Intelligence in the Network – Future Wireless Networks

Joseph B. Evans

Deane E. Ackers Distinguished Professor, Electrical Engineering & Computer Science
Director, Information & Telecommunication Technology Center

KU INFORMATION & TELECOMMUNICATION TECHNOLOGY CENTER
The University of Kansas

Intelligence in the Network

- Theme – the impact of increased capability (throughput, processing) on networking in general and wireless networking in particular…

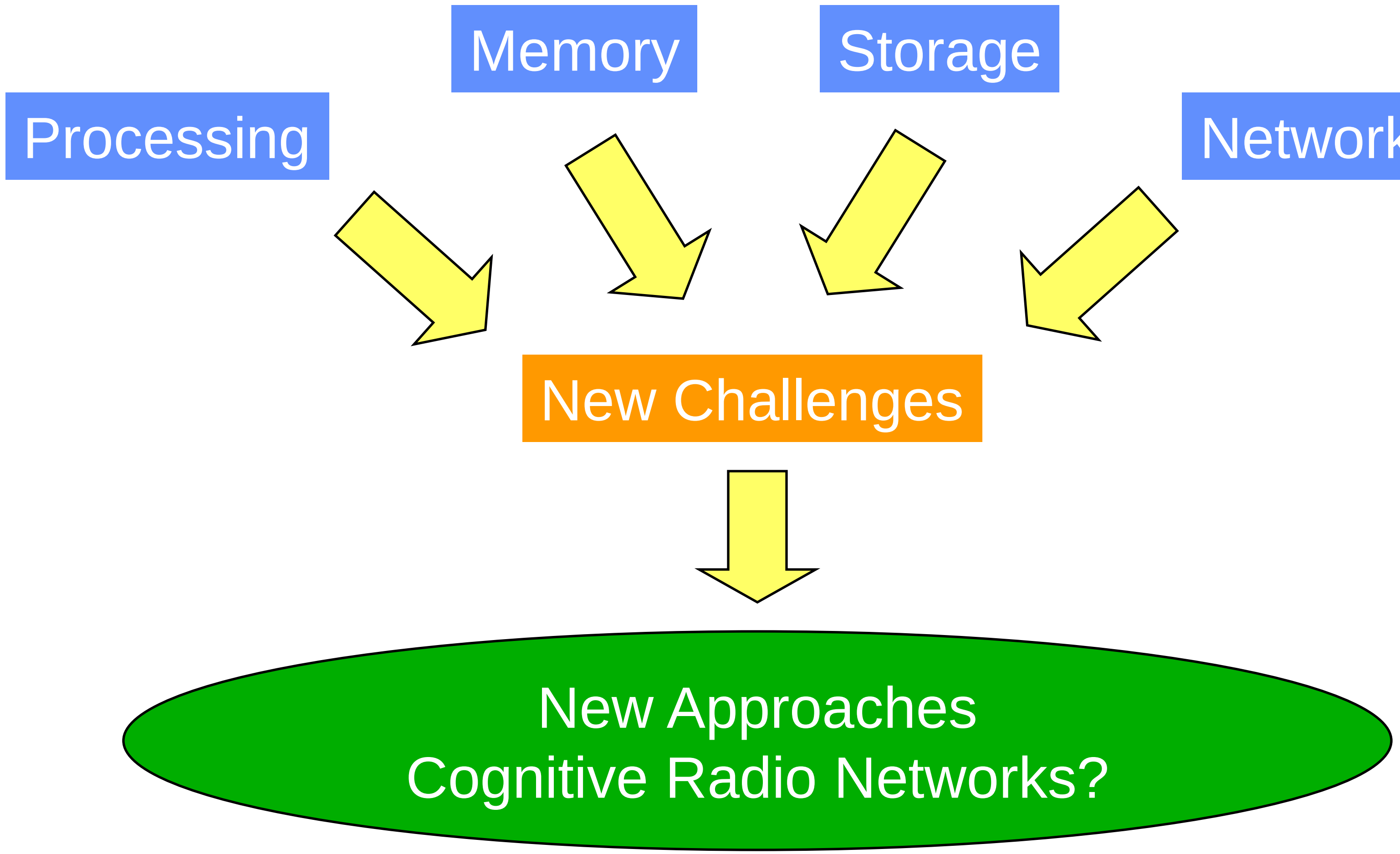
- Smarter nodes – cognitive radios
- Radio systems – radio adaptation, spectrum resource sharing
- Network architectures – topology discovery, layer adaptation, etc.
- Network management – sense/act/learn at scale
- Designing these systems…

Wireless Networking Challenges

Why is wireless networking hard?

- Mobility is inherent with untethered
- Resources are constrained
  - Spectrum “scarcity” → bandwidth & delay issues
- Environment changes
  - Mobility → different surroundings (indoor, urban, rural)
- Varying physical properties
  - Wireless communication path changes over time

Impact of Trends



Cognition in Wireless Networks

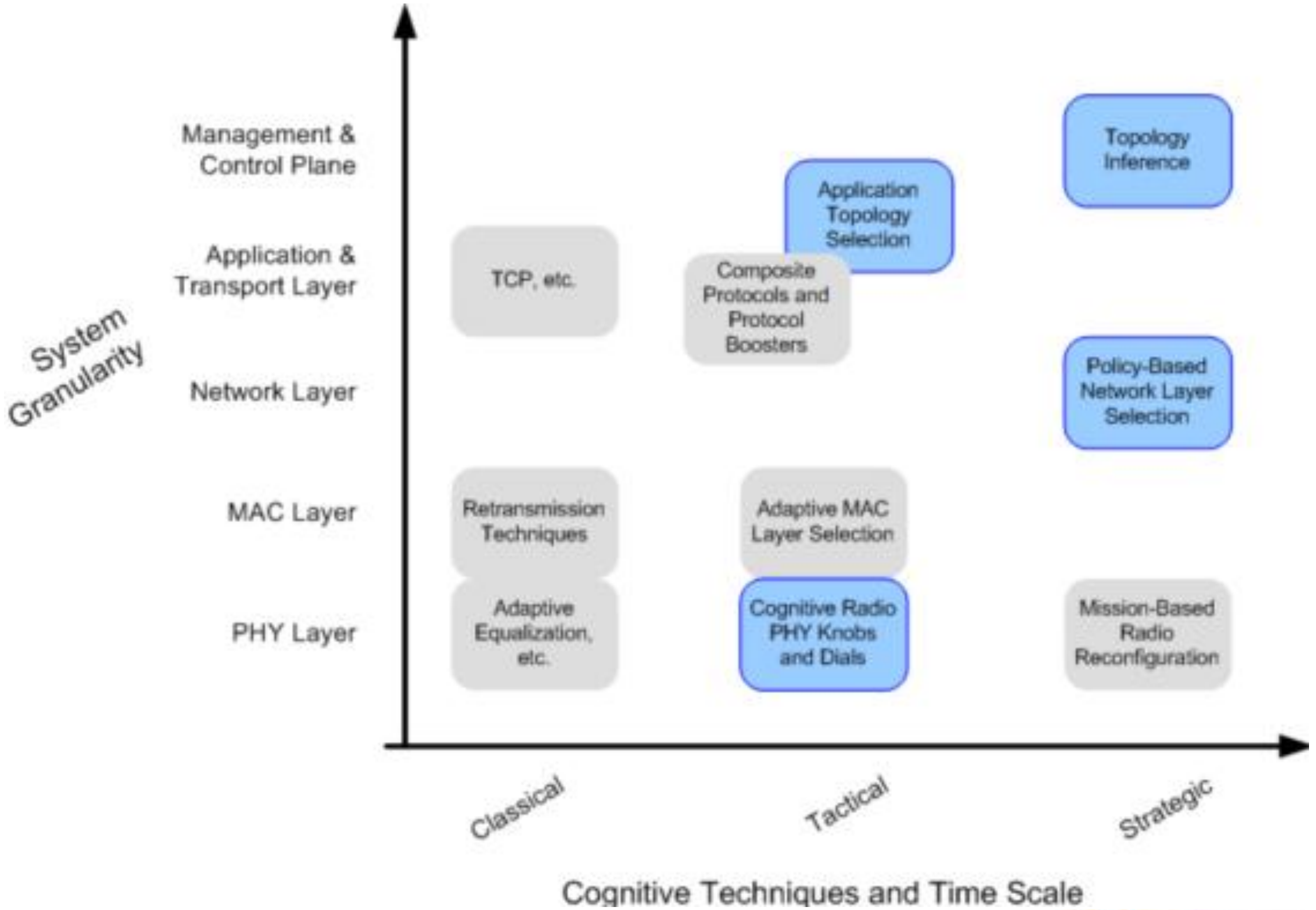
Generalized Cognition Approaches

Variety of Techniques and Technologies May Be Required

Characteristics		Time Rich	
Evaluate consequences of unique sequences of actions far into the future based on situation, policy, and prior experiences	Strategic		Planning Techniques
		Symbolic	
Enforce policies and invoke procedures based on assessment of current situation and limited projections into the future	Tactical		Policy-Based Reactive Control and Knowledge-Based Control
		Numeric	
Respond in predetermined ways to immediate situation based on current status and limited feedback	Classical		Classical Control Based on Stimulus & Response
		Time Limited	

Relationships

Cognitive approaches, system layers, and time scales



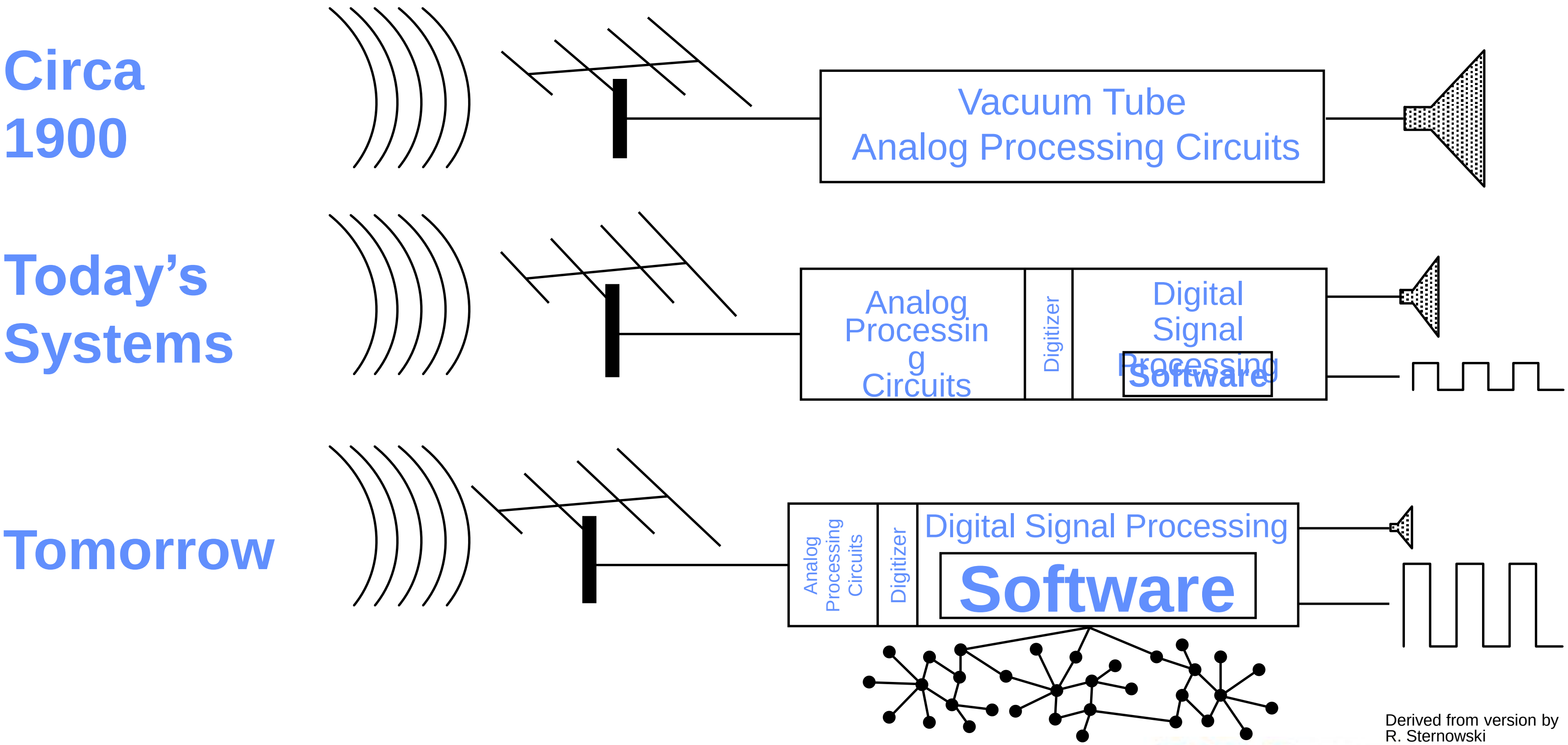
Smarter Networked Nodes

Building cognitive radios

Evolution of Wireless Systems

Receive → *Capture radio waves* → *Extract information from radio waves* → *Present information to user*

Transmit ← *Generate radio waves* ← *Implant information into radio waves* ← *Accept information from user*



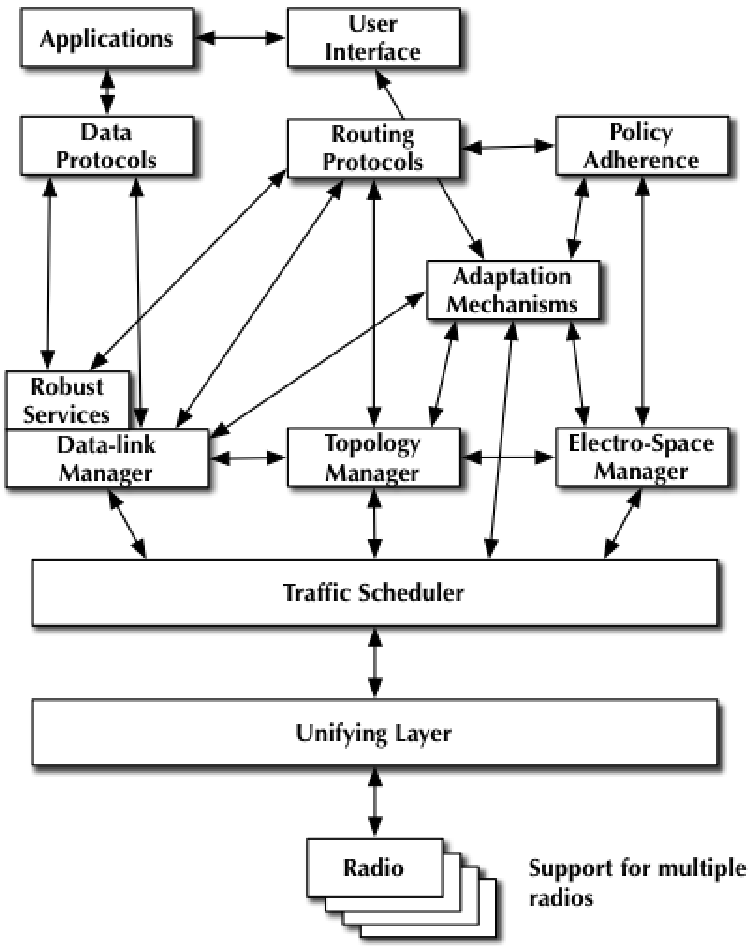
Derived from version by R. Sternowski

INFORMATION & TELECOMMUNICATION TECHNOLOGY CENTER
The University of Kansas

KU Agile Radio (KUAR)

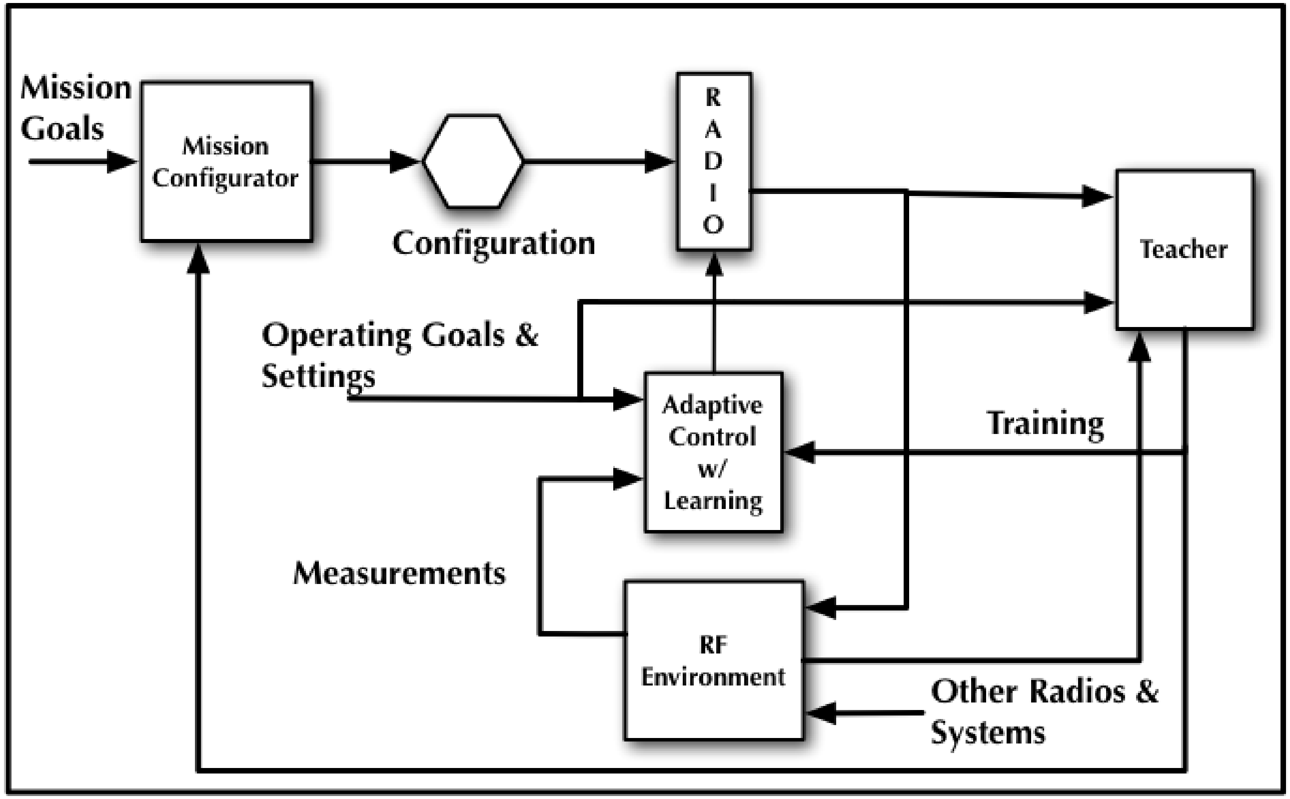
- Compact & portable agile radio
- Consists of RF, digital, and power supply boards
- Flexible hardware
  - Current KUAR supports 5 GHz band operations
  - Future implementation to support 2.4 GHz band
  - 30 MHz independent Tx/Rx complex baseband
  - Waveform: A/D 14-bit @ 80 Msps and D/A 16-bit @ 160 Msps
  - Pentium-M (1.4 GHz), 1 GB RAM / 8 GB Flash
  - Ethernet
  - Xilinx Vertex II Pro; 2 PPC cores; 21K logic cells; 300 MHz
- Flexible software
  - Linux OS (Kernel 2.4)
  - C++ SCA implementation from Virginia Tech (OSSIE) used for software framework
  - KUAR control processor fully participates in wired network with standard network services
  - Capability for dynamic service and spectrum access, and rapid service creation

Cognitive Radio Software Architecture



- Unifying Layer - adapts software to multiple radios
- Traffic Scheduler - time-division multiplexes access to a "conduit"
- ElectroSpace Manager - tracks used/un-used RF resources
- Topology Manager - decides which neighbors to communicate with
- Data-Link Manager & Robust Services - transmits/receives frames
- Adaptation Manager - monitors radio operation and directs operational changes
- Policy Adherence - checks that operation is within regulatory and operational bounds
- Routing Protocols - determines how packets are forwarded through network
- Data Protocols - conventional TCP/IP protocols
- Applications - conventional end-user applications
- User Interface - means for user to direct radio operation

Cognitive Radio Learning Structure



Hours

Milli-Seconds

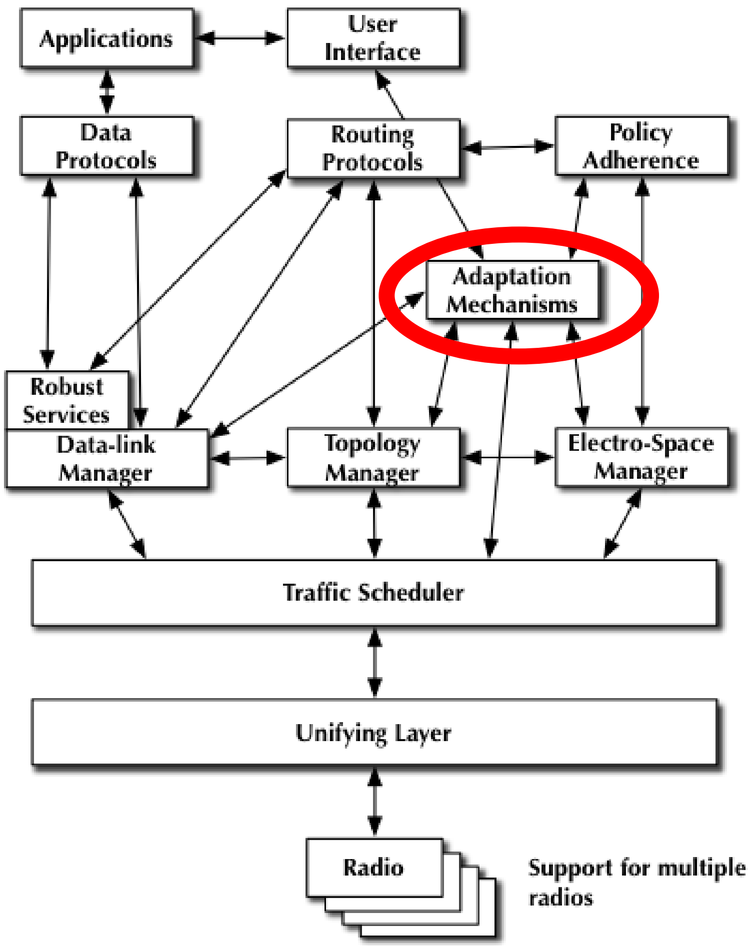
~Minutes/Hours

Radio Adaptation

Adapt radio functions

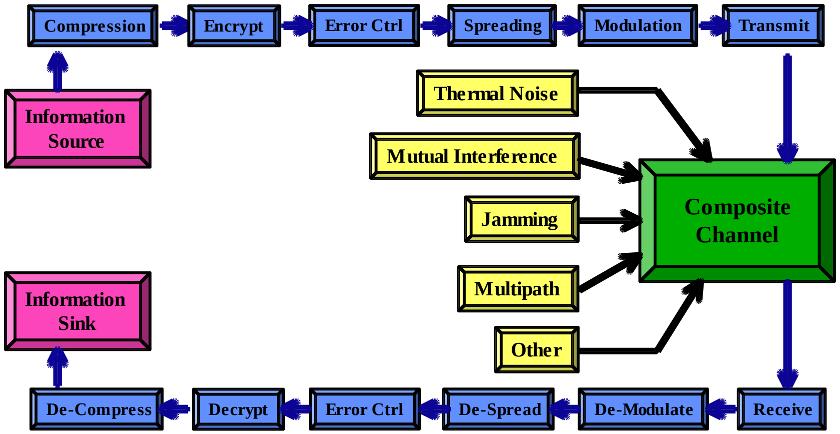
Use cognition to choose amongst options

Cognitive Radio Software Architecture

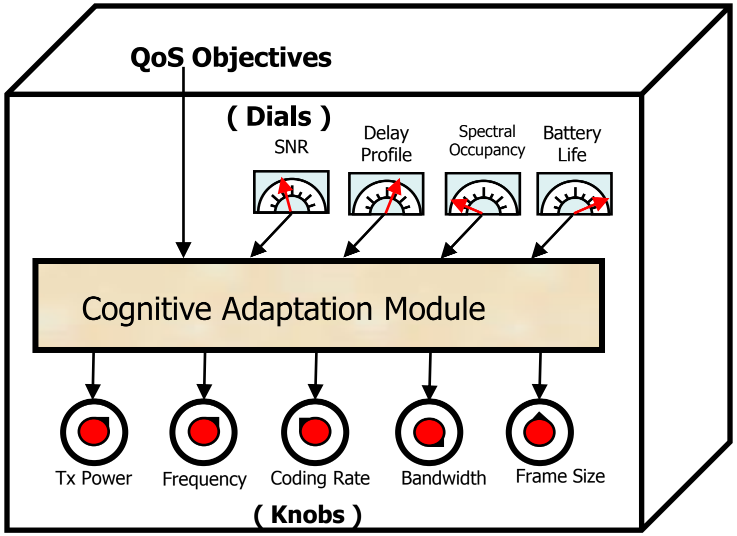


Cognitive Parameters

- General Radio Model
  - Every processing stage is programmable and controllable

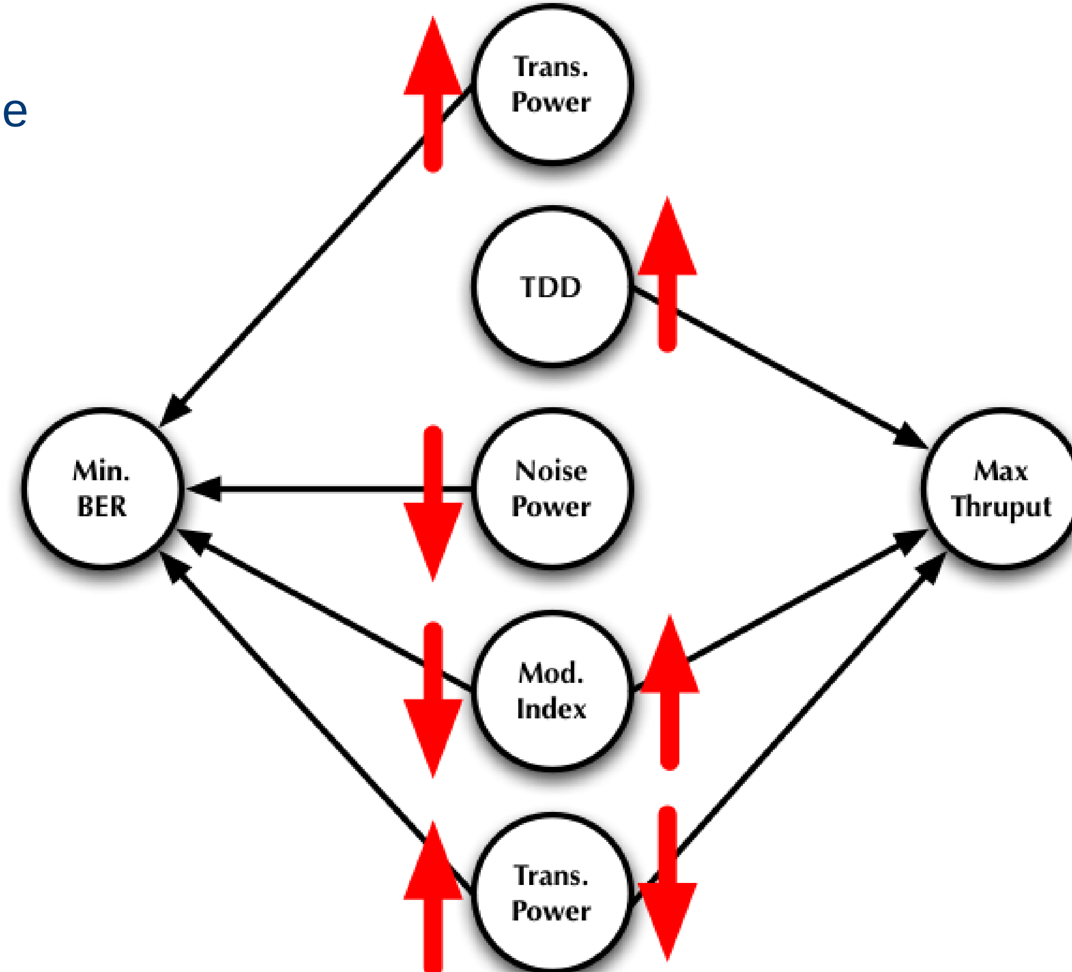


- Transmission parameters (Knobs)
  - Transmit power
  - Modulation
  - Code rate
  - Symbol rate
  - Frame length
- Environmental parameters (Dials)
  - SNR
  - Path loss
  - Battery life
  - Delay spread



Goals and Conflicts

- Radio objectives
  - Direct the solution of the cognitive engine
- Example objectives
  - Minimize BER
  - Maximize throughput
  - Minimize power



Reasoning/Control Approaches

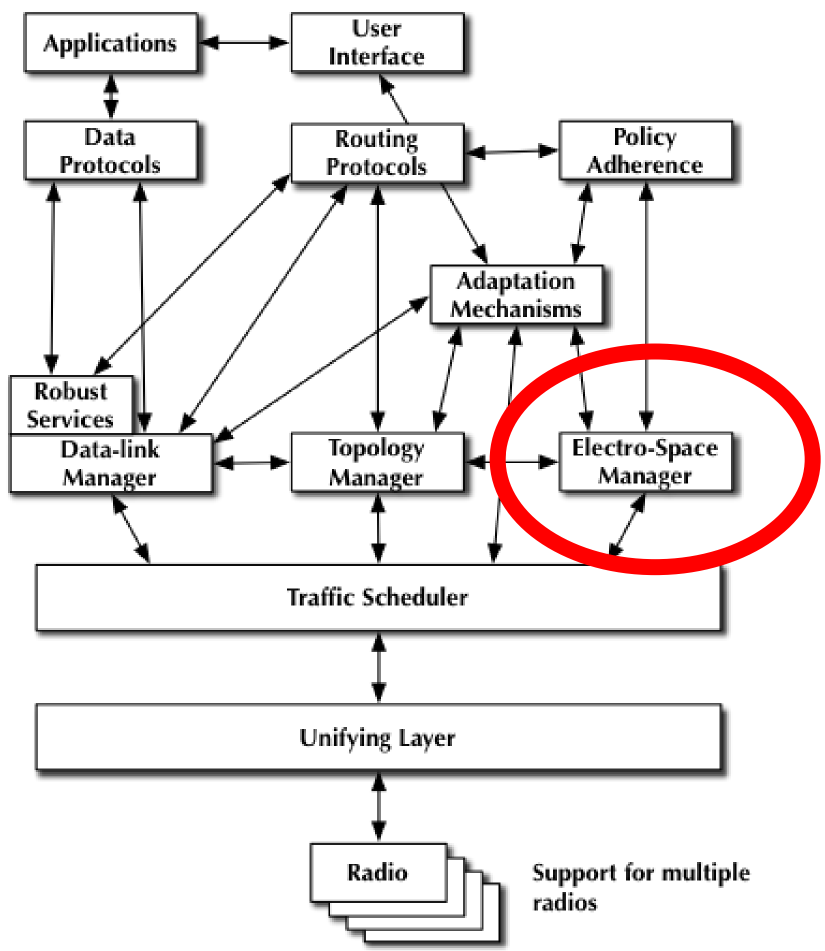
- Exact Methods
  - Advantages: Exact optimal solution can be found
  - Disadvantages: Typically requires at least first derivative of a complex equation; Time complexity (pure random)
- Heuristic Methods
  - Advantages: Lower complexity than exact methods; Increased flexibility with regards to changes in the fitness equation
  - Disadvantages; Sub-optimal solutions
- Simulated Annealing
  - Advantages: Ease of implementation
  - Disadvantages: Only works on single solution (Local optima problem)
- Neural Networks
  - Advantages: Low memory usage, fast output
  - Disadvantages: Processing complexity, training needed, final output not traceable (traceability is needed)
- Genetic Algorithms
  - Advantages: Parallel processing, well suited for large problem spaces
  - Disadvantages: Processing time

Better Radio Networks

Use cognitive techniques to make networks better

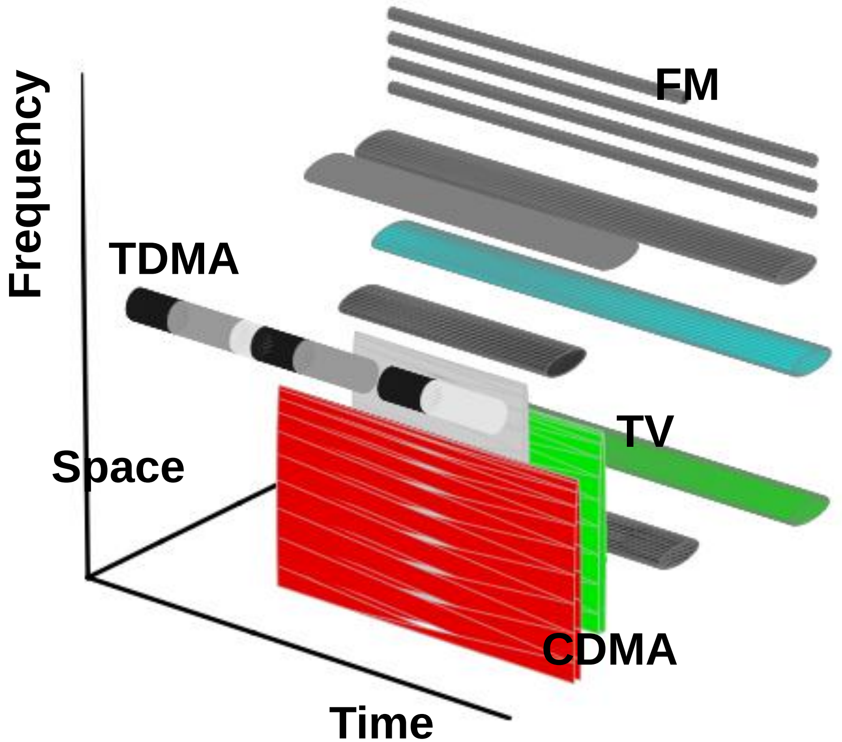
Enable spectrum (electrospace) sharing

Cognitive Radio Software Architecture

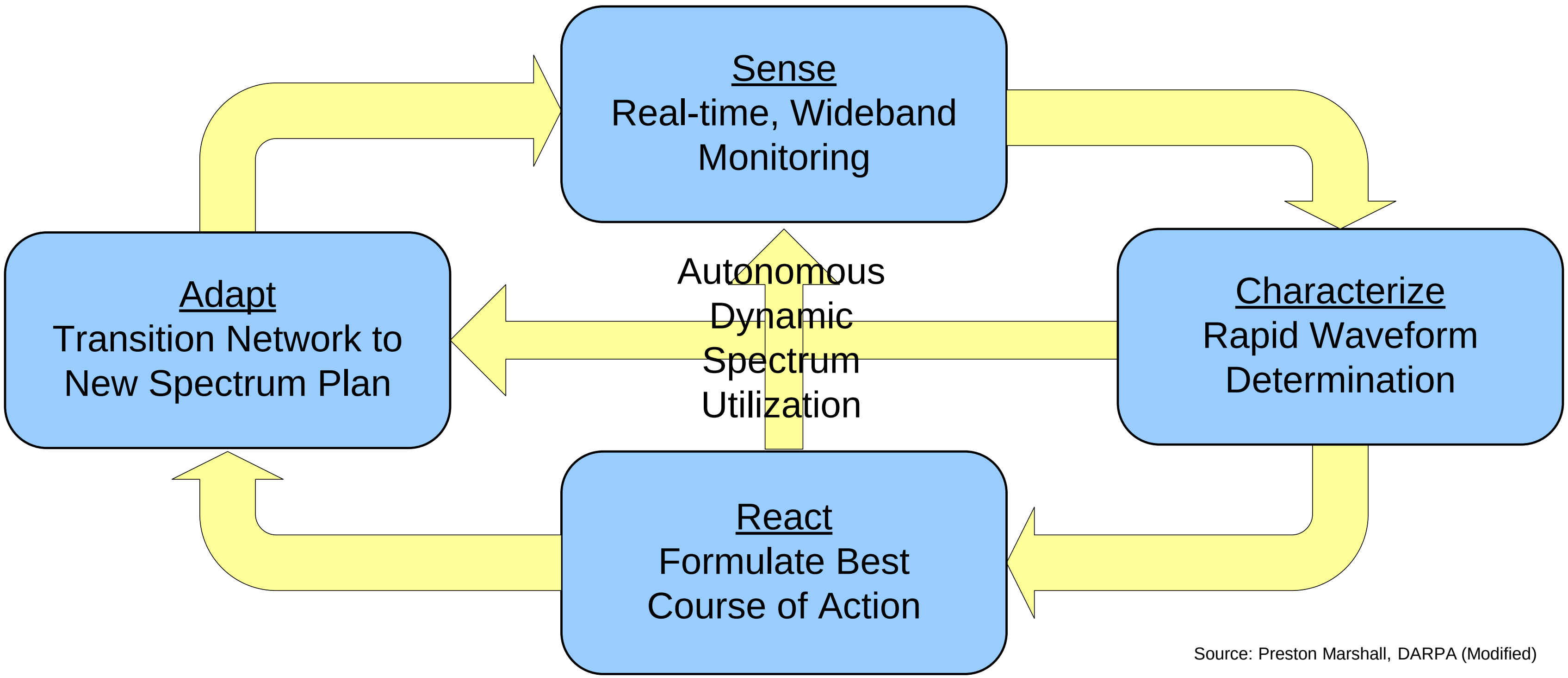


ElectroSpace Resource

- Each type of modulation affects allocation of:
  - Frequency
  - Time
  - Space
  - Signal format (DSS, FH, M-PSK, ...)
  - in the ElectroSpace
- Distributed, network resource
- Coordination between radios required... an orderwire
  - Amateur Q-Codes, AMPS, Cellular
- Develop an ElectroSpace language

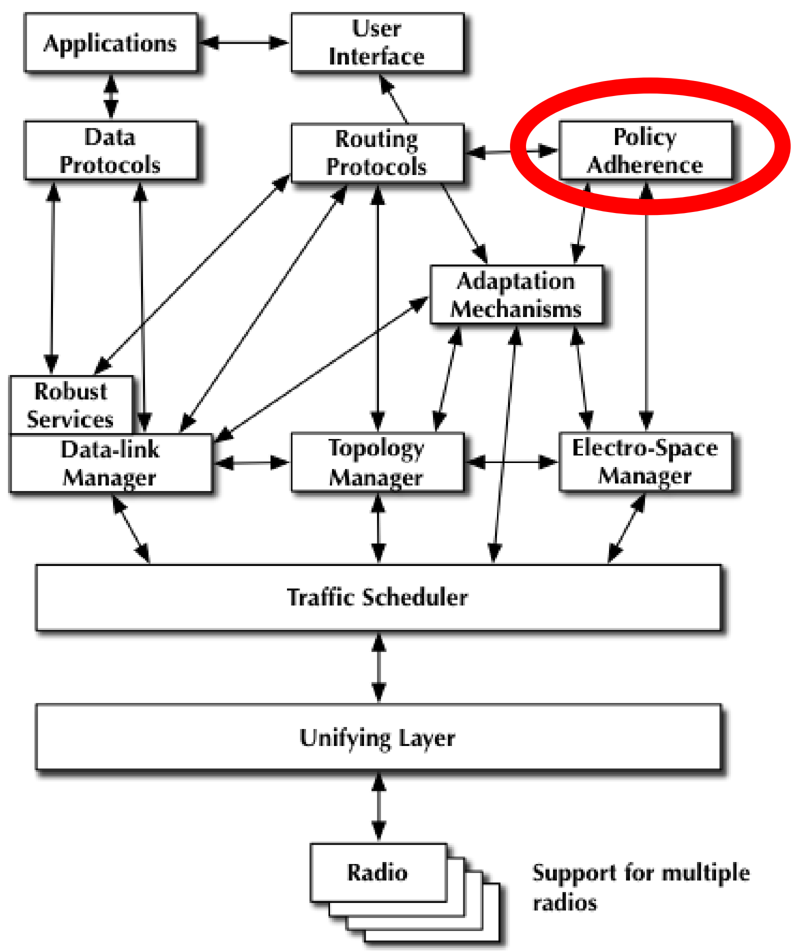


Cognition and Spectrum

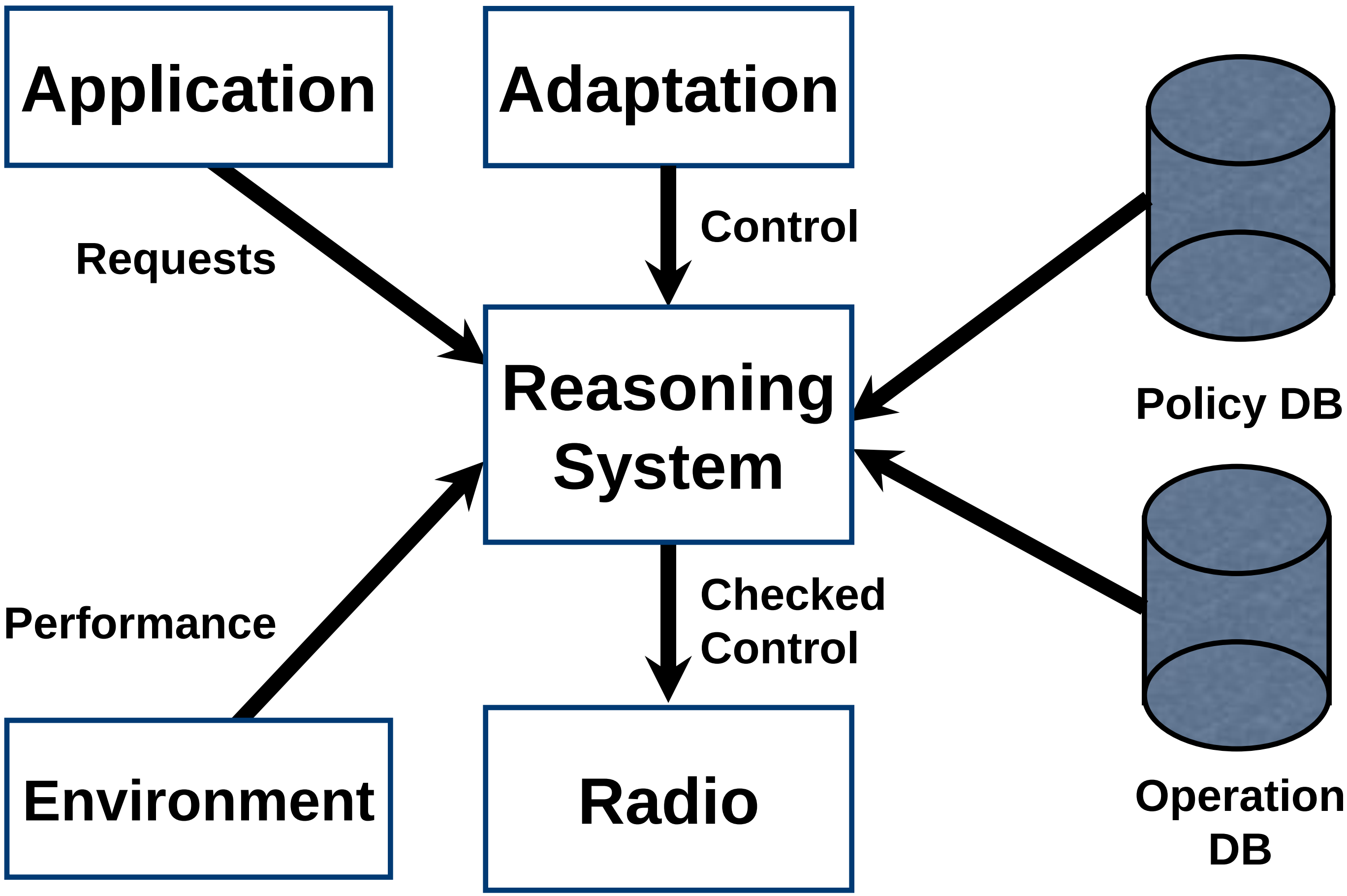


Source: Preston Marshall, DARPA (Modified)

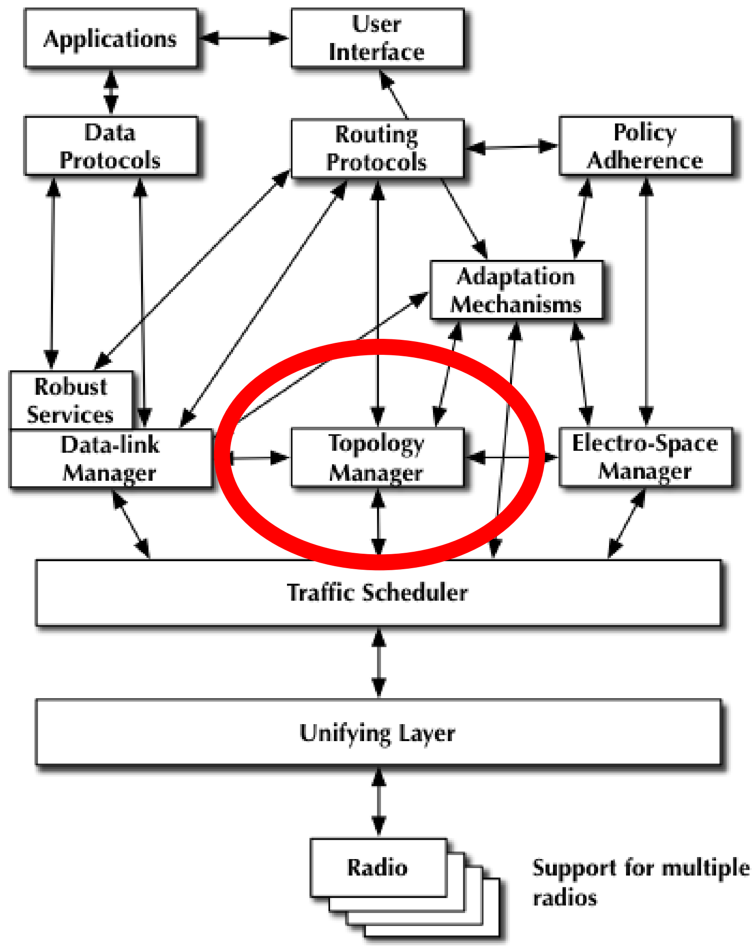
Cognitive Radio Software Architecture



Policy Adherence Software Architecure

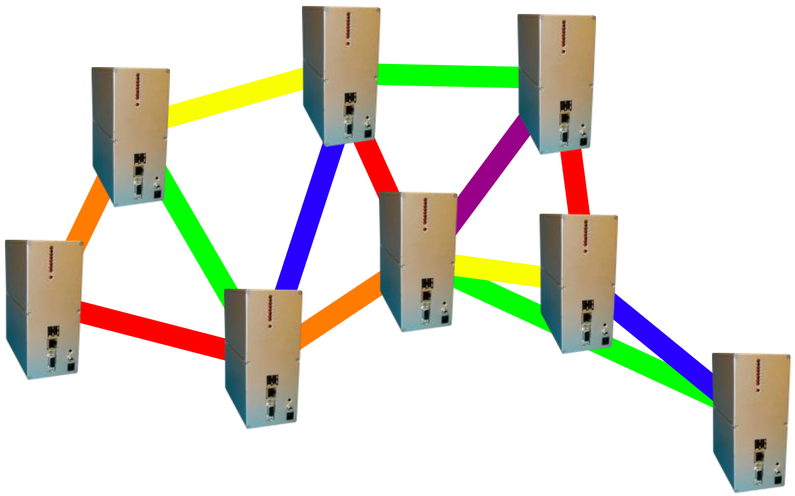


Cognitive Radio Software Architecture



Topology Manager

- Determine which radios should communicate
- Based on:
  - Available ElectroSpace Resources
  - Application load (network queues)
  - Adaptation (determining when to adjust)
- A connection involves
  - Allocation of ElectroSpace
  - Scheduling reception and transmission
  - Adding network routes



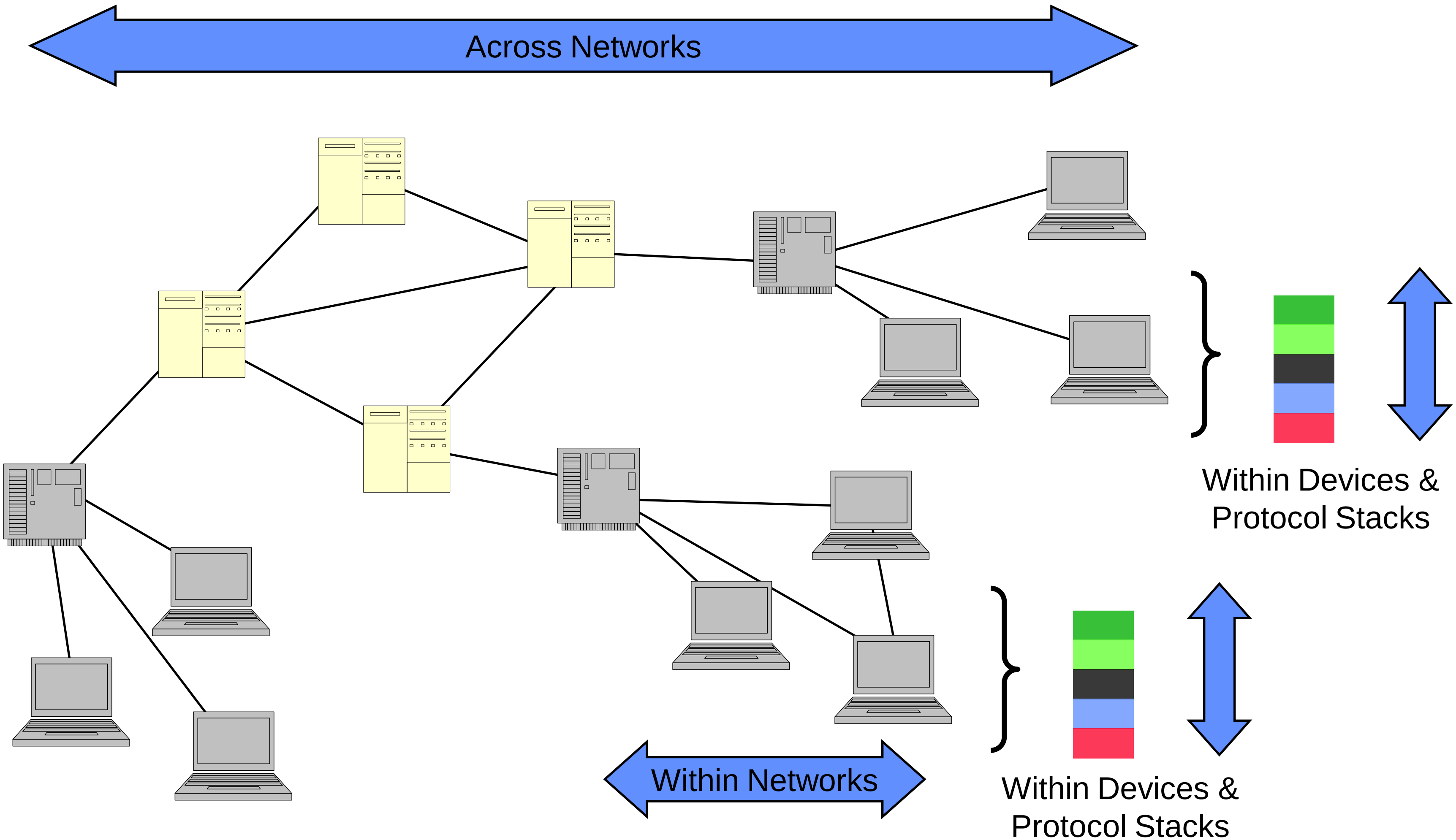
Network Adaptation

Free network architectures from many constraints

Network layers matched to needs

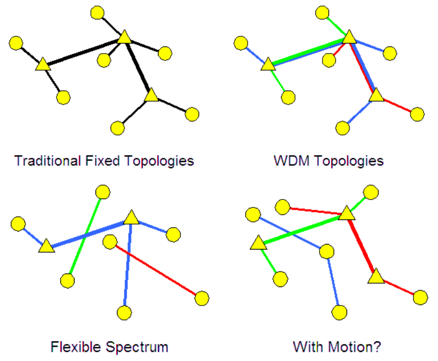
Topology and communication opportunities

Smarter Nodes to Smarter Networks

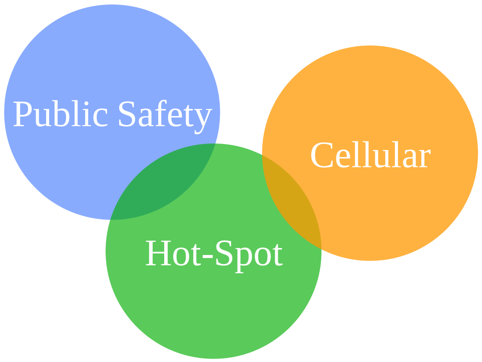


Topology Formation & Interdomain Operations

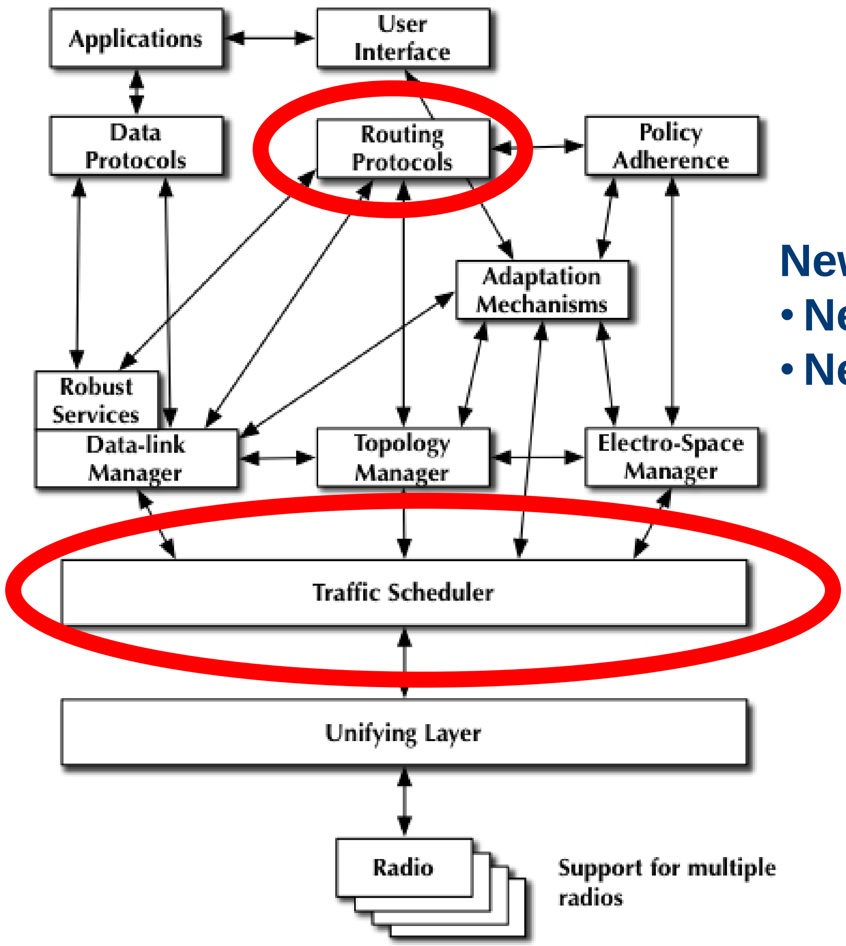
- Determine what topologies are possible & how to choose among them, and automatically configure network based upon topology chosen



- Exploit both spectrum flexibility and availability of several overlapping networks - flexible wireless topologies can provide richer connectivity by exploiting different systems & administrative domains



Cognitive Radio Software Architecture



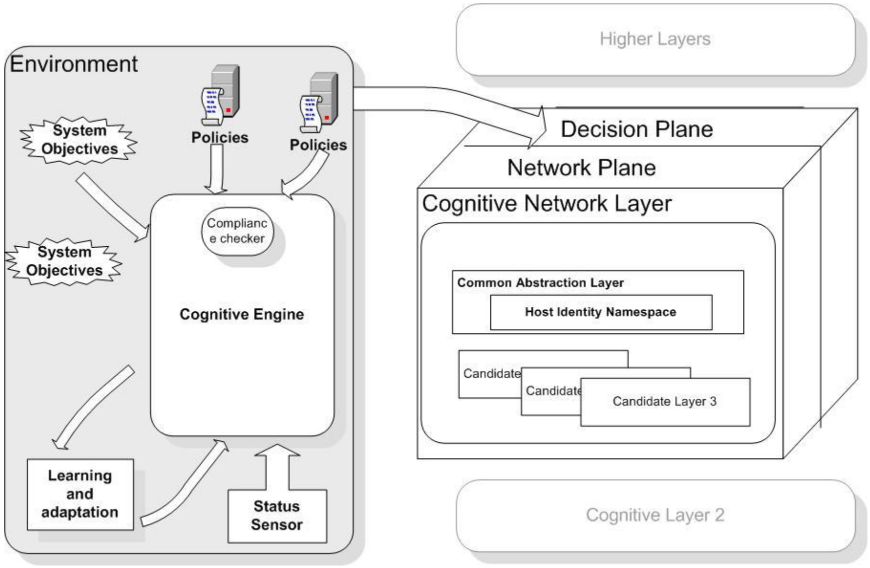
New connections ⇒

- New routes
- New traffic schedules

Network Layer Adaptation

- Dynamic network layer with multitude of sub-layers, each with separate routing and messaging mechanisms (DHT based, IP based, etc.)
- Identity name space to provide uniformity and to preserve host identity across network types → Implementation using Host Identity Protocol (HIP)
- Choice of sub-layer determined by decision function derived from environment parameters (system objectives, prevailing policies, network status, etc.)

- Two control planes
  - Decision plane for management
  - Network plane for operation



Layer Selection and Learning

- Cognitive engine uses decision function to choose appropriate layer
- Currently a rule based system reasoning on a set of environment parameters
- Table shows the list of parameters that are collected as input to the decision function
- Two sets of parameter categories, to capture network and application characteristics

Network characteristics	
Current number of nodes in the network	N
Average Node density	N_d
Mobility Factor	M
Average Latency	L
Maximum Bandwidth	B
Traffic characteristics	
Burstiness measure	T

- The function assigns a weight to each of such selections and calculates a cumulative layer based on the current values of the parameter values
- Need to isolate the combinations of decision parameters that distinctly differentiate candidate layers (e.g., IP and DHT based) in a multidimensional space to drive rule selection
- Need representation of decision parameters and output in standard knowledge representational language to leverage long-term learning

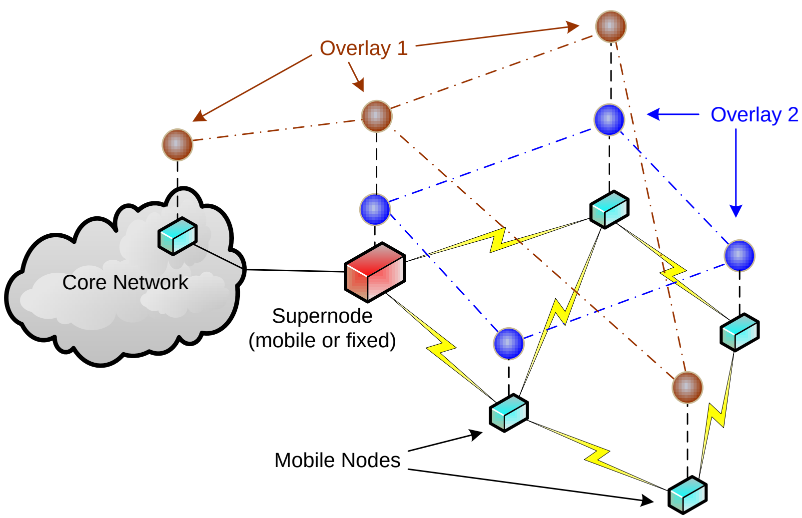
Network Management

Sharing management data with trusted peers

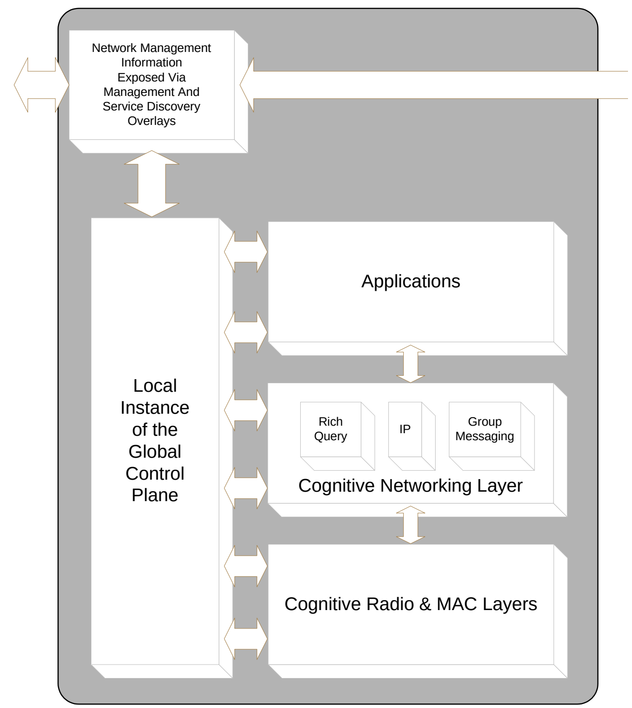
Using cognition for scaling to very large systems

Cognitive Wireless Networks

- Cognitive wireless network with multiple network-layer overlays



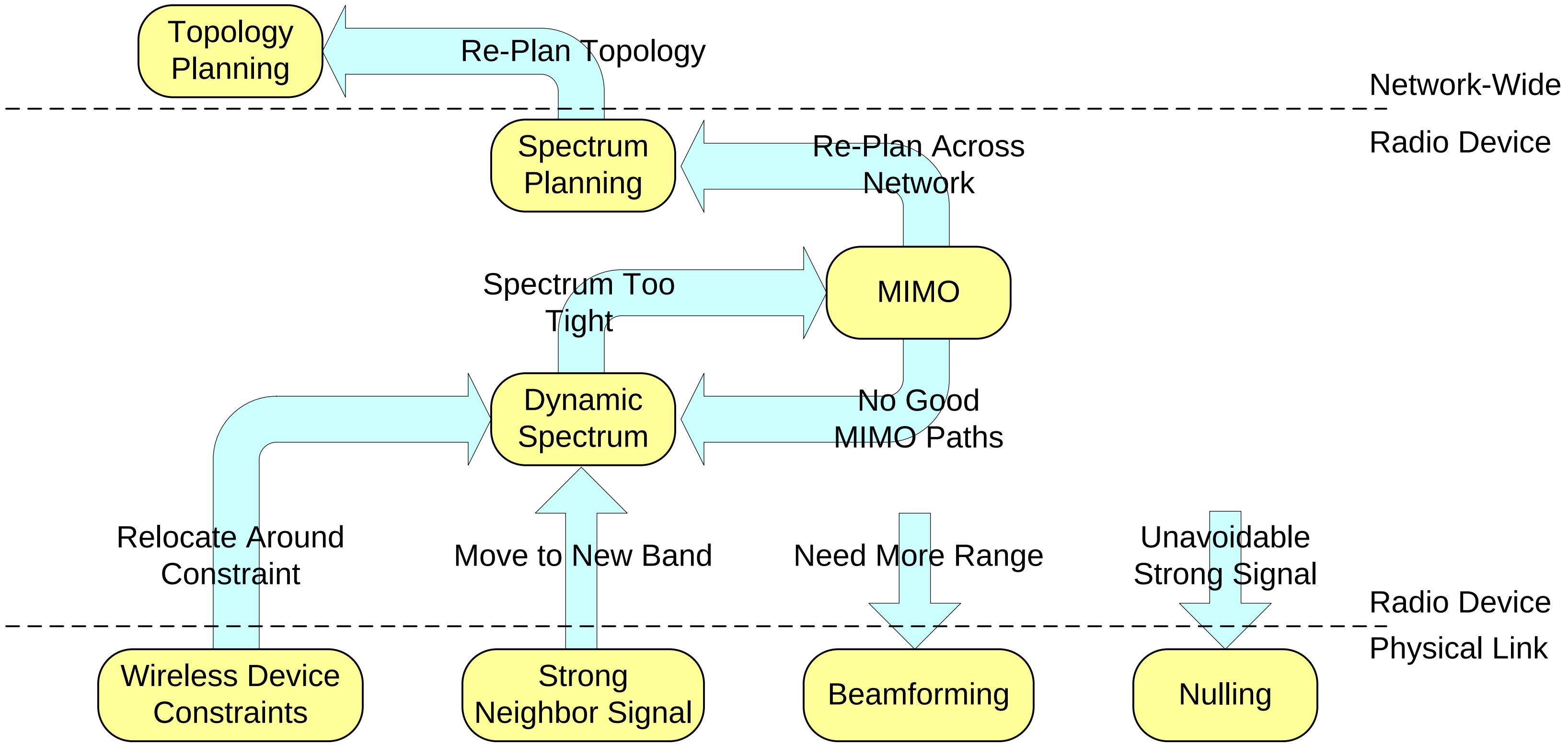
- Cognitive wireless network management



KU INFORMATION & TELECOMMUNICATION TECHNOLOGY CENTER
The University of Kansas

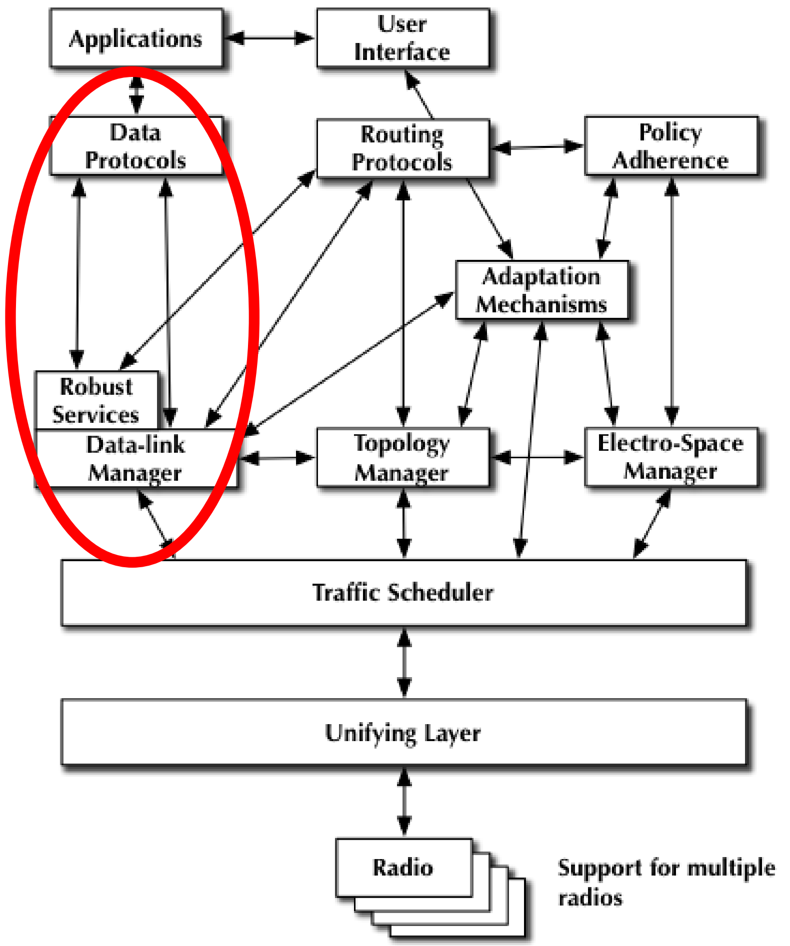
How Cognition Might Help

- Each technology can "throw" tough situations to other more suitable technologies



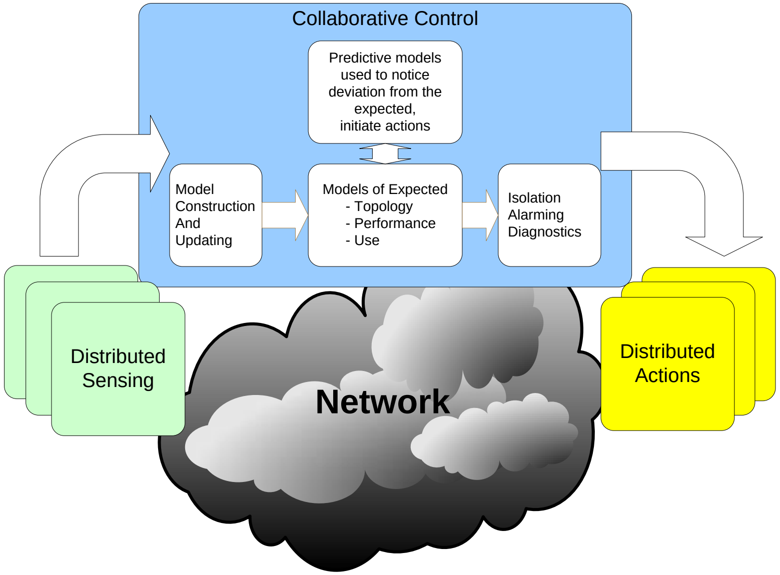
Source: Preston Marshall, DARPA (Modified)

Cognitive Radio Software Architecture



Collaborative Network Management

- Help enable cognitive systems that sense, evaluate, and react to the network
- Collaborate among nodes in distributed systems to perform measurements & cognitive functions - share measurements, inferences
- Tactical data network example
  - Distributed network sharing multimedia among disparate users and across network links
  - Tune distribution of tactical data to optimize for network conditions and policy



Topology Discovery & Selection

- Can we collaborate on measurements to get a more accurate topology model, and to choose the best topologies?
- Topology via collaborative measurements
  - Hybrid model utilizing both direct and indirect measurements from many sources to many destinations
  - MCMC using Gibbs Sampler for inference
  - Algorithm is iterative, use last best as starting point when a change is suspected
- Topology via collaborative inference
  - Extend the distributed data collection to now include distributed inference
  - Based on distributed approaches to create phylogenetic trees
  - Split topology space into regions
- Close the loop - react to topology inference
  - Collaborate to optimize data dissemination
  - Optimize based on policies about utilization of particular nodes and links
  - Optimize based on current status of node and link such as uptime, available bandwidth, and learned features such as reliability over time

ReTarget

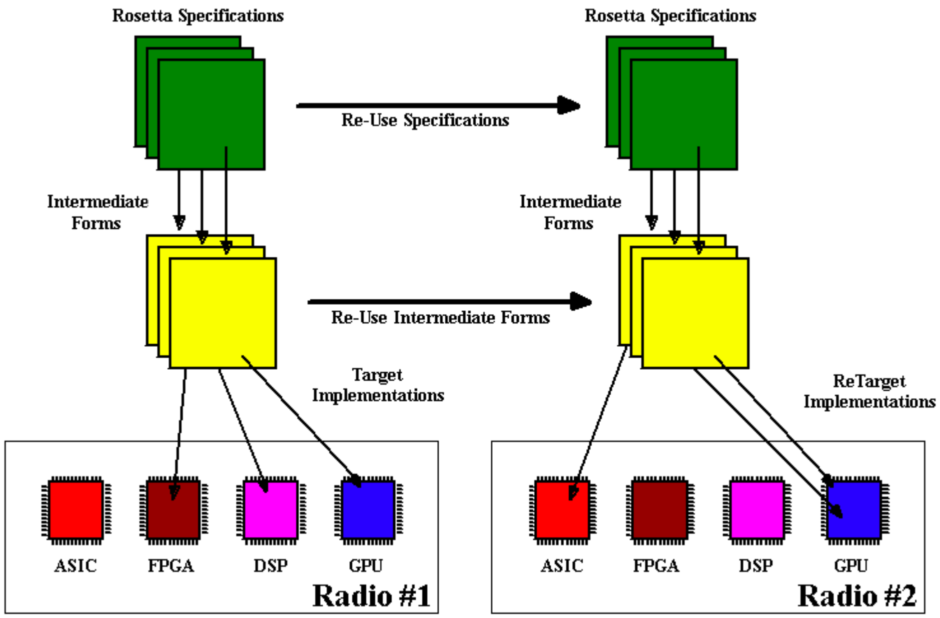
Rethinking design

Motivation

- Software defined radio (SDR) platforms allow radio systems to be dynamically reconfigured to adapt to new waveforms
- SDR waveform implementations combine a variety of hardware (analog, ASIC, FPGA) and software (DSP and CPU) components
- Radio hardware platforms will evolve quickly, approximately every 12-18 months, and be a combination of new hardware and programmable components
- Adapting existing components to different implementation fabrics requires burdensome re-engineering effort
- New hardware & programmable components create need to synthesize to disparate fabrics from a common design source
  $\Rightarrow$ re-target a radio design to new platforms

Design once, use many.

ReTarget Approach



- Rosetta
  - System level design language captures design intent and system constraints
  - IEEE standards project, P1699
  - Specifications translated to a functional intermediate language
- Intermediate language
  - Utilize non-functional specifications to drive system partitioning and synthesis
  - Perform rule-based optimizations as source-to-source transformations on intermediate language
- Platform
  - Generate fabric-specific synthesis artifacts

Why and Where

Impact

- Rosetta allows heterogeneous system specification and composition based on domains, allowing system-level design and analysis
- Functional intermediate form simplifies source-to-source optimizations, as well as target-specific optimizations

Status

- Prototype toolset developed
- Demonstrates synthesis from a common intermediate source
- Both software (C) and hardware (FPGA) implementations synthesized
- Rosetta to intermediate language mappings in development

Credits

- In general – Gary Minden
- Cognitive Radio Engine – Timothy Newman
- Network Layer Cognition – Muthukumaran Pitchaimani
- Collaborative Approaches to Probabilistic Reasoning in Network Management – Benjamin J. Ewy
- ReTarget – Garrin Kimmel, Ed Komp, Perry Alexander

Intelligence in the Network