Semantics and the Internet of Things

John Strassner, Joel Halpern, Qin Wu

HUAWEI TECHNOLOGIES CO., LTD.

HUAWEI

Motivation

- **Exponential Growth of Data Being Digitally Stored and Exchanged**
  - IoT and M2M imply that every device can be connected, and viewed as a sensor and/or actuator
  - Devices range from simple sensors and actuators that only provide pre-programmed functions to "smart objects" that change the data they produce at runtime
- **Understanding Data is Increasingly More Difficult**
  - IoT is increasingly being populated by diverse silos and systems operating as silos
  - Different data sources may use different devices that generate different types of data
  - Volume, velocity, variability mean understanding the context that generated the data is critical
  - Extracting *value* from huge amounts of data that change frequently is increasingly more difficult
- **Novel, Data-driven Business Models**
  - Increasing need for businesses to produce actionable insights requires an understanding of the context that produced data

The Role of Models and Ontologies

- **Both Are Important**

HIGH	Models	Ontologies
Type of Assumption	Closed World	Open World
Formal Language	NO Information is defined by descriptive text	YES Information is defined using formal logic
Purpose	Runtime realization of knowledge Info Model provides a common vocabulary Data Models provide implementation specifics	Runtime exploitation of knowledge Enables reasoning on data
Suitability for Devices	HIGH Up to other entities to interpret the data	LOW Requires computational reasoning to use
Dealing with uncertainty	Hard but straightforward; models define facts and metadata, which are used in logic	Very difficult; no standard reasoners are available for fuzzy or similar reasoning

- **Both Used to Drive the Design of a Domain Specific Language (DSL)**

The Importance of Semantics

"An object by itself is intensely uninteresting"

- Grady Booch, Object Oriented Design with Applications, 1991

Data	Examples	What You Get
Types of Data	Machine data, documents, multimedia, email, blogs, pictures, LOD, …	**Syntax** *Context and semantics are hidden*
Named Entities	Objects in a model, or concepts in an ontology	**Context** *Semantics are hidden*
Relationships	Typically *hidden in the data*	**Semantics** *Now the data are understood!*

- **Semantics**
  - The key to understanding data, and being able to make decisions
  - **Context** *orients* the data, **semantics** *helps interpret* the data

Dey's Popular Definition of Context

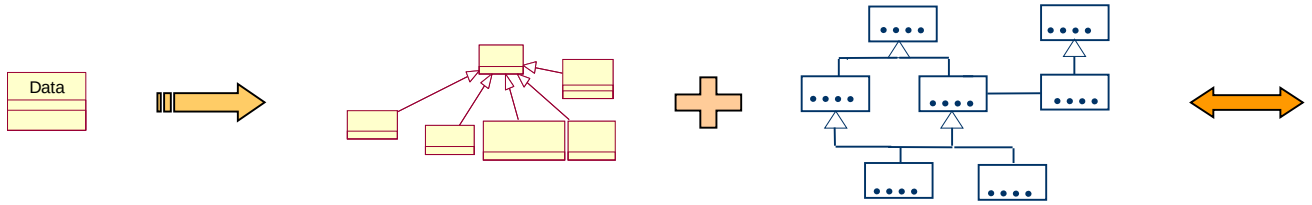
"any information ❶ relevant to an interaction that can be used to characterize the situation of an entity. ❷ An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, ❸ including the user and application themselves"

This is lacking a clear definition of how semantics relates to context

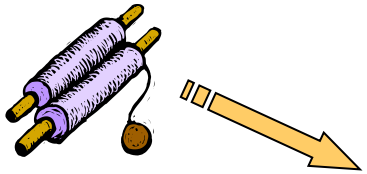
Ref [1]

❶ We Need *Knowledge*, not Just Data or Info

- **Context Is NOT Just Location!!!**
- **Context Is MORE Than "Just" Data**



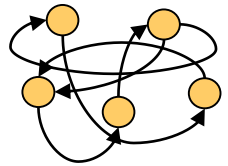
- **Context Can Alter Semantics and Behavior**



Regulatory Policies

SLAs

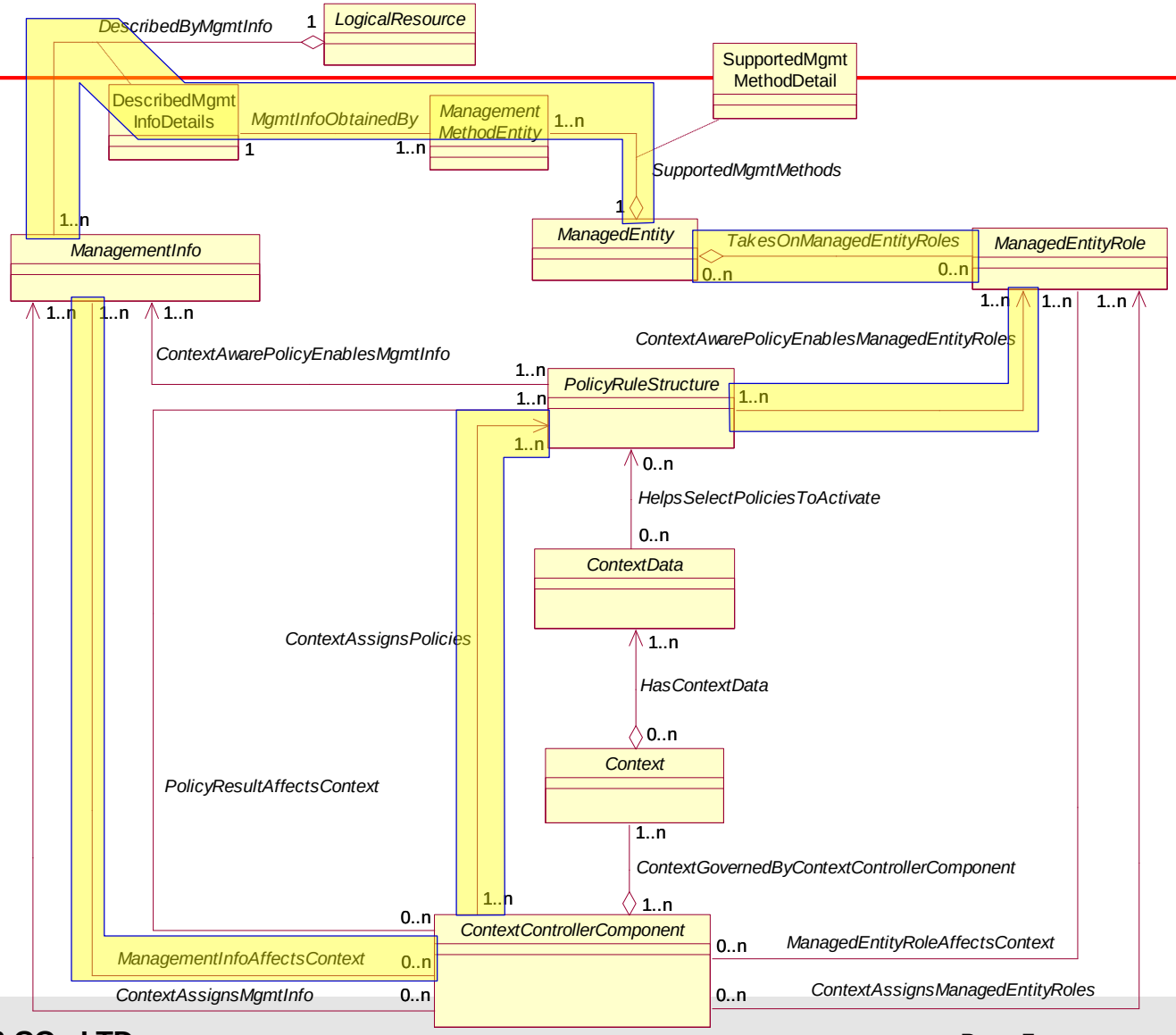
Models and Ontologies



Reasoning and Learning

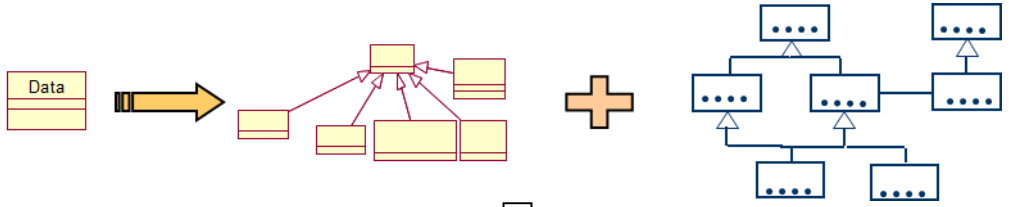
Ref [2]

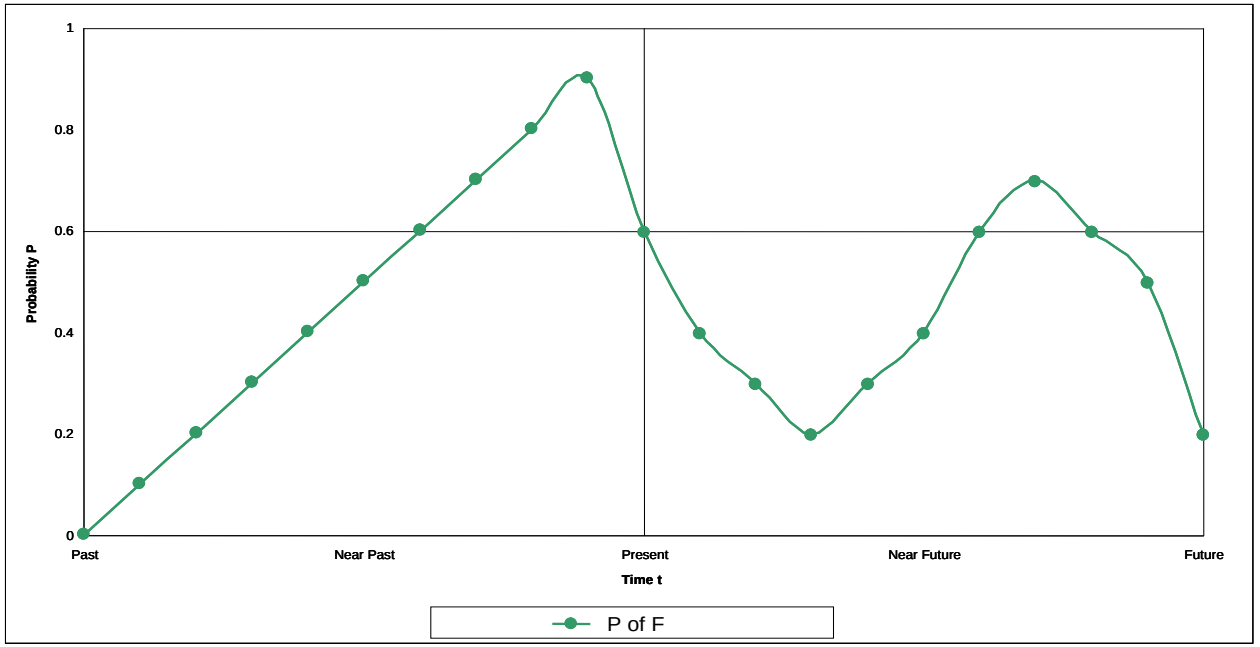
❷ Situatedness



Ref [2]

❸ Context May Change Relevance





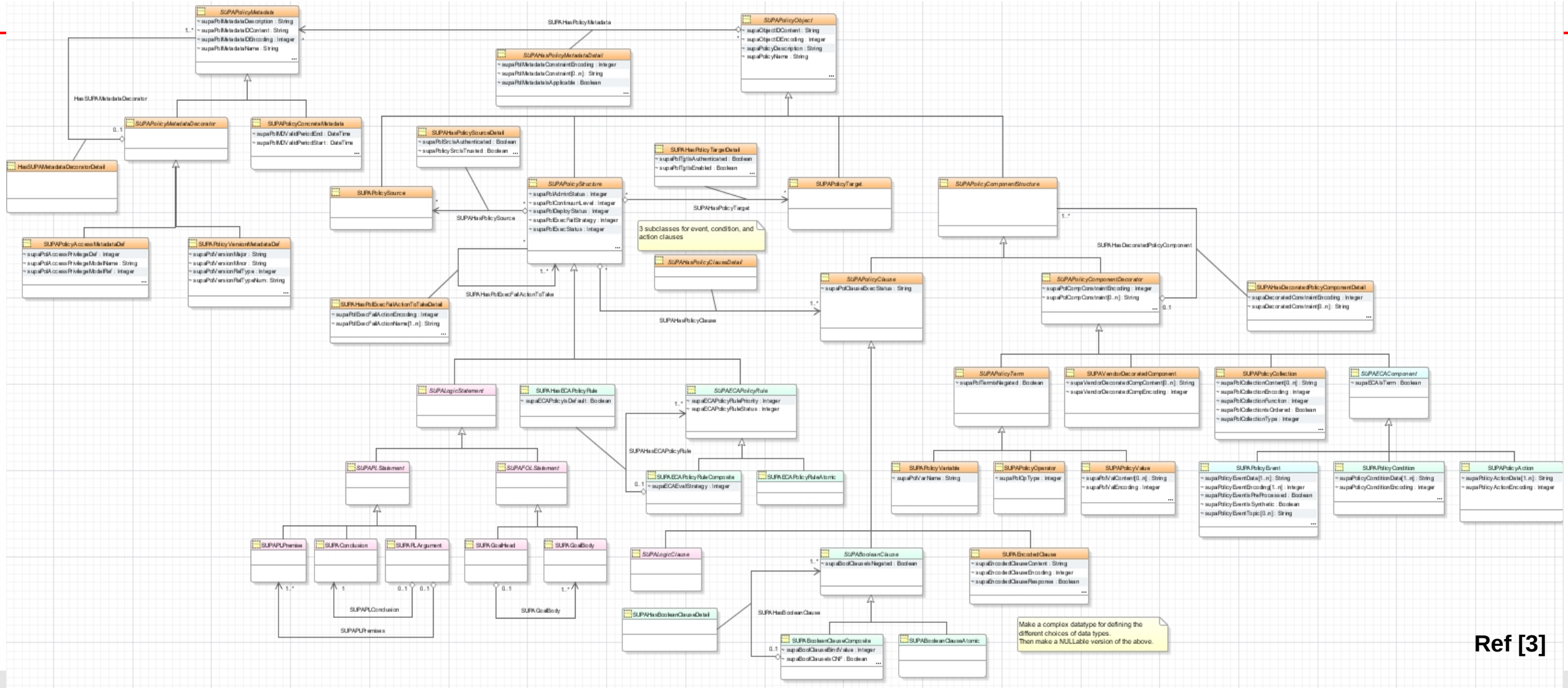
Ref [2]

DEN-ng Context Definition

> **The Context of an Entity is a collection of measured and inferred knowledge that describe the *state* and *environment* in which an Entity exists or has existed**

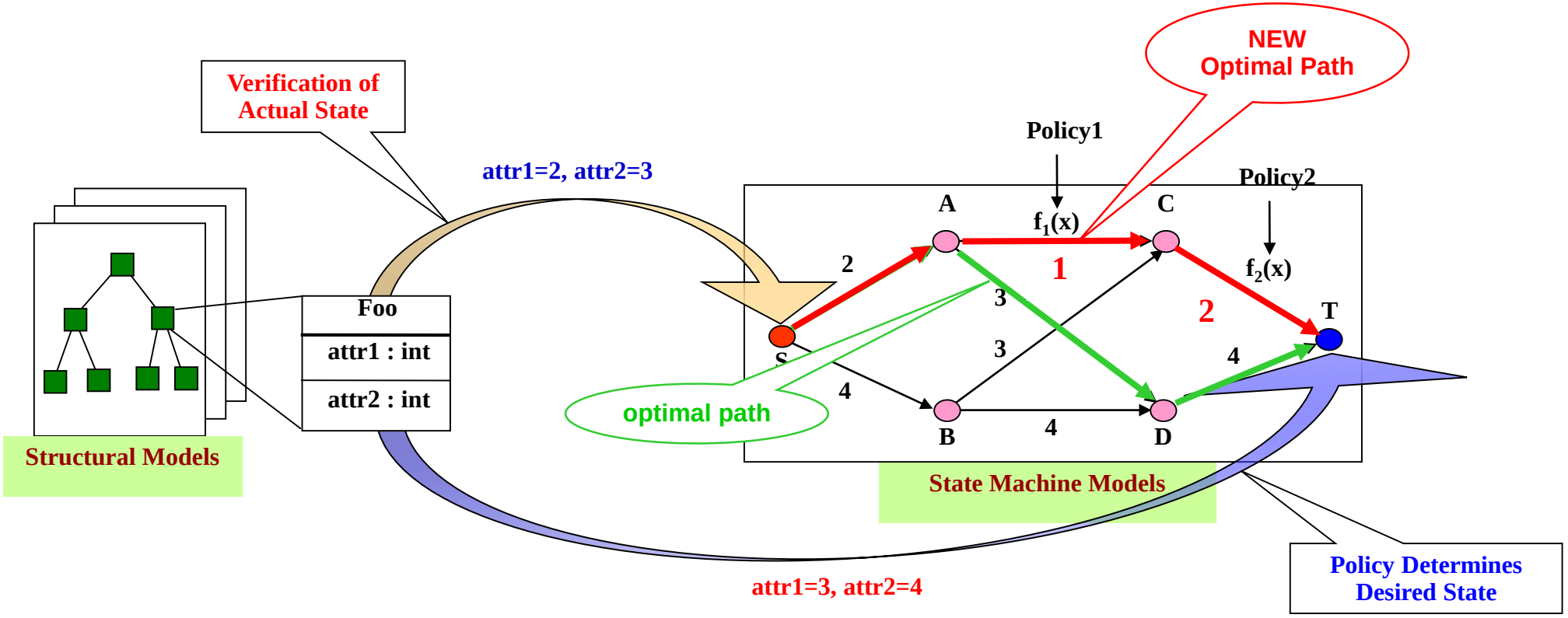
Ref [2]

SUPA Generic Policy Rules

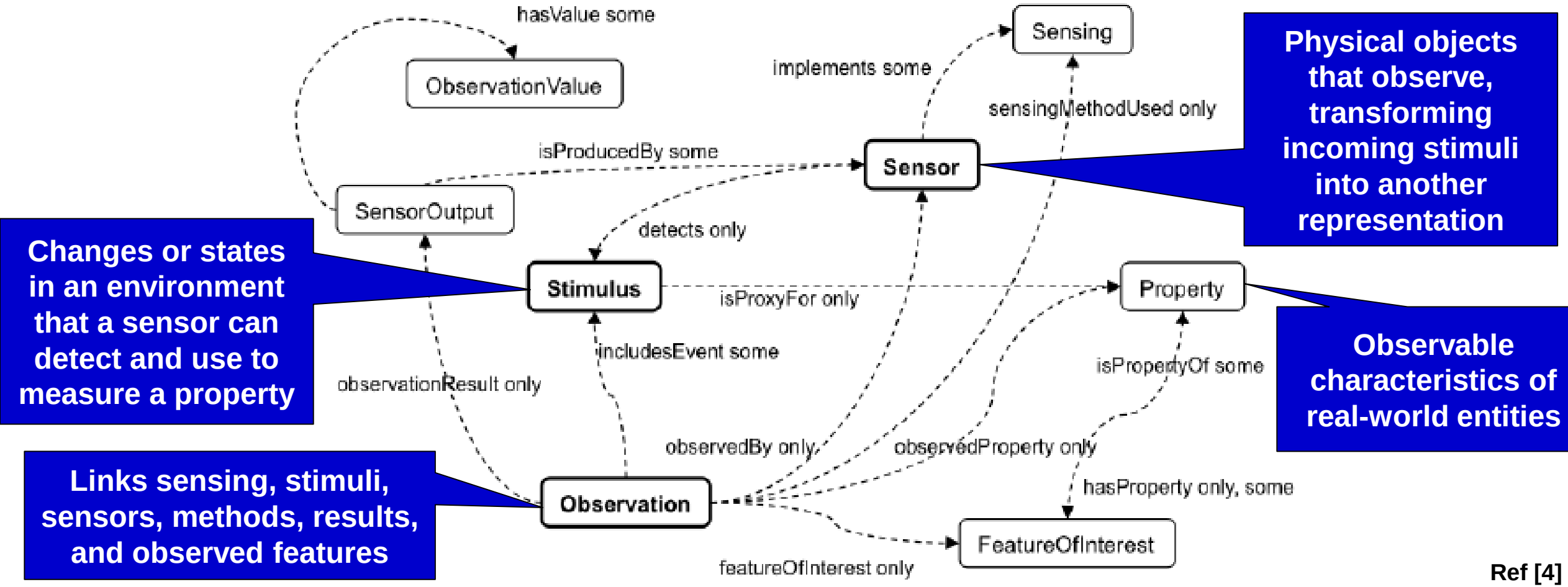


Ref [3]

Policy-driven Behavioral Orchestration

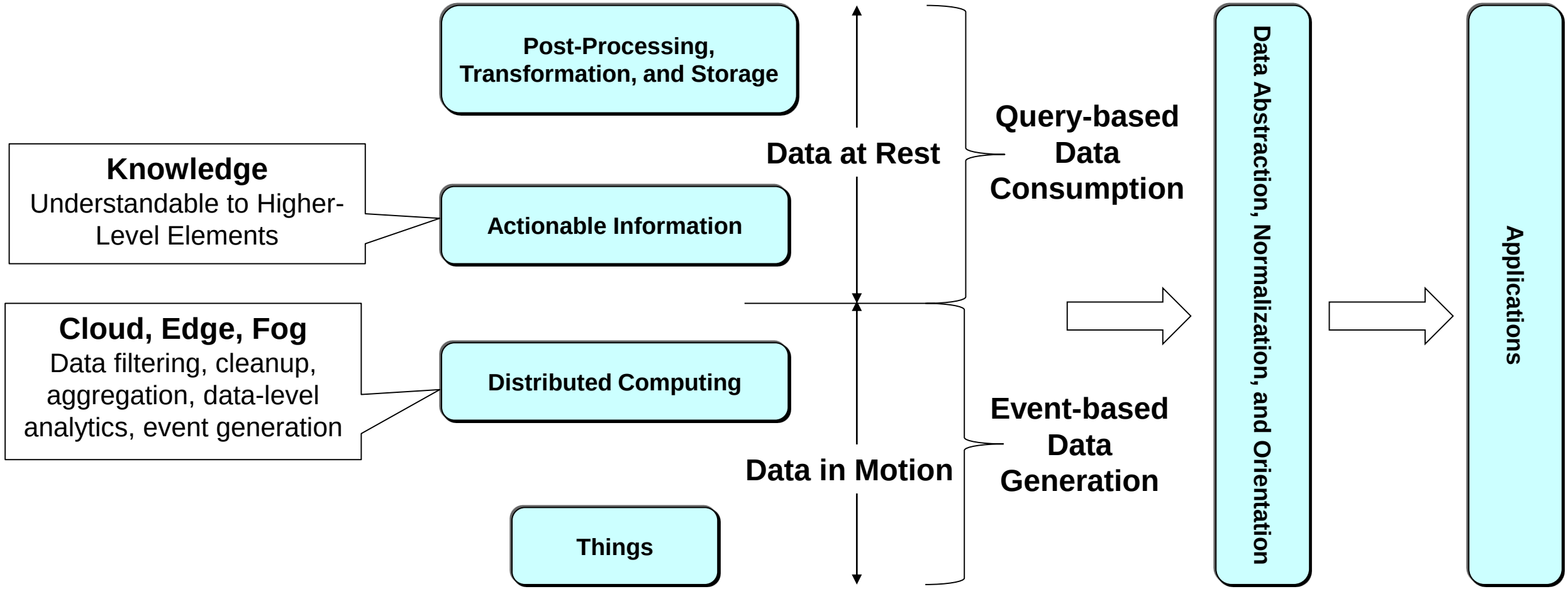


The Semantic Sensor Ontology Pattern

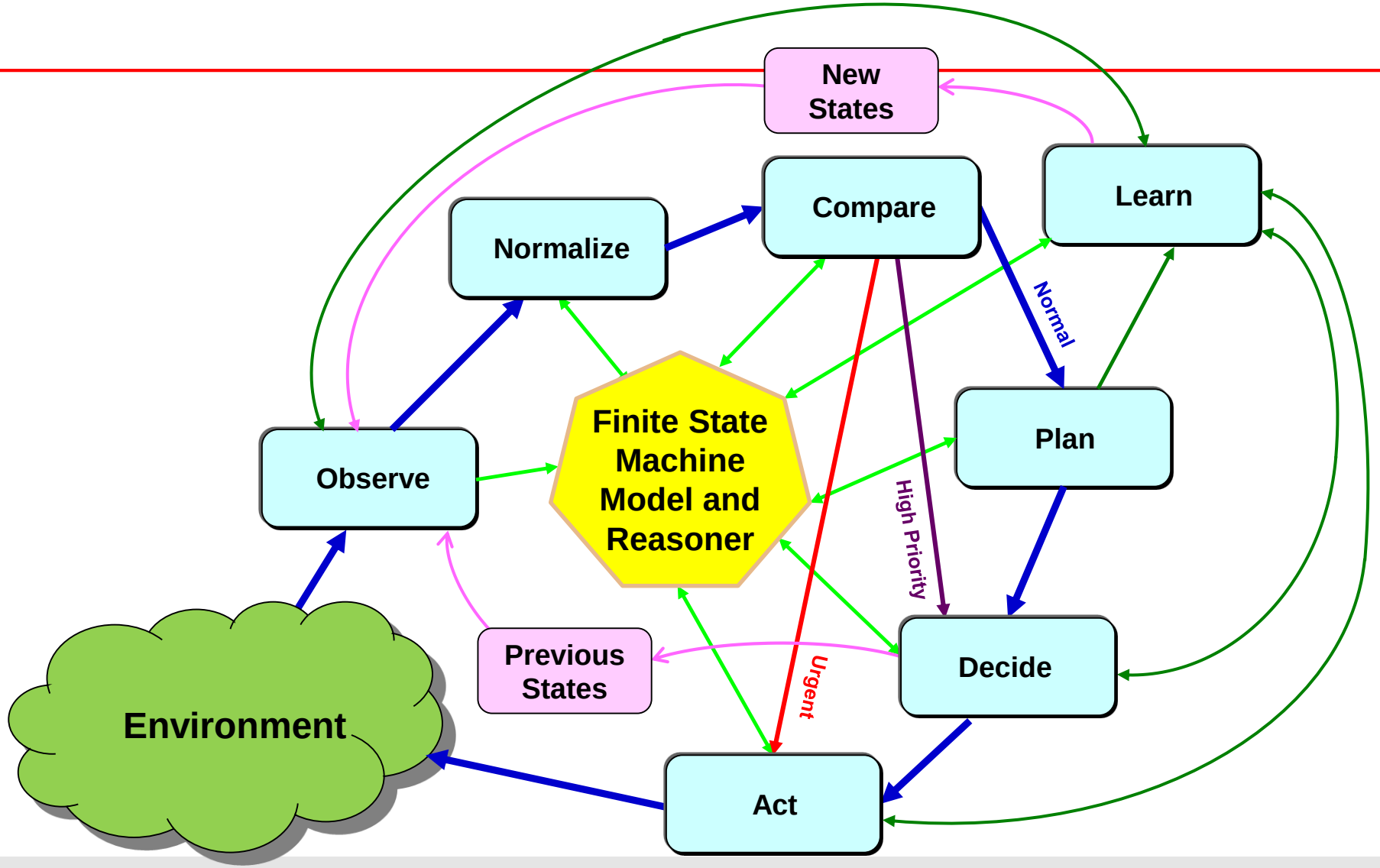


Ref [4]

High-Level Semantic Architecture



FOCALE Cognition Cycle



The Need for a DSL

- **Models and Languages**
  - "A Model is a description of a (part of a) system written in a well-defined language" [8]
  - Well-defined is a language with syntax and semantics suitable for automated interpretation by a computer
- **An Analogy**
  - Model: sentence in a language
  - Metamodel: definition of grammar for the language
  - Metametamodel: definition of similar grammars

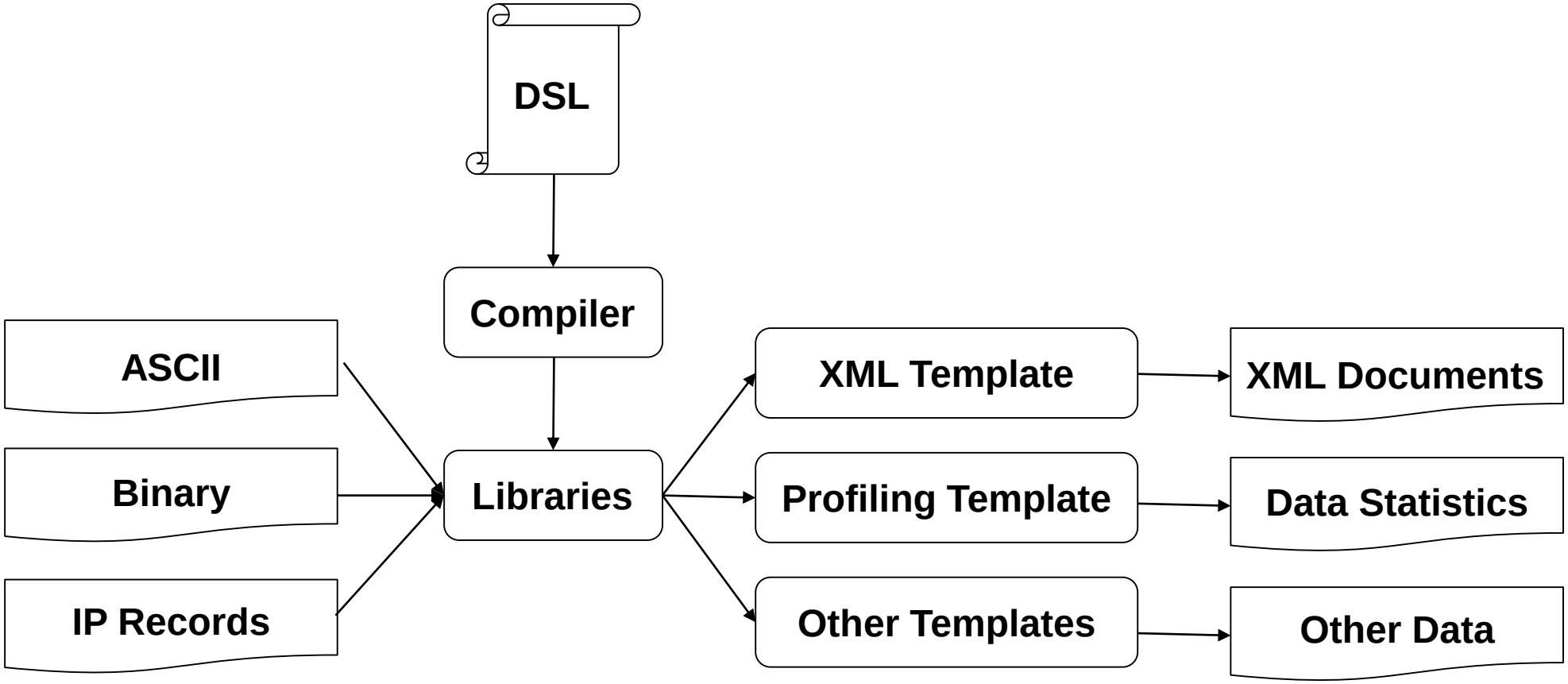
Overview of DSLs (1)

- **A DSL** is a **focused**, **processable language** for describing a set of specific **concerns** when building a system in a specific **domain**
  - The abstractions, terminology, and symbology used are natural for the stakeholders who specify that particular concern
  - A DSL can efficiently represent the characteristics and behavior of a domain
- **We Are Interested in Declarative DSLs**
  - Provides linguistic abstractions for common patterns and idioms of a language when used within its domain
  - Example:
    - for (int i=0; i<array.size(); i++) {…} // requires semantic analysis
    - for (int i in array) {…} // represents semantics declaratively

Overview of DSLs (2)

- **DSLs Simplify Tasks of Domain Experts**
  - Improve reliability with shorter, easier to understand programs
  - Restricting expressiveness simplifies validation and optimization
  - Enables program to serve as documentation
- **DSLs Usage**
  - Declarative specification of data sources
  - Declarative specification of policy-based behavior
  - Post-process into specific tools using template libraries

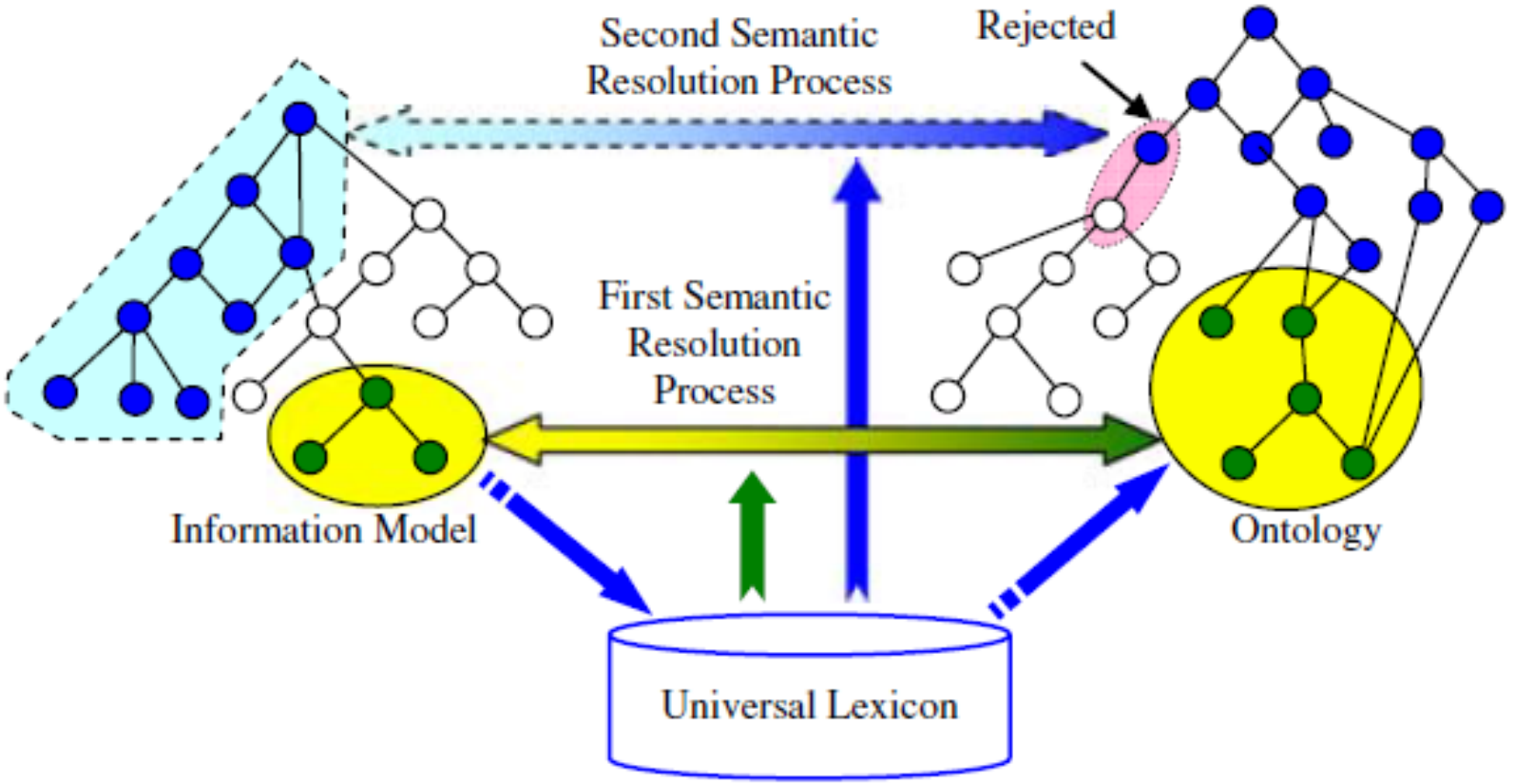
DSL-based Tools



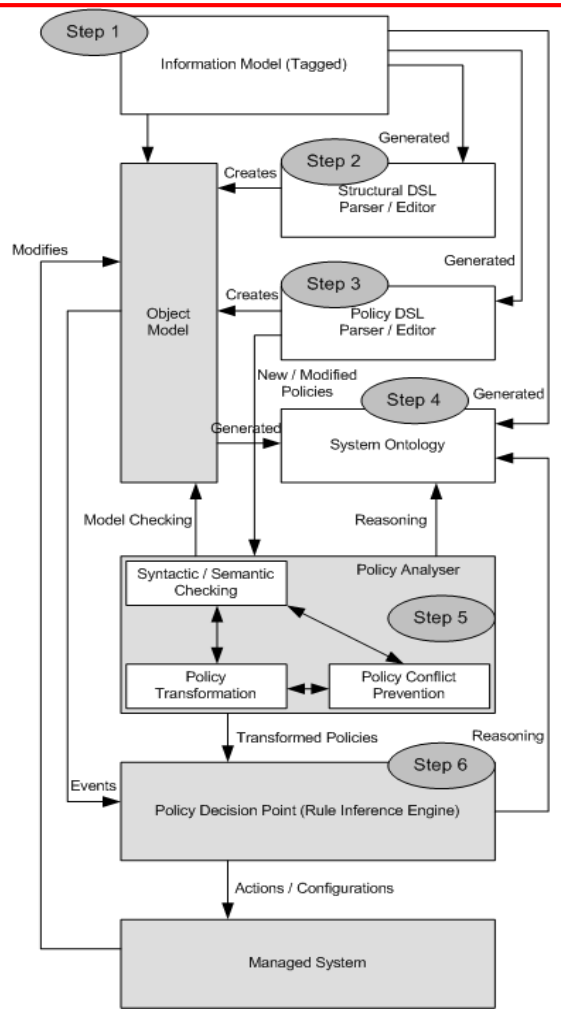
Understanding T2T and IoT Data

- **What About Data Whose Schema-level Understanding Is Missing**
  - e.g, raw tables, graphs, xml, logs, new machine data that has not been modeled
- **Such Data Needs Semantics for Interpretation**
  - Semantics can be used to "match" unknown data
    - Available from the Web, from domain-specific knowledge bases (e.g., clinical trials or other types of experiments), and industrial ontologies
  - Different semantic measures provide different levels of confidence
  - If data doesn't match…
  - …use large background knowledge bases (e.g., Freebase) and relax the level of semantic matching used
  - …but will inevitably have to manually engineer some knowledge bases

Exemplary Semantic Resolution Process



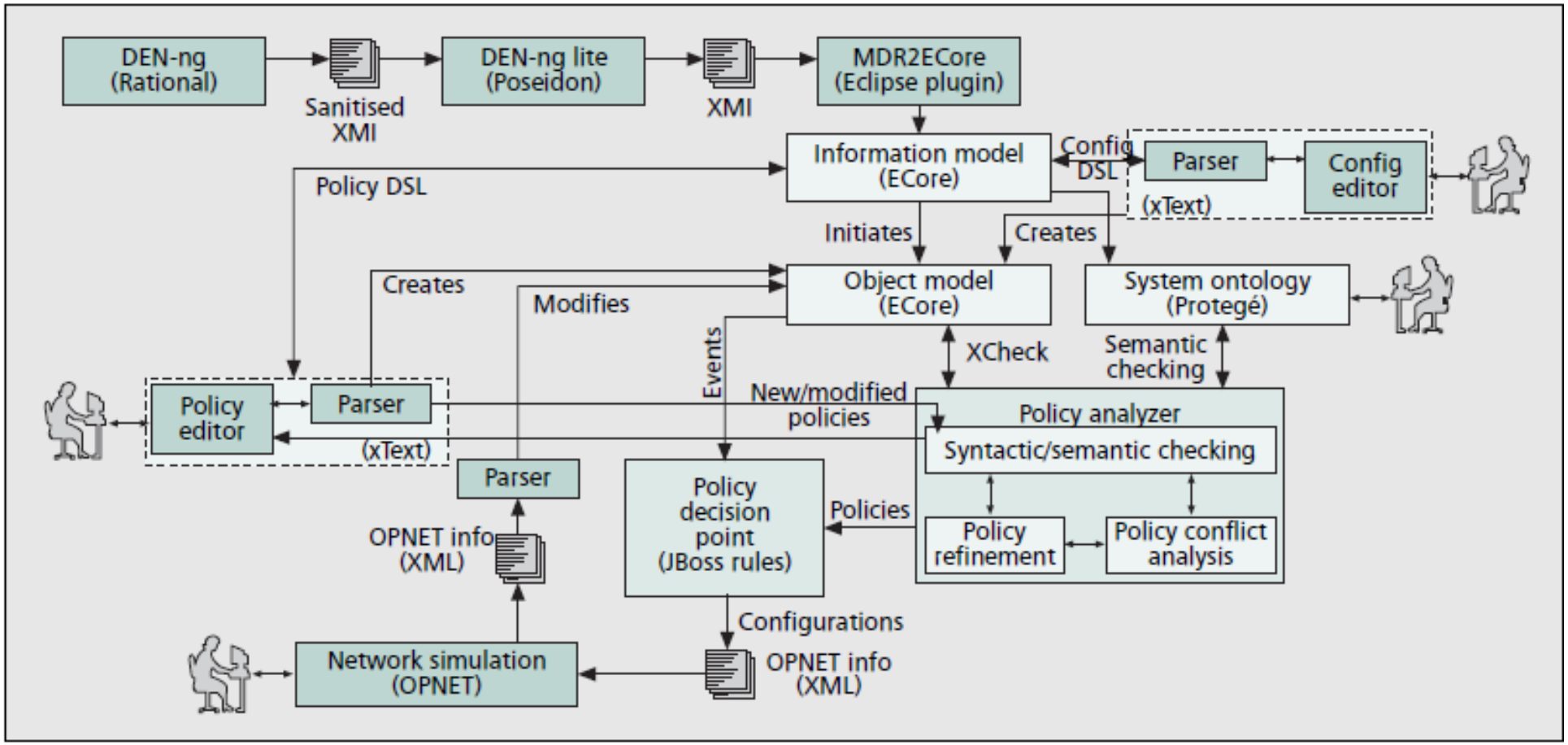
Model-driven Policy-based Management



- **Step 1: Define Model Subset**
  - Tagged-value pairs define a particular model
- **Step 2: Generate Structural DSL, Editor, and Parser**
  - Model Defines Grammatical Syntax and Terms
- **Step 3: Generate Policy DSL, Editor, and Parser**
  - Policy part of model defines behavior between model elements
- **Step 4: Generate a Matching Ontology**
  - Reasoning is used to overcome UML limitations
- **Step 5: Use DSLs and Tools for Policy Analysis**
  - Policy transformation and conflict detection/remediation
- **Step 6: Deploy the System**
  - Uses knowledge from models and ontologies to adapt system behavior

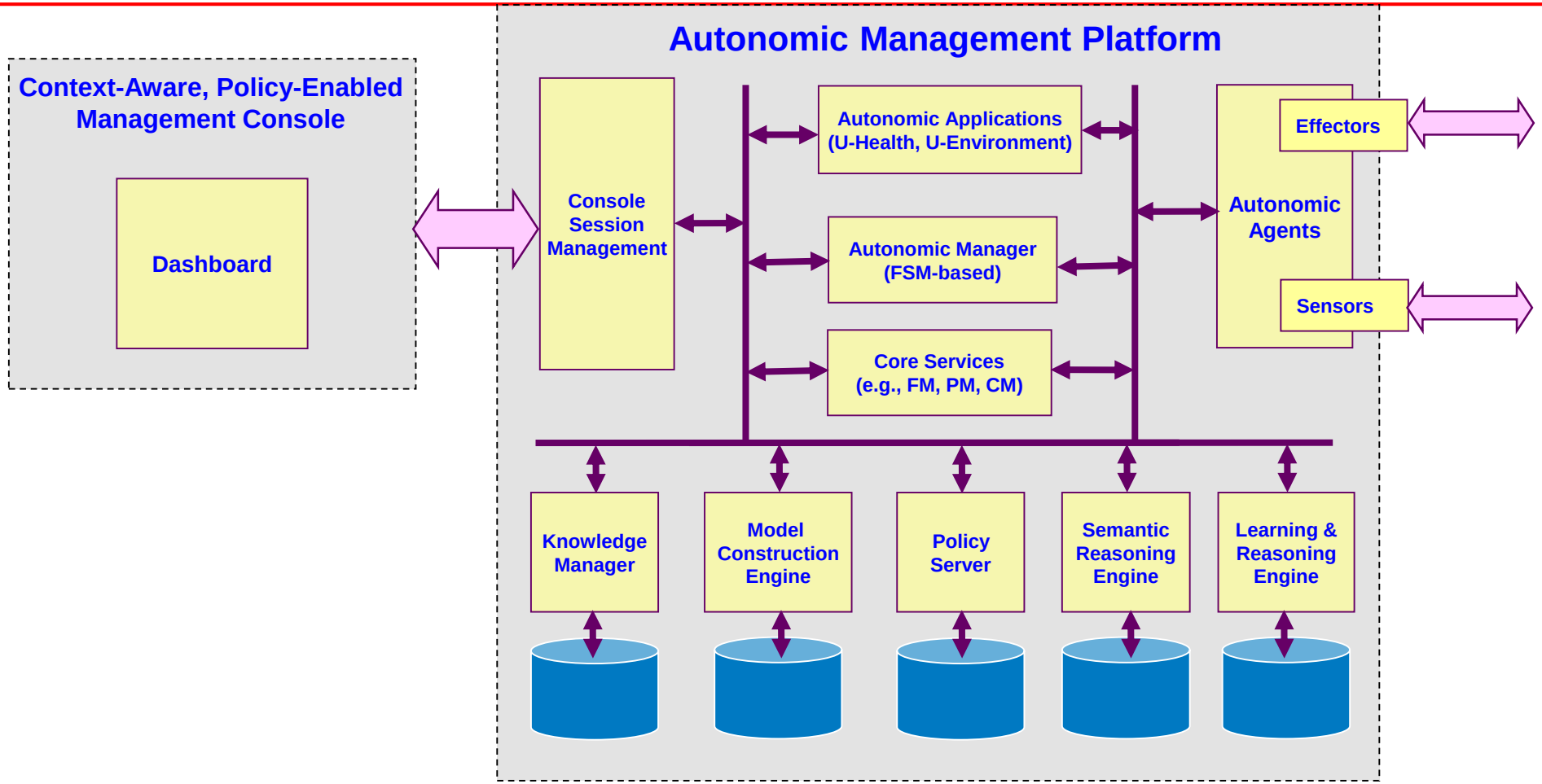
Ref [5]

Exemplar Implementation



Ref [6]

Autonomic Engine



References

[1] A. Dey, “Providing architectural support for building context-aware applications”, Ph.D. Thesis (2000)

[2] J. Strassner, S. van der Meer, D. O’Sullivan, S. Dobson, “The Use of Context-Aware Policies and Ontologies to Facilitate Business-Aware Network Management”, JNSM (17), pp 255-284, 2009

[3] J. Strassner, J. Halpern, J. Coleman, “Generic Policy Information Model for Simplified Use of Policy Abstractions (SUPA)”, draft-strassner-supa-generic-policy-info-model-04

[4] W3C, “Semantic Sensor Network XG Final Report”, http://www.w3.org/2005/Incubator/ssn/XGR-ssn-20110628

[5] K. Barrett, S. Davy, J. Strassner, B. Jennings, S. van der Meer, “Model Based Generation of Integrated Suites of Languages and Tools for Policy Specification, Analysis and Deployment”, IEEE Global Information Infrastructure Symposium, 2007

[6] B. Jennings, S. van der Meer, S. Balasubramaniam, D. Botvich, J. Strassner, M. Ó Foghlú, W. Donnelly, J. Strassner, “Towards Autonomic Management of Communication Networks”, IEEE Communications Magazine, Vol 45., No 10, pp 112-121, Oct 2007

[7] J. Strassner, N. Agoulmine, E. Lehtihet, “FOCALE – A Novel Autonomic Networking Architecture”, International Transactions on Systems, Science, and Applications (ITSSA) Journal, Vol. 3, No 1, pp 64-79, May, 2007

[8] A. Kleppe, J. Warmer, W. Bast, “MDA Explained. The Model Driven Architecture: Practice and Promise”, Addison-Wesley, 2003.

Questions?

“Create like a god. Command like a king. Work like a slave”

- Constantin Brancusi