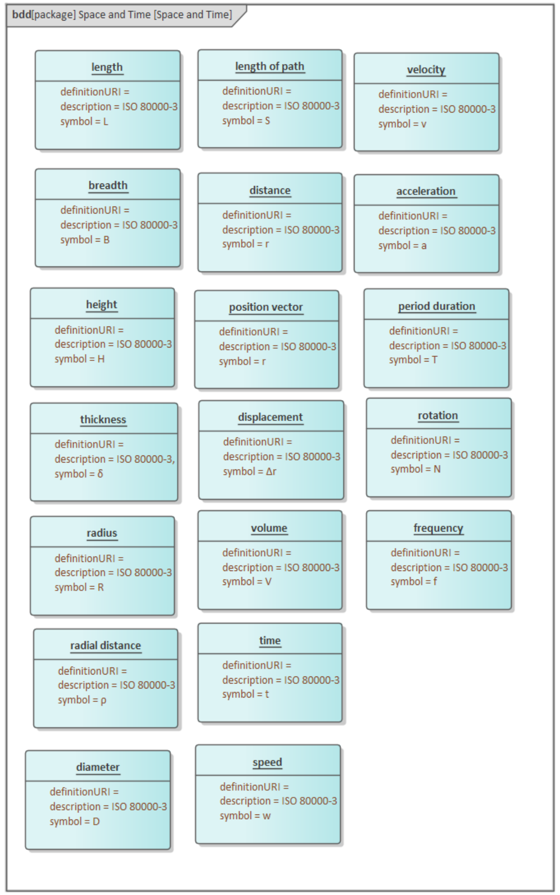


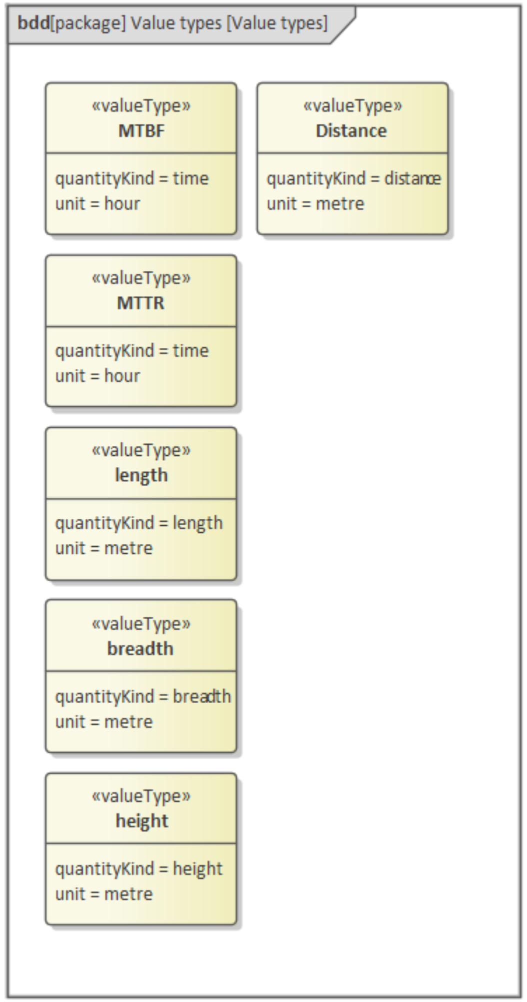


Library

- Define Engineering units
- Define Quantity kinds
- Define Value types



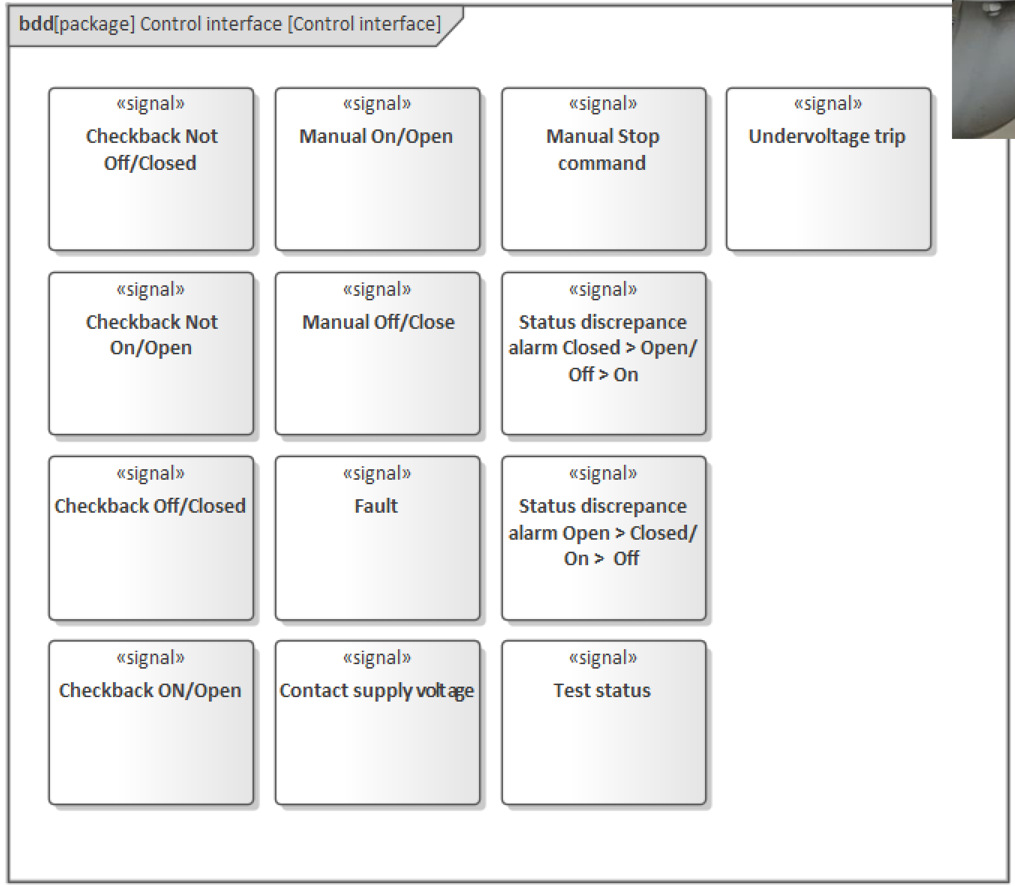




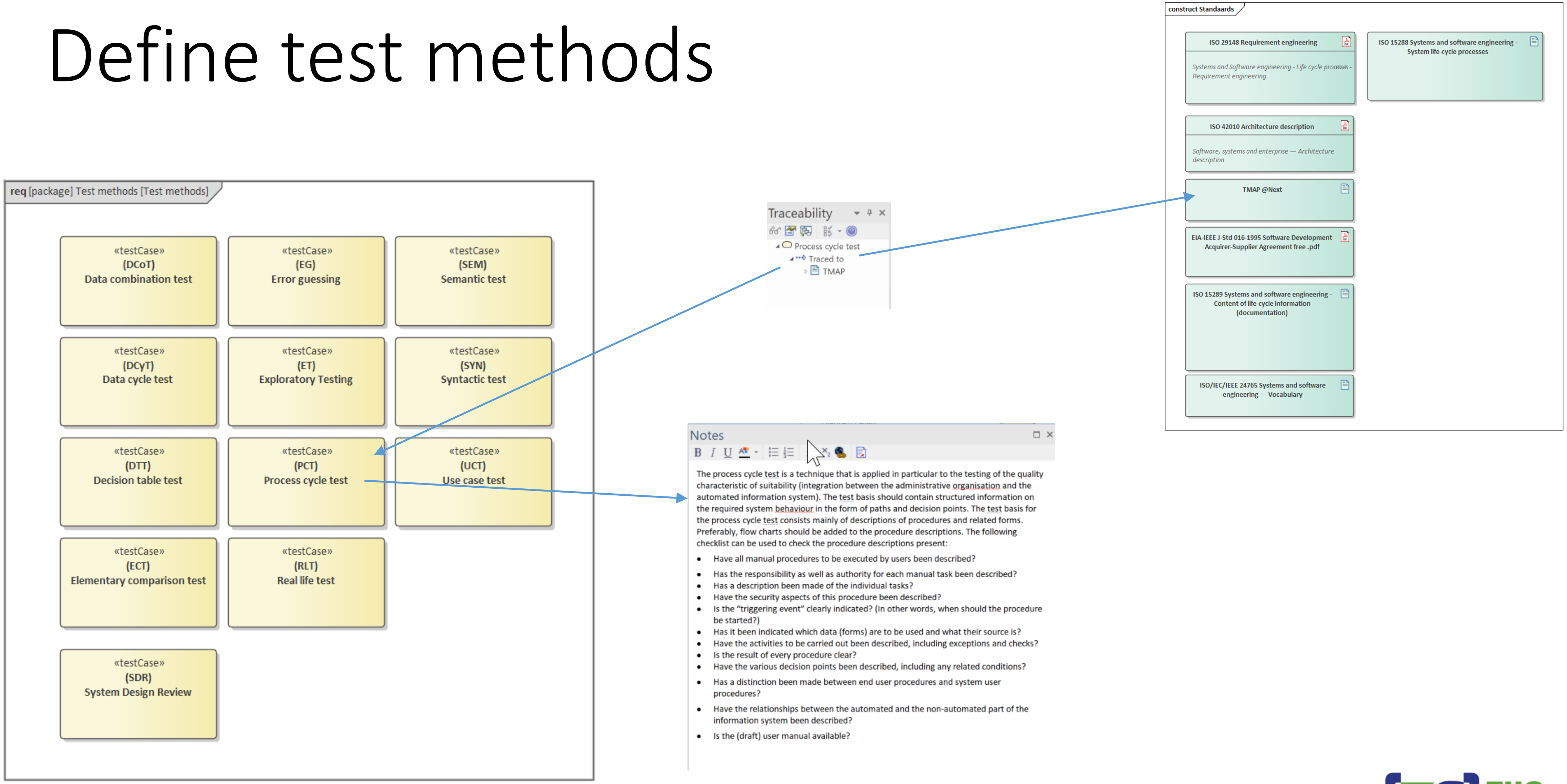
[TC] THE COLLECTIVE
the systems integration company

Library

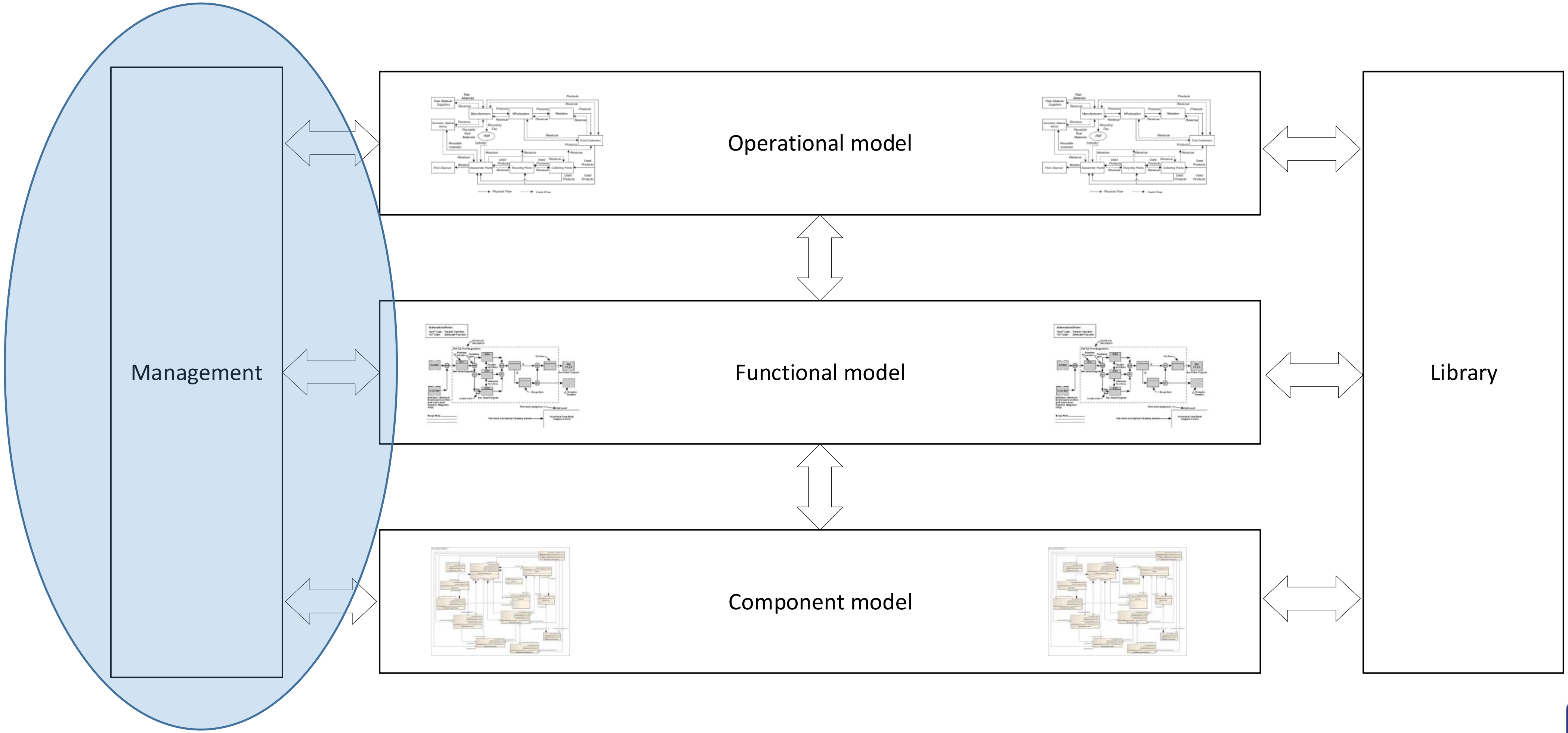
- Define reusable set of Signals



Define test methods



Model


Management
Operational model
Functional model
Component model
Library

Management information

- Organisation
- Processes and activities
- Products
- Planning
- Project risks

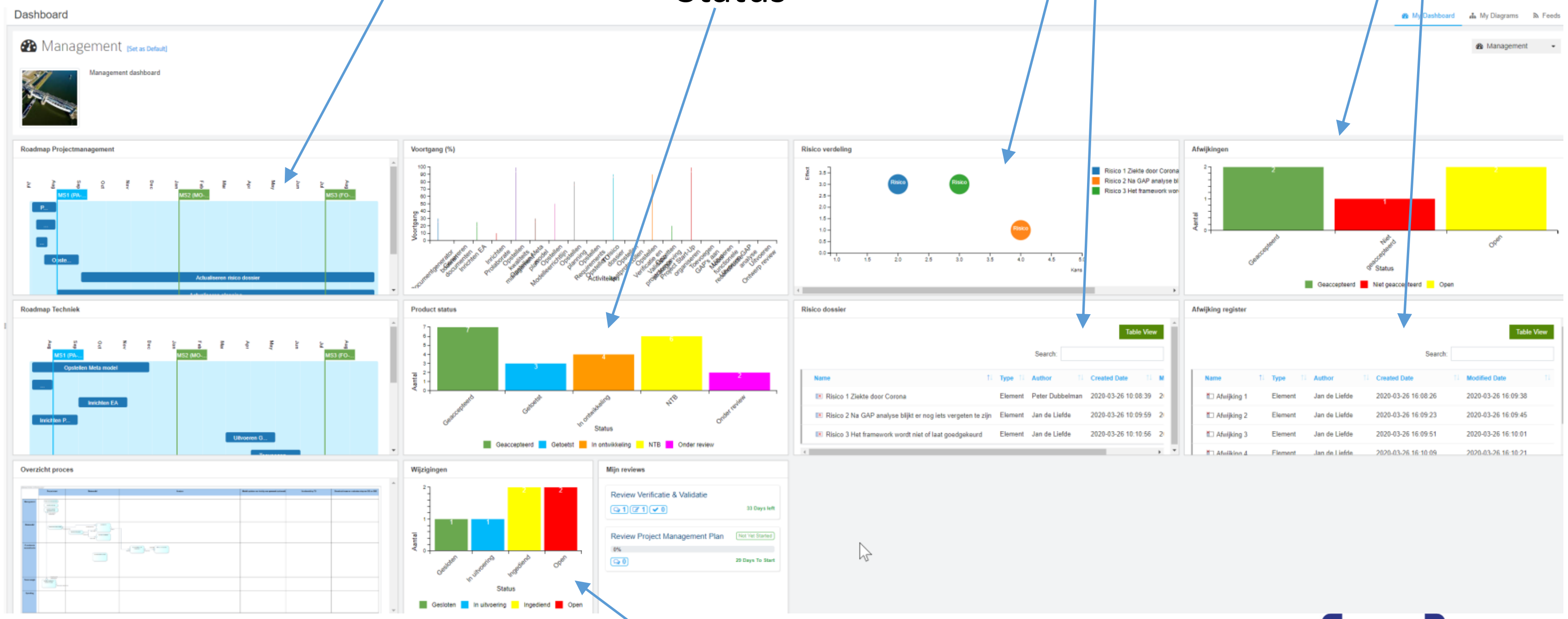
Roadmap

Status

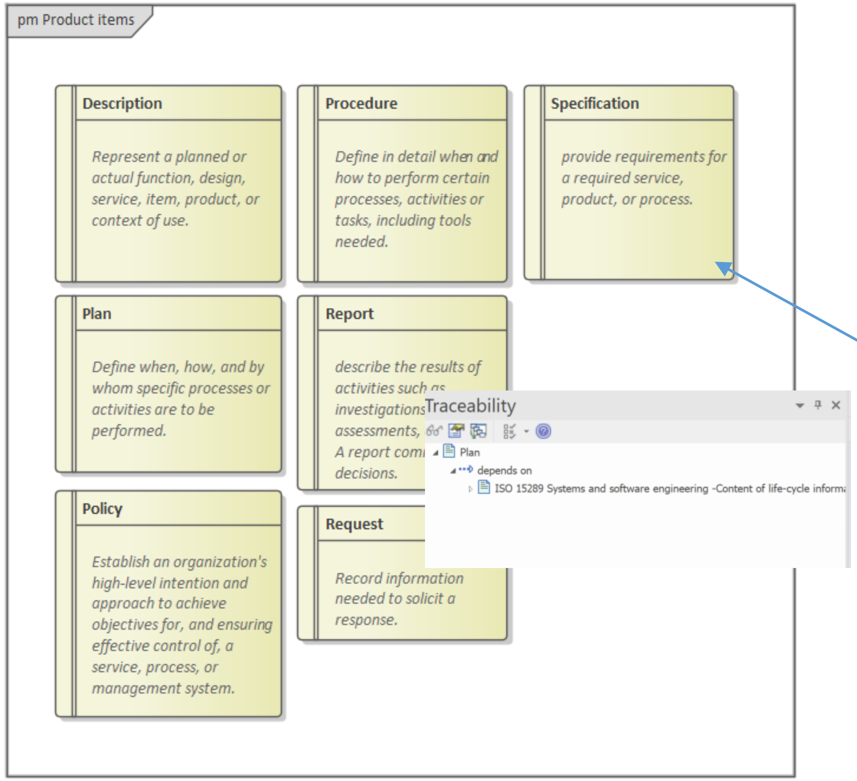
Risks

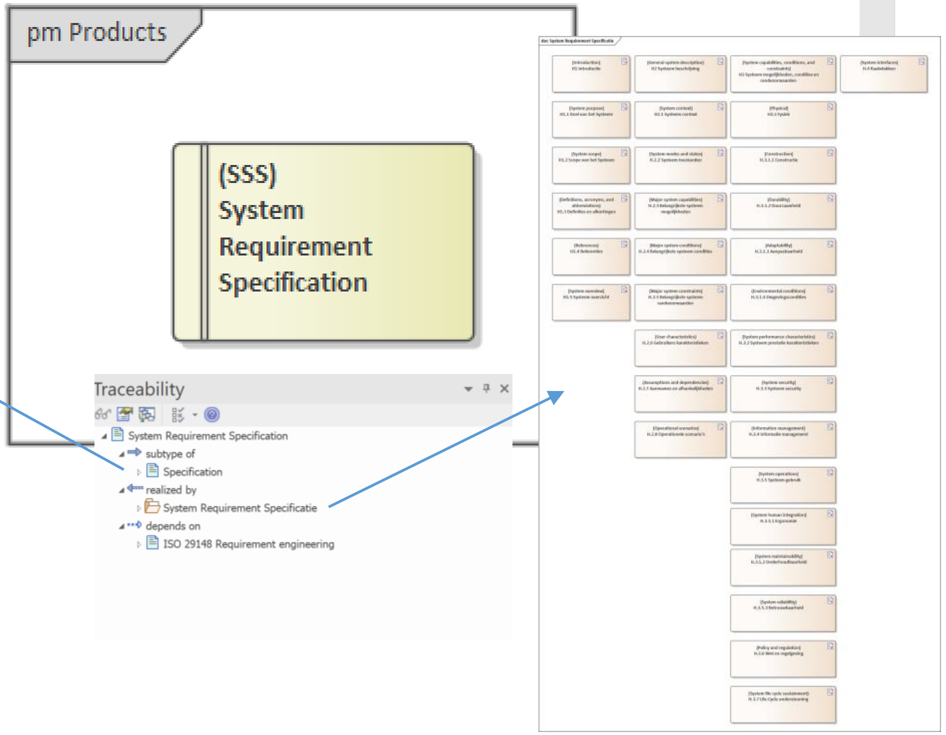
Deviations

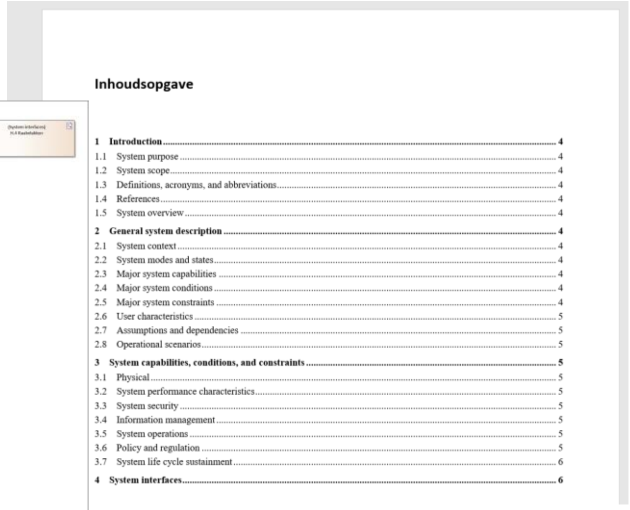
Changes



(Engineering) products



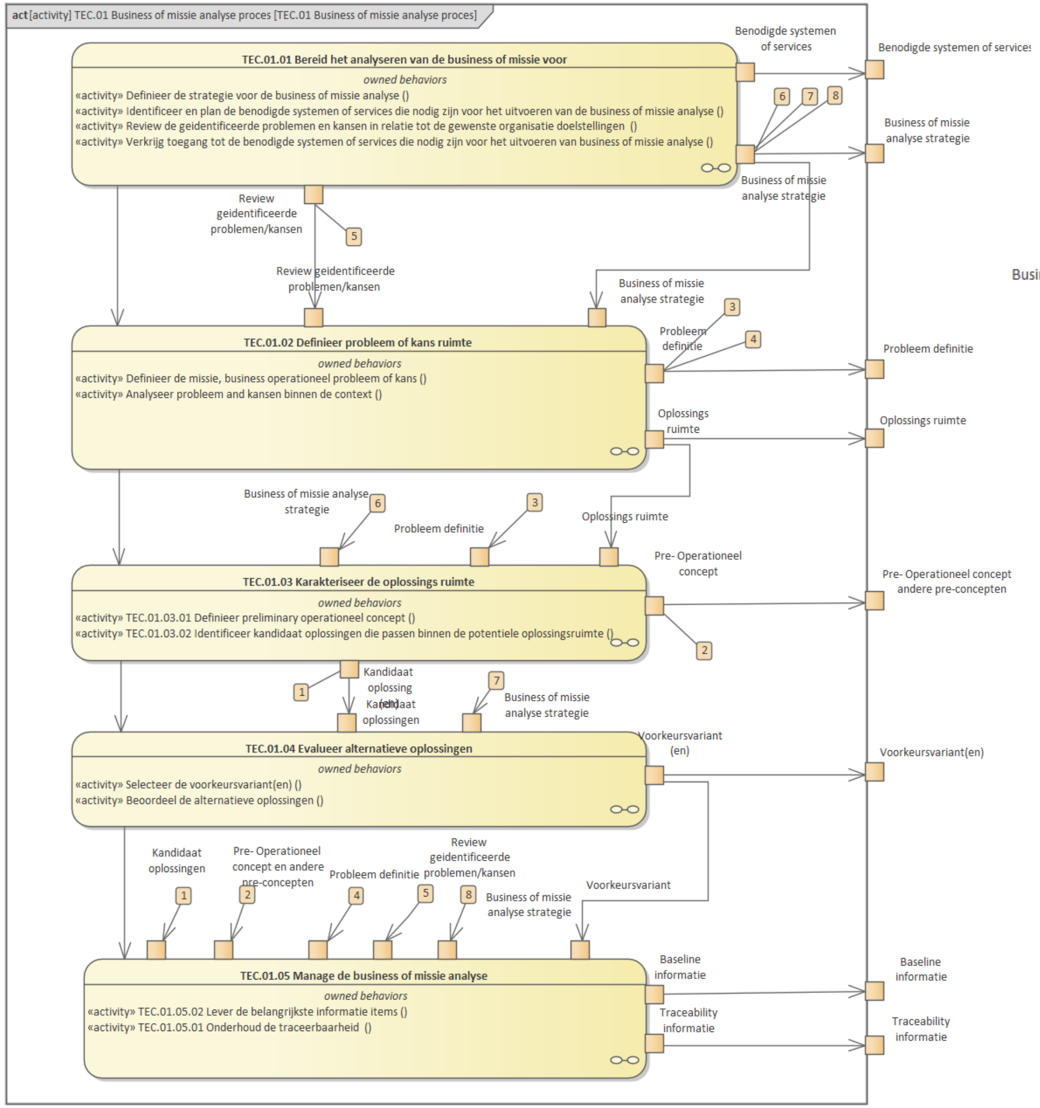


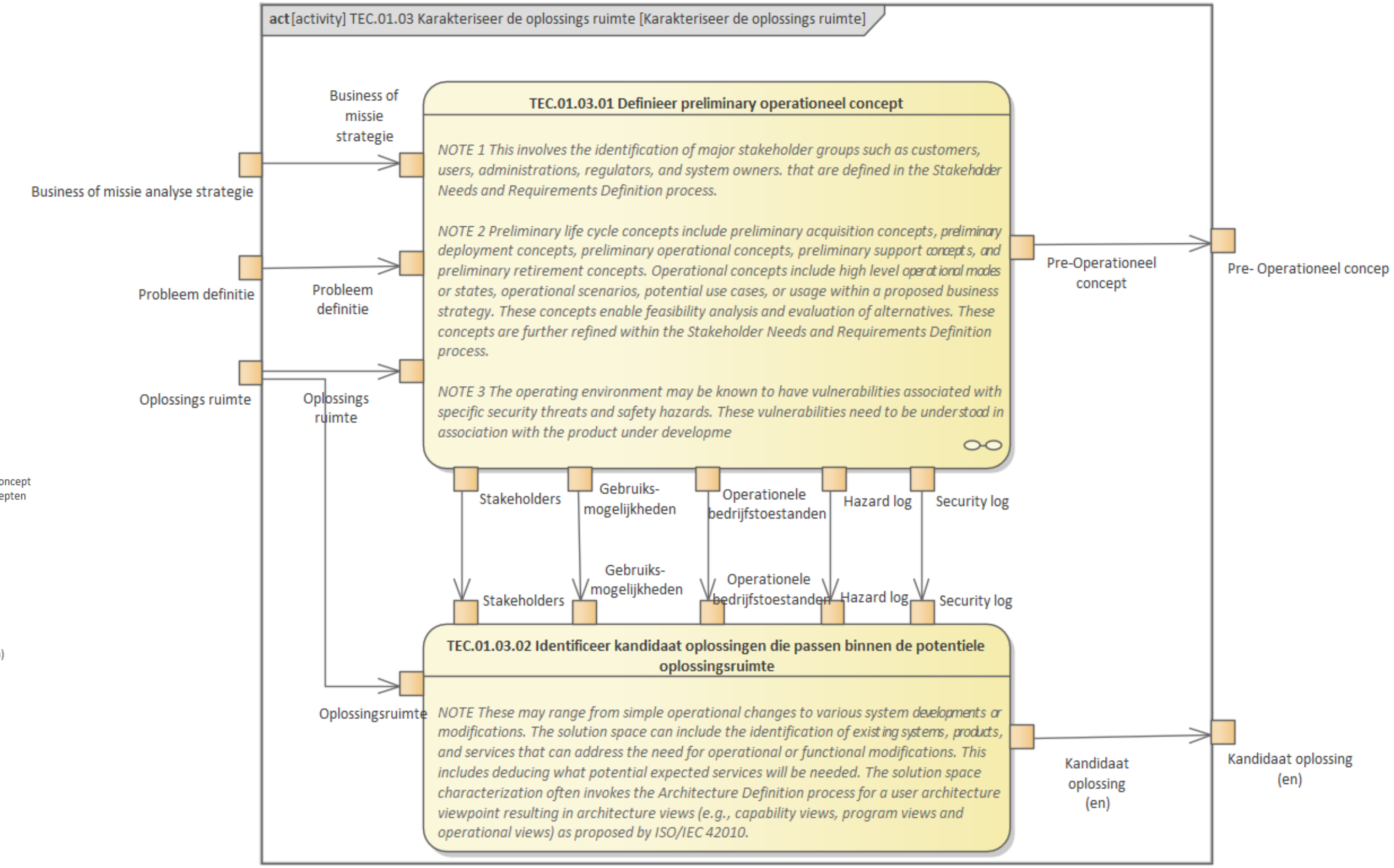


Inhoudsopgave

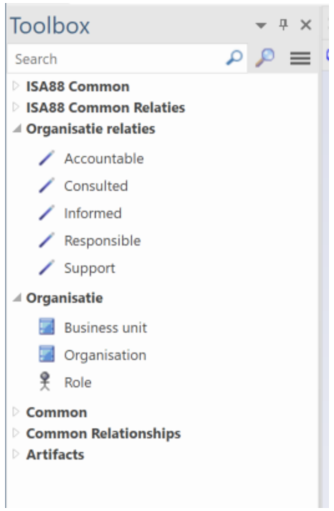
THE COLLECTIVE
the systems integration company

Management: Process



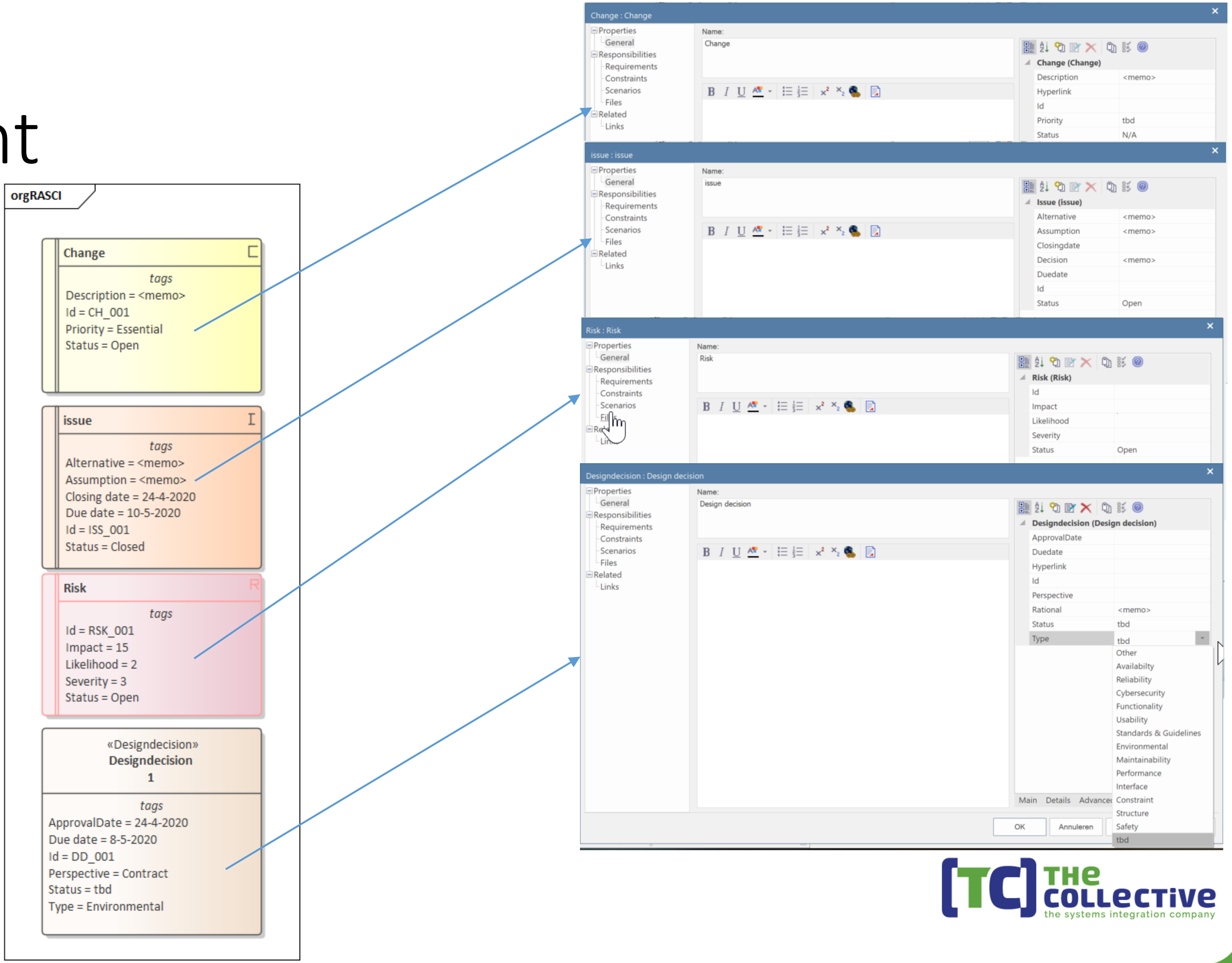


Management: RASCI



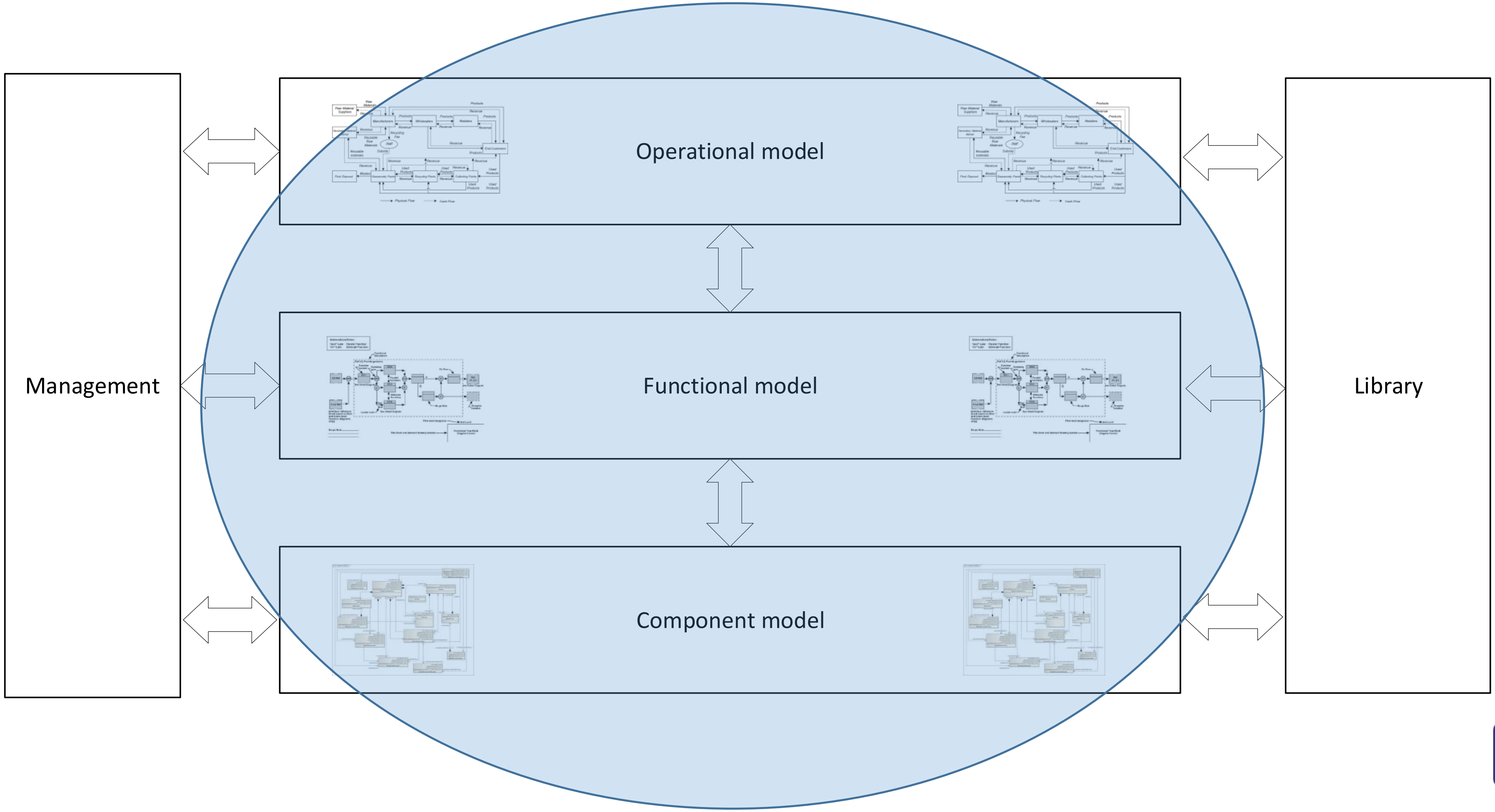
Target / Source	1 Identificeer de belangri...	2 Identificeer de Gebrui...	3 Identificeer Operatione...	4 Identificeer operationel...	5 Identificeer Safety Haz...	6 Identificeer security be...	7 Stel Pre-Operationeel ...
1 Business analist	responsible	responsible	responsible	responsible			responsible
2 Omgevingsmanagemer	consulted	support					
3 Project managemer	accountable	accountable	accountable	accountable	accountable	accountable	accountable
4 Safety analist					responsible		
5 Security analist						responsible	
6 Systeem architect		support			support		support

Management



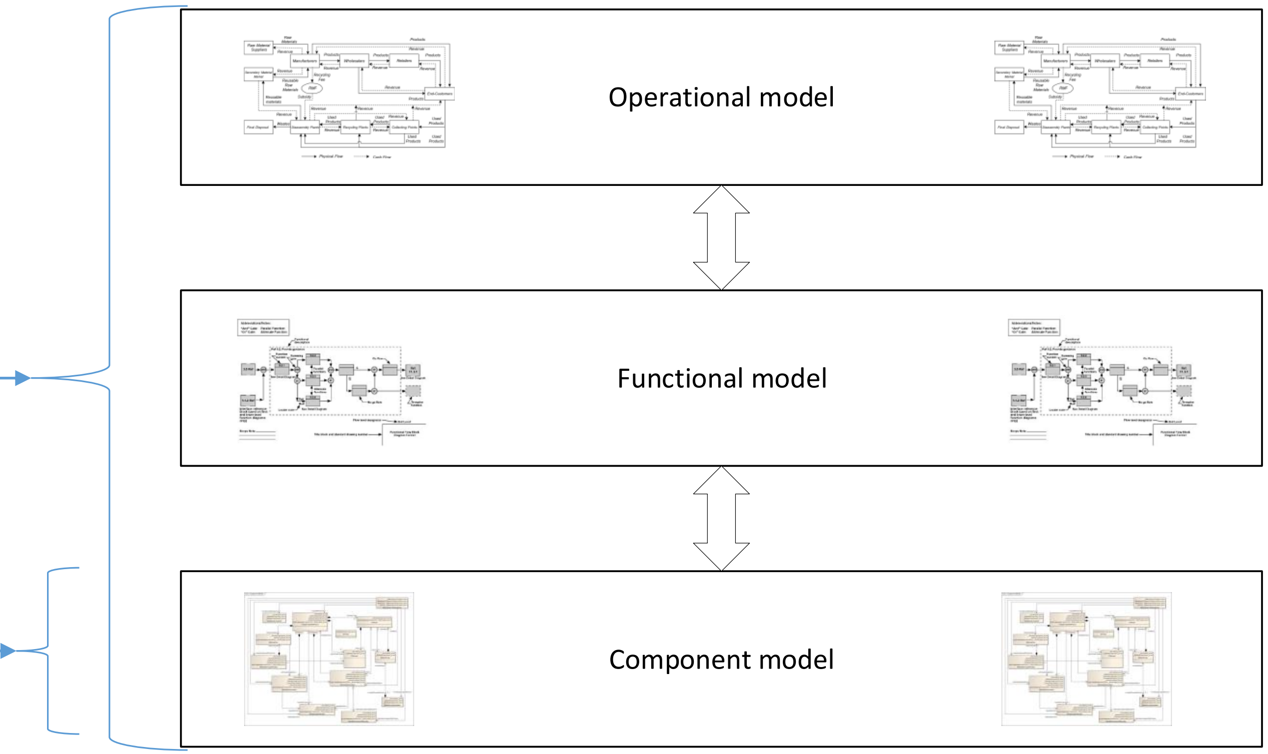
orgRASCI
Change
tags
Description = <memo>
Id = CH_001
Priority = Essential
Status = Open
issue
tags
Alternative = <memo>
Assumption = <memo>
Closing date = 24-4-2020
Due date = 10-5-2020
Id = ISS_001
Status = Closed
Risk
tags
Id = RSK_001
Impact = 15
Likelihood = 2
Severity = 3
Status = Open
«Designdecision»
Designdecision
1
tags
ApprovalDate = 24-4-2020
Due date = 8-5-2020
Id = DD_001
Perspective = Contract
Status = tbd
Type = Environmental
Change : Change
Properties
General
Responsibilities
Requirements
Constraints
Scenarios
Files
Related
Links
Name:
Change
Change (Change)
Description <memo>
Hyperlink
Id
Priority tbd
Status N/A
issue : issue
issue
Issue (issue)
Alternative <memo>
Assumption <memo>
Closingdate
Decision <memo>
Duedate
Id
Status Open
Risk : Risk
Risk
Risk (Risk)
Id
Impact
Likelihood
Severity
Status Open
Designdecision : Design decision
Design decision
Designdecision (Design decision)
ApprovalDate
Duedate
Hyperlink
Id
Perspective
Rational <memo>
Status tbd
Type tbd
Other
Availabilty
Reliability
Cybersecurity
Functionality
Usability
Standards & Guidelines
Environmental
Maintainability
Performance
Interface
Constraint
Structure
Safety
tbd
Main
Details
OK
Annuleren

Model


Management
Operational model
Functional model
Component model
Library

Phases and the Systemmodel

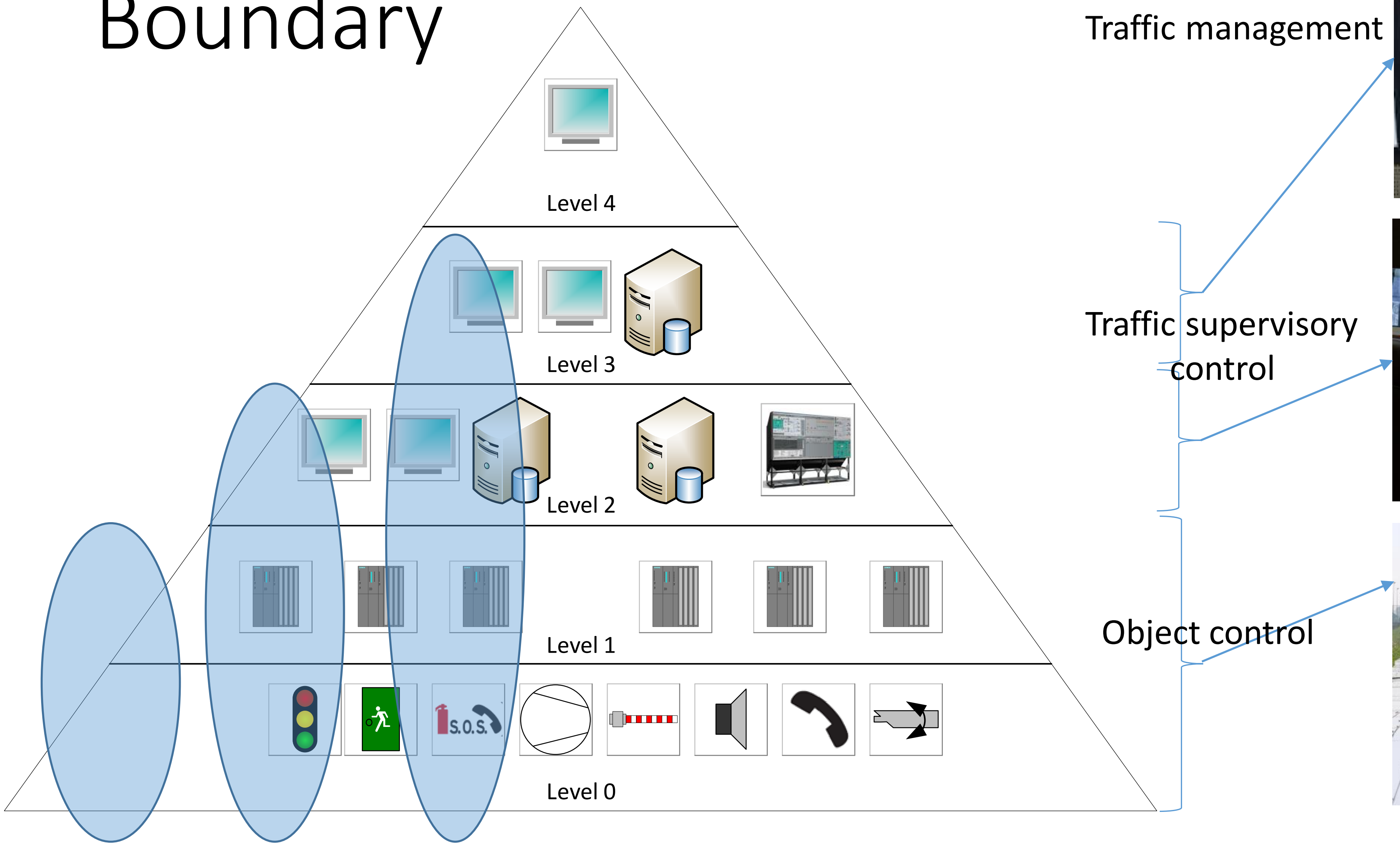
- Design phases
  - preliminary design
  - Final design
  - implementation design



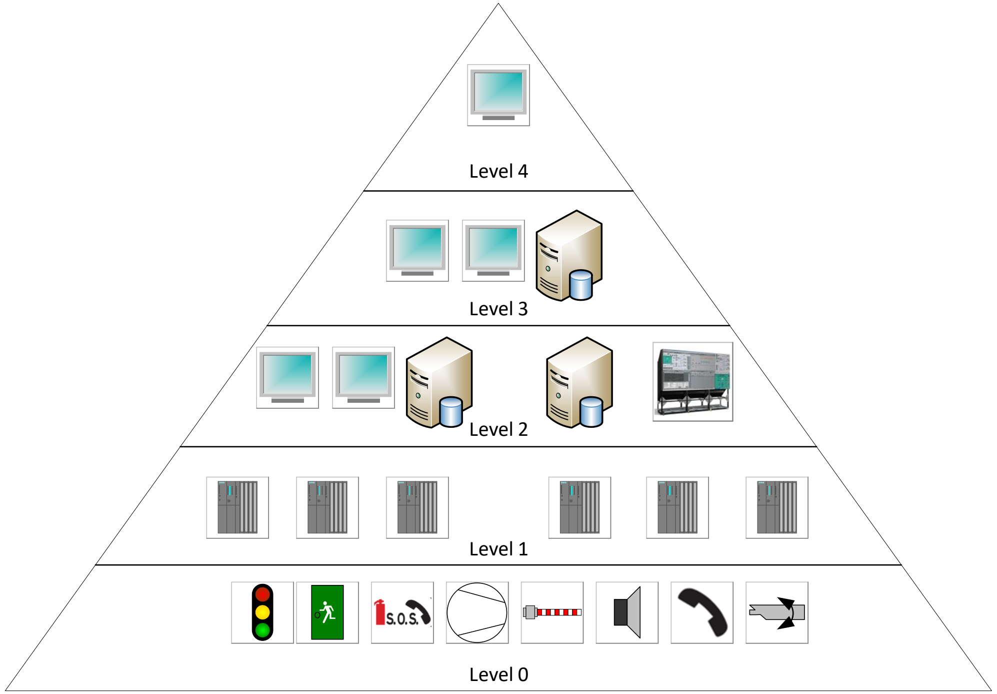
Operational model: Process

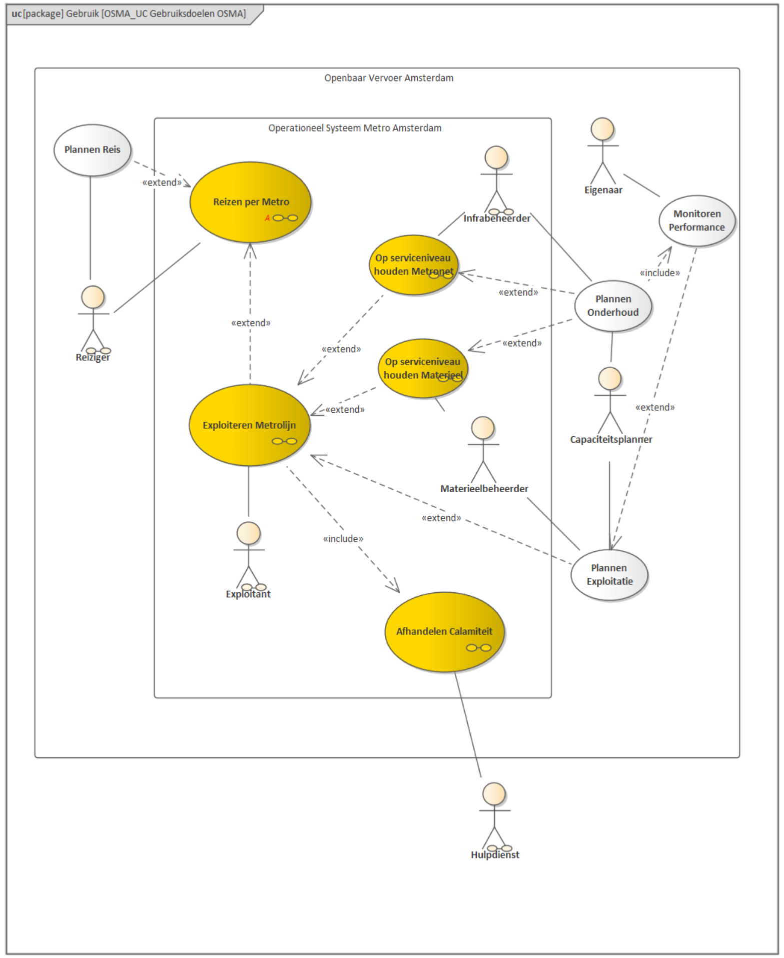
- Identify Stakeholders
- Identify Stakeholder goals
- Identify System boundary
- Define Capabilities
- Define operational usage
  - Operational scenario’s
  - System interaction
- Identify hazards
  - Mitigation
- Identify and define design constraints
  - Safety, usability, environment, maintainability, etc.
- Define acceptance criteria
- Traceability

Boundary


Level 4
Level 3
Level 2
Level 1
Level 0
Traffic management
Traffic supervisory
control
Object control

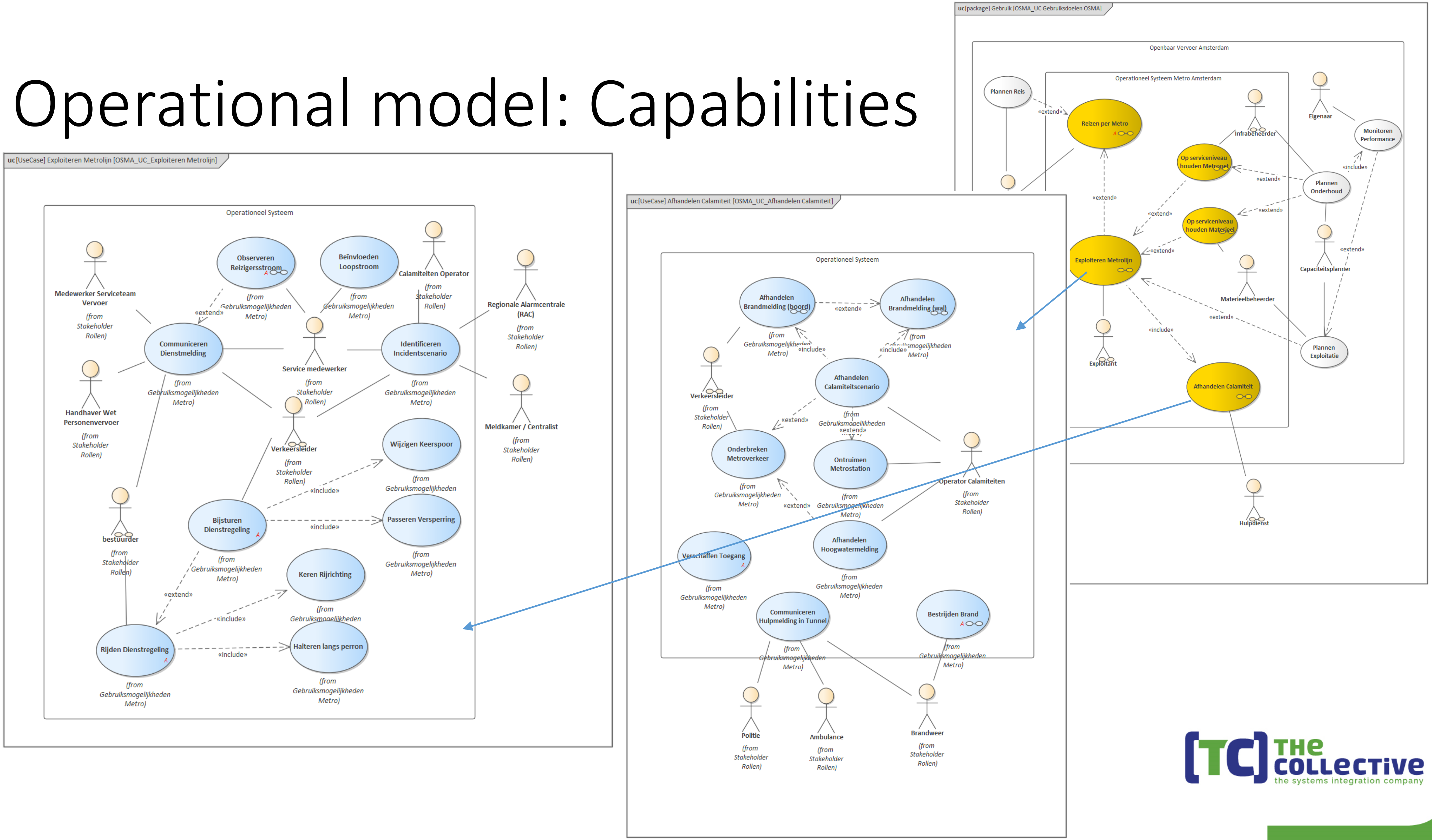
Operational model: Stakeholders, Goals & Systems Boundary





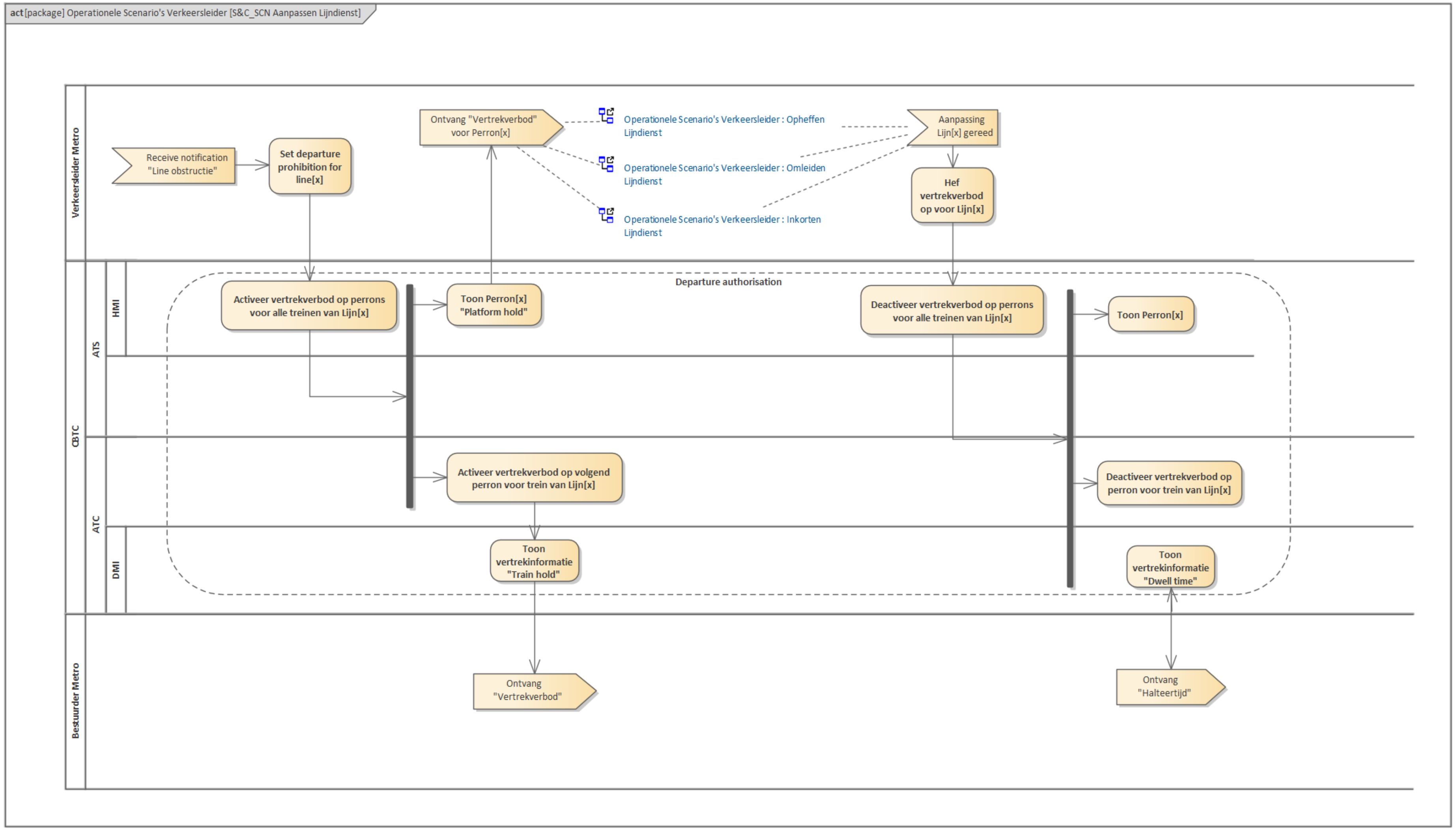
THE COLLECTIVE
the systems integration company

Operational model: Capabilities



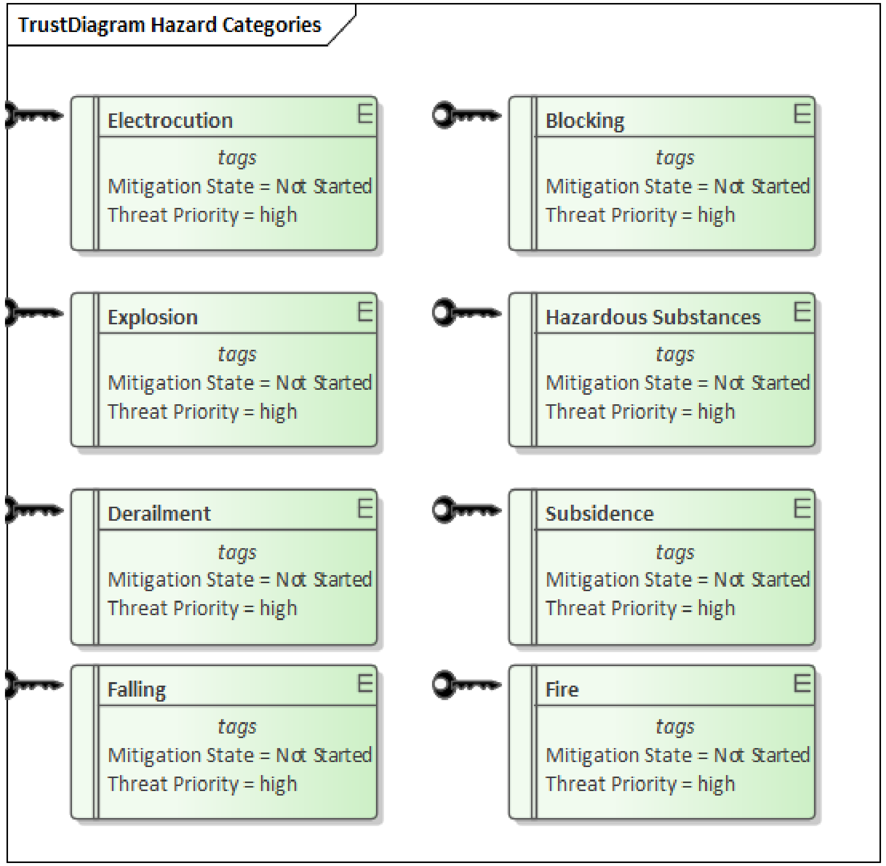
THE COLLECTIVE
the systems integration company

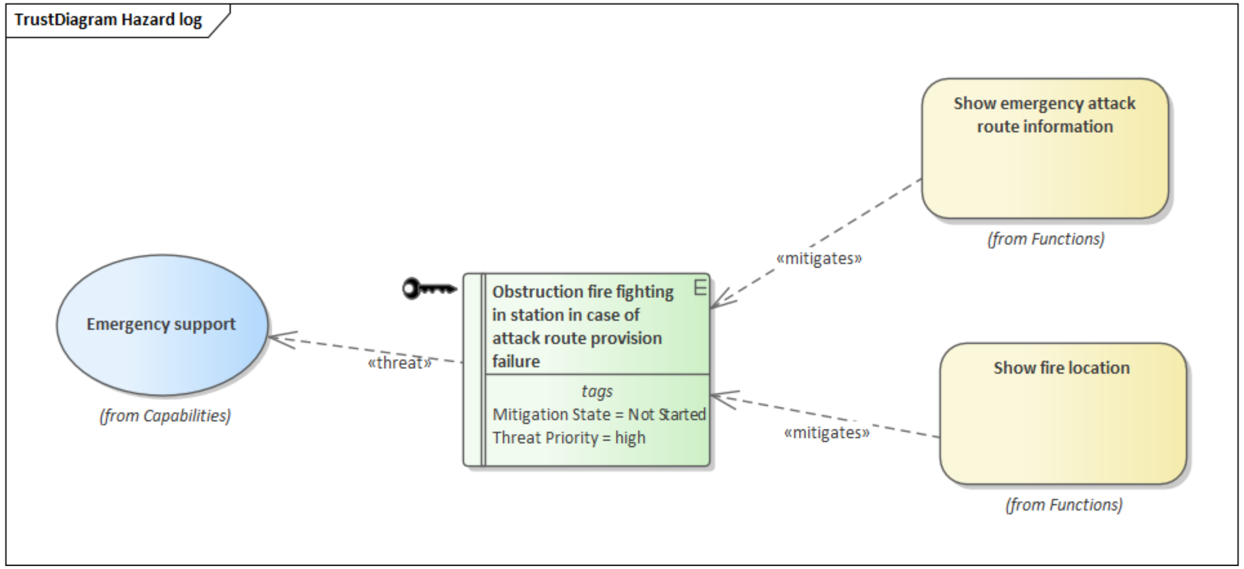
Operational model: Operational scenario’s



[TC] THE COLLECTIVE
the systems integration company

Operational model: Hazards





Operational model: Constraints

- Environmental
- Regulations
- Standards & Guidelines

ProRail

Assetmanagement

Ontwerpvoorschrift

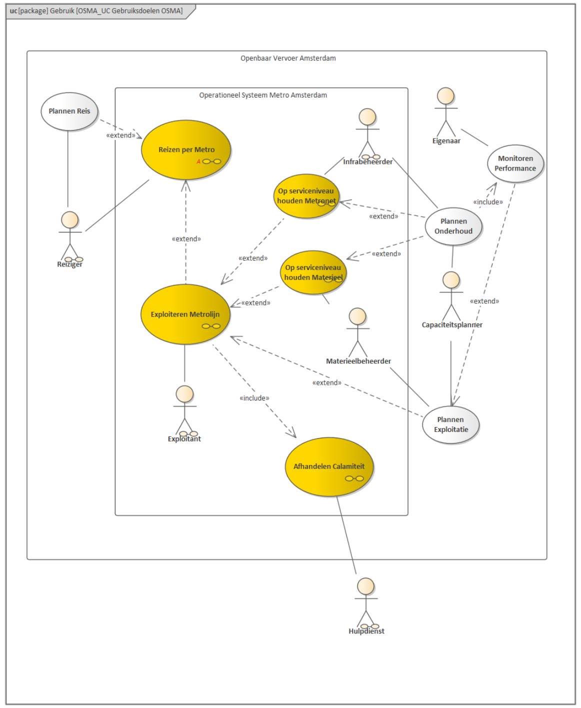
Tekenvoorschrift OBE- bladen
Overzicht - symbolen

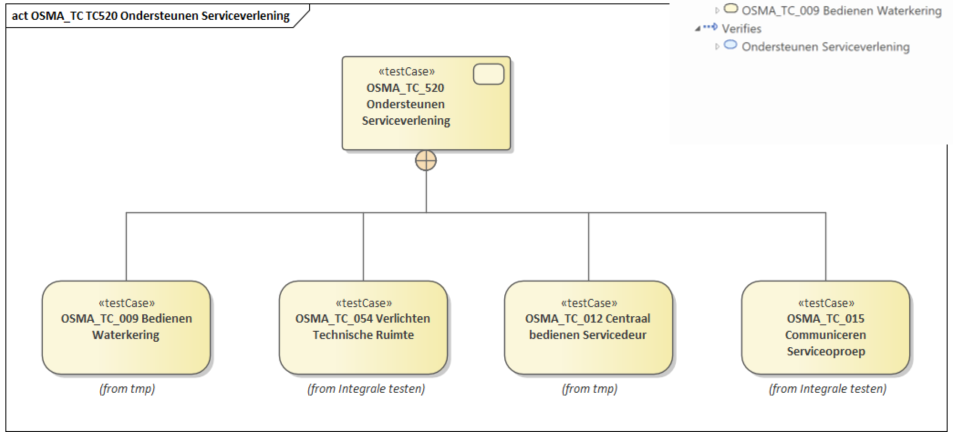
Beherende instantie:	*AM Architectuur en Techniek*
Inhoud verantwoordelijke:	*Manager AM Treinbeveiligingssystemen*
Status:	*Definitief*

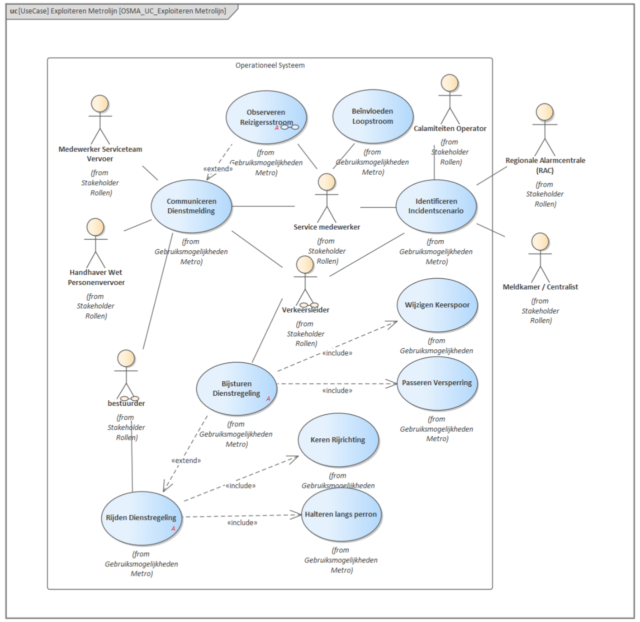
THE COLLECTIVE
the systems integration company

Operational model: Validation

- Define validation criteria
  - Define for every goal a validation criteria
  - Define for every capability a validation criteria

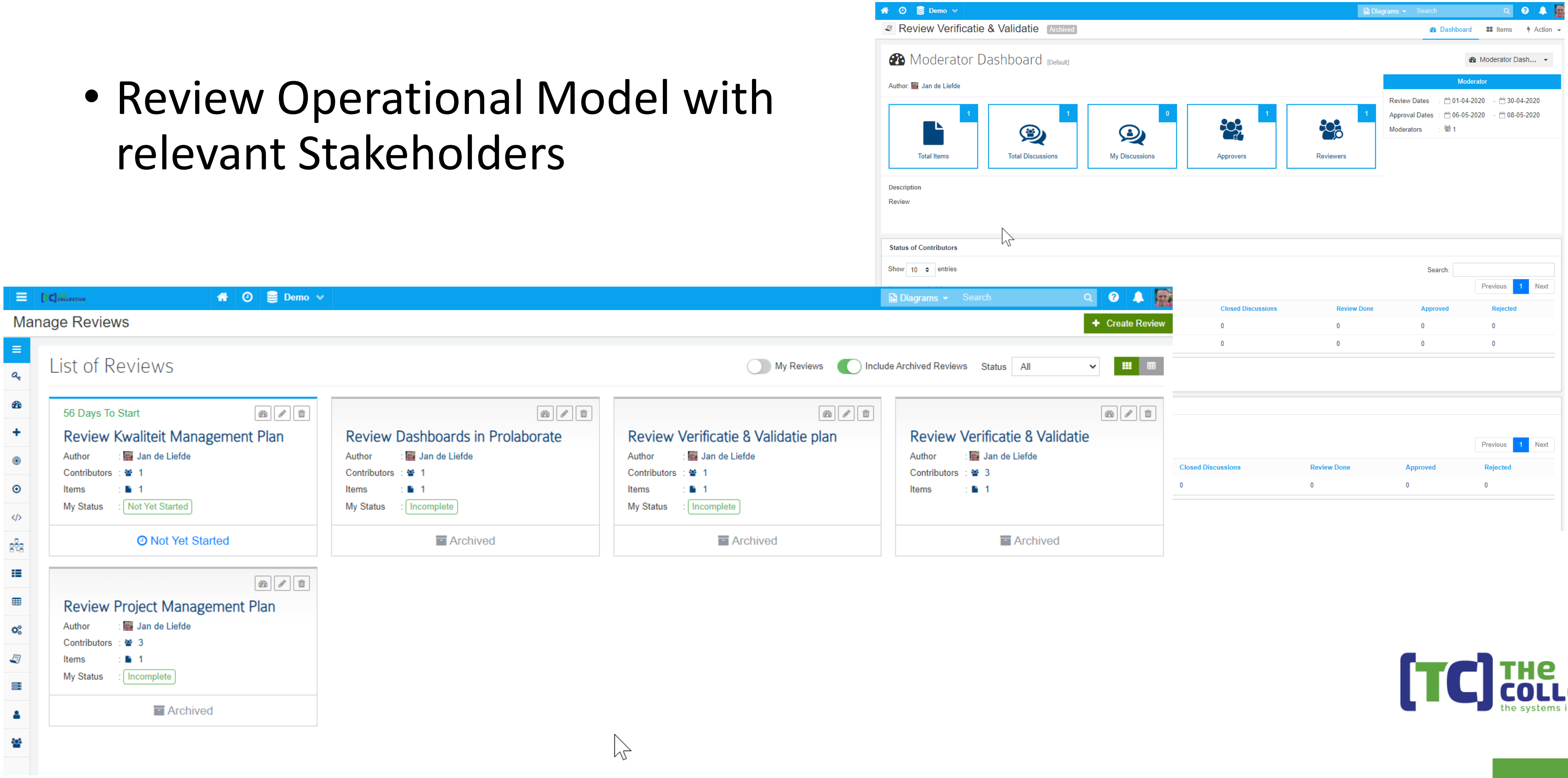






Operational Model: Review

- Review Operational Model with relevant Stakeholders

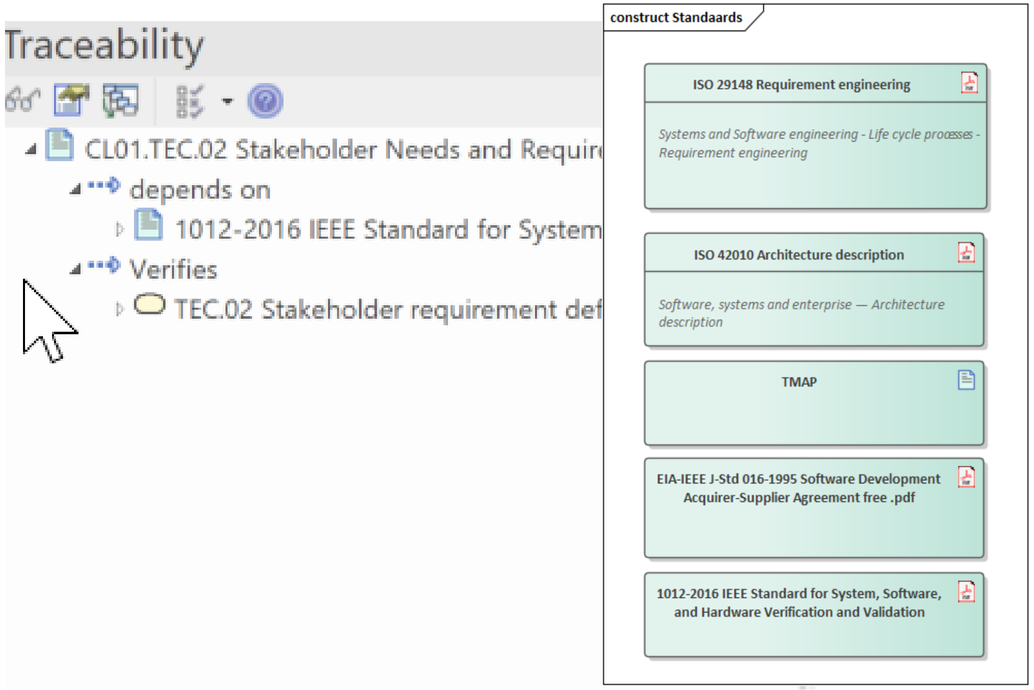


Operational model

orgTEC.02Stakeholder Needs and Requirements Definition V&V

CL01.TEC.02 Stakeholder Needs and Requirements Evaluation

- ☐ *Correctness:*
- ☐ *Verify and validate that the requirements satisfy the stakeholder needs for the system.*
- ☐ *Verify that the requirements comply with standards, references, regulations, policies, physical laws, and business rules.*
- ☐ *Validate that the requirements define the intended interaction of the system with its operating environment and other interfacing systems.*
- ☐ *Validate the suitability of the preferred candidate solution(s) to meet the stakeholder requirements.*
- ☐ *Consistency:*
- ☐ *Verify that all terms, concepts, and requirements are documented consistently in accordance with accepted syntax and structure (e.g., style guides and requirements modeling structure).*
- ☐ *Verify that there is consistency between assumptions, requirements, and between groups of requirements.*
- ☐ *Completeness:*
- ☐ *Validate that performance criteria and functionality are described in the requirements, within the assumptions and constraints of the operating environment and system boundaries.*
- ☐ *Verify that all stakeholders, or stakeholder classes (used here as groupings of stakeholders), are identified.*
- ☐ *Verify that the stakeholder requirements satisfy specified configuration management procedures. Verify stakeholder requirements are in a form suitable for requirements management throughout the life cycle.*
- ☐ *Readability:*
- ☐ *Verify that the documentation is legible, understandable, and unambiguous to the intended audience.*
- ☐ *Verify that the documentation defines all acronyms, mnemonics, abbreviations, terms, and symbols.*
- ☐ *Testability:*
- ☐ *Verify that objective acceptance criteria can be developed to validate the requirements.*



[TC] THE COLLECTIVE the systems integration company

Operational model

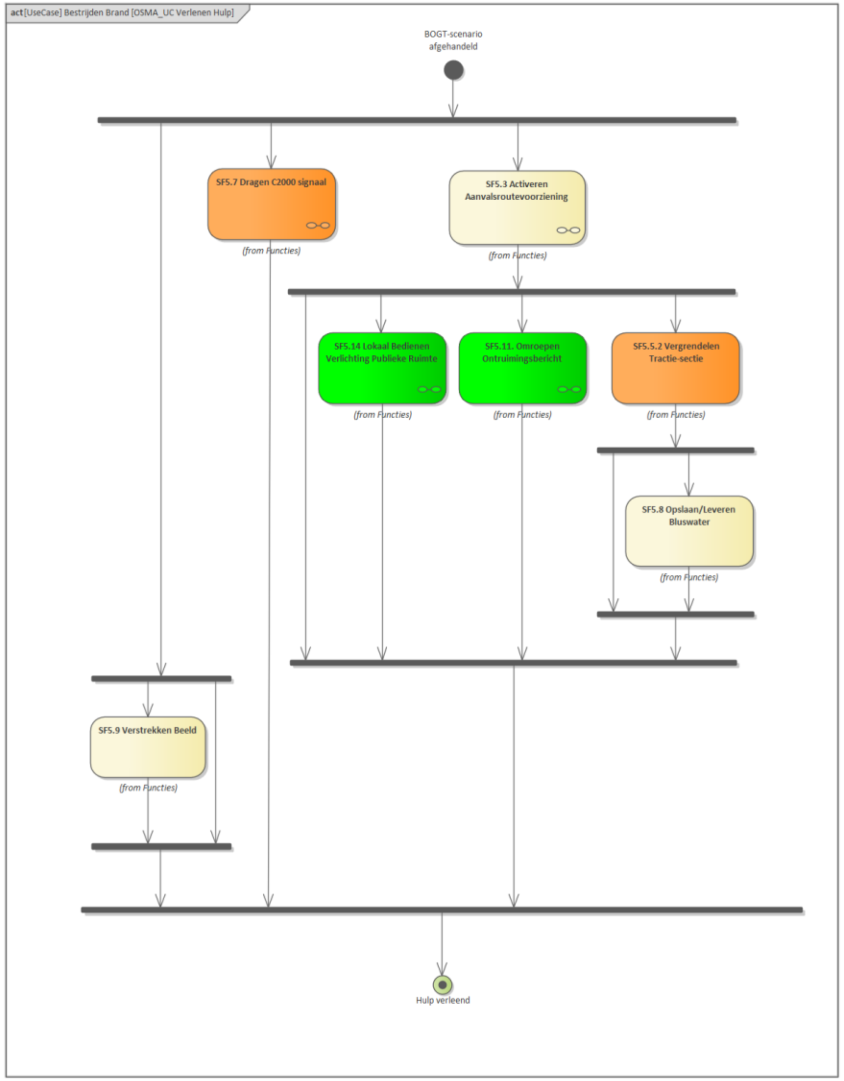
- Stakeholder identified
- Goals clear
- System capabilities defined
- Operational scenario's defined
- System interaction with users and external systems identified
- Hazards identified
- Constraints identified
- Design decisions documented
- Acceptance criteria defined
- Traceability guaranteed

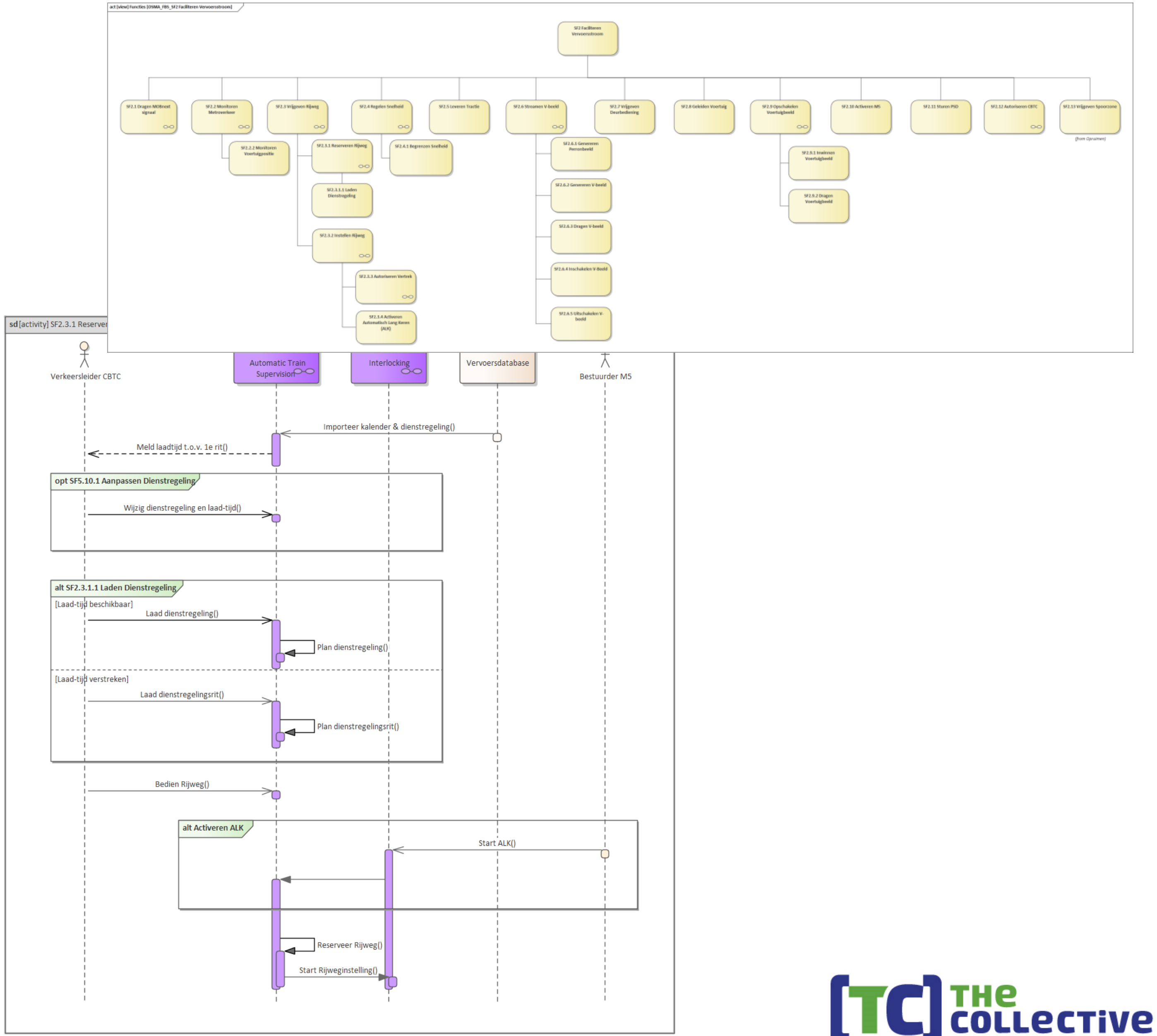
Functional model: Process

- Define system architecture
  - Define functional and physical boundary
  - Define functions
  - Define system elements
  - Define internal interfaces
  - Define interface protocols
  - Allocate functions to system elements
  - Allocate information to detailed interfaces
  - Define system element parametrics (power consumption, availability, heat dissipation, stability, etc)
  - Define constraints
  - Define testcases
  - Traceability

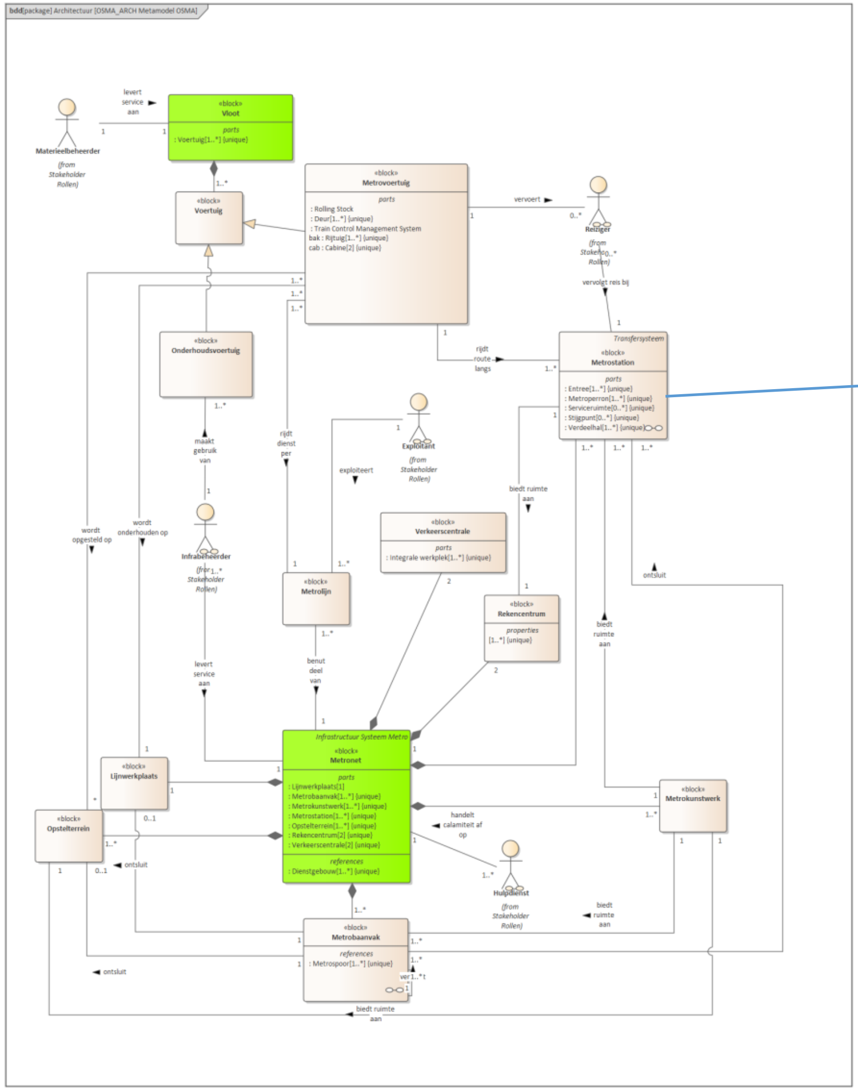
THE COLLECTIVE
the systems integration company

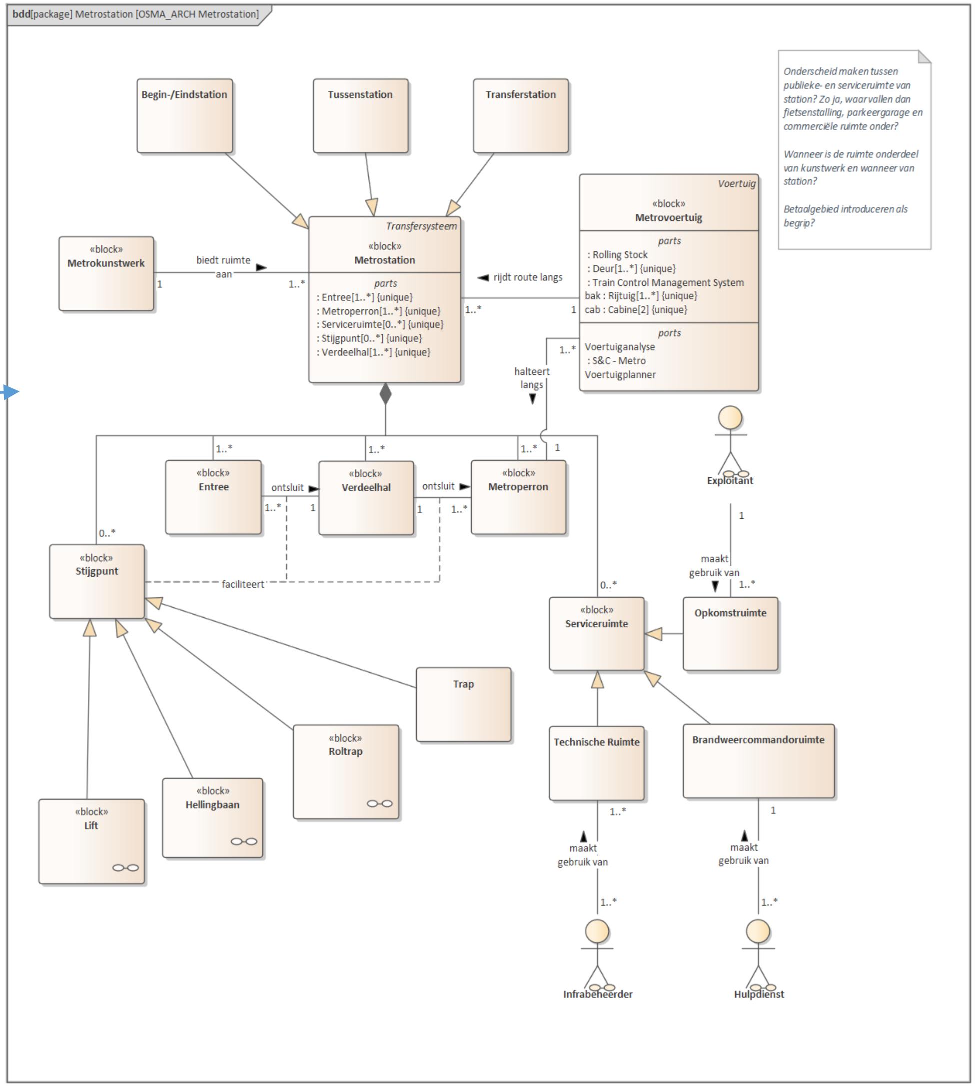
Functional model





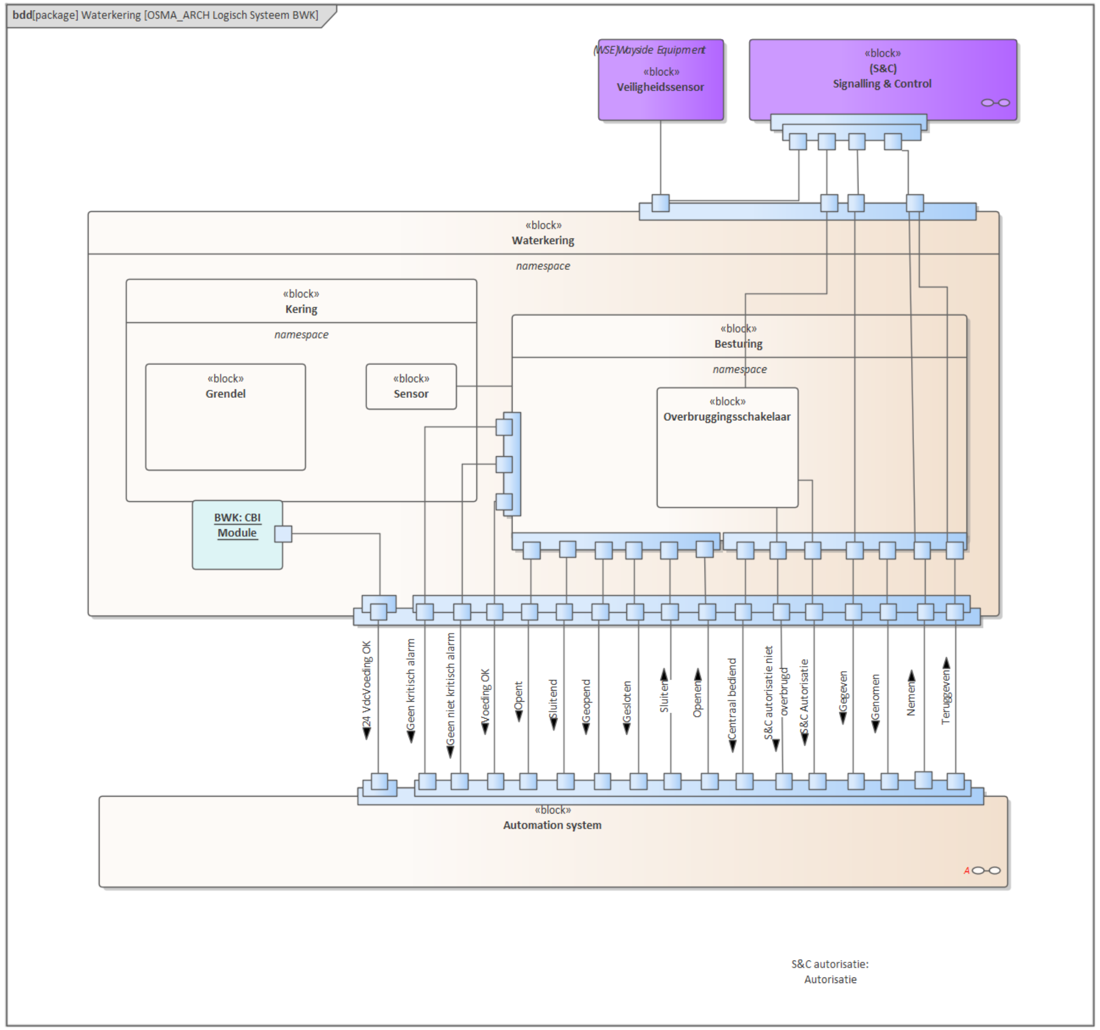
System elements

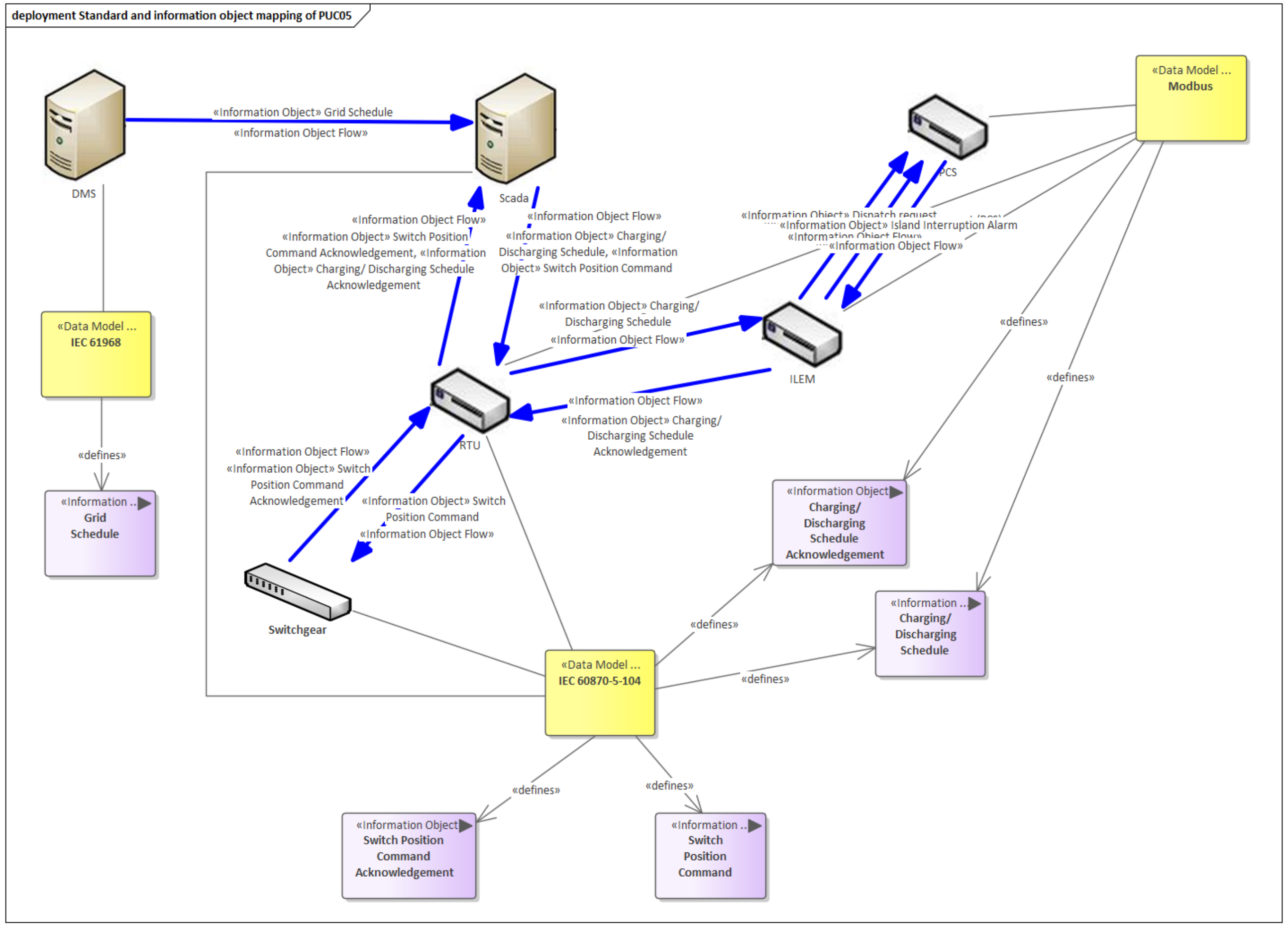




THE COLLECTIVE
the systems integration company

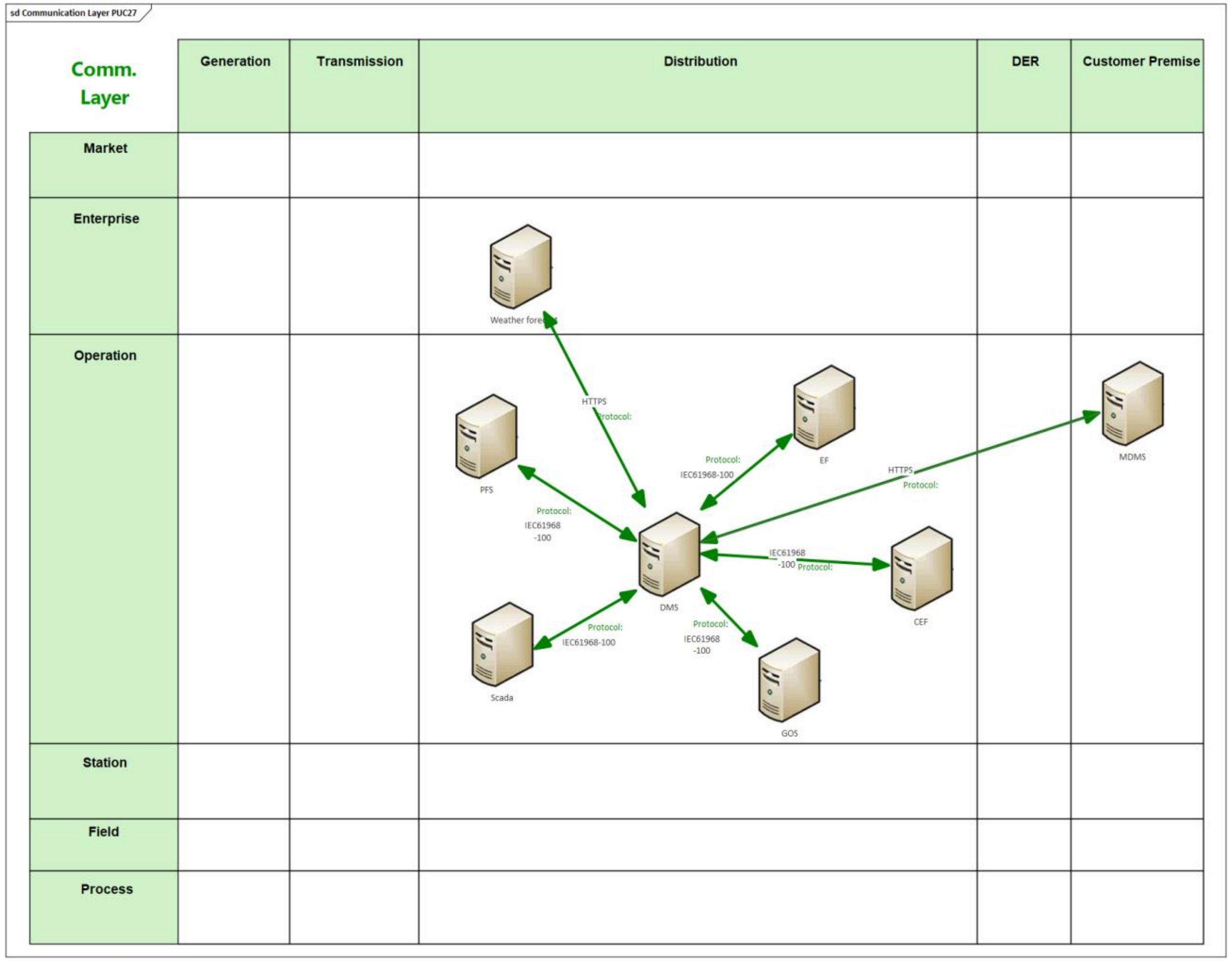
Interfaces

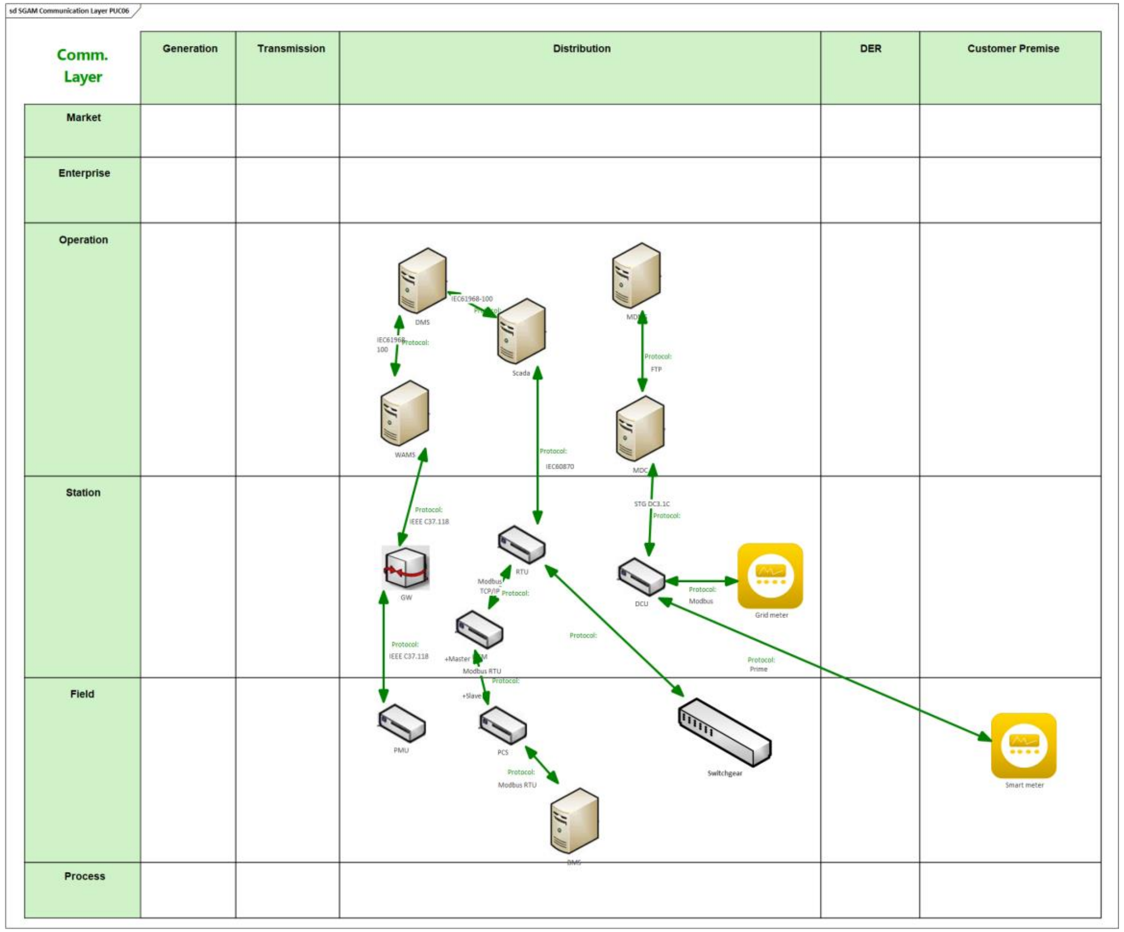




THE COLLECTIVE
the systems integration company

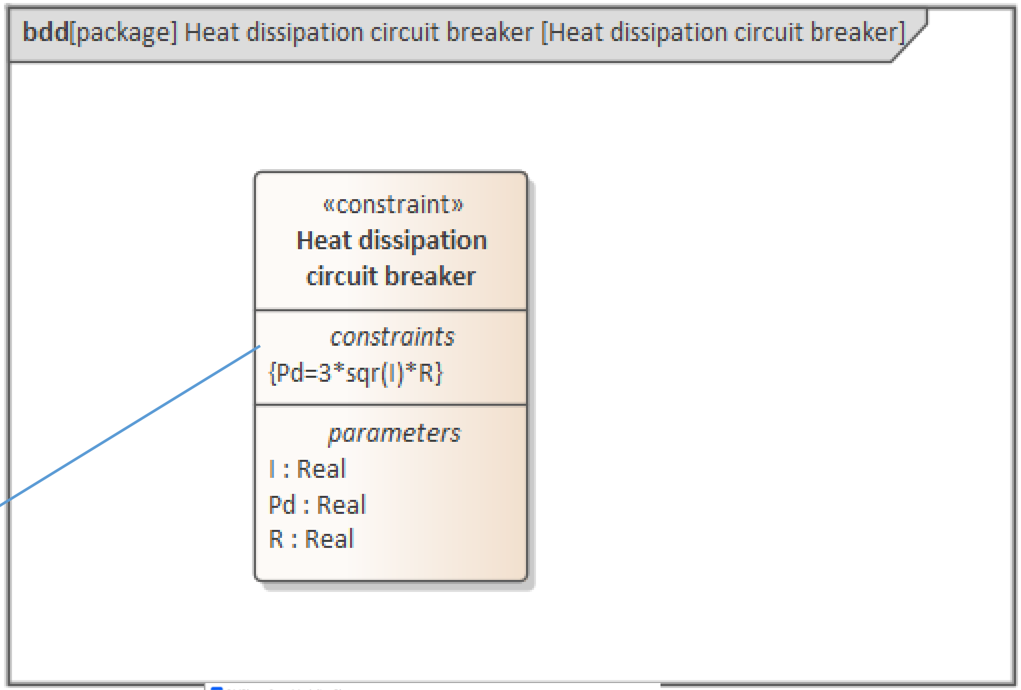
Communication

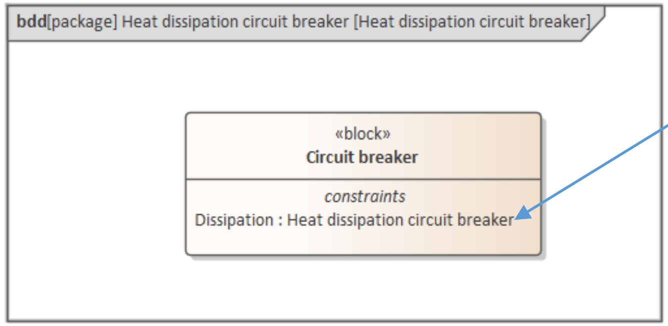


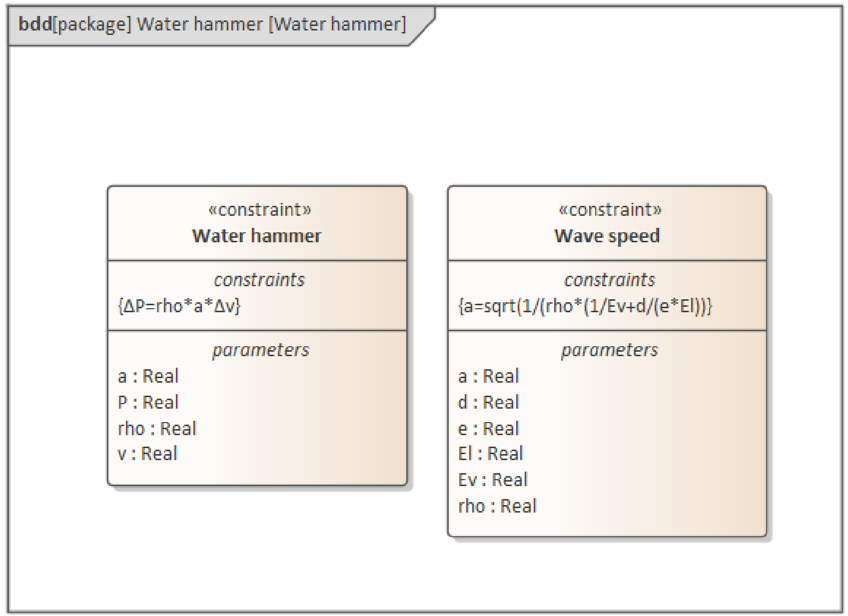


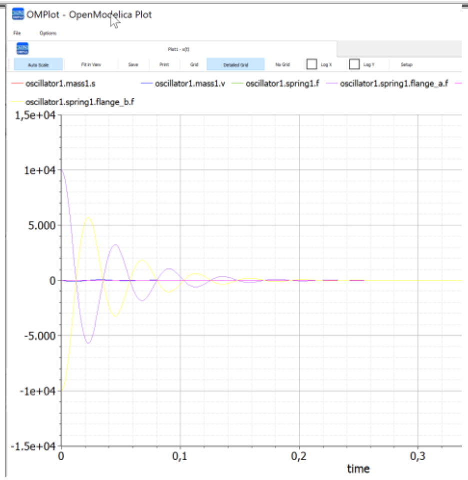
Parametrics

- Properties:
  - Porosity
  - Durability
  - Density
  - Density index
  - Specific gravity
  - Fire resistance
  - Frost resistance
  - Weathering resistance
  - Spalling resistance
  - Water absorption
  - Water permeability
  - Hygroscopicity
  - Coefficient of softening
  - Refractorines
  - Heat dissipation
  - Stability

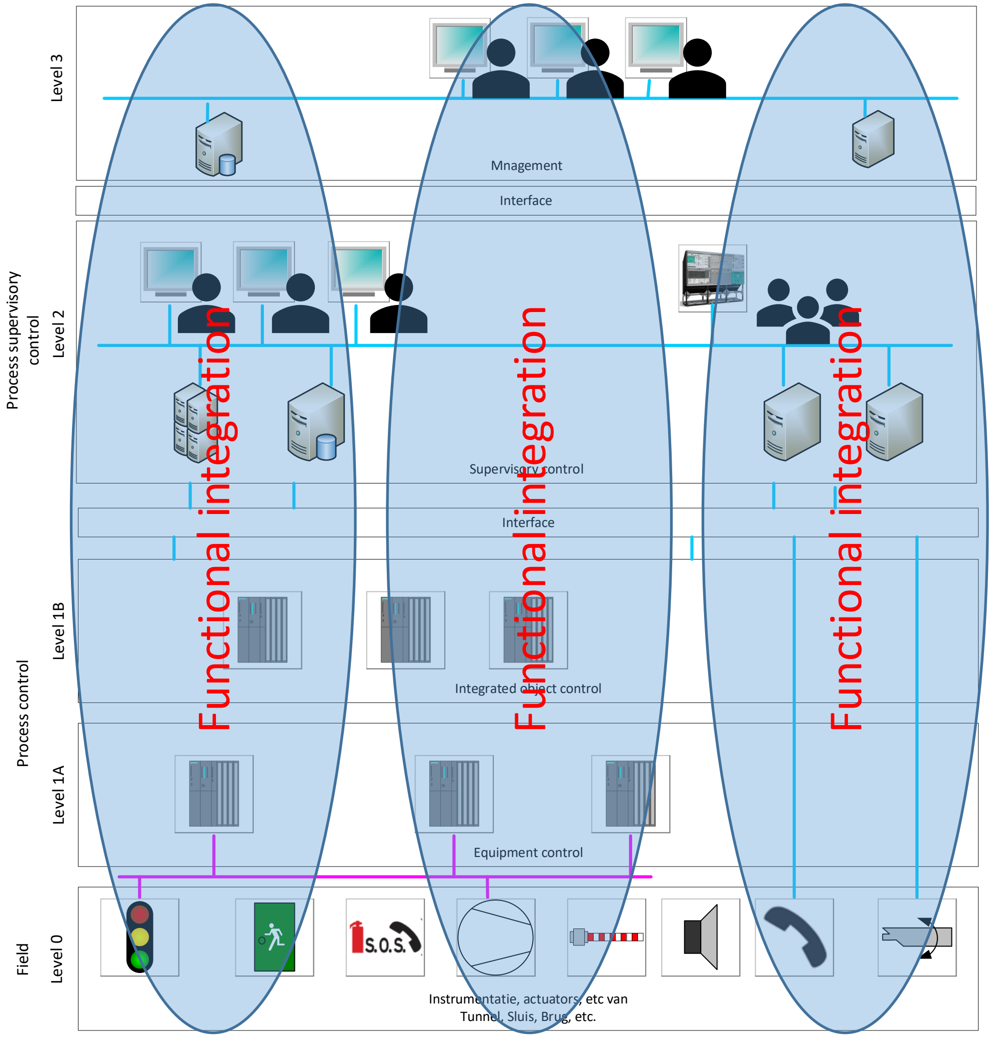


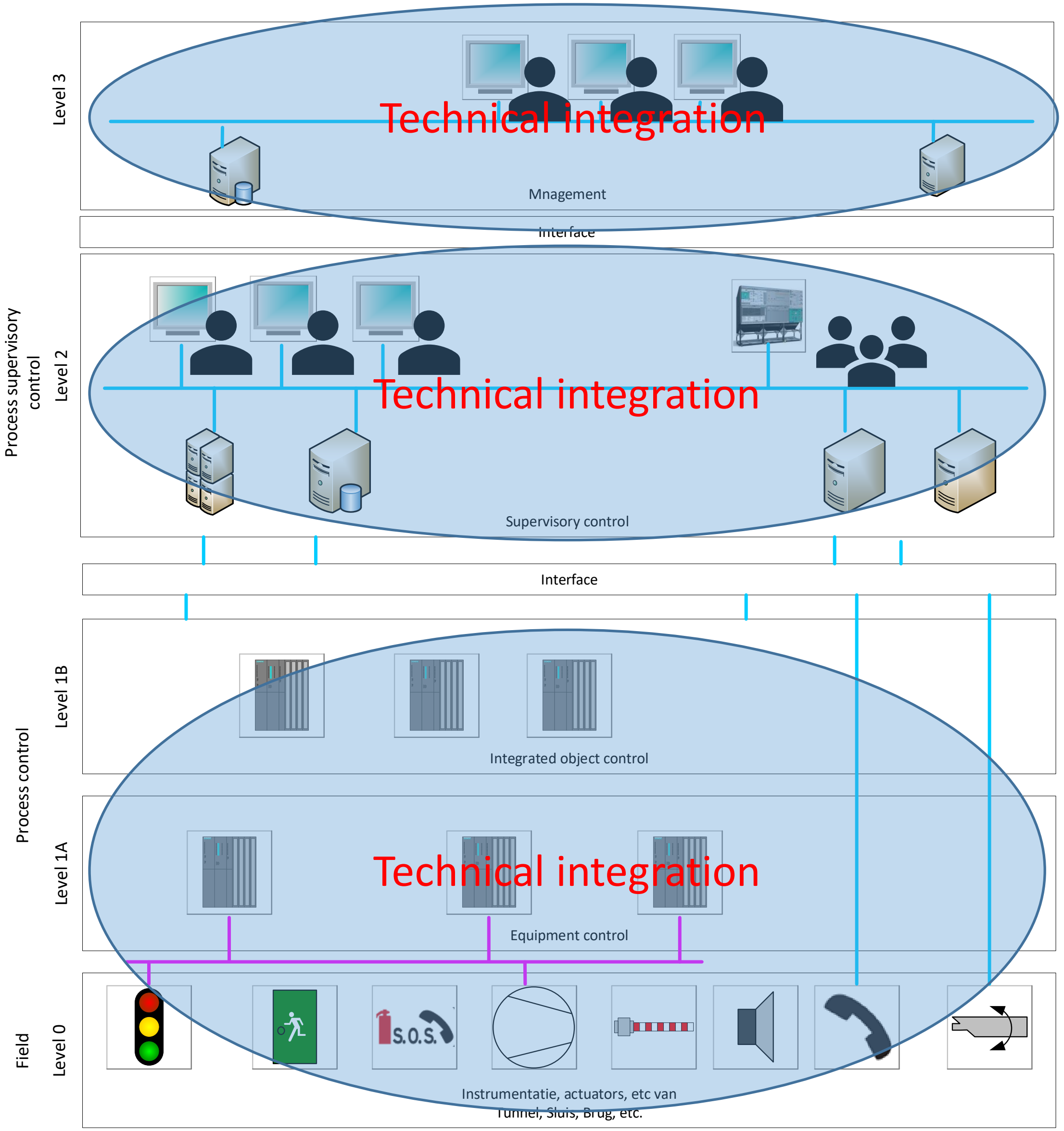






Integration

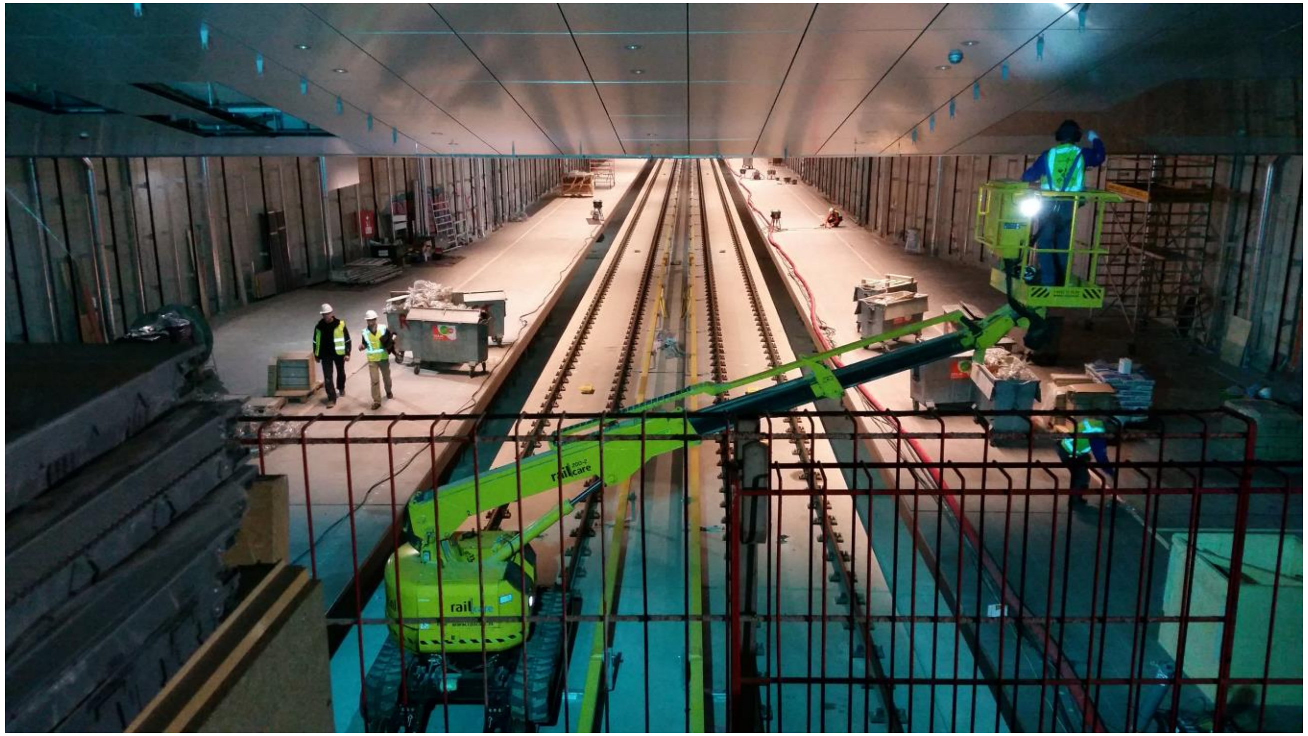




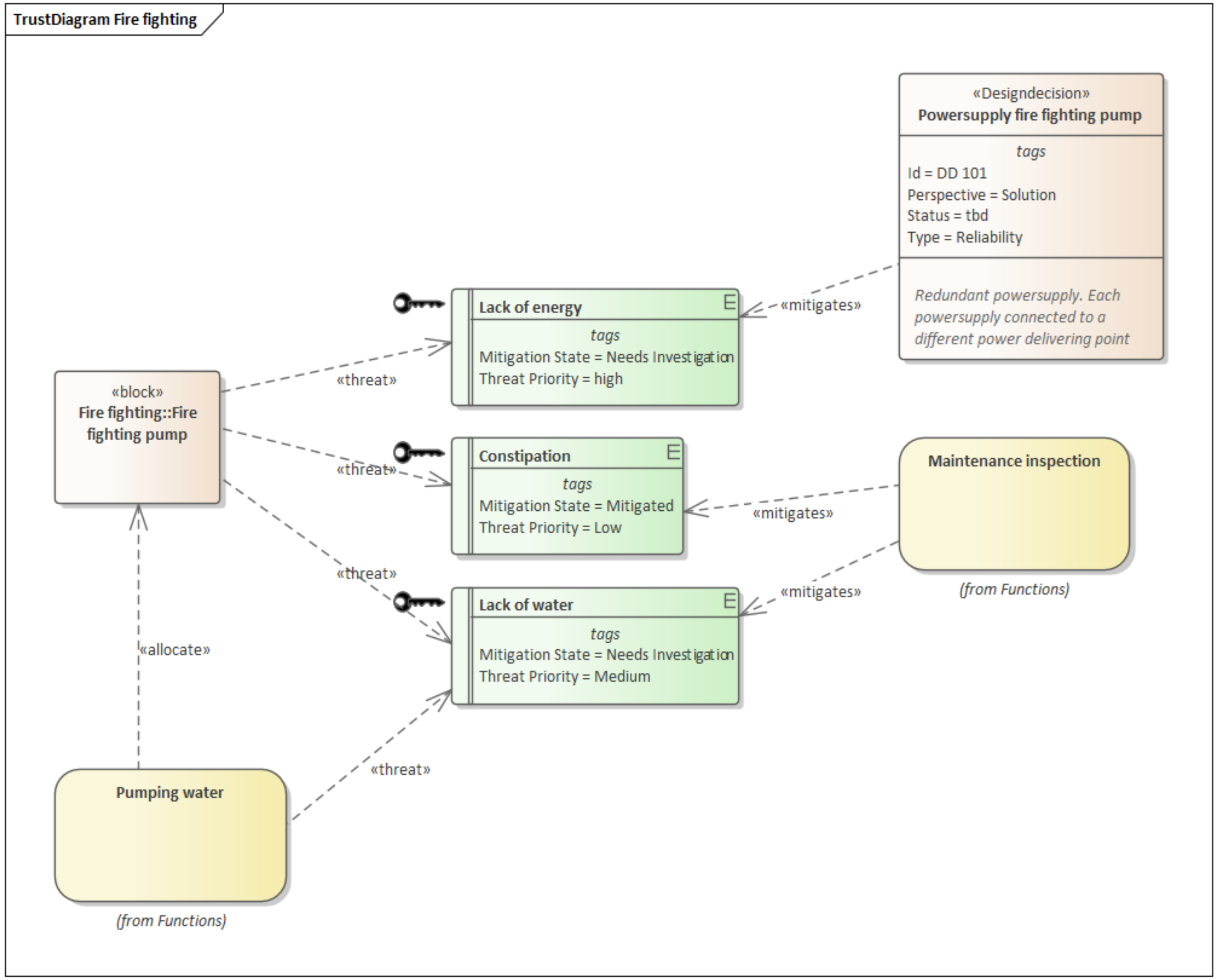
[TC] THE COLLECTIVE
the systems integration company

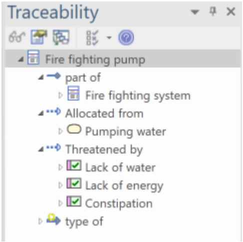
Constraints from

- Implementation
- Testing
- Transition
- Maintenance
- Logistic
- etc



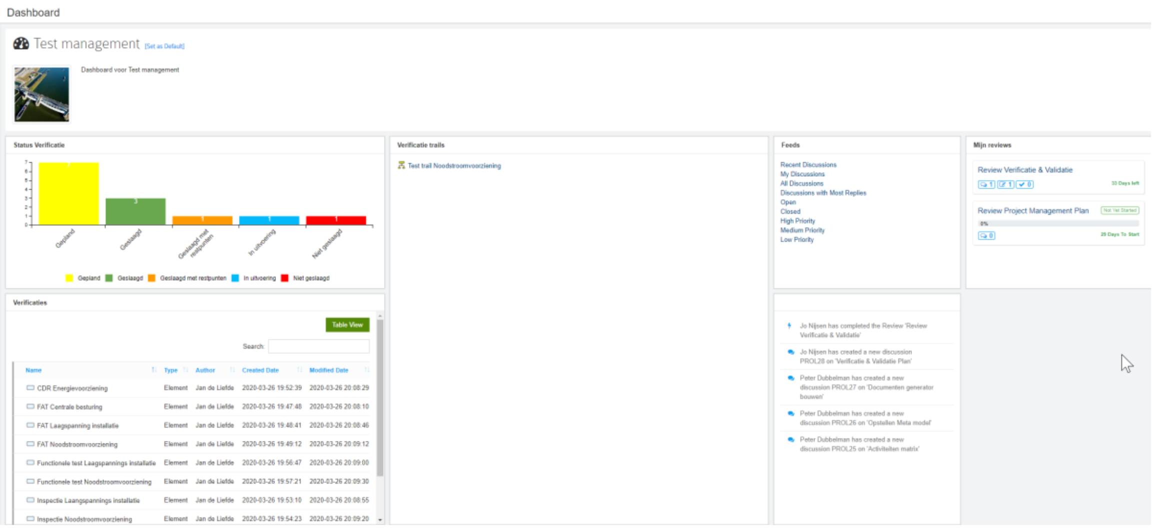
Failure Mode & Effect Analysis (FMEA)

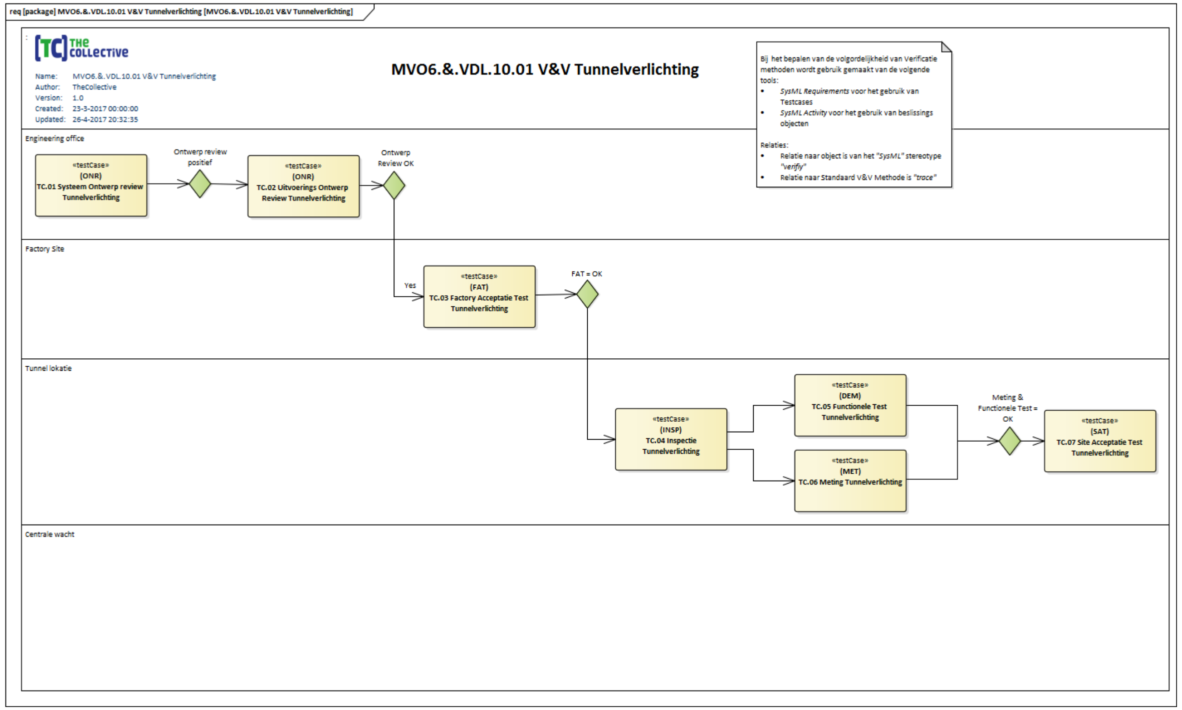


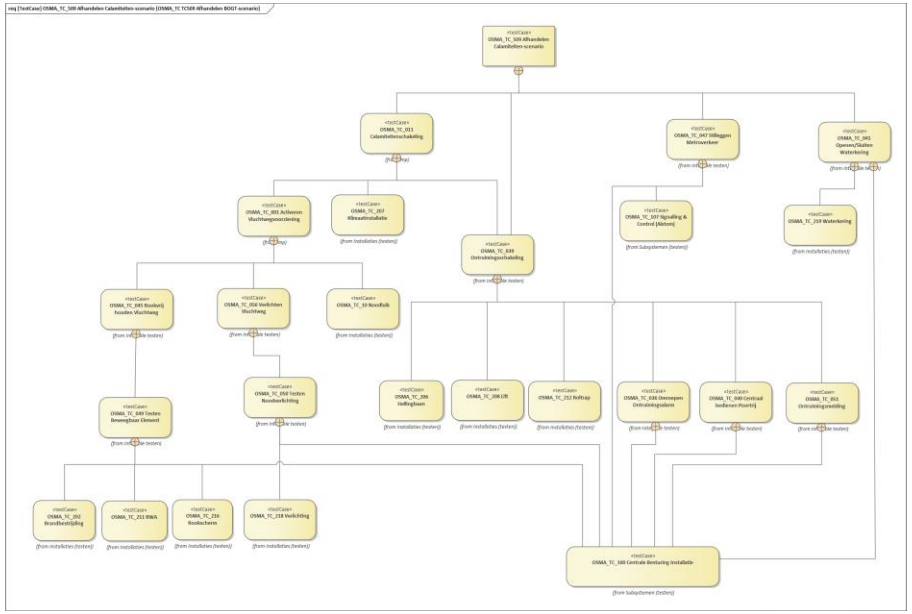


Verification

- Review functional model
- Define Testcases and Testcriteria

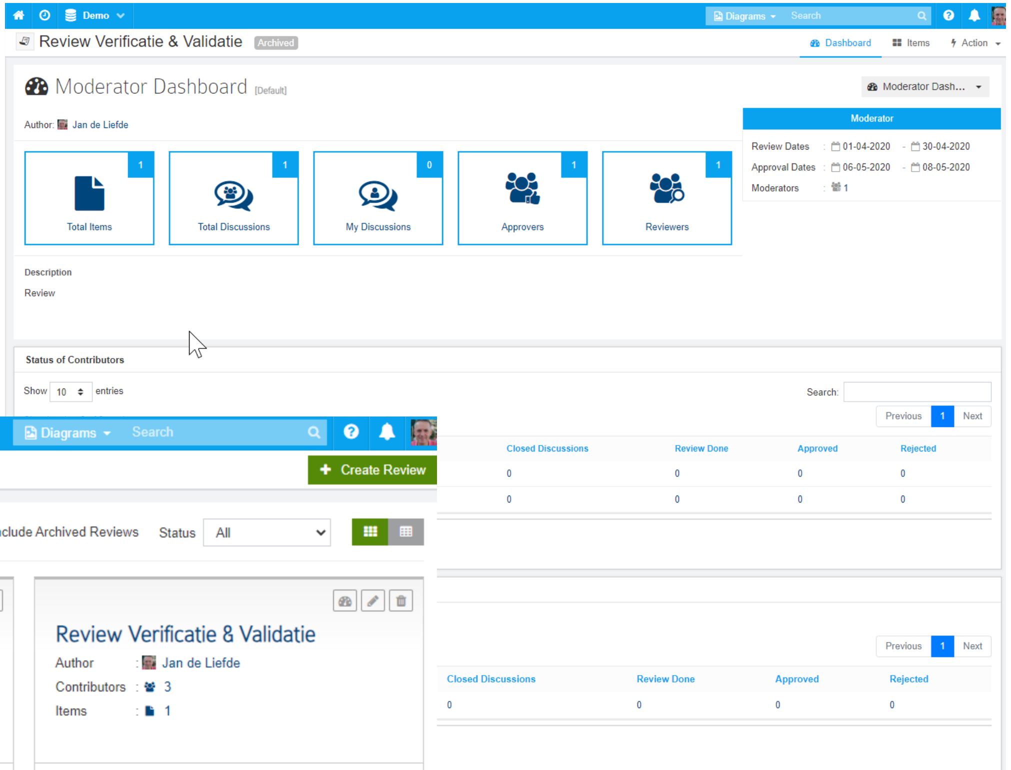


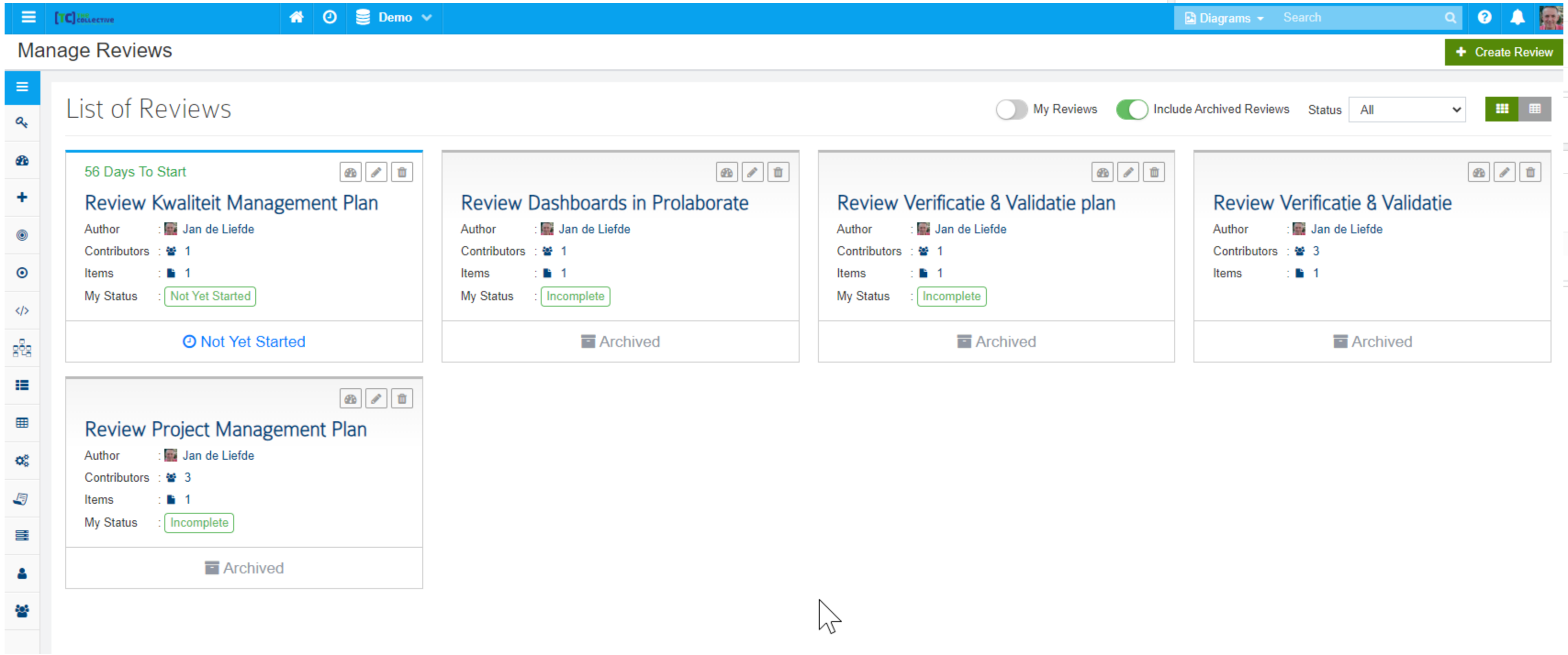




Review

- Review Operational Model with the Stakeholders





[TC] THE COLLECTIVE
the systems integration company

Review checklists

bdd[package] TEC.04 Architecture Definition V&V [TEC.04 Architecture Definition V&V]

CL01.TEC.04 Architecture Evaluation

- [ ] *Correctness:*
- [ ] *1) Verify the characteristics, attributes, constraints, and functional and performance requirements of the selected architecture(s) correctly implement the system requirements.*
- [ ] *2) Verify the selected architecture(s) complies with standards, regulations, policies, physical laws, and business rules.*
- [ ] *3) Validate the product solution(s) defined by the system architecture(s) satisfies the stakeholder needs.*
- [ ] *4) Validate that the selected architecture(s) and its element interactions do not result in unnecessary, unintended, or deleterious consequences.*
- [ ] *5) Validate the selected architecture(s) defines the intended interaction of the system with its operating environment.*
- [ ] *Consistency:*
- [ ] *1) Verify the selected architecture(s) conforms to the architectural guidance, principles, and tenets of the organization processes and procedures (e.g., serviceoriented architecture, modular open-systems architecture).*
- [ ] *2) Verify the architectural principles, characteristics, and rules established for the system are being applied across the selected architecture(s).*
- [ ] *Completeness:*
- [ ] *1) Verify the system functions are allocated to the elements of the selected architecture(s).*
- [ ] *2) Verify all system requirements are included in the selected architecture(s).*

CL02.TEC.04 Interface Analysis

- [ ] *Correctness:*
- [ ] *1) Verify the system architecture satisfies the system interface requirements between system elements.*
- [ ] *2) Validate the system architecture satisfies the system interface requirements with external systems (external system boundaries).*
- [ ] *Completeness:*
- [ ] *1) Verify the architecture describes all internal and external interfaces.*

CL01.TEC.03 Requirements Evaluation

- [ ] *Correctness:*
- [ ] *Verify and validate that the required characteristics, attributes, constraints (e.g., mechanical, electrical, mass, thermal, data, procedural flows), and functional and performance requirements for a product solution are correct (e.g., security, ergonomics, human-machine interface, safety, reliability, maintainability, response time).*
- [ ] *Verify and validate that the system requirements satisfy the stakeholder requirements.*
- [ ] *Verify that the system requirements comply with standards, references, regulations, policies, physical laws, and business rules.*
- [ ] *Validate that the system requirements define the intended interaction of the system with its operating environment and other interfacing systems.*
- [ ] *Consistency:*
- [ ] *Verify that all terms, concepts, and requirements are documented consistently in accordance with accepted syntax and structure (e.g., style guides and requirements modeling structure).*
- [ ] *Verify that there is consistency among the requirements, groups of requirements (functional interaction), and assumptions.*
- [ ] *Completeness:*
- [ ] *Validate that all stakeholder needs are satisfied by the set of system requirements.*
- [ ] *Validate that performance criteria and functionality are described in the requirements, within the assumptions and constraints of the operating environment and system boundaries.*
- [ ] *Verify that the System requirements satisfy specified configuration management procedures. Verify that system requirements are in a form suitable for requirements management throughout the life cycle.*
- [ ] *Readability:*
- [ ] *Verify that the documentation is legible, understandable, and unambiguous to the intended audience.*
- [ ] *Verify that the documentation defines all acronyms, mnemonics, abbreviations, terms, and symbols.*
- [ ] *Testability:*
- [ ] *Verify that objective acceptance criteria can be developed to validate the requirements.*

construct Standaards

ISO 29148 Requirement engineering

Systems and Software engineering - Life cycle processes - Requirement engineering

ISO 42010 Architecture description

Software, systems and enterprise — Architecture description

TMAP

EIA-IEEE J-Std 016-1995 Software Development Acquirer-Supplier Agreement free .pdf

1012-2016 IEEE Standard for System, Software, and Hardware Verification and Validation

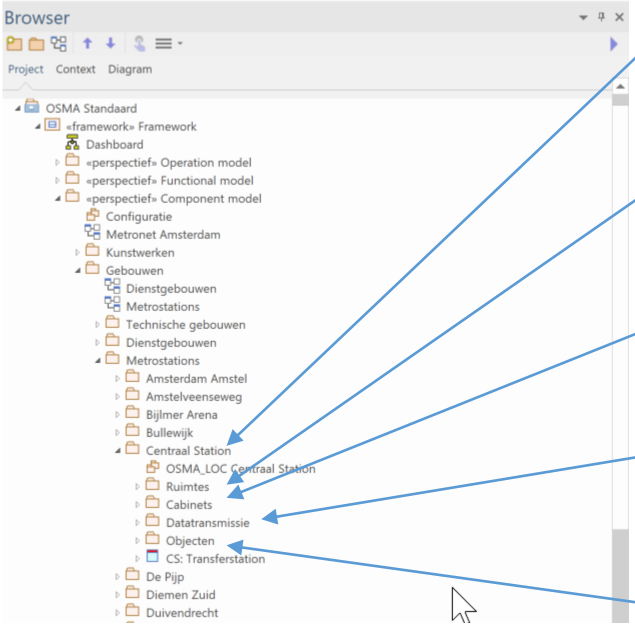
[TC] THE COLLECTIVE
the systems integration company

Functional model

- Functions defined
- System structure defined
- Interfaces including information defined
- Communication protocols defined
- Testcases defined
- Hazards identified and mitigated
- Constraints identified and implemented
- Parametrics defined and verified
- Design decisions documented
- Traceability guaranteed
  - Within the System model
  - Other sources (drawings, simulations results, etc.)

Component model: Process

- Define instances of Systemelements
- Define relations between instances



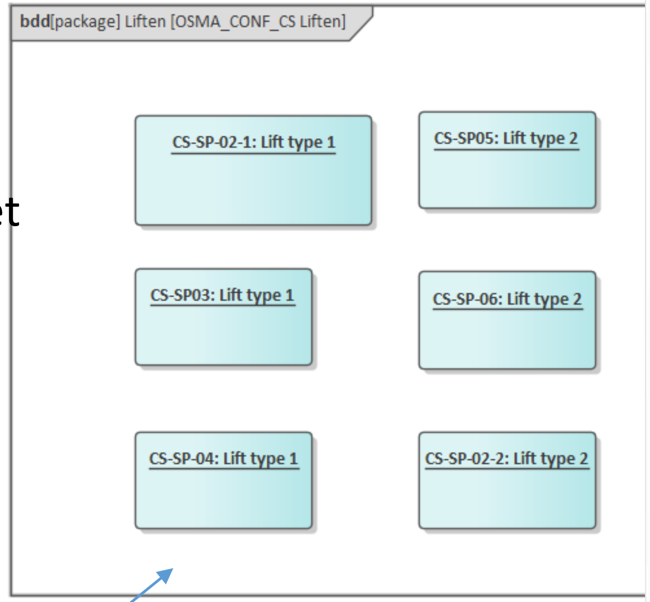
Civil asset

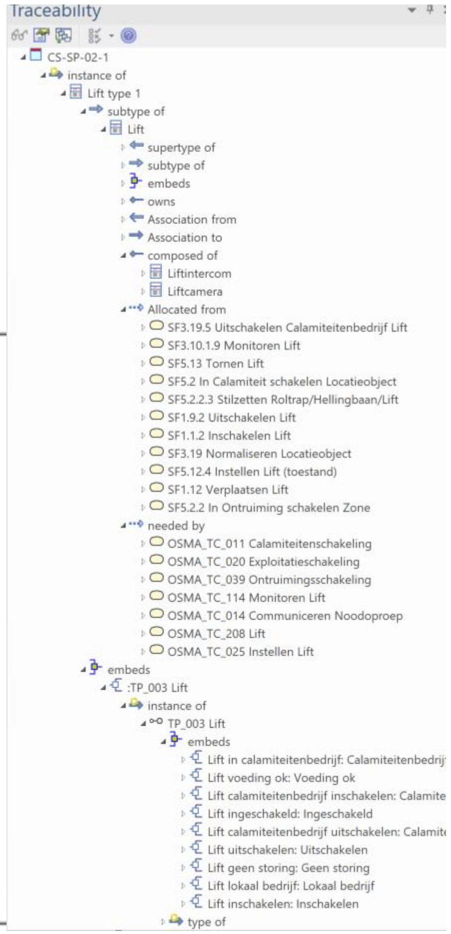
Locations within the asset

Cabinets

Network topologie

Installations (ventilation, elevators, etc.)





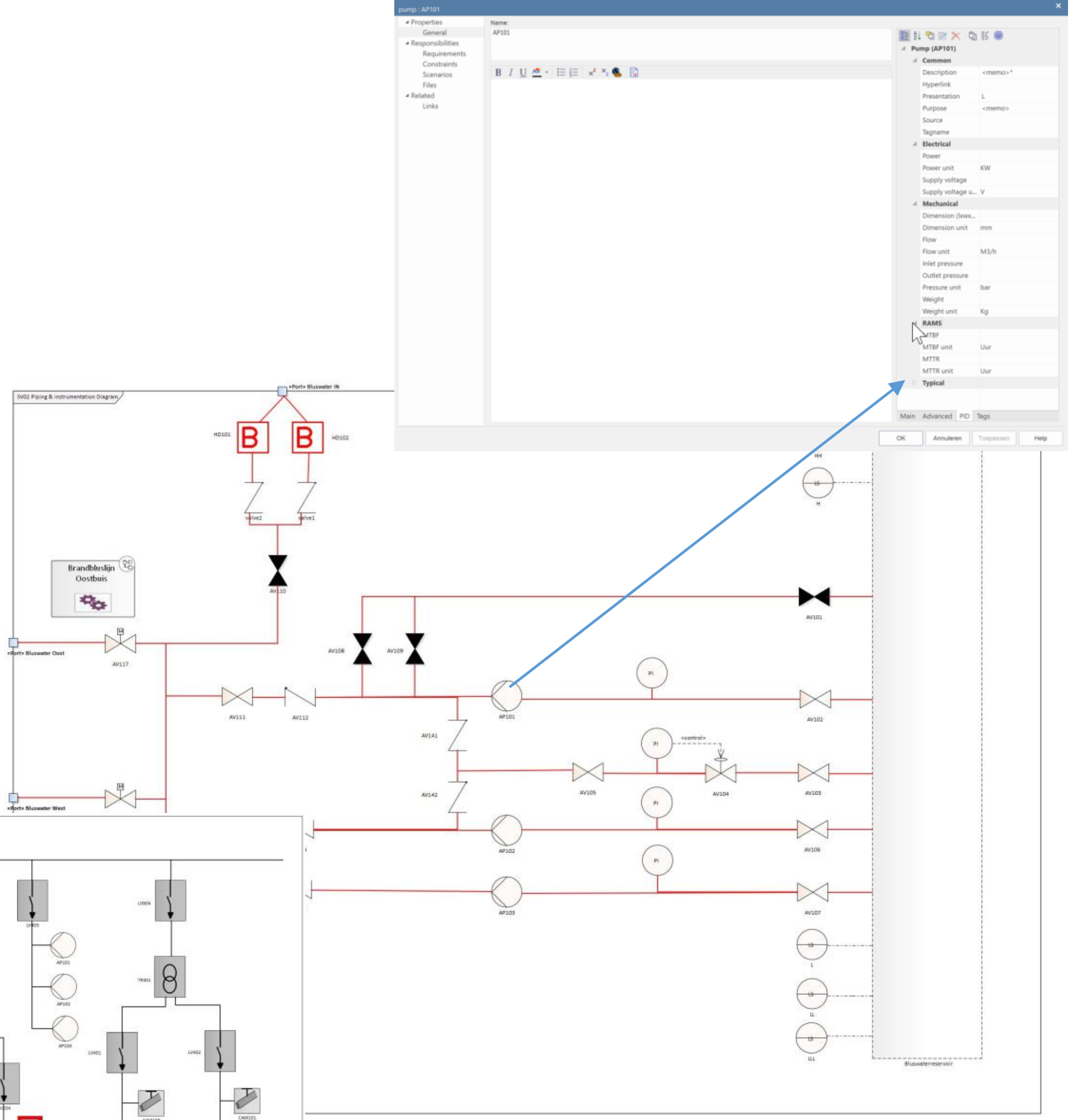
Functions

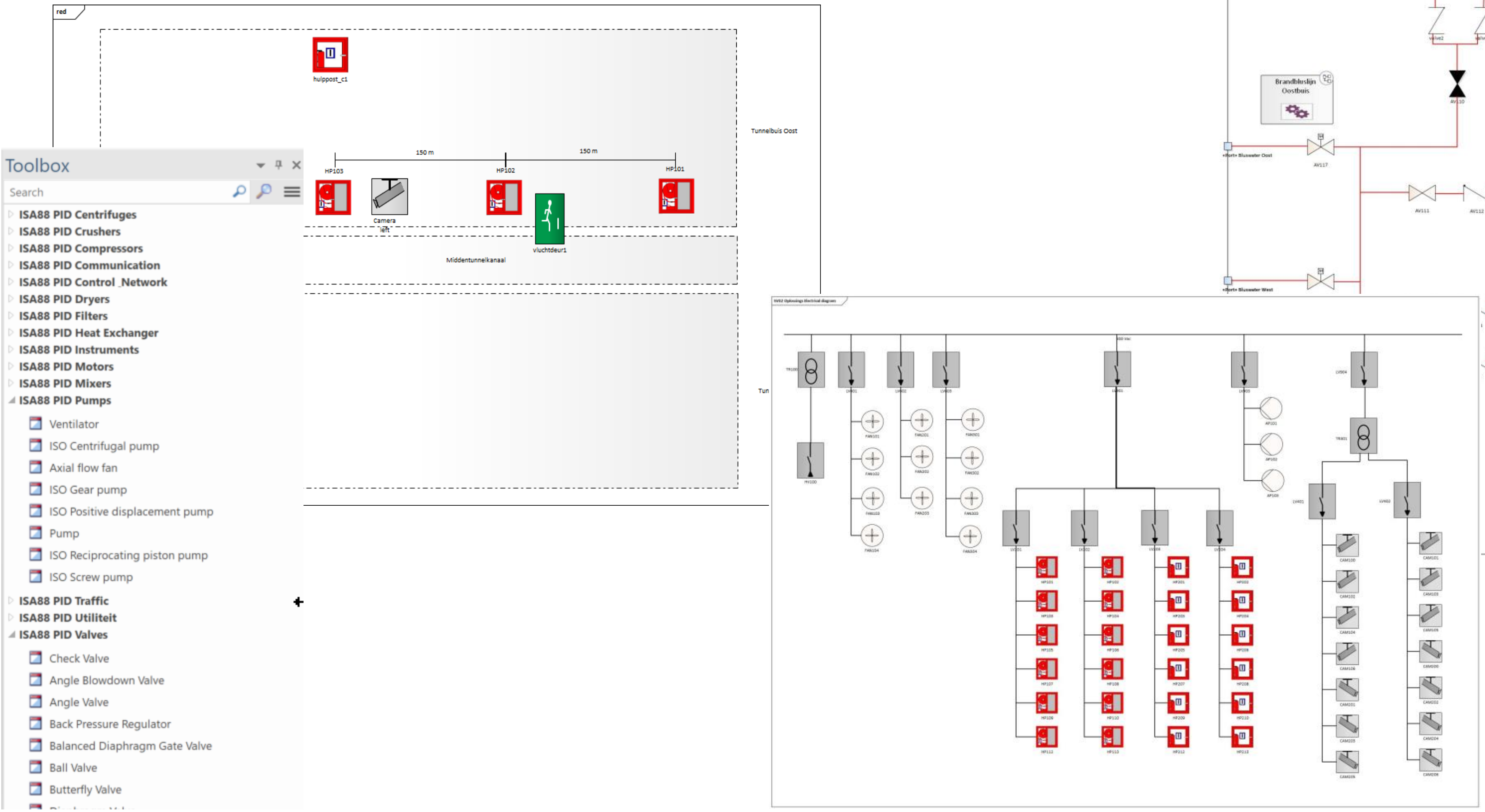
Testcases

Signals

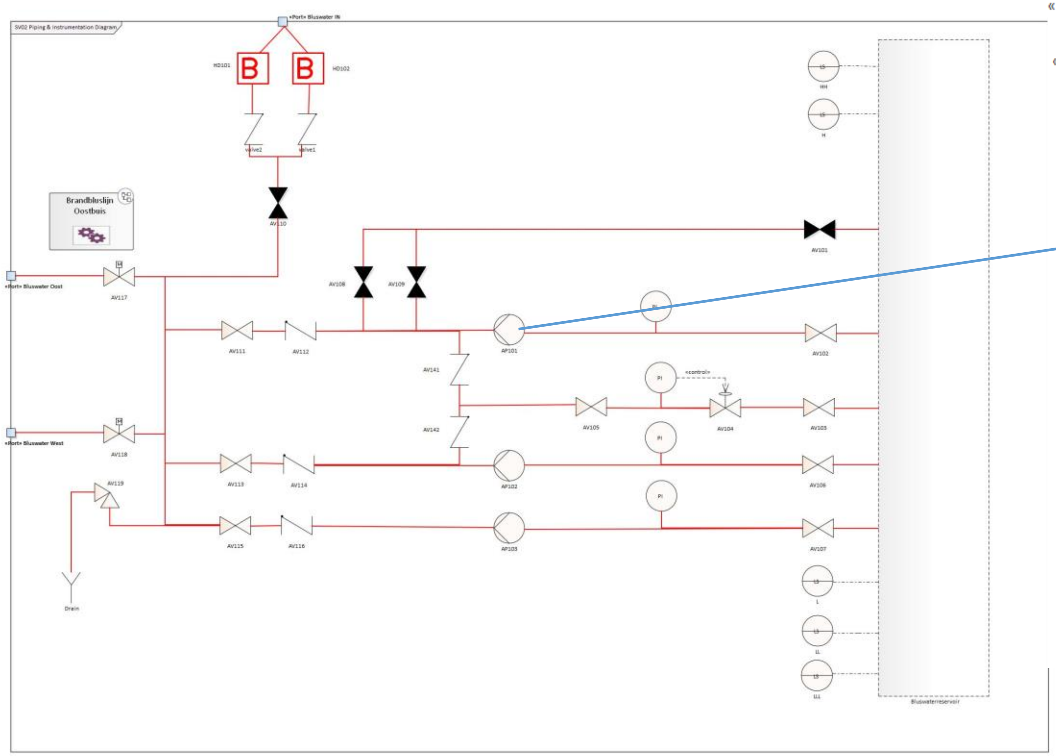
THE COLLECTIVE
the systems integration company

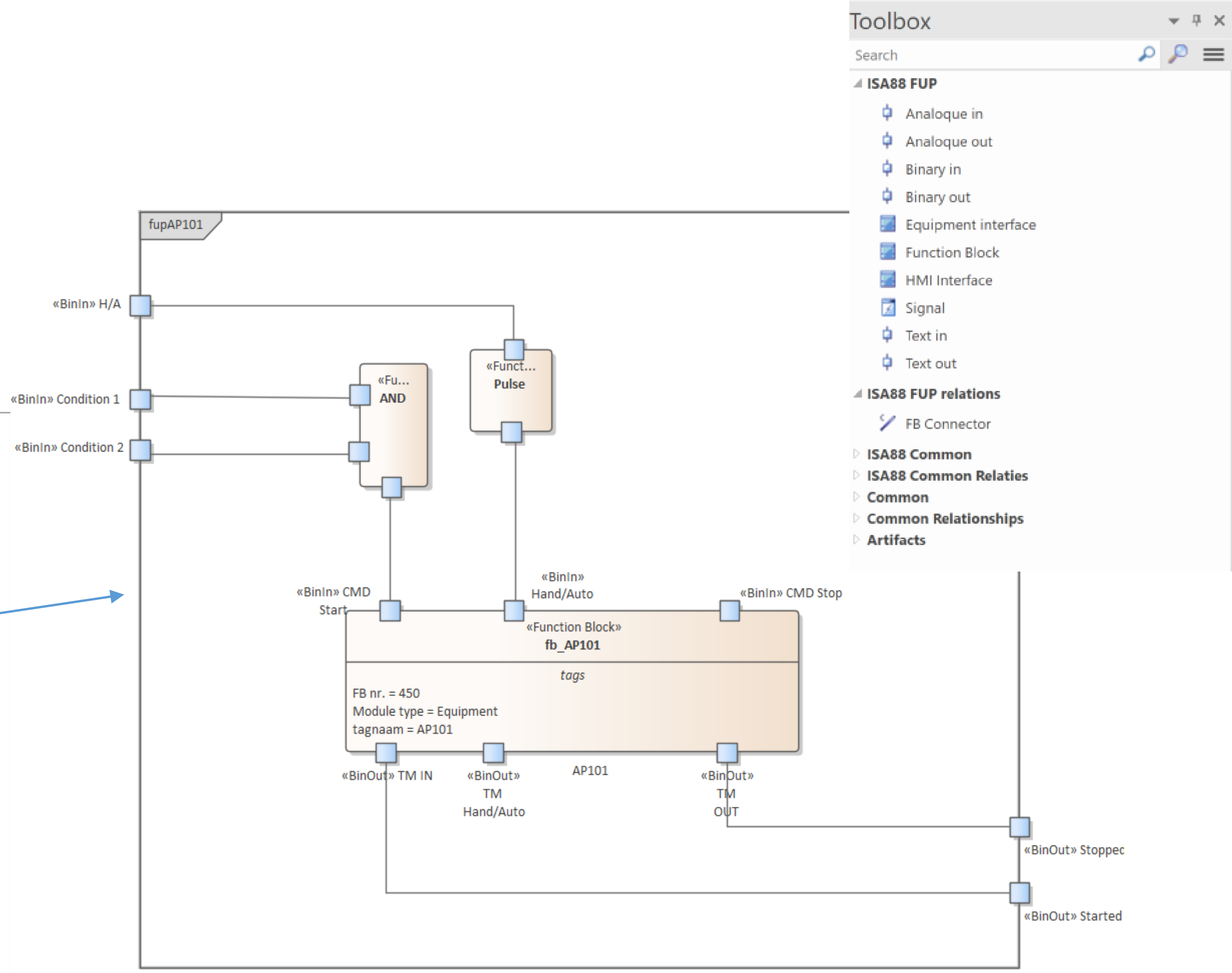
Readable diagrams





Readable diagrams





[TC] THE COLLECTIVE
the systems integration company

Component model

- Instances are defined as part of the model
  - All Locations (buildings, rooms, etc.)
  - All Equipment (Pumps, ventilators, etc.)
  - All Automation (sw blocks, interlocks, inputs/outputs, etc.)
  - All Relations between locations, equipment, sw blocks, test-/inspections
  - All Cable & Pipes
  - All Civil parts
- Cable and piping routing is defined
- All Integrations plans
- All Test-/Inspection plans & reports
- Traceability guaranteed
  - Within the model
  - To other data sources (CAD drawings, 3D visualisations, etc)

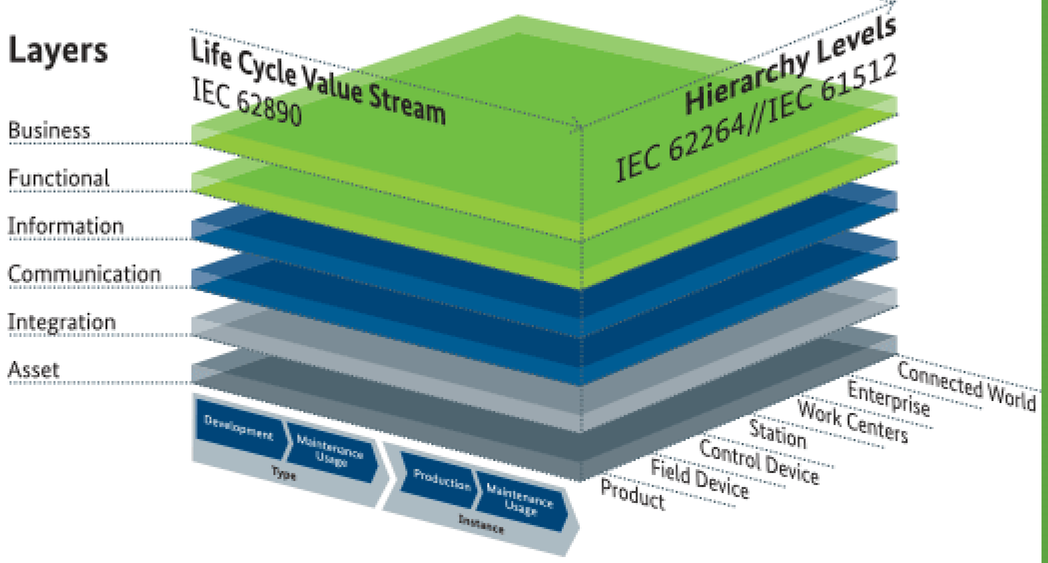
[TC] THE COLLECTIVE
the systems integration company

Implementation design

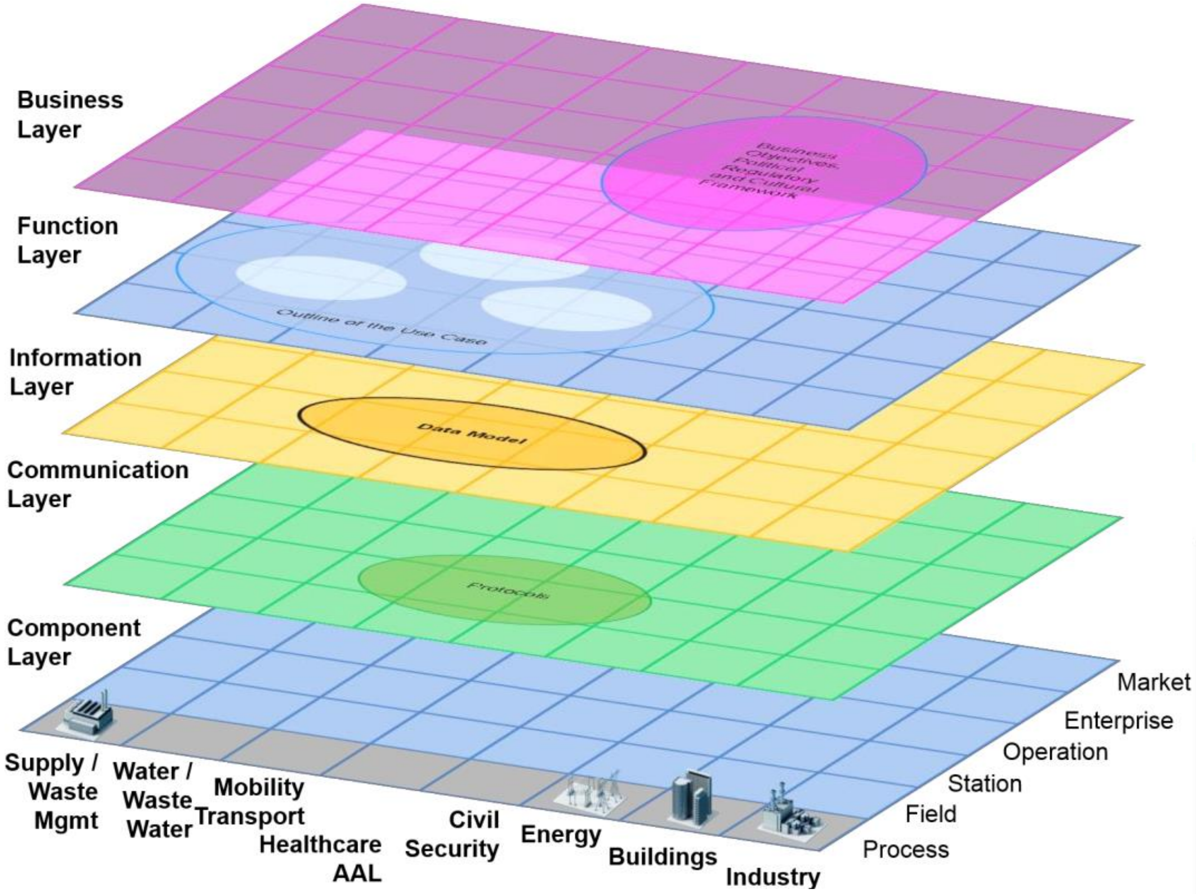
- Detail civil drawings
- Cabinet drawings
- Cable routing
- Piping routing
- Software design
- etc

Guarantee the traceability between the implementation design en the componentmodel

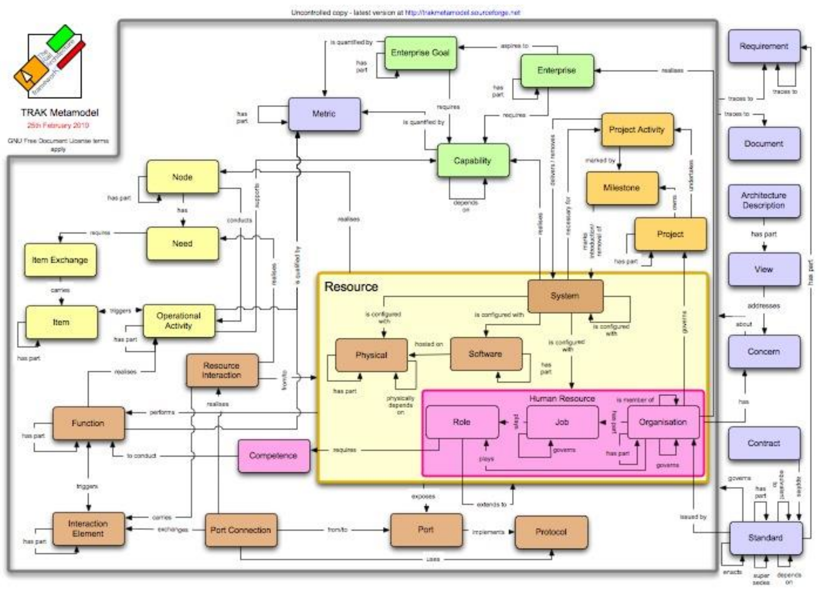
Frameworks



RAMI framework



SGAM framework



TRAK framework

[TC] THE COLLECTIVE
the systems integration company