IEC Common Information Model & Končar PowerCIM platform

KONČAR
Inspired by challenge

Hedge-IoT 2024 December

1

Končar

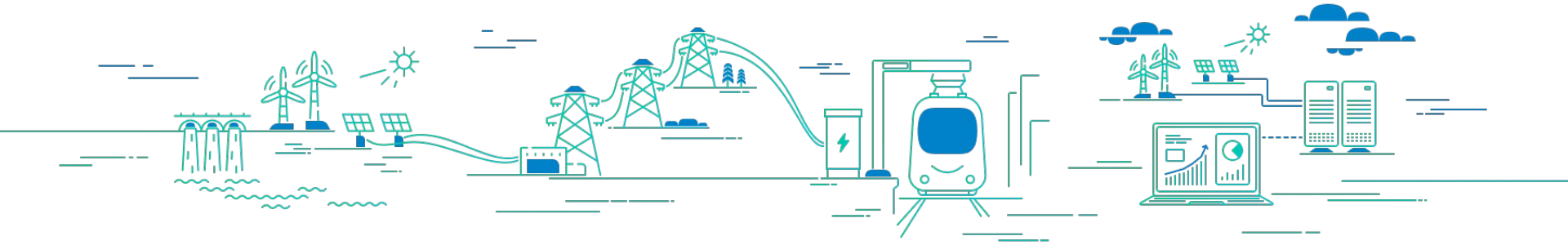
Končar group divisions

Power Generation

Power Transmission & Distribution

Rail Solutions & Infrastructure

Digital Solutions & Platforms



Digital Solutions & Platforms

- Control Center projects (Croatia, Albania, Slovenia, BiH)
  - TSO: HOPS, OST, ELES
  - DSO: HEP-ODS, ElektroPrimorska, EPHZHB
  - Gas: Plinacro, GPZ
  - Generation: HEP, KESH
- Solutions
  - Third party control center SCADA systems
  - EMS/MMS/CGMES platform NetVision
  - Substation SCADA ProzaNet/ProzaStation
  - River Basin Control, AGC & optimization

KONČAR

IEC TC57 Activity

- Since 2007:
  - WG 10; **WG 13; WG 14;** WG 15; **WG 16;**
- Solutions certified according to:
  - **CGMES** 2.4.15 (NetVision),
  - IEC 62443 (ProzaNet, LKKU, ProzaHAT, MARS),
  - IEC 62351 (ProzaHAT),
  - IEC 61870-5, IEC 61850 (ProzaNet, ProzaHAT)
  - …

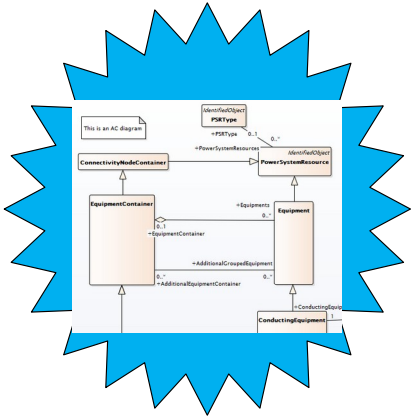
KONČAR

2

Common Information Model intro

KONČAR

IEC Common Information Model



Model Profile Serialization

KONČAR

CIM model

- Common information model for electrical network data exchange
  - Set of open standards for representing power system components
  - Used mostly for network model (metadata) and case (snapshot) data exchange
- Developed by IEC TC57 WG 13/14/16
  - Using Enterprise Architect tool
- CIM uses an **UML** based ontology
  - Classes, attributes and relations with multiplicity
  - Single class inheritance
  - Use of 128-bit UUID unique identifiers (MRID)
  - “metamodel”
  - Extendable by users
- Data encoded in **RDF** data structure
  - Class instance – attribute – value triplets
  - “model”

CIM data

- (Not only) object oriented model
  - Can be interpreted as a relational, document, graph (network topology) model
  - Includes geographic data, measurement snapshots, blobs (load curves etc...)
  - Every instance has a class and unique id (mRID UUID)
  - Objects correspond to physical elements (lines, transformers, switches...), organizational elements (station, region, voltage level), types, measurement locations etc...
  - Class hierarchy and attribute semantics prescribed by the basic CIM standard
  - Can be extended (both classes and attributes)

KONČAR

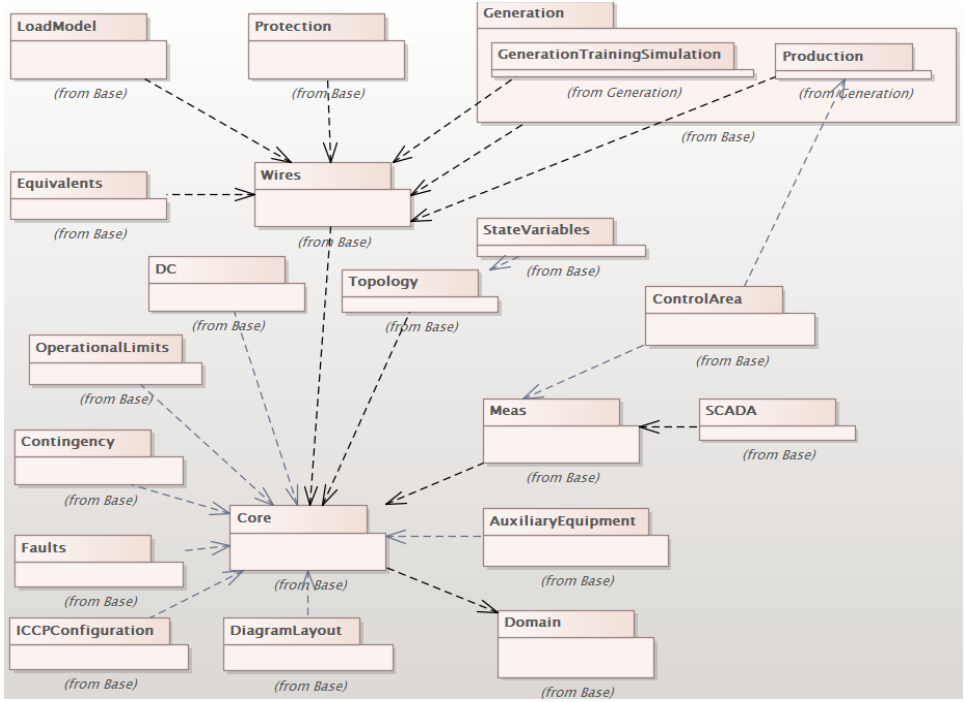
CIM data

- Serialization
  - XML, JSON-LD (future)
  - Simple listing of objects with attributes
  - Some attributes are of the mRID type and represent references/relations with other objects
- Profiles
  - Subset of classes and attributes needed for a specific use-case
- Uses
  - CRUD model exchange
  - CIM for messaging
  - CIM as an integration platform

KONČAR

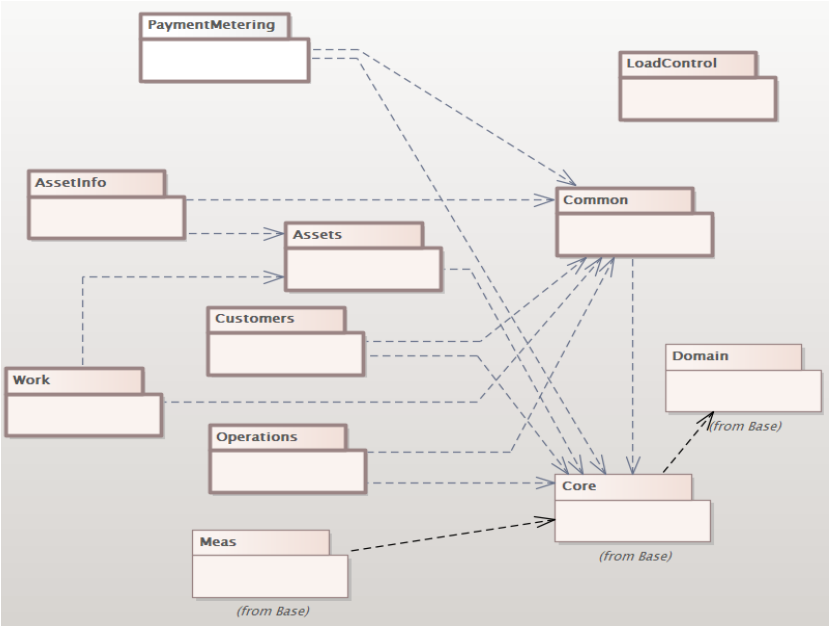
IEC TC57 WG13 – IEC61970 (TSO/network modelling/grid)

- Core CIM part
- Network model for basic network calculations (load flow, topology, short circuit...)
- Dynamic – element models for transient response modelling
- Meta – data exchange metadata, CRUD change modelling



IEC TC57 WG14 – IEC61968 (DSO/business integration/support)

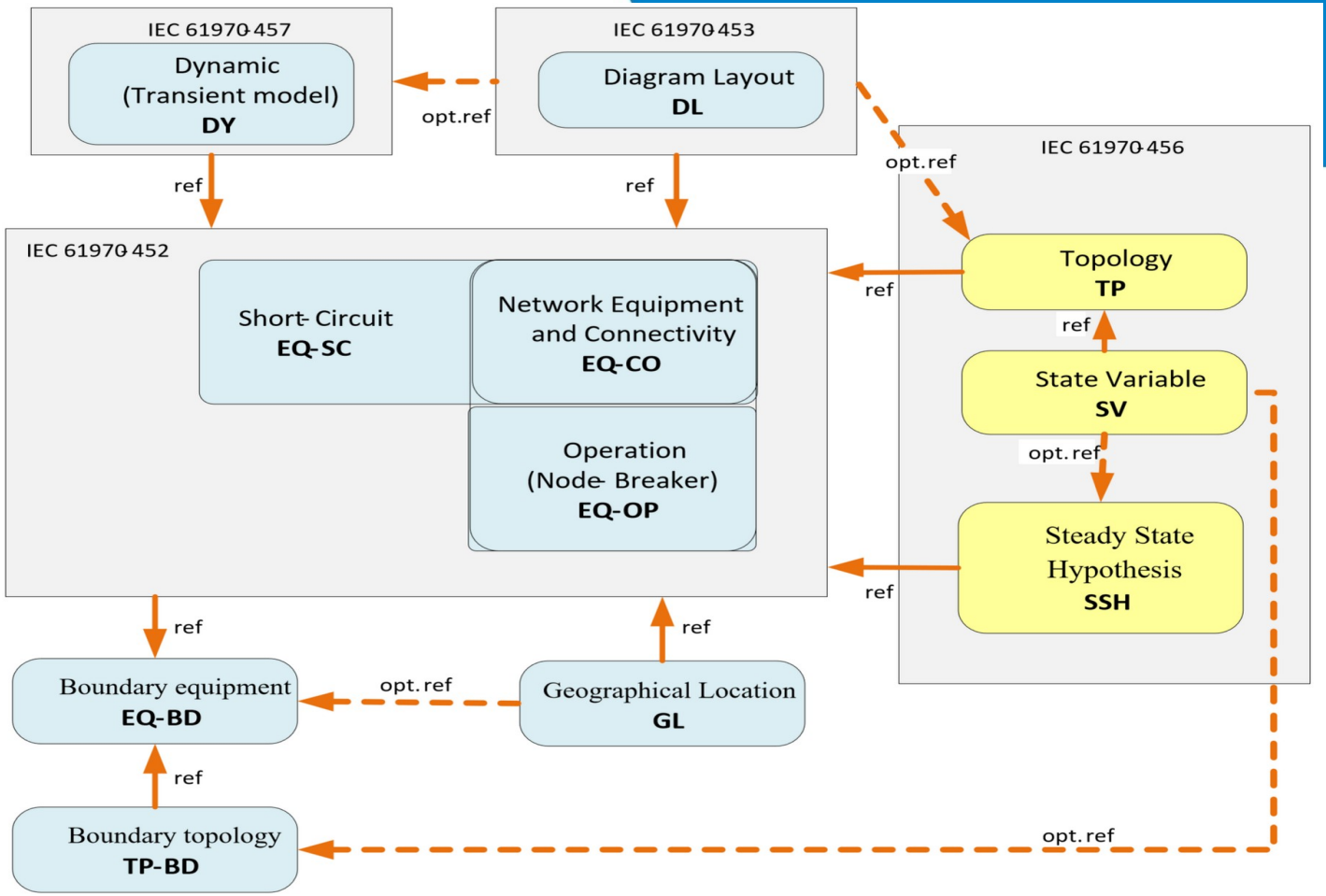
- 61968-3 Interface for network operations
- 61968-4 Interfaces for asset management
- 61968-5 Distributed energy optimization
- 61968-6 Interfaces for maintenance and construction
- 61968-8 Customer Operations
- 61968-9 Meter reading and control



CGMES

- Common Grid Model Exchange Standard
  - Concrete CIM profile for ENTSO-E use cases
  - Interface for data exchange between ENTSO-E TSOs
  - Mandatory for European TSOs
  - Enables data exchange and merging of individual TSO network models
- Applications dealing with:
  - power system data management
  - load flow
  - contingency analysis
  - short circuit calculations
  - market transparency
  - capacity allocation and congestion management
  - dynamic security assessment

CGMES


IEC 61970-457
Dynamic
(Transient model)
DY
opt.ref
IEC 61970-453
Diagram Layout
DL
opt.ref
IEC 61970-456
ref
ref
IEC 61970-452
Short- Circuit
EQ-SC
Network Equipment
and Connectivity
EQ-CO
Operation
(Node- Breaker)
EQ-OP
ref
Topology
TP
ref
State Variable
SV
opt.ref
Steady State
Hypothesis
SSH
ref
ref
ref
Boundary equipment
EQ-BD
opt.ref
Geographical Location
GL
ref
Boundary topology
TP-BD
opt.ref

Why CIM ?

- CIM models most elements and processes within electric energy sector
- CIM includes many engineer-years of built-in domain expertise
- Any integration of systems creates an implicit common information model
- CIM is extensible by users
- Enables interoperability with industry standard software
  - SCADA systems
  - Standalone network modelling software (PSS-E, NEPLAN, Digsilent...)
  - (some) GIS systems, workforce management, market systems...
  - TSO mandatory integrations with ENTSO-e software
- Enables modularity, phasing of changes and incremental system replacement

CIM profile

- Subset of complete CIM model for a specific use-case
  - List of included concrete classes
  - List of mandatory and optional attributes/relations per class
- RDF schema (**RDFS**) for IEC 61970 standard group
- XML schema (**XSD**) for IEC 61968 and 62325 standard group
- Additional constraints (natural language, OWL, OCL, **SHACL**)
- Profiles for equipment (EQ), topology (TP), load flow (SSH, SV), geographical layout (GL), diagram layout (DL), dynamics (DY)...

CIM serialization

- Standardized rendering of RDF model data into files/messages
- **CIM/XML**
  - Based on RDF/XML and RDFS
  - Unfortunately, not completely compatible with RDF/XML
  - W3C recommendations were not complete at the time
- **JSON-LD**
  - Not yet standardized

Important CIM concepts

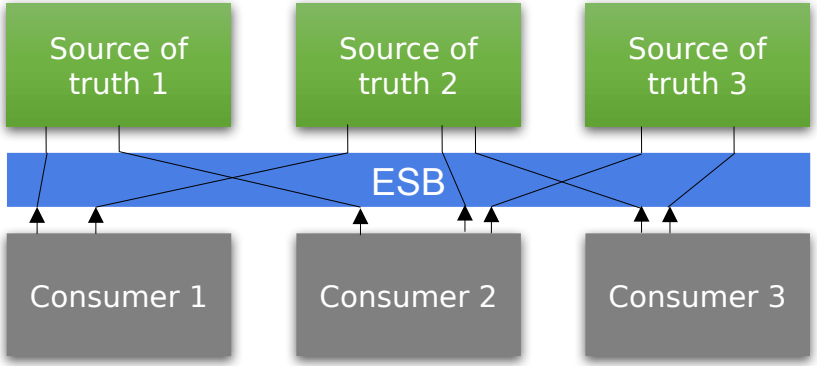
- Document metadata
  - Metadata for exchanged files/messages
  - Source, creation time, scenario time, description...
- IdentifiedObject
  - MRID (UUID) 128-bit unique decentralized object instance identifiers
- Names
  - Object identifiers in multiple contexts (other systems etc.)
- ConnectivityNode-Terminal
  - Universal network topology modelling
- PSR-Asset-AssetInfo
  - Correct modelling of assets separate from functions and characteristics
- External references, partial documents and model assembly

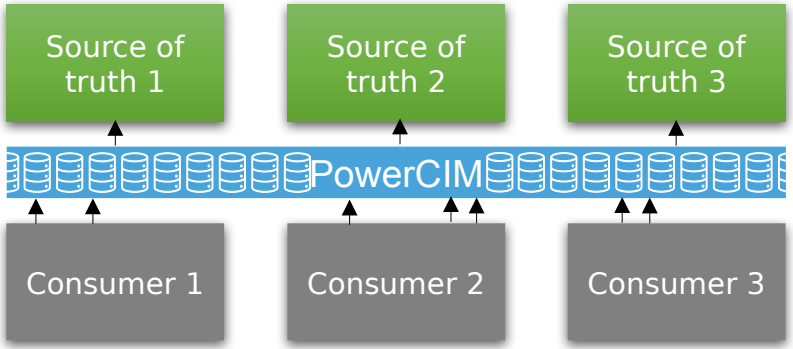
2

Why PowerCIM

Is CIM only for Data Exchange?

- File server ?
- Message queue ?
- Enterprise Service Bus ?
- Repository / Network Model Manager ?





KONČAR

Is CIM only for Data Exchange?

- Repository enables unified access to CIM data from different sources
- Single source of truth platform for network models
- Versioning
  - version control and change auditing
  - **fast** model version querying, retrieval and assembly
- Hierarchy
  - most (SCADA and other) systems organize data in hierarchies
  - although CIM doesn’t define a default hierarchy, one can be inferred
- Easier application development (DevEx)

Is CIM only for Data Exchange?

- CIM repository features vs. ESB
  - Persistence of exchanged data for later use / use by other systems
  - Single source of truth for network models
  - Model versioning
  - Multiple model branches and git-like branching
  - Hierarchies and multiple name cross-reference tables
  - Fast model querying and assembly
  - Validation
  - Profile based import/export
  - Easier integration of multiple systems
  - Platform for easy development of applications using network models

3

PowerCIM Inside

KONČAR

CIM EQ data

- Equipment data is slowly changing
  - Model with N objects will have ~N changes in X years
- Relatively small dataset
  - Millions of EQ objects can easily fit in RAM
  - Writes are rare – easy concurrency, easily scalable
- We store only model changes (“deltas”)
  - And assemble the needed model versions on demand
  - Record format and granularity use-case and profile specific
  - Same versioning system used for all use cases/profiles

Versioning

- Multiple repositories
  - multiple dependent or independent branches
  - multiple data sources (MAS), testing branches, various workflows…
- Bitemporal version parameters
  - Version sequence (Created time)
  - Valid time (Scenario time)
  - Allows for corrections in the past without destroying history
  - Allows for multiple future plans without overwriting
- No database record updates, all changes are appended

Considered tech stack

- Use case: fast model version retrieval and assembly
- Graph databases
  - in theory a perfect fit
  - complex topological processing not done on database layer
  - implementations not fully mature, specialized query languages
- Document databases
  - modern relational databases include JSON/XML storage types
- Naïve relational or object model built from CIM UML
  - a table per CIM class
  - huge generated schemas and object hierarchies
  - strict schema changes hard to manage

Chosen tech stack

- Persistence in **PostgreSQL**
  - **Json** for EQ data, blobs for SSH/SV…
- PowerCIM core lib and server in **.Net Core**
  - Backend, memory cache and indexing
  - versioning and model assembly queries
  - import/export, validation and other custom modules
- **REST API** for access
  - high performance streaming of queried model data
- PowerCIM web viewer (**React, MapLibre**)

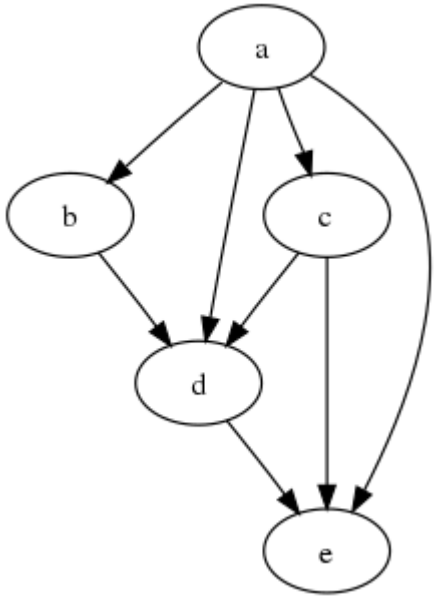
PowerCIM concepts

- Multiple sources -> **multiple branches** (repos)
- Small deltas -> **store only deltas/changes** (dataset)
- Store data according to type (EQ, snapshots, geo, DL...)
- Plans, corrections -> **bitemporal versioning**
- **Relational database** for persistence
- **In memory** storage for querying and serving data
- Transparent changing of CIM version (16, 17, 18...)
- **Schemaless storage** with late validation

KONČAR

Multiple branches

- Versioning + **branching**
- DAG (directed acyclical graph)
- As used in version control systems (Git, Mercurial...)



KONČAR

KONČAR PowerCIM

- Digital platform for data exchange of IEC CIM data
- High performance
- Advanced versioning and filtering
- Streaming REST API

- **Multibranch bitemporal data storage of CIM data**

- components
  - PowerCIM Postgres database
  - PowerCIM core library
  - PowerCIM REST server
  - PowerCIM frontend
  - PowerCIM import/export

KONČAR

PowerCIM components

- PowerCIM core lib
  - .NET library
  - import/export (-552, PowerCIM DB, SQL, csv...)
  - inmem storage of versioned data
  - querying and model assembly
  - SHACL validation
  - Future: topology processing, additional validations and algorithms

- PowerCIM backend server
  - REST server for querying CIM data
  - Data backend for PowerCIM viewer web app
  - And other (web?) applications that don't directly use the CIM standard (-552 files)

PowerCIM komponente

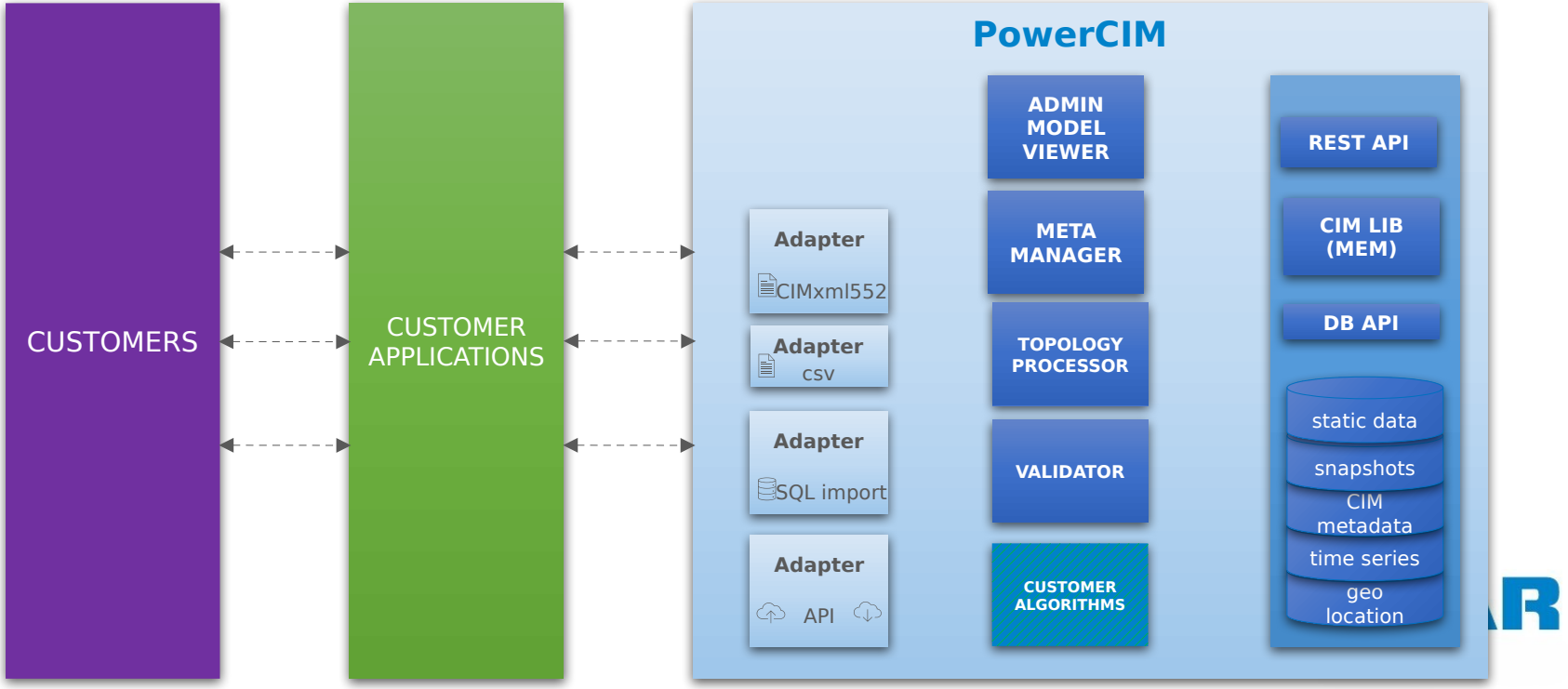
- PowerCIM Postgres DB
  - Persistence of PowerCIM data
  - Initial implementation of versioning
  - Too slow, query optimiser unpredictable
- PowerCIM frontend
  - React web app
  - CIM data browsing
  - Test frontend for PowerCIM
  - Read only, not a data engineering application
  - Data is edited in source systems (SCADA, GIS, ERP...)

KONČAR

PowerCIM Versions

Version	Features	Technologies	In use
1.0	bitemporal, multi hierarchy	SQL (tables)	HOPS 2015-2017
2.1	bitemporal, single hierarchy	SQL (json)	HOPS 2017-
3.3	+ DependentOn	SQL (json)	2022 - ATTEST, Elektra Zagreb
4.0	+ in-mem + CIM version aware + web viewer	C#, REST, React	2024 - HedgeIOT, ODS

Application platform



PowerCIM Projects

- HOPS (Croatian TSO) CIM export
  - Export from Hitachi Network manager, Končar NetVision EMS template
- HEP ODS ADMS migration
  - Smallworld GIS→PowerCIM→Hitachi NM ADMS
- HEP ODS CIM PowerCIM repo
  - Platform for internal app development
- ATTEST
  - TSO/DSO integration and model exchange
- Hedge-IoT
  - ElektroGorenjska GIS+DTR+weather->PowerCIM
- OPFP RFI proposal
  - HOPS optimal power flow with FER (Faculty of electric engineering and computing) Zagreb providing OPF algorithms

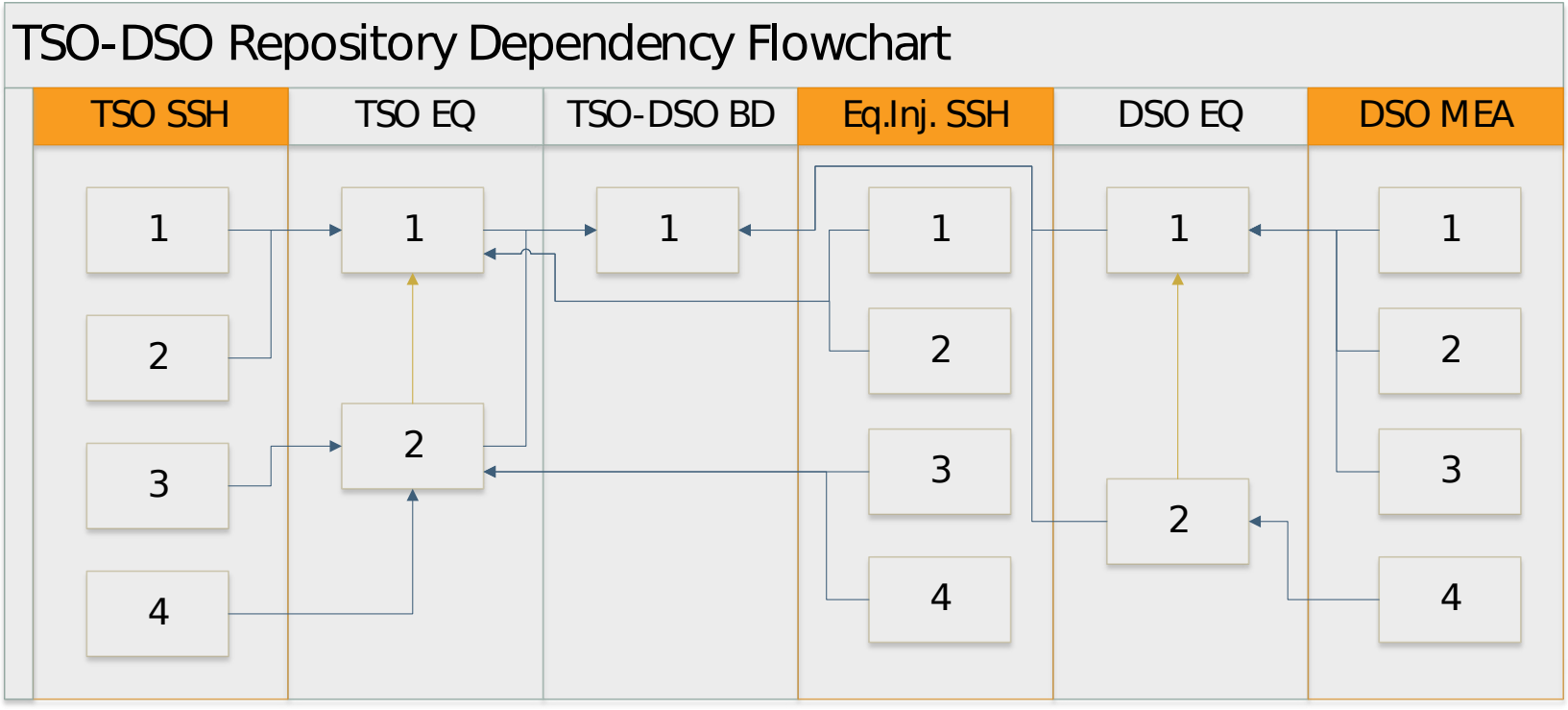
4

PowerCIM Use

KONČAR

Multiple Repositories

EU project ATTEST



ENTSO-E SSH Assembly Concept

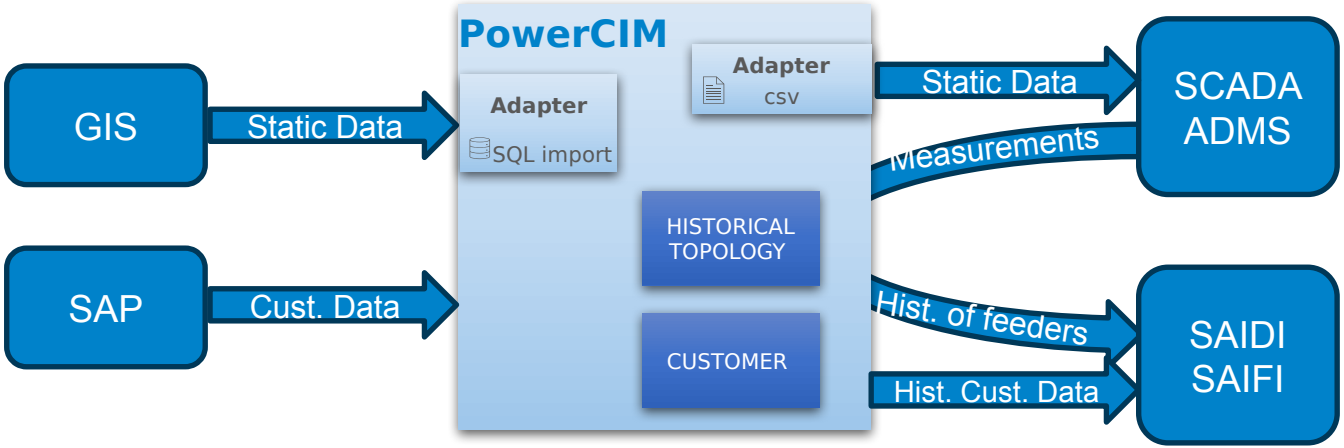
TSO1-TSO2 BD	TSO2-TSO3 BD	…
May-1@May-1	May-1@May-1	

TSO1 EQ	TSO2 EQ	TSO3 EQ	…
May-1@May-1	May-1@May-1	May-1@May-1	
May-15@May-15 03h			

TSO1 SSH D-1	TSO2 SSH D-1	TSO3 SSH D-1	…
May-14@May-15 01h	May-14@May-15 01h	May-14@May-15 01h	
May-14@May-15 02h	May-14@May-15 02h	May-14@May-15 02h	
May-14@May-15 03h	May-14@May-15 03h	May-14@May-15 03h	

SSH D-1
May-14@May-15 01h
May-14@May-15 02h
May-14@May-15 03h

PowerCIM Workflow
DSO Example



KONČAR

5

PowerCIM Demo

KONČAR

PowerCIM queries

- Basic version query is (repo, versionid, valid time)
- Response
  - snapshot, difference, all changes
- Additional filters/options
  - CIM classes (inheritance aware)
  - names
  - mRIDs
  - parent mRIDs (direct children)
  - ancestor mRIDs (all descendents)
  - dependent models on/off

KONČAR