Declarative Policies for DSA

Mieczyslaw (“Mitch”) M. Kokar

http://www.ece.neu.edu/faculty/kokar

http://www.vistology.com

Northeastern University

Objectives

- Discuss basic concepts of Policy Based Dynamic Spectrum Access Radio System PBDRS (terminology of IEEE P1900.5)
- Discuss computer infrastructure to support the implementation of PBDRS
  - Languages - representation
  - Tools – automatic inference
- Explain the approach
  - Discuss scenarios for the purpose of evaluating/explaining the approach
- Cognitive Radios: Beyond DSA

Spectrum Shortage

- Spectrum is a valued resource
- Shortage
- But at the same time inefficient utilization (next slide)

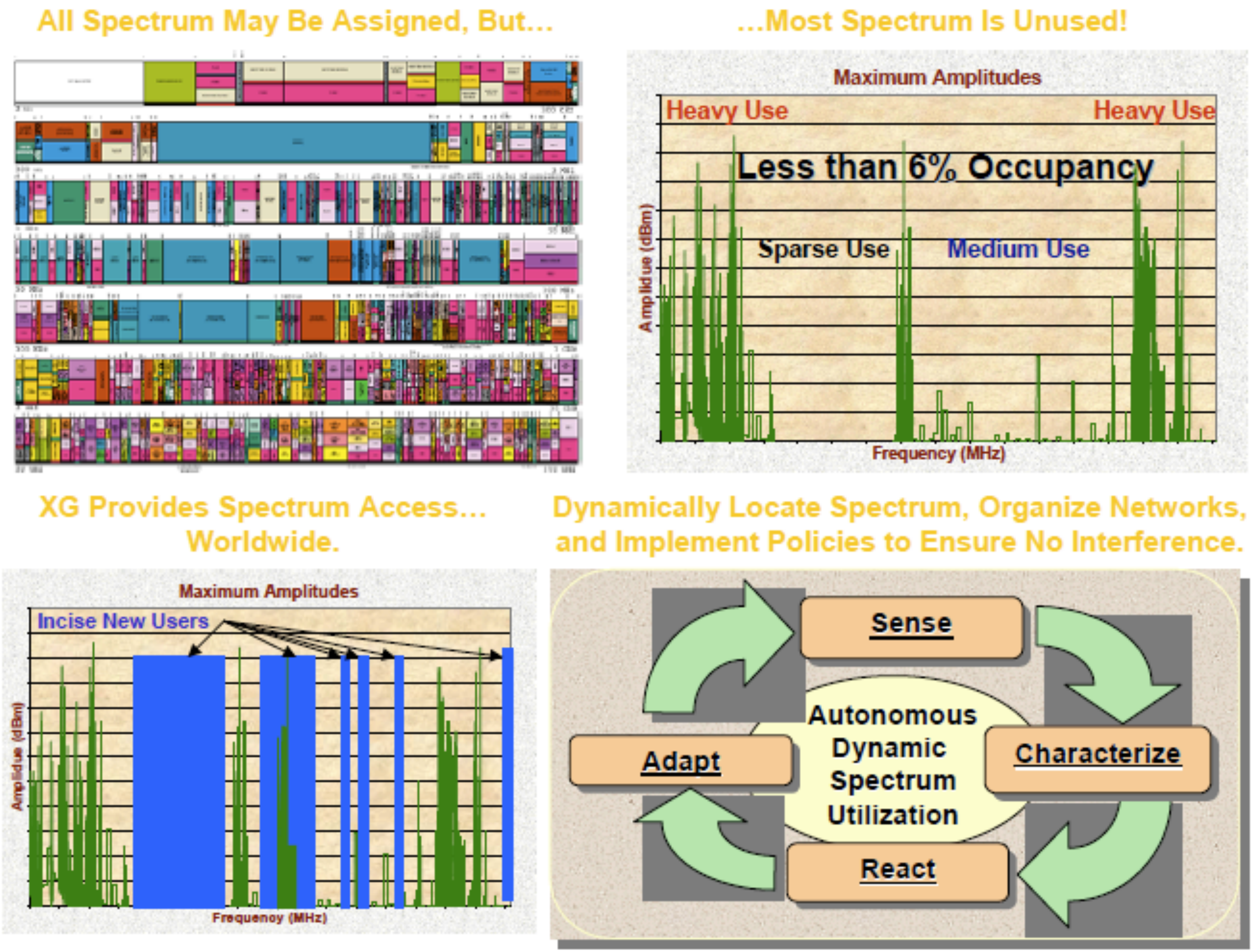


Figure 1: Spectrum is wasted. Opportunistic spectrum access can provide 10x improvement by reusing wasted spectrum.

Tactical Battlefield Spectrum Management

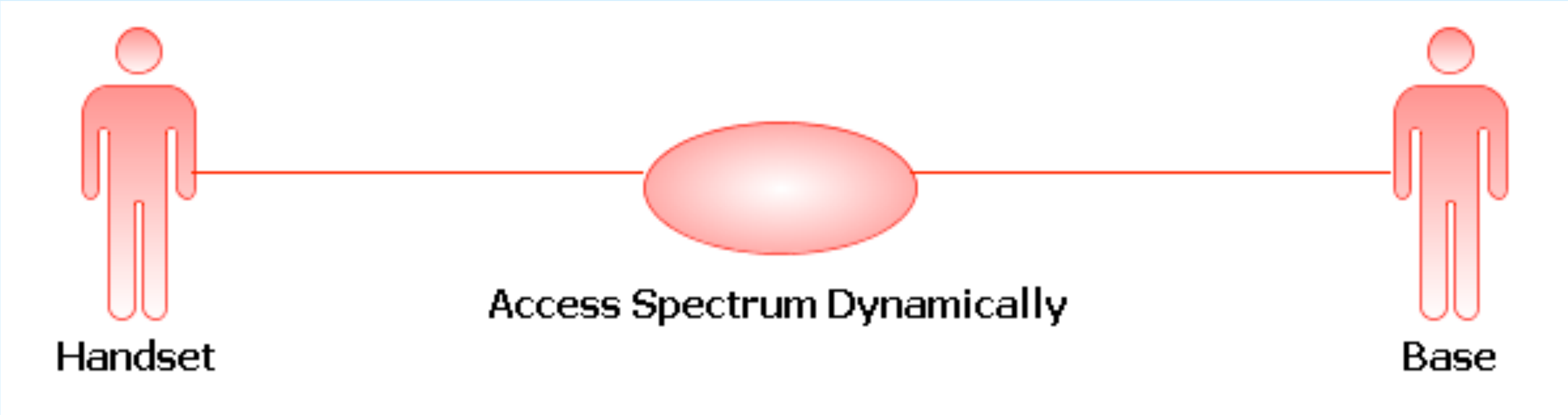
- *US Army Field Manual FM 24-2, Chapter 4*:
  In developing to the greatest extent possible a conflict-free electromagnetic spectrum usage plan, comprehensive and current information on emitter characteristics and frequency availability is essential. Currently, spectrum management is largely a manual process. However, with the arrival of automated systems, maintenance of the data base for this information becomes a simpler and easier task and a more efficient process.
- But still not automated …

Approaches to DSA

- Centralized spectrum management
  - Policy at the controller
  - Database of assignments
- Decentralized spectrum management; aka Dynamic Spectrum Selection (DSS)
  - Sensing and selection
  - Policy interpretation is needed
- Should assignments be “fair”?
  - Tactical Battlefield DSA
  - Public safety
  - Role based, attribute based assignments
- Should policies be fixed over time, space, roles, attributes?

Centrally Controlled DSA

DSA: a Use Case

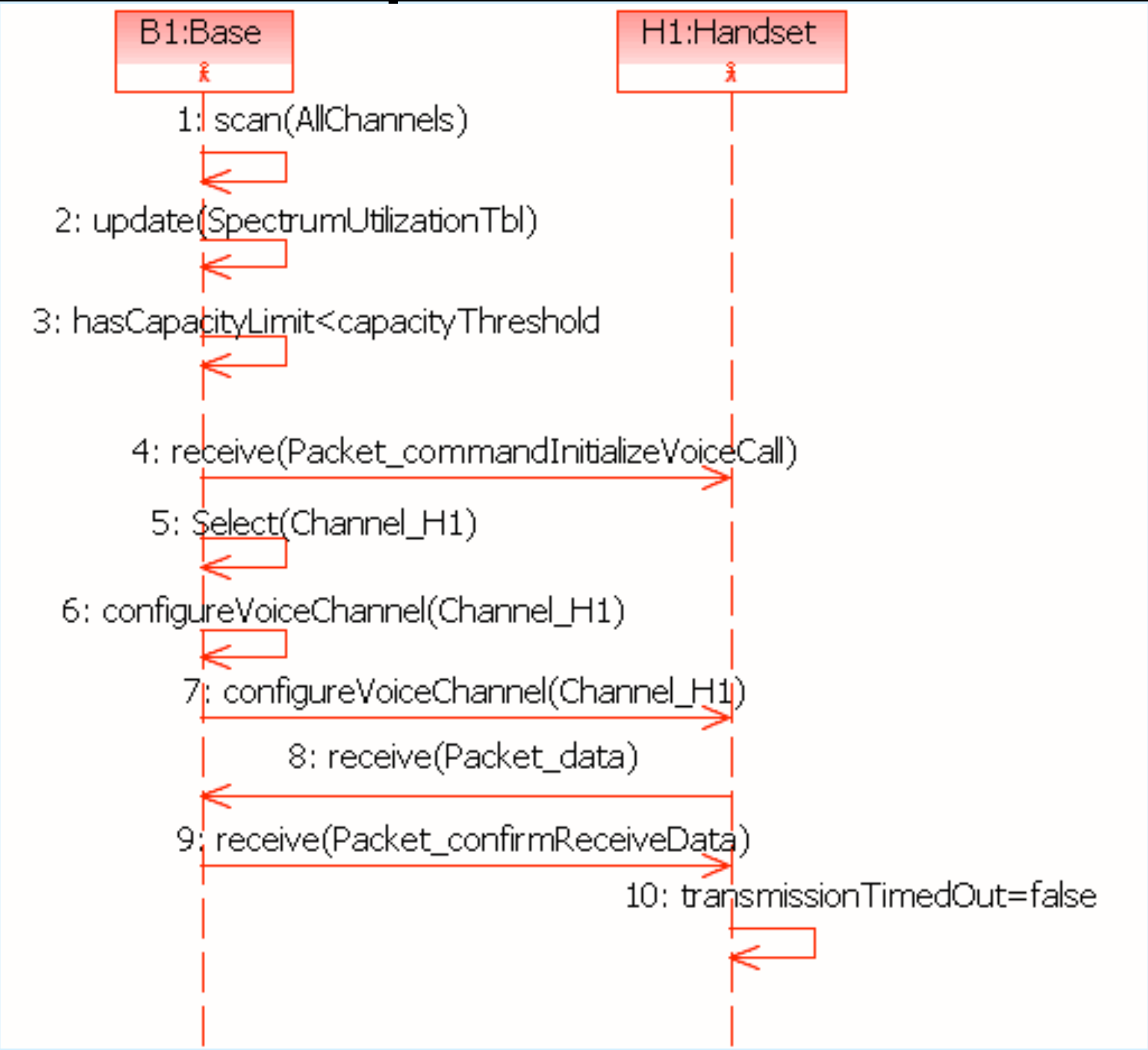


"Use Cases for MLM Language in Modern Wireless Networks," SDR Forum, SDRF-08-P-009-V1.0.0, Jan-09

Sequence of Events (*BS* and *HS1* follow own policies)

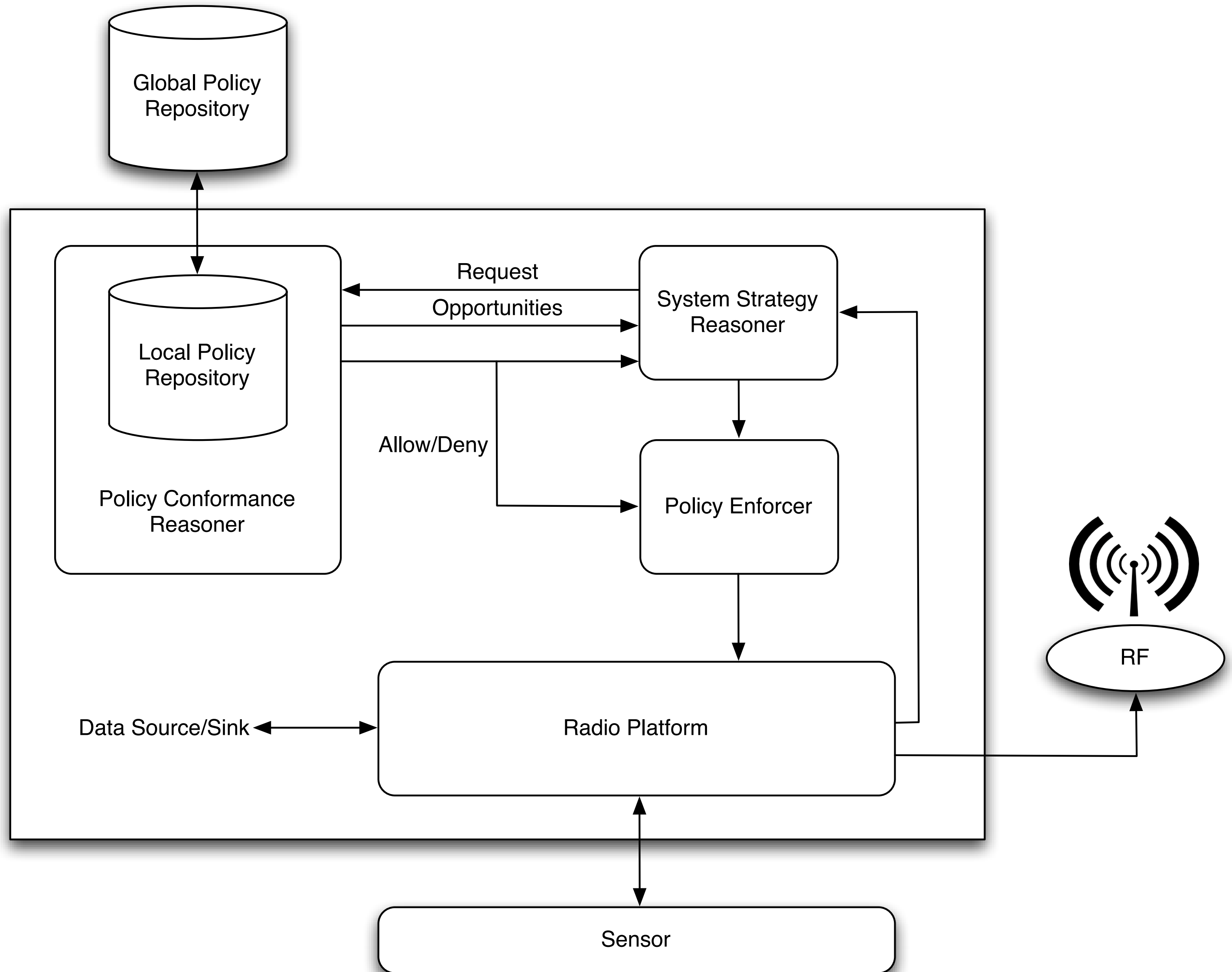
1. *BS* scans environment and updates spectrum utilization (*SpectrumUtilizationTbl*)
2. When *capacity=overloaded*, DSA is triggered
3. User's handset *H1* requests from *BS* a voice call initialization
4. *BS* uses its current policy to select *channel_H1* as forward and reverse channel pair
5. BS sends a *command* message to *H1* and configures *channel_H1* as *H1's* reverse voice channel
6. BS configures *channel_H1* as its reverse voice channel
7. *H1* sends an inquiry to *BS* whether it is ready to receive data
8. *BS* sends a "*confirmed*" message to *H1*
9. *H1* starts to send data traffic to *BS*
10. When *BS* receives the data, it sends an acknowledgment back to *H1*
11. If *H1* receives the acknowledgment within a predefined time, then data transmission is finished. Otherwise, the data transmission is considered lost and steps 3 to 10 are repeated.

DSA: Sequence of Events



"Use Cases for MLM Language in Modern Wireless Networks," SDR Forum, SDRF-08-P-009-V1.0.0, Jan-09

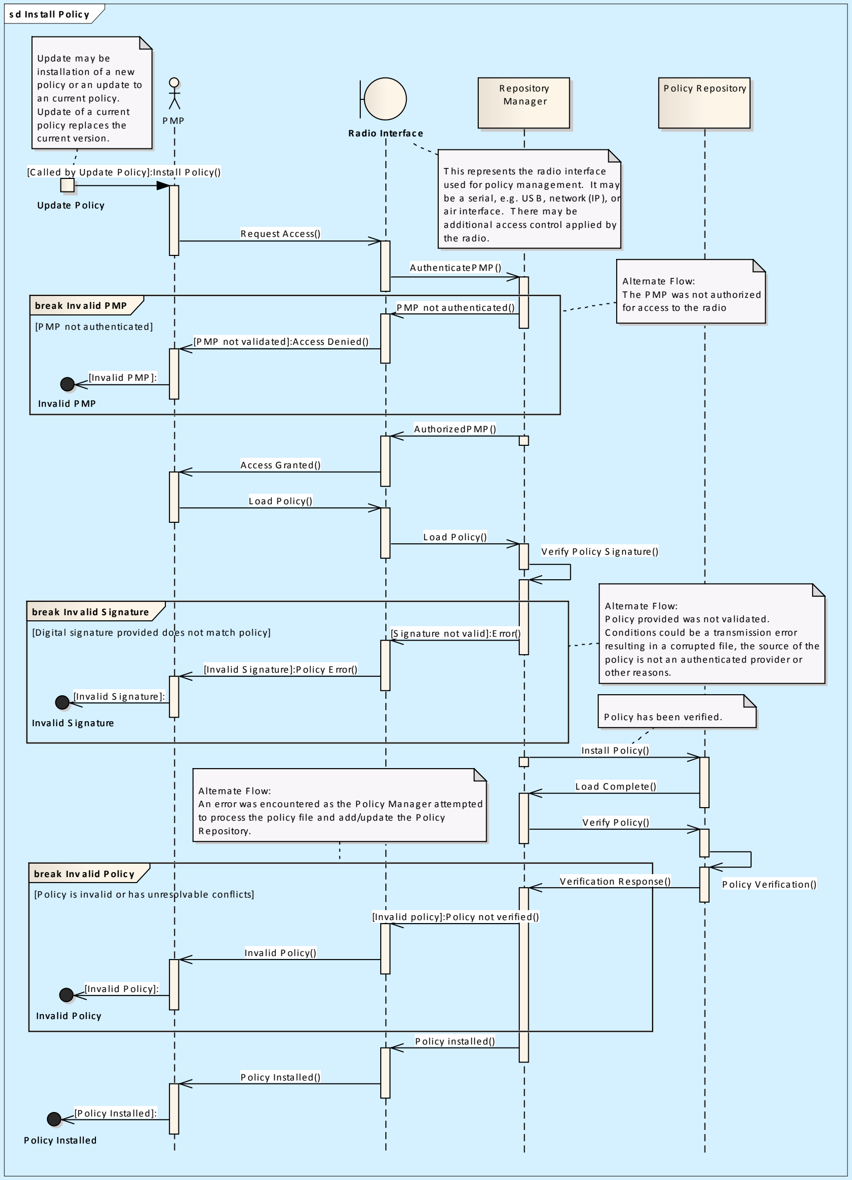
Architecture of PBDRS

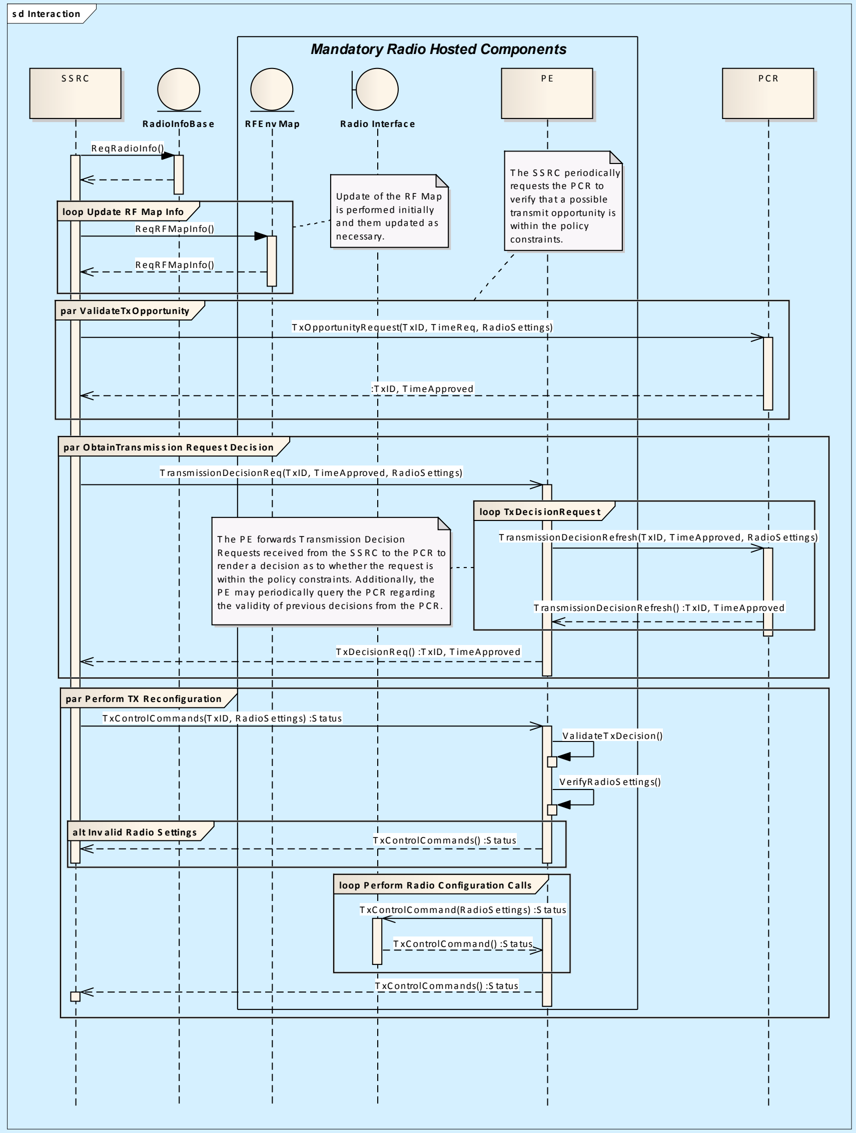


Policy Load and Conformance

- Policies can be loaded and modified at run time
- Policy sources can be verified
- Policies can be checked for validity
- Then requests for transmission can be verified against policies
- The sequence diagram below was contributed to P1900.5 by Vince Kovarik (diagram)

The diagram is better viewed in a separate pdf file.


sd Install Policy
Update may be installation of a new policy or an update to an current policy. Update of a current policy replaces the current version.
PMP
Radio Interface
Repository Manager
Policy Repository
This represents the radio interface used for policy management. It may be a serial, e.g. USB, network (IP), or air interface. There may be additional access control applied by the radio.
[Called by Update Policy]:Install Policy()
Update Policy
Request Access()
AuthenticatePMP()
Alternate Flow:
The PMP was not authorized for access to the radio
break Invalid PMP
[PMP not authenticated]
PMP not authenticated()
[PMP not validated]:Access Denied()
[Invalid PMP]:
Invalid PMP
AuthorizedPMP()
Access Granted()
Load Policy()
Load Policy()
Verify Policy Signature()
break Invalid Signature
[Digital signature provided does not match policy]
[Signature not valid]:Error()
[Invalid Signature]:Policy Error()
[Invalid Signature]:
Invalid Signature
Alternate Flow:
Policy provided was not validated. Conditions could be a transmission error resulting in a corrupted file, the source of the policy is not an authenticated provider or other reasons.
Policy has been verified.
Install Policy()
Load Complete()
Verify Policy()
Alternate Flow:
An error was encountered as the Policy Manager attempted to process the policy file and add/update the Policy Repository.
break Invalid Policy
[Policy is invalid or has unresolvable conflicts]
Verification Response()
Policy Verification()
[Invalid policy]:Policy not verified()
Invalid Policy()
[Invalid Policy]:
Invalid Policy
Policy installed()
Policy Installed()
[Policy Installed]:
Policy Installed


sd Interaction
Mandatory Radio Hosted Components
SSRC
RadioInfoBase
RFEnvMap
Radio Interface
PE
PCR
ReqRadioInfo()
loop Update RF Map Info
ReqRFMapInfo()
ReqRFMapInfo()
Update of the RF Map is performed initially and them updated as necessary.
The SSRC periodically requests the PCR to verify that a possible transmit opportunity is within the policy constraints.
par ValidateTxOpportunity
TxOpportunityRequest(TxID, TimeReq, RadioSettings)
:TxID, TimeApproved
par ObtainTransmission Request Decision
TransmissionDecisionReq(TxID, TimeApproved, RadioSettings)
loop TxDecisionRequest
TransmissionDecisionRefresh(TxID, TimeApproved, RadioSettings)
The PE forwards Transmission Decision Requests received from the SSRC to the PCR to render a decision as to whether the request is within the policy constraints. Additionally, the PE may periodically query the PCR regarding the validity of previous decisions from the PCR.
TransmissionDecisionRefresh() :TxID, TimeApproved
TxDecisionReq() :TxID, TimeApproved
par Perform TX Reconfiguration
TxControlCommands(TxID, RadioSettings) :Status
ValidateTxDecision()
VerifyRadioSettings()
alt Invalid Radio Settings
TxControlCommands() :Status
loop Perform Radio Configuration Calls
TxControlCommand(RadioSettings) :Status
TxControlCommand() :Status
TxControlCommands() :Status

A Policy Example

- Policy: "A node can transmit in frequency range 350.00 – 370.00 MHz between the hours of 1100 and 1200."
- SSR must provide:
  - Operation ID: e.g., T001
  - Operation type: transmit
  - Frequency range: e.g., 355.00-360Mhz
  - Transmit time: e.g., 1105-1110
- PR returns
  - <type T001 allowed>

Policy: Another Example

- Policy: “A node can transmit in the frequency range 350.00 MHz - 370.00 MHz in locations between latitude 37.123456 and 38.123456 and between longitude 68.123456 and 79.123456.”
- SSR must provide:
  - Operation ID: e.g., T002
  - Operation type: transmit
  - Frequency range: e.g., 375.00-380Mhz
  - Transmit location:
    - e.g., latitude=37.234567, longitude=70.123456
- PR returns
  - <type T002 disallowed>

Policy: Another Example

- Policy: “A node can transmit in the frequency range 350.00 MHz - 370.00 MHz if the security label of the node is at least “Secret”.”
- SSR must provide:
  - Operation ID: e.g., T003
  - Operation type: transmit
  - Frequency range: e.g., 375.00-380Mhz
  - Security label:
    - e.g., Security label = Top Secret
- PR returns
  - <type T003 allowed>
  - PR needs means to verify the validity of the security label

How should policies be represented?

- Procedural code (C/C+/Java)
  - Problems with interoperability and modifiability – would require multiple versions of policies for different platforms
  - Rewrite, recompile, reload, re-deploy with any change in policies
  - Possibly would require re-certification of the platform code when policies change
- A language with formal syntax (XML)
  - Requires procedural code to interpret XML tags for each platform
  - Expand the code library for the modified or additional policies
  - Re-certify the code
- A language with formal semantics (OWL, ++?)
  - Requires a generic interpreter running on different platforms
  - Load new sets of policies into local policy base
  - The interpreter does not need to be changed because it is specific to the language and not to the policies
  - No need to re-certify the interpreter when policies change

Need a Common Vocabulary

- For PR to “understand” policies, the policies must be represented in a common vocabulary
- Standardized vocabulary is preferred
- The terms in the Policy Examples in the previous slides highlighted in red are candidates for inclusion in the vocabulary

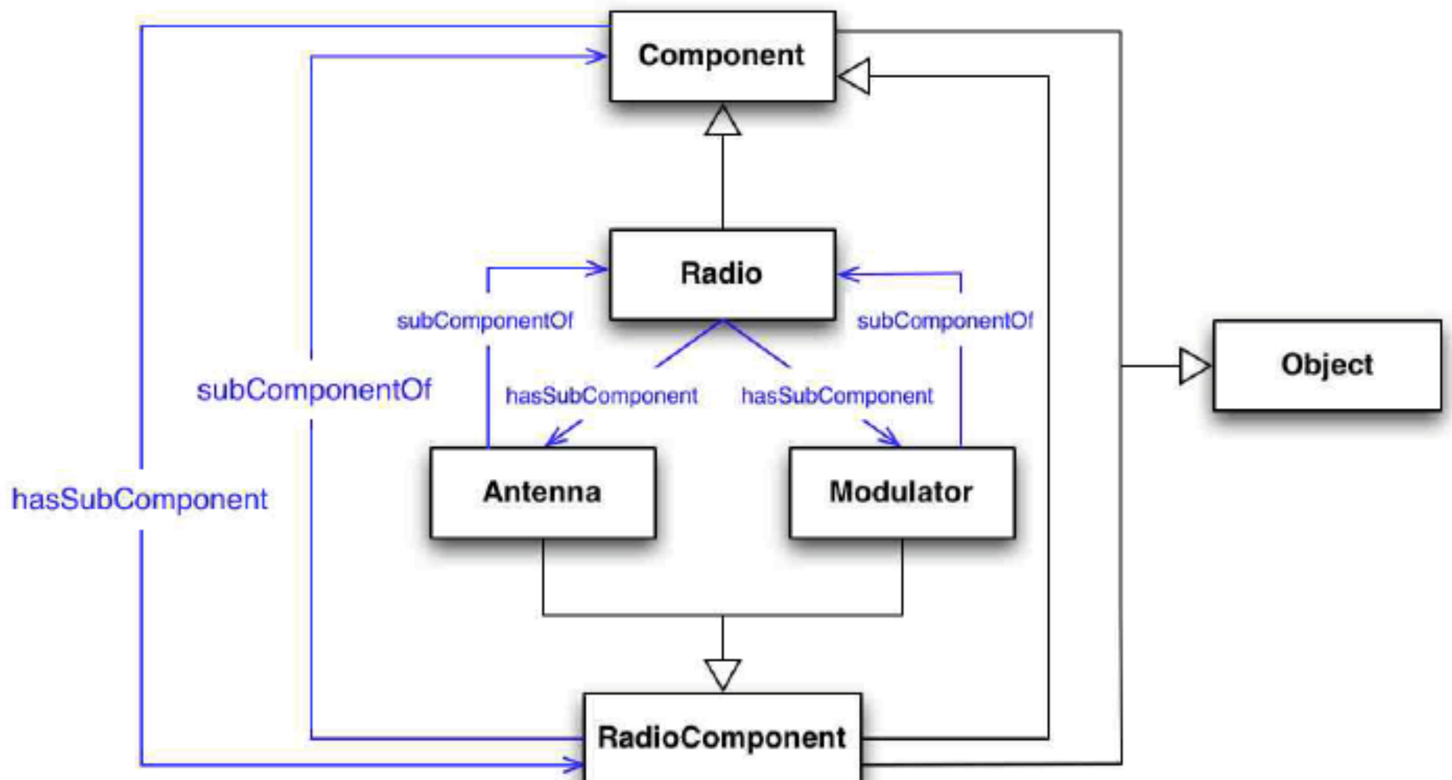
Kinds of Terms in a Vocabulary

- Specific individuals
  - "J.Smith", "10:34", "blue"
- Classes of individuals
  - Frequency, Node, Handset, Time
- Relationships between/among classes of individuals
  - Subclass
    - Handset *subClassOf* Node
  - Aggregation/composition
    - Receiver *partOf* Node
  - Any relationships specific to the domain
    - *hasFrequencyRange*
- Also relations between/among specific individuals
  - J.Smith *partOf* Bravo

Graphical Representation of Classes and Relations

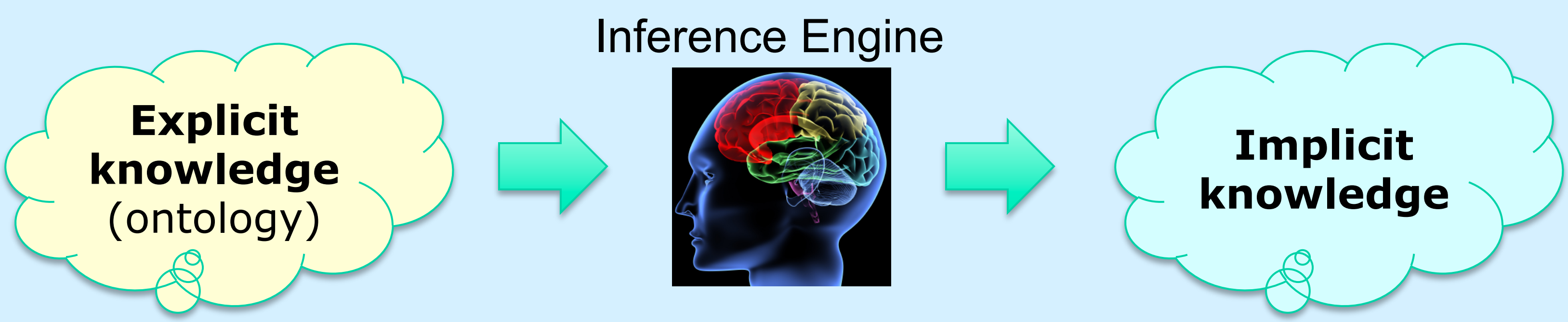
WIRELESS INNOVATION FORUM

Modeling Language for Mobility Work Group
CR Ontology Description
WINNF-10-S-0007-V1.0.0



Formal Ontologies

- Explicit representation of:
  - Concepts (classes, objects)
  - Relationships (relations, properties, attributes)
- Language
  - Formal grammar
  - Machine interpretable semantics (inference capability)



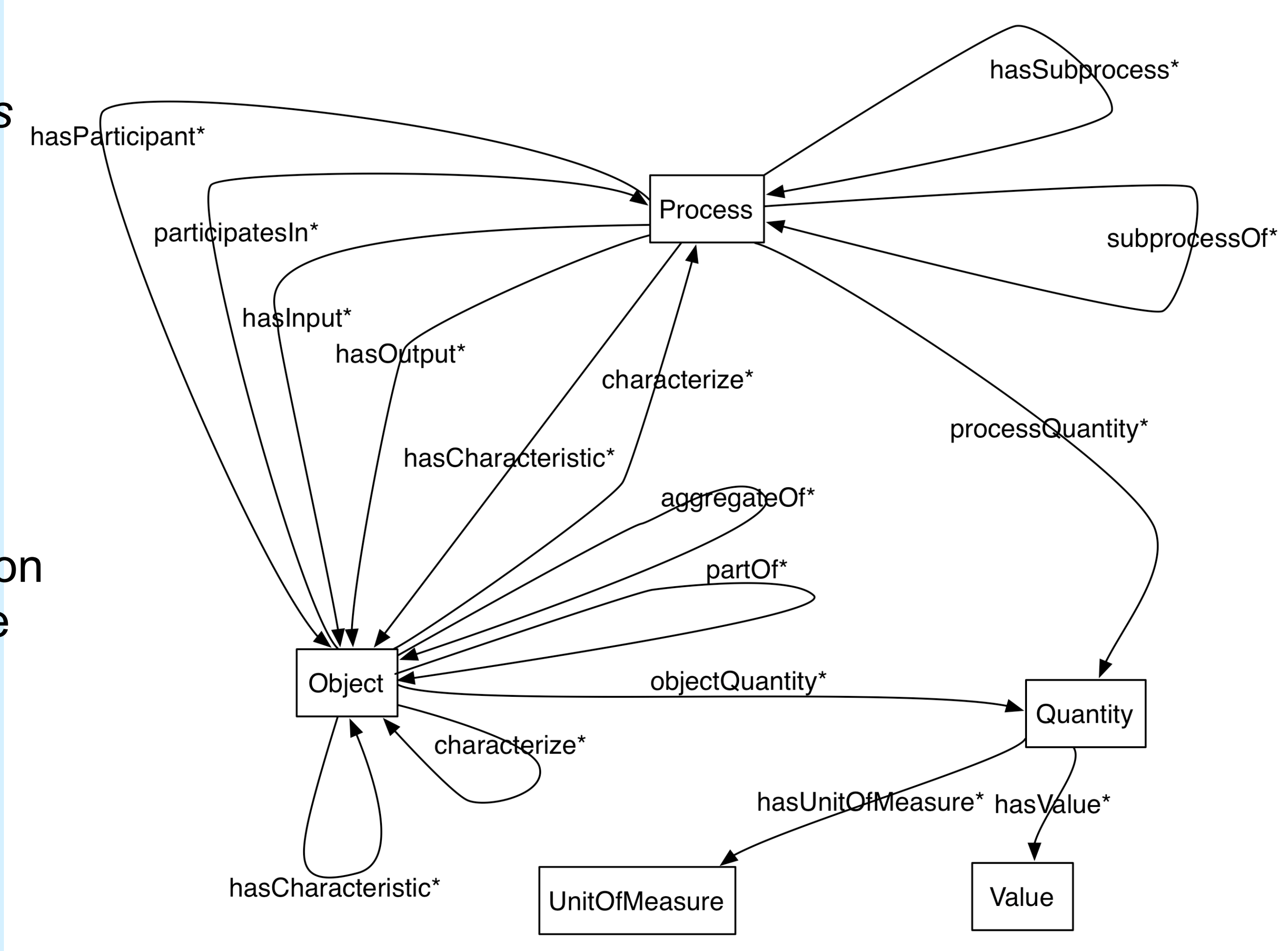
Databases lack this capability.

OWL Representation (a small fragment)

```
<owl:Class rdf:ID="Modulator">
  <rdfs:subClassOf rdf:resource="#RadioComponent"/>
 </owl:Class>
 <owl:Class rdf:about="#InputPort">
  <rdfs:subClassOf rdf:resource="#Port"/>
 </owl:Class>
<owl:Class rdf:about="#RadioComponent">
  <rdfs:subClassOf>
   <owl:Restriction>
    <owl:onProperty>
     <owl:ObjectProperty rdf:ID="hasSubComponent"/>
    </owl:onProperty>
    <owl:allValuesFrom>
     <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
       <owl:Class rdf:ID="BasicComponent"/>
       <owl:Class rdf:about="#RadioComponent"/>
      </owl:unionOf>
     </owl:Class>
    </owl:allValuesFrom>
   </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
   <owl:Class rdf:ID="Component"/>
  </rdfs:subClassOf>
 </owl:Class>
```

CRO: Cognitive Radio Ontology

- Based on concept of *Objects* and *Processes*
  - *Derived from DOLCE*
  - http://www.loa-cnr.it/DOLCE.html
- Objects represent entities that have state
- Processes represent events or actions that perform a transformation or state change on one or more objects.



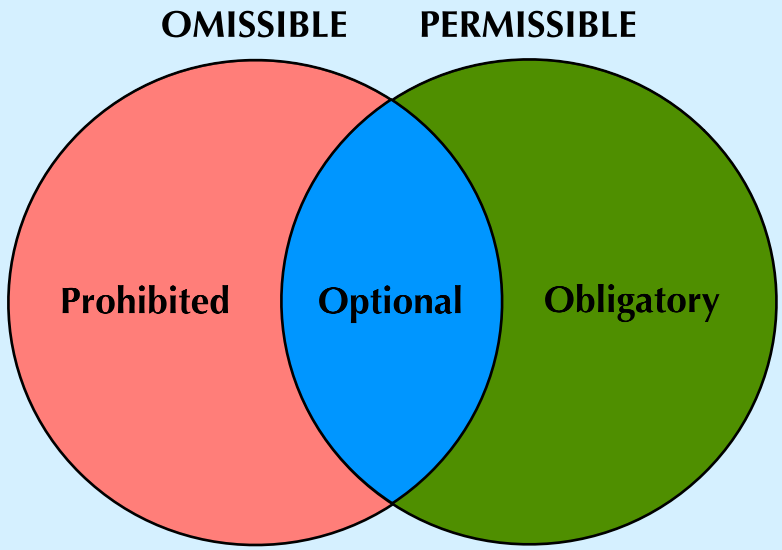
CRO: Cognitive Radio Ontology

- Document: "*Description of the Cognitive Radio Ontology*"
- Approved at the WIF 67th Working Meeting, Schaumburg, Sep. 2010
- Available here: http://groups.winnforum.org/d/do/3370)
  - Core ontology (covering basic terms of wireless communications from the PHY and MAC layers)
  - Concepts needed to express the MLM use cases developed earlier
  - Partial expression of the FM3TR waveform (structure and subcomponents, FSM)
  - Partial expression of the Transceiver Facility APIs
  - 230 classes and 188 properties
- Work is underway to extend CRO in order to express concepts of Model Based Spectrum Management (MBSM) – see MITRE Technical Report number 110131, May 10, 2011.

Deontic Action Ontology

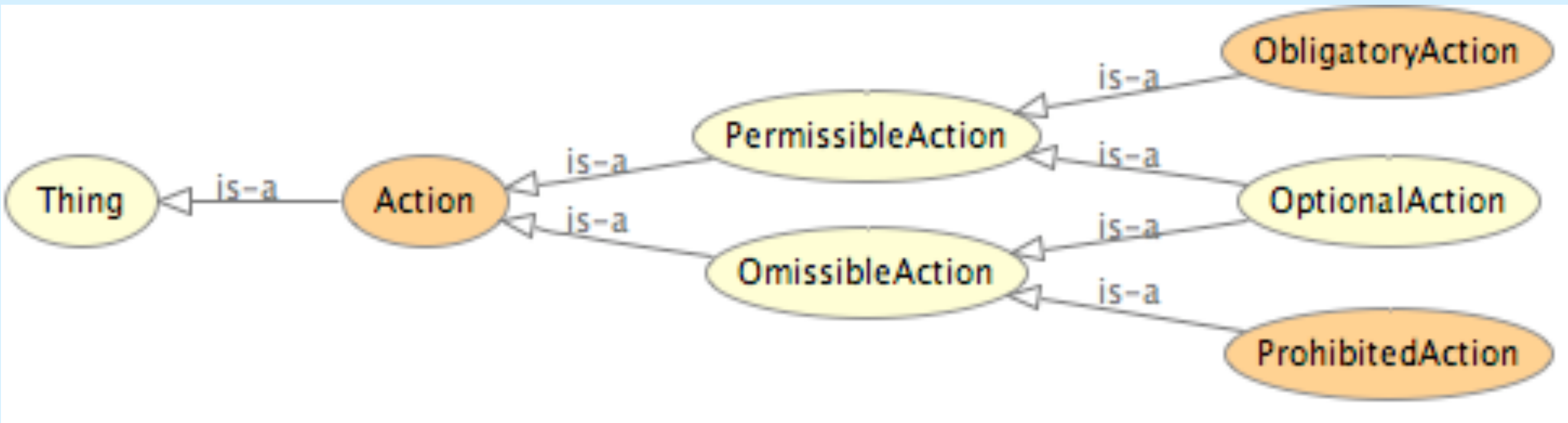
Operations/Actions

PEp ↔ ~OB~p
PRp ↔ OB~p
OMp ↔ ~Obp
OPp ↔ (~OBp & ~OB~p).



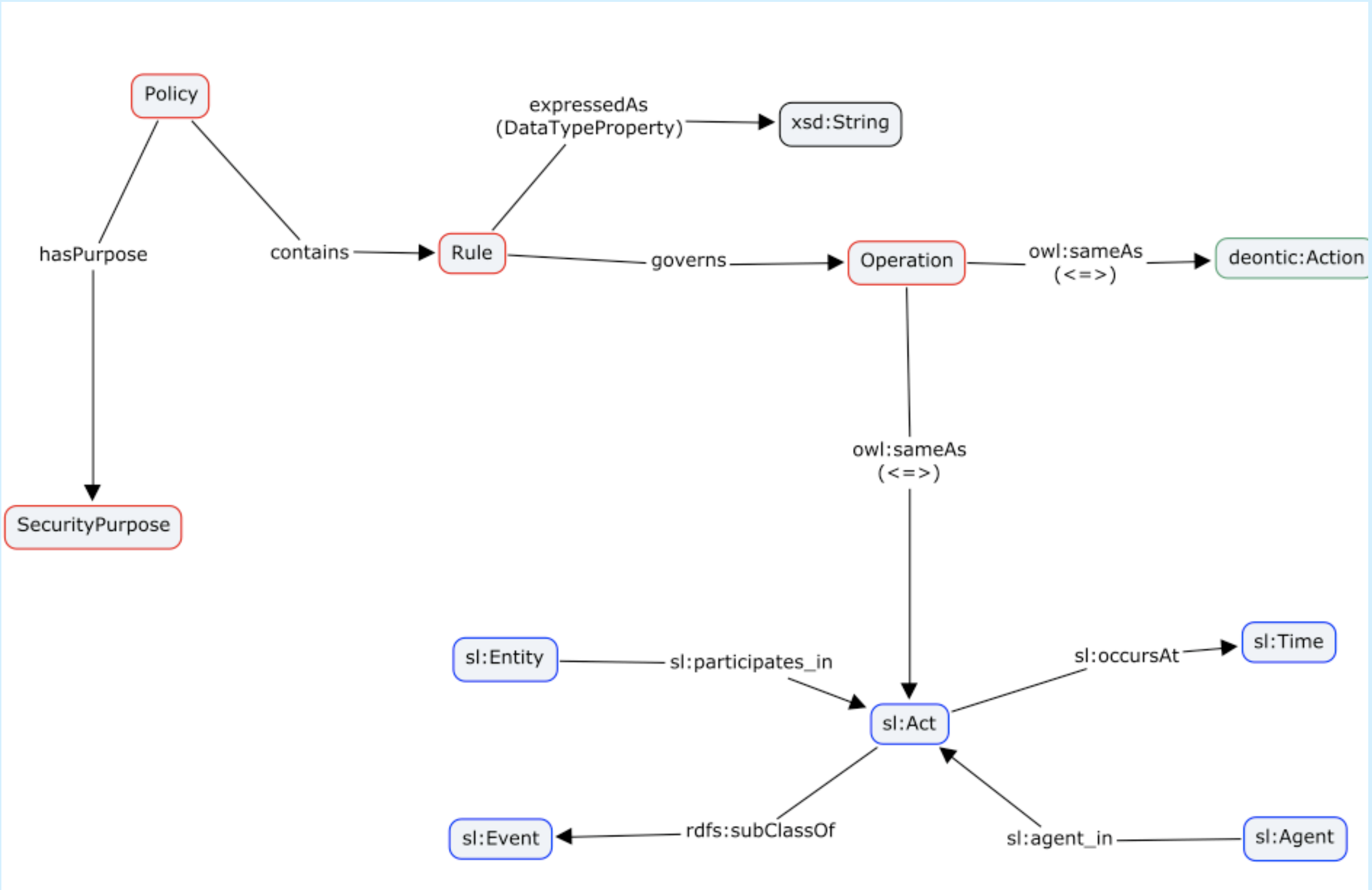
Expressed in OWL using disjointClass, subClassOf, complementOf, intersectionOf…

Deontic Action Ontology



Expressed in OWL using disjointClass, subClassOf, complementOf, intersectionOf…

Policy Ontology



Expressing Policies in Restricted Natural Language

- PolVISor – an engine for editing and processing policies
- PolVISor uses policies expressed in **SBVR** Structured English
  - An OMG Standard
  - Semantics of Business Vocabulary and Rules
  - Provides constructs for representing modalities:
    - Deontic: "It is obligatory/permitted/forbidden that ..."
    - Alethic: "It is necessary/possible/impossible that …"
  - Has corresponding XML (XMI) representation of logical structure
  - First-order quantification (some/a, all/each/every)
  - Non-standard coindexing:
    - "a boy shaved the boy" means "a boy shaved himself"
  - N-ary predicates represented as "fact types"
    - Binary: read(actor,file), write(), delete()
    - Ternary: send_to(sender,message,recepient), receive_from(), etc

Expressing Security Policies in Restricted Natural Language

- Color/font-coded syntax
  - term : noun concepts (other than individual concepts), e.g. adult
  - Name : individual concepts, e.g. John
  - *verb* : fact types, e.g. *reads*
  - keyword : reserved words or phrases with special meaning in SBVR, e.g. each, it is obligatory that, the, a, and, …

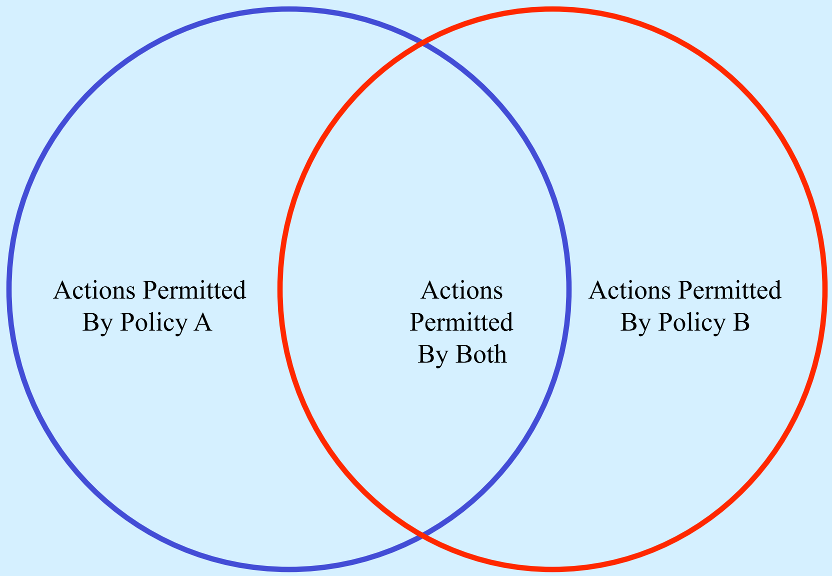
Rule

- It is prohibited that
  a person that *is* not an adult *reads* a pictureWithMatureContent

Policy Editing-Translation-Execution

- Policy Rules written in a subset of Semantics of Business Vocabulary and Rules (SBVR) Structured English (SE)
  - A rule author uses the PolVISor editor to enter the SBVR vocabulary corresponding to the domain ontology to support the rules
  - Then the editor writes rules in that vocabulary
- After converting the SBVR SE rules to XMI, PolVISor can translate the XMI into BaseVISor rule language (BVR) via ATL scripts
  - ATL is a transformation language and execution environment that converts a source model conforming to a source metamodel (e.g. SBVR) to a target model conforming to a target metamodel (e.g. BVR).
- After converting the SBVR XMI to BVR, the resulting BVR rules can be executed by the BaseVISor OWL 2 RL reasoner

Policy Reconciliation



- Incompatible policies can be reconciled (finding a common ground)
- PolVISor

Cognitive Radio: Not just Dynamic Spectrum Access

a) A type of radio in which communication systems are aware of their environment and internal state and can make decisions about their radio operating behavior based on that information and predefined objectives.

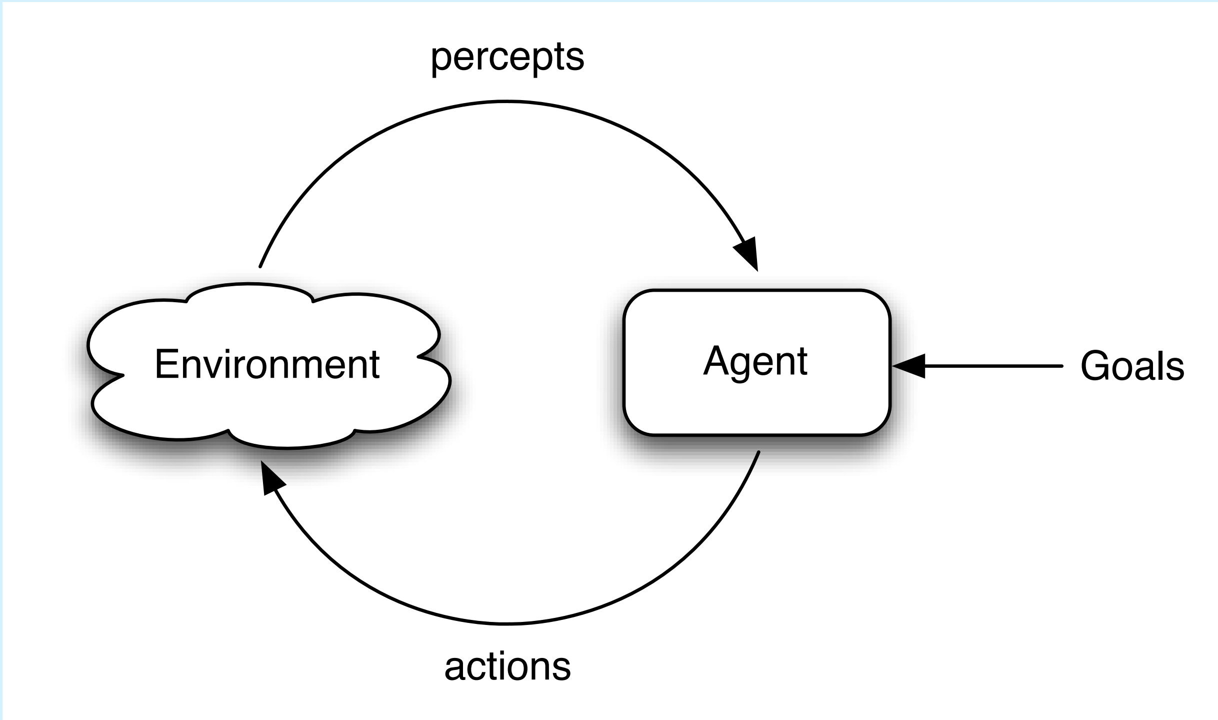
b) Cognitive radio [as defined in item a)] that uses software-defined radio, adaptive radio, and other technologies to adjust automatically its behavior or operations to achieve desired objectives.

Definition started by Mitola, then adopted by Cognitive Radio WG at the Wireless Innovation Forum and IEEE P1900.1 standardization work

Other Cognitive Radio Capabilities

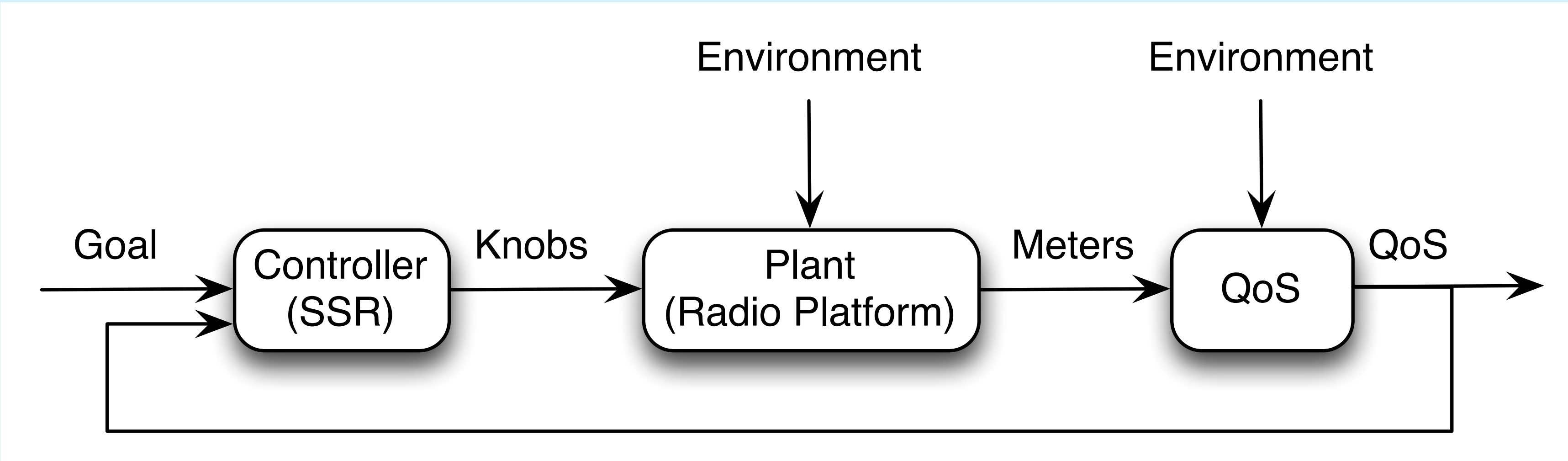
- Sensing and information collection
- Query by user or other radios
- Situational and self-awareness
- Autonomous decision making
- Query execution
- Command execution

Cognitive Radio - Agent

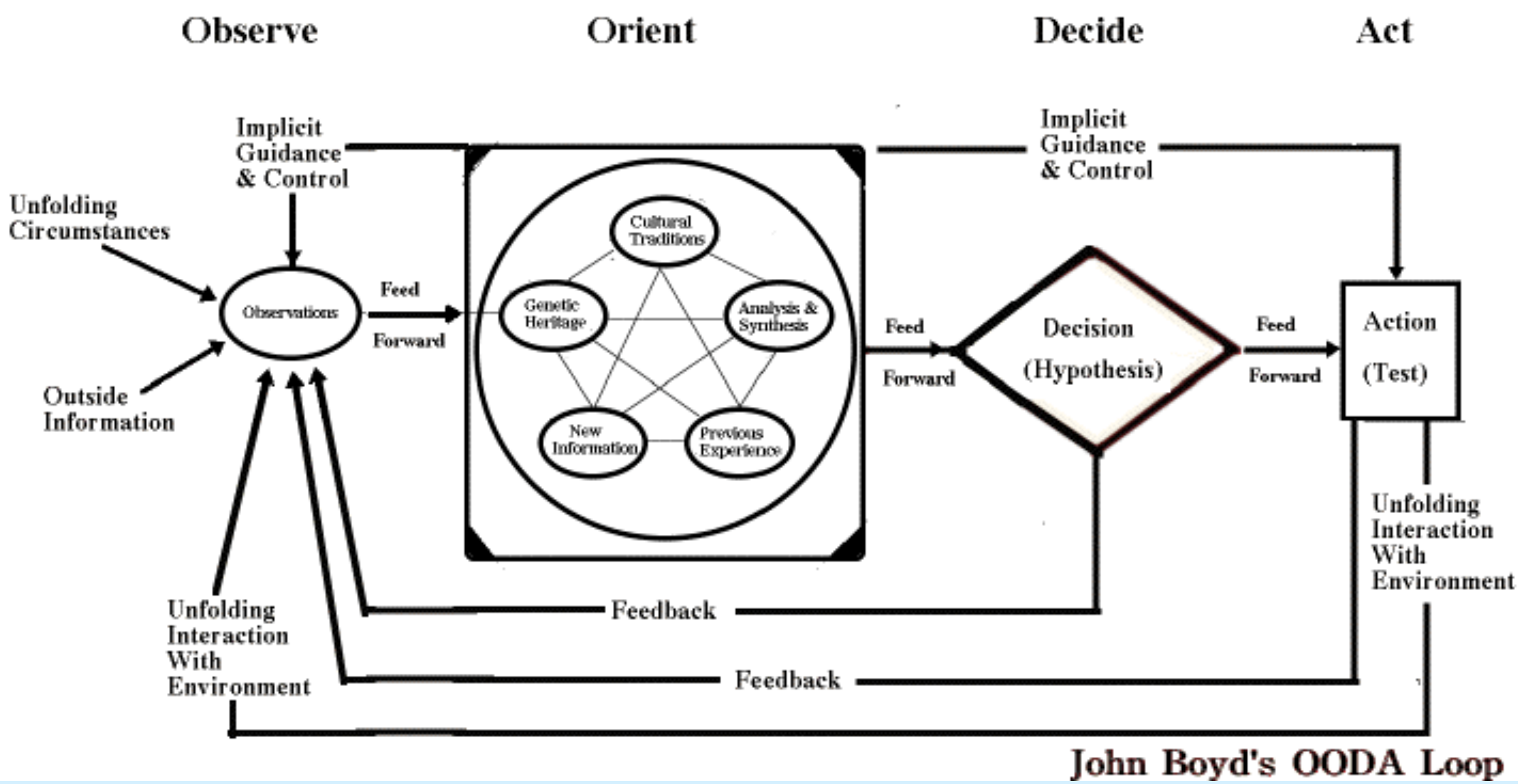


In the context of CR – Goals, percepts and actions are called “knobs” and “meters”

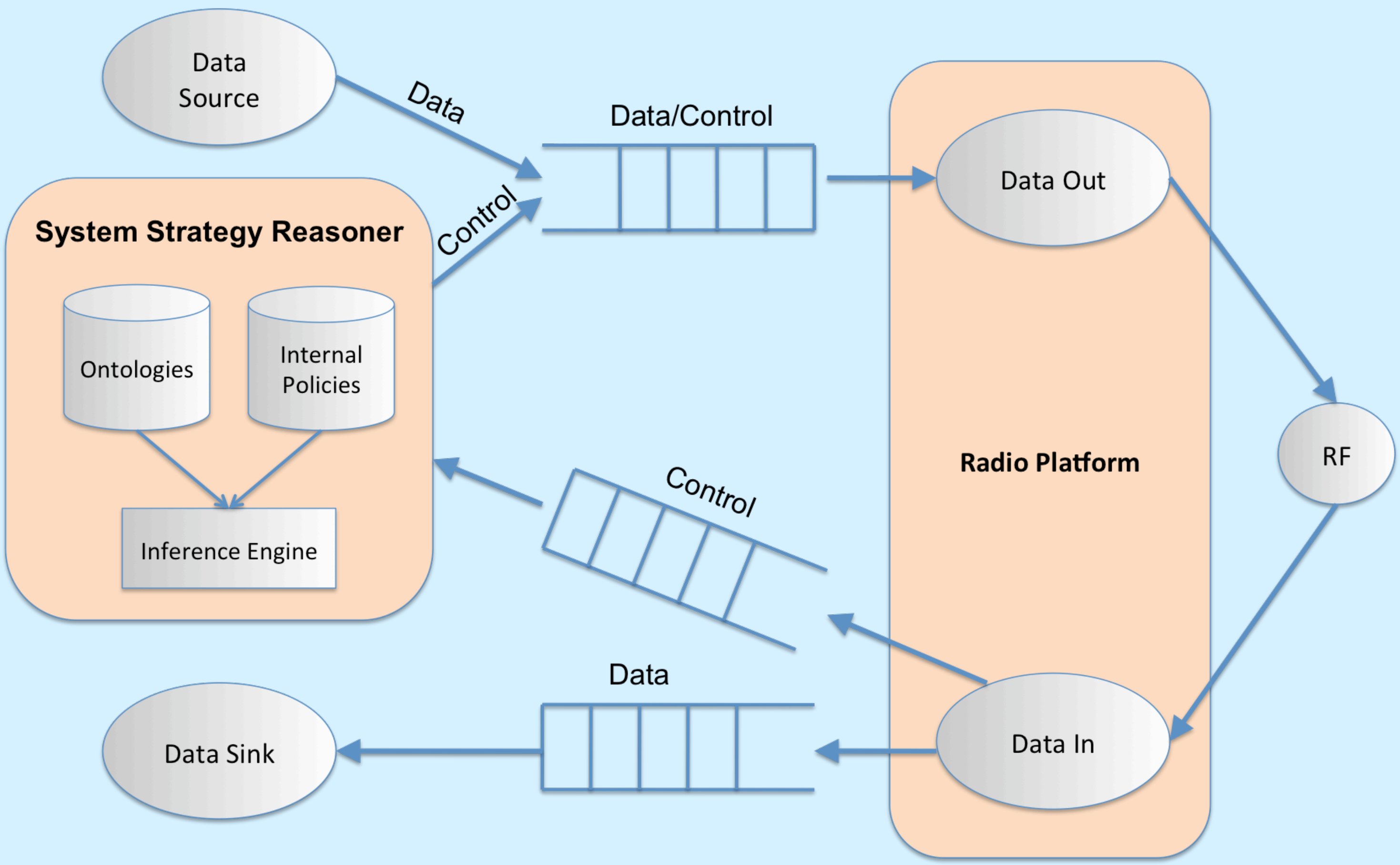
Adaptation


Environment
Environment
Goal
Controller (SSR)
Knobs
Plant (Radio Platform)
Meters
QoS
QoS

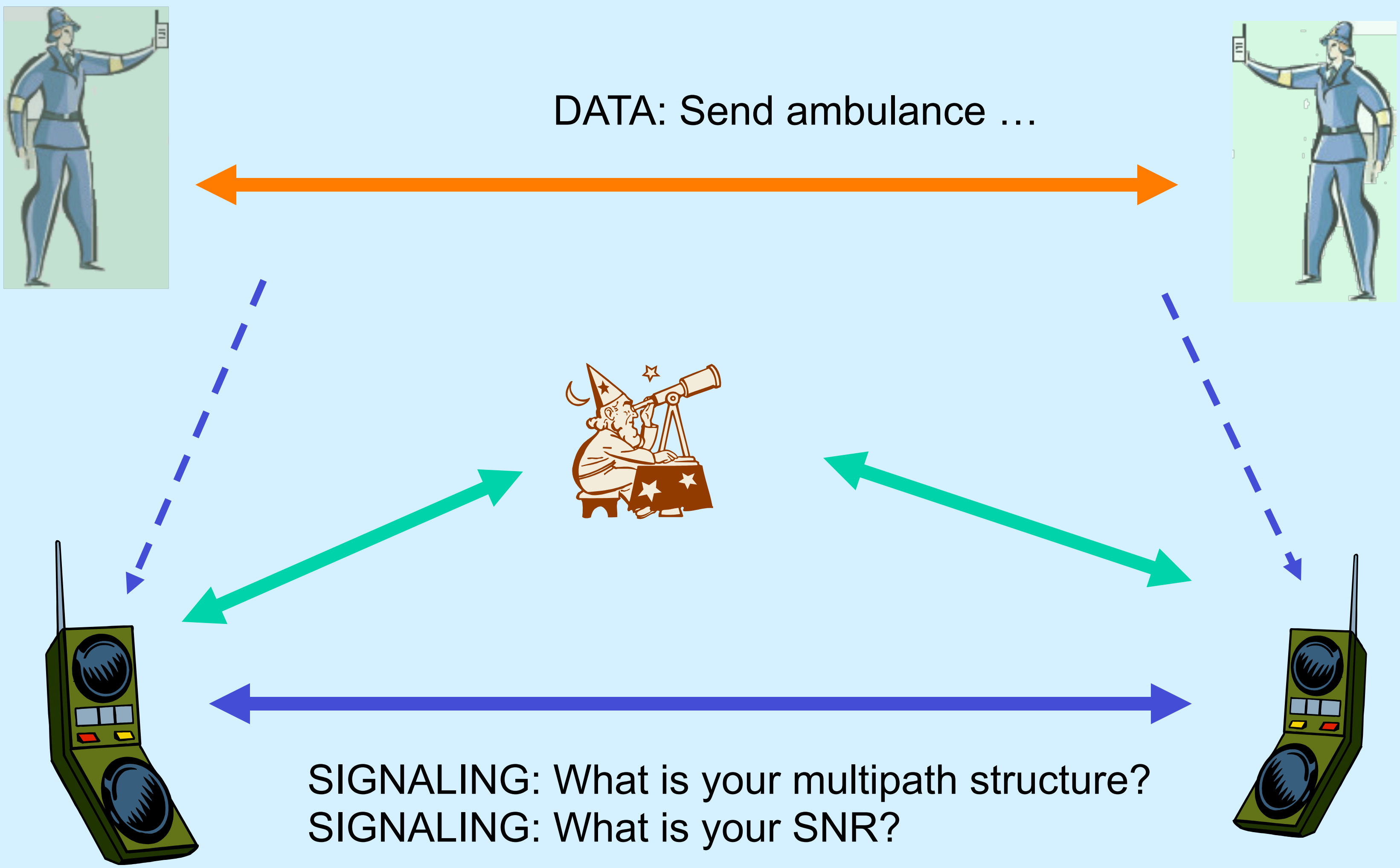
Cognitive Radio Requires "Cognitive Architecture"



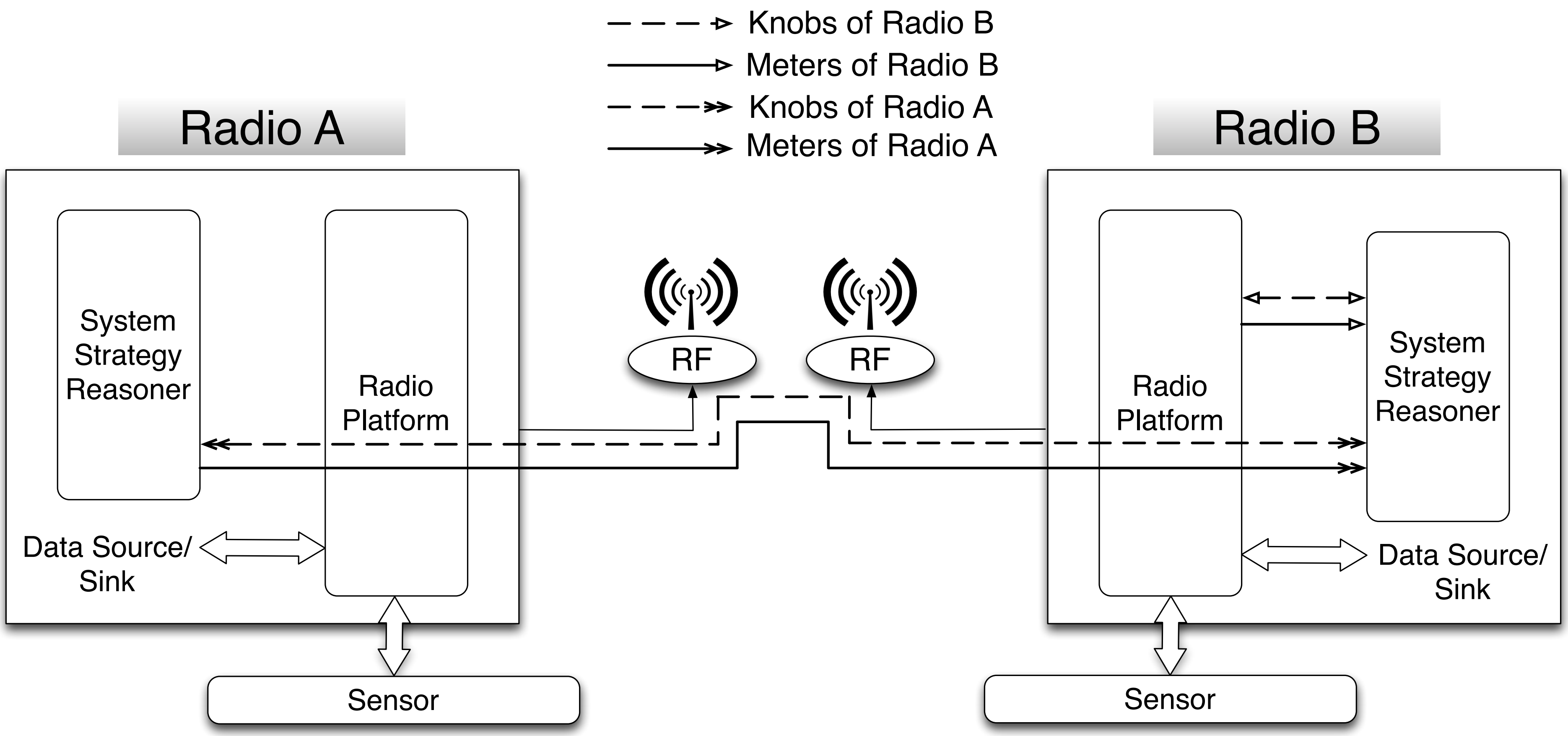
OBR: Ontology Based Radio (2003)



Two Conversations



It Takes Two to Tango



Need flexible communication protocol (exchange of control messages, signaling!

Limits of Current Software Radios

- **Reason 1**: Local information is stored in a data model that does NOT have high expressivity and machine processable semantics.

?

What control information can be exchanged

Scalar variables	Structure of a component	FSM of a component	More complicated information …
✓	✗	✗	✗

XML might help?

But XML does NOT have machine processable semantics

Limits of Current Software Radios

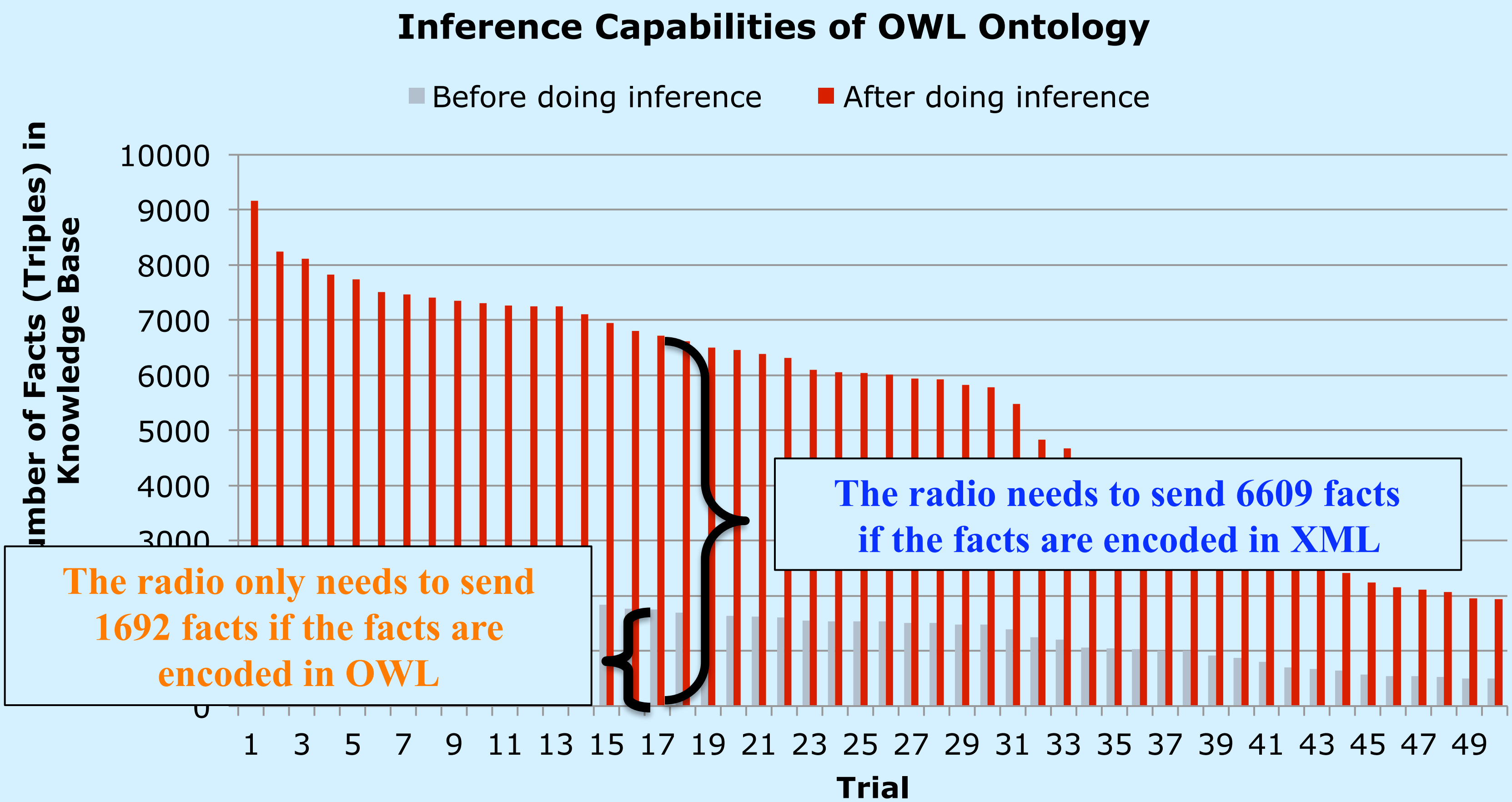
- **Reason 2**: Signaling messages are limited in the frame structure defined by the protocol

Frame format of 802.11

Frame Control	Duration ID	Address1 (source)	Address2 (destination)	Address3 (rx node)	Sequence Control	Address4 (tx node)	Data	FCS
2	2	6	6	6	2	6	0 - 2,312	4

It is **NOT** possible to exchange some signaling messages that are **NOT** defined in the protocol

Comparison: XML vs. OWL



- **XML**: the radio must send all the information explicitly
- **OWL**: the radio only needs to send parts of the message, while the rest of the information can be inferred locally by the inference engine based on the generic knowledge encoded in the shared ontology
  - ➔ Less communication overhead imposed to the network

Example Inferences

- Infer whether particular frequency bands are contiguous, overlapping, covering
- Whether requested frequencies, bands, times of transmission, locations, power levels satisfy particular policies
- From description of responsibilities a user's node has, infer the user's *role* and then derive decision on whether allow transmission (e.g., fireman in an emergency situation)
- Whether security requirements of a specific request are satisfied
- Whether the content of a message can be sent (provided meta-information about the content is provided)
- Execution of rules (condition-action) to determine which knob should be adjusted and by how much
- Radio A sends a description of a component to Radio B; infer whether a specific component it has satisfies the description
- Whether a specific transmission is within the constraints of a given model (refer to MITRE Model Based Spectrum Management)

MLM: Modeling Language for Mobility

- *Development of use cases for wireless communication in which the MLM language can facilitate flexible communication,*
- *Development of Cognitive Radio Ontology (CRO) that is capable of expressing structural, functional and behavioral aspects of models for wireless communication,*
- *Corresponding signaling plan, requirements and technical analysis of the information exchanges that enable these next generation features,*
- *Policies and rules for policy based radio control,*
- *Ontology extensions needed to support policy based radio control.*

MLM Work Group of the Wireless Innovation Forum

MLM Work Group: Goal

A formal language, with a computer-processable semantics, that could be used for describing various aspects of network devices, capabilities, operations and policies for component and network management.

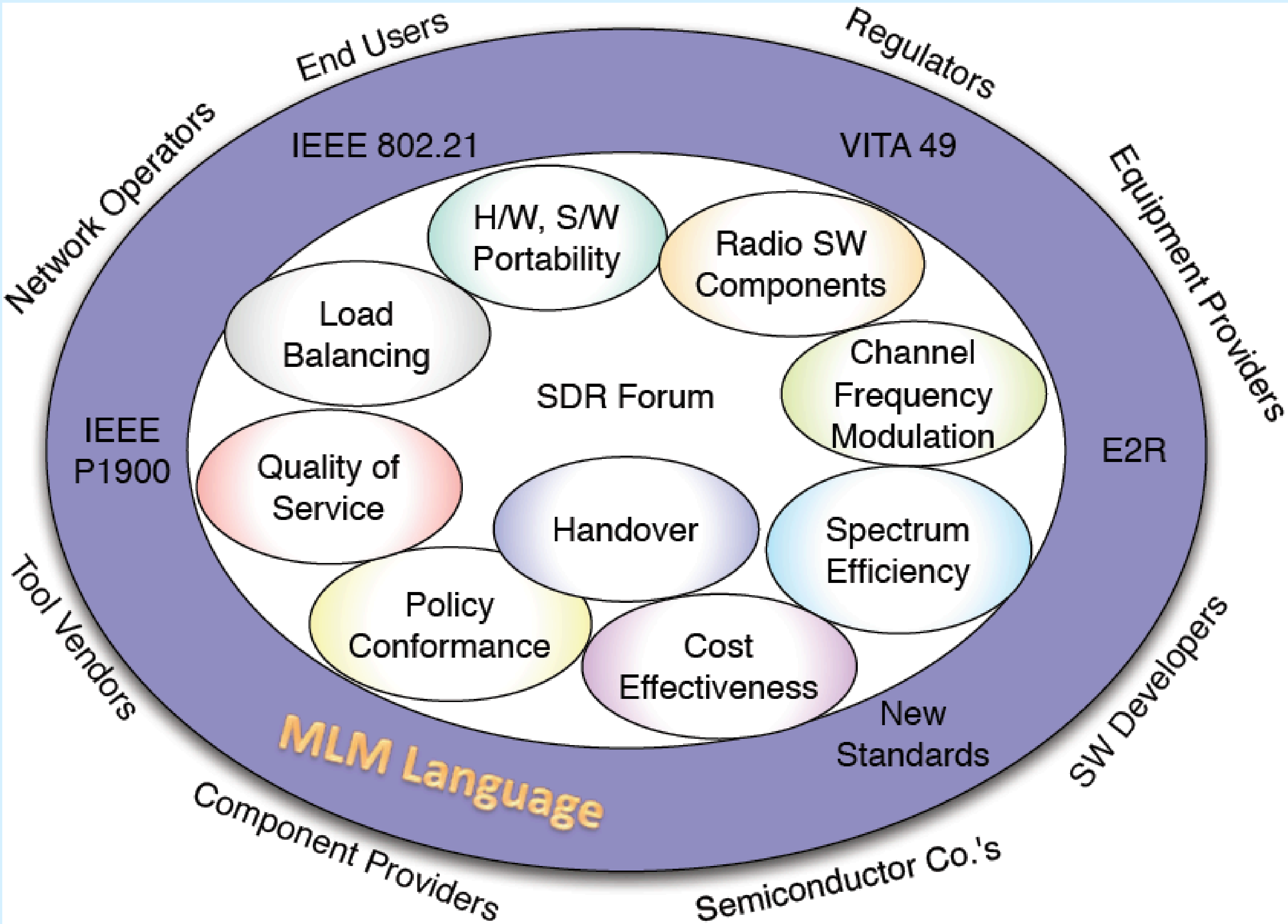
Descriptions and policies to be interpreted by general purpose Inference Engines (Reasoners) rather than being hard coded into devices at the design time.

Close collaboration (MoU) with IEEE P1900.5:

Lynn Grande (P1900.5 Chair): *lynngrande@ieee.org*

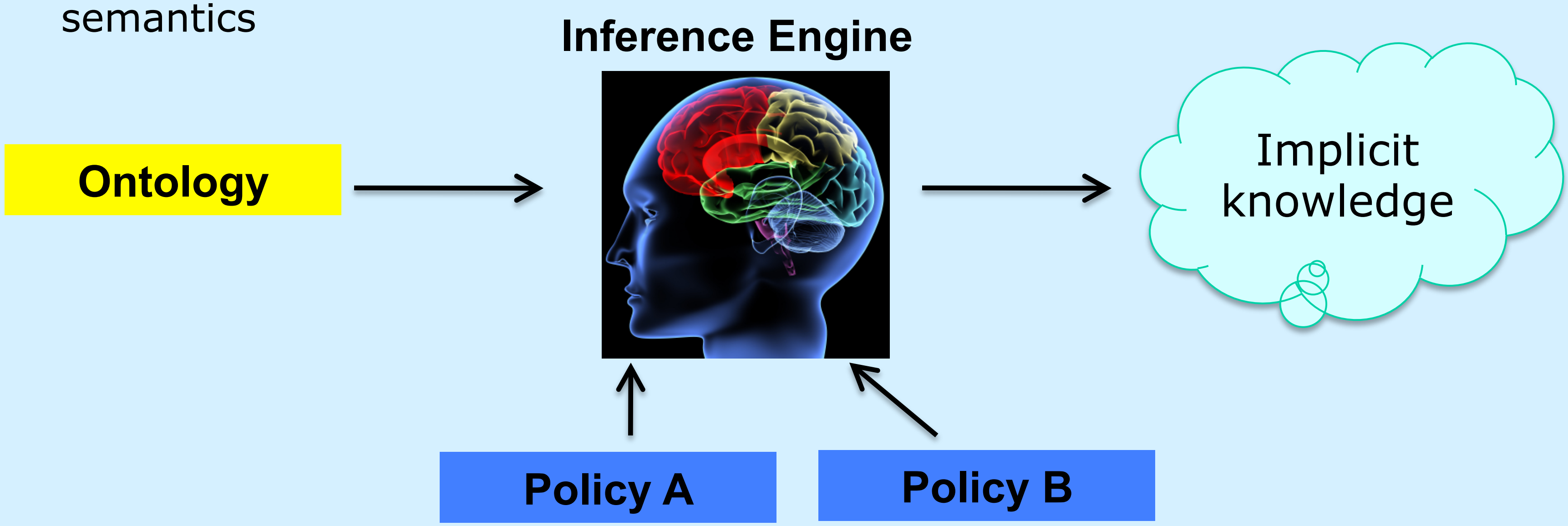
Policy Language and Policy Architectures for Managing Cognitive Radio for Dynamic Spectrum Access Applications

MLM Addresses Interoperability



Summary: Policy-based Radio Control

- Policy-based radio control
  - The behavior of the radio is controlled by (local) policies
  - Policies are expressed in declarative form with unambiguous semantics, e.g., OWL and rules
  - Standards Based Inference Engine: e.g., BaseVISor
- Policies are separated from implementation
  - Modification of radio behavior becomes flexible
  - Simpler certification process
  - Represent policies at a more abstract level and with easier understood semantics



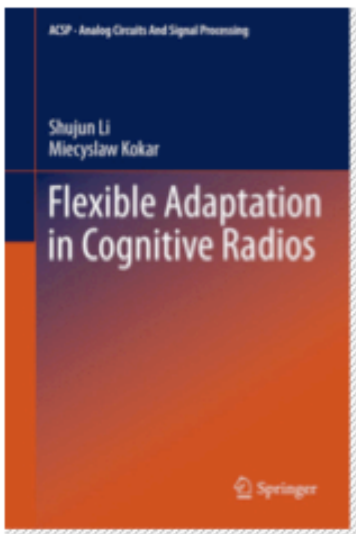
Springer

HOME | MY SPRINGER | SUBJECTS | SERVICES | IMPRINTS & PUBLISHERS | ABOUT US

» *Circuits & Systems*

Home > Engineering > Circuits & Systems

SUBDISCIPLINES | JOURNALS | BOOKS | SERIES | TEXTBOOKS | REFERENCE WORKS



Flexible Adaptation in Cognitive Radios

Series: » Analog Circuits and Signal Processing

Li, Shujun, **Kokar**, Miecyslaw

2013, 2013, XIX, 155 p. 84 illus.

Free Preview

Available Formats:

eBook (net) price ISBN 978-1-4614-0968-7 Immediately available per PDF-download (no DRM, watermarked)	**$99.99**
Hardcover	**$129.00**

Thank You!

Questions?