Towards a Unified Policy Language for Future Communication Networks: A Process

Mieczyslaw M. Kokar[1], Donald Hillman[2], Shujun Li[1], Bruce Fette[3], Preston Marshall[4], Mark Cummings[5], Todd Martin[6], John Strassner[7]
[1] Northeastern University, [2] Lehigh University, [3] General Dynamics C4 Systems, [4] DARPA, [5] en Via, [6] Science and Technology Associates, [7] Motorola

Abstract- **This paper describes an effort to develop an integrated set of ontologies for the particular sub-domains of network communication and a formal language, common to all the sub-domains, capable of expressing domain specific policies. Principal among these is the dynamic spectrum access (DSA) domain. This effort is led by the Software Defined Radio Forum MLM Work Group, with participation from DARPA and the IEEE Standards Coordinating Committee 41 (SCC 41). The focus of this paper is a description of the process by which this effort is carried out. The main goals of the paper are to provide the motivation for the effort with respect to DSA, describe the process being used, and promote participation of the wider networking community in the described activities.**

I. INTRODUCTION

The growing complexity and heterogeneity of communication networks requires new approaches to network management and control that can account for the dynamic changes in communication environment, user demands, topology and composition. One of the approaches that is being developed by various organizations is to replace the static design time / standard solutions with more flexible and adaptable policy based solutions. In this approach, the management and control mechanisms (and possibly their specification as well) need to have awareness of network status and demands. In advanced versions of this approach, such solutions would be aware of changing context and adjust the services and resources offered by the network accordingly. This is possible only if the particular network components are capable of exchanging information about their users, components, capabilities, needs, and other managed objects. Since the information in such exchanges needs to be processed by the network components without much human intervention, the components must have a language that all of them "understand". Towards such a goal, the Software Defined Radio Forum (SDRF) is leading an effort to develop a formal language, with computer processable semantics, that could be used for describing all aspects of network operations and management. In particular, network components, e.g., wireless handsets, could use the language to describe their capabilities, which then could be used by the network to configure its own processing that is a best fit to this particular handset. Similarly, wireless systems around network edges could negotiate with the infrastructure to achieve optimal solutions. This is becoming more critical with the adoption of Femtocellular architectures.

Figure 1 shows a conceptual view of where standardized languages may play a role in the communication among various actors [1]. The actors are shown at the outside of the figure. These are the individuals and organizations that are interested in communicating with regard to many issues. Examples of such issues are shown in the ovals, e.g., HW/SW portability, channel frequency modulation, etc. The intermediate layer shows some languages that the actors could possibly use. The SDRF is working on a formal language with computer processable semantics that could be used as a common language among the various automated processes used by the actors to address their communications and networking needs.

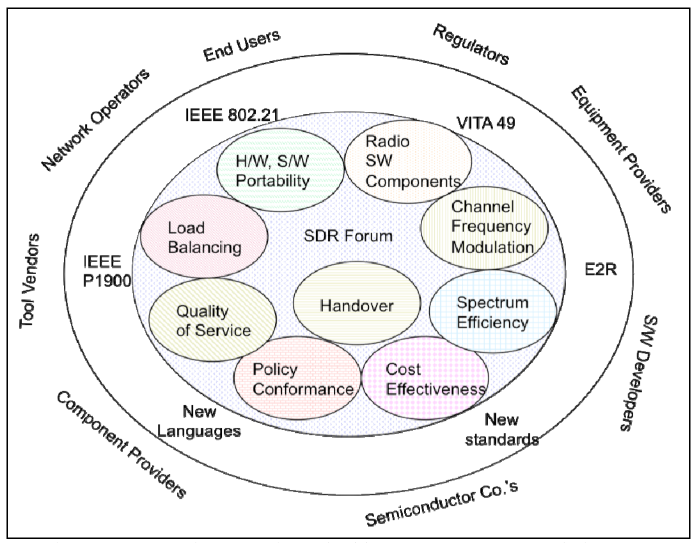

Figure 1. Actors, issues and standard languages: a conceptual view [1].

Dynamic Spectrum Access (DSA) is one of the issues of interest to Regulators as well as End Users, who in this case might be represented by the software agents running on various mobile devices. A number of emerging approaches to DSA systems employ rule-based mechanisms to adapt radio behaviors to application needs, host system capabilities, in situ spectrum environment, and regulatory constraints. The Defense Advanced Research Projects Agency (DARPA) neXt Generation (XG) Communications Program proposed the use of non-procedural computer languages and associated reasoners as a means for expressing and enforcing sets of policies to enable and govern radio behaviors. Thus radios could roam the world while autonomously enforcing spectrum access rules according to the policies provided by the local spectrum governing authorities. Furthermore, that approach separates radio technologies, regulatory policies, and optimizing techniques governing spectrum access such that each of them could evolve asynchronously. As a parallel effort, the End-to-End Reconfigurability (E2R) project is working on a markup language for describing the functionality of various components. More recently, the IEEE SCC 41 has begun efforts to develop a set of interoperable and vendor-independent languages and architectures for policy-based DSA systems.

The standardization effort initiated at the Software Defined Radio Forum (SDRF) is currently tasked with the following two deliverables:

1. An integrated set of ontologies for the particular sub-domains.
2. A formal language, common to all the sub-domains, capable of expressing domain specific policies.

In this effort, two fundamental dangers must be avoided:

- Development of a Perfect Language that nobody wants to use (similar to Ada and Esperanto)
- Development of a large number of non-compatible languages that each solve one piece of the problem but do not make an efficient composite solution.

To avoid these dangers, the semantic mechanisms and the language must be developed and approved by the largest possible cross section of the communities of interest while maintaining a strict focus on domain needs. This goal will be achieved by an active outreach program. Outreach activities will include contributions to broadly-read publications and cooperation with key industry groups seeking input to and review of our proposed standards using a publicly available web site. The outreach will try to involve the broadest and largest possible number of organizations representing affected user and developer communities to vote on the proposed solutions.

In this paper, we will describe a process that has been developed based on a modification of a widely accepted and effective software development process.

II. Process Requirements

It is recognized that what is being developed here is a language, not a software module, and therefore care must be taken in modifying the process to meet the objectives of delivering:

- A widely accepted language for defining and/or negotiating communication system and network configuration.
- A supporting set of semantic mechanisms enabled through supporting knowledge engineering tools such as ontologies, for enabling machine-based understanding of configuration commands and options.

Specifically, we expect the language and associated semantic reasoning mechanisms to address the following areas as a minimum:

1 Capabilities of the nodes (e.g., frequency bands, modulations, MAC protocols, access authorizations, etiquettes, bandwidths, and interconnections)
2 Networks available to a user (parameters, restrictions, costs)
3 Security / privacy (capability, constraints, policies)
4 Information types (an emergency call vs. just a "how are you" message)
5 Local spectrum situation (spectrum activity, propagation properties)
6 Network to subscriber & subscriber to network control (policies)
7 Manufacturer matters (hardware and software policy)
8 Types of users (authority, priority, etc.)
9 Types of data (Async., Isoc., narrow band, broad band, etc.)
10 Local regulatory framework (e.g., policies at a given geo location, time of day, emergency situation, etc.)
11 Time of Day (at both ends of session and important points in between)
12 Geographic Location (in three space, surrounding geography/architecture).

The Process described in this paper is based on the following top-level requirements and constraints:

1. The process should identify self-controlling feedback mechanisms that would drive the development of the deliverables towards quality, and ultimately to the approval of the proposed solutions by the largest cross section of the communities of interest.

2. In order to provide actionable feedback, the process must include the development of a "complete" set of artifacts that can facilitate the assessment of the quality of the ontologies and the language. This set should include, at a minimum:
   (a) Use cases
   (b) Scenarios for implementing the use cases
   (c) Data for the execution of the scenarios
   (d) Policies
   (e) Tools and knowledge representation necessary to express policies for all scenarios
   (f) Test bed inference engines necessary to demonstrate use case scenarios (actual inference implementation in practice will be left up to the implementer)
   (g) Test results, which will provide tangible data to assess both the policies and the tools used to learn and reason about policy-based decisions.
3. The development should consist of two phases, carried out in a bottom-up fashion. Phase I should focus on the development of ontologies and policies. Phase II should

generalize the results of Phase I and lead to a proposal of a standard policy representation language.

4. The deliverables should be developed in a collaborative fashion. Consequently, the process should include tools for supporting such collaboration.

To begin the development of a process for these requirements, a process framework needs to be selected first. One such framework that addresses all of the issues listed above is the Rational Unified Process (RUP) framework [2]. While the RUP approach is crafted specifically to the software development process, it can be adapted to the process needed for the MLM effort. We will use the RUP framework as a guiding principle and as a way of representing a process as outlined in the following section.

III. The RUP: A Brief Overview

The main aspects and “spirit” of the RUP framework are that it is (1) iterative, (2) architecture-centric and (3) use case driven. Moreover, it is (4) a well-defined and structured process and (5) a customizable framework. The RUP allows for a sufficiently abstract framework to be developed in parallel with rapid development of well-defined executable architectures assembled from specific use cases. Thus the policy language and architecture can be developed iteratively with impact and benefit of each new change evaluated across the range of use cases or implementations in a build-test-build approach. Thus RUP balances the desire for policy language and architectural abstraction to support a wide range of implementations while enabling iterative and practical evaluations of the language to support an incremental standardization process.

A RUP process includes four phases: Inception, Elaboration, Construction and Transition. Each of the phases may include a number of iterations. Inception is about an understanding of what to build. Elaboration leads to the understanding how to build the artifact. Construction is the actual implementation of the software system. Transition is the deployment of the software to the user.

The structure of a process includes: Roles, Artifacts and Activities. Views of the process are shown using various UML diagrams. The symbols are shown in Figure 2 below, and the whole process and sub-processes are represented by UML Activity Diagrams as shown in Figure 3.

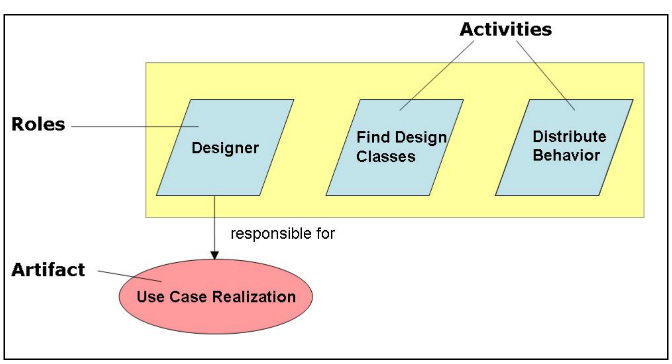

Figure 2. RUP representation components.

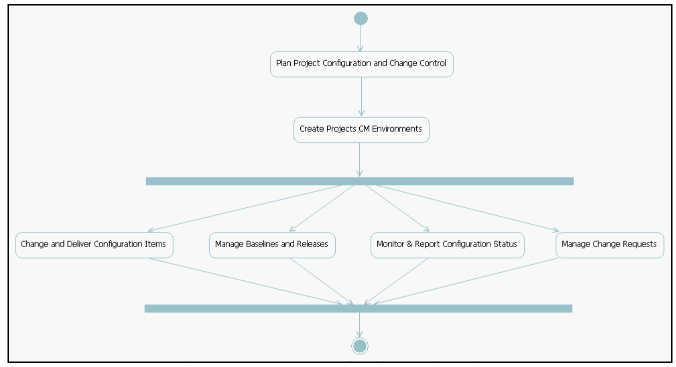

Figure 3. An example of RUP iteration.

Each phase includes some of the basic activities and consists of a number of iterations. An iteration is a distinct sequence of activities with an established plan and evaluation criteria, resulting in an executable release. The number of iterations in any phase is not prescribed by the RUP model and can be defined for each project separately to meet project needs. The phases of a process can be represented graphically as in Figure 4.

Inception	Elaboration		Construction		Transition	
Preliminary Iteration	Architect. Iteration	Architect. Iteration	Devel. Iteration	Devel. Iteration	Transition Iteration	Transition Iteration

Figure 4. RUP phases and examples of iterations.

IV. Applying RUP to the MLM Process

In adopting the RUP model to the MLM process, a number of steps are required. First the language development process is mapped into the four phases of the RUP model as shown in Table 1.

TABLE I
Relationships between RUP and MLM phases

RUP Phase	Phase Objectives	MLM Process Phase
1. Inception	Understand what to build	Use cases, including sequence diagrams.
2. Elaboration	Understand how to build	Prototype - partial implementation of all of the work products: simple ontology, initial language selection, examples of policies, a simulator, scenarios, testbed, test data, inference engine, tests, test results.
3. Construction	Actual development	Full implementation of the work products.
4. Transition	Deployment to user	Evaluation and testing in the user testbed/certification environment, leading to a final standard.

With the processes mapped to the four phases, we then develop workflows for each of the phases. We begin by showing a table (see Table 2) that includes all the activities, indicating which activities are included in the particular phases. Activity Diagrams built from the mapping are then developed as shown in Figure 4 through Figure 8.

TABLE II
MAJOR ACTIVITIES WITHIN THE PHASE OF THE MLM PROCESS

MLM Activity		RUP Phase			
		1	2	3	4
1	Use Cases	X		X	
2	MLM Architecture			X	
3	Scenarios	X		X	X
4	Data/simulation/testbed		X	X	X
5	Ontologies	X		X	X
6	Policies		X	X	X
7	Inference engines		X	X	
8	Test results/data		X	X	X
9	Language specification	X	X	X	
10	Outreach	X	X	X	X
11	Feedback	X	X	X	X

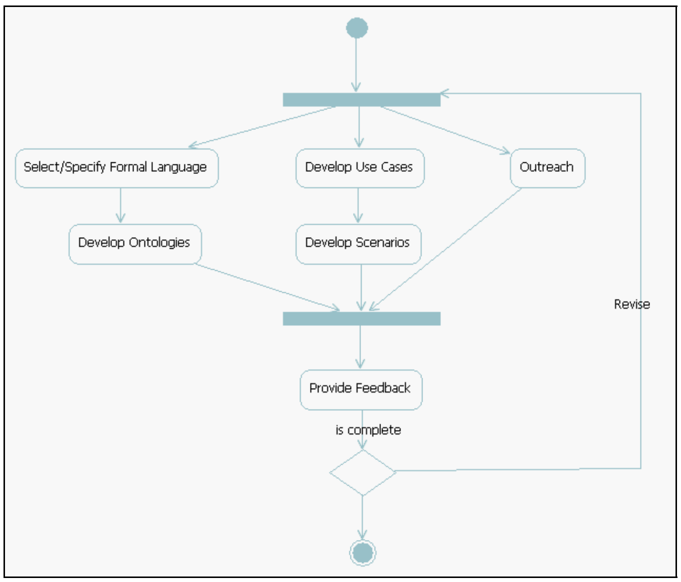

Figure 5. Inception workflow.

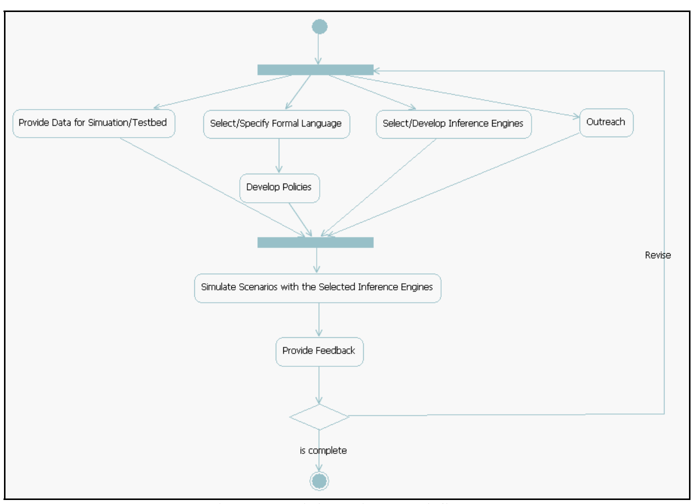

Figure 6. Elaboration workflow.

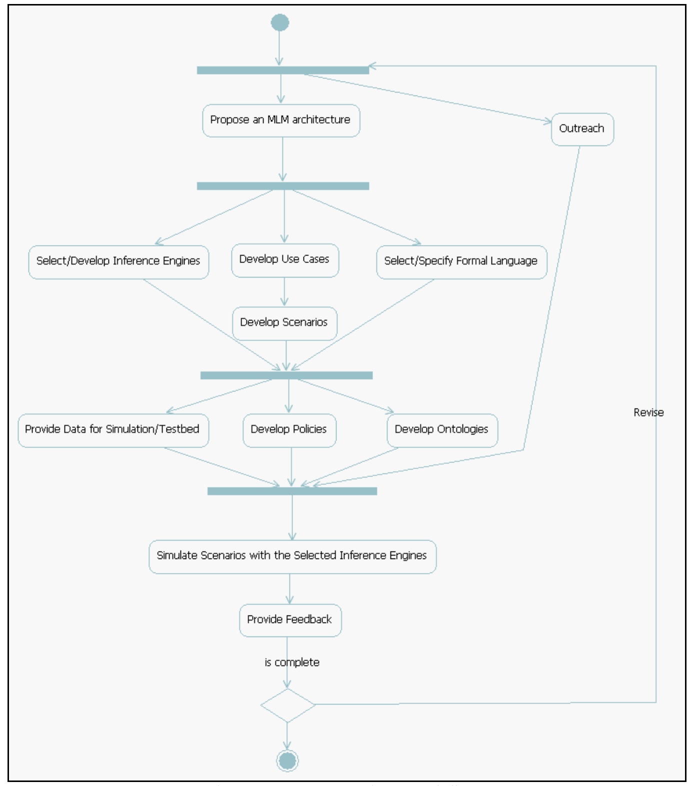

Figure 7. Construction workflow.

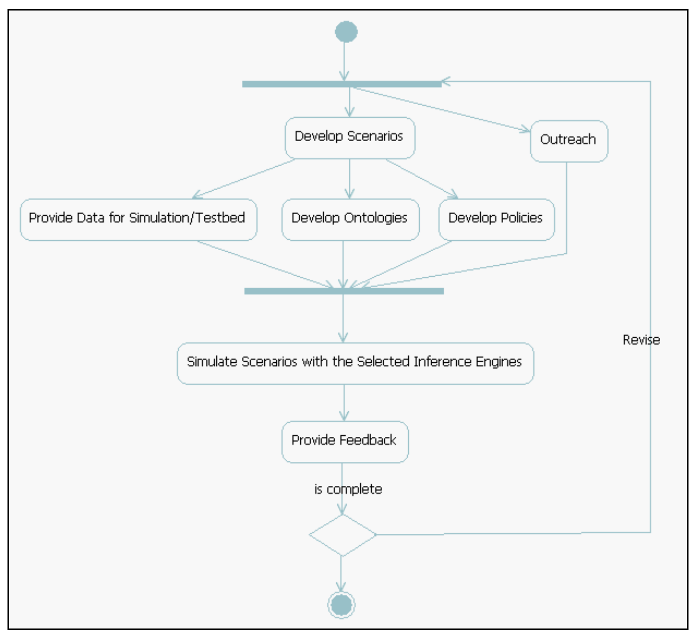

Figure 8. Transition workflow.

The deliverables of each activity are derived from the mapping and workflows. Below we describe each of the 11 activities and include the artifacts produced by the MLM process, which are in roughly one-to-one relationship with the activities.

A. Use Cases

Use cases are used in software and systems engineering to capture the functional requirements of a system [3,4,5]. They describe the interaction between a primary actor who initiates the interaction and the system itself, represented as a sequence of simple steps. Actors are something or someone that exist outside the system being modeled. They may be end users, other systems, or hardware devices. Each use case is a complete series of events, described from the point of view of the actor. Each use case describes how the actor interacts with the system to achieve a specific goal.

This section summarizes the use cases on actual or credible events relevant to public safety and commercial applications that have been considered so far within the SDRF. The actors in these use cases are both the users of cognitive radios as well as the radios themselves. A Dynamic Spectrum Access use case is discussed in more detail. The analysis of the DSA scenario shows how cognitive radio capabilities could positively impact the communications of the public safety and commercial activities.

London Bombing Use Case

The London bombing use case is based on the terrorist event that happened on July 7, 2005 in London. This real-world scenario identifies the circumstances in which future cognitive radio capabilities could provide more efficient and effective communications in similar situations.

- Network Extension for Coverage and Reachback: Cognitive radio can be automatically reconfigured to create peer-to-peer links that can link those radios to infrastructure when radios are cut off from their infrastructure, particularly during initial response to an incident prior to additional communication resources being deployed.
- Dynamical Spectrum Access: Cognitive radio can identify unused or underutilized spectrum when system is overloaded.
- Dynamical Priority Reconfiguration: Cognitive radio is capable of being temporarily reconfigured with higher priorities based on the circumstances of the emergency responders.
- Interface to Non-First Responders: Cognitive radios could allow non-first responders communications access to first responders in specific situations in which the non-first responders are actively participating in the response, while ensuring that mission critical public safety networks are not impacted.

Urban Fire Use Case

In this scenario there is a fire at a chemical plant in an urban environment. There are three roles: an Incident Commander (Leader), a Fire Service Person and an Emergency Medical Technician (Medic).

- The leader's radio is able to: (1) determine which basestation can provide the most reliable service for its need; (2) register with the basestation as a leader; (3) deduce what service each role needs; (4) determine the best available spectrum to each role; (5) instruct the fireman and the medic on what actions to take.
- The fireman and medic's radios are able to: (1) download different AIS in order to connect with the leader; (2) switch between different spectrum in order to download the required software; (3) set up VOIP and streaming video session with the leader.

Load Balancing Use Case

In this scenario load balancing is required at congested locations where a major sports figure breaks a significant record at a sports stadium. There are voice only phones, phones with still cameras, and phones with streaming video cameras and 100,000 of them all want to communicate at the same time. Load balancing might also involve moving into unoccupied spectrum assigned to another primary user by the governing regulatory body.

Software Upload Use Case

In this scenario, a user requests a service from the handset. If the requested service is within the handset's local capabilities, the service is initiated either in its current configuration or in a reconfigured form. Otherwise, the MLM Reasoner (MLMR) will ask the infrastructure or other handsets within the communications range if they can provide software for the requested service. If yes, the software is uploaded and the requested service is established. The actors involved in this use case are: the user's cognitive (ontology based) radio (OBR) and other cognitive (ontology based) radios (Other OBR Systems).

Software Certification Use Case

In this scenario, a software vendor (SW Vendor) submits a request to use new software to an Approved Certification Lab. The Certification Lab invokes its certification process that includes two subprocesses – Simulate and Approve. The testing results of the simulation for various devices will be evaluated to determine which devices can be approved.

Dynamic Spectrum Access Use Case

Below we describe the sequence of events as they might occur in the use case of Dynamic Spectrum Access. It is assumed that the radios in this scenario and the basestations have the necessary cognitive radio capabilities. A UML diagram that depicts this use case is shown in Figure 9. This figure shows that the actors that are involved in this use are Handset and Base (basestation).

The scenario proceeds in the following manner:

1. Base periodically scans the RF environment and updates the spectrum utilization information.
2. When the capacity limit is lower than a pre-defined level, it is determined that the system is overloaded. Dynamic spectrum access is triggered.

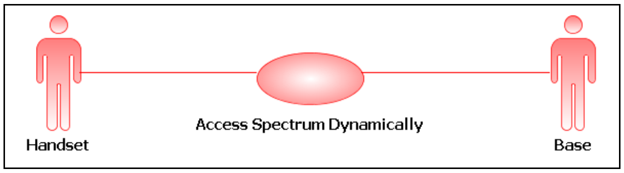

Figure 9. Use Case diagram for Dynamic Spectrum Access.

3. A user's Handset sends a request to Base for a voice call initialization.
4. Base identifies the unused or underutilized spectrums and chooses one of the available channels as the forward and reverse voice channel pair.
5. Base sends a command message to Handset and configures the channel as the handset's reverse voice channel.
6. Base configures the channel as its reverse voice channel.
7. Handset sends a command message to Base, asking if it is ready to receive data.
8. Base sends back a 'confirm' message to handset.
9. Handset starts to send a data message to Base.
10. When the data message arrives at Base, it sends back a 'confirm' message (acknowledgement) to Handset.
11. If Handset receives the 'confirm' message within a predefined period of time, then data transmission is finished. Otherwise, the data is considered to be lost, and then steps (3) to (10) are repeated.

B. Scenarios

One or more scenarios may be generated from each use case, corresponding to the detail of each possible way of achieving that goal. In Figure 10 we show one such scenario for the Dynamic Spectrum Access use case shown in Figure 9. Note that this sequence involves a concrete handset and a concrete basestation, both identified in the diagram. The diagram is a UML diagram, called the sequence diagram. It represents the lifelines of the two objects (basestation B1 and handset H1). The vertical lines show the flow of time (downward). The diagram has two kinds of arrows. The horizontal arrows represent messages sent from one object to another. The arrows that start and end at the same lifeline represent activities that take place in the object itself. Since our intent is to use these diagrams to capture rules for each of the objects (in this case for a basestation and a handset), and since the rules are to be constructed on top of an ontology expressible in terms of binary relations, we restrict the syntax of the messages to only one argument so that they can be interpreted as RDF triples. A message consists of a property (as in RDF or OWL) and the object (in parentheses). The subject is the object to which the arrow is pointing.

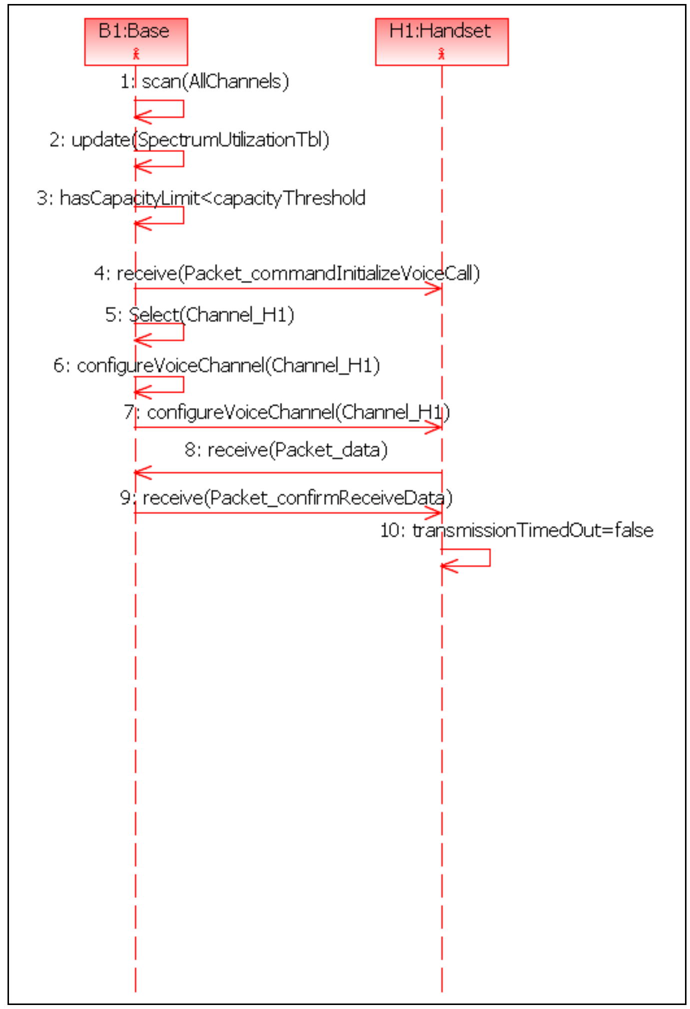

Figure 10. Dynamic Spectrum Access Sequence Diagram.

C. The MLM Architecture

Since the goal of the MLM effort is to develop a language (or set of languages) that will satisfy the needs of all of the actors as shown in Figure 1, the issue of the architecture needs to be addressed. The first decision that needs to be made is whether it is practical to have one language that fits all. At the other extreme would be a continuum of languages, as proposed in [6]. While this idea is being discussed within the MLM Working Group, an architectural solution for Dynamic Spectrum Access with two components – one for endogenous reasoning and another for exogenous reasoning – has been proposed in [7] and is being adopted by the implementers of the DSA policy based control solution (cf. [8]). In this architecture, a System Strategy Reasoner (SSR) is responsible for generating requests for transmissions and proposing parameters of the transmission. A Policy Conformance Reasoner (PCR), on the other hand, is responsible for the verification of the PCR's requests with respect to the policies applicable at a particular time and location. Moreover, the PCR may provide "opportunities", i.e., a set of possible transmission parameters, provided that the radio enforces some additional constraints specified in the opportunities. In that case, the SSR is responsible for selecting an opportunity that is optimal for a given situation. Given that the architecture separates the two functions and a well-defined interface and ontology are developed, the SSR and PCR could each utilize a different language.

The separation of the compliance and enforcement mechanisms from the strategy and optimizing mechanisms are important for policy-based cognitive radios. First and foremost, the distinct functions of the PCR and SSR creates a clear delineation between those mechanisms needing accreditation by regulatory bodies and those that are subject only to the review of the software and radio developers. Second, it disentangles the two policy types and reasoning processes allowing the development community to innovate within the SSR without affecting the accredited PCR. Similarly, it allows regulatory policies to be updated into the PCR without affecting the functionality of the SSR.

As noted above, a well-defined interface is needed between the PCR and SSR. It is reasonable to expect that once the architecture is standardized multiple inference engines for the language would be developed. In order to make this possible, a common API and ontology would need to be published so that tool vendors could implement it and be assured of the compatibility of their tools with the language. This would promote competition among tool vendors and result in an ultimate benefit to the users.

D. Ontologies

The Semantic Web has been responsible for the introduction of several useful tools for improving the flow of information in the World Wide Web (WWW). One of these is the notion of an ontology, which facilitates the sharing and reuse of knowledge among agents. It defines the vocabulary that is used to exchange assertions and queries in a given domain, or area of knowledge.

Ontologies will be developed for the principal domains encompassed by the cognitive radios and communication networks. These are expected to include dynamic spectrum access, security, and routing, among others. Each ontology will be constructed in a series of steps, beginning with a description of the functions performed by an actor in satisfying a knowledge need. Subsequent steps will serve to refine and expand knowledge needs into more detailed declarative descriptions of the knowledge objects and their interrelationships.

The Web Ontology Language (OWL) is the most likely candidate for the expression of an ontology. Each ontology will be a formal specification of the set of concepts and their interrelationships used to articulate a given domain. It is also likely that an editor such as Protégé will be employed to describe these concepts (classes) and their properties. The OWL/Protégé combination is a widely employed tool that can enhance the community's acceptance of the ontologies.

As indicated above, the ontologies that are developed will be computer-understandable specifications of the conceptual vocabularies used to denote the knowledge objects of a given domain together with a description of their interrelationships. Such ontologies are critical for expressing the semantics of the domain and for ensuring that actors are communicating with the correct meanings of terms. The most beneficial result of this would be the development of a canonical form ontology structure to serve as the guide for ontologies in general. That development may not be achievable within the assumed time horizon (see the Timeline discussion), but nevertheless should be initiated.

E. Policies

Agent communication is governed by policies that establish the conditions under which they can operate with each other. A policy will consist of a set of requirements or constraints, together with an enumeration of agent capabilities and preferences. Some policies are permissive in that they allow certain types of agent behavior, while other policies are prohibitive in that they forbid certain kinds of behavior.

A cardinal principle of the policies of interest in this project is that they must not occur in hard-coded form. That is, they must be modifiable without reloading and recompiling software in the radio. The key feature is flexibility that will enable them to operate in loosely coupled service environments.

The policies to be developed should be representative of several communication domains. As noted above, these include dynamic spectrum access, security, and routing. A survey will be conducted to determine the structure of major policy types.

Spectrum access policies have several subtypes:

- Listen-Before-Talk, to guarantee no harm to other users
- Spatial, such as geographical constraints
- Temporal, such as time-of-day restrictions
- Device-Based, such as measurement of 2nd and 3rd harmonics
- Connectivity-Based, such as required beacon reception
- Group-Based, such as detection of non-cooperative users
- Distributed Control, such as update by Policy Authority.

Security policies include:

- Access Control Policies
- Trust Management Policies
- Attribute Acknowledgement Policies
- Routing policies are of importance in that routing is often policy-based, such as restrictions in the scope of routing that exclude paths involving “undesirable” nodes, or insistence that some paths must either pass through or avoid several predetermined nodes.

F. Inference Engines

This activity will involve the identification and evaluation of available inference engines that are commonly used with ontologies to extend their utility. There are many examples of these, including Racer [9], FaCT++ [10], KAON2 [11], Flora-2

[12], BaseVISor [13], OWLIM [14], Pellet [15], and others. Most of these inference engines are limited to OWL inference, while some others are also capable of processing rules. W3C has the Rule Interchange Format (RIF) under development to provide a uniform mechanism for sharing rules [16].

G. Policy Language

A language (or a family of languages) needs to be either identified or developed that is at a high enough level to be able to encompass all the different types of policies described above. The language should be capable of expressing multiple combinations of constraints and capabilities to permit cooperating agents to choose mutually satisfactory subsets of those constraints and capabilities in negotiating policy-governed communications in a dynamic fashion. The requirements of the language are:

- A denotational semantics for expressing ontologies
- A capability for interfacing with different reasoners
- Procedural attachments to invoke functions and procedures written in other (mostly imperative) languages
- An API for interfacing with different tools (e.g., reasoners)
- Good expressivity (to handle complex policies)
- Tractability (for adequate performance)
- Ability to express prohibitive and permissive policies
- Ability to express meta-level policies
- Ability to describe contradictory/overlapping policies in such a way that they can be deconflicted
- Ability to incorporate new policies without code changes
- Ability to handle complex interactions among policies
- Ability for the language to be updated as technology evolves.

It is proposed to follow a bottom-up approach in developing the language, which will permit successful (but limited) language elements to be incorporated and tested in the policy language. In other words, use cases and scenarios from the radio domain will be indentified first and then solutions will be sought among the available resources and new development will be proposed / undertaken only if the existing languages and inference engines will not satisfy the needs. A top down approach may also be used to check for completeness.

H. Test Data

In order to assess the correctness of the implementation of the logic on a cognitive radio, a number of test cases need to be developed. In Figure 9 we showed one such example. While such scenarios play a very important role in the process of developing the logic for the behaviors of cognitive radios, a single scenario is never sufficient to obtain a satisfactory assurance that the functionality of the radio will satisfy some external requirements. Towards this aim, it is necessary either to gather real data, and then test the system with the real data, or if this is not available, simulated data may be a good substitute, at least in the early stage of development. The MML Work Group is considering various alternatives in this respect at this time.

I. Test Results

There will be two major types of results to be evaluated:

- Performance of the interface with different reasoners
- Correctness of results

The first of these is a test of the language's capability to work with any suitable reasoner. Reasoners vary in their syntax and interface requirements. They also vary in the type of reasoning performed, whether data-driven (forward chaining) or theorem-proving (backward chaining). Both types of reasoning have a priori plausibility in the proposed project.

The second type of testing is an evaluation of the correctness of the knowledge representation used by the reasoner. One frequently used way of checking correctness is to establish the required result ahead of time and to verify that the reasoner has generated this solution. This technique can also be used as the basis of a learning procedure for future decision-making.

J. Feedback

The process shown in Figure 5 through Figure 8 shows Provide Feedback activity in each of the phases, from Inception through Transition. SDRF will seek feedback from all interested parties, the actors identified in the description of this process. Towards this aim, working documents will be posted at the web site for the MLM Workgroup and feedback will be collected and incorporated into consecutive versions of the work products.

K. Outreach

Outreach is important in three phases of this project:

1. Getting the best input from the broadest cross section of the whole industry value chain and market segments.
2. Harmonizing the related standards efforts underway in a variety of standards organizations.
3. Having the resulting standard incorporated into and supported by key industry and governmental bodies.

Number 1 requires publication in a broad range of technical and business media. This publication effort should be aimed at helping industry segments understand the importance of the effort and the significance of their input. Media include industry conferences, academic and technical journals, industry press, the general business press, etc. In the past, RFI's have been used and this technique may prove useful going forward. The great danger is that this effort is seen as serving only one industry segment or one slice of the value chain. If this happens, the effectiveness of the result will be greatly compromised.

Number 2 requires a continuing effort to discover, liaise with and coordinate activities of related standards groups, industry consortia and government-funded initiatives. Many of the groups currently identified are represented by co-authors of this paper. Other groups include IEEE802.21, Vita 49, E2R, Daidilos, Ubiquitous Computing, Autonomous Computing, Autonomic Computing, and Next Generation Internet. It should be noted that the MLM WG is coordinating with the Cognitive Radio WG within the SDR Forum.

As the results of this project begin to reach the transition and implementation stages, attention should turn to number 3. Here efforts should focus on government and industry groups. In the government area, outreach should encompass the ITU and National regulatory organizations. The intention here should be to have the projects results incorporated into appropriate government standards and regulations. In the industry area, efforts should focus on such broad based industry standards groups as 3GPP, OMA and others of a similar nature.

If the outreach activity is performed correctly, we will see a comprehensive language developed and widely deployed. This outcome will justify all the effort required to create it.

V. Responsibilities (The Association of Artifacts with Roles)

This section briefly describes the roles played by each of the following actors:

a. Organizations representing the user community (SDRF, IEEE, and other, like representatives of developers, network equipment manufacturers, end user, all of whom could be part of the SDRF role.

b. DARPA and its Performers

c. Regulators (not included so far)

A. SDRF and SDRF Members

The SDR Forum is a non-profit international industry association dedicated to supporting the development and deployment of software defined and cognitive radio technologies that enable flexible and adaptable architectures in advanced wireless systems. Currently numbering over 100 organizations, the Forum's membership comprises world-class technical, business and government leaders who are passionate about creating a revolution in wireless communications based on reconfigurable radio technologies. Membership spans commercial, defense and civil government organizations, including wireless service providers, network operators, component and equipment manufacturers, hardware and software developers, regulatory agencies, and academia from Asia, Europe, and North America. The Forum holds yearly technical conferences, and has been a major venue for advanced technology publications in SDR and CR topics. The Forum also develops industry positions on important new technologies required to enable new businesses. One such technology is the exchange of information required amongst next generation cognitive radios and their support infrastructure for next generation, and SDR Forum is working to develop this topic in order to enable the CR industry. SDR Forum has two work groups, the Cognitive Radio Work Group, which has developed a CR definitions document, a Nomenclature document that discusses Cognitive functionality architecture, and is now developing a response to ITU questions on Cognitive Radio. The SDR Forum also has the MLM Work Group developing flexible and efficient communication protocols between advanced radio systems to support next generation features of vertical and horizontal mobility, spectrum awareness and dynamic spectrum adaptation, waveform optimization, capabilities, feature exchanges, and advanced applications. This includes use cases, corresponding signalling plans, requirements and technical analysis of the information exchanges that enable these next generation features.

B. IEEE

The IEEE SCC41 P1900.5 working group was officially approved on 27 March 2008. The purpose of this working group is to define a policy language (or a set of policy languages or dialects), and their relation to policy architectures, for specifying interoperable, vendor-independent control of Cognitive Radio functionality and behavior for Dynamic Spectrum Access resources and services. The definitions of policy language, architecture, and their relation with each other shall be done with respect to the needs of at least the following constituencies: the regulator, the operator, and the network equipment manufacturer.

C. Other Consortia

The Outreach section contains names of other consortia currently identified. The key here is to continue to identify others. The problems we are working on here have aspects that are broadly felt globally and there are sure to be consortia that currently exist working on some related aspect that we have not yet found. Also, other consortia will come into existence. The key here is to avoid turf battles and seek to encourage all to work on theses problems while harmonizing and coordinating their efforts.

D. DARPA Performers

In addition to the SDRF, IEEE, and other consortia, the process requires input from the stakeholders developing the software for networking and cognitive processes. The DARPA Wireless Networking after Next (WNaN) Program and its performers are currently acting in that role as surrogates until a broader industrial base can be established.

The networking and cognitive software developers fulfill several roles in cooperation with the other actors. They will be involved in nearly all the activities, including development of use cases, scenarios, and ontologies as well as providing data and feedback based on simulations and prototype implementations. They will ensure that the activities and artifacts support adaptive and cognitive processes to be used by the radios. Those processes include DSA algorithms, network adaptation, and environmental learning functions. The networking and cognitive software actors will also work within the process to ensure that special purpose reasoners (e.g. spectrum, networking, security) that may be developed by different organizations are able to exchange information.

VI. Timeline

Both SDR Forum and the IEEE SCC41 have committees that are working on tasks related to policy languages for management of cognitive radios and cognitive radio networks. The SDR Forum established the Modeling Language for Mobility Work Group (MLMWG) in April of 2004. Previously known as the commercial handset working group, this work group is working on languages necessary to enable next generation communication devices, including PDAs, cell phones and cognitive radios, to be able to interact with each other and with their support infrastructure to achieve new capabilities. A simple example of this would be a portable subscriber determining which wireless networks are locally available, what services are available on those local networks, determining cost effectiveness of service for a particular application, registering to the network and using the service. There are much more sophisticated examples; each example is supported with a use case analysis. Additional examples include services valuable to Public Safety applications, defense applications, and more complex commercial applications. The MLMWG is currently developing a series of use case documents to describe these next generation applications, and the signaling required to accomplish them, and from the use cases to establish the requirements for language components to achieve these functions. MLMWG has also established liaison with E2R and with IEEE P1900.5 in order to have broad awareness of other industry activities, to develop industry support, and focus on commercially important topics. MLMWG roadmap is to complete a first use case document for selected MLM applications in public safety and vote it out of the MLMWG up to the SDR Forum Technical Committee by June of this year. Additional documents will continue to be developed adding depth to the requirements and evolving details of signaling and languages over the course of the 2008 calendar year. The IEEE timeline for completing the first draft (i.e., ready for ballot) specification is targeted at March 2009, but should be no later than June 2009.

VII. Collaboration Tools

MLM Work Group will be using the SDR Forum's collaboration tools. The SDR Forum has selected the tools provided by Socious. They are accessible at http://sdr.socious.com. At the time of this writing, through this website, members of the SDR Forum are able to communicate and share information related to their forum related activities. The tool site features: forums and list servers for communicating and sharing ideas, file libraries for posting records and documents that are important to the specific community (sub-community of the forum), event calendars, polls and surveys, news and more. Users can build their own email lists, establish controls for their own groups, create their own group calendars, and store documents in a shared repository. RSS feeds and discussion threads are available now, and wiki pages will be available. All the documents of the MLM Work group will be accessible through this site.

VIII. Conclusions

In this paper we have described a process that is underway in one government organization (DARPA), a professional society (SDRF) and the IEEE. The goal of the process, as described in this paper, is to develop a collection of ontologies that capture a significant portion of the radio and network domain knowledge using a formal language with computer-processable semantics. Such a language will need to be developed as an extension and/or combination of existing languages. The process is represented using the RUP process modeling framework.

References

[1] Fette, B., Kokar, M. M. and Cummings, M. "Next-Generation Design Issues in Communications", Portable Design Magazine, No.3, pages 20 - 24, 2008.

[2] Per Kroll (Author), Philippe Kruchten. The Rational Unified Process Made Easy: A Practitioner's Guide to the RUP. Addison-Wesley, 2003.

[3] "Use case", Wikipedia, the Free Encyclopedia, http://en.wikipedia.org/wiki/Use_cases, 2008.

[4] Ivar Jacobson. Object-Oriented Software Engineering. Addison Wesley Professional, 1992.

[5] Kurt Bittner and Ian Spence. Use Case Modeling. Addison Wesley Professional, 2002.

[6] John Strassner, Mícheál Ó Foghlú, Willie Donnelly and Nazim Agoulmine. "Beyond the Knowledge Plane: An Inference Plane to Support the Next Generation Internet",

[7] Preston F. Marshall. "Extending the Reach of Cognitive Radio", paper under review, 2008.

[8] Filip Perich. "Policy-based Network Management for NeXt Generation Spectrum Access Control", pages 496-506, DySPAN 2007, 2nd IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks, Dublin, 2007.

[9] V. Haarslev, R. Muller and M. Wessel. RACER. Available at: http://www.sts.tu-harburg.de/~r.f.moeller/racer/, 2005.

[10] FaCT++ homepage. http://owl.man.ac.uk/factplusplus/

[11] Motik, B., Studer, R.: KAON2 – A Scalable Reasoning Tool for the Semantic Web. In: Proceedings of the 2nd European Semantic Web Conference (ESWC'05), Heraklion, Greece (2005).

[12] Michael Kifer, Georg Lausen, and James Wu. Logical Foundations of Ob ject-Oriented and Frame-Based Languages. Journal of ACM, May 1995.

[13] C. Matheus, K. Baclawski and M. M. Kokar. BaseVISor: A Triples-Based Inference Engine Outfitted to Process RuleML and R-Entailment Rules. In Proceedings of the 2nd International Conference on Rules and Rule Languages for the Semantic Web, Athens, GA, Nov. 2006.

[14] Kiryakov, Atanas and Ognyanov, Damyan and Manov, Dimitar, "OWLIM – a Pragmatic Semantic Repository for OWL". International Workshop on Scalable Semantic Web Knowledge Base Systems (SSWS 2005), Springer-Verlag LNCS series, LNCS 3807, pages 182-192, 2005.

[15] Pellet, 2005. Available at: http://pellet.owldl.com/.

[16] World Wide Web Consortium, "RIF Basic Logic Dialect," www.w3.org/TR/2007/WD-rif-bld-20071030/, October 2007.