Architectural Elements Categorization

Smart Grid Architecture Committee (SGAC) Architecture Development Team

Release 0.9

Smart Electric Power Alliance

Proceed With Caution

- The following material is offered as **guidance** for practitioners involved in Enterprise Architecture.
- This is not a tutorial and the assumption is made that the reader has at least a basic understanding of enterprise architecture frameworks and semantic information concepts.
- Background information on enterprise architecture and semantic frameworks, including TOGAF, Zachman, SOA, and OWL, is widely available and as near as a Google query.

Agenda

Smart Electric Power Alliance

- Background
- Key Contributors
- Problem Statements
- Objective/Goal
- Contributions
- Data Reduction/Normalization Process
- Architectural Element Categorization Guidance
- Neutral Concepts Model Approach
- Potential Future Phase

Background

Smart Electric Power Alliance

This effort advances smart grid / grid modernization architecture development efforts

- Built upon earlier smart grid architecture efforts: NIST Framework and Roadmap for SG Interoperability Standards, SGAC smart grid architectural conceptualization and gathering of architectural elements, GWAC Interoperability Layers, and SGAM development and release
- Using a methodology from common Architecture Frameworks: TOGAF, Zachman, and SOA

Key Contributors

Smart Electric Power Alliance

- Chair (2014-present): Dr. Elizabeth Sisley (Calm Sunrise Consulting)
- Dr. Steven Ray (Carnegie Mellon University)
- Ron Cunningham (AEP)
- Dr. Gerald Gray (EPRI)
- Stephan Amsbary (Emeritus Member)
- John Ruiz (Emeritus Member/Chair)

Problem Statements

IT/OT Projects:

- If fortunate, have access to prior/related projects' artifacts to work from, but many times start from scratch when updating existing system or application architecture/designs
- Deal with business, system, application, terms and definitions differently:
    - Assume that named terms/vocabularies used in industry or found in standards are interpreted the same across all users, groups, organizations, nations, or
    - Define their own Terms/Definitions without reuse of existing company or industry terms/definitions that are commonly accepted, or
    - Discover after the fact, what are in reality the same thing, are named differently across users, groups, organizations, nations, or
    - Project-defined context and abstraction levels of architectural terms and elements are not the same from one system-application project to the next, or
    - Struggle to gain understanding when integrating a project's application(s), data, and messaging with another set of applications, data, and messaging,
- End up identifying/naming/defining their own set of Actors, ServiceCompositions (~roles), Services used to document business scenarios, use cases, and stories

Objectives/Goals

- Provide Guidance/Examples on how to align from one project’s set of architectural/business elements to another. Alternately encourage reuse of a reduced number of sets of architectural/business terms via understanding how a setA-element relates to a setB-element
- Build up a common set of architectural/business elements and the relationships between them~~,~~ that can be used, interpreted, understood, and modeled the same way across projects and organizations
- Reduce the time/effort of each project’s re-identifying/re-naming/re-defining Actors, ServiceCompositions (~roles), Services, relationships, and associated missteps from one integration to the next
- Help create a common vocabulary of terms and elements that is expected to make it easier to create new business use cases/stories that in turn are used to help identify the business requirements and services/capabilities that are used in project business scenarios and use case development, and advancing interoperability standards work

Contributions

Produced SGAC Architecture Element Metadata Model

- Clarified architectural abstraction term definitions, categorized gathered use case architectural elements, developed a Decision Tree to classify:
    - Abstraction levels: Conceptual, Logical, Physical, Implementation
    - Architectural element types: Service, ServiceComposition, Actor
- Leveraged architectural standards and business use cases for architectural elements from 18 organizations including, e.g., NIST, SGIP, NAESB, IEEE, IEC, and European Union (EU)
- Used Web Ontology Language (OWL) and various architectural tools
- Introduced “Neutral Concepts” multi-level model including the above metadata

Smart Electric Power Alliance

DATA REDUCTION/NORMALIZATION PROCESS

www.sepapower.org

Data Summary

SGAC gathered a set of Use Case elements from 18 organizations and performed initial categorization of elements to Conceptual, Logical, Physical architecture abstraction levels. Combined set was also categorized as:

- **~ 180 Actors**
- **~ 230 ServiceCompositions (~role)**
- **~ 800 Services**

Submitting Organizations

- NIST (AMI Systems, Electric Transportation, Distribution Grid Mgmt, WASA, Demand Response, Electric Storage, Conceptual Model, and Reference Diagram)
- NAESB
- CSWG NISTIR 7628
- IEEE P2030 Draft 3
- EIS
- EU Commission
- EU WGSP Vers 0.5
- SGAC Conceptual Arch
- IEC TC57-WG19 & TC8 (SyC) WG6 DER Actor/Roles
- EU SC-CG WP1
- ENTSO-E role model
- AhG Charging
- IEC61968 IRM
- IEC SG3
- GUC
- DKE Repository
- SMCG
- EG3

Data Reduction & Normalization Process

Smart Electric Power Alliance

For each named Element, in TopBraid Composer's (TBC) OWL model:

- Where there were multiple submissions, definitions were reviewed to normalize entries
    - When definitions were either identical or very similar, or one had a missing definition – source data was consolidated, and all references kept for traceability
    - A few elements remain ambiguous, and will need to be reviewed by the submitting organizations for determination
        - These are flagged via a SeeAlso relation in TBC: listing one or more likely related elements
        - A few have no similar entries but are missing definitions, these remain uncategorized
- Elements that are redundant or out of scope (such as 'ActorA' and 'ActorB') were marked as deprecated, but kept within the model

Elements were then categorized as described in the following slides

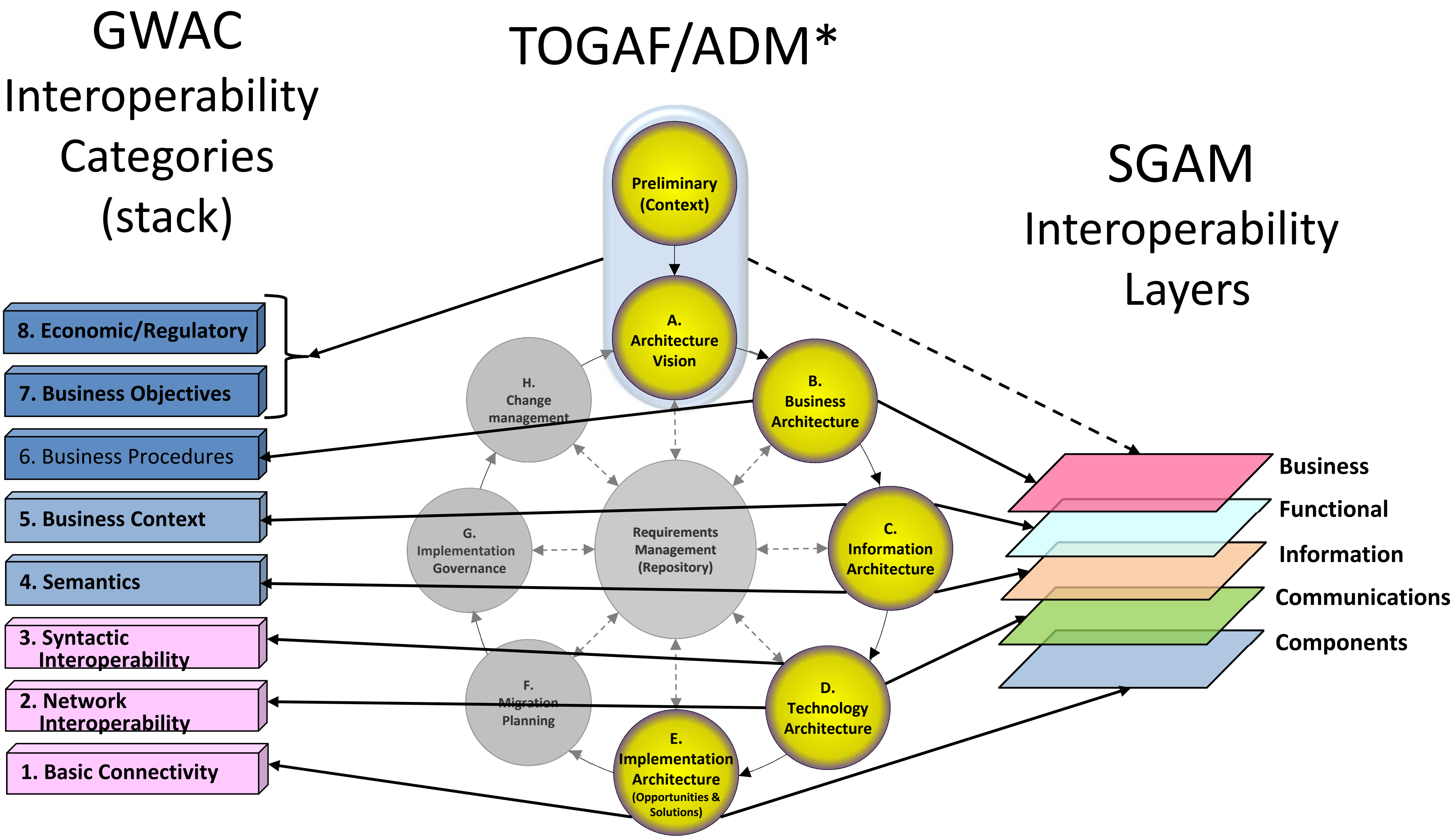
Smart Electric Power Alliance

ARCHITECTURAL ELEMENT CATEGORIZATION GUIDANCE

Architectural Abstraction Level and Element Categorization Guidance

- Architecture Abstraction Levels
  - Conceptual, Logical, Physical, Implementation
- Service Oriented Architecture (SOA) Elements
  - Categorization of Elements into Actors, Services, and ServiceCompositions (~roles)
- Categorization Guidance Decision Tree

Framework Mapping
GWAC & SGAM Alignment with TOGAF



GWAC Interoperability Categories (stack)

TOGAF/ADM*

SGAM Interoperability Layers

8. Economic/Regulatory

7. Business Objectives

6. Business Procedures

5. Business Context

4. Semantics

3. Syntactic Interoperability

2. Network Interoperability

1. Basic Connectivity

Preliminary (Context)

A. Architecture Vision

B. Business Architecture

C. Information Architecture

D. Technology Architecture

E. Implementation Architecture (Opportunities & Solutions)

F. Migration Planning

G. Implementation Governance

H. Change management

Requirements Management (Repository)

Business

Functional

Information

Communications

Components

* The Open Group Architecture Framework – Architecture Development Methodology (TOGAF/ADM)

SGAM 5x5x6 Model

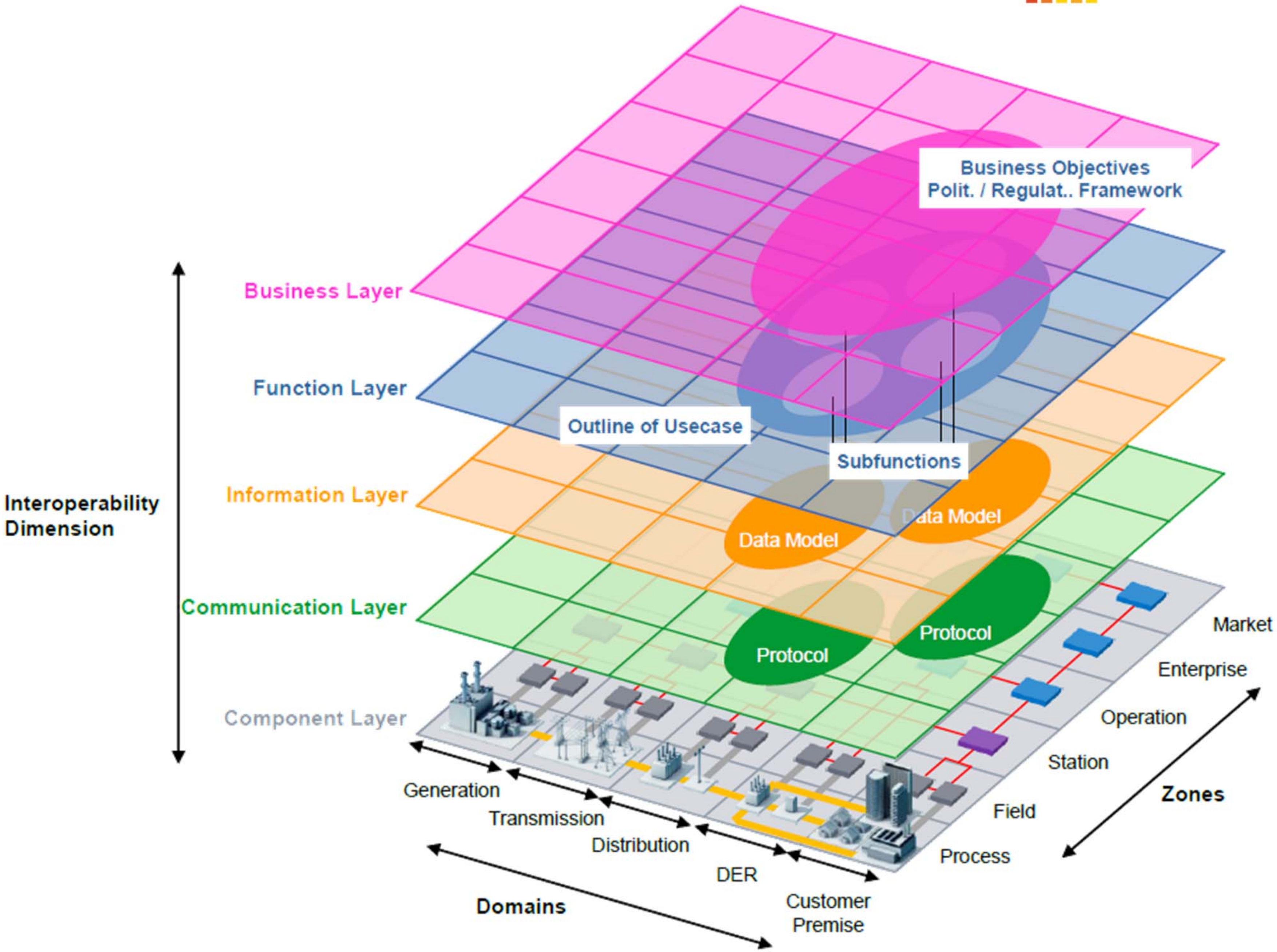
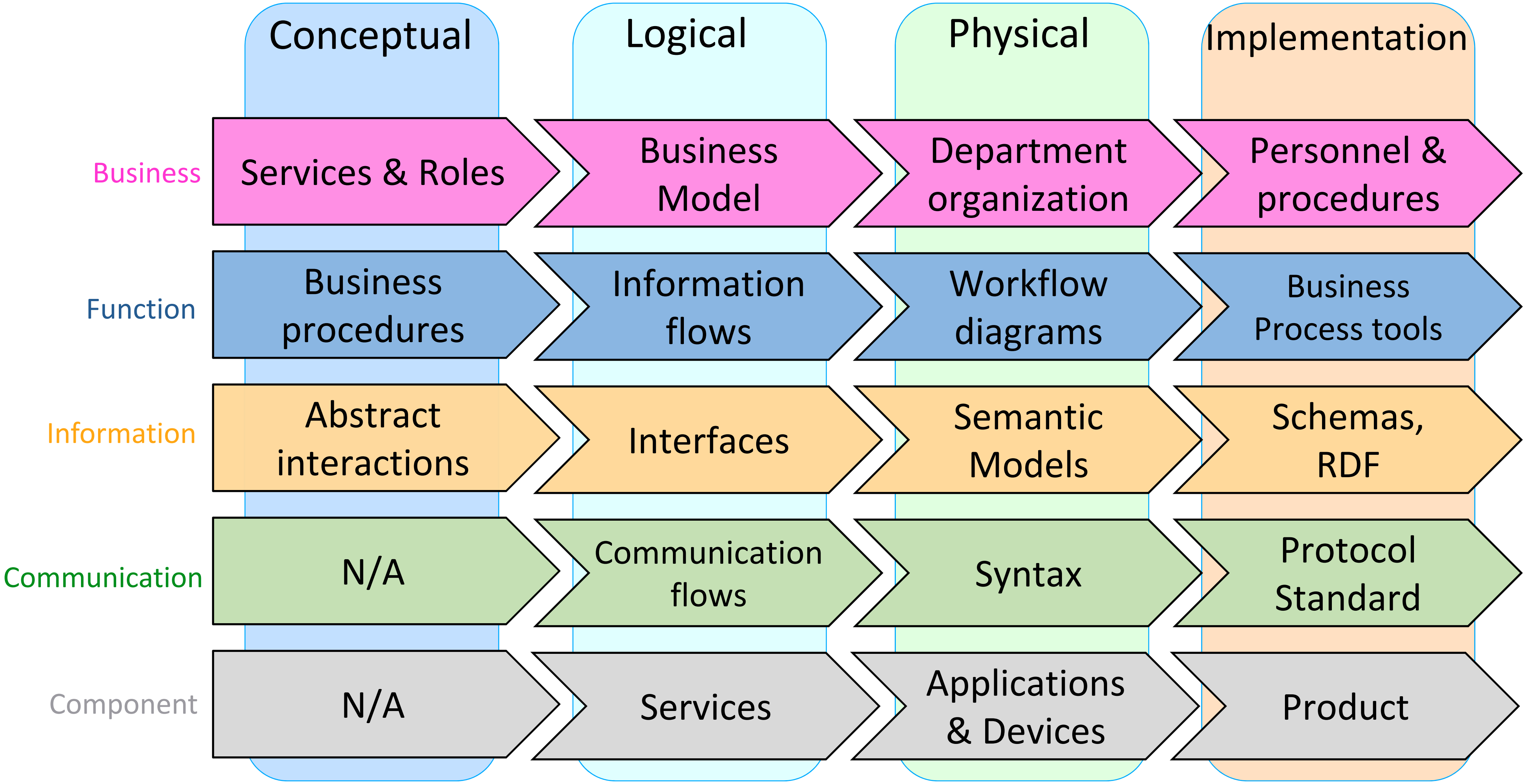


Figure 12: Layers of SGAM framework from SGCG Technical Report Reference Architecture for the Smart Grid v1.0 2012-03002

Architectural Abstraction Level mapping onto SGAM Interoperability Layers, with examples

Smart Electric Power Alliance



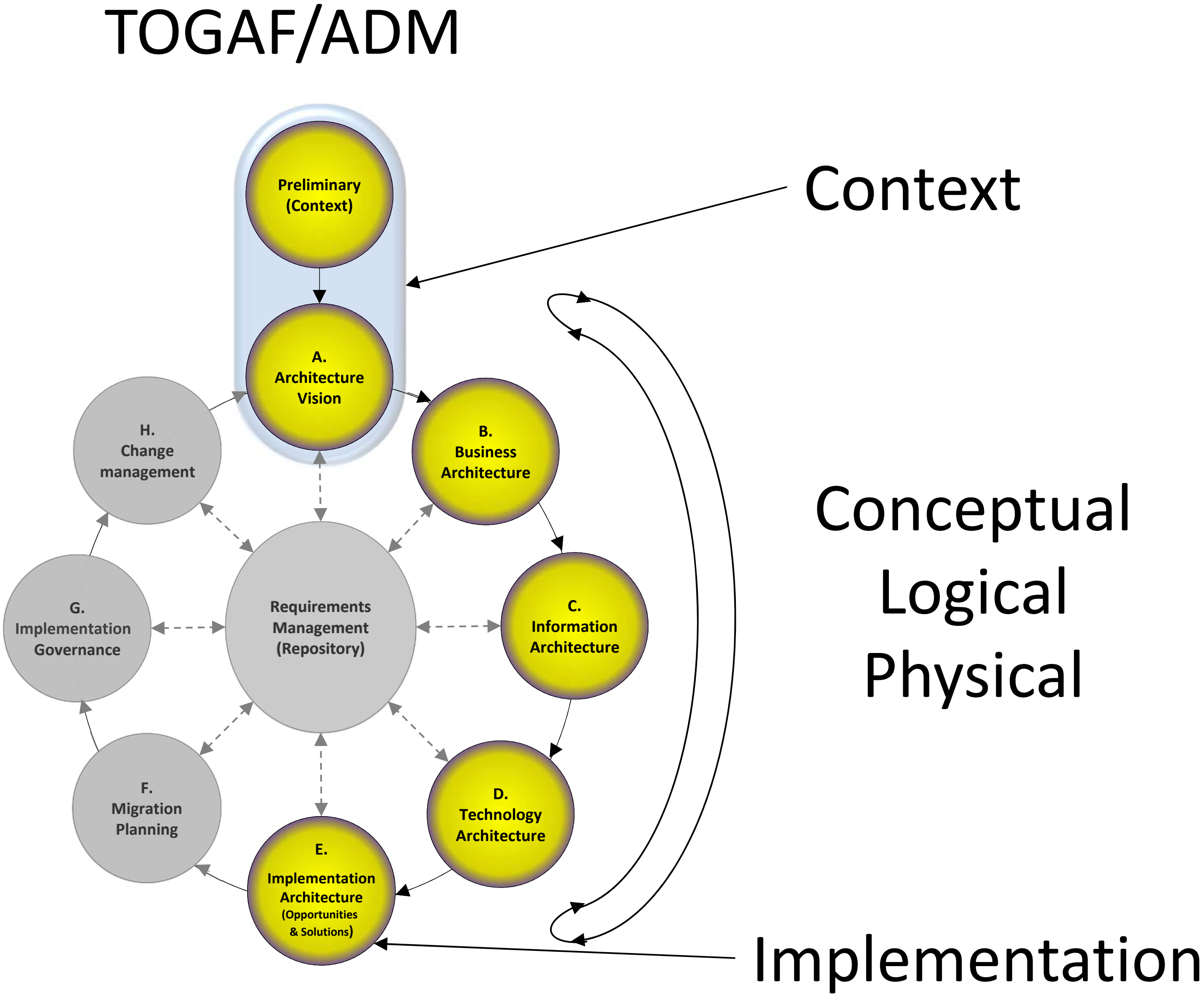
	Conceptual	Logical	Physical	Implementation
Business	Services & Roles	Business Model	Department organization	Personnel & procedures
Function	Business procedures	Information flows	Workflow diagrams	Business Process tools
Information	Abstract interactions	Interfaces	Semantic Models	Schemas, RDF
Communication	N/A	Communication flows	Syntax	Protocol Standard
Component	N/A	Services	Applications & Devices	Product

Work-in-progress

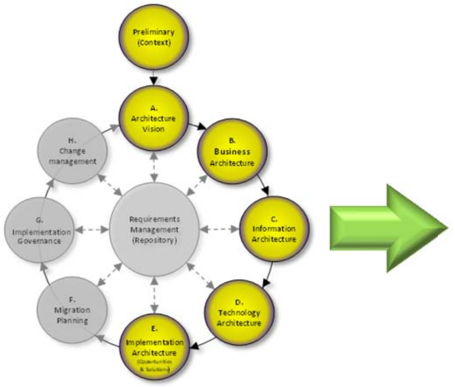
www.sepapower.org

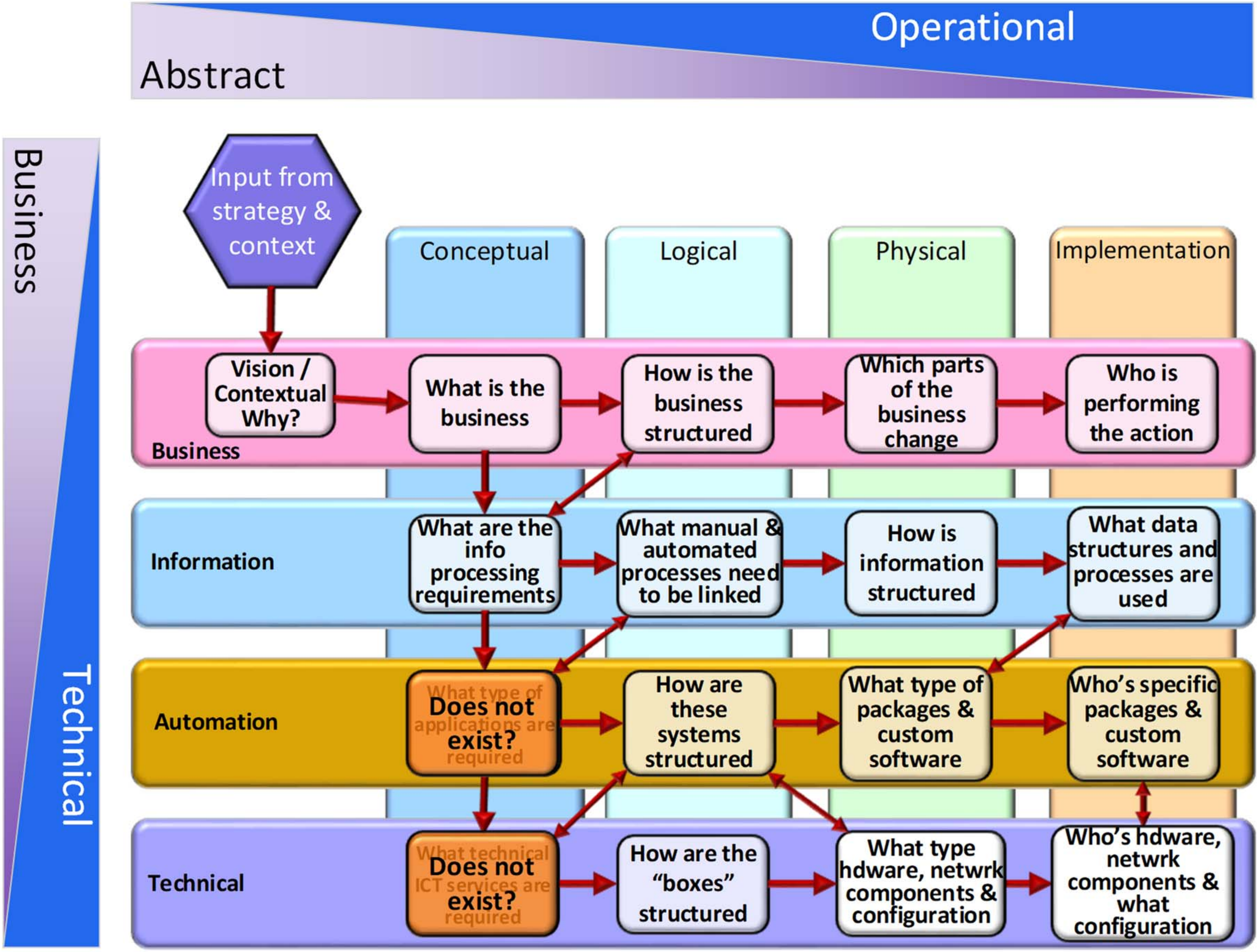
Architectural Abstraction Iterations within TOGAF

Smart Electric Power Alliance


TOGAF/ADM
Preliminary (Context)
A. Architecture Vision
B. Business Architecture
C. Information Architecture
D. Technology Architecture
E. Implementation Architecture (Opportunities & Solutions)
F. Migration Planning
G. Implementation Governance
H. Change management
Requirements Management (Repository)
Context
Conceptual
Logical
Physical
Implementation

Conceptual, Logical, Physical, Implementation applied to Zachman Levels


Preliminary (Context)
A. Architecture Vision
B. Business Architecture
C. Information Architecture
D. Technology Architecture
E. Implementation Architecture (Opportunities & Solutions)
F. Migration Planning
G. Implementation Governance
H. Change management
Requirements Management (Repository)

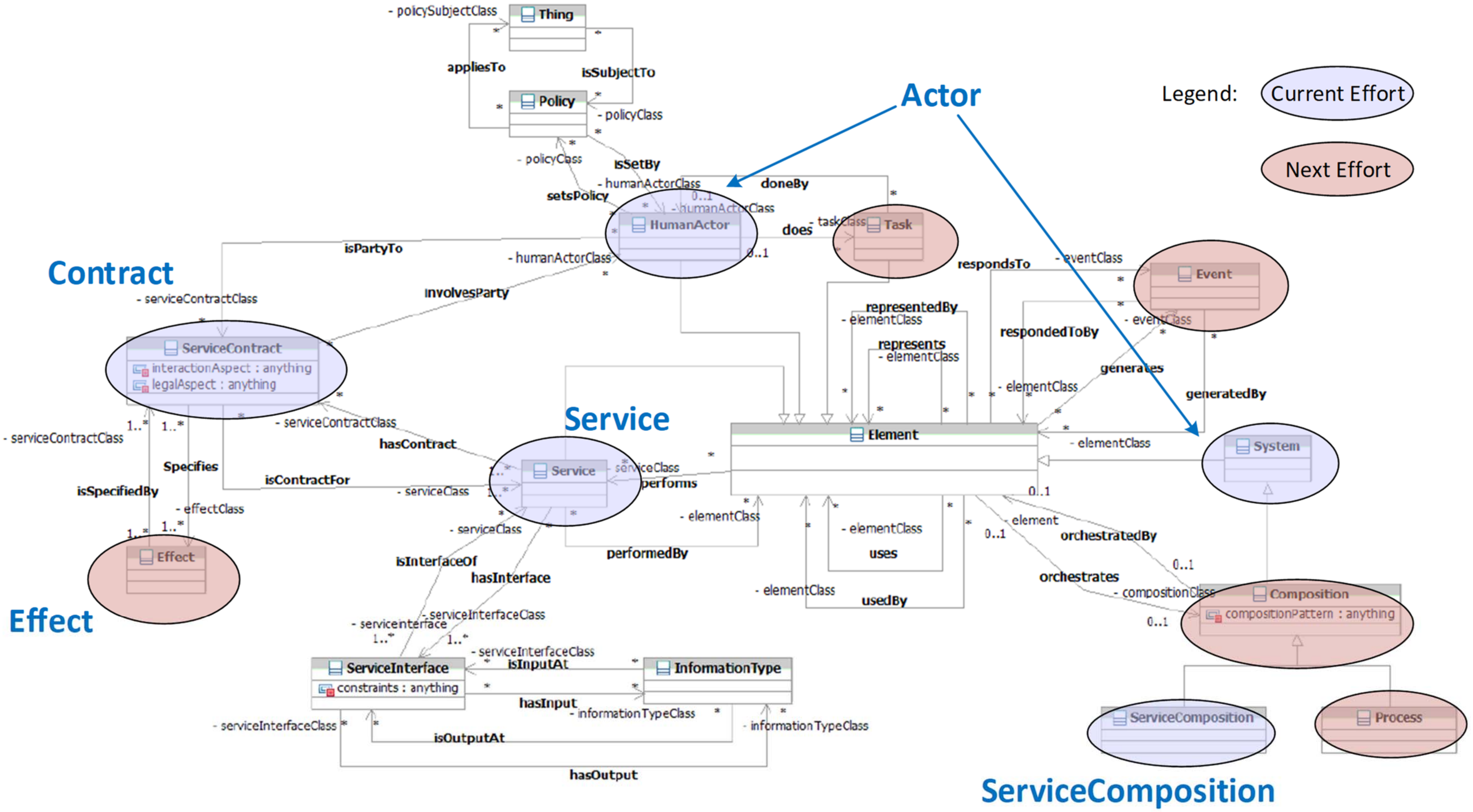


		Conceptual	Logical	Physical	Implementation
Business	Vision / Contextual Why?	What is the business	How is the business structured	Which parts of the business change	Who is performing the action
Information		What are the info processing requirements	What manual & automated processes need to be linked	How is information structured	What data structures and processes are used
Automation		What type of applications are required Does not exist?	How are these systems structured	What type of packages & custom software	Who's specific packages & custom software
Technical		What technical ICT services are required Does not exist?	How are the "boxes" structured	What type hdware, netwrk components & configuration	Who's hdware, netwrk components & what configuration

Note: TOGAF Phase C maps into Zachman Information and Automation

Full Open Group SOA Ontology Mappings to SGAC Arch.Dev. Effort

Smart Electric Power Alliance



www.sepapower.org

Conceptual Architecture Definition*

Smart Electric Power Alliance

Conceptual Architecture models the business as the organization owner conceptually thinks the business is or wants the business to be. (e.g. black box view)

<u>What are the services that are required to satisfy the business needs?</u>

- Business Services at the Conceptual abstraction level represent the goals of the owner
  - **WHAT** the organization *wants* to do, which is driven by its mission
  - **WHAT** it *needs* to do, which is often driven by regulations
- Conceptual Services have a lot of attributes
  - Service levels and other non-functionals
  - Specifications of functionality
- Simplest questions to identify Conceptual Elements:
  - WHAT are the essential elements within scope?
  - Which services are needed to provide to the outside world to fulfil its mission?
  - Which services from the outside world are needed to do that?

* Based upon Zachman Framework

Logical Architecture Definition

Logical Architecture models the technology-agnostic "systems" and representations of the business that are used to realize the services defined at the conceptual level. This includes business functions and actors. (e.g. white box view)

How are the business functions and actors ideally structured?

- **HOW** are basic elements related, according to the specific objectives and constraints of the architecture? (Logical Structure)
- What are the building blocks needed to run the business as defined in the Conceptual Architecture?
- Building blocks are technology independent services
- This is more about the interfaces than the technologies

Physical Architecture Definition

Physical Architecture further constrains the Logical architecture. This is the abstraction level where application and system architecture/design/technology choices are made that address and may constrain the business, application, and data model. This level represents HOW the Architecture ideally is being designed, excluding further Implementation level restrictions.

Addresses what actors, software and processes are necessary

- **HOW** elements of structure will be realized, according to architecture and design strategies and implications, and with **WHAT** will this be achieved?
- Describes physical implementation components. This set of components is a representation of components or building blocks an architect will use:
  - A person
  - A process
  - A piece of software
  - A piece of infrastructure
- Provides technical implementation specifications; the details needed for white box development processes (eg: ITIL, Rational Unified Process, Software Development Life Cycle, Agile), vendor solutions, and manual processes (TOGAF Phase E).

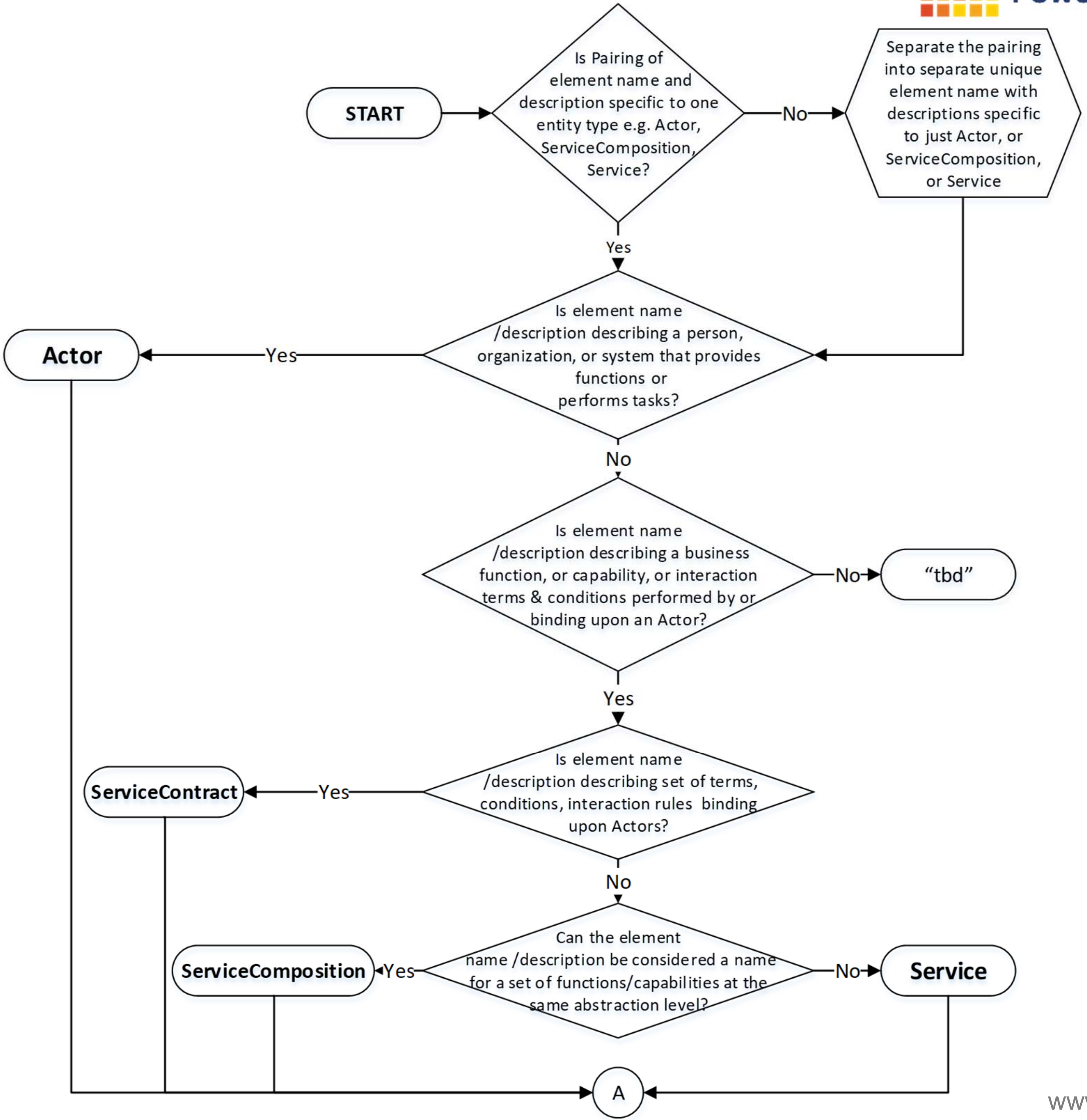
Categorizing the Data

SGAC modelers categorized the data in two ways:

- Type of Element
  - Actor
  - Service
  - ServiceComposition
- Architecture Abstraction Levels
  - Conceptual
  - Logical
  - Physical

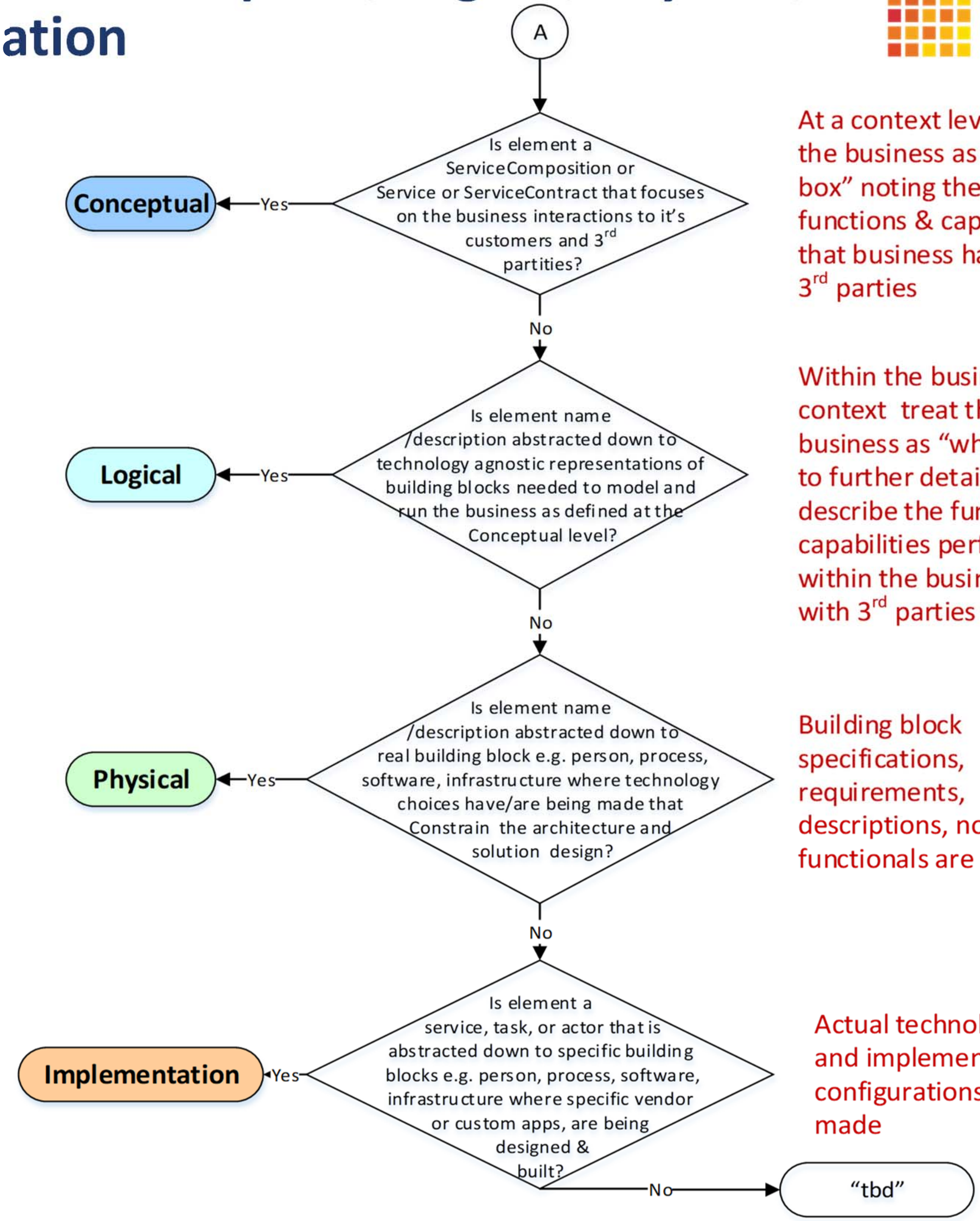
Using the Decision Tree on the following slides

Decision Tree: Actor, Service, ServiceComposition, ServiceContract


START
Is Pairing of element name and description specific to one entity type e.g. Actor, ServiceComposition, Service?
No
Separate the pairing into separate unique element name with descriptions specific to just Actor, or ServiceComposition, or Service
Yes
Is element name /description describing a person, organization, or system that provides functions or performs tasks?
Yes
Actor
No
Is element name /description describing a business function, or capability, or interaction terms & conditions performed by or binding upon an Actor?
No
"tbd"
Yes
Is element name /description describing set of terms, conditions, interaction rules binding upon Actors?
Yes
ServiceContract
No
Can the element name /description be considered a name for a set of functions/capabilities at the same abstraction level?
Yes
ServiceComposition
No
Service
A

www.sepapower.org

Decision Tree: Conceptual, Logical, Physical, Implementation



A

Is element a ServiceComposition or Service or ServiceContract that focuses on the business interactions to it's customers and 3rd partities?

Yes → Conceptual

No

Is element name /description abstracted down to technology agnostic representations of building blocks needed to model and run the business as defined at the Conceptual level?

Yes → Logical

No

Is element name /description abstracted down to real building block e.g. person, process, software, infrastructure where technology choices have/are being made that Constrain the architecture and solution design?

Yes → Physical

No

Is element a service, task, or actor that is abstracted down to specific building blocks e.g. person, process, software, infrastructure where specific vendor or custom apps, are being designed & built?

Yes → Implementation

No → "tbd"

At a context level, treat the business as "black box" noting the major functions & capabilities that business has with 3rd parties

Within the business context treat the business as "white box" to further detail and describe the functions & capabilities performed within the business and with 3rd parties

Building block specifications, requirements, descriptions, non-functionals are created

Actual technology and implementation configurations are made

www.sepapower.org

Semantic Architecture

Modeled in OWL

- Similar to Unified Modeling Language (UML), but with added expressiveness to capture constraints, arbitrary relationships and a supporting query language (SPARQL)
- Supports automated reasoning with popular reasoning engines (e.g. SPIN, OWLIM, Apache Jena)
- TopQuadrant's TopBraid Composer™ was used as the modeling environment, under license to SGIP

Meta Object Modeling Approach

- **Meta Classes -** Used to categorize classes
  - Actor
  - Service
  - Service Composition (Role)
  - Service Contract

  …with relations between them

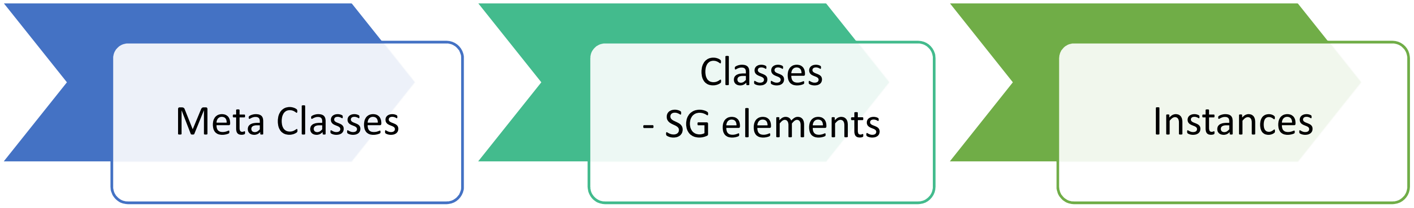
- **Classes -** Used to represent terms

  e.g.
  - Smart Grid Elements
  - Premise Management
  - Communication Network

  … categorized by the Meta Classes

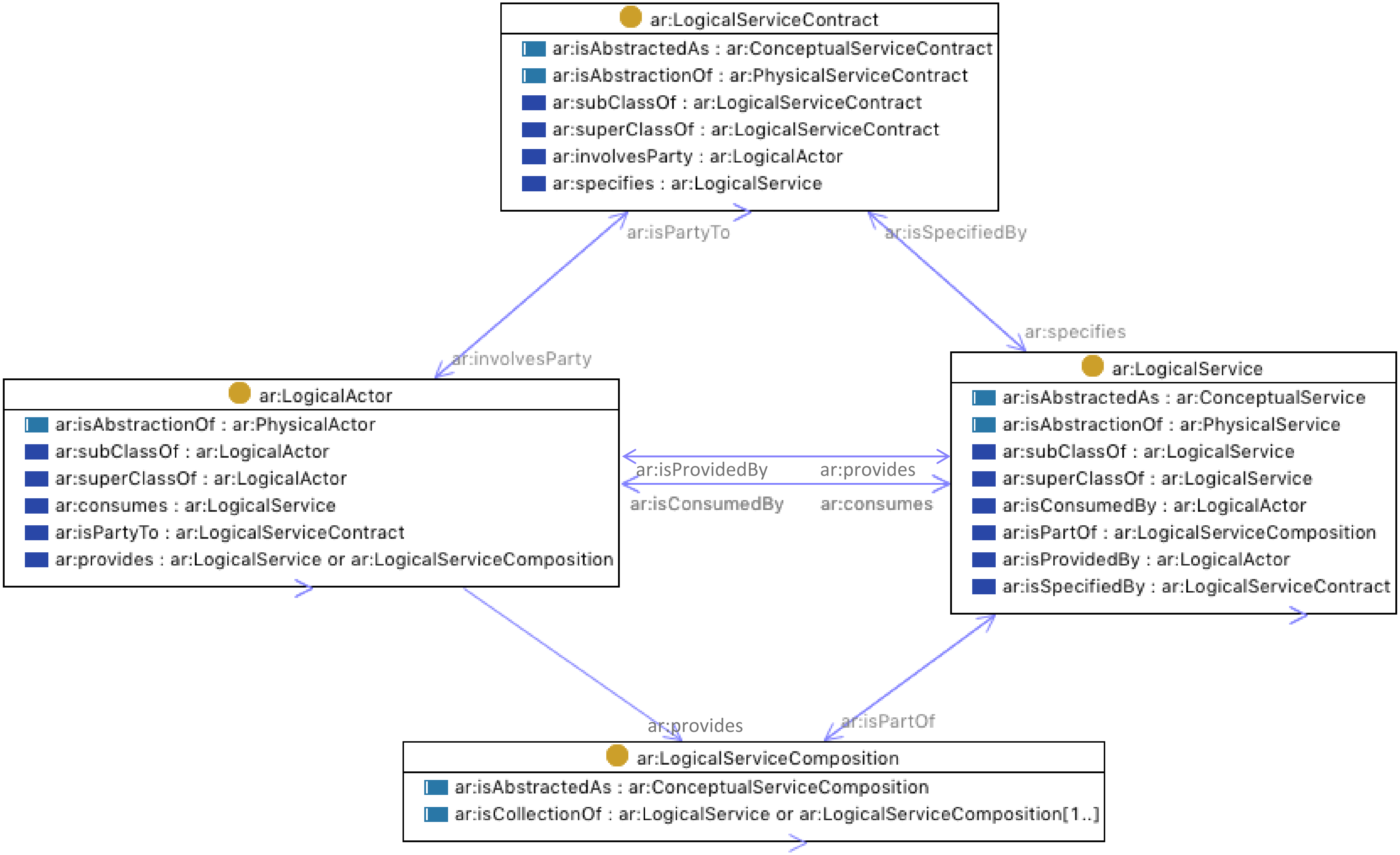
Instances - Used to inventory specific Systems and Services e.g.

- Meter #4127
- Web service implemented at http://fakesite.com/premise
- Network A at 129.6.24.159



Architectural “Meta Classes”

Logical Level



ar:LogicalServiceContract
- ar:isAbstractedAs : ar:ConceptualServiceContract
- ar:isAbstractionOf : ar:PhysicalServiceContract
- ar:subClassOf : ar:LogicalServiceContract
- ar:superClassOf : ar:LogicalServiceContract
- ar:involvesParty : ar:LogicalActor
- ar:specifies : ar:LogicalService

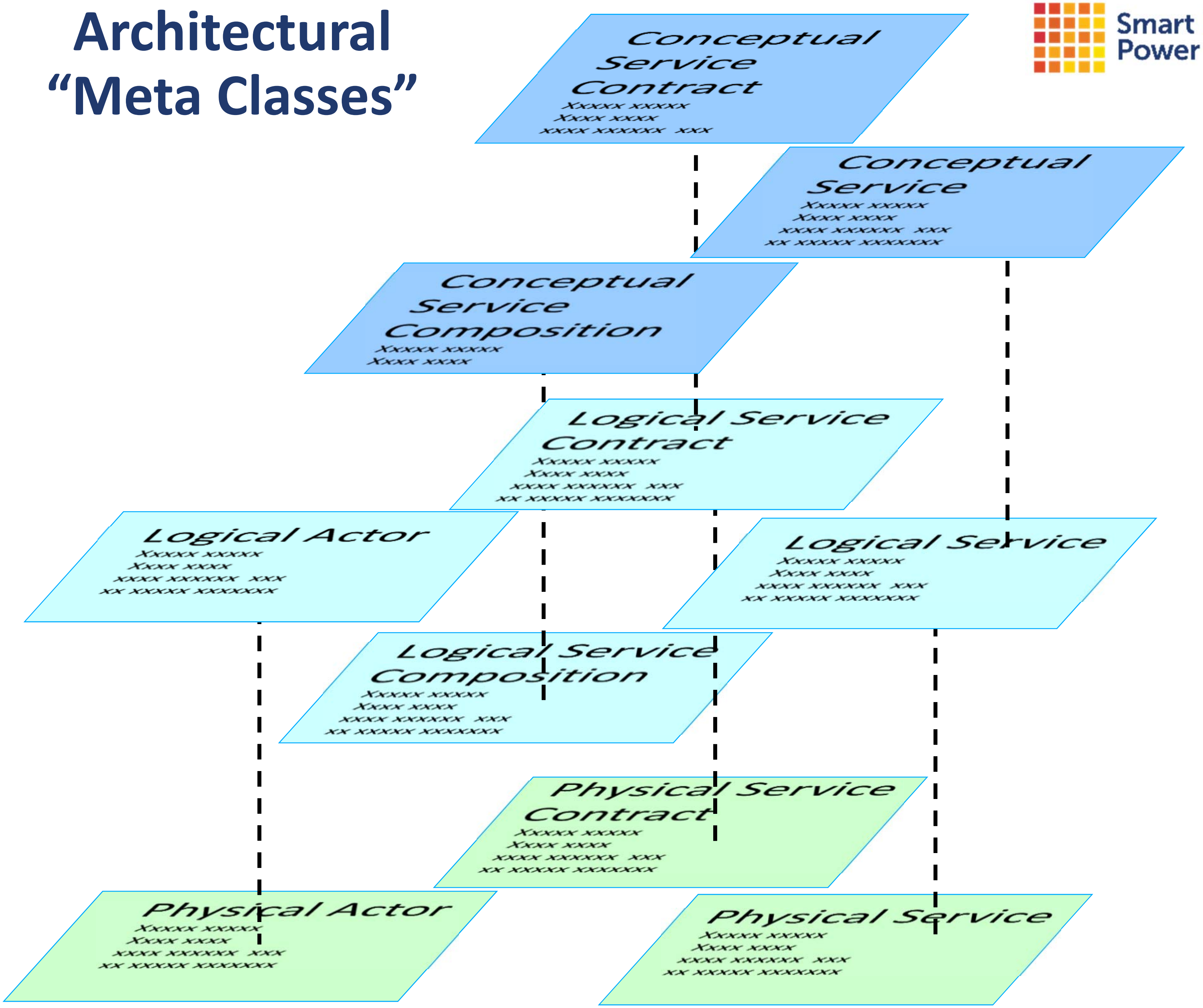
ar:LogicalActor
- ar:isAbstractionOf : ar:PhysicalActor
- ar:subClassOf : ar:LogicalActor
- ar:superClassOf : ar:LogicalActor
- ar:consumes : ar:LogicalService
- ar:isPartyTo : ar:LogicalServiceContract
- ar:provides : ar:LogicalService or ar:LogicalServiceComposition

ar:LogicalService
- ar:isAbstractedAs : ar:ConceptualService
- ar:isAbstractionOf : ar:PhysicalService
- ar:subClassOf : ar:LogicalService
- ar:superClassOf : ar:LogicalService
- ar:isConsumedBy : ar:LogicalActor
- ar:isPartOf : ar:LogicalServiceComposition
- ar:isProvidedBy : ar:LogicalActor
- ar:isSpecifiedBy : ar:LogicalServiceContract

ar:LogicalServiceComposition
- ar:isAbstractedAs : ar:ConceptualServiceComposition
- ar:isCollectionOf : ar:LogicalService or ar:LogicalServiceComposition[1..]

Architectural "Meta Classes"

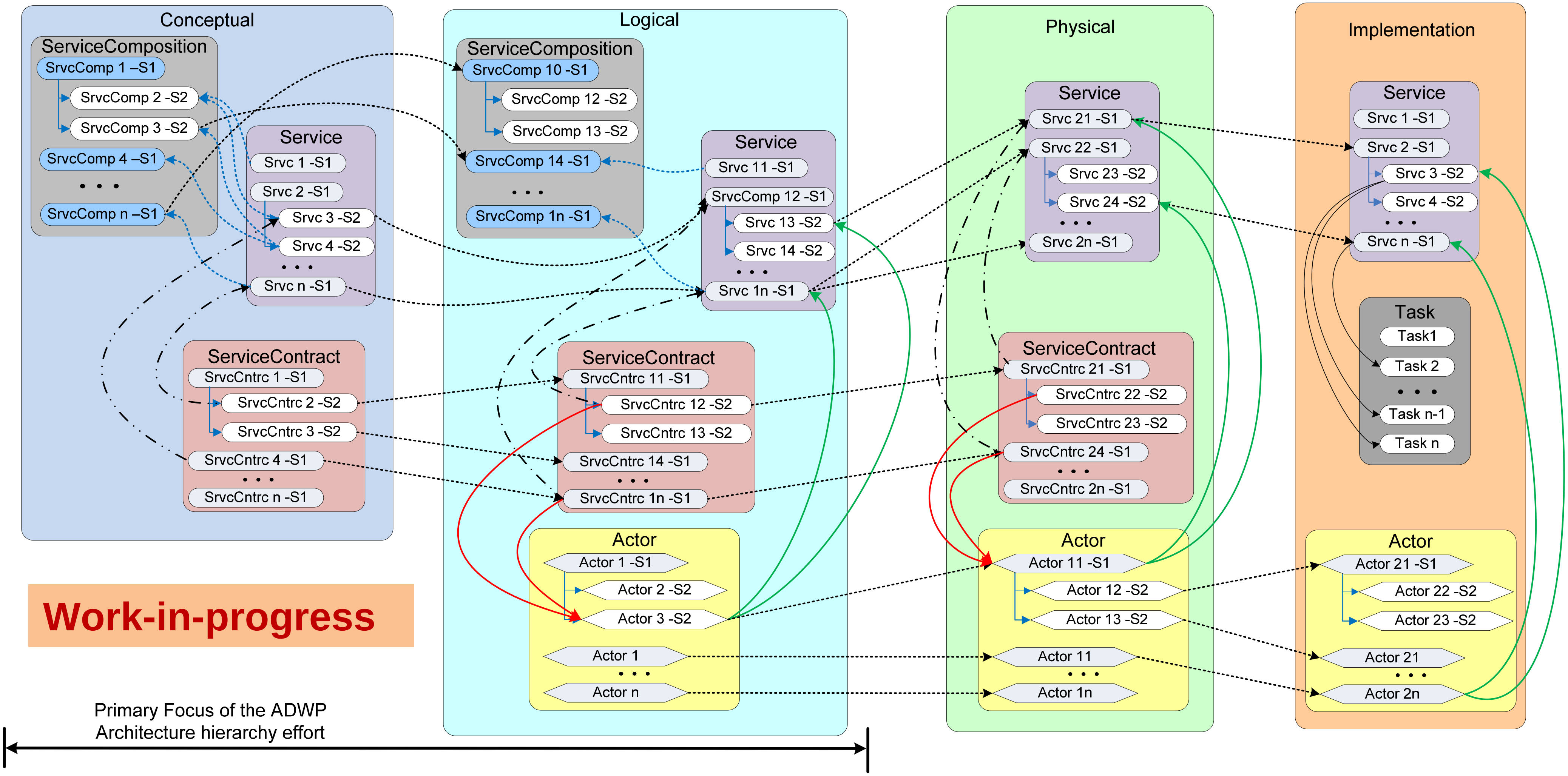
Smart Electric Power Alliance



www.sepapower.org

Abstraction Levels Applied to SOA

Smart Electric Power Alliance



Work-in-progress

Primary Focus of the ADWP
Architecture hierarchy effort

Legend

Mapping from higher to lower level of Element abstraction, with further detailed classification and instantiation of the Element

subsetOf (← memberOf)
S2 ... is hierarchical subset of S1

includedIn (← composedOf)

involvesParty (← isPartyTo)

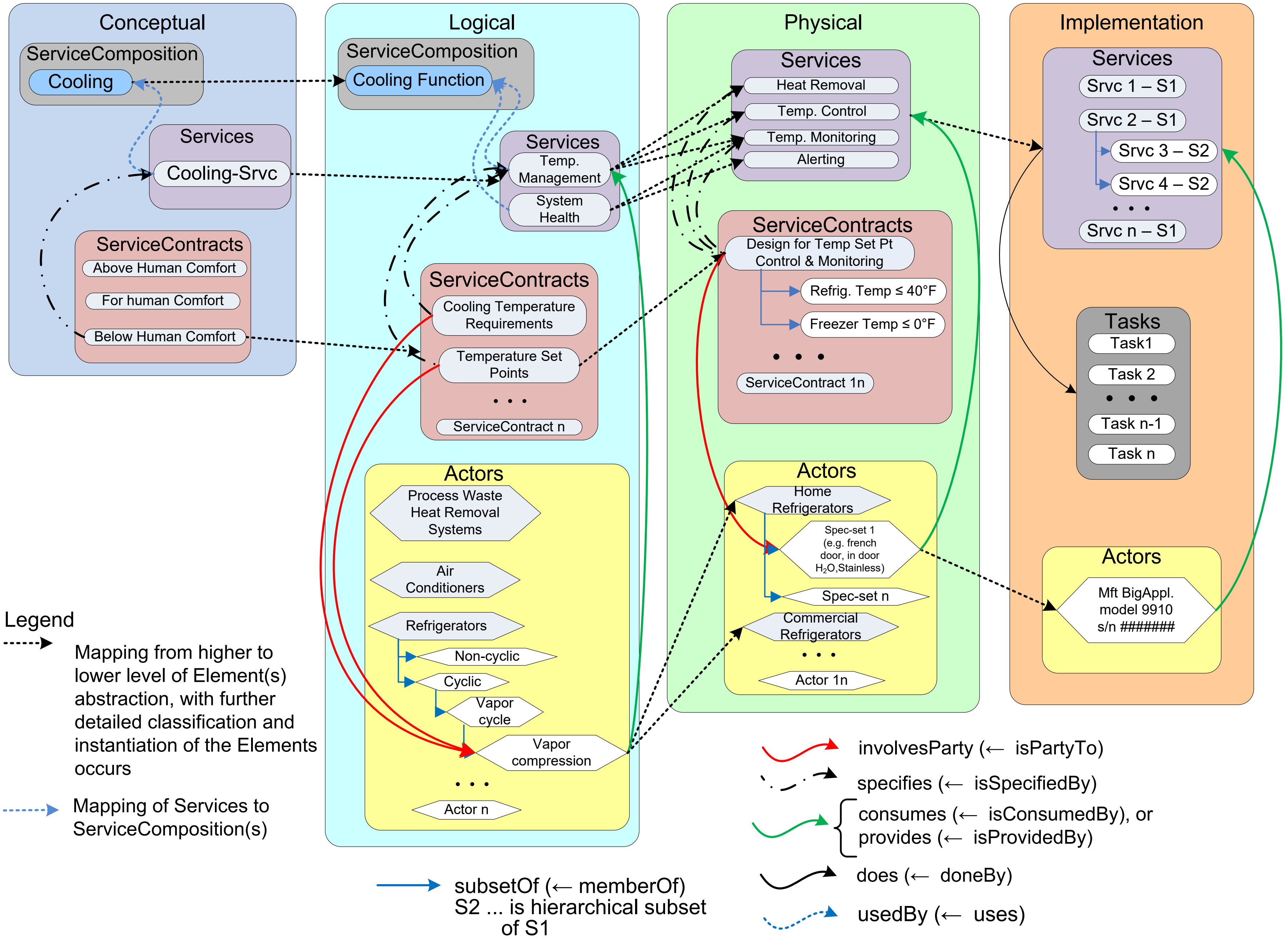
specifies (← isSpecifiedBy)

consumes (← isConsumedBy), or
provides (← isProvidedBy)

does (← doneBy)

www.sepapower.org

Cooling Example



Conceptual

ServiceComposition

Cooling

Services

Cooling-Srvc

ServiceContracts

Above Human Comfort

For human Comfort

Below Human Comfort

Logical

ServiceComposition

Cooling Function

Services

Temp. Management

System Health

ServiceContracts

Cooling Temperature Requirements

Temperature Set Points

• • •

ServiceContract n

Actors

Process Waste Heat Removal Systems

Air Conditioners

Refrigerators

Non-cyclic

Cyclic

Vapor cycle

Vapor compression

• • •

Actor n

Physical

Services

Heat Removal

Temp. Control

Temp. Monitoring

Alerting

ServiceContracts

Design for Temp Set Pt Control & Monitoring

Refrig. Temp ≤ 40°F

Freezer Temp ≤ 0°F

• • •

ServiceContract 1n

Actors

Home Refrigerators

Spec-set 1 (e.g. french door, in door H_2O,Stainless)

Spec-set n

Commercial Refrigerators

• • •

Actor 1n

Implementation

Services

Srvc 1 – S1

Srvc 2 – S1

Srvc 3 – S2

Srvc 4 – S2

• • •

Srvc n – S1

Tasks

Task1

Task 2

• • •

Task n-1

Task n

Actors

Mft BigAppl. model 9910 s/n #######

Legend

Mapping from higher to lower level of Element(s) abstraction, with further detailed classification and instantiation of the Elements occurs

Mapping of Services to ServiceComposition(s)

subsetOf (← memberOf) S2 ... is hierarchical subset of S1

involvesParty (← isPartyTo)

specifies (← isSpecifiedBy)

consumes (← isConsumedBy), or provides (← isProvidedBy)

does (← doneBy)

usedBy (← uses)

Categorization Guidelines for Architectural Elements

Categorize the original elements

- Identify elements as Actors, Services, ServiceCompositions, and ServiceContracts
- Identify the Architecture Abstraction Levels
  - **Conceptual** - these are the business-to-business (B2B) relationships, the B2C (consumer), and the B2R (regulatory & policy). Thus a black-box view of the business.
  - **Logical** – this is where Actors come in, but there are also Services. This is a white-box view of the business, showing the internal elements and their relationships.
  - **Physical** - the representation of a Specification, where the required features (functional and non-functional) are included. Technology approaches are chosen.
  - **Implementation** – actual technology product choices and implementation configuration decisions are made.

Categorization Guidelines for the Elements

- Semantics should be explicitly documented to reduce ambiguity
- An Actor is considered to be a person, organization, or system that has at least one role (ServiceComposition) that initiates or interacts with other Actors. Actors may be internal or external to an organization
- A ServiceComposition is the usual or expected function, or context, in which an actor consumes/provides a service. An Actor may play a number of roles
- At the highest level, a Conceptual ServiceComposition represents the goals of the stakeholder (e.g.: Energy Provider). A Conceptual ServiceComposition will have many defining characteristics
- A Logical ServiceComposition decomposes/partitions a conceptual role into functional components. At the Logical level is where the mapping from roles to Actors occurs
- There is no such thing as a Physical ServiceComposition. At this level an Actor simply offers services according to advertised interfaces. Similarly, there is no such thing as a Conceptual Actor
- A Conceptual ServiceComposition maps to one or more Logical Service Compositions, which get related to one or more Logical Services which are consumed or provided by Logical Actors
- A Physical Actor is an instantiation of a Logical Actor. A Physical Actor performs a set of actions which is a Physical instantiation of a Logical ServiceComposition

Summary

- Created an OWL model (SOA focused), that shows the relationships of the elements across the Conceptual, Logical, Physical abstraction levels
- Started work on:
  - using a specific real world use case set of Actors, Services, messages to vet out the OWL model;
  - relating the use case set of elements to the gathered set of elements using a “neutral concepts” OWL model of the business scenario that the use case came from;
- Intent is to create the appropriate relationships of one set of elements to another for the purposes of helping identify a common vocabulary of architectural/business terms and elements.

Smart Electric Power Alliance

NEUTRAL CONCEPTS MODEL APPROACH

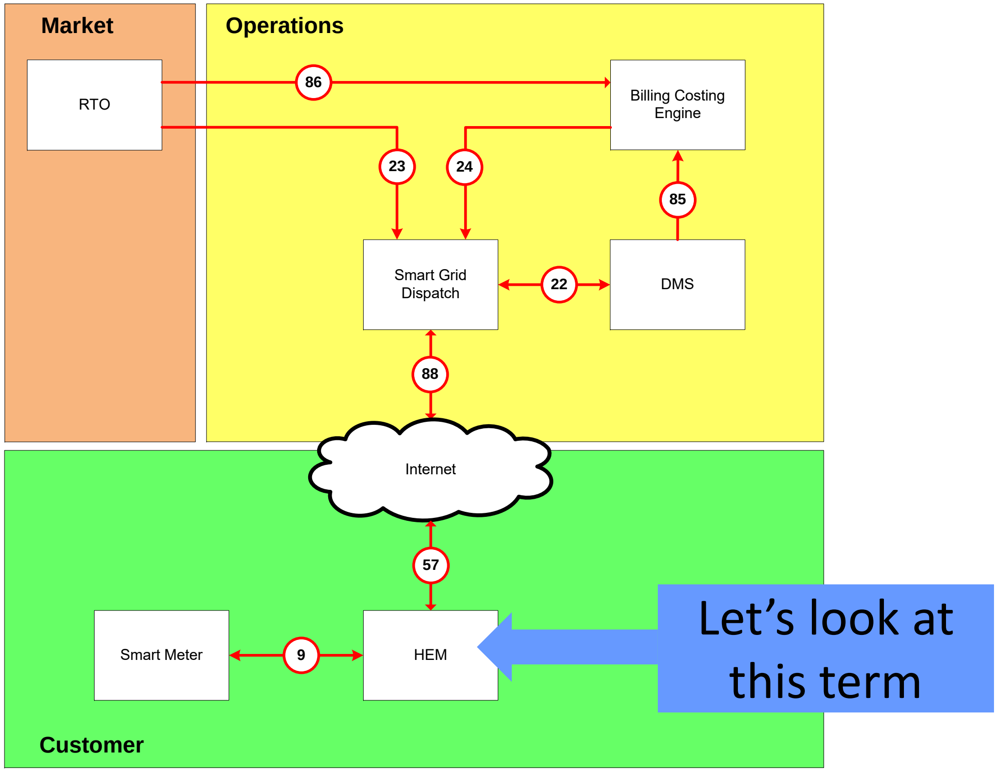
Mapping Use Case Terms

(example from EPRI IntelliGrid Use Case Repository)

Performing Real Time Price Auction

"Acknowledgment: This material is based upon work supported by the Department of Energy under Award Number DE-OE0000193."

Disclaimer: "This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof."

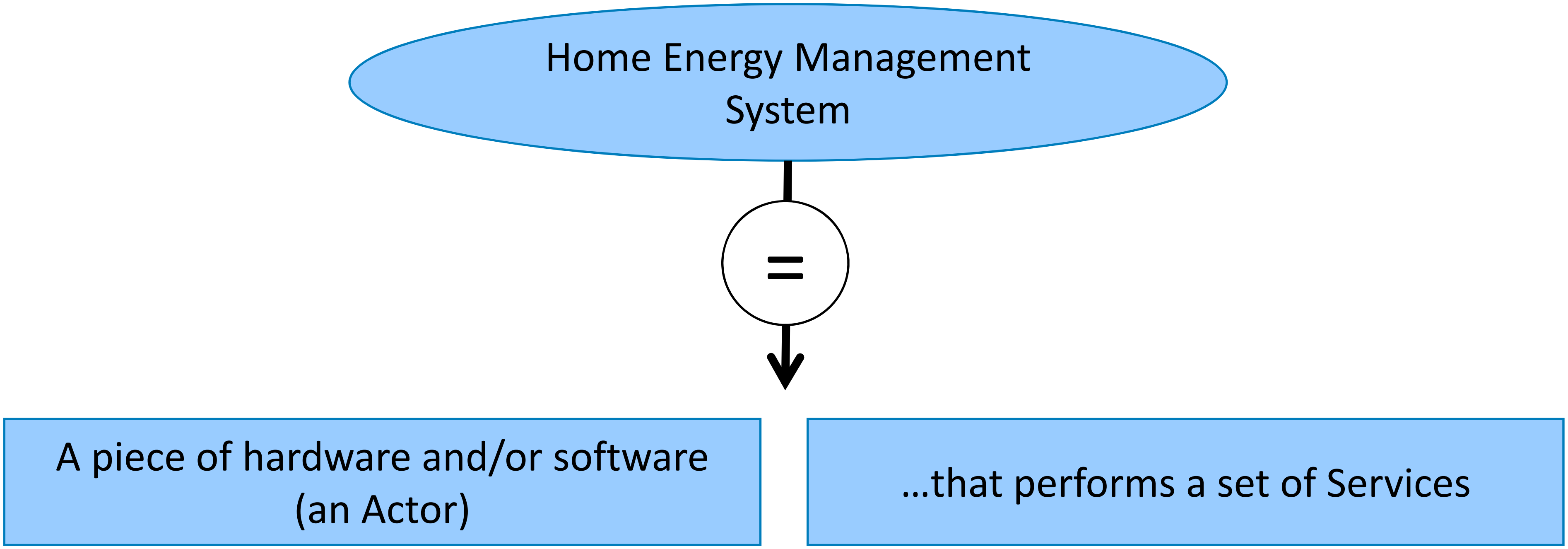


Let's look at this term

The HEM in the AEP/EPRI use case

HEM stands for “Home Energy Management” (System)”

What does that really mean?



Home Energy Management System

=

A piece of hardware and/or software (an Actor)

…that performs a set of Services

www.sepapower.org

The HEM in the AEP/EPRI use case

Smart Electric Power Alliance

A piece of hardware and/or software (an Actor)

There are different kinds of hardware.
What kind of hardware?

Sensor hardware

Actuator hardware

→ Control hardware

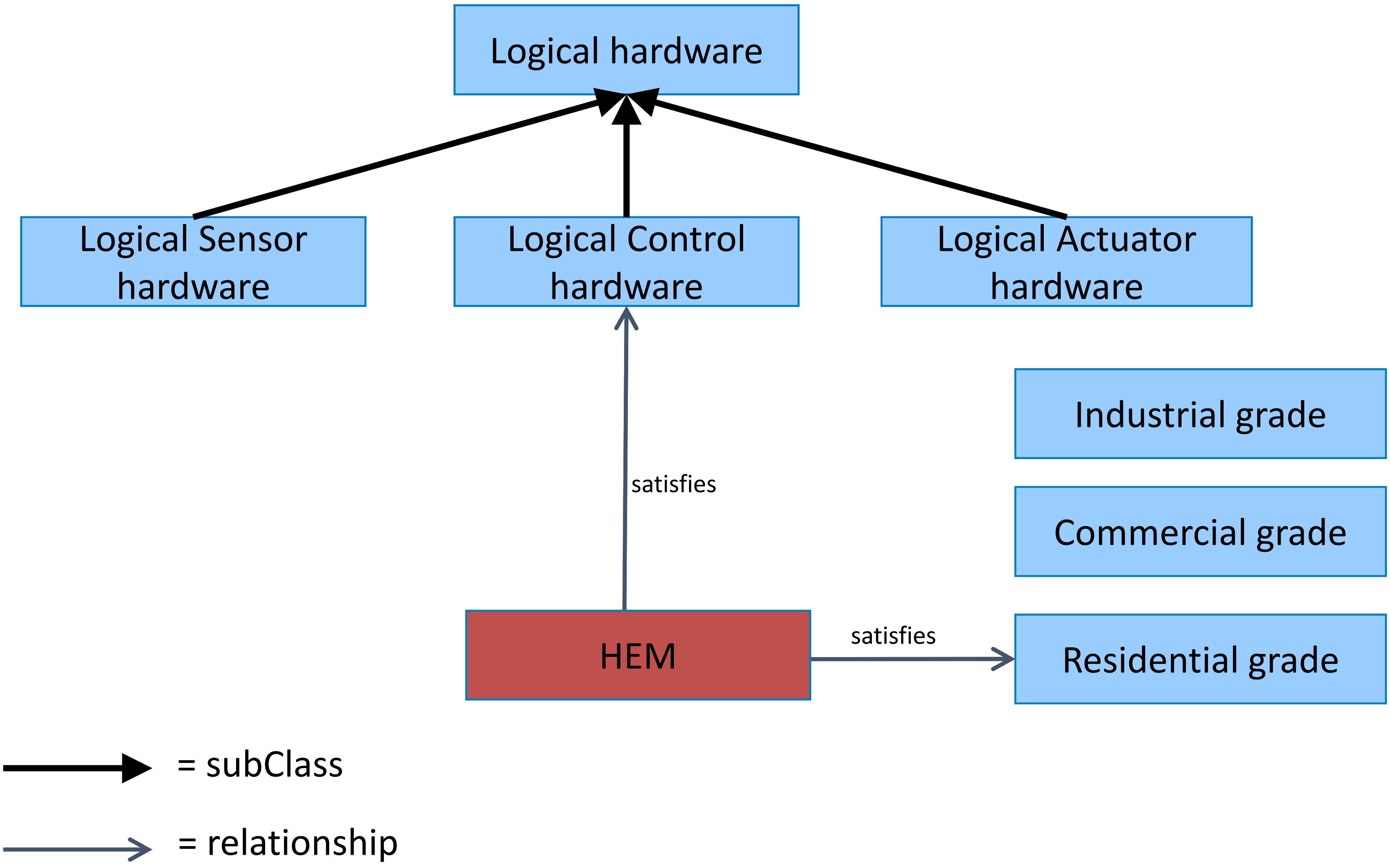
There are different grades of hardware.

Industrial grade

Commercial grade

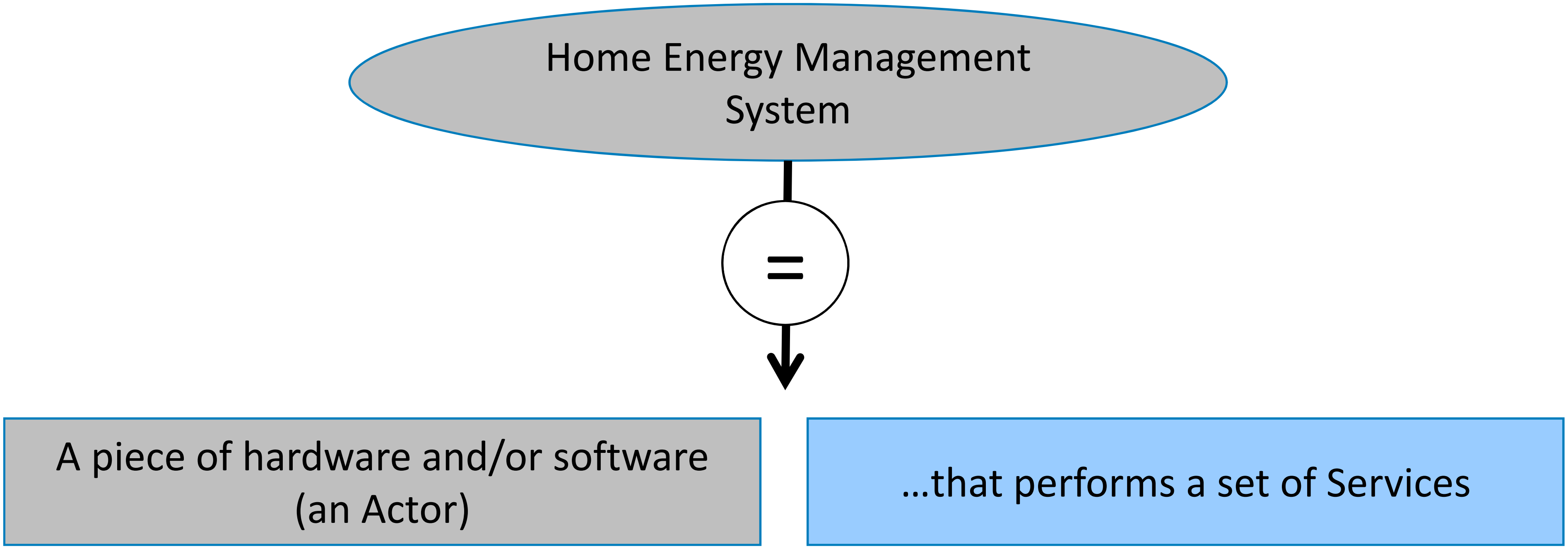
→ Residential grade

The HEM in the AEP/EPRI use case



The HEM in the AEP/EPRI use case

OK, we have characterized the hardware.
What about the Services it performs?



Home Energy Management System

=

A piece of hardware and/or software (an Actor)

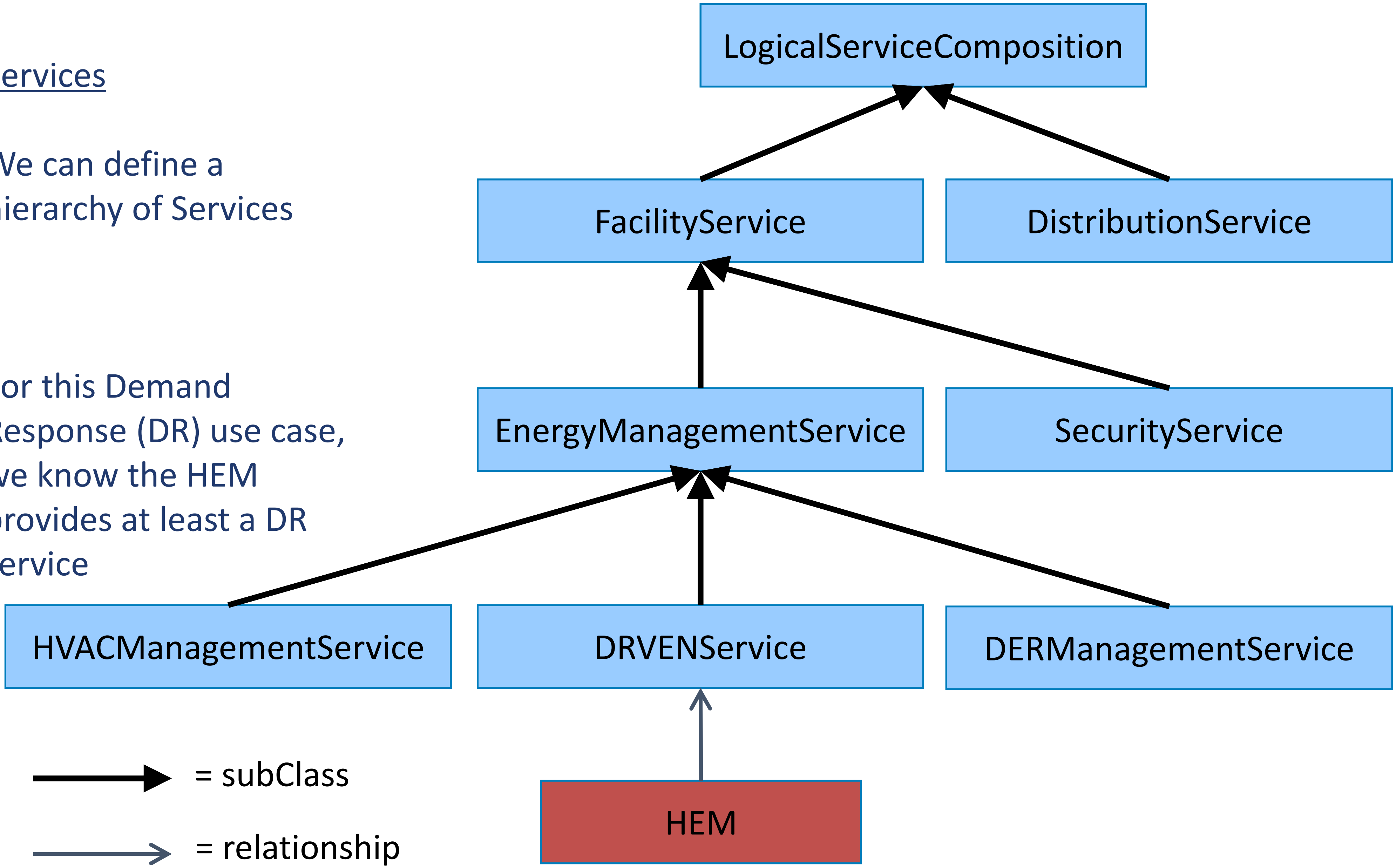
…that performs a set of Services

The HEM in the AEP/EPRI use case

Services

We can define a hierarchy of Services

For this Demand Response (DR) use case, we know the HEM provides at least a DR service

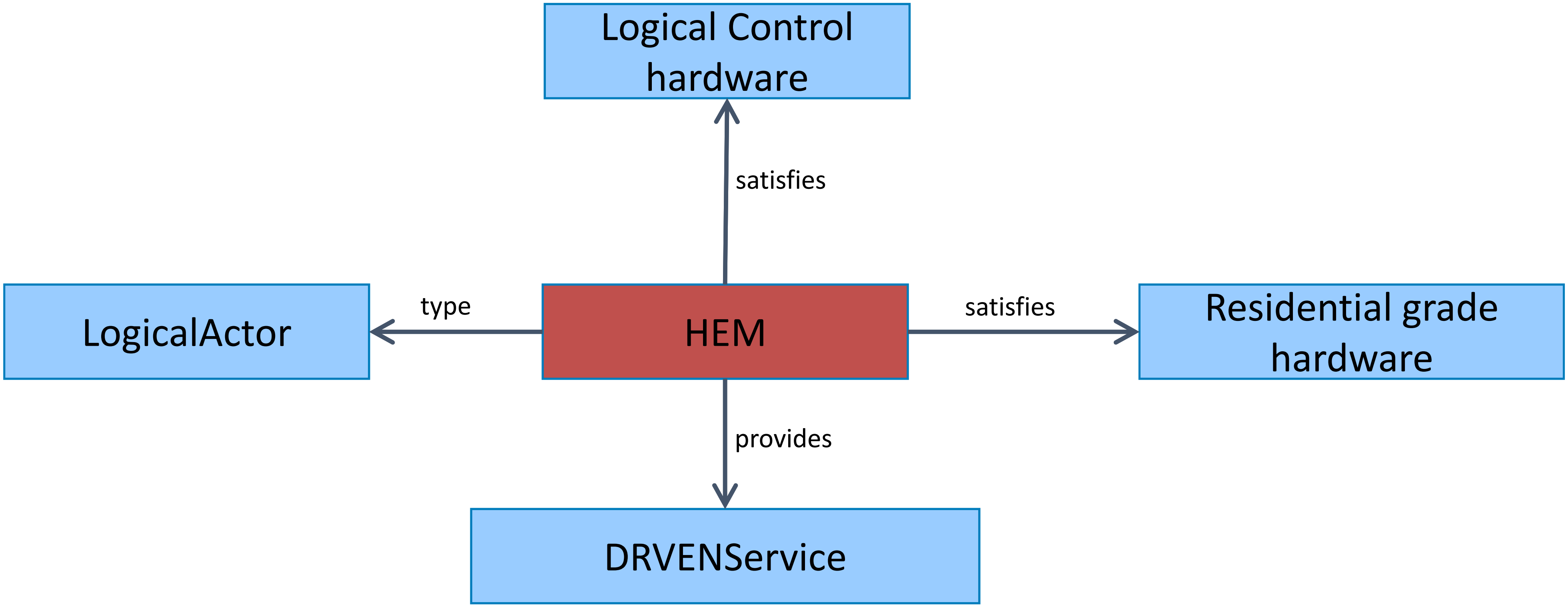


The HEM in the AEP/EPRI use case

Smart Electric Power Alliance

Definition of HEM

So, a definition of the HEM in this use case might be captured as follows:

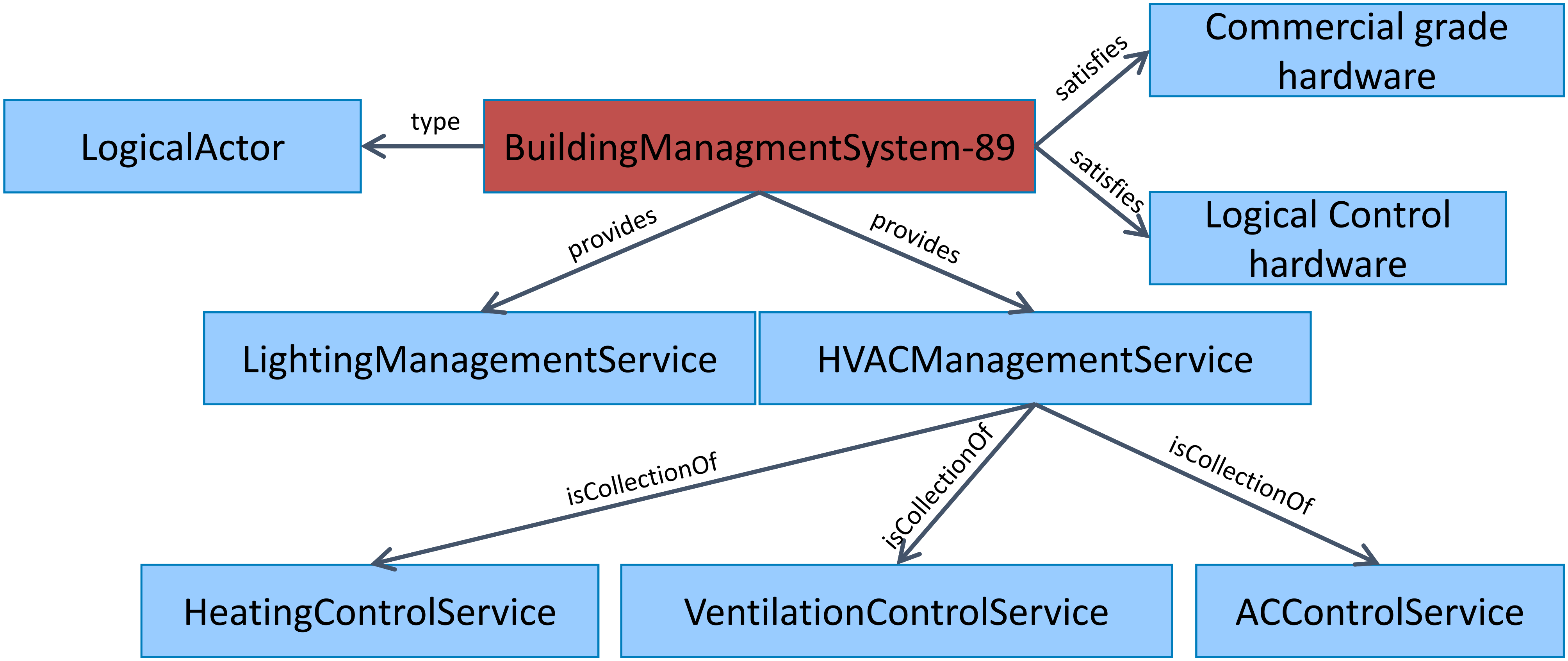


www.sepapower.org

"BuildingManagementSystem"

Definition from the EU

A system consisting of several decentralized controllers and a centralized management system to monitor and control the heating, ventilation, air conditioning, light and other facilities within a building.



LogicalActor

type

BuildingManagmentSystem-89

satisfies

Commercial grade hardware

satisfies

Logical Control hardware

provides

provides

LightingManagementService

HVACManagementService

isCollectionOf

isCollectionOf

isCollectionOf

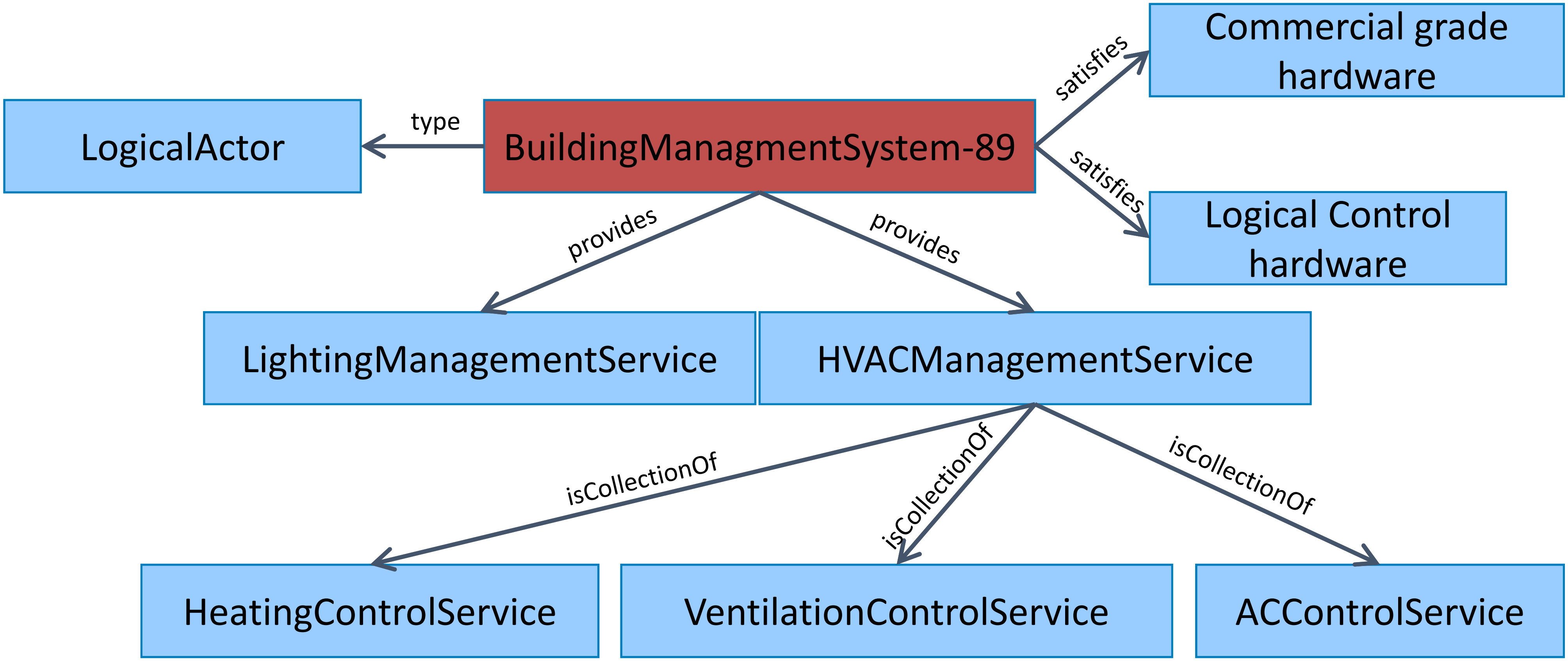
HeatingControlService

VentilationControlService

ACControlService

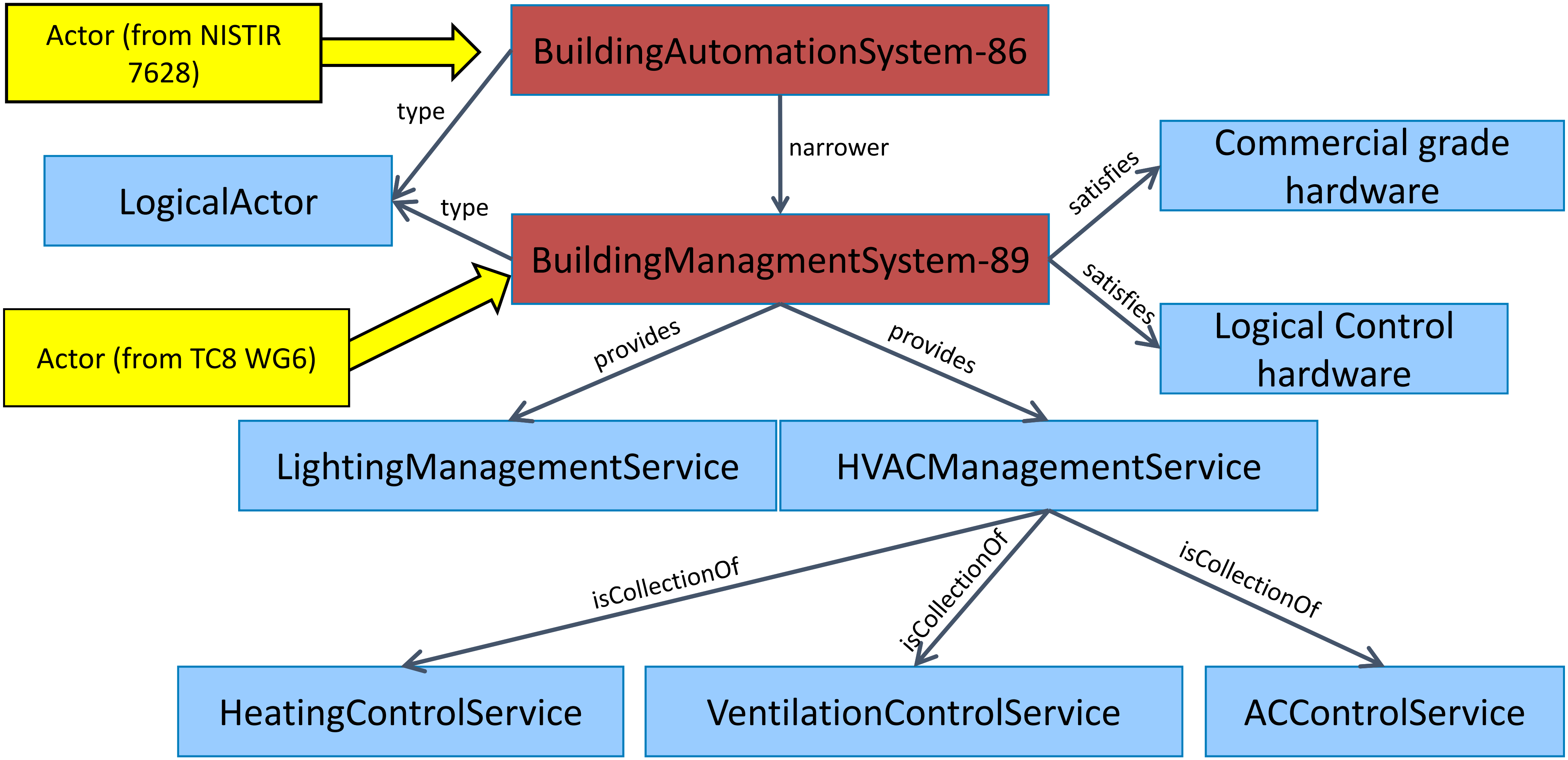
“BuildingManagementSystem”

How does this help? Because if you want to interface with something that provides HVACManagementService, then you are guaranteed that it can perform HeatingControlService (for example).



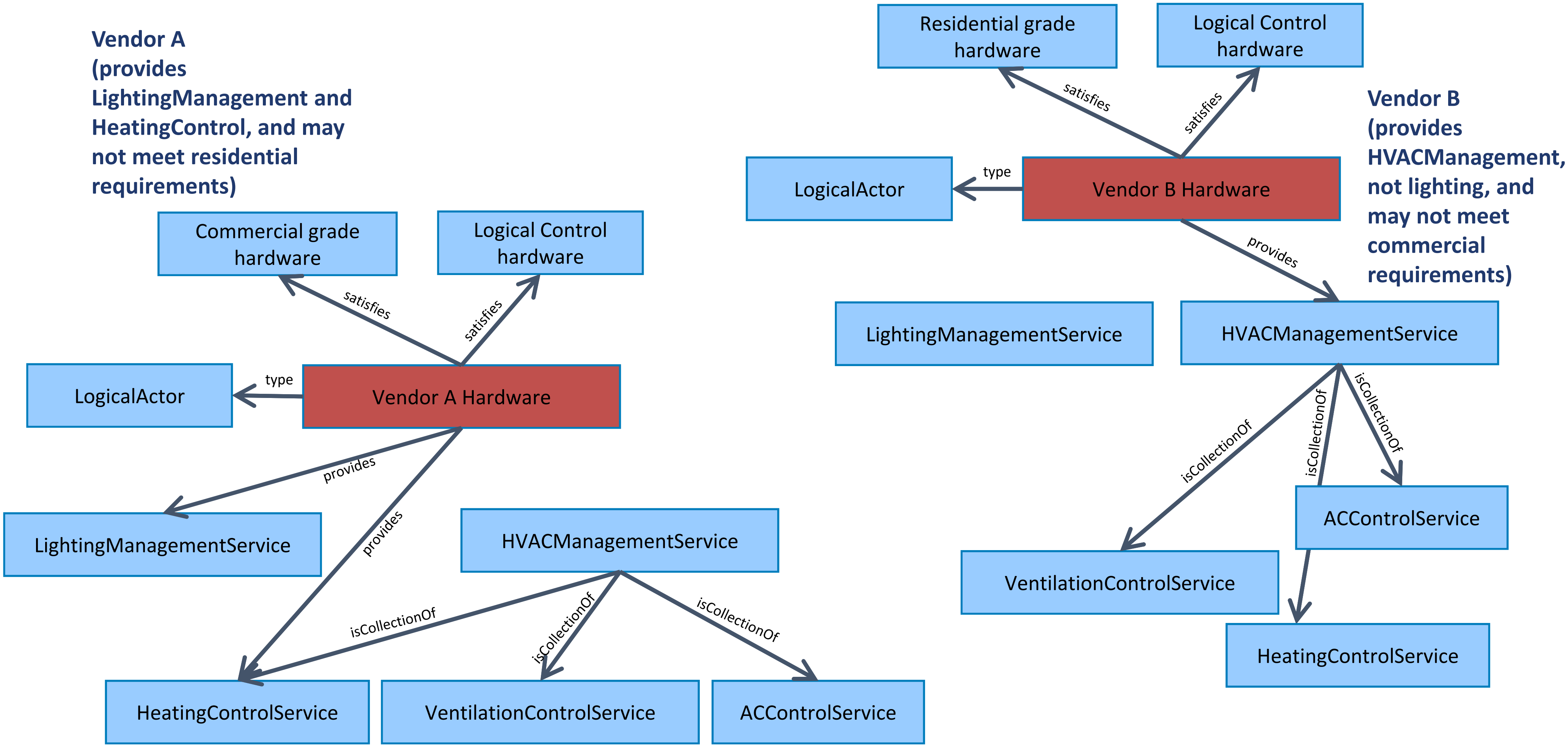
“BuildingAutomationSystem”

…and we can infer that BuildingAutomationSystem-86 can do everything BuildingManagementSystem-89 can do because of the skos:narrower relation.



Implications

Because all this information is represented formally in a semantic language (OWL), it is possible to perform logical reasoning on it. Therefore, if we correctly categorize systems and services, we can begin to automatically identify where interoperability is possible, and where there will be problems.



Vendor A (provides LightingManagement and HeatingControl, and may not meet residential requirements)

Vendor B (provides HVACManagement, not lighting, and may not meet commercial requirements)

Smart Electric Power Alliance

POTENTIAL FUTURE PHASE

Potential Future Phase

- Add relationships between concepts based on latest tool capabilities
- Populate and expand Neutral Concepts model
- Expand interoperability between modeling tools

Future Phase - Details

Smart Electric Power Alliance

Adding Relationships

- Serves to validate and clarify the initial categorizations
- Reveals ambiguities, missing concepts, distinctions between terms. **This is a key effort in the next phase.**
- Ties together concepts originating from different domains (as in the BuildingAutomationSystems example)
- Formalizes the terms in a computable manner

Future Phase - Details

Populating the Neutral Concepts Model

- Provides the concepts that didn’t happen to show up in the contributed terms/vocabularies (as in the HEM example)
- Provides abstractions (superclasses) that tie together terms
- Provides context for contributed terms

Future Phase - Details

Interoperability: TBC OWL to Sparx EA's including support for ArchiMate 3.0

- SGAC modelers have started on mapping Architectural Elements between SOA and ArchiMate 2.1 – DRAFT r0.6
- A plan for data sharing between tools

Mapping: Architectural Elements between SOA and ArchiMate 2.1 – DRAFT r0.6

table row set 1

SOA (aka ADWP Interpretation)			ArchiMate 2.1 (aka ADWP Interpretation)		
Item	Definition (extracted from http://www.opengroup.org/soa/source-book/ontologyv2/index.htm)	Abstraction Level	Item	Model Layer	Definition (extracted from Evaluation Copy of ArchiMate 2.1 Specification)
Actor	SOA HumanActor and/or System (an organized collection of Services and Tasks)	Logical, Physical, Implementation	Business Actor	Business	an organizational entity that is capable of performing behavior.
			Application Component	Application	a modular, deployable, and replaceable part of a software system that encapsulates its behavior and data and exposes these through a set of interfaces.
			Node	Technology	a computational resource upon which artifacts may be stored or deployed for execution.
			Network	Technology	a communication medium between two or more devices.
			Application Collboration	Application	an aggregate of two or more application components that work together to perform collective behavior.
Actor – level 1, 2,3			How to represent in ArchiMate?		
ServiceComposition	aka Role. Result of assembling a collection of services in order to perform a new higher-level service.	Conceptual, Logical	Business Role	Business	the responsibility for performing specific behavior, to which an actor can be assigned.
ServiceComposition – level 1,2,3	A nested level of granularity of detail of Services that make up the Service Composition at a given abstraction level		Business Collaboration	Business	an aggregate of two or more business roles that work together to perform collective behavior.

Work-in-progress

Mapping: Architectural Elements between SOA and ArchiMate 2.1 – DRAFT r0.6

table row set 2

Service	a logical representation of a repeatable activity that has a specified outcome. It is self-contained and is a 'black box' to its consumers.	Conceptual, Logical, Physical, Implementation	- Business Service	Business	a unit of functionality that a system exposes to its environment, while hiding internal operations, which provides a certain value (monetary or otherwise) Fulfills a business need for a customer (internal or external to the organization).
			Application Service	Application	exposes automated behavior
			Infrastructure Service	Technology	an externally visible unit of functionality, provided by one or more nodes, exposed through well-defined interfaces, and meaningful to the environment
Service – lvl 1,2,3			How to represent in ArchiMate?		
Task	an atomic Activity within a Process flow	Implementation	Application Function	Application	a behavior element ~~that groups~~ of automated behavior that can be performed by an application component
ServiceContract	defines the terms, conditions, and interaction rules that interacting participants must agree to (directly or indirectly). explicitly regulate both the interaction aspects and the legal agreement aspects of using a service.	Conceptual, Logical, Physical	Contract	Business	a formal or informal specification of an agreement that specifies the rights and obligations associated with a product
ServiceContract – levels 1,2,3			How to represent in ArchiMate?		

Work-in-progress

Mapping: Architectural Elements between SOA and ArchiMate 2.1 – DRAFT r0.6

table row set 3

Event	Something that happens		Business Event	Business	something that happens (internally or externally) and influences behavior.
			Business Function	Business	a behavior element that groups behavior based on a chosen set of criteria (typically required business resources and/or competences).
			Infrastructure Function	Technology	a behavior element that groups infrastructural behavior that can be performed by a node.
			Business Process	Business	a behavior element that groups behavior based on an ordering of activities. It is intended to produce a defined set of products or business services.

Work-in-progress

Mapping: Architectural Elements between SOA and ArchiMate 2.1 – DRAFT r0.6

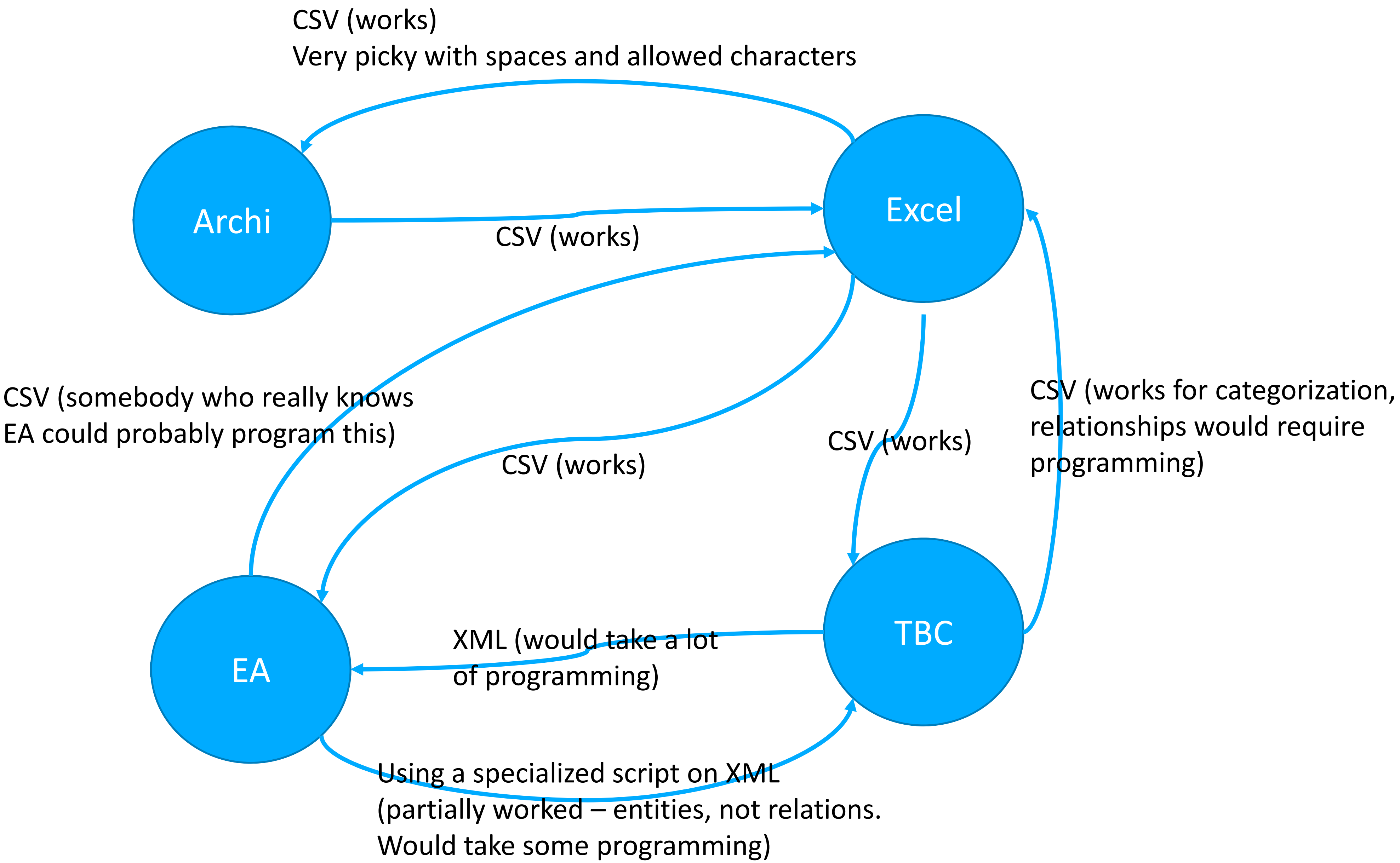
table row set 4

Smart Electric Power Alliance

Conceptual	models the actual business as the Owner conceptually thinks the business is or maybe, wants the business to be. Treat the business as "black box" noting the major functions & capabilities that business has with customers and other 3rd parties	[redacted]	Some users of ArchiMate have: conceptual abstraction elements positioned in the Business layer; logical abstraction elements positioned in the Application layer; physical abstraction elements in the Technology layer. Some also equate the Business layer to Conceptual. EA can absolutely mix UML class diagrams to behavior diagrams. When some diagramming concepts are complementary: draw a business process, then decompose into an activity diagram and/or sequence diagrams. How does ArchiMate support this: doesn't care, not using it to generate other artifacts. E.G.: business layer's inactive objects (meter reading), application layer (map to CIM UML class diagram for the application layer data objects), then technology layer (map the technology data object to DDL - database definition language). Different UML diagrams complement the ArchiMate diagrams. Informational links, hyperlinks, drag&drop to create a hyperlink to so the diagrams are related. [note that the ArchiMate architecture diagrams services are associated with interfaces, components, functions, and objects (where information is stored/passed). ArchiMate 2.1 metamodel does not explicitly support the associations or linkages across the conceptual, logical, physical abstraction levels as ADWP illustrates in the SOA Visio illustration. ArchiMate does support relationships from e.g. business service to an application service via derived relationships.]
Logical	models of the "systems" of the Business, and treats the business as "white box" to further detail and describe the functions & capabilities performed within the business and with 3rd parties. How the Architecture is (ideally) structured?	[redacted]	
Physical	the technology-constrained, physical implementation design of the systems of the business. Documents the Building block specifications, requirements, descriptions, non-functionals.	[redacted]	
Implementation	This is where the systems specialize and document the processes and software and actual technology and implementation configurations used to execute. E.g.: –For ICT these are the vendor offerings or custom applications –For organizations is the personnel, and workflow execution	[redacted]	

Work-in-progress

Tools: Exchange Experiments



CSV (works)
Very picky with spaces and allowed characters

CSV (works)

CSV (somebody who really knows EA could probably program this)

CSV (works)

CSV (works)

CSV (works for categorization, relationships would require programming)

XML (would take a lot of programming)

Using a specialized script on XML (partially worked – entities, not relations. Would take some programming)

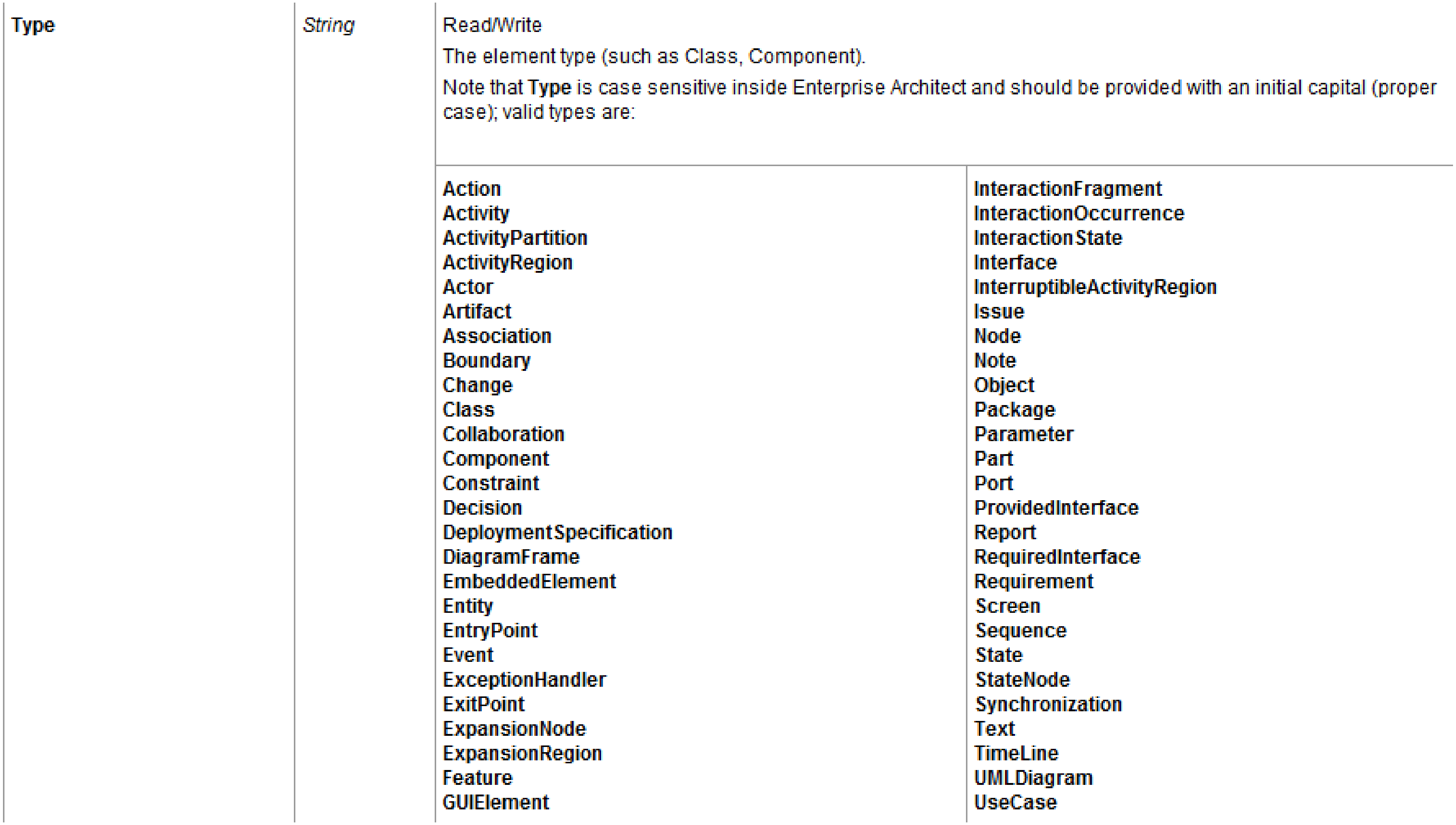
Tools: Archi v3.x Export Format

ID	Type	Name
5544b54c	ArchimateModel	(new model)
b5995997	BusinessActor	Business Actor 3
fef227a6	BusinessActor	CustomerAppliance
d89934c8	BusinessFunction	Business Function
59de4964	BusinessRole	Business Role
8066cb08	BusinessRole	Business Role 2
9cdbbc83	BusinessService	ApplianceControl
ff57a9b5	ApplicationService	Application Service
8ff1067d	InfrastructureService	Infrastructure Service
d5d8a675	Junction	Junction

Tools: Archi Import Format

ID	Type	Name	Documentation
1	BusinessRole	EnergyDispatcher	Actor Role in charge of dispatching Electricity usually at the transmission level
2	ApplicationComponent	Access Point ie. Data Aggregation Point	Collects and aggregates all data fromto end devices through Neighbourhood andor Field Area andor AMI networks. These devices are IEEE 802.11n compliant. They interface the Backhaul WAN.
3	BusinessService	Account Management	Business function or service that manages supplier and customer business accounts.
4	ApplicationComponent	Account Management System	Business function or service that manages supplier and customer business accounts.
5	BusinessRole	Actuator	An actuator is a transducer that accepts a signal and converts it to a physical action. In other words
7	BusinessRole	Aggregator	The role that combines curtailment
8	ApplicationComponent	AMI Head-end	Provides the Interface to the Advanced Metering Infrastructure and manages the information exchanges between third party systems. The head-end is the command application for AMI solution. Optionally it manages required or desired meter reading schedules
9	ApplicationComponent	AMI Operator	General operator of the AMI system
10	ApplicationComponent	AMI Service Engineer	External actor responsible for the installation

Tools: Valid Sparx EA 12.1 Import Types – Partial List



Type	*String*	Read/Write The element type (such as Class, Component). Note that **Type** is case sensitive inside Enterprise Architect and should be provided with an initial capital (proper case); valid types are: Action Activity ActivityPartition ActivityRegion Actor Artifact Association Boundary Change Class Collaboration Component Constraint Decision DeploymentSpecification DiagramFrame EmbeddedElement Entity EntryPoint Event ExceptionHandler ExitPoint ExpansionNode ExpansionRegion Feature GUIElement InteractionFragment InteractionOccurrence InteractionState Interface InterruptibleActivityRegion Issue Node Note Object Package Parameter Part Port ProvidedInterface Report RequiredInterface Requirement Screen Sequence State StateNode Synchronization Text TimeLine UMLDiagram UseCase

Tools: TBC Format

(sample subset)

Entity Name	Description	Abstraction Level	Element Type	skos	NAESB	CSWG NISTIR 7628	CSWG Actors	IEEE P2030 Draft 3	EIS	EU Commission	EU WGSP Vers 0-5	SGAC Conceptual Arch
Aggregator	offers services to aggregate energy production from different sources (generators) and acts towards the grid as one entity, including local aggregation of demand (Demand Response management) and supply (generation management). In cases where the aggregator is not a supplier, it maintains a contract with the supplier.	Conceptual	ServiceComposition	TBD								
AMI Operations	General operator of the AMI system	Logical	Actor	TBD							X	
AMI Service	ServiceComposition responsible for the installation, operation, maintenance and de-installation of the system components. It may access, if properly identified and authorized, those components either directly, via local operation and maintenance interfaces, or from a system component from a higher hierarchical level (e.g. meters may be accessed for maintenance purposes via NNAPs or the HES).	Logical	ServiceComposition	TBD							X	

The Entity Name and Description were key to determining the abstraction level and element type. The submitting organization's identity was imported for traceability.

Smart Electric Power Alliance

Thank you

Smart Electric
Power Alliance