



GEFÖRDERT VOM



Bundesministerium  
für Bildung  
und Forschung

# Method and Tool Support for Deriving Architectures from Partner Network Modeling

Marktplatz der CrEST-Abschlusspräsentation

Oktober 2020

# Content

- 1. Ecosystems of Collaborative Systems**
- 2. Meta-Model**
- 3. Terms and Concepts**
- 4. Modeling / Evolvment Method**
- 5. Tool Support**
- 6. Deriving Architectures from Partner Networks**

# 1. Ecosystems of Collaborative Systems

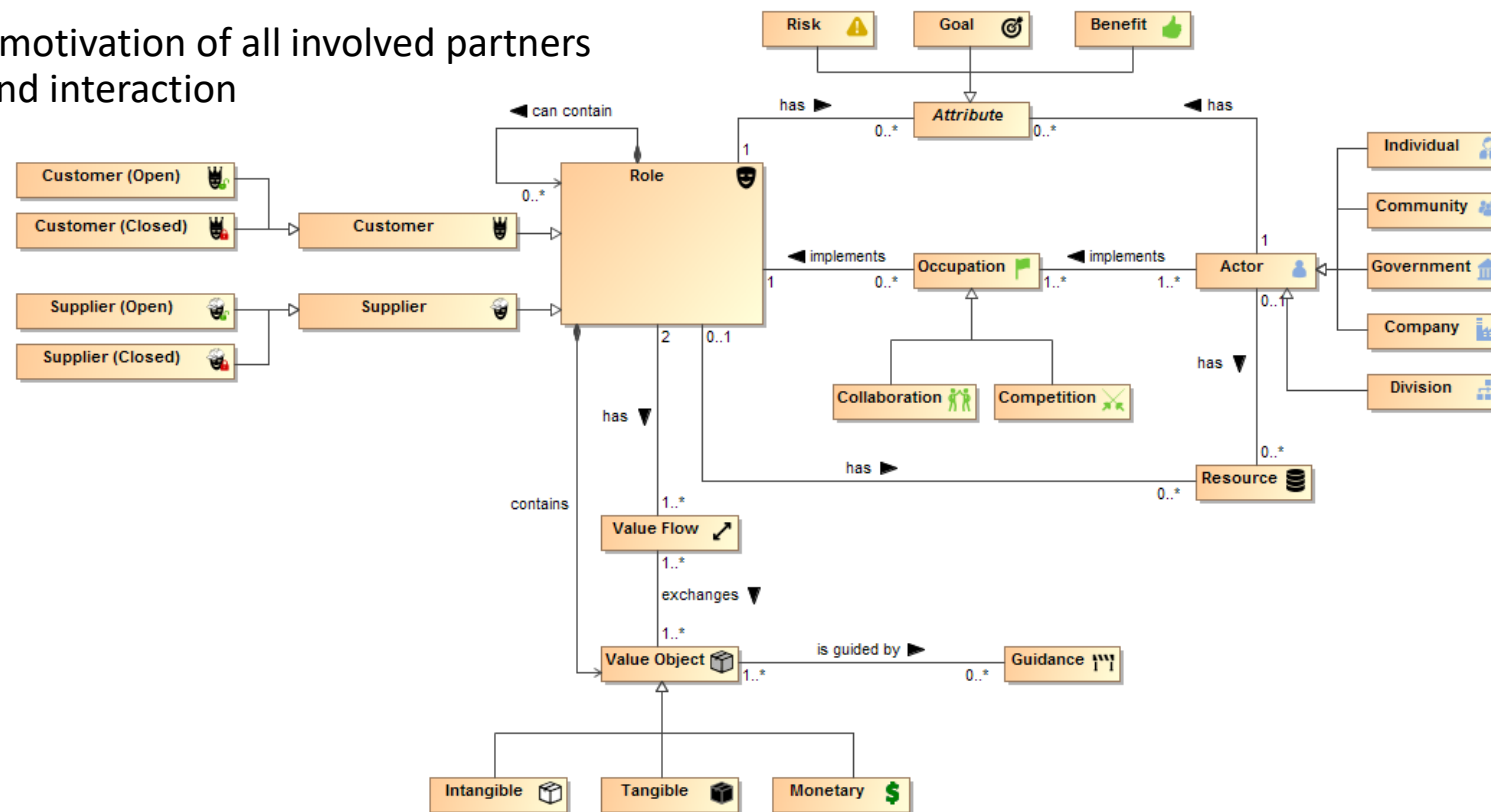
- **Motivation:**
  - More and more systems are developed or operated with joined efforts of several partners
  - Together the partners can tackle more complex problems, spend less effort, and reach more customers
  - For all partners, participation should result in a win-win situation
  - Examples: Google Playstore, Hadoop Software stack
- **Definition:** An “Ecosystem” consists of:
  - a set of **independent, interacting systems, software and service providers and consumers**, together with the **relationships** among them;
  - a **shared market** for systems, software and services;
  - a **common technological base** comprising a reference architecture, core assets and/or standards.
- **Relevance for CrEst**
  - Different systems in a Collaborative System Group (CSG) are developed and operated by independent partners
  - Developers of CESs do not know the complete system structure and behavior in advance
  - Systems have to collaborate smoothly on the one hand, each partner must have a motivation to build/operate a system on the other hand
  - CESs as well as CSGs have specific goals to be achieved, even competition

## 2. Meta-Model for Ecosystems of Collaborative Systems

⇒ **Partner Network helps to understand and document ecosystem business case and to structure problem space:**

- Common language and understanding
- Way to analyze and visualize ecosystems
- Basis for communication and discussion
- Helps to understand motivation of all involved partners about contribution and interaction

⇒ **Meta-Model:**



## 3. Terms and Concepts

### ***Partner Network***

- should be understood as a set of tasks that are logically grouped where each group of tasks is assigned to a *Role*
- granularity of grouping is defined by the modeler
- set of logically-grouped tasks needs to be taken over by someone – a *Role*

### ***Role***

- is defined by the tasks it takes over in the *Partner Network*
- is an abstract representation of a participant who contributes to the *Partner Network* by providing *Value Objects* (goods and services) related to the tasks
- abstracts from concrete participants of a *Partner Network* and does not express who performs these tasks
- may aggregate sub-roles
- exchanges goods and services with each other by means of *Value Flows*
- can be of type
  - customer (consumes goods and services produced in the *Partner Network* ) or
  - supplier (produces such goods and services)
- can be marked as
  - open (can be occupied by arbitrary actors, visualized with open lock) or
  - closed (can be occupied by selected actors only due to *Partner Network* specific barriers, visualized with closed lock)

## 3. Terms and Concepts – cont'd

### ***Value Flow***

- represents an exchange relationship between two *Roles*
- two *Roles* should only be connected by a single *Value Flow*
- exchange direction (from-to) is illustrated by an arrow on the *Value Flow*
- each *Value Flow* comprises one or many *Value Objects* that are exchanged

### ***Value Object***

- is a service, product, information, or any kind of valuable goods being exchanged between roles through *Value Flows*
- are created by a *Role* and assigned to that *Role*
- *Roles* may act as sender and receiver of *Value Objects*
- granularity of a *Value Object* is defined by the modeler
- kind of *Value Objects* may impact the distribution channel and the exchange logistics of *Value Objects*
- can be classified as
  - tangible (e.g., APIs, frameworks, and hardware),
  - intangible (e.g., information and services), or
  - monetary (e.g., fees)

## 3. Terms and Concepts – cont'd

### **Actor**

- *Roles* can be occupied by *Actors* by means of a so-called *Occupation*
- *Actors* represent concrete participants that occupy one or more *Roles* through one or more occupations
- *Actors* can be specialized as
  - individual,
  - community,
  - government,
  - Company, or
  - division
- *Actors* can have individual interests expressed by *Attributes*

### **Occupation**

- connect *Roles* and *Actors* and specify the type of relationship among *Actors* that occupy the same *Role*
- can be of type
  - collaboration (i.e., multiple *Actors* that occupy the same role collaborate for fulfilling the *Roles'* tasks),
  - competition (i.e., multiple *Actors* that occupy the same *Role* compete among each other for fulfilling the *Roles'* tasks), and
  - neutral (i.e., no statement about the kind of interaction among *Actors* that occupy the same *Role*).
- *Actors* may occupy the same *Role* with different *Occupations*  
(*Actor* can collaborate with one *Actor* while competing with another one)

## 3. Terms and Concepts – cont'd

### **Resource**

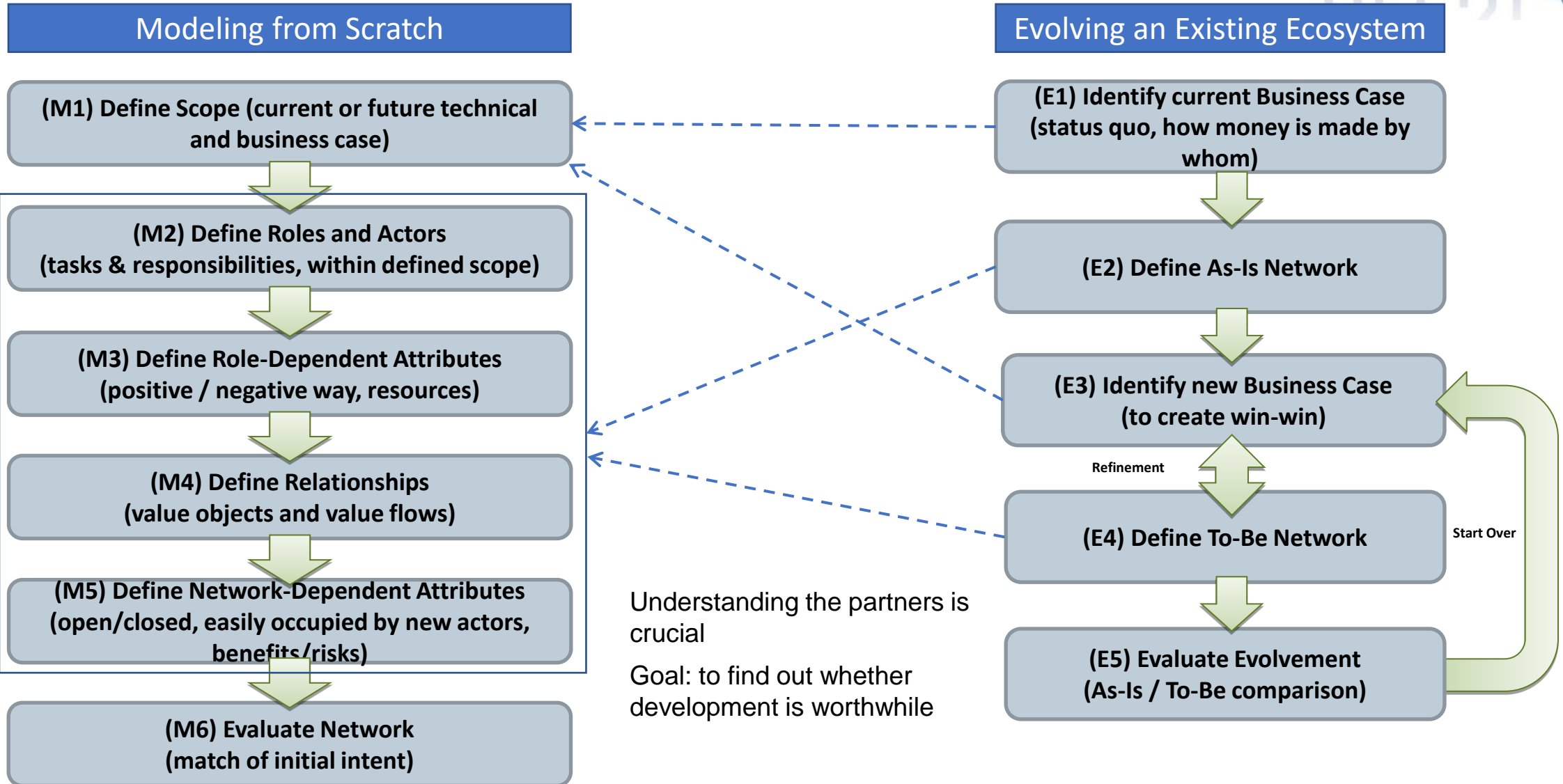
- *Roles* and *Actors* usually possess *Resources* that help them to achieve their tasks
- *Resources* are assets that belong to *Actors* or *Roles* and that are used to occupy *Roles* or to fulfill the tasks associated with *Roles*
- *Resources* can be knowledge, skills, or some legal ownership over some artifact or good

### **Attribute**

- *Actors* participate in the *Partner Network* for a certain reason (the goal) which gives them some kind of positive result (the benefit)
- An *Actors'* participation in a *Partner Network* depends on the benefits s/he expects from the participation in the *Partner Network* and the risks they take when participating. If an *Actor* is identified to have significantly higher risks than benefits, then there is a high probability that this *Actor* will leave the *Partner Network* in the near future unless this imbalance is remedied.
- The goal concept helps to choose and aligning partners and competitors. It also supports to derive impact scenarios, for example, whether the strategy works as planned and the *Partner Network* evolves.
- *Attributes* express interests of roles and can be of type
  - Risk: event with a negative impact on a *Role* which may be triggered with a certain probability by the role's participation in the *Partner Network*.
  - Goal: desired result or effect which this entity wants to gain or achieve with his or her participation in the *Partner Network*
  - Benefit: good or helpful result or effect an entity expects from his or her participation in the *Partner Network*.



# 4. Modeling & Evolvment Method



## 5. Tool Support (cf. video): Partner Network DSL in MagicDraw

Domain independent design methodology for deriving Partner Networks for collaborative embedded systems

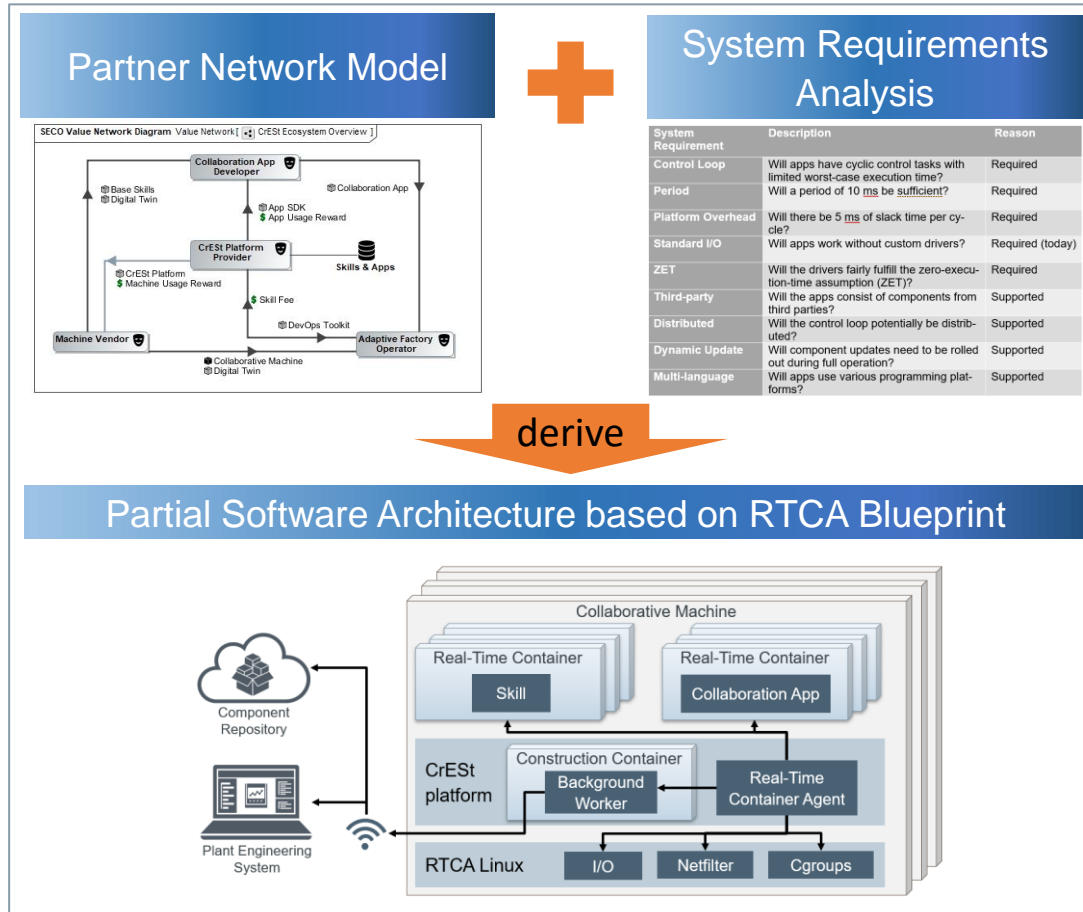
- integration into SPES\_XT Modeling Framework
- based upon a specific Domain-Specific Language (DSL) for Partner Network Modeling
- implemented in MagicDraw (in collaboration with MQ1)
- allows for formal, tool-based specification of partner networks and architectures

Tool is deployed in Amazon AWS Cloud

- to be used from anywhere
- without any need for local installation of MagicDraw
- without an individual MagicDraw licenses
- can be used for other DSLs (arc42 architectures, micro-service architectures etc.), too)

The accompanying video illustrates how to develop a Partner Network for the Use Case  
“Adaptable and Flexible Factory”

# 6. Partner Network to Software Architecture



- Based on the Real-Time Container Architecture, we proposed a method to quickly derive a partial software architecture and detailed requirements for collaborative embedded systems from a Partner Network model (cf. EC1.AP2.D1)
- The evaluation of the derived high-level architecture showed that the Partner Network model is a good input to software architecture work as our method quickly yielded a feasible architecture sketch.
- **Application example for Use Case „Adaptable and Flexible Factory“**