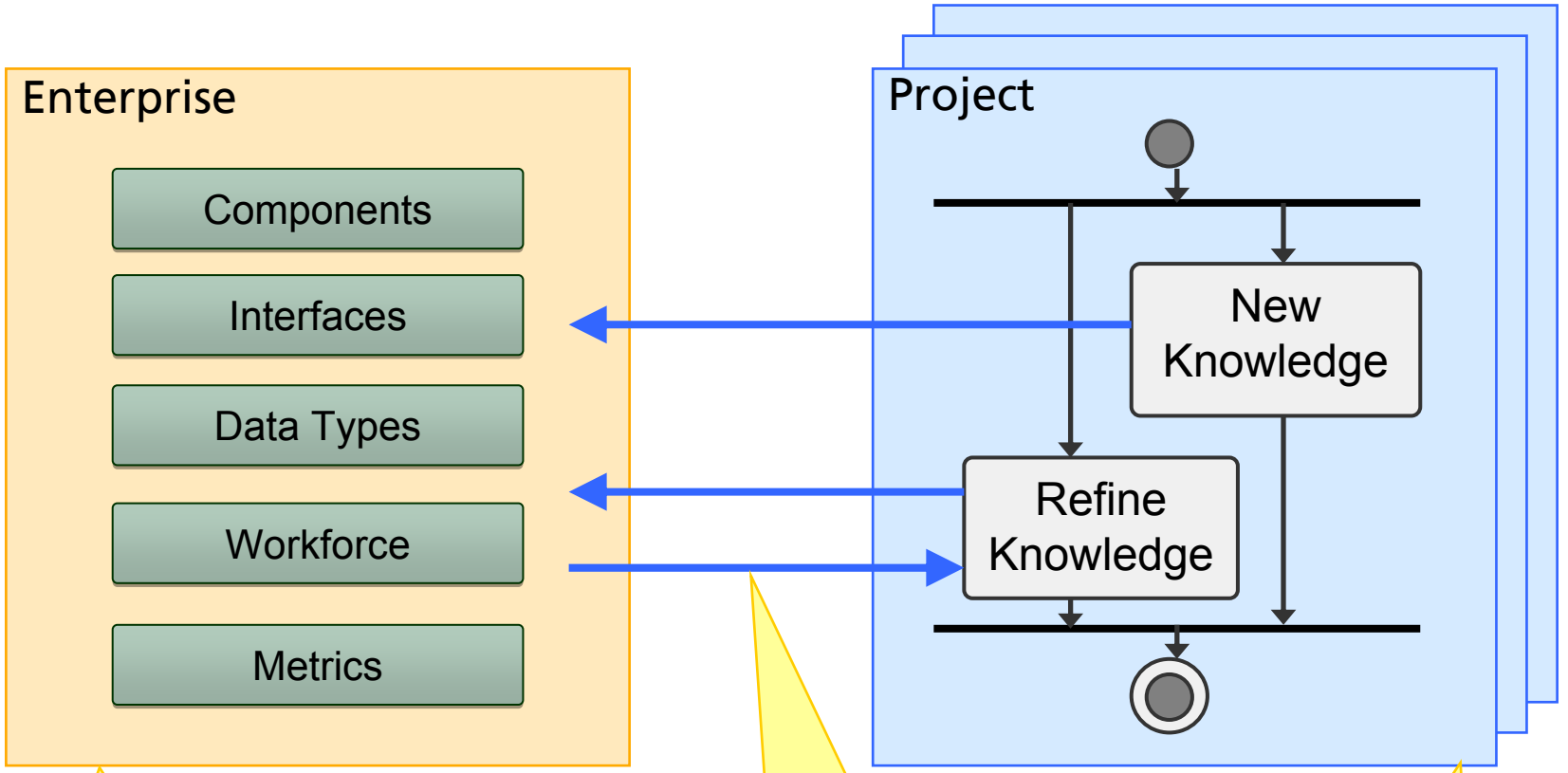
Interface Management in a Large Enterprise

Eclipse Finance Day 2012

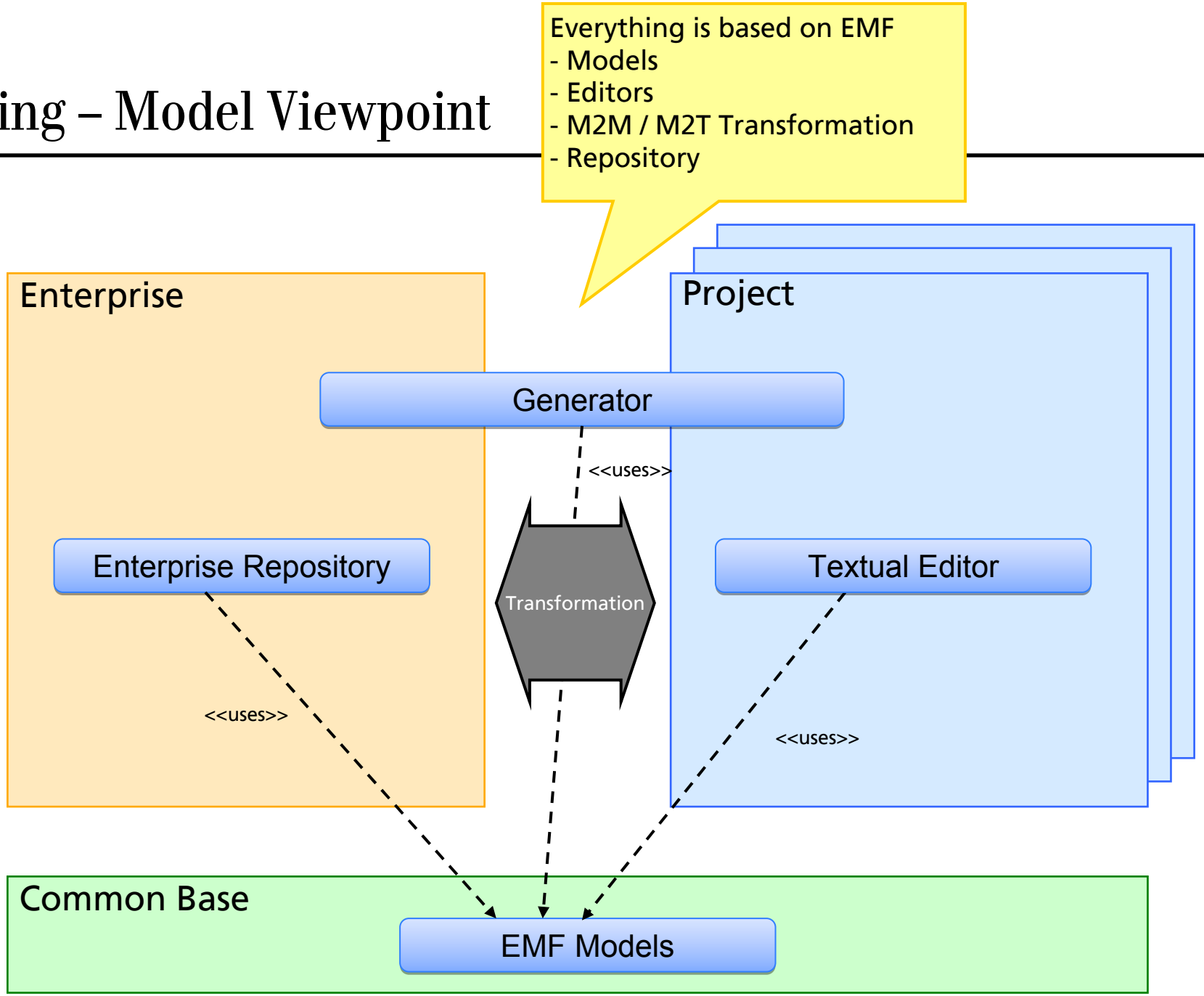
Robert Blust

October 16, 2012

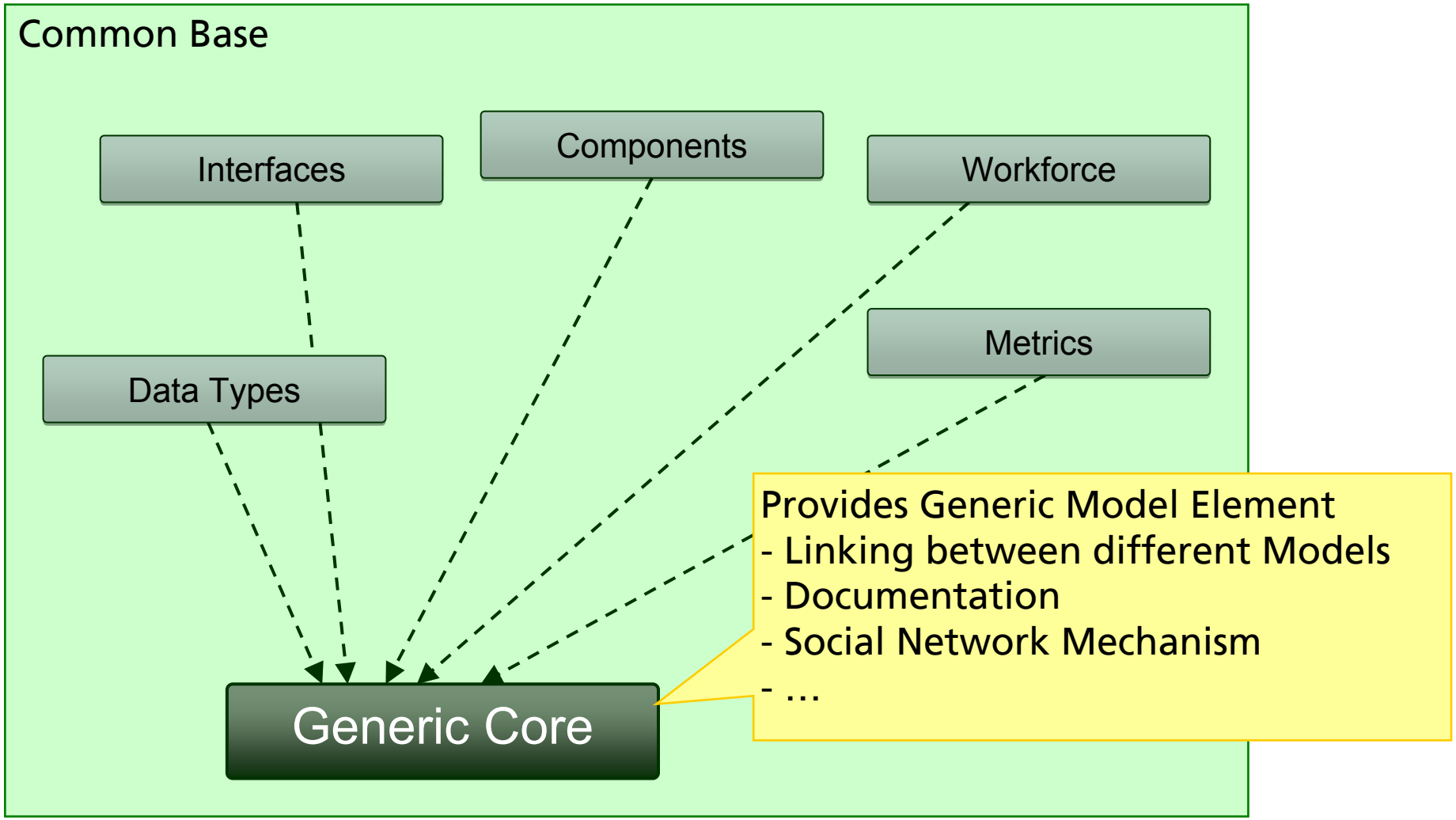
Tooling – Context



Tooling – Model Viewpoint



Tooling – Common Base



Technology Requirements

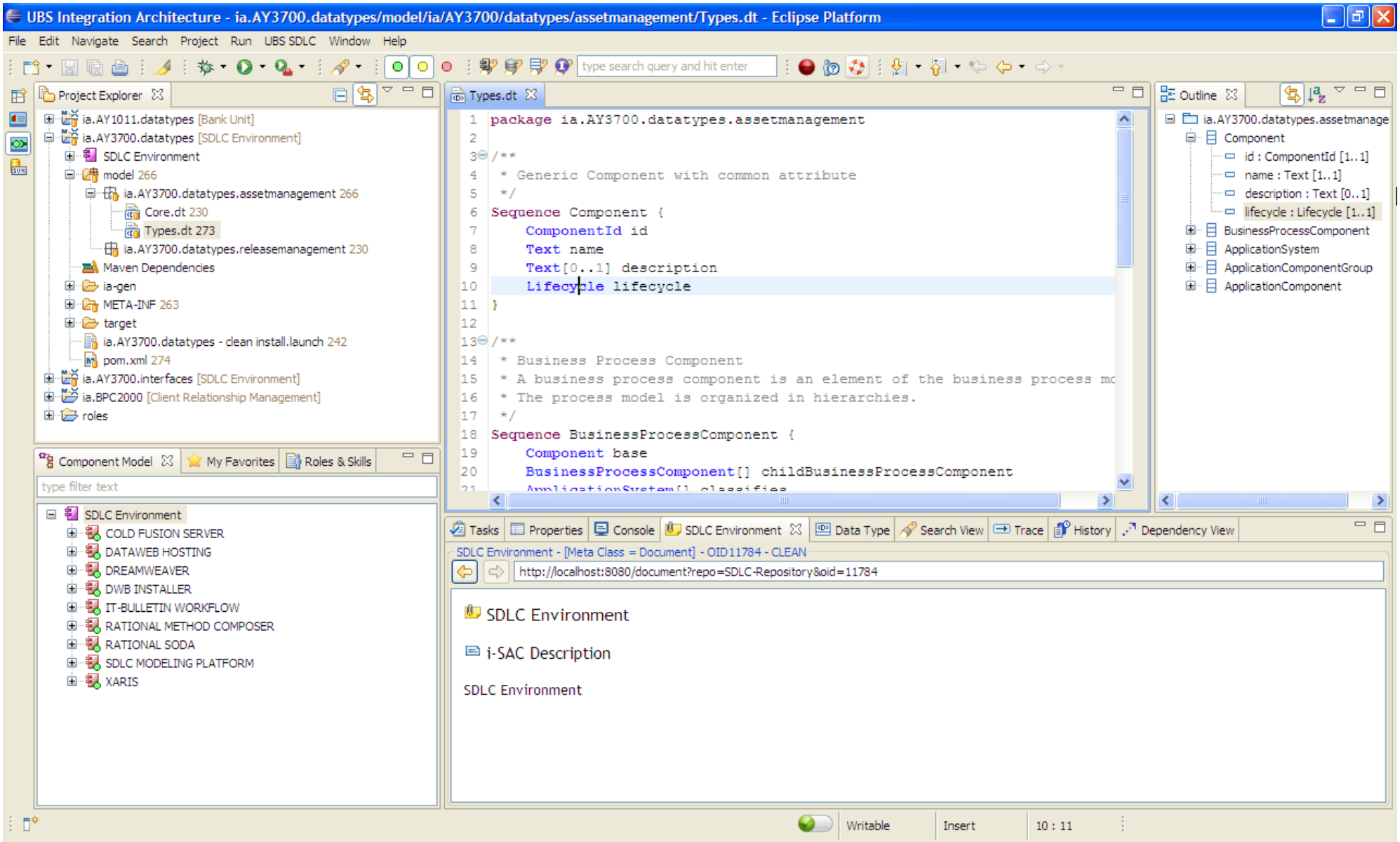
- **Structured Data Models**
  - Describe
  - Create
  - Persist

- **Orthogonal Functionalities 'Enterprise Level'**
  - Repository Persistence
  - Scalability
  - Temporality
  - Collaboration
  - Model Fragment loading
  - Browsing
  - Referential Integrity

Technology Requirements

- **Orthogonal Functionalities 'Project Level'**
  - Integration into IDE
  - Integration into Source Control Management
  - Integration into Central Build Infrastructure
    - Maven
    - Nexus
    - Jenkins
  - Usability
    - Early Validation
    - Guidance
  - Convention over Configuration

Demo Time


UBS Integration Architecture - ia.AY3700.datatypes/model/ia/AY3700/datatypes/assetmanagement/Types.dt - Eclipse Platform
File Edit Navigate Search Project Run UBS SDLC Window Help
Project Explorer
ia.AY1011.datatypes [Bank Unit]
ia.AY3700.datatypes [SDLC Environment]
SDLC Environment
model 266
ia.AY3700.datatypes.assetmanagement 266
Core.dt 230
Types.dt 273
ia.AY3700.datatypes.releasemanagement 230
Maven Dependencies
ia-gen
META-INF 263
target
ia.AY3700.datatypes - clean install.launch 242
pom.xml 274
ia.AY3700.interfaces [SDLC Environment]
ia.BPC2000 [Client Relationship Management]
roles
Types.dt
package ia.AY3700.datatypes.assetmanagement
/**
* Generic Component with common attribute
*/
Sequence Component {
ComponentId id
Text name
Text[0..1] description
Lifecycle lifecycle
}
/**
* Business Process Component
* A business process component is an element of the business process mo
* The process model is organized in hierarchies.
*/
Sequence BusinessProcessComponent {
Component base
BusinessProcessComponent[] childBusinessProcessComponent
Outline
ia.AY3700.datatypes.assetmanage
Component
id : ComponentId [1..1]
name : Text [1..1]
description : Text [0..1]
lifecycle : Lifecycle [1..1]
BusinessProcessComponent
ApplicationSystem
ApplicationComponentGroup
ApplicationComponent
Component Model
My Favorites
Roles & Skills
type filter text
SDLC Environment
COLD FUSION SERVER
DATAWEB HOSTING
DREAMWEAVER
DWB INSTALLER
IT-BULLETIN WORKFLOW
RATIONAL METHOD COMPOSER
RATIONAL SODA
SDLC MODELING PLATFORM
XARIS
Tasks Properties Console SDLC Environment Data Type Search View Trace History Dependency View
SDLC Environment - [Meta Class = Document] - OID11784 - CLEAN
http://localhost:8080/document?repo=SDLC-Repository&oid=11784
SDLC Environment
i-SAC Description
SDLC Environment
Writable
Insert
10 : 11

Contact Information

Robert Blust
Flurstrasse 62
8048 Zürich
+41-44-236 48 59
robert.blust@ubs.com

Twitter: @robertblust
Facebook: www.facebook.com/rob.blust

Robert Blust works as an IT Architect for UBS WM&SB IT and is responsible for the strategic tool landscape supporting the software development lifecycle.
Since 2009 he leads a growing team realizing the vision of an integrated tooling platform based on the Eclipse Modeling Framework with a strong focus on model based engineering, scalability and collaboration.

www.ubs.com

Appendix

Abstract

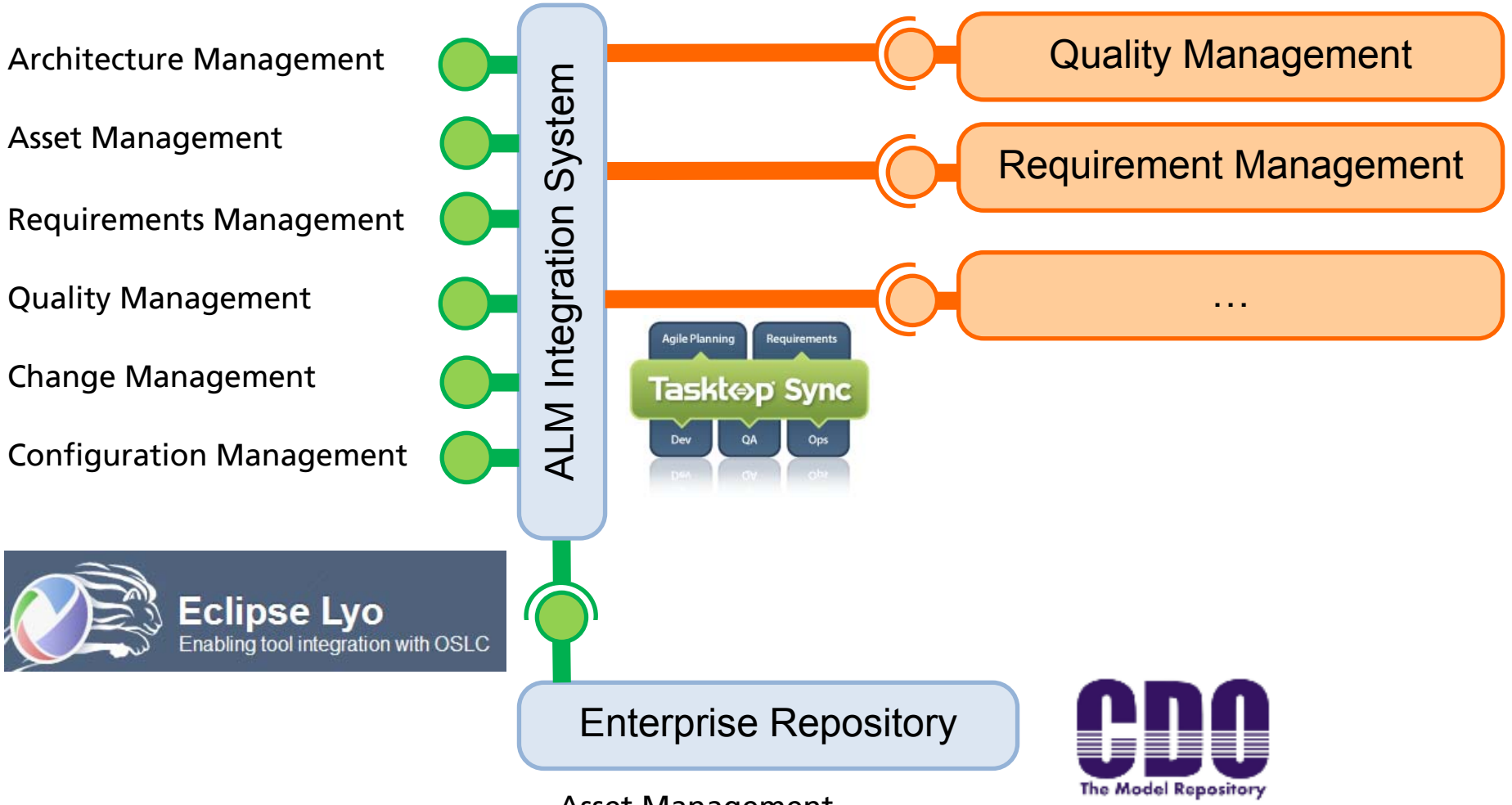
- Enterprise Architecture decomposes IT systems into subsystems and defines rules and principles for their communication. To keep the realization conformant to these rules and principles can be a challenge. To handle this challenge, UBS launched the Integration Architecture initiative. It aims to provide a usable, end-to-end process and tooling along with the necessary standards and methods to identify, specify, design, and govern interfaces between parts of the IT System.
- A first, already realized step was to visualize the current state by capturing the actual communication patterns of subsystems by analyzing their static source code and producing EMF-based graphs. This allows identifying, for instance, non conformant interaction patterns.
- The current step is to realize an appropriate tool chain for the specification, implementation and management of interfaces, which supports
  - Platform independent definition of data types and interfaces in local project scope
  - Complete specification of interfaces, their operations, policies (SLA), visibility, etc
  - Conformance to the architectural rules and principles
  - Automatic generation of artifacts like XML-Schema and WSDL files
  - A central repository for interface portfolio management (e.g. usage of interfaces)
  - Merging the interface models from the local projects back into the central repository
  - The governance process with corresponding ownership model and role based reviews

Marchitecture

Public OSLC Service Provider

Private Vendor proprietary API

ALM Overview



Scalability – Some Metrics

- UBS Component Model
  - **~ 25'000 Objects**
  - **~ 50'000 References**

- SSP Artifact Model
  - **~ 250'000 Objects**
  - **~ 1'000'000 References**

- Identity Model
  - **~ 60'000 Objects**
  - **~ 15'000 References**

Temporality – Some Metrics

- UBS Component Model
  Daily updates ~ 1% of Object changes
- SSP Artifact Model
  Monthly updates ~ 20% of Object changes

- Seamless platform integration
  - Switch between Latest and temporal states
  - Compare different states (EMF Compare)

Features UBS is Interested in

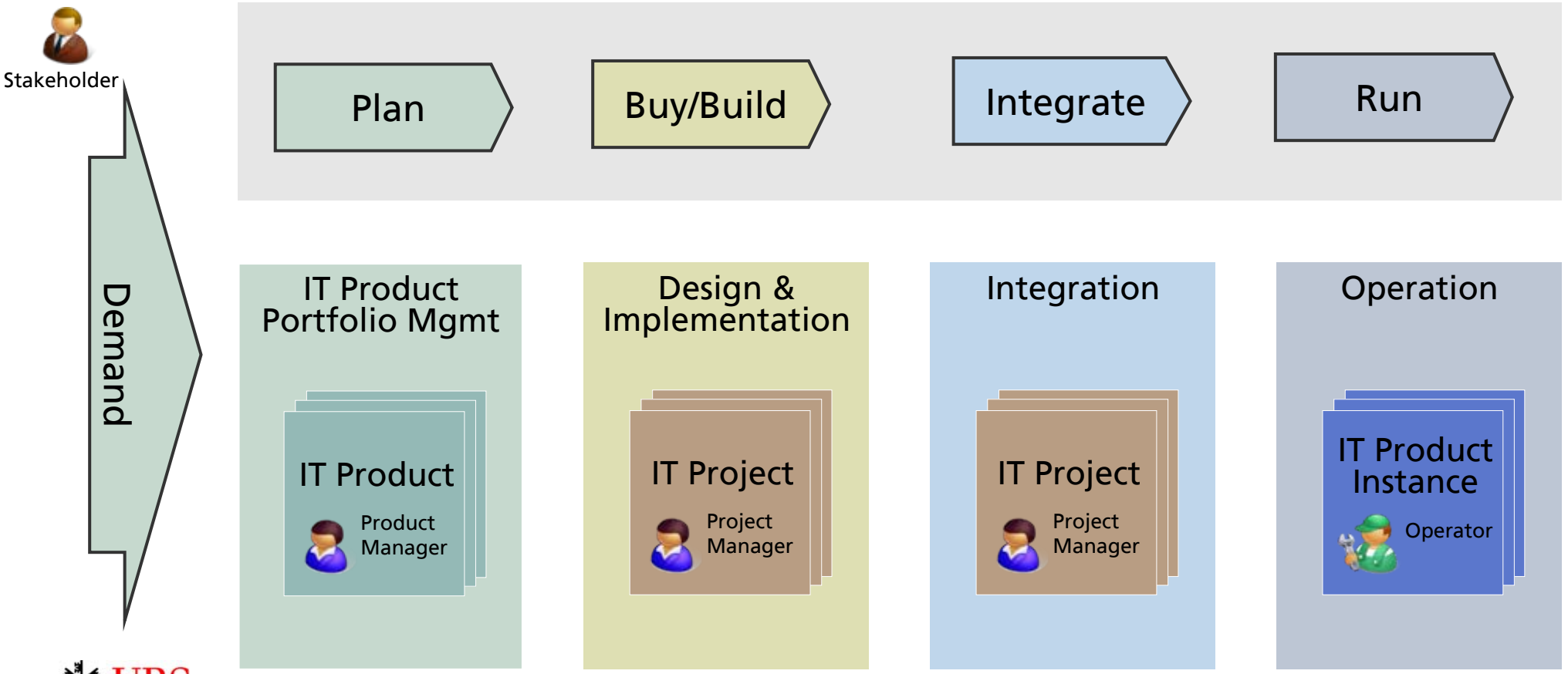
- Model Evolution
  Migrate existing instances after changing the Ecore Model

- Native UML support for CDO
  Storage of UML models with the scalability CDO provides

Vision

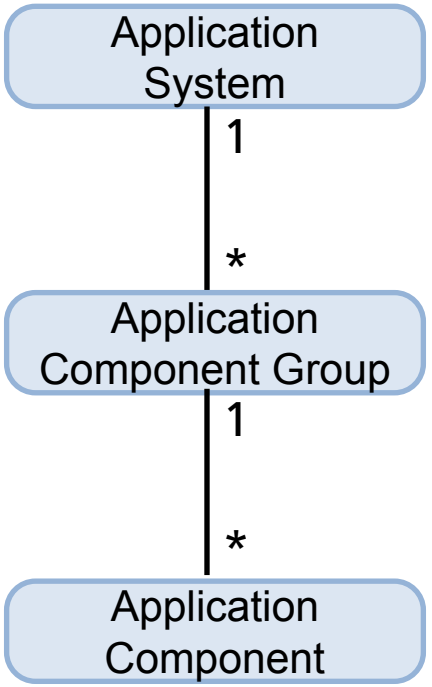
Model-Based Engineering

- **Capture & Preserve knowledge as models**
  - Along the IT Supply Chain



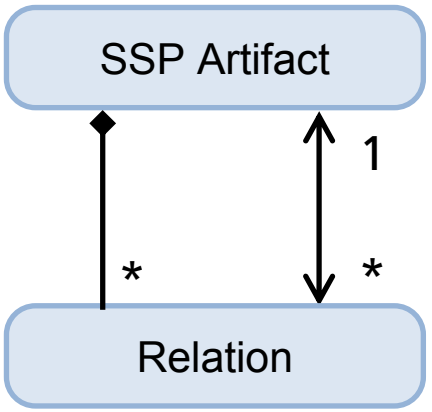
Appendix

Features – UBS Component Model



Appendix

Features – SSP Artifact Model



Appendix

Features - Identity Model