Interoperability of Future Information Systems

ONR Grant Number: N00014-02-1-0499

Daniel Siewiorek, Katia Sycara (PI's)

Joseph Giampapa, Ritika Sanghi, Aaron Steinfeld

School of Computer Science

Carnegie Mellon University

Carnegie Mellon

Outline

- **Motivation**
- Research Approach
- Taxonomy Findings
- Agent Development Process
- What's Next

Carnegie Mellon

Motivation

- Resolving network interoperability problems is difficult and time consuming
  - heterogeneity, admin policies, etc
- Advances in network flexibility will improve underlying performance
  - New HCI methods and tools will be required to enhance user awareness and problem resolution

Carnegie Mellon

Outline

- Motivation
- **Research Approach**
- Taxonomy Findings
- Agent Development Process
- What's Next

Research Questions

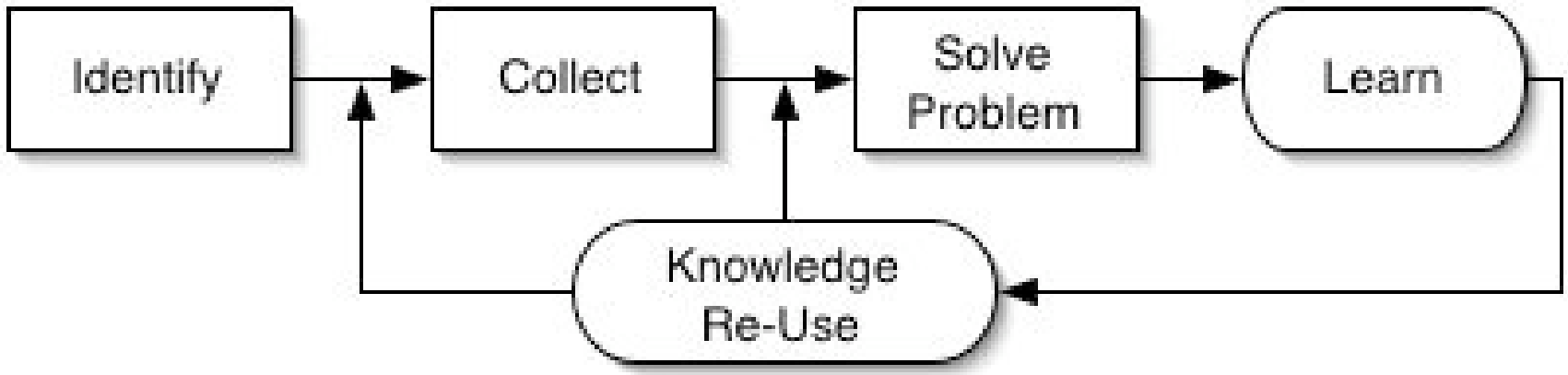
- How does the user diagnose and remedy network interoperability problems?
- What options exist given the obstacles imposed by intermediary policies?

Research Plan

- Generate taxonomy of remote access interoperability problems
- Define agent interactions with existing network tools and formulate service profiles for future tools
- Develop agents for the resolution of network interoperability problems

Carnegie Mellon

Interoperability Problem Resolution Model (IPRM)



Carnegie Mellon

Interoperability Problem Resolution Model - 1

Model	Current	Research Goals
Identify the problem	Recognition of a real problem rather than a temporary event	Proactive monitoring and problem avoidance, periodic state maintenance, and formulation of constraining hypotheses
Collect Parameters and Symptoms	Manual, often requiring the user and expert to negotiate common definitions, mental models, and actions	Automatic and proactive, the ability to provide an intelligently structured "parameter dump"

Carnegie Mellon

Interoperability Problem Resolution Model - 2

Model	Current	Research Goals
Solve the Problem	a) Human reasoning, often reduced to previously solved problem, *or* b) Human reasoning augmented with Case Based Reasoning tools	A suite of autonomous and semi-autonomous actions: a) Agent negotiation and suggestions b) Middle agent interactions with network tools c) Agent-guided user actions
Learn	Cheat sheets and, rarely, failure analysis	Sharing of solutions between end-user and help desk agents
Re-use Knowledge	Knowledge base, FAQs, bug reports	Dissemination, indexing of problems and solutions

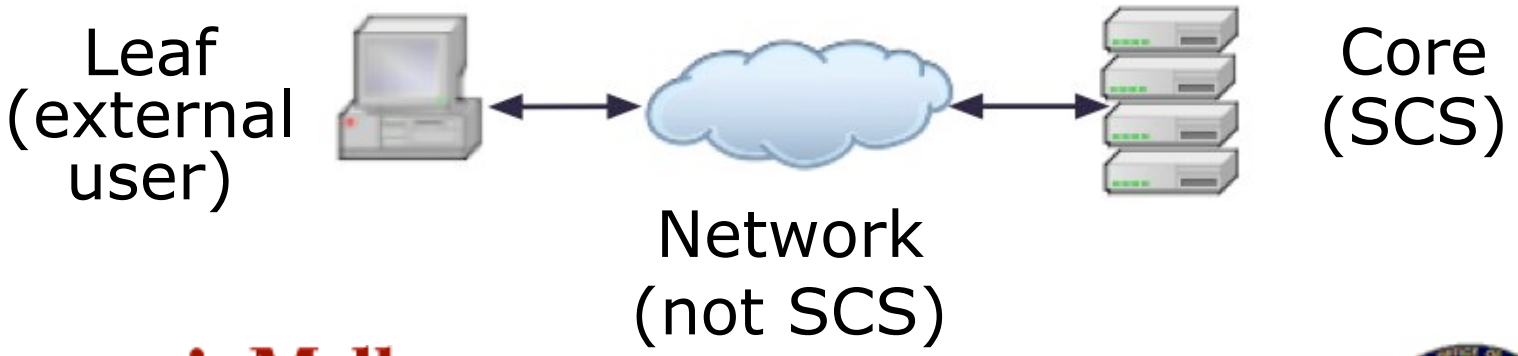
Outline

- Motivation
- Research Approach
- **Taxonomy Findings**
- Agent Development Process
- What's Next

Carnegie Mellon

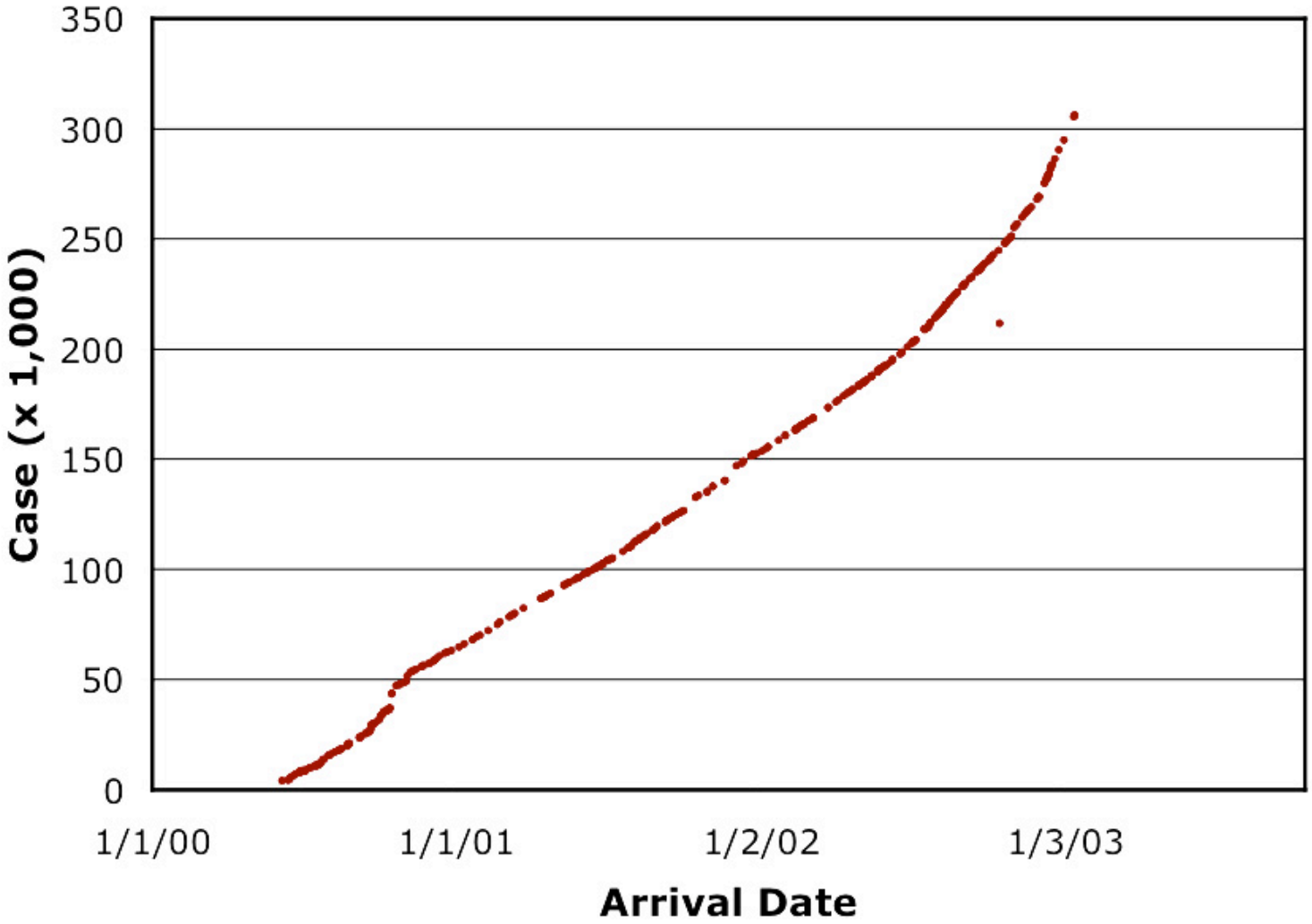
Taxonomy Findings

- SCS remote access trouble ticket case data for 6/5/2000 - 1/15/2003
- 528 Cases
  - Help only: 414, of these…
    - Single configuration events (“Single”): 88
    - Requests for modem numbers: 137



Carnegie Mellon

Case Flow



Carnegie Mellon

Analysis Set 1

- 414 Help cases without outliers:
  - Zero or null Hours to Resolve: 12
  - Over 1,000 Hours to Resolve (notes): 4

By Type

- Phone number queries (significant)
- Network problems consume time fast

Problem Type	N	Hours to Resolve		
		Mean	Std Dev	Sum
Core	69	58	111	4,013
Network	45	77	132	3,447
Leaf	66	60	133	3,957
Single	86	52	104	4,490
Phone Number	132	27	86	3,523
Overall	398	49	110	19,431

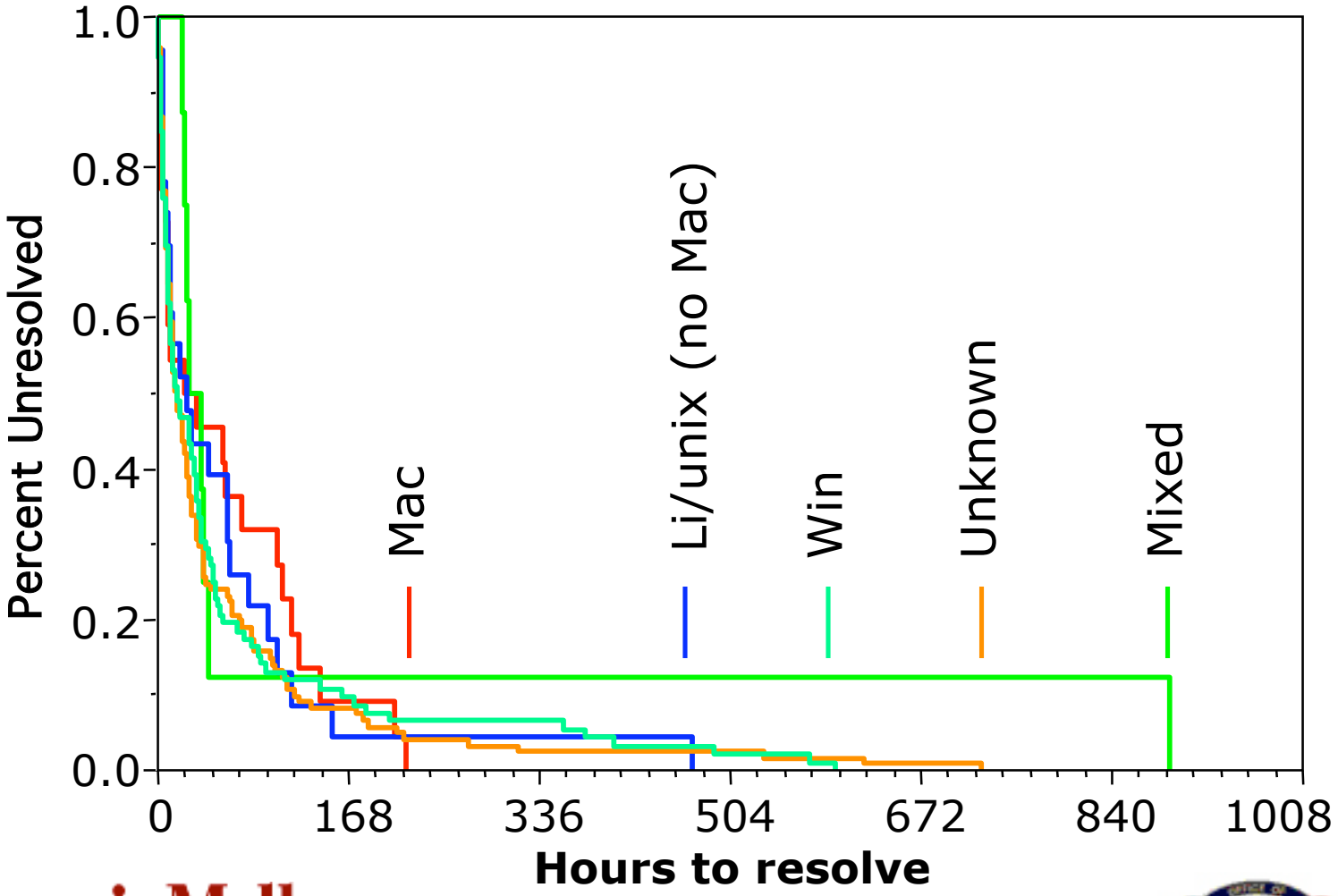
Carnegie Mellon

By Operating System

- Macs: quicker & less variable
- Mixed OS & Unknown (significant)

Operating System	N	Hours to Resolve		
		Mean	Std Dev	Sum
Win	123	52	113	6,450
Mac	31	47	62	1,444
Li/unix (no Mac)	25	56	97	1,393
Unknown	211	43	101	9,036
Mixed	8	139	304	1,109
Overall	398	49	110	19,431

Duration, by Operating System

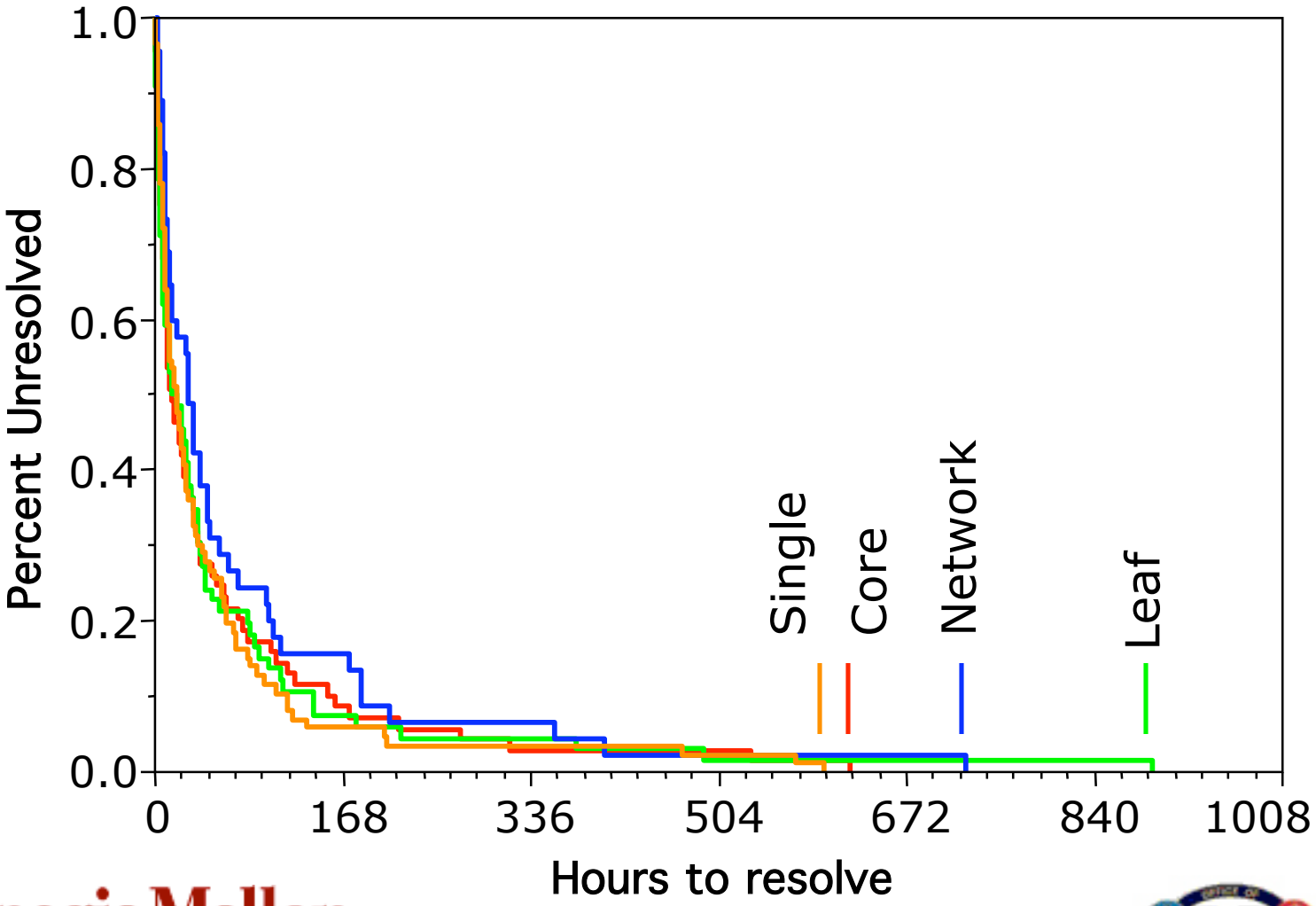


Carnegie Mellon

Analysis Set 2

- 414 Help cases without outliers or phone number requests:
  - Zero or null Hours to Resolve: 12
  - Over 1,000 Hours to Resolve: 4
  - Requests for modem numbers: 132

Duration, by Type



Carnegie Mellon

Modes: DSL, Modem, Wireless

- Combined are usually requests for same IP # in both modes
- No significant effects

	N	Hours to Resolve		
		Mean	Std Dev	Sum
Other	70	73	124	5,134
Modem	119	54	103	6,482
Wireless	26	50	121	1,312
DSL	43	44	82	1,890
DSL, Modem	5	29	47	147
DSL, Wireless	3	314	500	943
Overall	266	60	118	15,908

Carnegie Mellon

Security Policies: VPN, Realm

- 41% cases & 47% time involved either VPN or other security, authentication, or registration issues
- VPN and VPN*Realm (significant)

	N	Hours to Resolve		
		Mean	Std Dev	Sum
None	158	53	112	8,356
Realm	54	53	108	2,855
VPN	42	104	156	4,375
VPN, Realm	12	27	29	322
Overall	266	60	118	15,908

Carnegie Mellon

Very Little Knowledge Re-use

- Root Cause or Solution either
  - Not found
  - Not documented
- No significant effects

	N	Hours to Resolve		
		Mean	Std Dev	Sum
Fully Documented	102	56	108	5,681
Unknown Solution	35	41	99	1,435
Unknown Root Cause	34	58	93	1,972
Both Unknown	95	72	141	6,820
Overall	266	60	118	15,908

Carnegie Mellon

Taxonomy Findings Summary

- 22% from configuration changes
- 49 hrs/case for all help related
  - 60 hrs/case for subset not including phone number requests
- Security policy issues are frequent
- Very little knowledge sharing/re-use
  - Extracted by hand, rarely in existing database fields

Outline

- Motivation
- Research Approach
- Taxonomy Findings
- **Agent Development Process**
- What's Next

Carnegie Mellon

Agent Development Process

- Model the Problem Domain
- Map Agents and Service Descriptions to:
  - Interoperability Problem Resolution Model, and
  - Problem Domain
- Implement, Deploy, Test, Evaluate
- Automatic Process Refinement

Carnegie Mellon

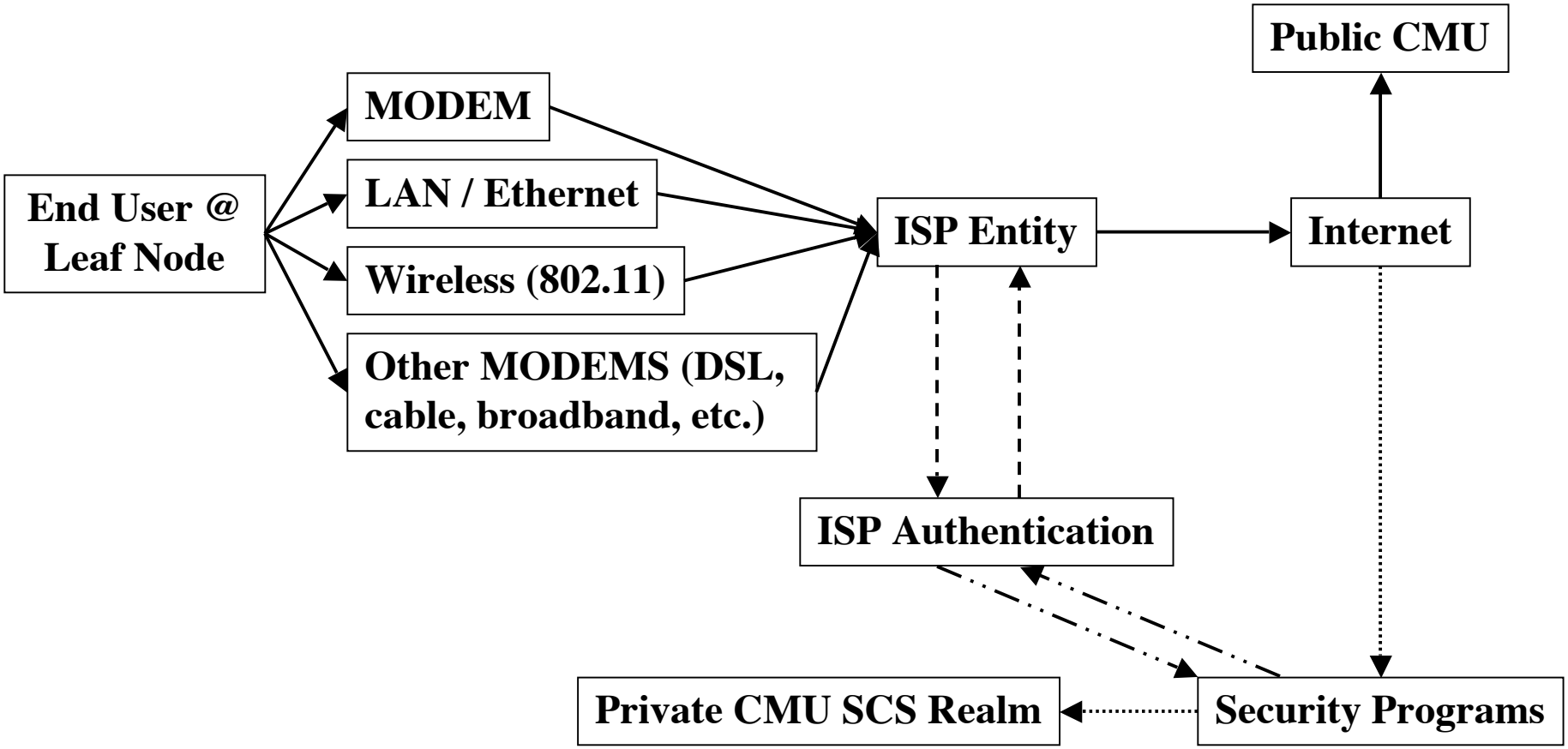
The Problem Domain Model

- Multiple Views and Options Intersect
  - Connectivity Model
  - Connectivity Security Model
  - Security Programs and Features
    - VPN, SSH, SCP, Kerberos
- Typical Motivating Applications
  - Interact with the above 3 models
  - Multiple ways to achieve application goals
  - Users get lost in the intersections

Typical Motivating Applications

- E-mail
  - Send and receive
  - From: on- or off- campus
- Intranet Quality of Service (QoS)
  - Institute-wide access
    - Printing, e-service subscriptions
    - Software licenses, downloads and updates
  - Bandwidth/speed
- File Transfer & File System Access

Generic Connectivity Model

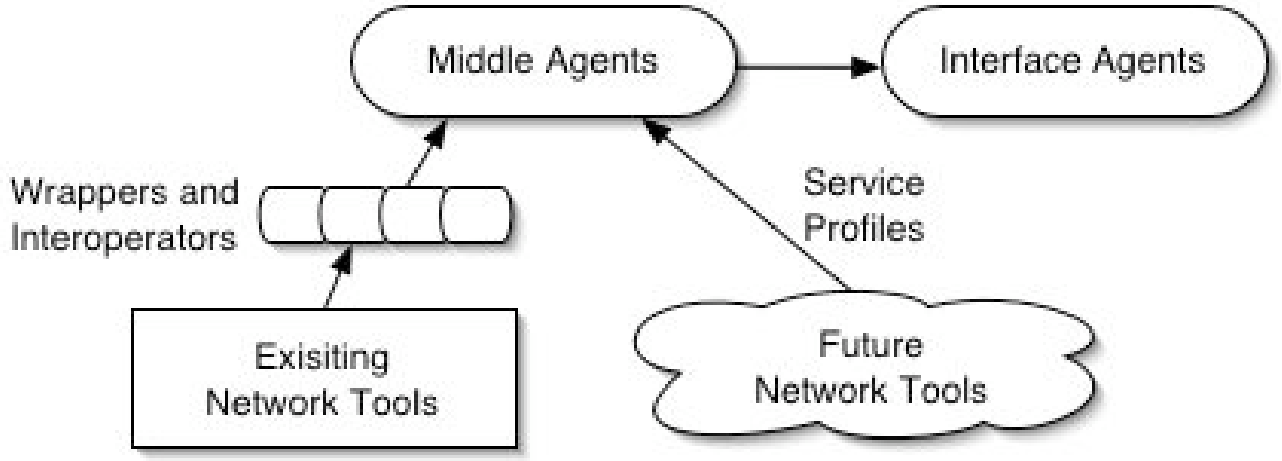


Carnegie Mellon

Mappings

- Each Connection between nodes:
  - Indicates a possibly new authentication step
  - Puts the user in a new application and access rights context
  - Potentially grants the user new privileges
  - Can be modeled as a service description
- Each Node:
  - Has (potentially) verifiable configuration parameters
  - Can be modeled as an agent
- A User's Connectivity Problem
  - Possibly parameterized by goals of using certain applications
  - Resolved automatically by agent / service description matching

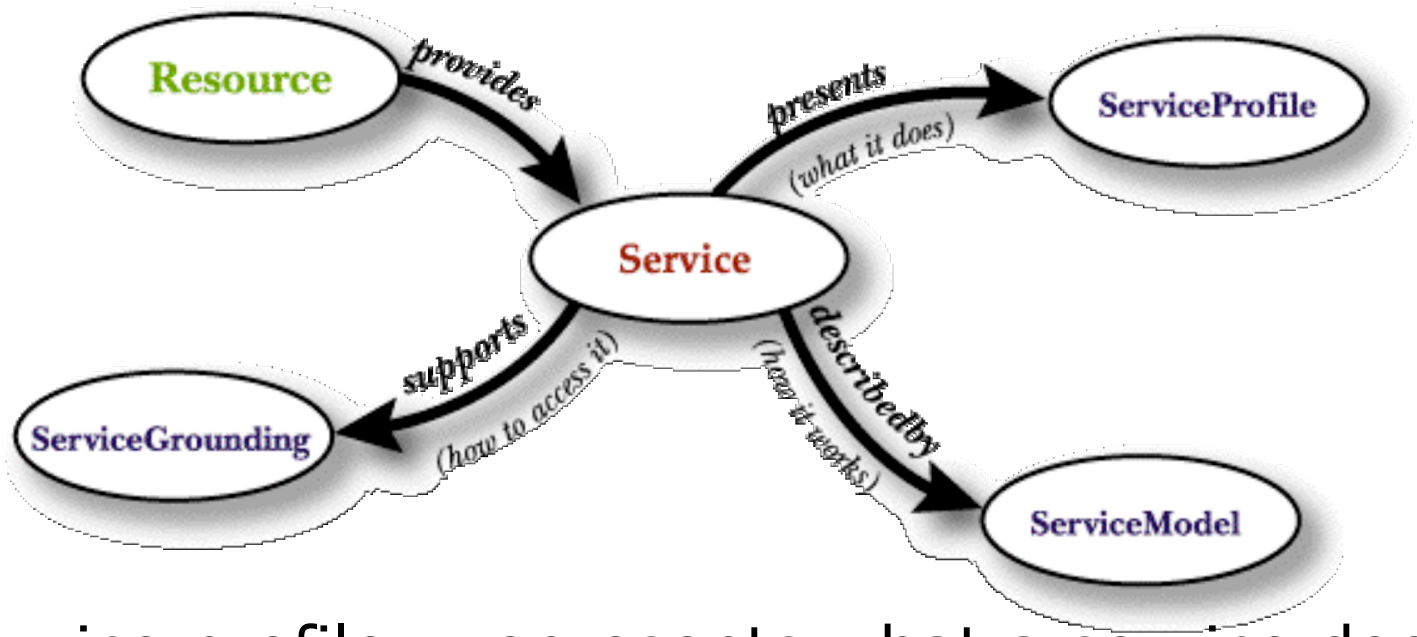
System Architecture



- Agents will have models of application, connectivity, and security tasks
- Agents will shadow local and remote applications
- Agents will also interact with SysAd agents for updates and policy changes

Carnegie Mellon

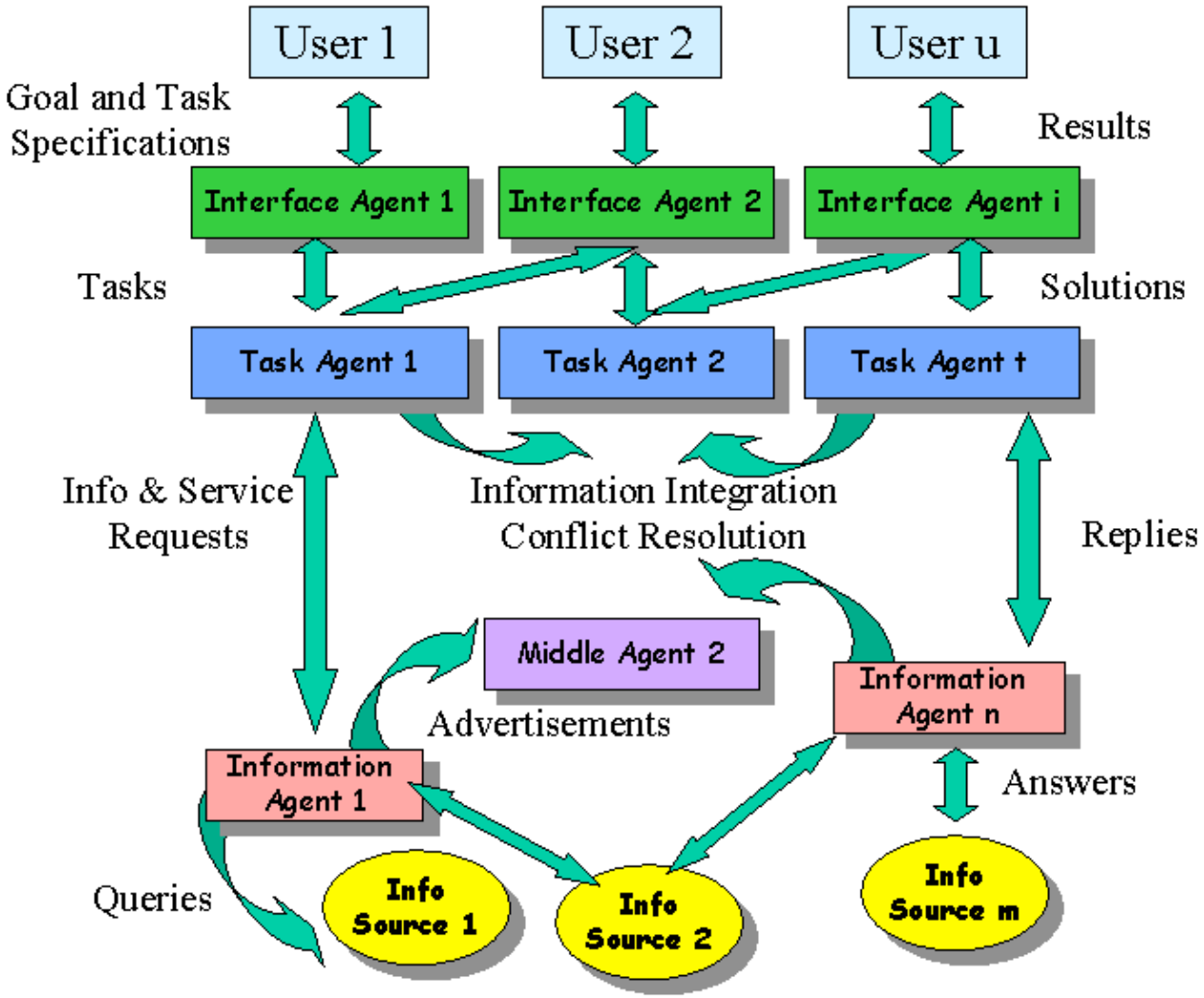
Service Descriptions



- Service profile: represents what a service does
- Service model: describes how a service works
- Service grounding: specifies service access information

Carnegie Mellon

RETSINA Functional Architecture

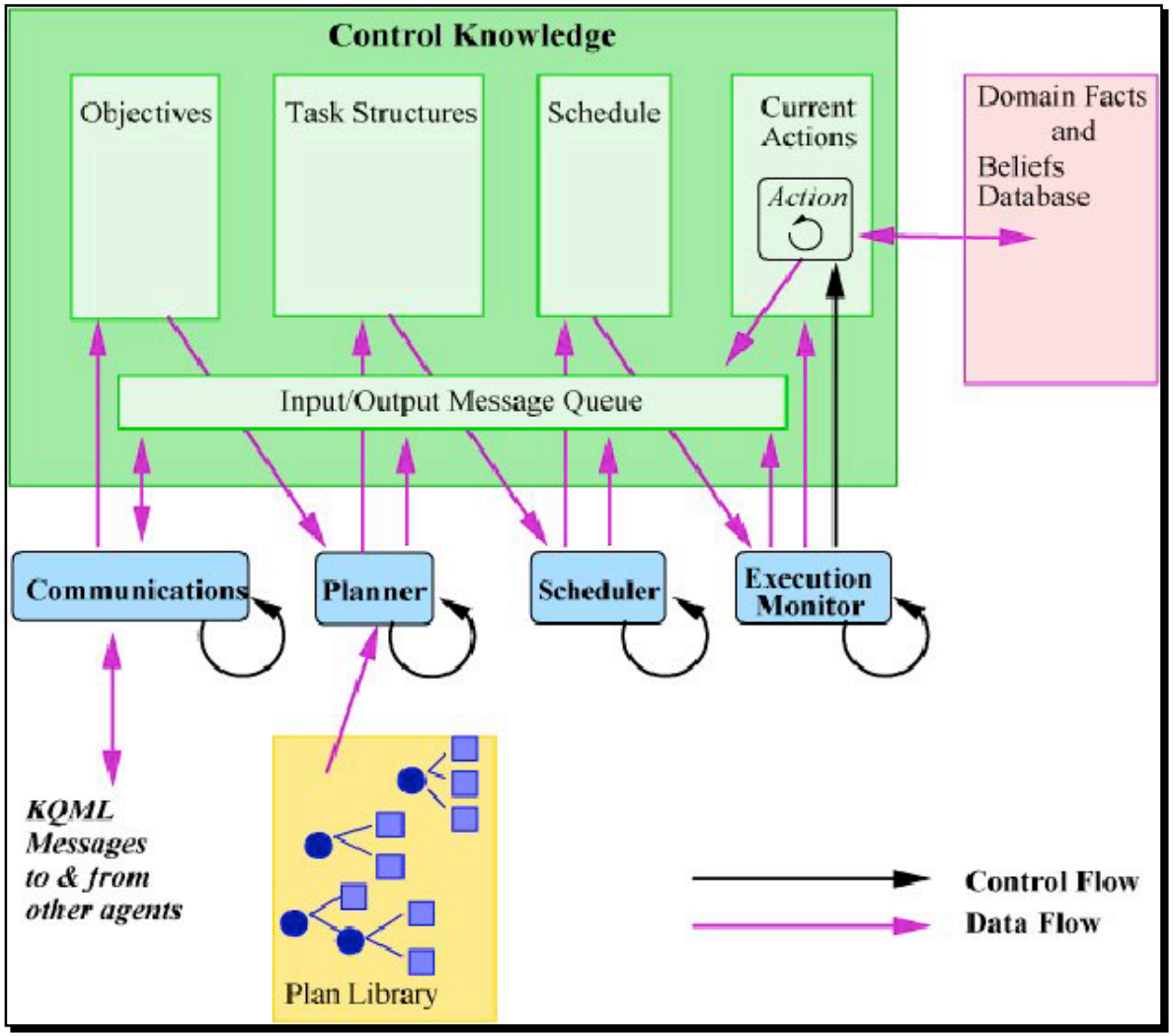


Carnegie Mellon

RETSINA Agent Architecture

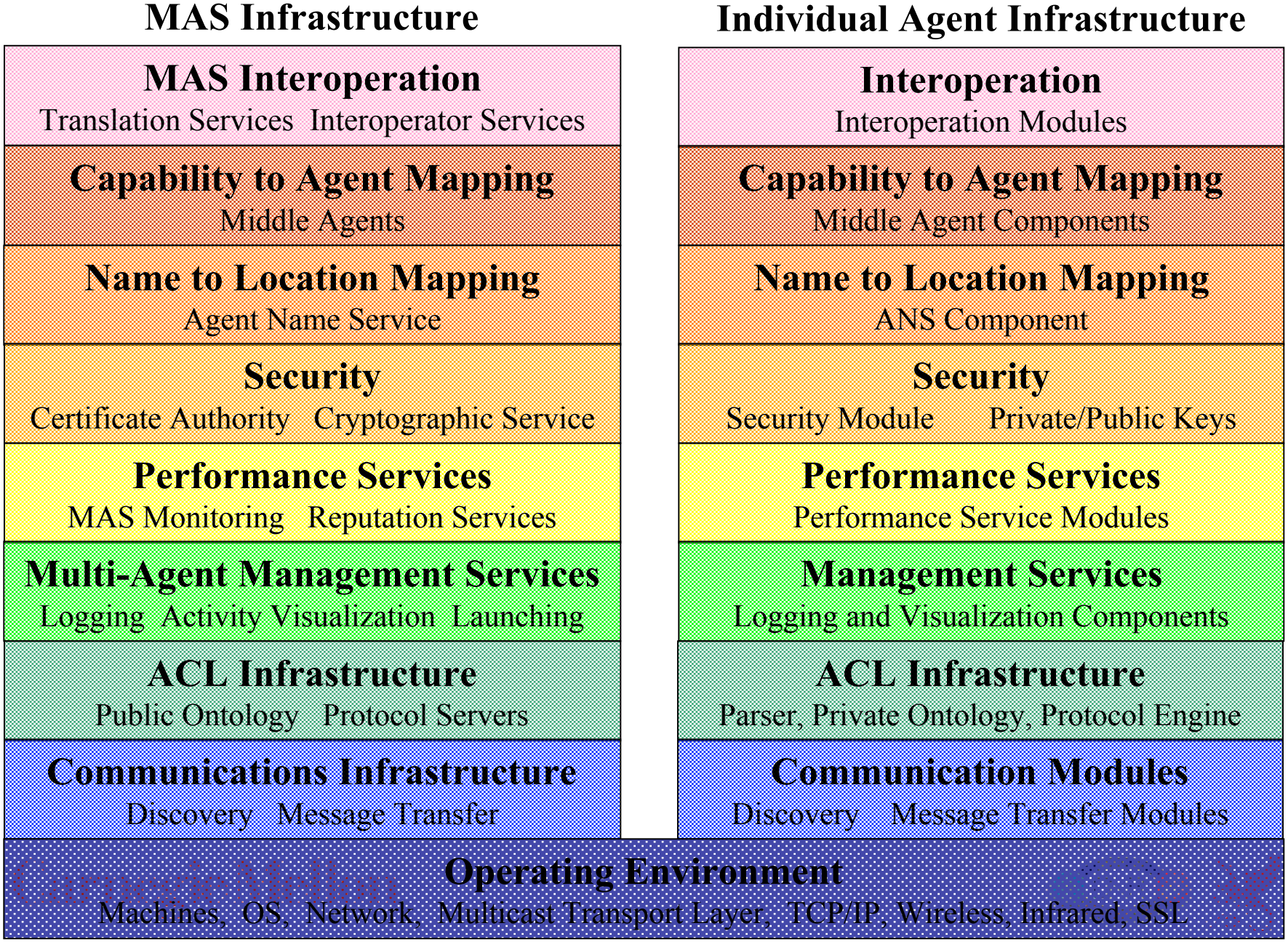
Four parallel threads:

- Communicator
  - for conversing with other agents
- Planner
  - matches "sensory" input and "beliefs" to possible plan actions
- Scheduler
  - schedules "enabled" plans for execution
- Execution Monitor
  - executes scheduled plan
  - swaps-out plans for those with higher priorities



Carnegie Mellon

RETSINA MAS Infrastructure



Outline

- Motivation
- Research Approach
- Taxonomy Findings
- Agent Development Process
- **What's Next**

Carnegie Mellon

What's Next

- Implement proof of concepts
- Monitoring agent that collects parameter settings during problem solving and stores them in a centralized location
- Implement resolution models
- Quantitative analysis of resolution agent use

Carnegie Mellon