ESOSEG
Environment for Simulation, Operation and Optimization of Smart Energy Grids

Hochschule Ulm · TUM · SEKAS · SWU · SW PAN Stadtwerke Pfarrkirchen

CIMug Meeting 2017 - Herzogenaurach

Open-Source Middleware Platform for the data exchange within a Distribution Grid Operator

Basem Idlbi, Ulm University of Applied Sciences
Markus König, SEKAS GmbH

14.06.2017

ESOSEG
Environment for Simulation, Operation and Optimization of Smart Energy Grids

Agenda

- Project partners & objectives
- Software architecture:
  - Architectural patterns and principles
  - Adapter Generation
  - Prototype Implementation
- Utilization of CIM:
  - CIM mapping
  - Next steps
  - Open discussion

ESOSEG
Environment for Simulation, Operation and Optimization of Smart Energy Grids

Project partners

- University of Applied Science Ulm (HSU)
- SEKAS GmbH
- Technical University Munich (TUM)

Associated Partners:

- Stadtwerke Ulm (SWU)
- Stadtwerke Pfarrkirchen (SWPAN)
- Hessware GmbH

Project duration: 3 years

01.10.2015 – 30.09.2018

Funding agency: Federal Ministry of Economic Affairs and Energy (BMWi) (Promotion Reference Number 0325811C)

Project objectives

Background:

- High installation of distributed RE generation in the distribution grids
- High need for systematic grid analysis and data exchange

Main objective:

To develop a flexible and expandable platform, in order to ease the analysis of distribution grid structures, detect the critical grid conditions and investigate the possible grid reinforcement measures and their feasibility

Requirements of the platform:

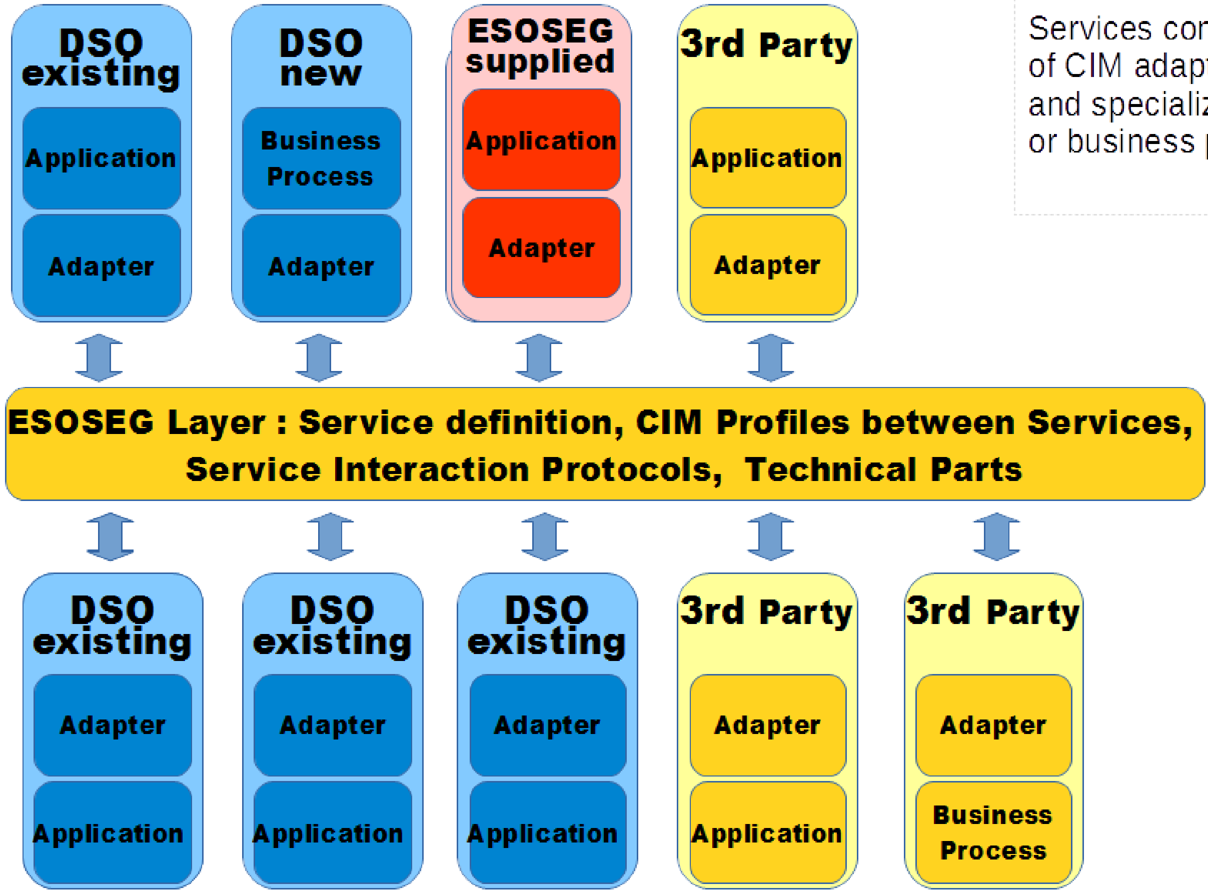
- Standardized information modelling for the data exchange
- Cost savings for DSOs → open-source framework
- Simple integration to the existing IT environment of the DSO

Architectural patterns and principles

- Lean microservices – ready for mobile, desktop, application server, cloud
- REST-based - proven technology stack
- No central database, data mastership remains in services – distributed system
- CIM-based data exchange between services – open for 3rd parties
- CIM adapters by model-based generation
- Existing applications wrapped in services
- Simple basic services (reusable),
  complex business case services (specific)
- External webservices can be integrated

ESOSEG
Environment for Simulation, Operation and Optimization of Smart Energy Grids

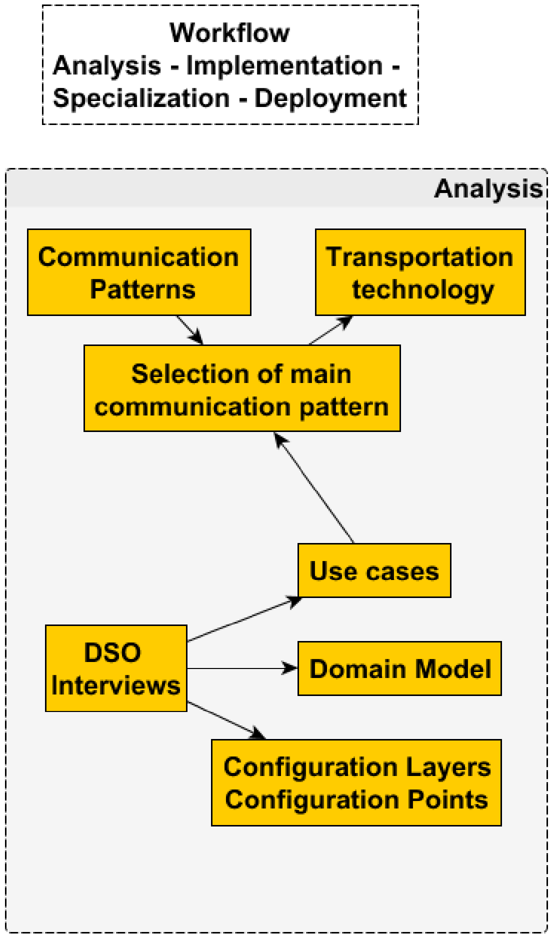
System Overview



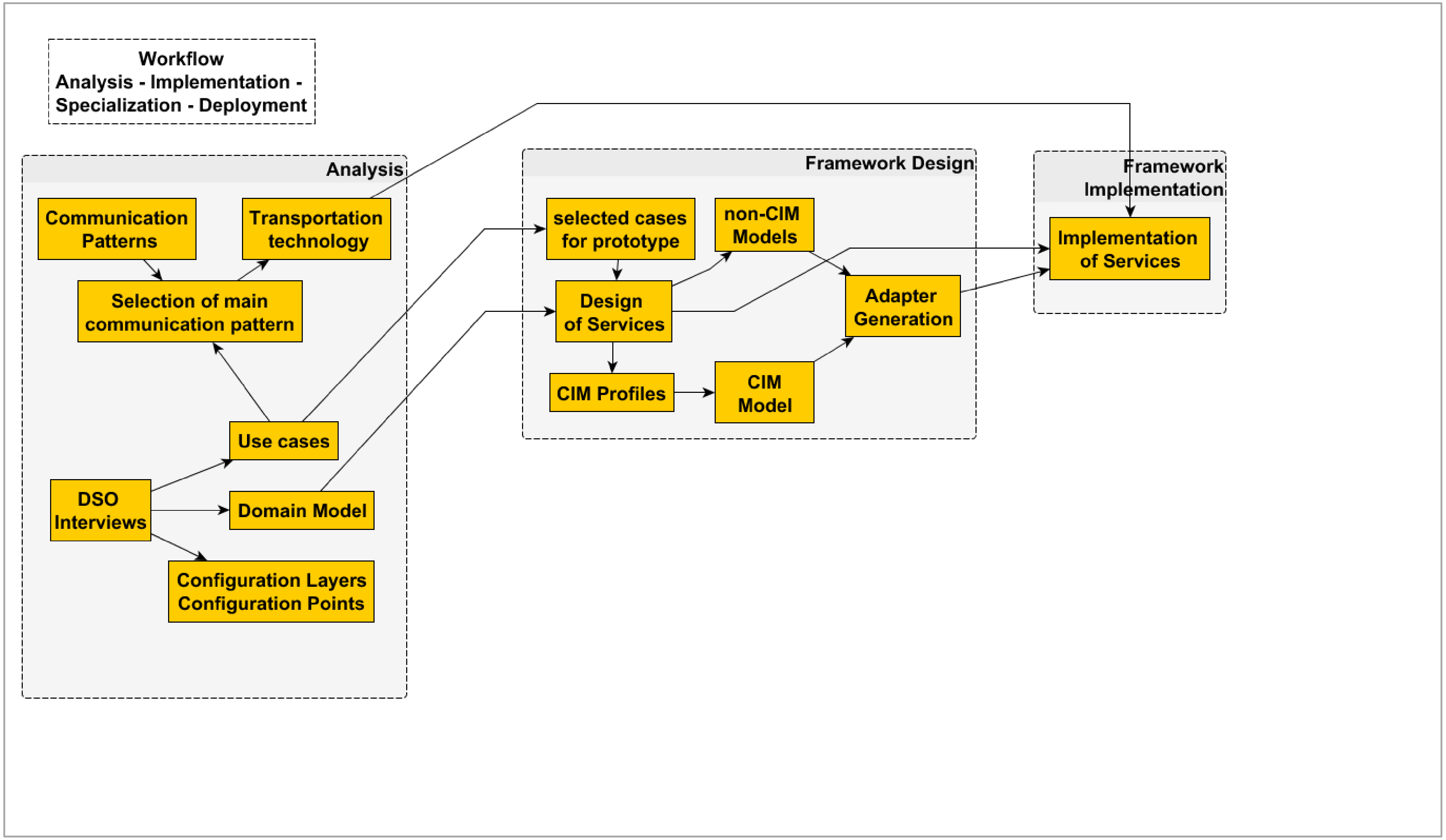
System Overview

Services composed of CIM adapter and specialized application or business process

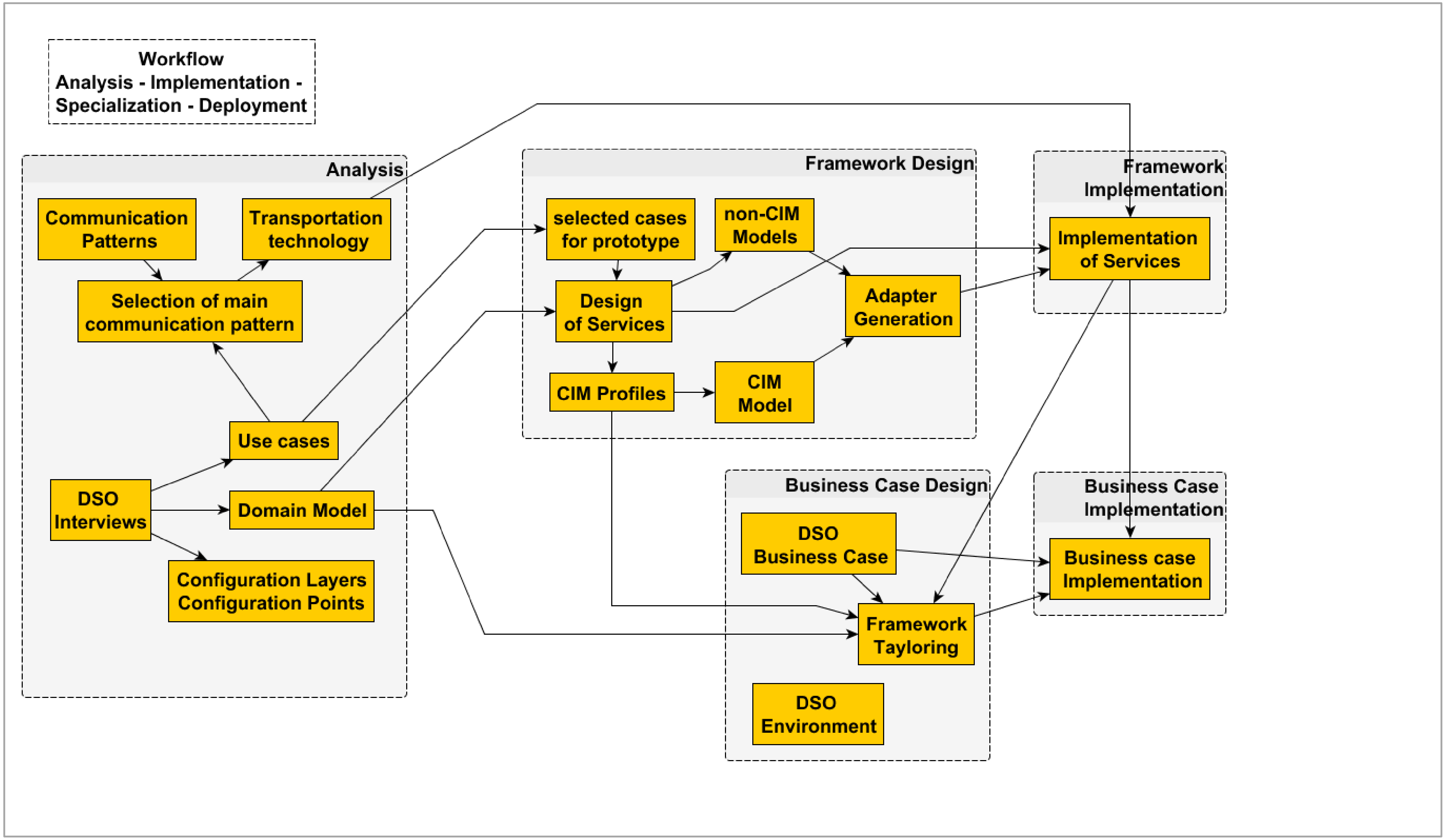
Development Workflow



Development Workflow



Development Workflow



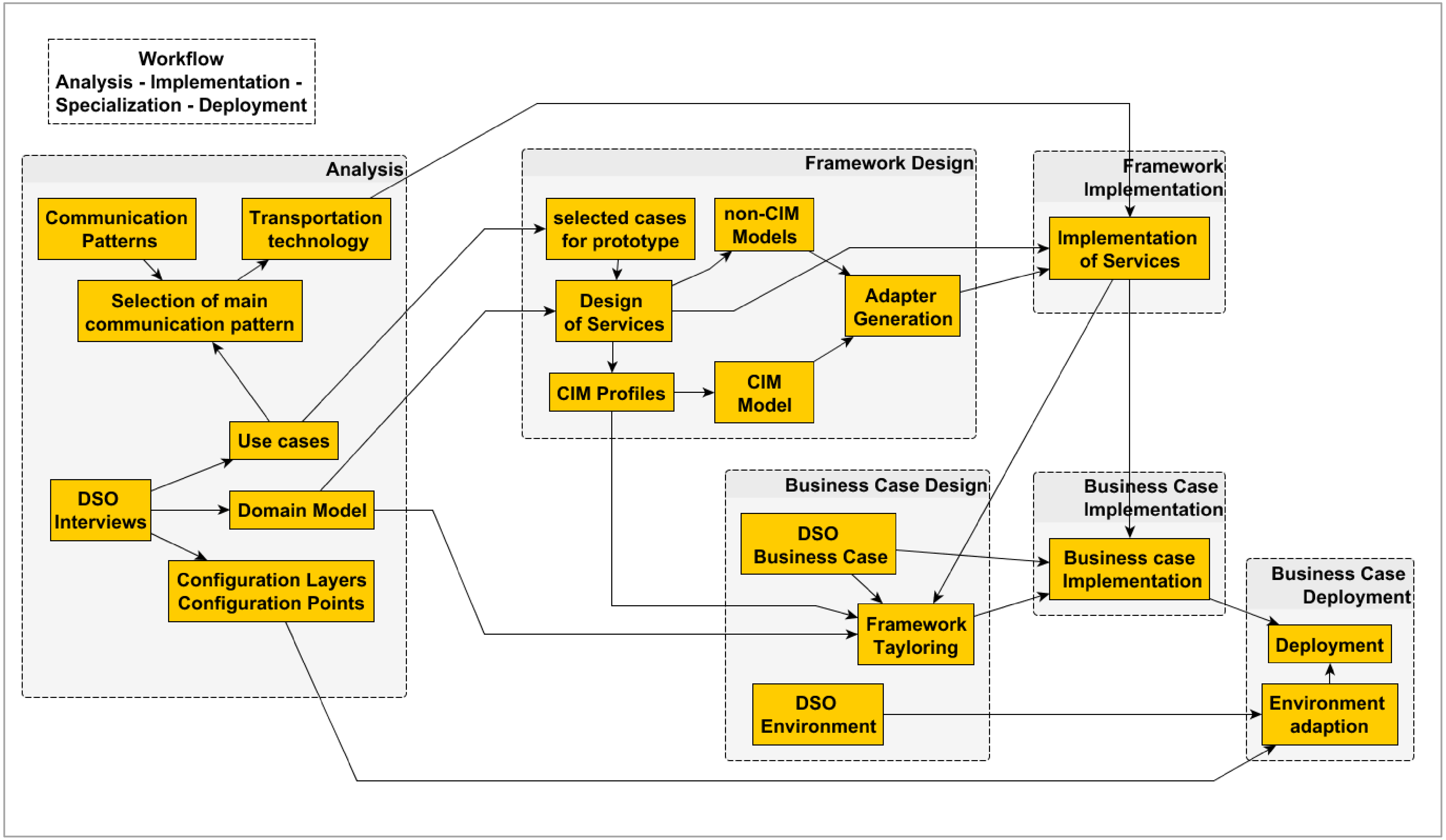
Basem Idlbi, Markus König, 14.06.2017

ESOSEG
Environment for Simulation, Operation and Optimization of Smart Energy Grids

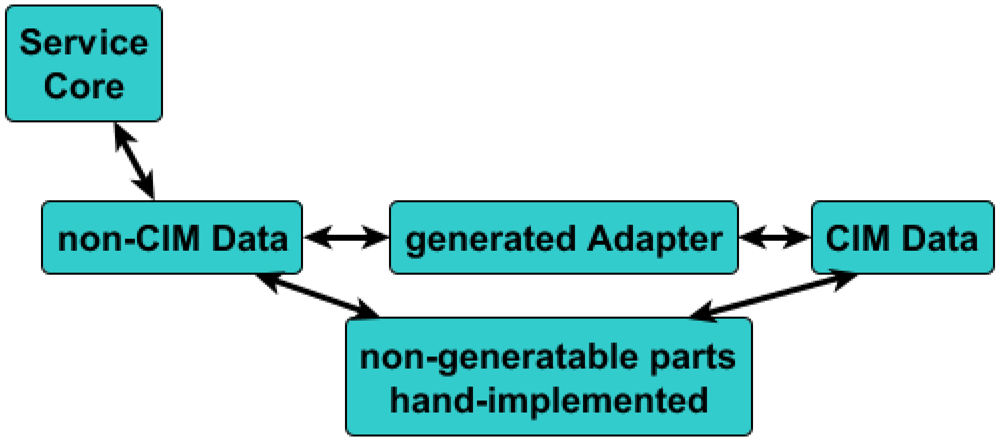
ESOSEG
Environment for Simulation, Operation and Optimization of Smart Energy Grids

Hochschule Ulm

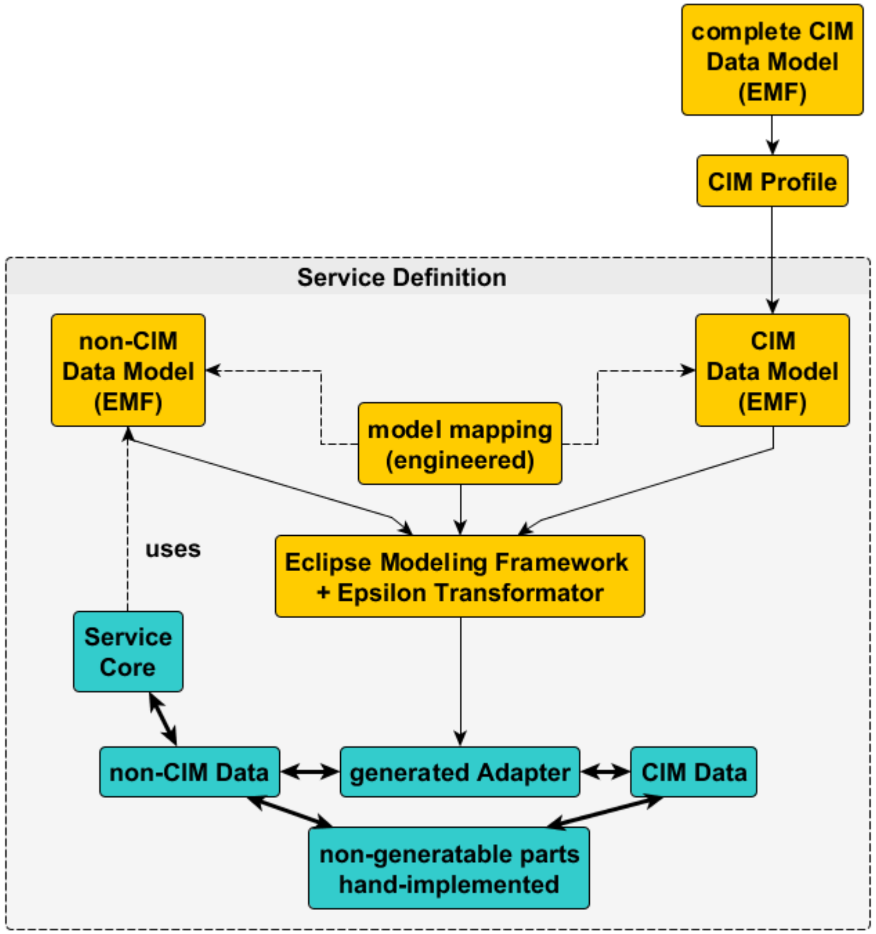
Development Workflow



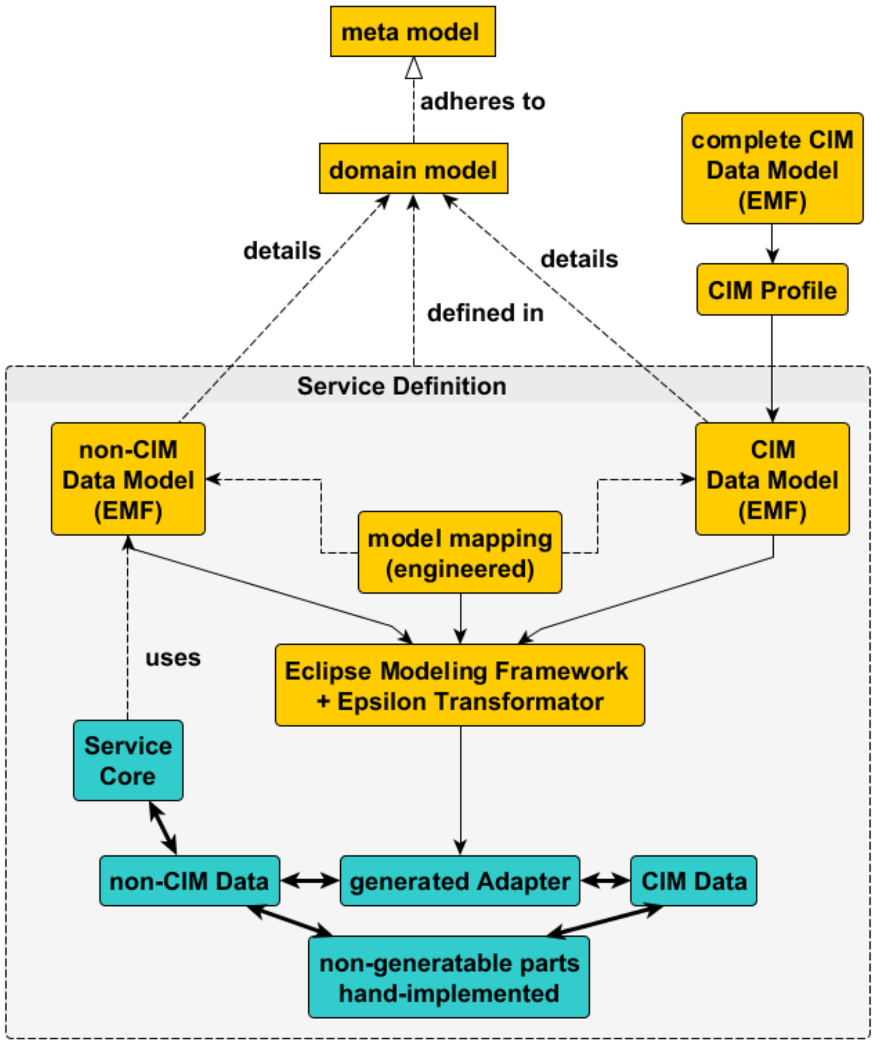
Adapter Generation



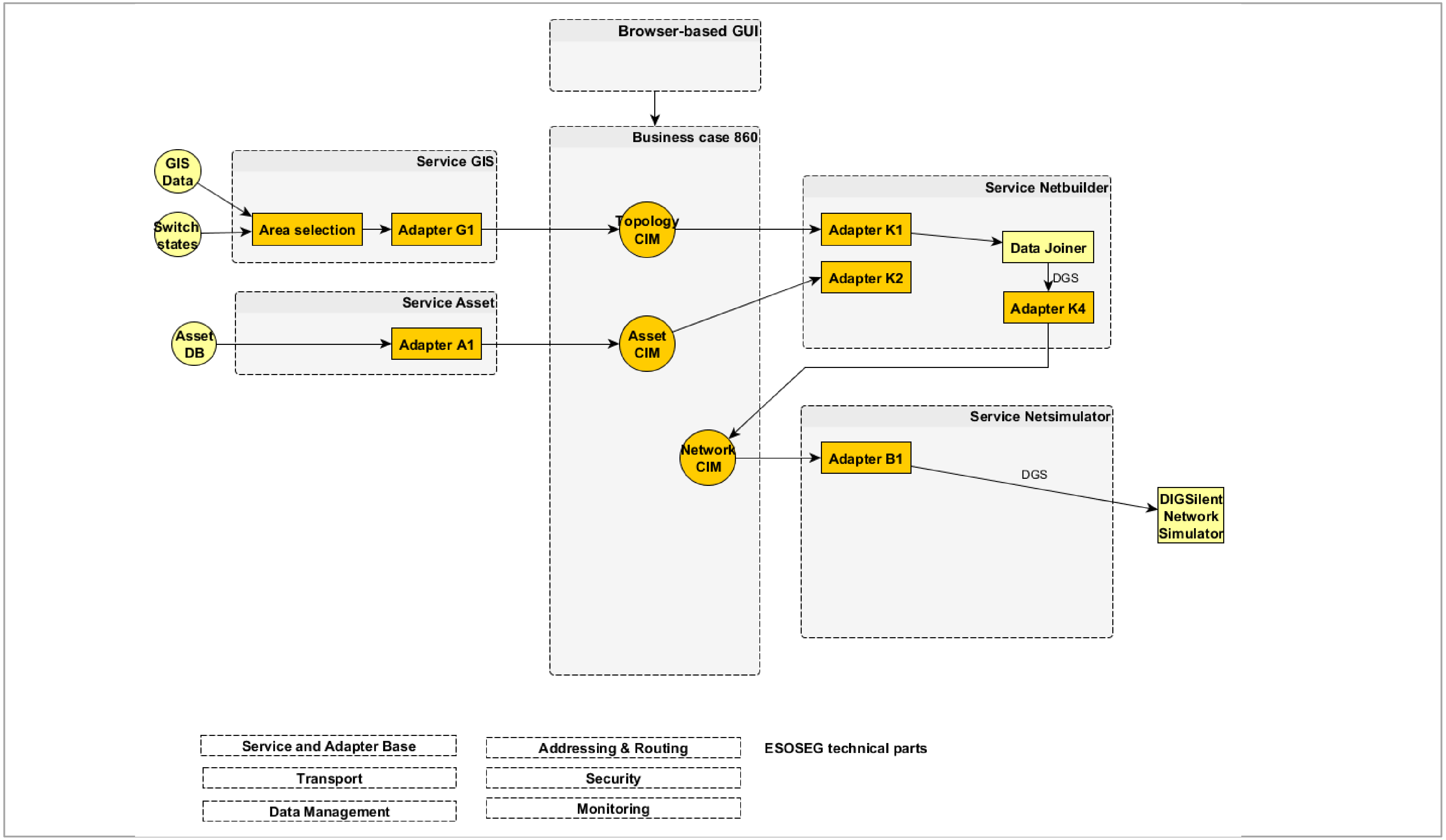
Adapter Generation



Adapter Generation

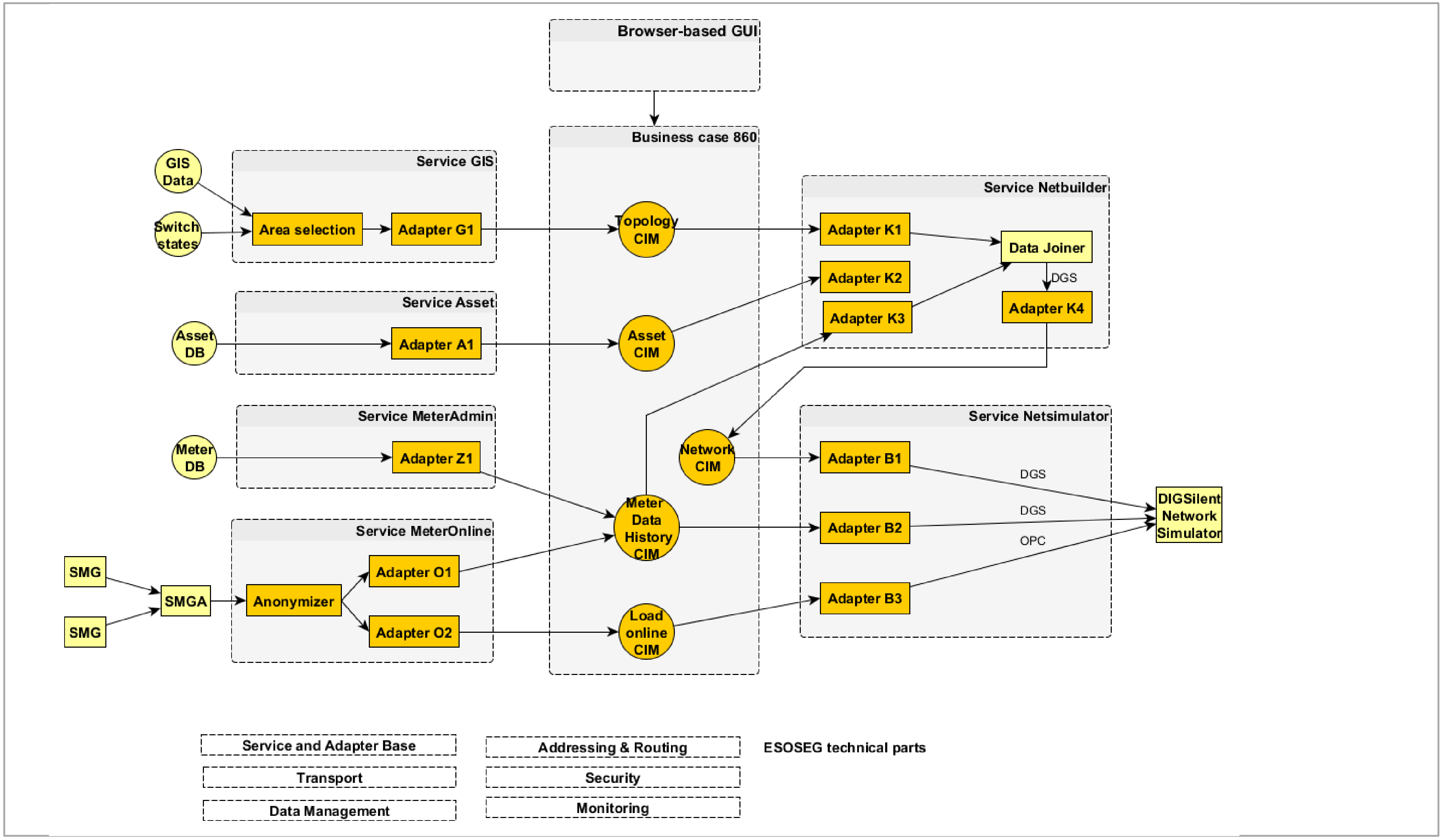


Prototype Implementation



ESOSEG
Environment for Simulation, Operation and Optimization of Smart Energy Grids

Prototype Implementation



Implementation summary

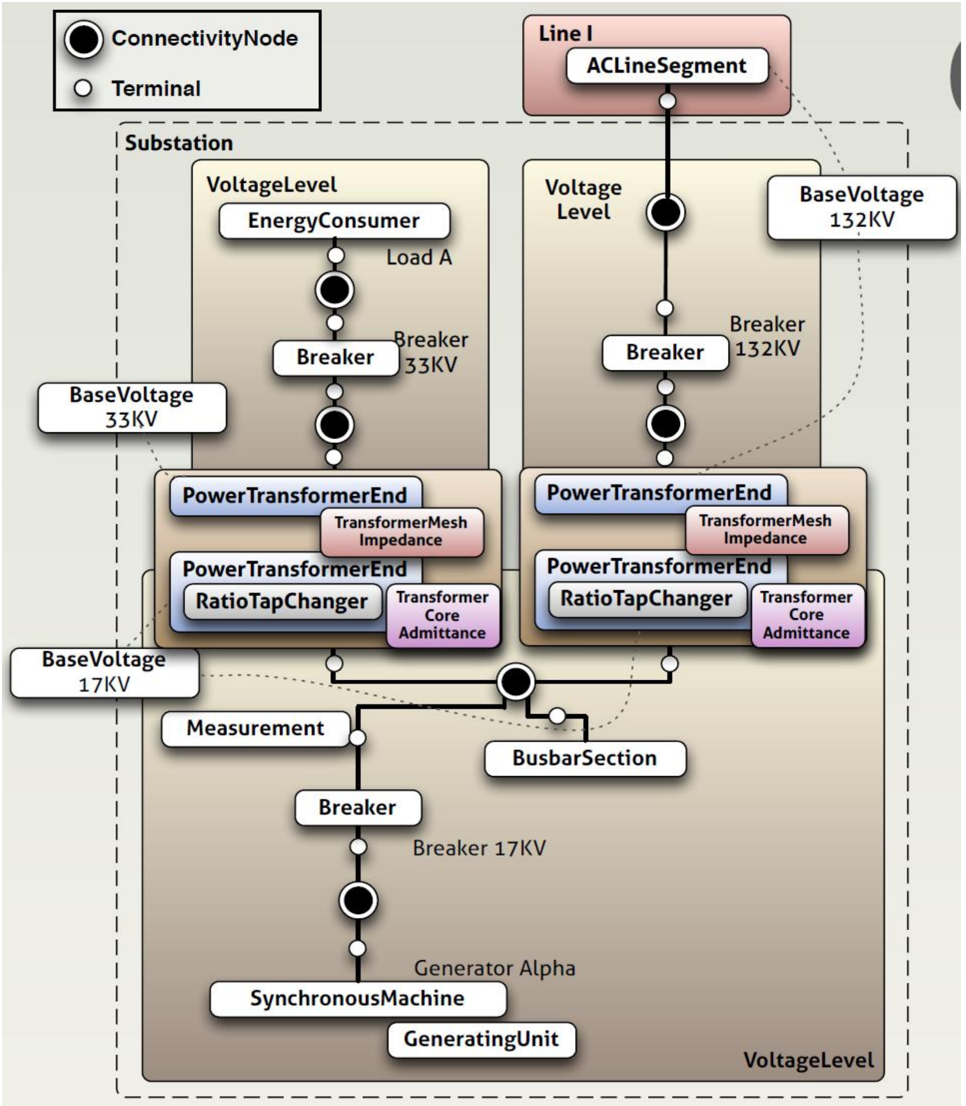
- Domain model as base for technical decisions, service cut, CIM profiles and choice of prototype scenarios
- Domain model as base for business case tailoring and well-defined concretization points for deployment
- Wrappers for business applications, no big bang integration
- Business alignment possible: Service – Data Responsibility – Department, e.g. Asset or GIS

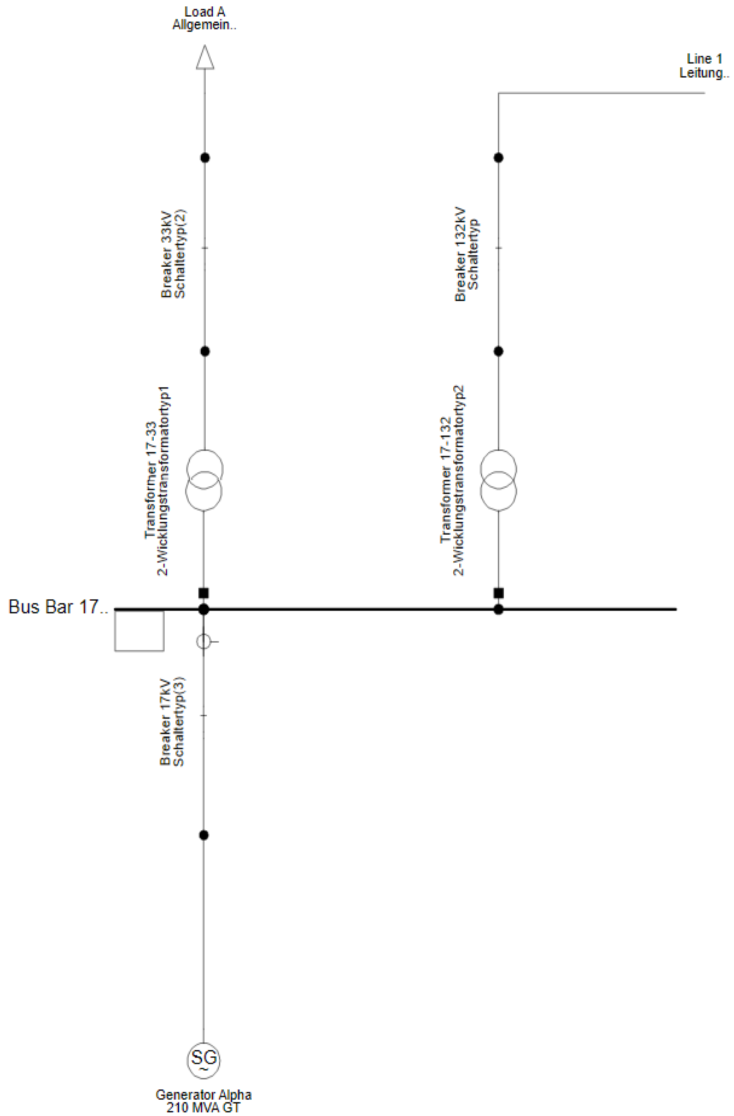
CIM mapping

- Hand-implemented mapping environment:
  Python based on PyCIM with CIM 15

- Implemented CIM adapters:
  - Mapping of LV grid model in DGS format from DIgSILENT
  - Mapping of topological grid data derived from GIS system
  - Mapping of asset data obtained from asset data bank

CIM mapping: Example





Source: http://cimug.ucaiug.org/Meetings/Europe2015/Documents/KURT1 CIM U Track II Network Modelling - foundational EQ - OGS.pdf

CIM mapping: Example

- First step:

Specification of the required CIM classes

Physical Components	DGS Components	CIM Instances	CIM Instances with logic	CIM Instances for additional parameters
General Info	General	_	_	
	ElmNet	1- EquivalentNetwork	_	
Synchronous Generator	ElmSym (row)	1. GeneratingUnit 2. SynchronousMachine	BaseVoltage, VoltageLevel	
	TypSym (row)	_	_	
Breaker	ElmCoup (row)	1. Breaker	BaseVoltage, VoltageLevel	Asset
	TypSwitch (row)	_	_	
Current Transformer	StaCt (row)	1. CurrentTransformer 2. Measurement	_	
Power Transformer	ElmTr2 (row)	1. PowerTransformerEnd 2. RatioTapChanger 3. TransformerMeshImpedance 4. TransformerCoreAdmittance 5. PowerTransformerEnd 6. PowerTransformer	BaseVoltage, VoltageLevel BaseVoltage, VoltageLevel	Asset
	TypTr2 (row)	_	_	
Load	ElmLod (row)	1. EnergyConsumer 2. LoadResponseCharacteristic	BaseVoltage, VoltageLevel	
	TypLod (row)	_	_	
Line	ElmLne (row)	1. ACLineSegment 2. Line	BaseVoltage, VoltageLevel	ConductorInfo WireArrangement WireType
	TypLne (row)	_	_	
Node	ElmTerm (row)	1. ConnectivityNode	if the node is a bus bar: BusbarSection Terminal BaseVoltage, VoltageLevel	
Connection	StaCubic (row)	1. Terminal	Terminal for a bus bar	
Graphical Info	IntGrf	Not considered yet		
	IntGrfcon			
	IntGrfnet			

CIM mapping: Example

- Second step: mapping of element's parameters

DGS - CIM				
DGS Components	**DGS Parameter**	**Mapping Type**	**CIM Parameter**	**CIM Classes**
StaCubic (row of a table)	ID	Direct	mRID: String	Terminal
	loc_name	Direct	name: String	
	chr_name	Not mapped		
	obj_bus	Indirect	aliasName: String	
	fold_id	Direct	mRID: String	Associated ConnectivityNode
	obj_id	Direct	mRID: String	Associated ConductingEquipment

Mapping of CIM Terminal: relatively direct mapping

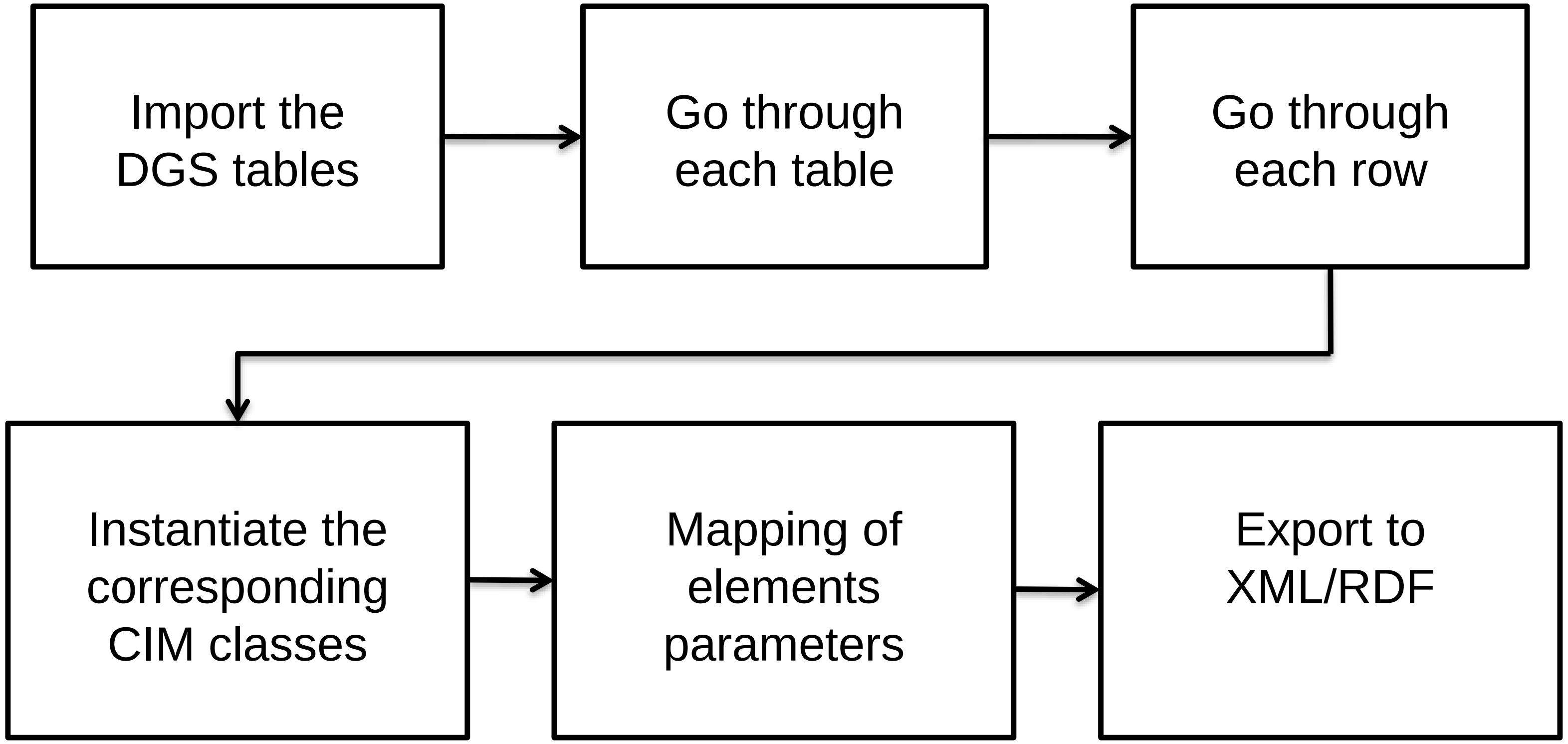
CIM mapping: Example

- Second step: mapping of elements parameters
  - Mapping of transformer
    - Mainly indirect mapping
    - Mathematical representation is different:
      - in CIM (R, X, C)
      - in DGS (open circuit and short circuit test parameters)

DGS - CIM				
DGS Komponen	DGS Parameter	Kommentare	CIM Parameter	CIM Klassen
ElmTr2	ID(a:40)	keine direkte Mapping	mRID: String	PowerTransformer
	loc_name(a:40)	direkte Mapping	name: String	
	typ_id(p)	keine direkte Mapping	aliasName: String	
	outserv(i)	direkte Mapping	normallyInService: Boolean	
	i_auto	nicht relevant/ nicht gefunden		
	chr_name(a:20)	nicht relevant/ nicht gefunden		
	cgnd_h	nicht relevant/ nicht gefunden		
	cgnd_l	nicht relevant/ nicht gefunden		
	ratfac	nicht relevant/ nicht gefunden		
	fold_id(p)	keine direkte Mapping	mRID: String:	Asset
	sernum(a:20)	direkte Mapping	serialNumber: String	
	constr(i)	direkte Mapping	installationDate: String	
	nntap(i)	direkte Mapping	normalStep: Integer	RatioTapChanger
	ntrcn(i)	direkte Mapping	regulationStatus: Boolean	
	i_cont(i)	direkte Mapping	tculControlMode: TransformerControlMode	
	t2ldc(i)	nicht relevant/ mögliche Mapping	"1"	
	ilcph(i)	nicht relevant/ mögliche Mapping	"6"	
	imldc(a)	nicht relevant/ mögliche Mapping	"V"	
	uset_mode(i)	nicht relevant/ mögliche Mapping	"0"	
	usetp(r)	direkte Mapping	neutralU: Voltage	
	Tctrl(r)	direkte Mapping	supsequentDelay: Seconds	
	ildc(i)	nicht relevant/ mögliche Mapping	"0"	
	usp_low(r)	keine direkte Mapping	limitVoltage: Voltage	TapChangerControl
	usp_up(r)			
TypTr2	ID(a:40)	nicht relevant/ mögliche Mapping	aliasName: String	PowerTransformer
	loc_name(a:40)	nicht relevant/ mögliche Mapping	name: String	PowerTransformerEnd (HV and LV)
	fold_id(p)	nicht relevant/ nicht gefunden		
	strn(r)	direkte Mapping	RatedS: Apparant Power	
	frnom(r)	keine direkte Mapping	Global Variable of CimGrid: frequency	
	utrn_h(r)	direkte Mapping	ratedU: Voltage	PowerTransformerEnd (high voltage)
	tr2cn_h(a:2)	nicht relevant/ mögliche Mapping	connectionKind: String	
	utrn_l(r)	direkte Mapping	ratedU: Voltage	PowerTransformerEnd (low voltage)
	tr2cn_l(a:2)	nicht relevant/ mögliche Mapping	connectionKind: String	
	nt2ag(r)	keine direkte Mapping	vectorGroup: String	PowerTransformer
	zx0hl_n(r)	nicht relevant/ nicht gefunden		
	tap_side(i)	keine direkte Mapping	aliasName: String	RatioTapChanger
	dutap(r)	direkte Mapping	stepVoltageIncrement: PerCent	
	phitr(r)	keine direkte Mapping	0	
	nntap0(i)	direkte Mapping	neutralStep: Integer	
	ntpmn(i)	direkte Mapping	lowStep: Integer	
	ntpmx(i)	direkte Mapping	highStep: Integer	
	manuf(a:20)	nicht relevant/ mögliche Mapping	aliasName: String	Asset
	chr_name(a:20)	nicht relevant/ mögliche Mapping	name: String	
	uk0tr(r)	short circuit test	r: Resistance	TransformerMeshImpedance
	ur0tr(r)		x: Reactance	
	uktr(r)			
	pcutr(r)			
	pfe(r)	open circuit test	b: susceptance	TransformerCoreAdmittance
	curmg(r)		g: Conductance	

CIM mapping: Example

- Contents of the CIM-DGS Adapter



ESOSEG
Environment for Simulation, Operation and Optimization of Smart Energy Grids

Next steps

- Further development of the CIM mapping to reach a standardized adapters
- Transition from hand-implemented adapters to model-based generation using Epsilon (Eclipse Modelling Framework)
- Moving from CIM 15 based on PyCIM to the up-to-date CIM version
- Further development of the domain model, which reflects the needs and requirements of different DSOs
- Further development of the ESOSEG platform e.g. the provided flow control mechanism, security, data transportation
- Development of new use-cases, e.g. installation and operation of a grid battery system, grid optimization through switches, etc.
- Field test of the ESOSEG framework by our participating grid operators
- Project Goal: Distribution of the platform to the open-source community

ESOSEG
Environment for Simulation, Operation and Optimization of Smart Energy Grids

Open discussion

- How to validate the grid modelling in CIM? Is there a standard grid model in CIM format available for comparison?
- How to depict the hierarchy of grid components in XML/RDF formats (e.g. according to voltage levels)?
- Are there default values of CIM parameters (e.g. if a value is not available)?
- Which load flow solvers using the CIM formats are available in the market?
- How to serialize CIM compound classes to RDF/XML (e.g. StreetAddress)?

Open discussion

- How to associate a ConnectivityNode to a PositionPoint (to assign a geographical position)?
- How to associate a GeneratingUnit to a Terminal?
- Is there a converter from RDF to XDF format?
- How to map physical parameter of PV GeneratingUnit (e.g. tilt angle, orientation)?
- Is there a possibility to modify type of input parameters (e.g. inputting *S*, *cosφ* to EnergyConsumer instead of *p*, *q*)

ESOSEG
Environment for Simulation, Operation and Optimization of Smart Energy Grids

www.esoseg.de

Ulm University of Applied Siences Eberhard Finckh Str. 11 D-89075 Ulm Basem Idlbi: idlbi@hs-ulm.de Christoph Kondzialka: kondzialka@hs-ulm.de Matthias Casel: casel@hs-ulm.de	SEKAS GmbH Baierbrunner Str. 23 D-81379 Munich Markus König: markus.koenig@sekas.de
Technical University Munich Boltzmannstr. 3 D-85748 Garching Vasileios Koutsoumpas: koutsoum@in.tum.de Dominik Ascher: ascher@in.tum.de	Stadtwerke Ulm/Neu-Ulm Netze GmbH Karlstraße 1-3 D-89073 Ulm Florian Meier: florian.meier@ulm-netze.de