Generating Visualizations of Enterprise Architectures using Model Transformations

Sabine Buckl
Christian M. Schweda

Software Engineering betrieblicher Informationssysteme (sebis)
Ernst Denert-Stiftungslehrstuhl, Lehrstuhl für Informatik 19
TU München

wwwmatthes.in.tum.de

Agenda

Visualizing the Application Landscape: Why and How? (Status Quo)

Our Approach to Generating Visualizations of Application Landscapes

Glimpse on Our SoCaTool

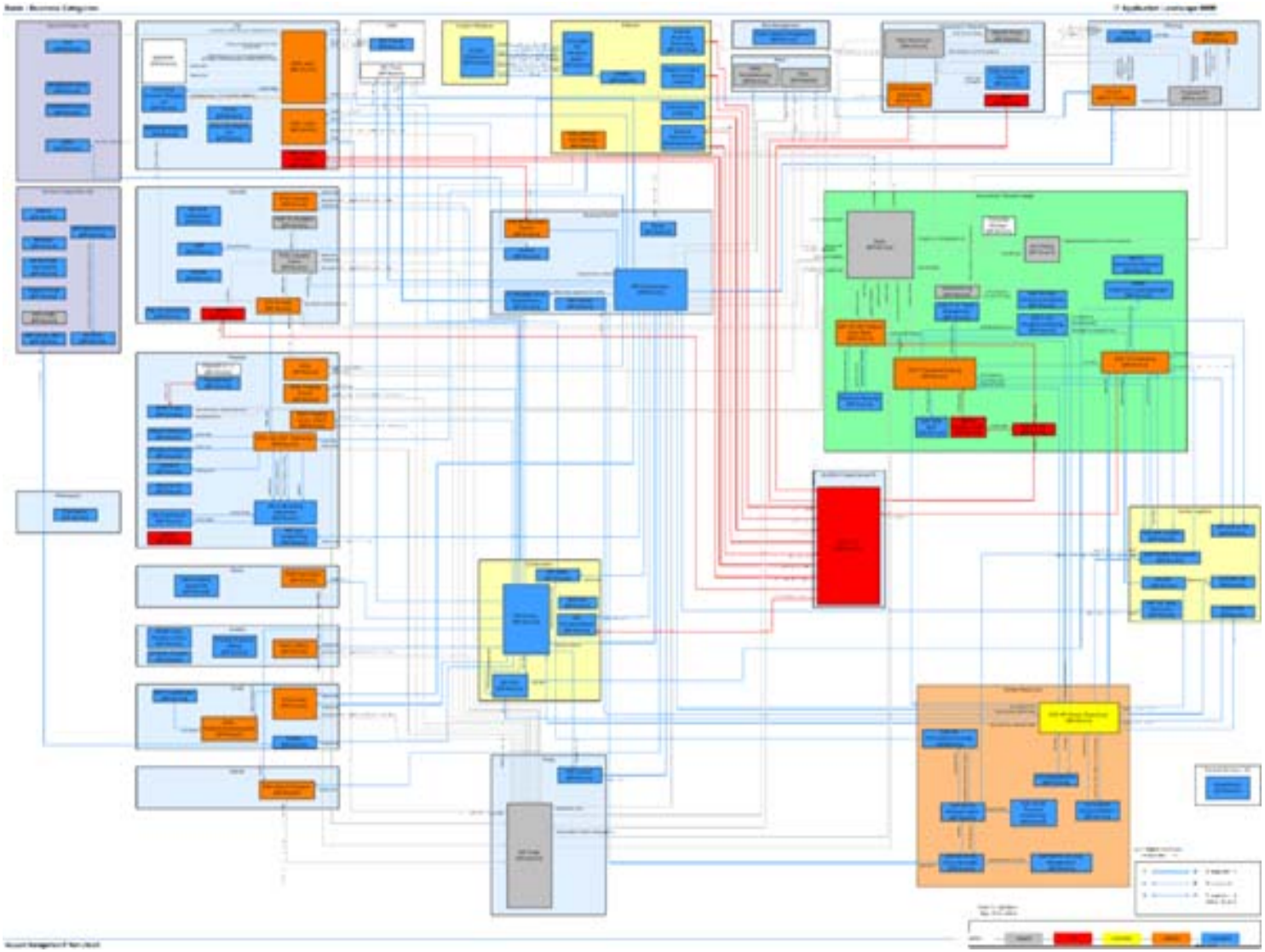
Why do enterprises visualize their application landscape?

Enterprise Architecture (EA) and the management thereof has recently gained rising attention in research and in practice.

Commonly, the application landscape is regarded to be an essential part of the Enterprise Architecture.

Concluding, the application landscape as part of the management body of EA management is visualized for documenting and communicating.

Visualizations like below exist in many Enterprises …



… and they are mostly “drawn”.

Causes for this:

- No established method for the creation and maintenance of such visualization yet exists.
- Most EA management tools do not support different visualizations, especially when concerning automated positioning.

Consequences of “drawing”:

- Manual creation of the visualizations is an error prone and time consuming task.
- Visualizations therefore tend to contain aged data.
- No links between the data and the visualizations exists, therefore visualizations might display incorrect information.
- Most EA management tools provide the user with the possibility to introduce symbols without defined semantics in the context of the visualization.

How we started studying visualizations of the application landscape …

Motivation

- Increasing number of application systems
- Long lifetime of application systems

needs

- Protect existing and planned IT investments
- Global and holistic view on the application landscape
  - Functions, processes, products,
  - Costs, benefits,
  - Lifecycles of elements, projects,
  - …

Status quo

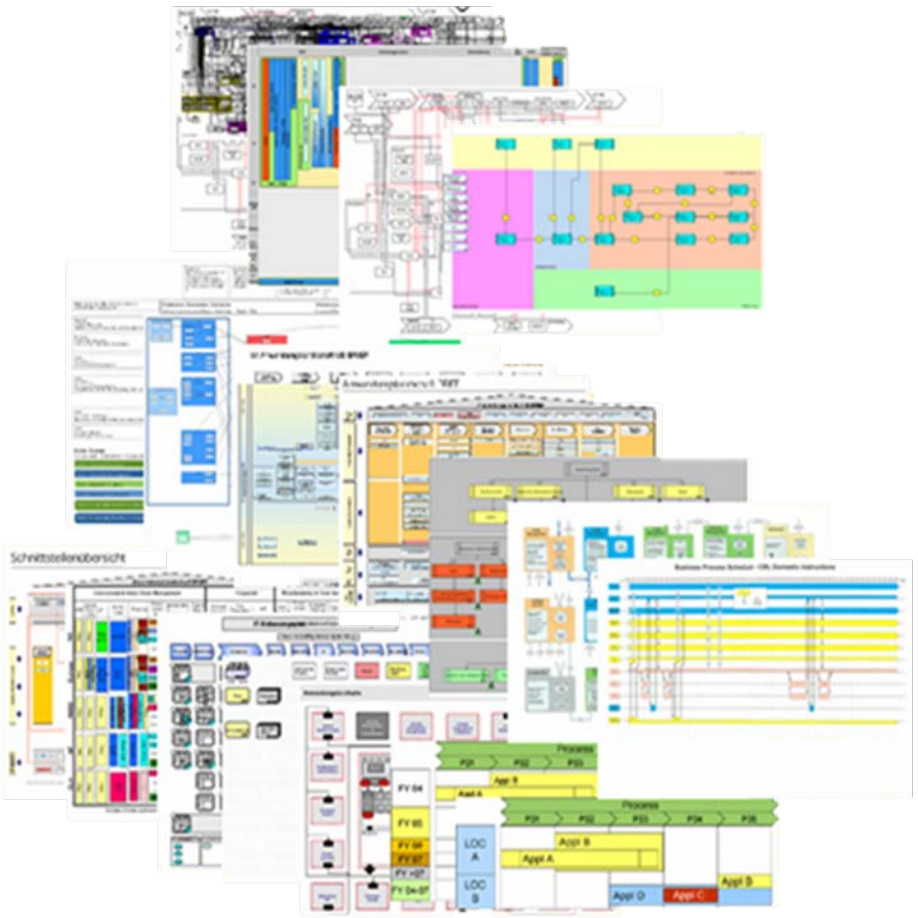
- Models and methods for documenting and developing application systems
  - Unified Modeling Language (UML)
  - Event-driven Process Chains (EPC)
  - Entity/Relationship Diagrams (ER)
- Models and Methods for documenting and planning application landscapes
  - ???

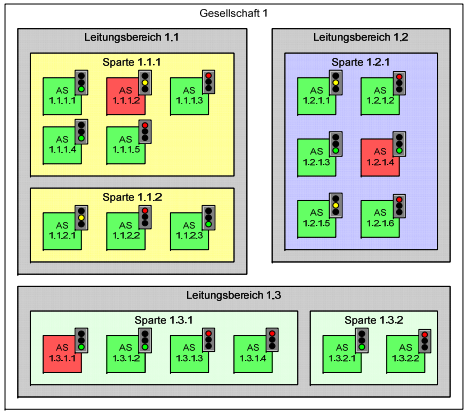
What are the goals of the project?

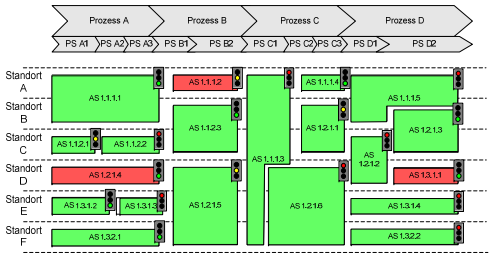
- Analyze existing models and methods for documenting and planning application landscapes
- Develop a model for software maps
  - Providing possibility for automatic view generation
    - Including prototype for a software cartography tool
  - Focusing on modeling, not on drawing!
- Identify successful visual representations
  - Driven by concerns of stakeholders
  - Communication instrument from IT operators to CIOs
- Establish a common terminology for documenting and planning application landscapes
- An information model for documenting and planning application landscapes
  - Including information elements relevant for application landscapes
  - Defining attributes and relationships between these relevant elements

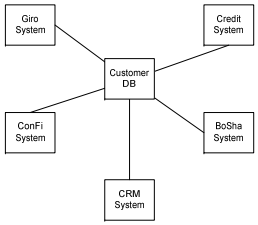
➔ Models and methods for documenting and planning application landscapes

What did we achieve?





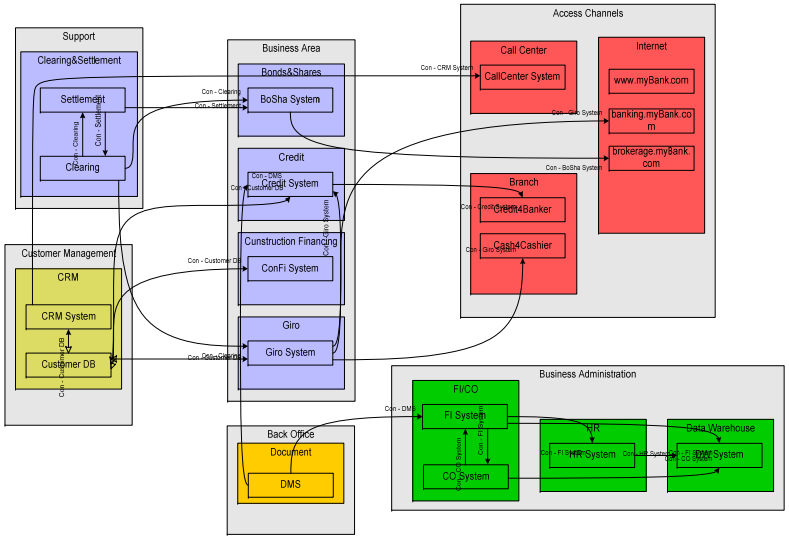


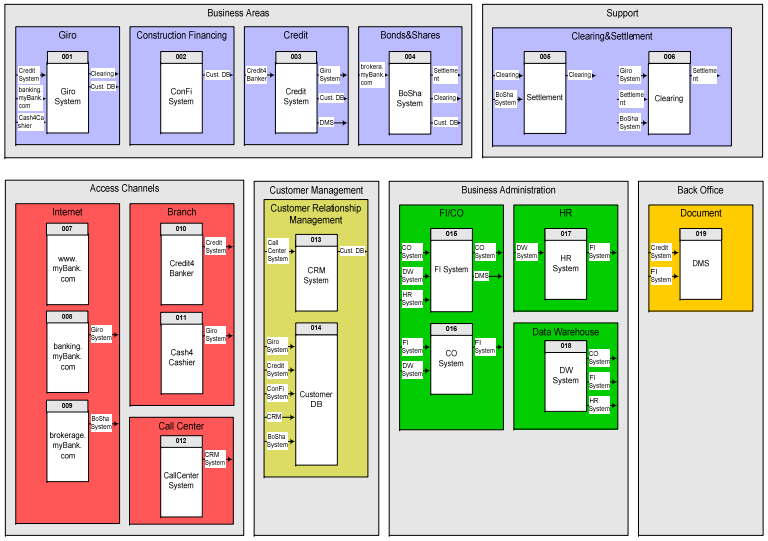


- Consolidation of different maps to three map types.
- Initial interviews with stakeholders about relevant aspects

Example 1: Cluster Map

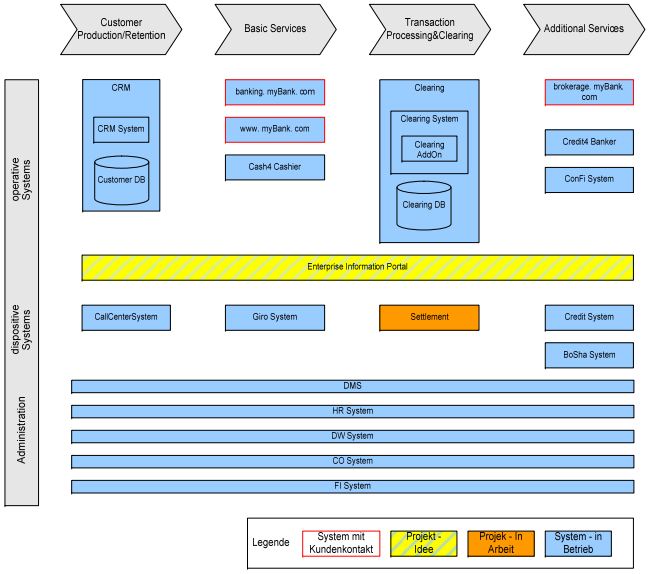
- Partitioning maps in logical units using cluster for
  - Functional areas
  - Organizational units, ...
- Positioning the cluster *manually* by
  - optimizing map size
  - putting related elements close to each other
  - using enterprise standards (e.g. customer at the right side)

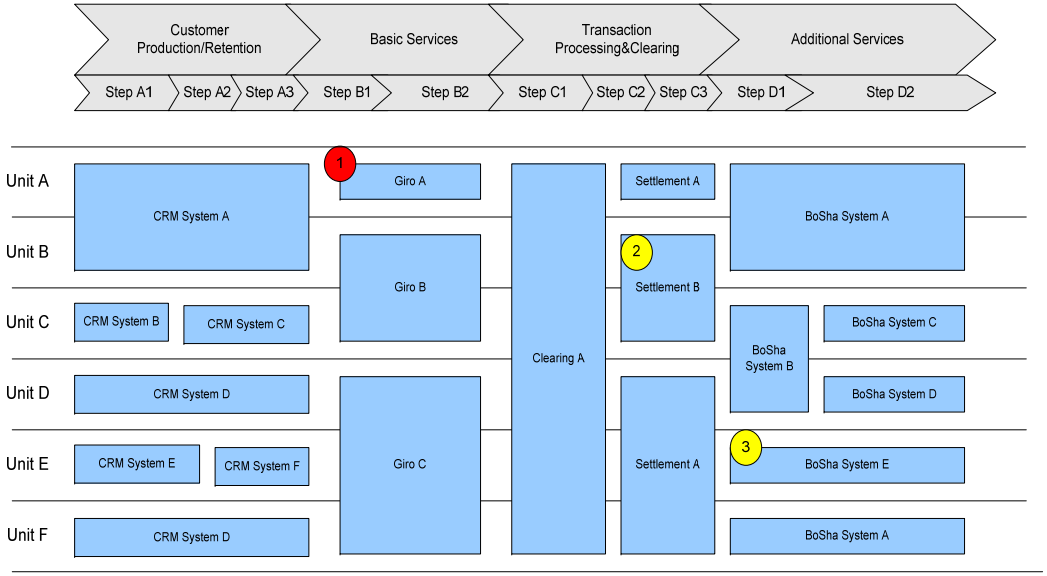




Example 2: Process Support Map

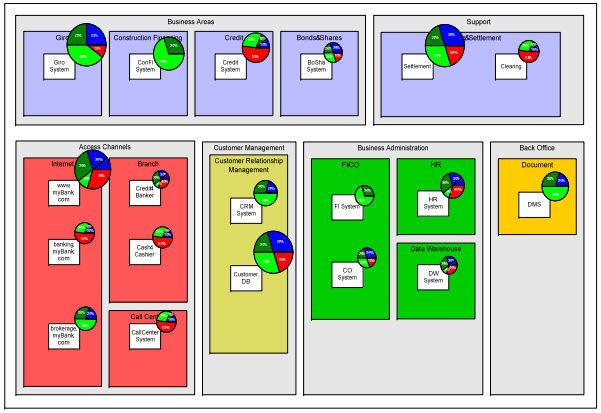
- Positioning along x- and y-axis
  - x-axis for business processes
    - Levels 0 up to 3
    - Linear processes
    - Notation comparable with value chains
  - y-axis for organizational units, system classes, products, etc.

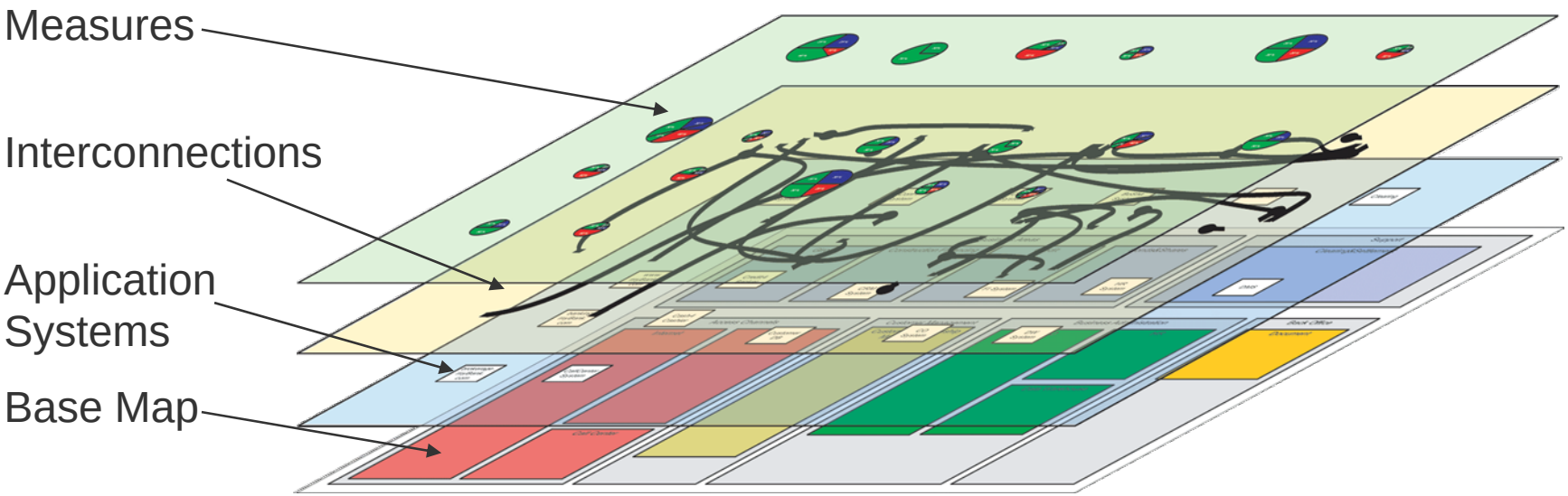




Different Aspects can be visualized on a Software Map using a Layer Principle

- Visualizing aspects on layers
  - Associations between application systems and other objects (processes, business objects, ...)
  - Measures with different symbols
  - Interconnections (Interfaces)





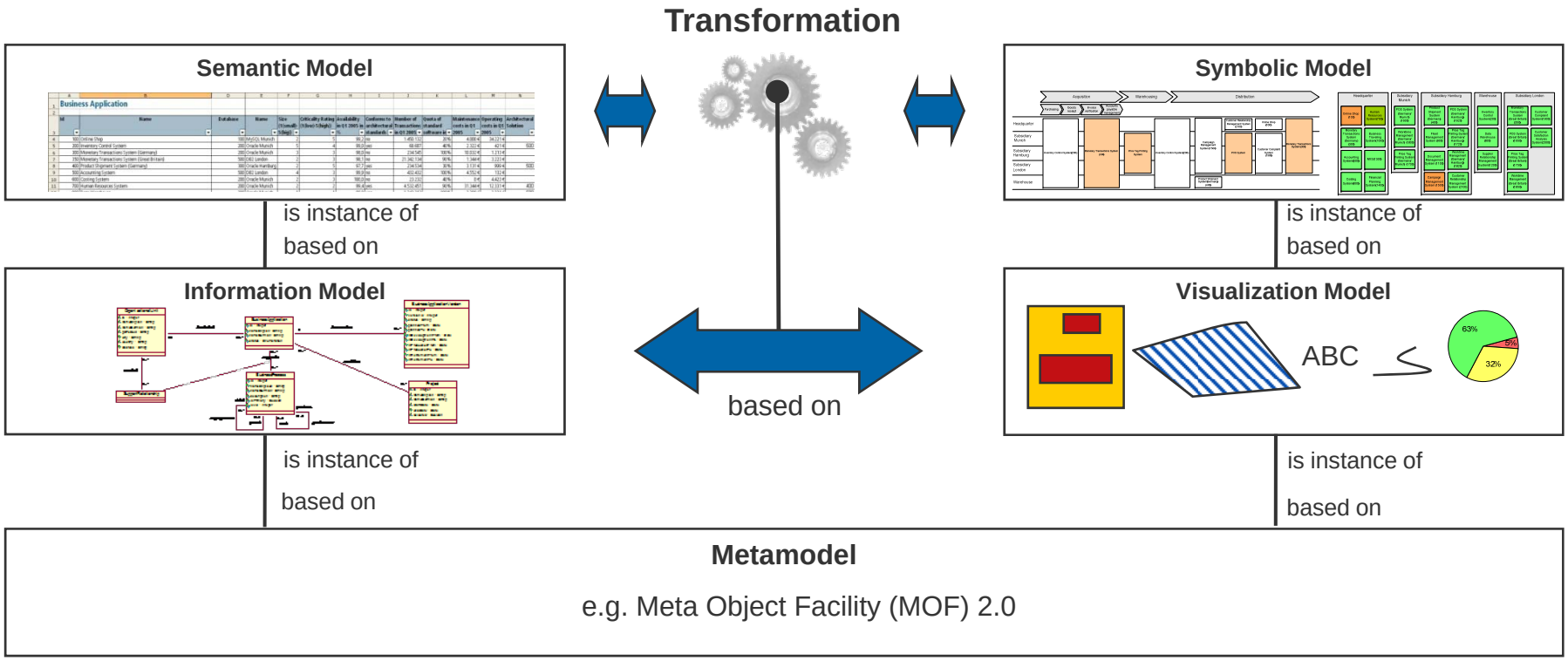
Agenda

Visualizing the Application Landscape: Why and How? (Status Quo)

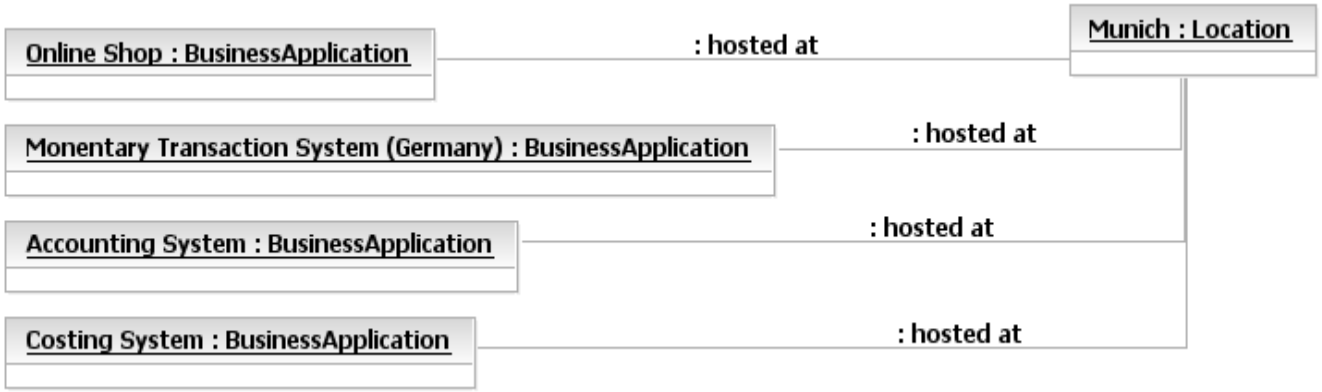
Our Approach to Generating Visualizations of Application Landscapes

Glimpse on Our SoCaTool

Our Model Transformation Approach includes a Strict Separation of Data and Representation



The Left Side …



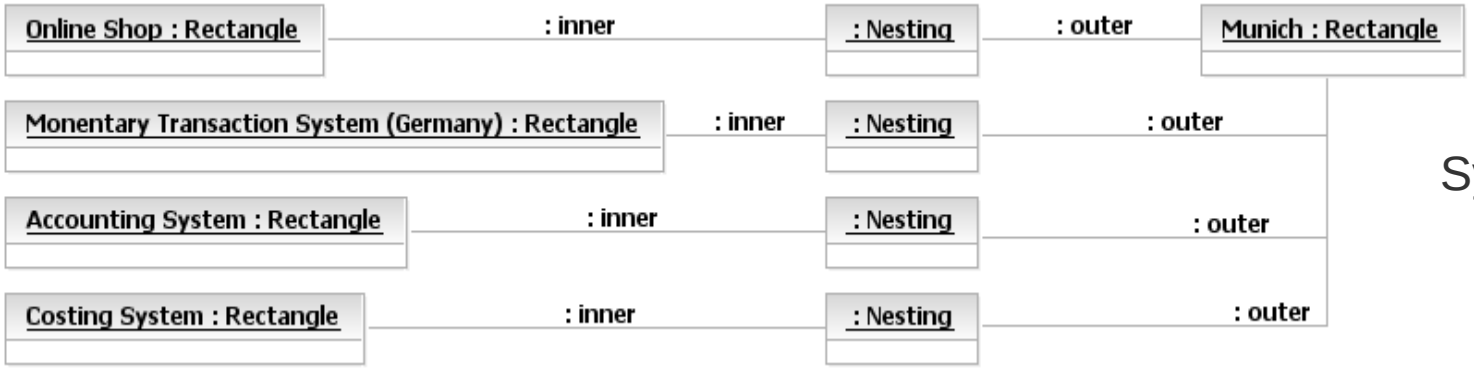
Semantic Model

DeployedApplicationSystem

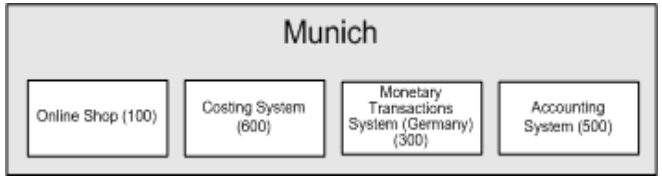
id	name	availability	mainenanceCosts	version	criticality	users	plannedFrom	plannedTo
100	Online Shop	99,2%	4.000,00 €	1.0	1	50	01.06.2006	30.09.2006
120	Online Shop	99,5%	- €	1.5	1	50	01.12.2007	31.01.2008
200	Inventory Control System	99,0%	2.322,00 €	1.0	2	30	01.07.2006	30.11.2006
220	Inventory Control System	99,1%	- €	1.5	2	30	01.01.2008	31.01.2008
300	Monetary Transactions System (Germany)	98,0%	10.032,00 €	1.0	2	10	01.05.2006	30.09.2006
350	Monetary Transactions System (Great Britain)	98,1%	1.344,00 €	1.0	2	10	01.01.2006	28.02.2006
400	Product Shipment System (Germany)	97,7%	3.131,00 €	1.0	3	15	01.05.2005	30.09.2005
500	Accounting System	99,9%	4.552,00 €	1.5	2	10	01.03.2006	01.05.2006

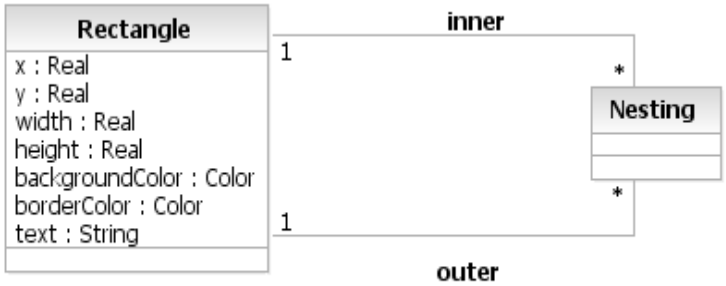
Information Model

… the Right Side …



Symbolic Model



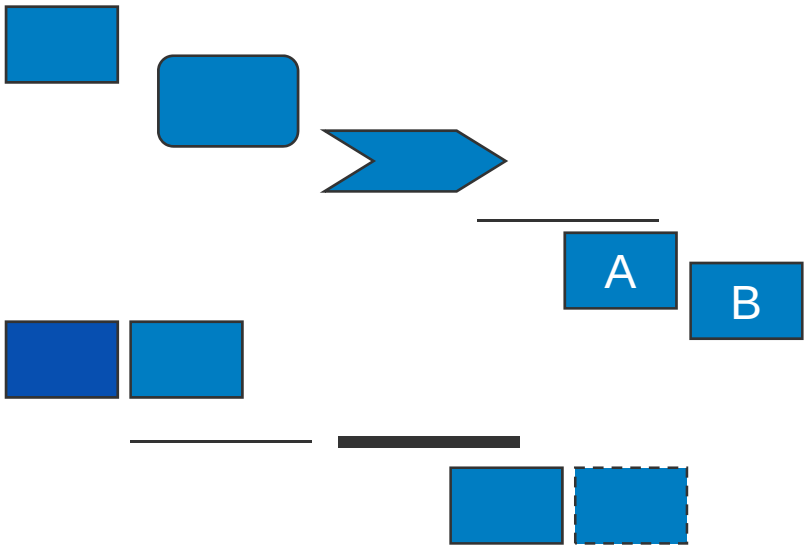


Visualization Model

… contains Map Symbols, Visual Variables, and Visualization Rules …

Map Symbols

- Rectangle
- Chevron
- Line
- …

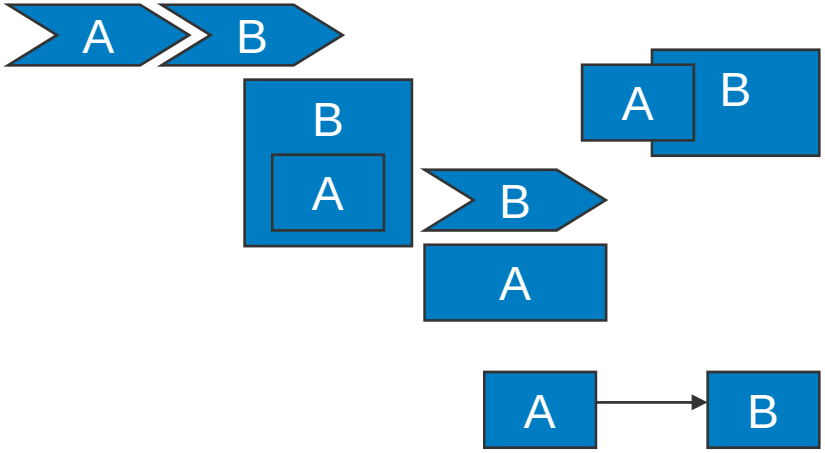


Visual Variables

- Background color
- Line size
- Line type
- …

Visualization Rules

- Planar map symbol to planar map symbol
  - Order
  - Non-intersection
  - Partial overlap
  - Nesting
  - Alignment (horizontally/vertically)
- Planar map symbol to linear map symbol
  - Connection



…, and the Middle

```
rule Location2Rectangle {
  from
    infoObject : Semantic.Location
  to
    symbol : Symbolic.Rectangle (
      text = infoObject.name,
      backgroundColor = #CCCCCC
    )
)
rule BusinessApp2Rectangle {
  from
    infoObject : Semantic.BusinessApplication
  to
    symbol : Symbolic.Rectangle (
      text = infoObject.name + “(“ + infoObject.id +“)“
    ),
    rule : Symbolic.Nesting (
      inner = symbol,
      outer = transforming (infoObject.hostedAt)
    )
)
```

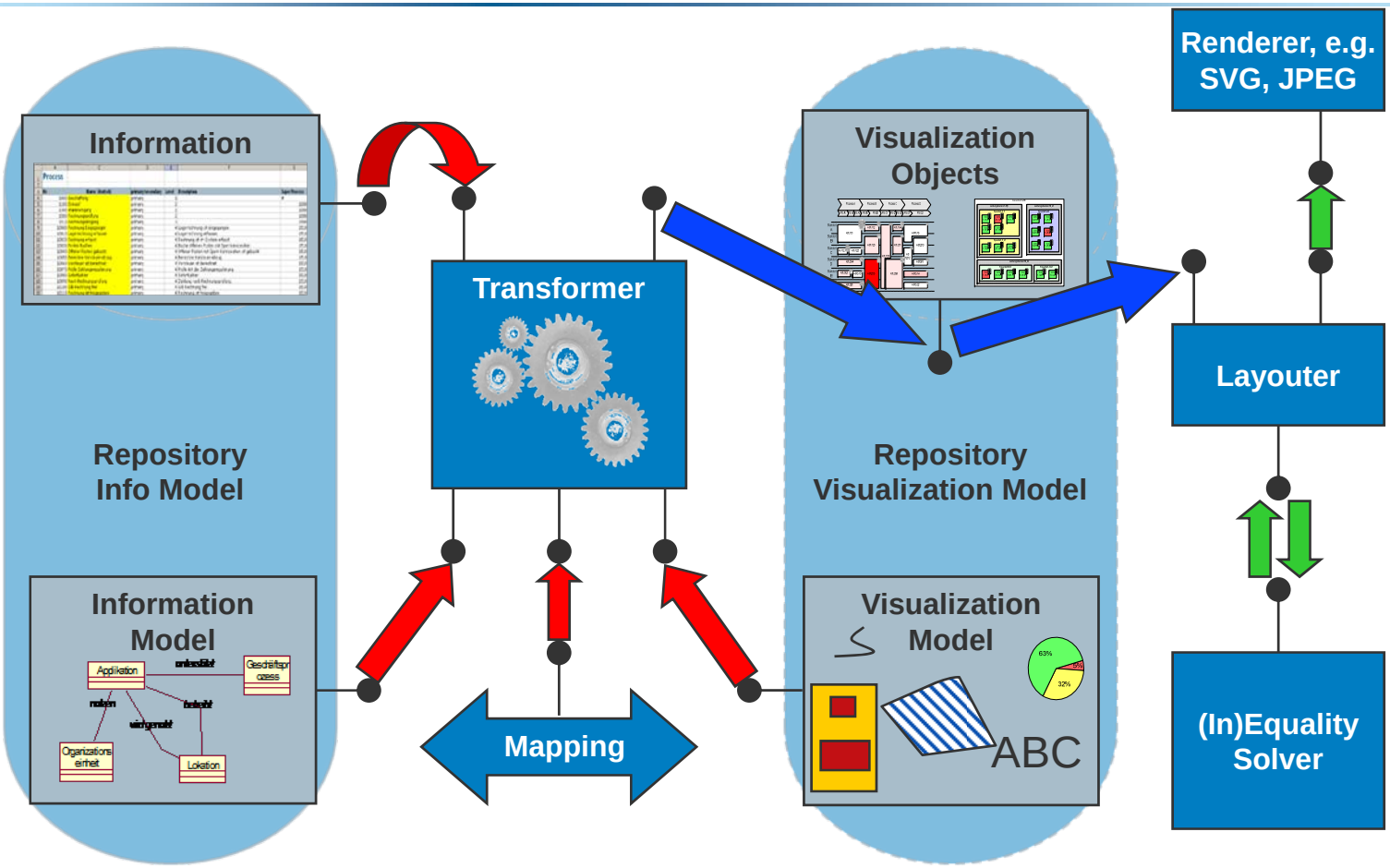
Agenda

Visualizing the Application Landscape: Why and How? (Status Quo)

Our Approach to Generating Visualizations of Application Landscapes

Glimpse on Our SoCaTool

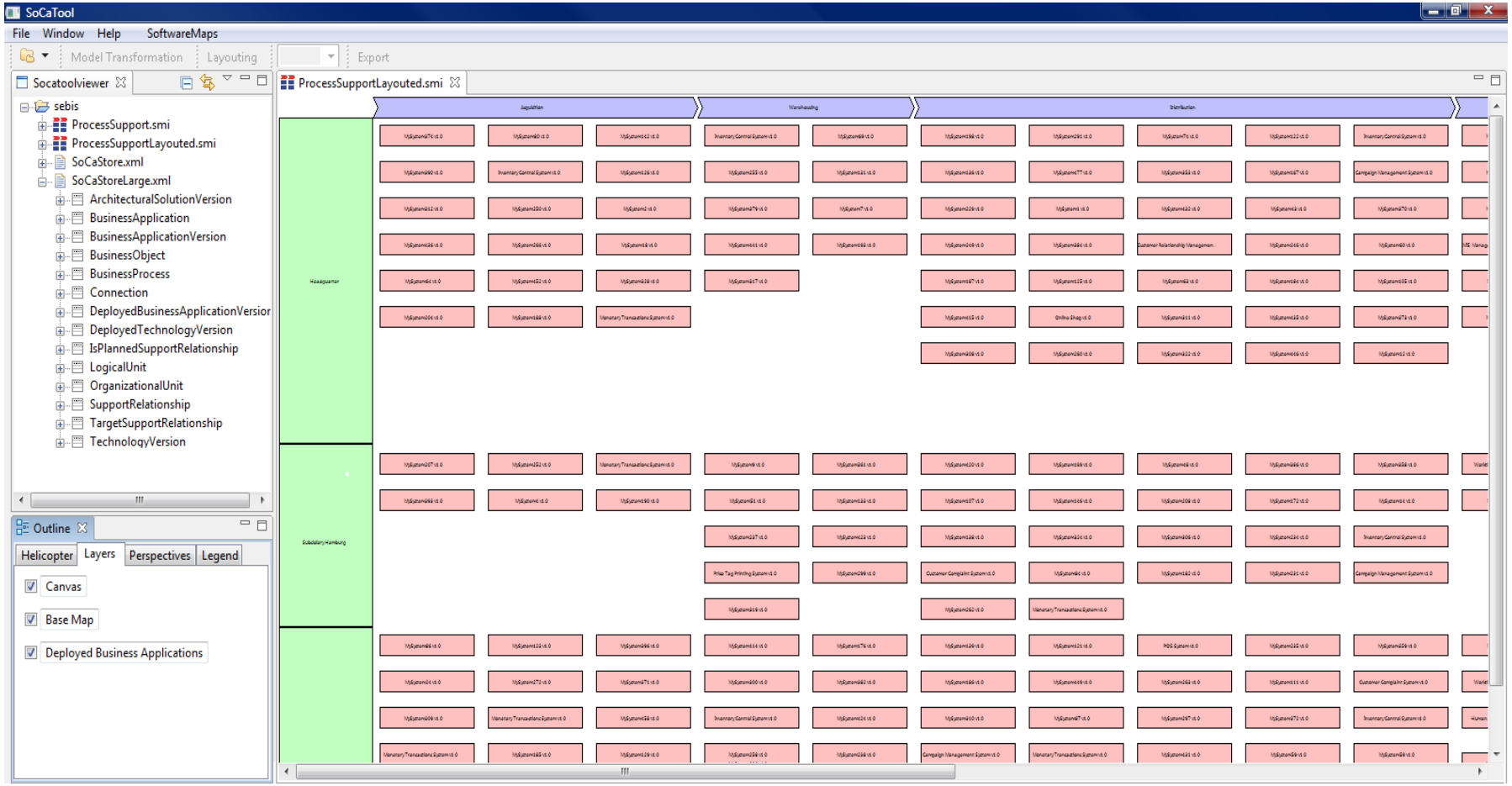
The Inside of the SoCaTool



Legend:

Step 1 Step 2 Step 3

How the SoCaTool looks like



What is still to be done?

The tool is just prototypical:

- Make it easier to specify the rules of the transformation.
- Make the layouter more powerful, especially concerning linear symbols.
- Enable interactions with the visualizations:
  - Semantic preserving ones
  - Semantic changing ones
- Incorporate an add-on for calculating application landscape metrics.
- …

Thank you very much!

Discussion...

For more information about the project:
www.softwarekartographie.de (in German)