Design and Run-Time Quality of Service Management Techniques for Publish/Subscribe Distributed Real-Time and Embedded Systems

http://www.dre.vanderbilt.edu/~jhoffert/dissertation.pdf

Joe Hoffert
jhoffert@dre.vanderbilt.edu

Institute for Software Integrated Systems
Vanderbilt University
Nashville, Tennessee

Context: QoS-enabled Publish/Subscribe for DRE Systems

- Pub/Sub enables separation of concerns - decouples senders & receivers
- QoS enables finer-grained control of system behavior/properties

Client-server technology may not suffice for all DRE systems => move towards publish/subscribe middleware

- *i.e.*, client-server & pub/sub are complementary technologies

Characteristics of Pub/Sub

- Decouples location via anonymous pub/sub
- Decouples time via asynchronous, time-independent data distribution
- Decouples redundancy via unbounded # of senders/receivers

OASIS
WS Brokered Notification

DDS
Data Distribution Service

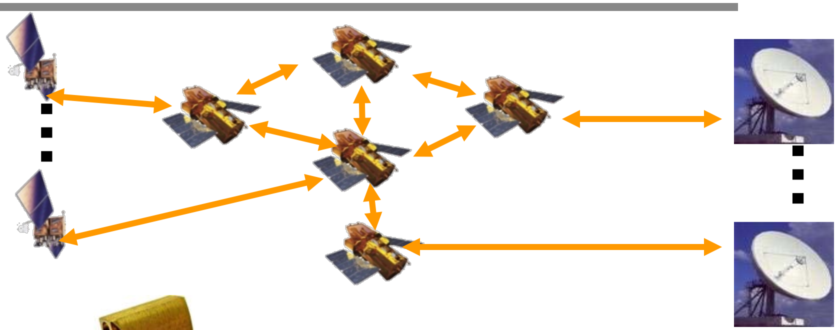
Java Message Service

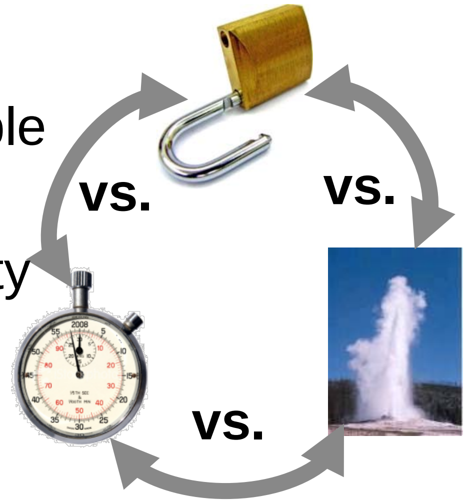
CORBA™
Event & Notification Services

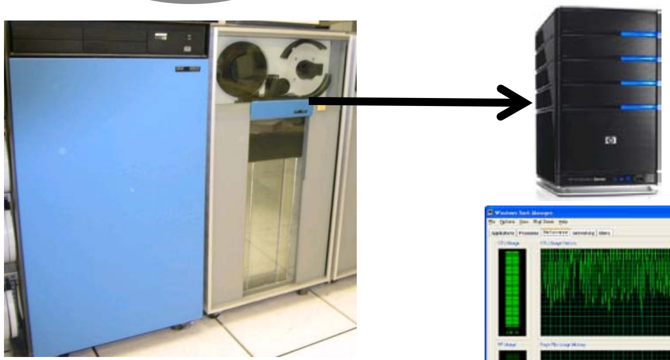
Manifesto for <u>QoS-enabled</u> Pub/Sub
The right data…to the right place…at the right time

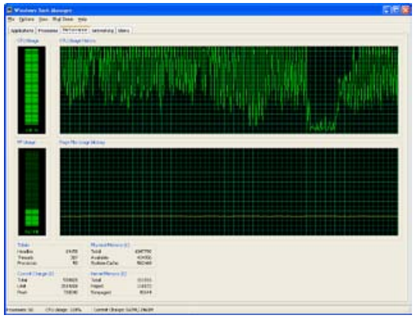
Example: QoS-enabled Pub/Sub DRE Systems

- Net-centric & large-scale "systems of systems"
  - e.g., satellite systems, shipboard computing environments, emergency response systems
- Satisfying tradeoffs between multiple (often conflicting) QoS demands
  - e.g., security, timeliness, reliability
- Regulating & adapting to (dis)continuous changes in runtime environments
  - e.g., online prognostics, dependable upgrades, availability of critical tasks, dynamic resource management







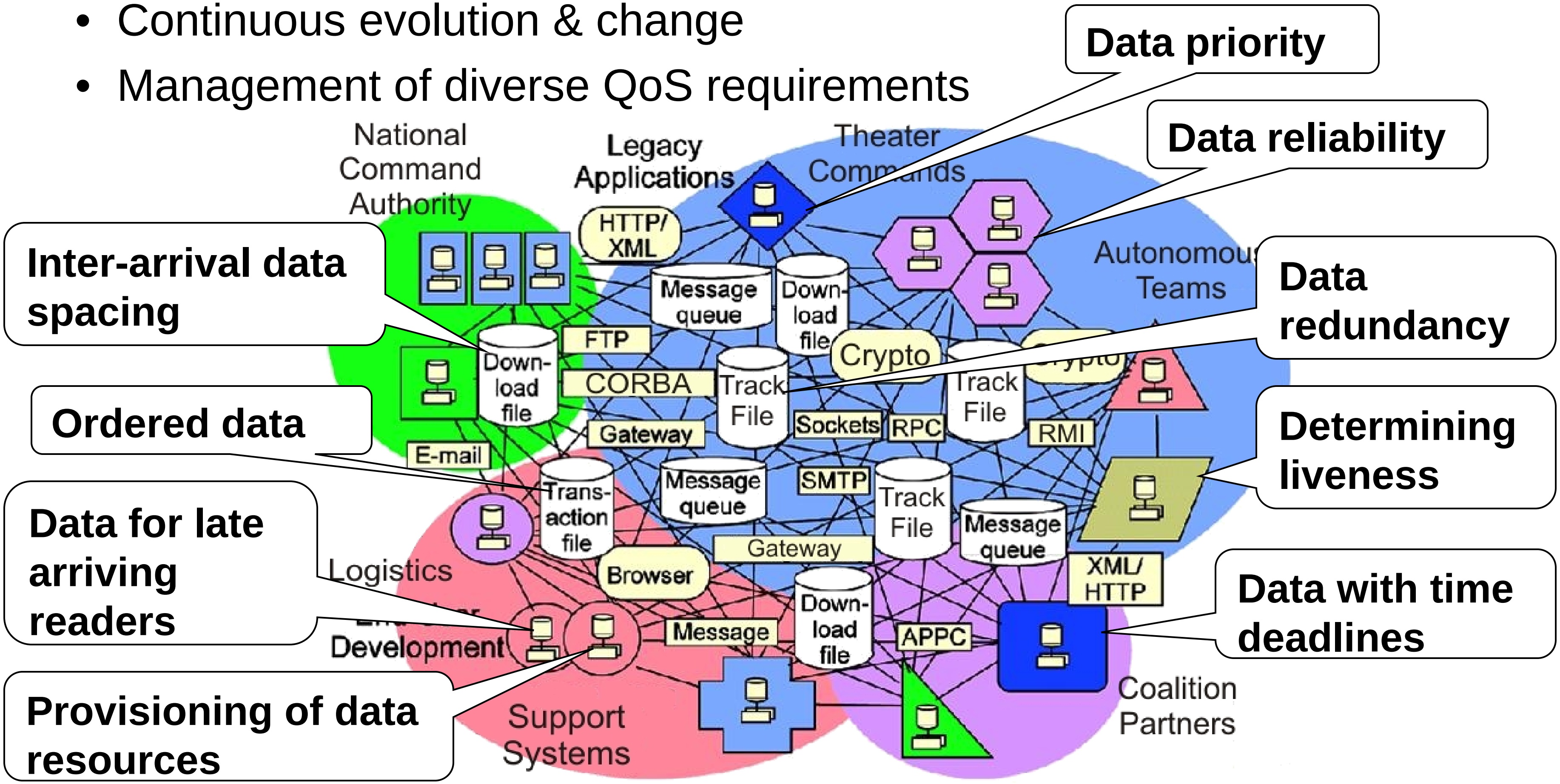


DRE systems increasingly realized by composing loosely-coupled services (*e.g.*, pub/sub)

Challenges in Realizing DRE Pub/Sub Systems

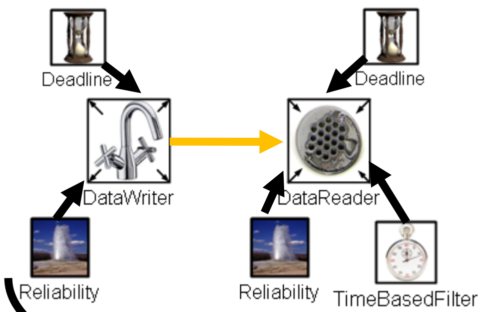
Variability in the ***solution space*** (both design- and run-time)

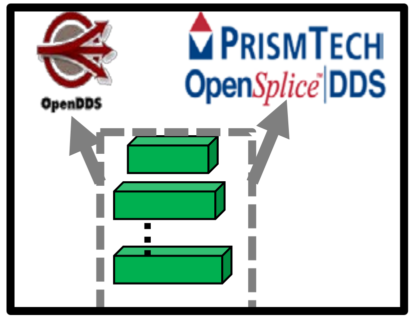
- Diversity in platforms, languages, protocols & tool environments
- Enormous accidental & inherent complexities
- Continuous evolution & change
- Management of diverse QoS requirements

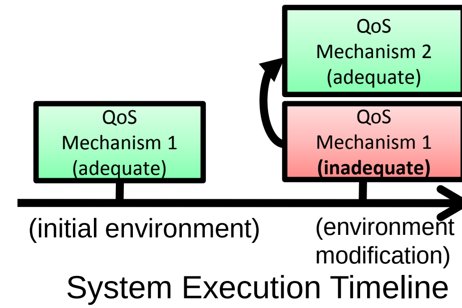


Focus on QoS Management

Overview of QoS Management Focus Areas



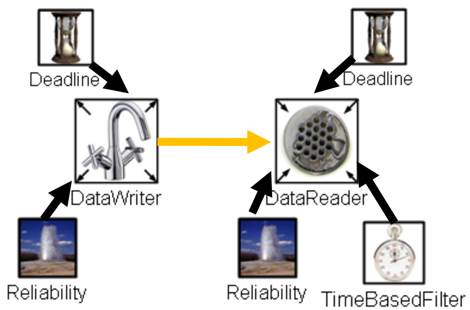


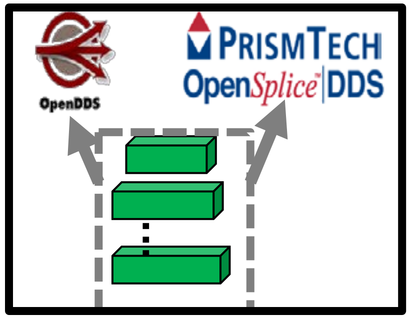


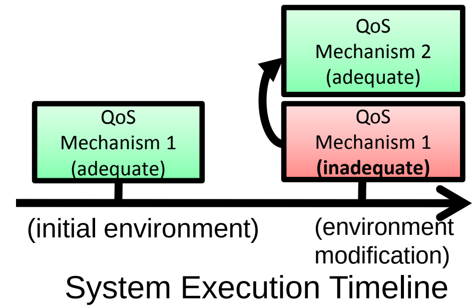
My PhD dissertation addresses 4 aspects of QoS management complexity.

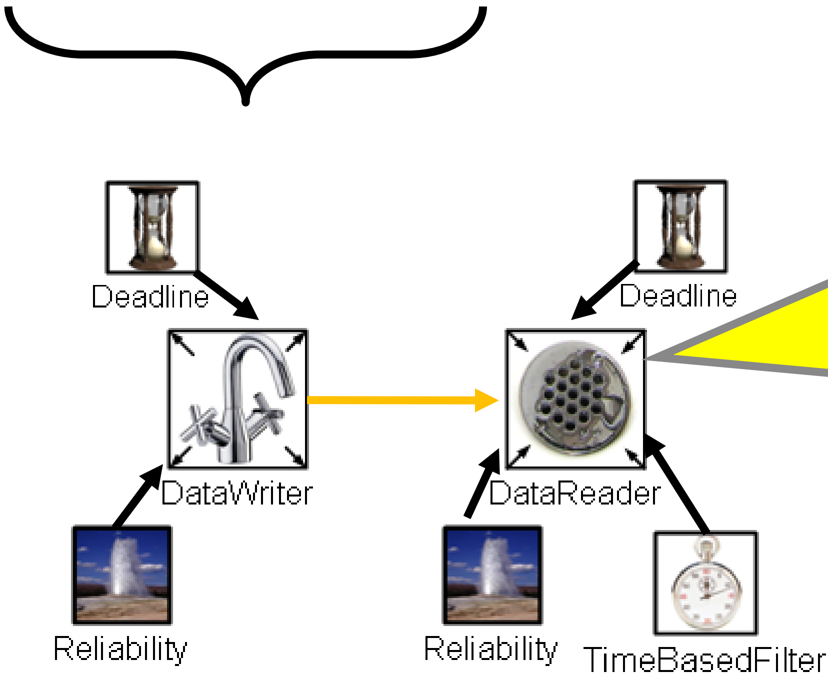
1. **Developed model-based techniques to reduce manual effort & ameliorate accidental complexities in deploying pub/sub DRE systems.**
2. **Designed new composite metrics & a flexible middleware framework to evaluate & benchmark QoS mechanisms.**
3. **Designed machine learning-based adaptation logic to provide accurate configurations & predictable response times in flexible envs.**
4. **Designed monitoring mechanisms & improved machine learning-based logic to improve adaptation accuracy in dynamic envs.**

QoS Management Focus Areas Overview (cont.)





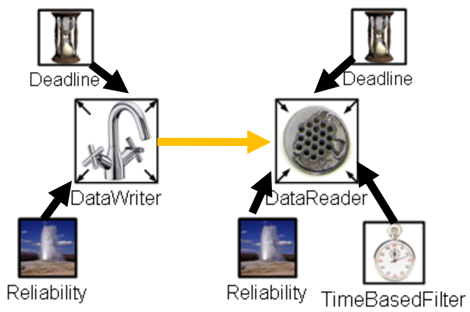


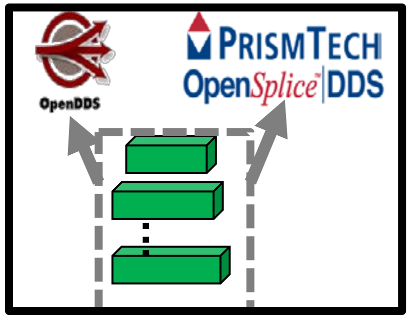


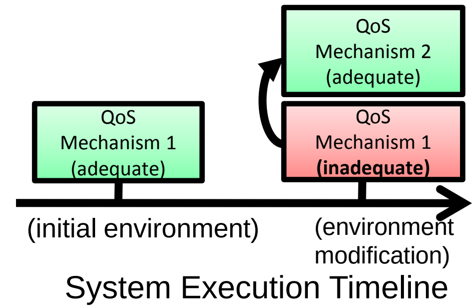
1. QoS Configuration Development Support:

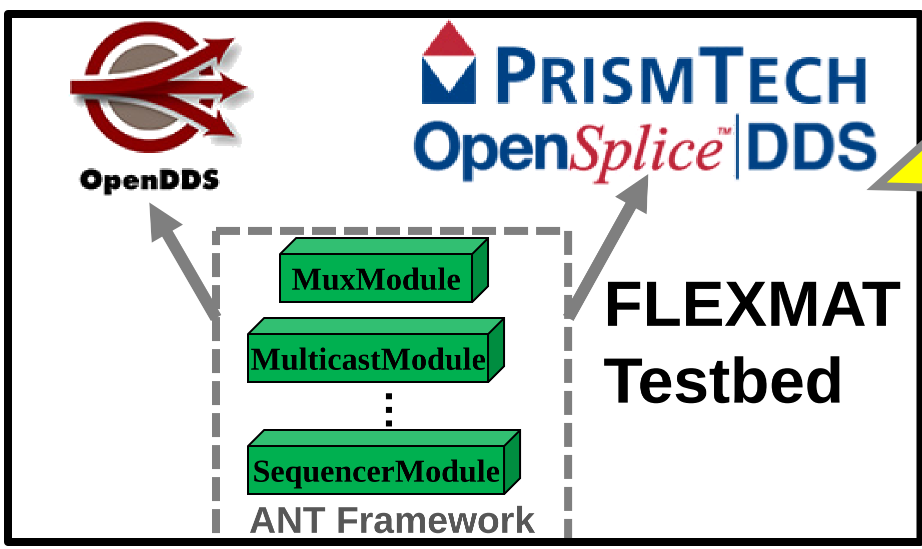
QoS configurations can have numerous entities & QoS policies; how can we help DRE developers manage the complexity of developing configurations?

QoS Management Focus Areas Overview (cont.)





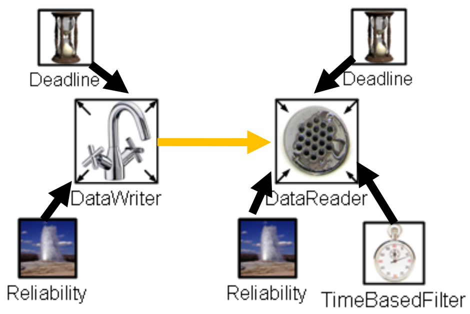


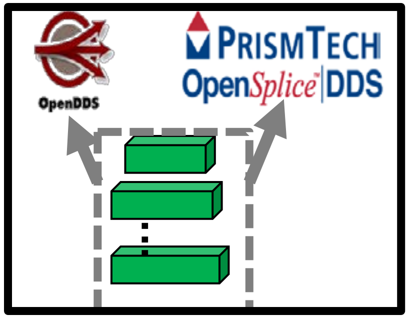


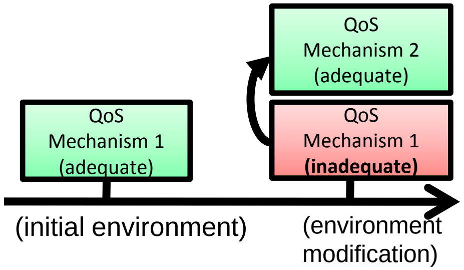
2. Evaluation of QoS mechanisms:

Several QoS mechanisms are available; how can we help developers evaluate QoS mechanisms for pub/sub middleware?

QoS Management Focus Areas Overview (cont.)

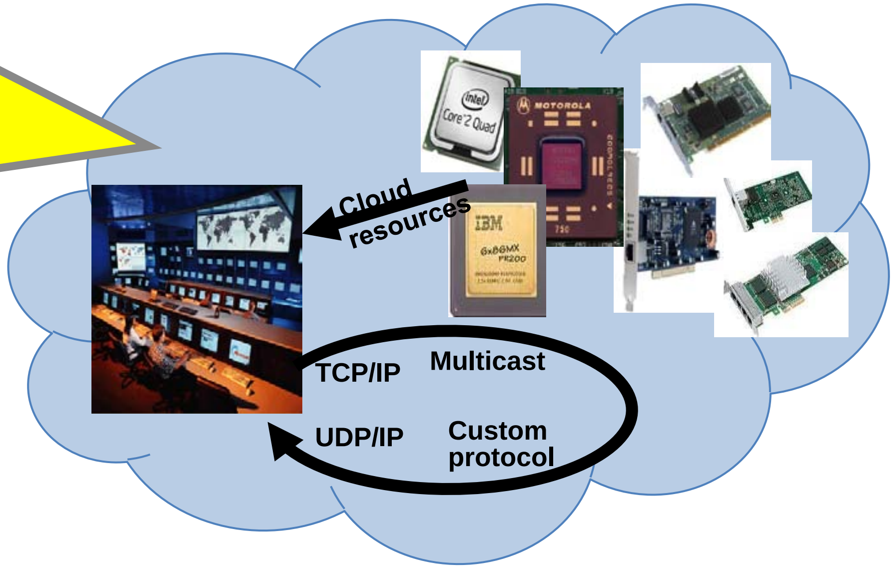




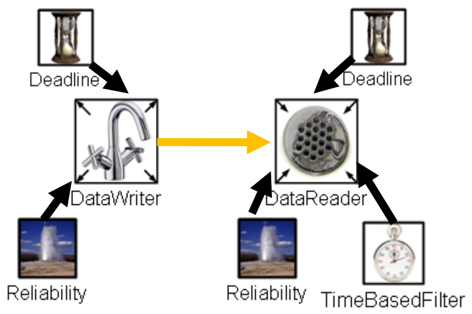


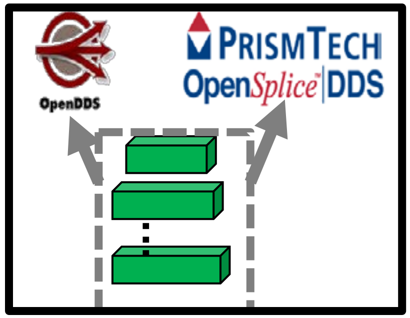
3. QoS Configuration for cloud computing environments:

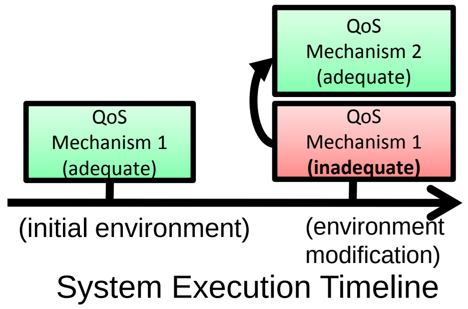
Cloud computing resources which affect QoS aren't known until runtime; how can we configure the middleware based on resources provided?



QoS Management Focus Areas Overview (cont.)

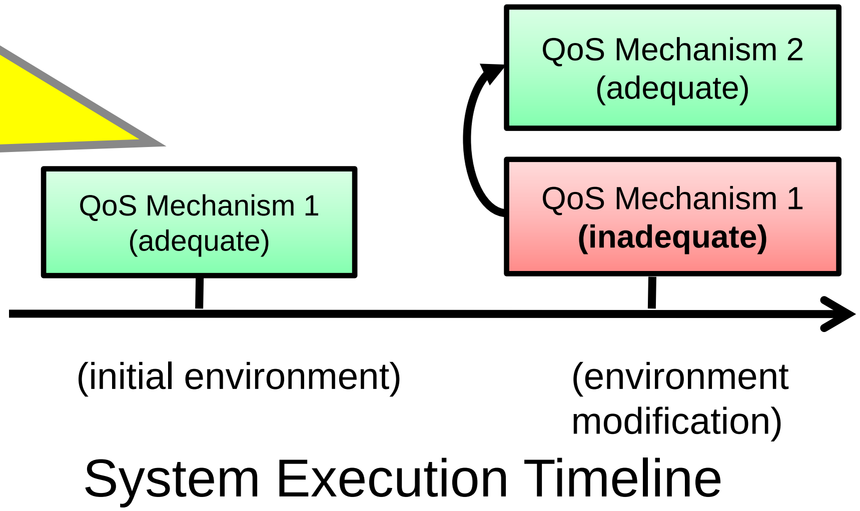




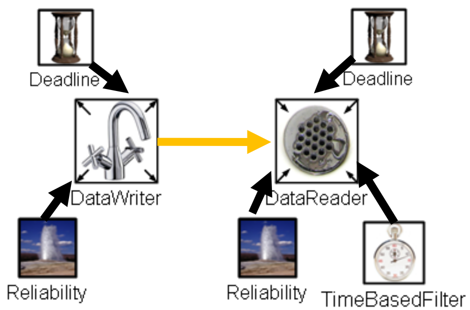


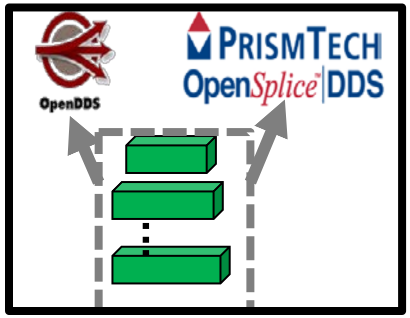
4. QoS adaptation in dynamic environments:

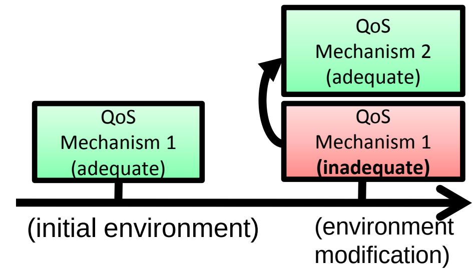
As environments or operating conditions change, QoS can diminish; how can we adapt the middleware to support predictable QoS?



QoS Management Focus Areas Overview (cont.)



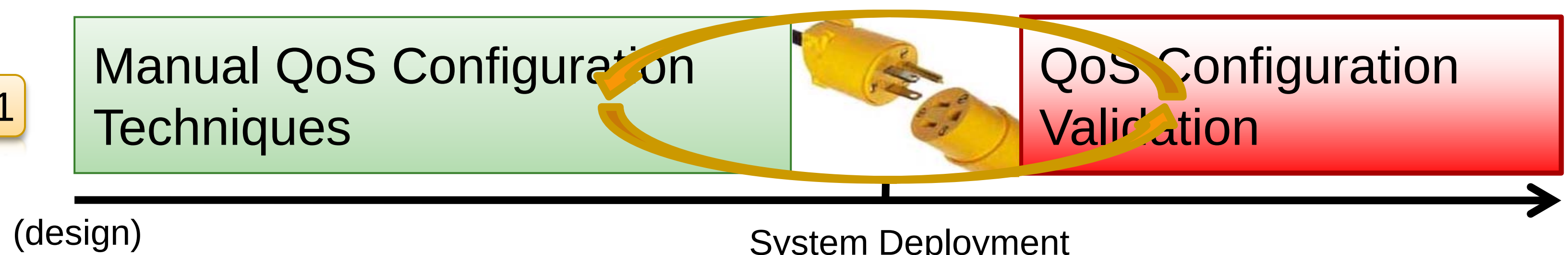




System Execution Timeline

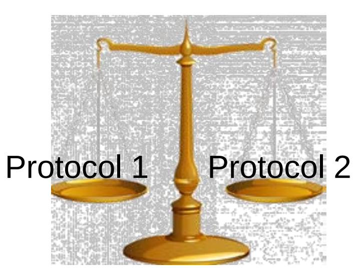
Presented solutions to these in qualifying exam; briefly review here

Focus Area 1



System Lifecycle Timeline

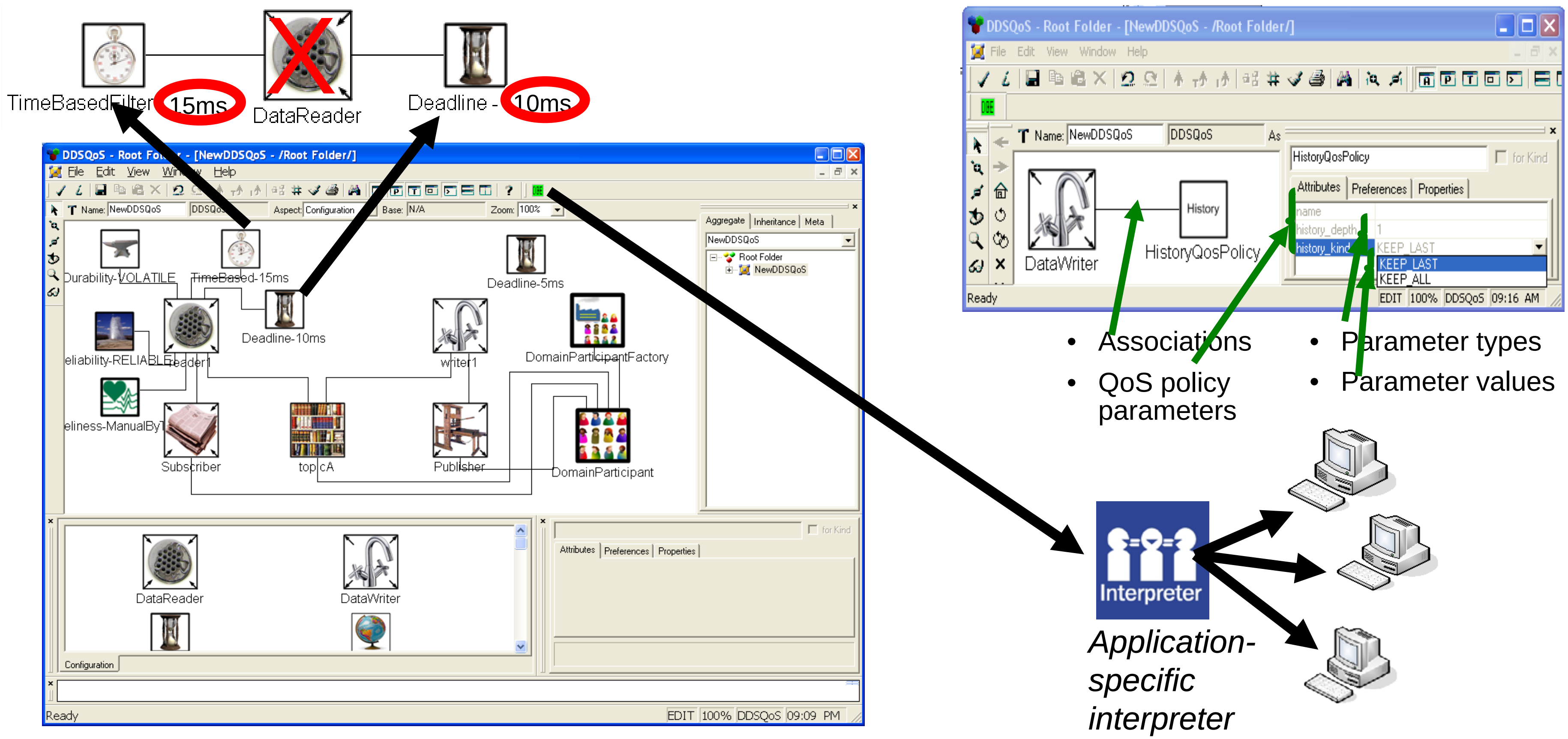
Focus Area 2



Operating environment

Distributed QoS Modeling Language (DQML)

Focus Area 1 → Focus Area 2 → Focus Area 3 → Focus Area 4



DQML addresses the QoS configuration management challenges of

(1) Correctly specified QoS properties,

(2) Correctly managed related & interacting QoS, and

(3) Implementation artifacts that accurately represent design

DQML Related Publications & Presentations

Book Chapter

1. Hoffert, J., Schmidt, D., & Gokhale, A. (2011). Productivity Analysis for the Distributed QoS Modeling Language. *Model-Driven Domain Analysis & Software Development: Architectures & Functions.* Ed. Dr. Janis Osis & Dr. Erika Asnina, Riga Technical University, Latvia.

Conference Publications

2. Hoffert, J., Schmidt, D., & Gokhale, A. (2007, June). A QoS Policy Configuration Modeling Language for Publish/Subscribe Middleware Platforms. *Proceedings of the Inaugural International Conference on Distributed Event-Based Systems (DEBS)*, Toronto, Canada.
3. Hoffert, J., Schmidt, D., & Gokhale, A. (2008, November). DQML: A Modeling Language for Configuring Distributed Publish/Subscribe Quality of Service Policies. *Proceedings of the 10th International Symposium on Distributed Objects, Middleware, & Applications (DOA)*, Monterrey, Mexico.

Poster Publications

4. Hoffert, J., Dabholkar, A., Gokhale, A., & Schmidt, D. (2007, March). Enhancing Security in Ultra-Large Scale (ULS) Systems using Domain-specific Modeling. Spring 2007 Conference for Team for Research in Ubiquitous Secure Technology (TRUST), Berkeley, CA.
5. Hoffert, J., Schmidt, D., Balakrishnan, M., & Birman, K. (2008, April). Trustworthy Conferencing via Domain-specific Modeling & Low Latency Reliable Protocols. Spring 2008 Conference for Team for Research in Ubiquitous Secure Technology (TRUST), Berkeley, CA.
6. Hoffert, J., Gokhale, A. & Schmidt, D. (2007, September). QoS Management in Publish/Subscribe Systems using Domain-specific Modeling. ACM/IEEE 10th International Conference on Model Driven Engineering Languages & Systems (MoDELS), Nashville, TN.

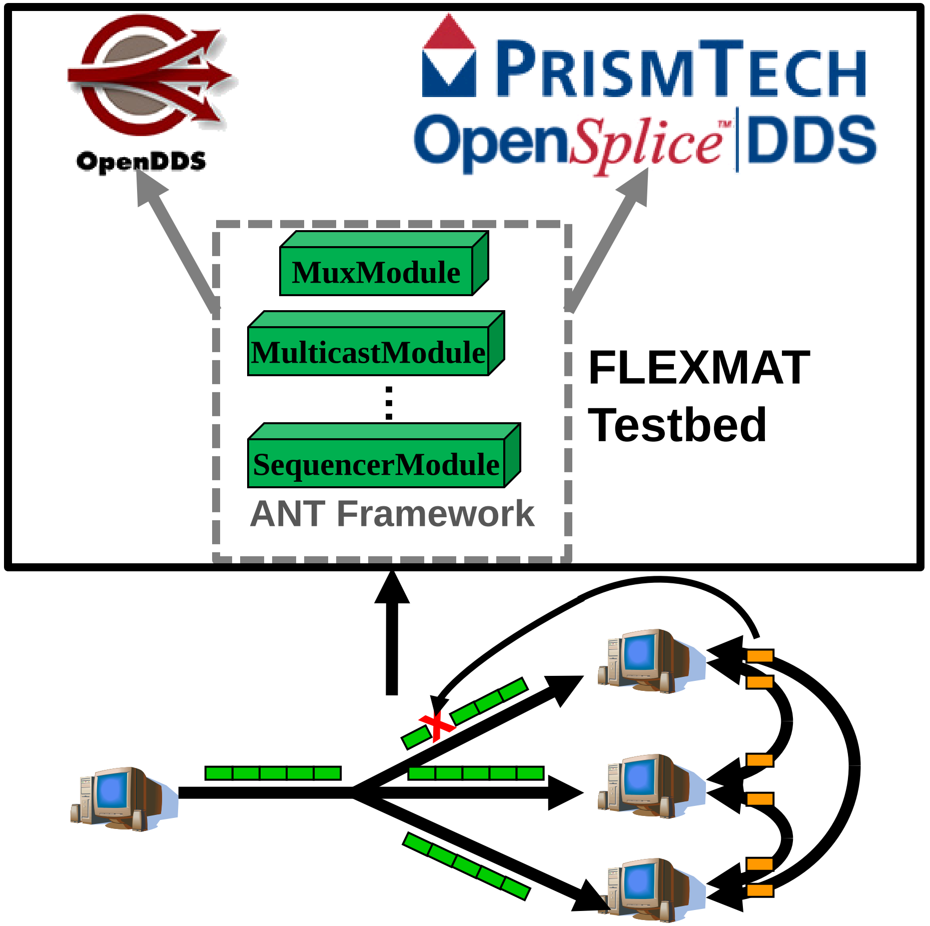
First Author

FLEXible Middleware & Transports (FLEXMAT)

Focus Area 1 → Focus Area 2 → Focus Area 3 → Focus Area 4

Evaluate transport protocols with multiple operating environments using FLEXMAT testbed:

- OpenSplice, OpenDDS
- Various # of senders, % loss, sending rate
- Standard & custom protocols
- Leverage FLEXMAT testbed integrated with DDS implementations
- Leverage composite QoS metrics



FLEXMAT addresses the challenges of

(1) Supporting multiple "antagonistic" QoS via new, custom protocols &

(2) Understanding how environments affect multiple QoS concerns

FLEXMAT Related Publications & Presentations

Conference Publications

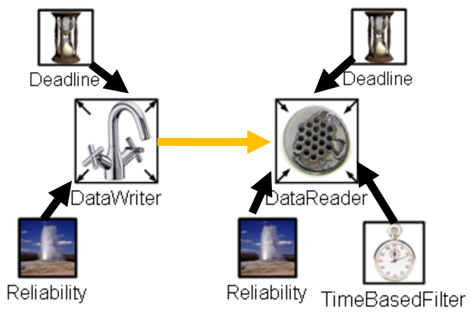
1. Hoffert, J., Schmidt, D., & Gokhale, A. (2009, November). Evaluating Transport Protocols for Real-time Event Stream Processing Middleware & Applications. *Proceedings of the 11th International Symposium on Distributed Objects, Middleware, & Applications (DOA'09)*, Algarve, Portugal.

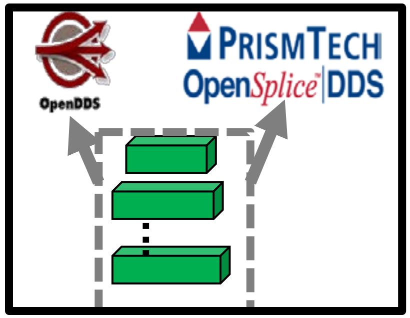
Workshop Publications

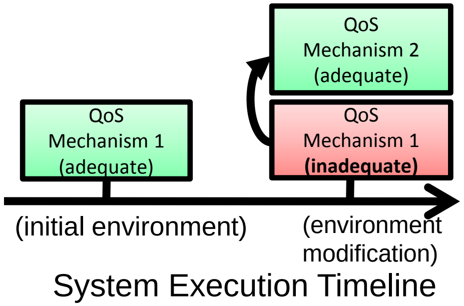
2. Hoffert, J., & Schmidt, D. (2008, July). Supporting Scalability & Adaptability via Adaptive Middleware & Network Transports. *Proceedings of the OMG's Workshop on Distributed Object Computing for Real-time & Embedded Systems*, Washington, D.C., USA.
3. Hoffert, J., Schmidt, D., Balakrishnan, M., & Birman, K. (2008, September). Supporting Large-scale Continuous Stream Datacenters via Pub/Sub Middleware & Adaptive Transport Protocols. *Proceedings of the 2nd Workshop on Large-Scale Distributed Systems & Middleware*, Yorktown, NY.
4. Balakrishnan, M., Hoffert, J., Birman, K., & Schmidt, D., (2008, September). Rethinking Reliable Transport for the Datacenter. *Proceedings of the 2nd Workshop on Large-Scale Distributed Systems & Middleware*, Yorktown, NY.
5. Hoffert, J., & Schmidt, D. (2009, July). FLEXible Middleware & Transports (FLEXMAT) for Real-time Event Stream Processing (RT-ESP) Applications. *Proceedings of the OMG's Workshop on Distributed Object Computing for Real-time & Embedded Systems*, Washington, D.C., USA.

First Author	Second Author

QoS Management Focus Areas Overview (cont.)

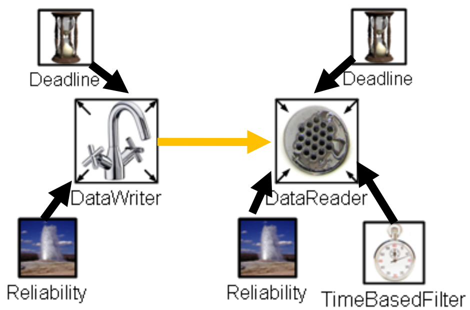


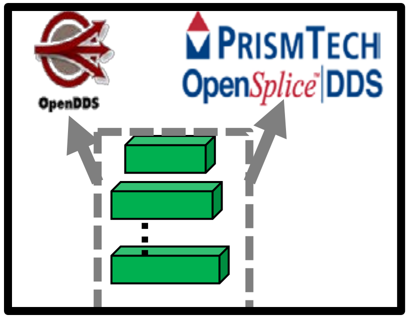


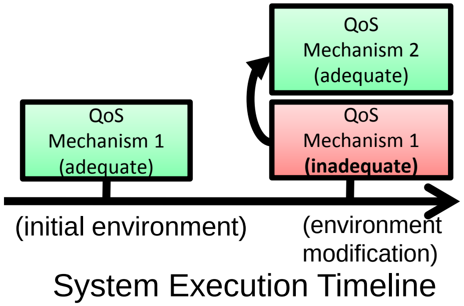


This was the focus area in my qualifying exam that I proposed to complete my dissertation.

QoS Management Focus Areas Overview (cont.)







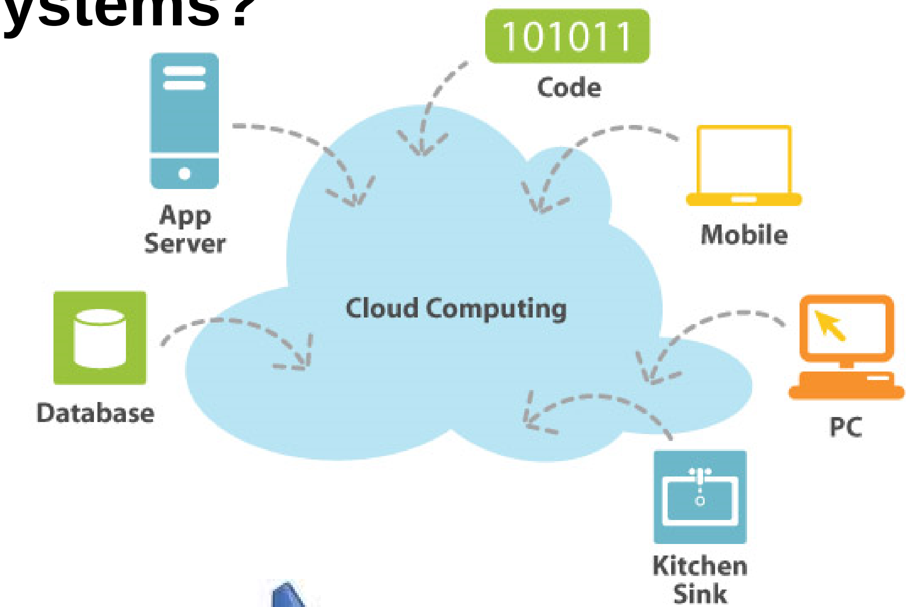
This is a new focus area that I included while completing my dissertation.

Focus Area 3: Configuring DRE Systems in Flexible Envs.

Focus Area 1 → Focus Area 2 → Focus Area 3 → Focus Area 4

Cloud Computing for DRE systems?

- Resources provided as service
  - Resources on demand
  - “Pay-as-you-go” usage fee
  - Computing resources
    - CPUs, RAM
  - Networking resources
    - Bandwidth, network latency



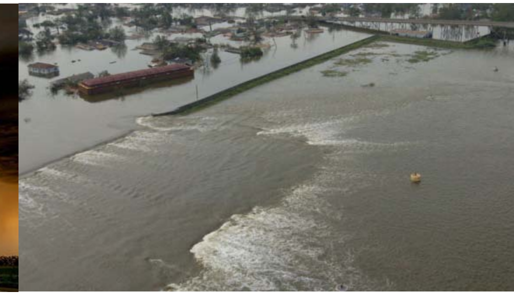
- Popular implementations
  - Amazon Elastic Compute Cloud (EC2), Google App Engine, GoGrid, AppNexus, Emulab
  - OS, Database, RAM, CPU, Disk space, cores, load balancing, applications (e.g., Apache, Facebook servers), bandwidth

Not straightforward to use Cloud in DRE systems

Motivating Example: Search & Rescue Missions (1/2)

DRE Cloud Scenario

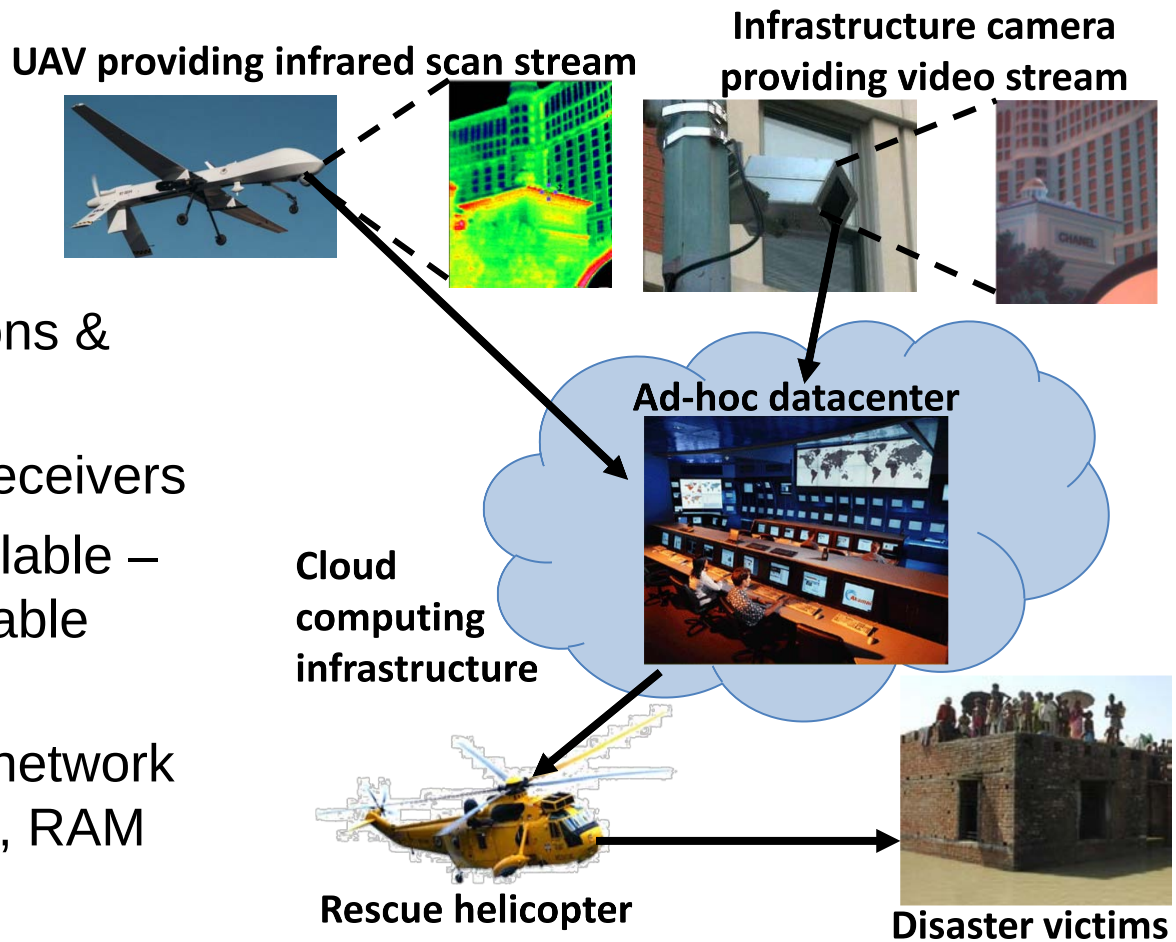
- Regional disasters (e.g., hurricane, flooding)
- Survivors trapped
- Search & rescue mission initiated
- Search application fuses multiple sensor streams
  - Thermal scans from unmanned aerial vehicles (UAVs)
  - Video from existing camera infrastructure
  - Data streams sent to **ad-hoc datacenter** for fusion & dissemination

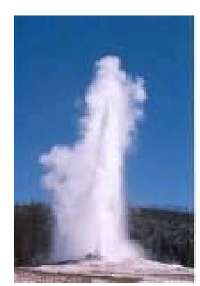


Motivating Example: Search & Rescue Missions (2/2)

Datacenter Requirements

- Operate in flexible environments, e.g.,
  - Support multiple missions & applications
  - Varying # of senders, receivers
  - Local resources unavailable – adapt to leverage available resources
  - Cloud resources, *e.g.*, network bandwidth, CPU speed, RAM
- Support Multiple QoS
  - Reliability & latency
    - e.g., video & streamed thermal scans
    - Multimedia data





Challenges for Datacenter in Cloud Environments (1/3)

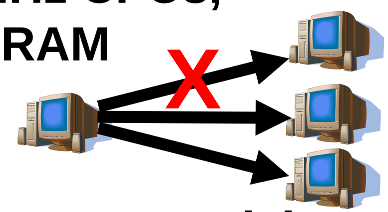
Challenge 1: Reduction of Development Complexity

Developing adaptive behavior is challenging:

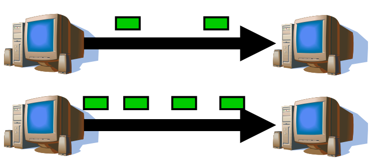
- Inherent complexity – designing appropriate responses for environment
- Accidental complexity – transforming & managing appropriate responses from design to implementation

850 MHz CPUs, 2 GB RAM

100 Mb/s LAN



network loss



data sending rate

```
if (network_loss_percent == 1 && num_receivers < 5
    && sending_rate <= 25Hz && CPU_speed == 3GHz
    && RAM == 2GB && net_bw == 1Gb
    && DDS_impl == OpenSplice
    && metric == reliability_and_avg_latency) {
  transport_framework->use (transport1);
} else if (network_loss_percent == 3
           && num_receivers >= 5 && num_receivers < 10
           && sending_rate > 50Hz && CPU_speed == 850MHz
           && RAM == 500MB && net_bw == 100Mb
           && DDS_impl == OpenDDS
           && metric == reliability_and_jitter) {
  transport_framework->use (transport2);
} else if ...
```

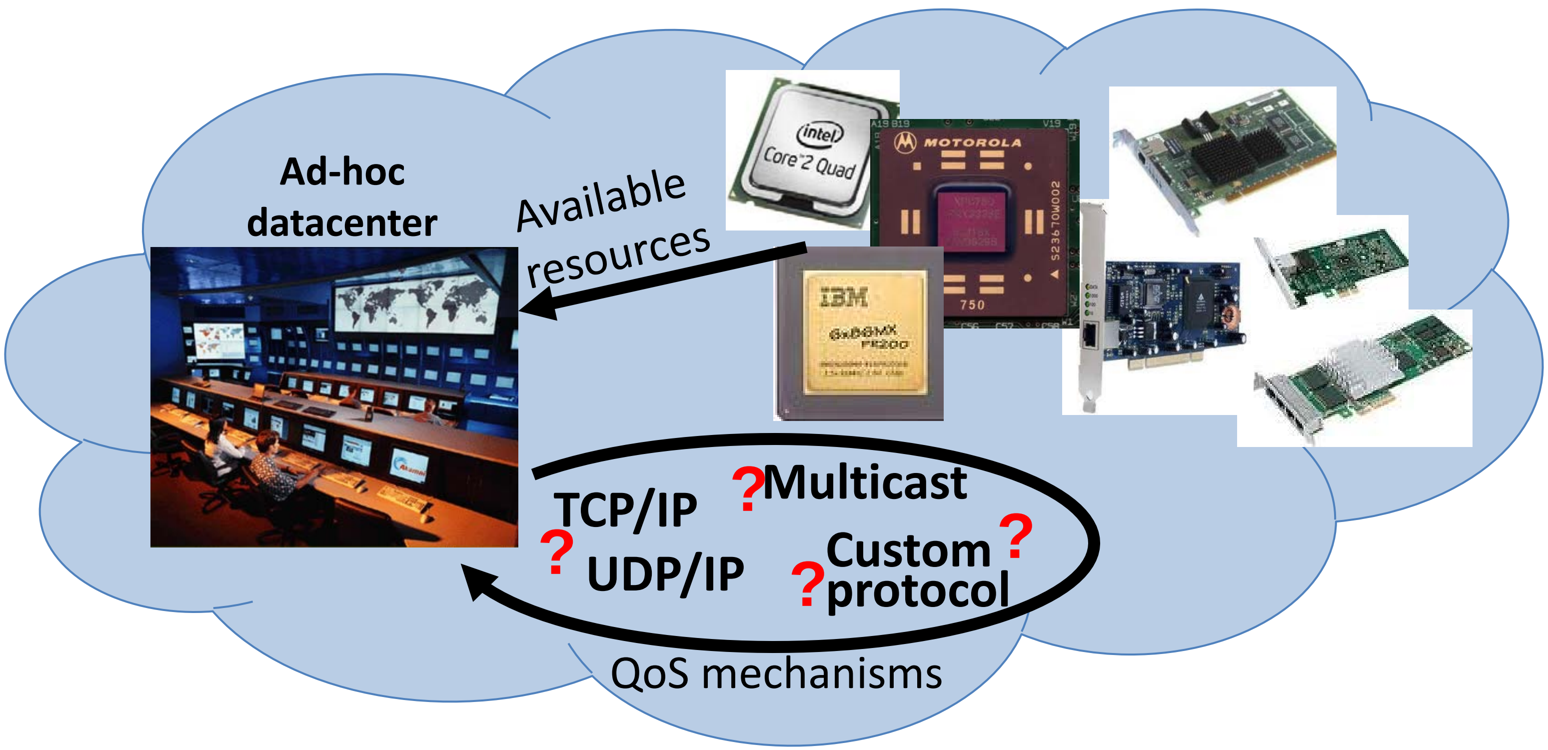
Increased development complexity reduces availability, assurance, and portability

Challenges for Datacenter in Cloud Environments (2/3)

Challenge 2: Accurate Configuration in Cloud Environments

Environment resources unknown *a priori* make static configuration inadequate

Cloud computing infrastructure



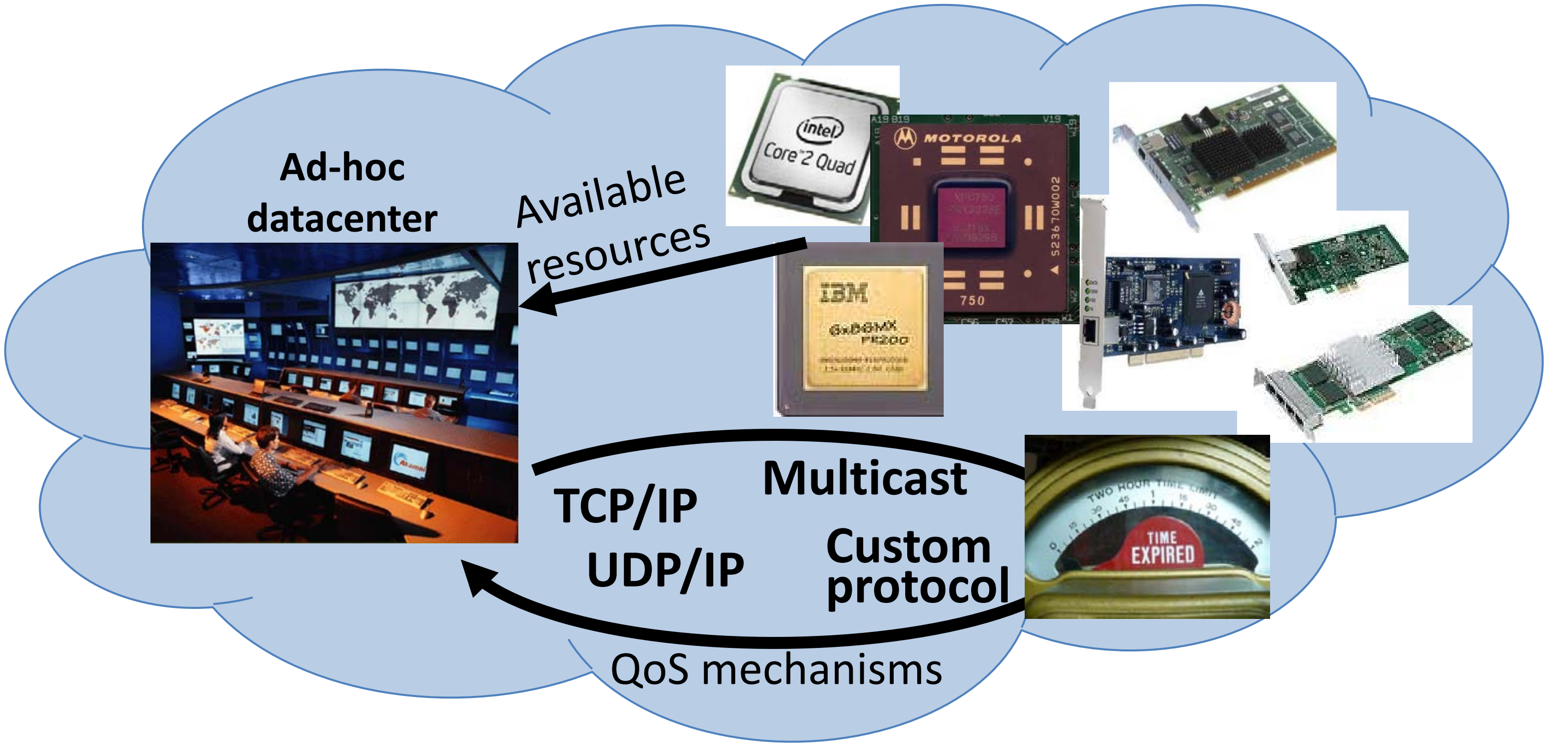
Inaccurate configuration can result in loss of life & property

Challenges for Datacenter in Cloud Environments (3/3)

Challenge 3: Timely Configuration in Cloud Environments

DRE systems require timely configuration

Cloud computing infrastructure



Untimely configuration can result in loss of life & property

QoS in Cloud Environments: Related Research

Related Research
Y. Eustache & J.-P. Diguet. “Reconfiguration Management in the Context of RTOS-Based HW/SW Embedded Systems”, *EURASIP Journal on Embedded Systems*, pages 1 – 10. Hindawi Publishing Corp., 2008.
A. Zoitl *et al*. "A real-time reconfiguration infrastructure for distributed embedded control systems," *Proceedings of the 2010 IEEE Conference on Emerging Technologies and Factory Automation (ETFA),* September 2010, Bilboa, Spain.
P.-C. David & T. Ledoux. “An Aspect-Oriented Appro[illegible]h for Developing Self-Adaptive Fractal Components”, *Software Composition*, pages 82–97. [illegible]ger LNCS, 2006.
P. Grace *et al*. “Deep Middleware for the Divergent Grid”, [illegible]*dings of the ACM/IFIP/USENIX 2005 International Conference on* [illegible] Grenoble, France.
G. Valetto *et al.* “Towards Service Awareness and Auto[illegible] Network”, *Autonomic Communication*, pp. 202–213. Springer-Verlag, 2006.
J. Imtiaz *et al*. “A Novel Method for Auto Configuration of Realtime Ethernet Networks”, *Proceedings of the IEEE International Conference on Emerging Technologies and Factory Automation*, September 2008*,* Hamburg, Germany.
Xiangping Bu *et al*. "A Reinforcement Learning Approach to Online Web Systems Auto-configuration“, *29th IEEE International Conference on Distributed Computing Systems,* June 2009, Montreal, Canada.

Good for configuring
local components

QoS in Cloud Environments: Related Research

Related Research
Y. Eustache & J.-P. Diguet. “Reconfiguration Management in the Context of RTOS-Based HW/SW Embedded Systems”, *EURASIP Journal on Embedded Systems*, pages 1 – 10. Hindawi Publishing Corp., 2008.
A. Zoitl *et al*. "A real-time reconfiguration infrastructure for distributed embedded control systems," *Proceedings of the 2010 IEEE Conference on Emerging Technologies and Factory Automation (ETFA),* September 2010, Bilboa, Spain.
P.-C. David & T. Ledoux. “An Aspect-Oriented Approach for Developing Self-Adaptive Fractal Components”, *Software Composition*, pages 82–97. Springer LNCS, 2006.
P. Grace *et al*. “Deep Middleware for the Divergent Grid”, *Proceedings of the ACM/IFIP/USENIX 2005 International Conference on Middleware*, November 2005, Grenoble, France.
G. Valetto *et al.* “Towards Service Awareness and Autonomic Features in a SIP-Enabled Network”, *Autonomic Communication*, pp. 202–213. Springer-Verlag, 2006.
J. Imtiaz *et al*. “A Novel Method for Auto Configurati f Realtime Ethernet Networks”, *Proceedings of the IEEE International Conference or ing Technologies and Factory Automation*, September 2008, Hamburg, Germa
Xiangping Bu *et al*. "A Reinforcement Learning A uto-configuration“, *29th IEEE International Confere ems,* June 2009, Montreal, Canada.

Good for ***developing*** autoconfiguration applications

QoS in Cloud Environments: Related Research

Related Research
Y. Eustache & J.-P. Diguet. “Reconfiguration Management in the Context of RTOS-Based HW/SW Embedded Systems”, *EURASIP Journal on Embedded Systems*, pages 1 – 10. Hindawi Publishing Corp., 2008.
A. Zoitl *et al*. "A real-time reconfiguration infrastructure for distributed embedded control systems," *Proceedings of the 2010 IEEE Conference on Emerging Technologies and Factory Automation (ETFA),* September 2010, Bilboa, Spain.
P.-C. David & T. Ledoux. “An A [illegible] ptive Fractal Components”, *Software Com* [illegible]
P. Grace *et al*. “Deep Middlewa [illegible] *ACM/IFIP/USENIX 2005 Internationa* [illegible] *ce on Middleware*, November 2005, Grenoble, France.
G. Valetto *et al.* “Towards Serv [illegible] wareness and Autonomic Features in a SIP-Enabled Network”, *Autonomic Com* [illegible] *unication*, pp. 202–213. Springer-Verlag, 2006.
J. Imtiaz *et al*. “A Novel Method for Auto Configuration of Realtime Ethernet Networks”, *Proceedings of the IEEE International Conference on Emerging Technologies and Factory Automation*, September 2008, Hamburg, Germany.
Xiangping Bu *et al*. "A Reinforcement Learning Approach to Online Web Systems Auto-configuration“, *29th IEEE International Conference on Distributed Computing Systems,* June 2009, Montreal, Canada.

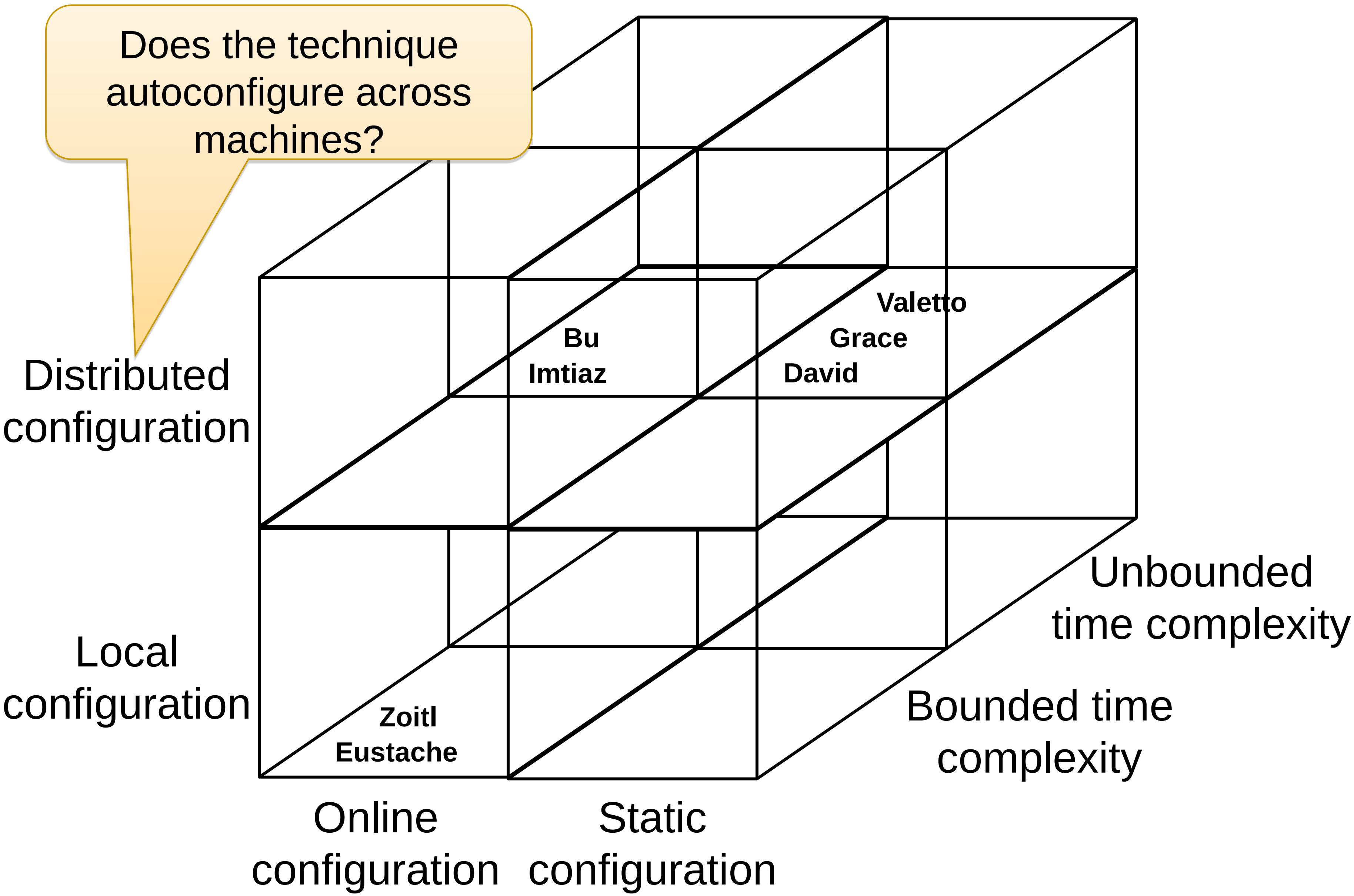
Good for autoconfiguration when ***timeliness is not a driving concern***

Properties for QoS Support in Cloud Environments

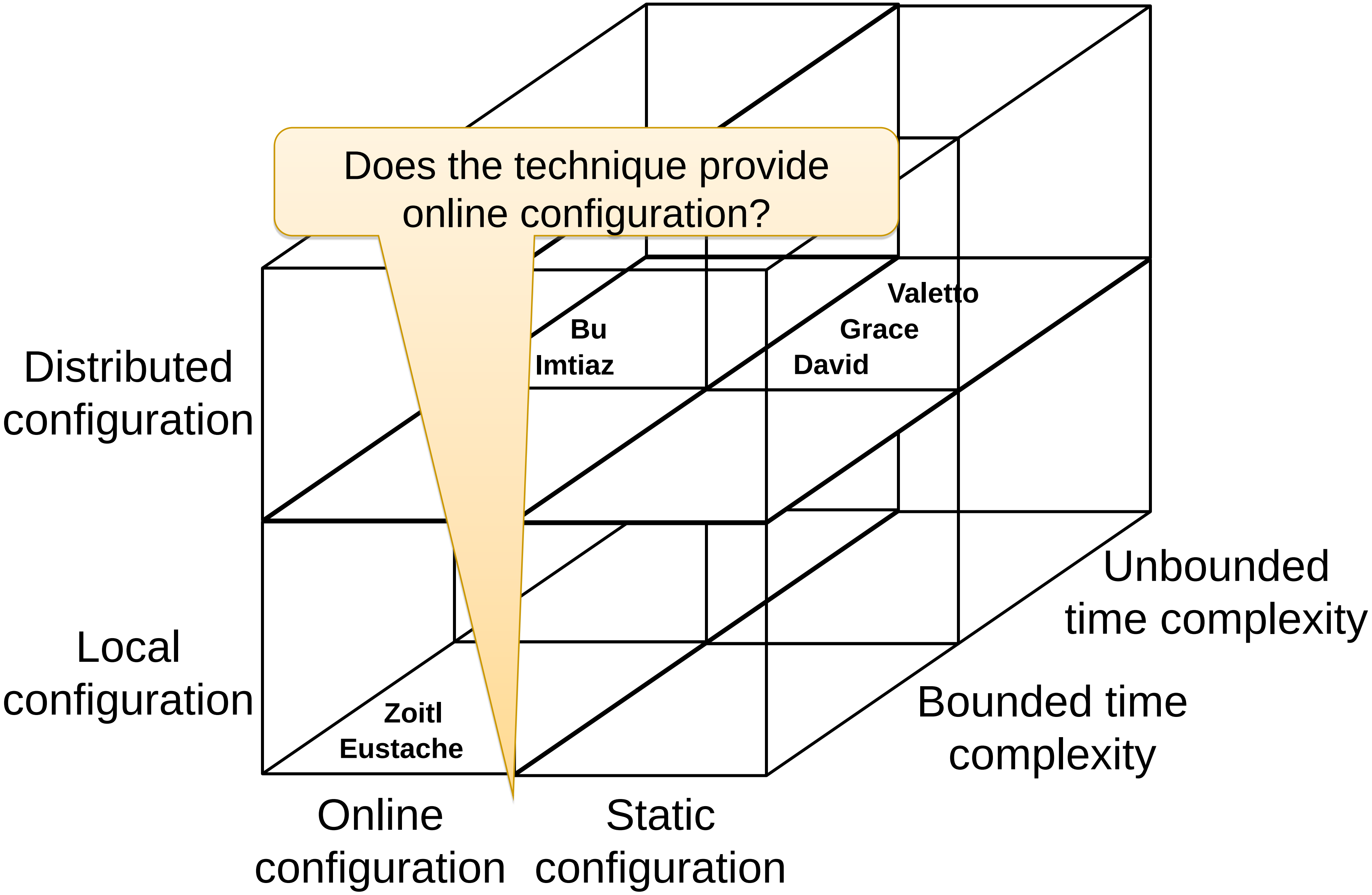
What properties help us assess research to configure QoS in cloud environments?

Property	Description
Distributed Configuration	Does the technique help autoconfiguration of QoS across machine boundaries?
Online Configuration	Does the technique perform configuration adjustments while system is running?
Timely Configuration	Does the technique provide bounded-time – ideally, constant-time – response?

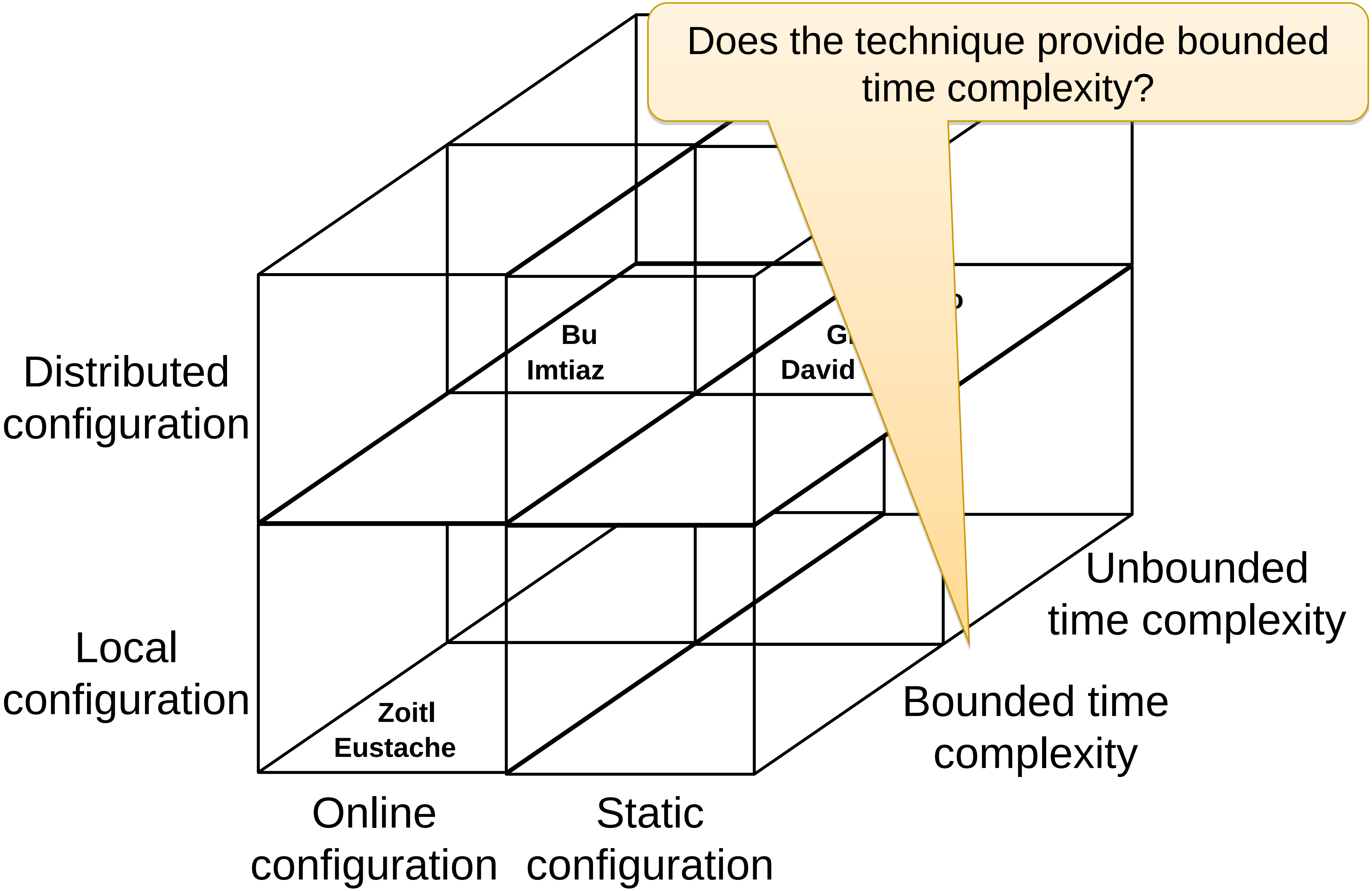
QoS in Cloud Environments: Related Work



QoS in Cloud Environments: Related Work

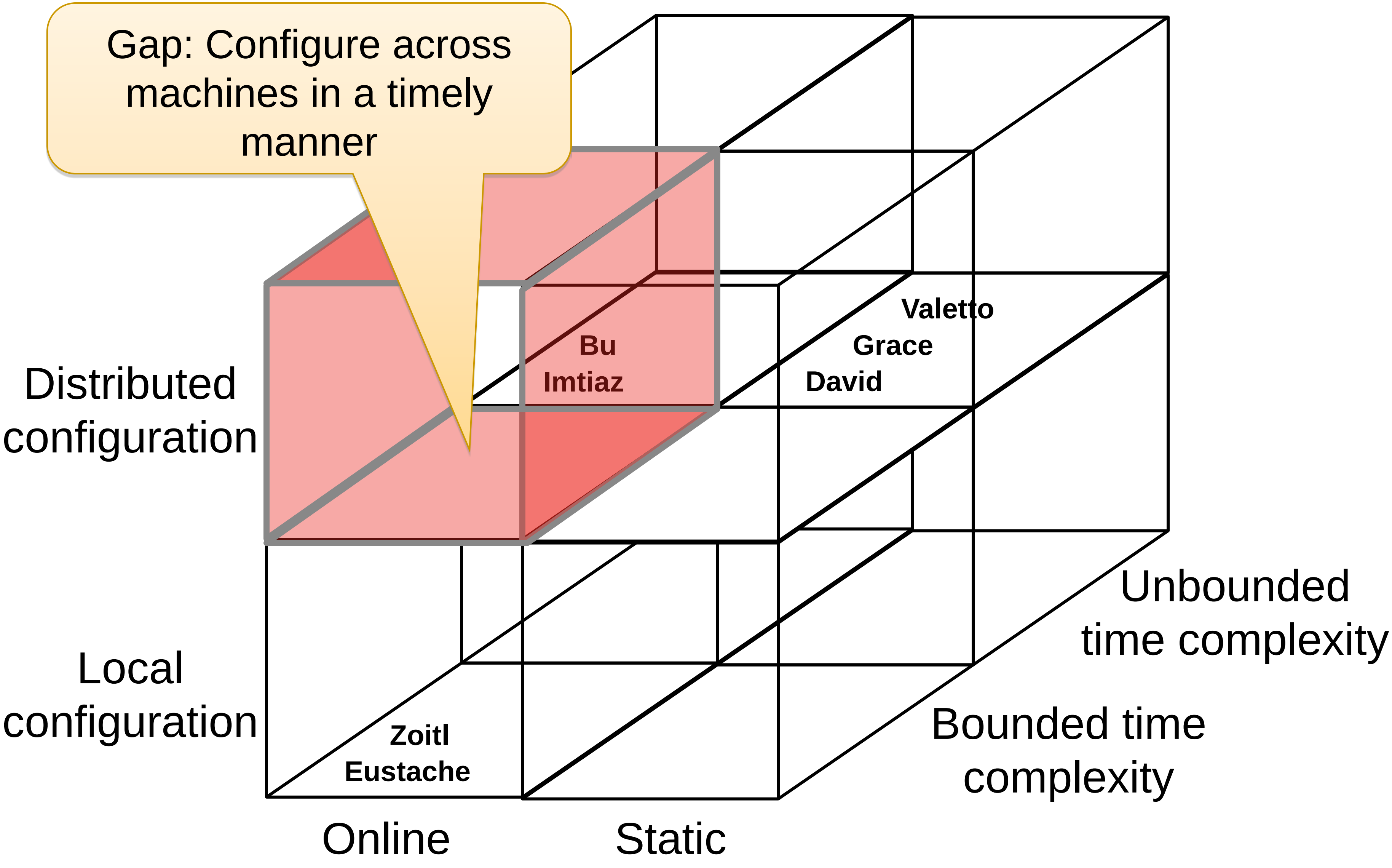


QoS in Cloud Environments: Related Work



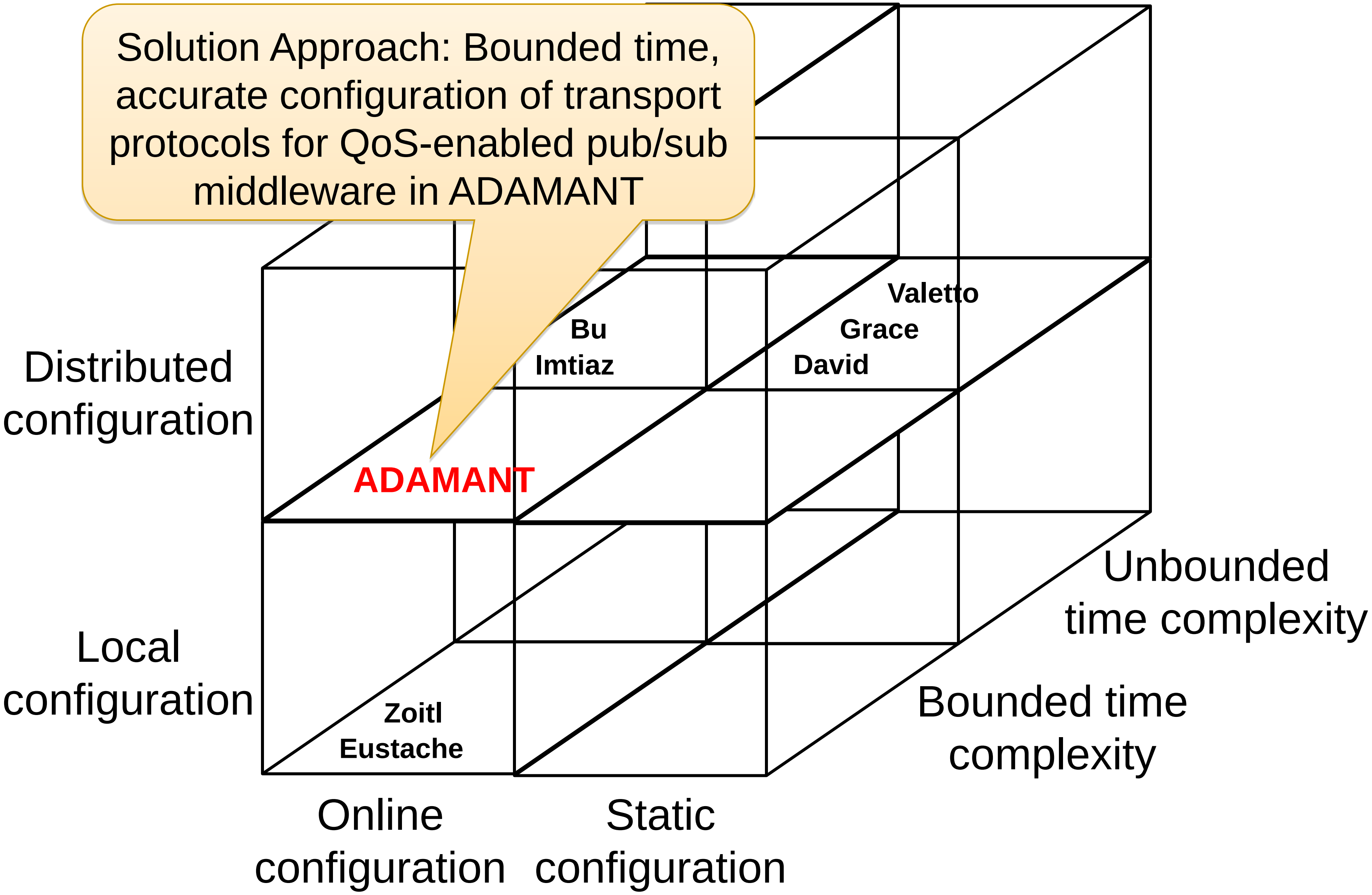
QoS in Cloud Environments: Related Work

Gap: Configure across machines in a timely manner



Current gap makes it *hard* for DRE systems in cloud environments to configure QoS in a timely manner

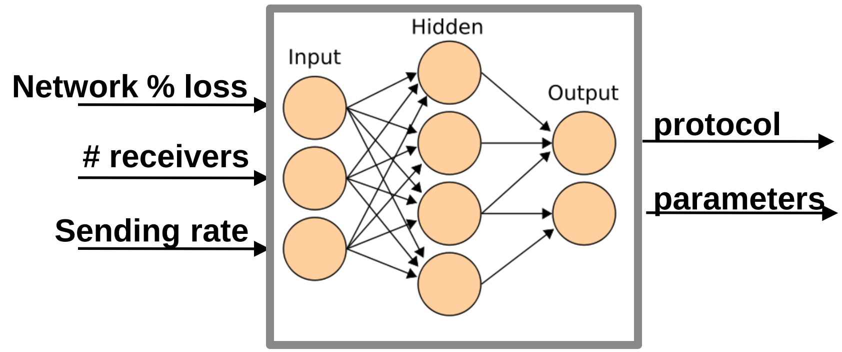
Solution Approach: ADAptive M/W And Network Transports



Some Configuration Approaches Considered

```
if (network_loss_percent == 1
    && num_receivers < 5
    && sending_rate < 0.01) {
  transport_framework->use (transport1);
} else if (network_loss_percent == 5
           && num_receivers < 5
           && sending_rate < 0.01) {
  transport_framework->use (transport2);
} else if (network_loss_percent == 10
           && num_receivers < 5
           && sending_rate < 0.01) {
  transport_framework->use (transport3);
} else if (network_loss_percent == 1
```

Policy-based configuration

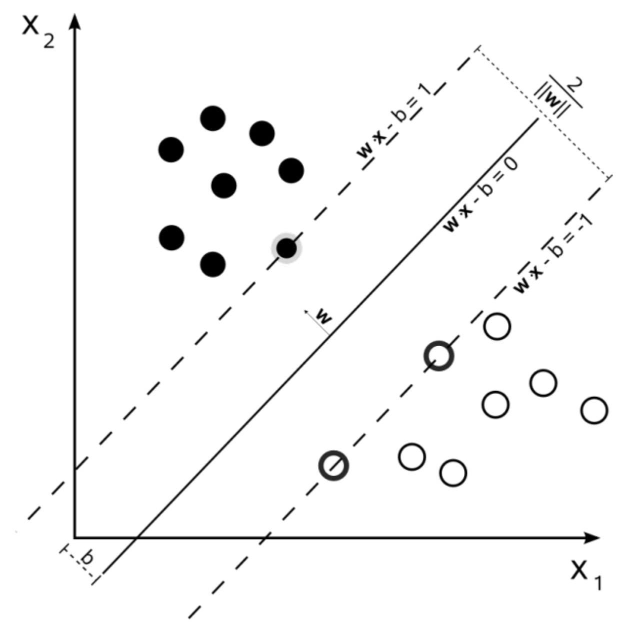


Artificial neural network

```
J48 pruned tree
------------------

network_bytes <= 25612604
|   percent_packet_loss <= 1
|   |   network_bytes <= 11024041
|   |   |   percent_packet_loss <= 0
|   |   |   |   network_bytes <= 3275210: NAKcast-0.05 (3.0)
|   |   |   |   network_bytes > 3275210: NAKcast-0.025 (11.0)
|   |   |   percent_packet_loss > 0
|   |   |   |   num_receivers <= 3
|   |   |   |   |   network_bytes <= 4260177: NAKcast-0.05 (6.0)
|   |   |   |   |   network_bytes > 4260177
|   |   |   |   |   |   duration <= 2127.917476: NAKcast-0.025 (2.0)
|   |   |   |   |   |   duration > 2127.917476: NAKcast-0.1 (2.0)
|   |   |   |   num_receivers > 3: NAKcast-0.05 (4.0)
|   |   network_bytes > 11024041: NAKcast-0.025 (17.0/1.0)
|   percent_packet_loss > 1: NAKcast-0.025 (37.0/3.0)
network_bytes > 25612604
|   network_bytes <= 25729972: Ricochet-R8C3 (2.0)
|   network_bytes > 25729972
|   |   num_receivers <= 20: Ricochet-R4C3 (62.0)
|   |   num_receivers > 20
|   |   |   std_dev <= 5439.952831: Ricochet-R4C3 (2.0)
|   |   |   std_dev > 5439.952831: Ricochet-R8C3 (2.0)
```

Decision tree



Support vector machine (SVM)

Initial Evaluation of Configuration Approaches

Evaluated approaches based on:

- Boundedness/time complexity
- Accuracy for environments known at training time
- Accuracy for environments unknown *a priori*
- Complexity of managing environments with responses

Approach	Boundedness	Accuracy (known)	Accuracy (unknown)	Development complexity
Policy based	**Yes**	**Perfect (100%)**	**Low (default)**	**Medium -High**

Initial Evaluation of Configuration Approaches

Evaluated approaches based on:

- Boundedness/time complexity
- Accuracy for environments known at training time
- Accuracy for environments unknown *a priori*
- Complexity of managing environments with responses

Approach	Boundedness	Accuracy (known)	Accuracy (unknown)	Development complexity
Policy based	**Yes**	**Perfect (100%)**	**Low (default)**	**Medium -High**
Reinforcement Learning	**No**	High	Medium	**Low**

Initial Evaluation of Configuration Approaches

Evaluated approaches based on:

- Boundedness/time complexity
- Accuracy for environments known at training time
- Accuracy for environments unknown *a priori*
- Complexity of managing environments with responses

Approach	Boundedness	Accuracy (known)	Accuracy (unknown)	Development complexity
Policy based	**Yes**	**Perfect (100%)**	**Low (default)**	**Medium -High**
Reinforcement Learning	**No**	High	Medium	**Low**
Decision Tree	**Yes (data dependent)**	High (99%)	High (87%)	**Low**

Initial Evaluation of Configuration Approaches

Evaluated approaches based on:

- Boundedness/time complexity
- Accuracy for environments known at training time
- Accuracy for environments unknown *a priori*
- Complexity of managing environments with responses

Approach	Boundedness	Accuracy (known)	Accuracy (unknown)	Development complexity
Policy based	**Yes**	**Perfect (100%)**	**Low (default)**	**Medium -High**
Reinforcement Learning	**No**	High	Medium	**Low**
Decision Tree	**Yes (data dependent)**	High (99%)	High (87%)	**Low**
Neural Network	**Yes (constant)**	**Perfect (100%)**	High (85%)	**Low**

Initial Evaluation of Configuration Approaches

Evaluated approaches based on:

- Boundedness/time complexity
- Accuracy for environments known at training time
- Accuracy for environments unknown *a priori*
- Complexity of managing environments with responses

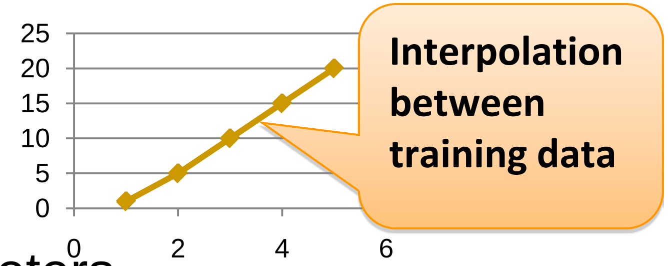
Approach	Boundedness	Accuracy (known)	Accuracy (unknown)	Development complexity
Policy based	**Yes**	**Perfect (100%)**	**Low (default)**	**Medium -High**
Reinforcement Learning	**No**	High	Medium	**Low**
Decision Tree	**Yes (data dependent)**	High (99%)	High (87%)	**Low**
Neural Network	**Yes (constant)**	**Perfect (100%)**	High (85%)	**Low**
SVM	**Yes (constant)**	**Perfect (100%)**	High (79%)	**Low**

ADAptive Middleware & Network Transports (ADAMANT)

ADAMANT incorporates:

Artificial Neural Network (ANN)

- Trained on protocol properties
- Interpolates/Extrapolates for new environments
- Determines optimal protocol/parameters
- Constant time performance



Protocol Optimization

Data Distribution Service (DDS)

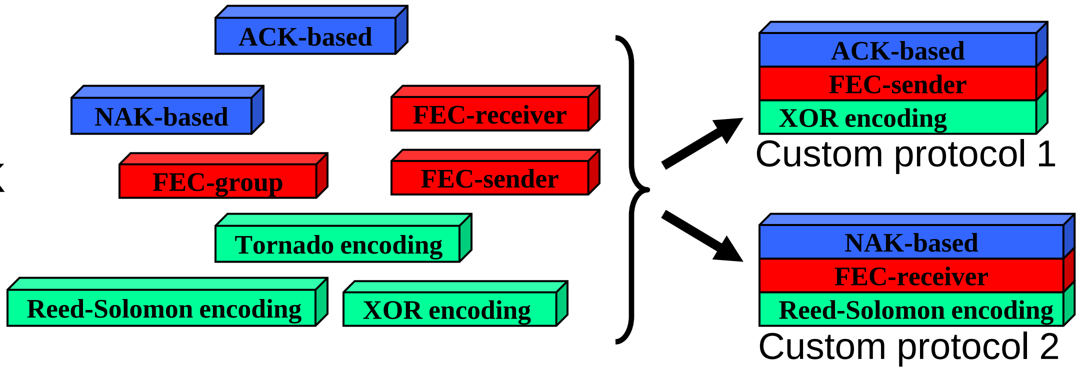
- OMG pub/sub standard, rich QoS support
- OpenDDS, OpenSplice implementations
  - Pluggable transport protocol frameworks
  - Open source

OpenDDS

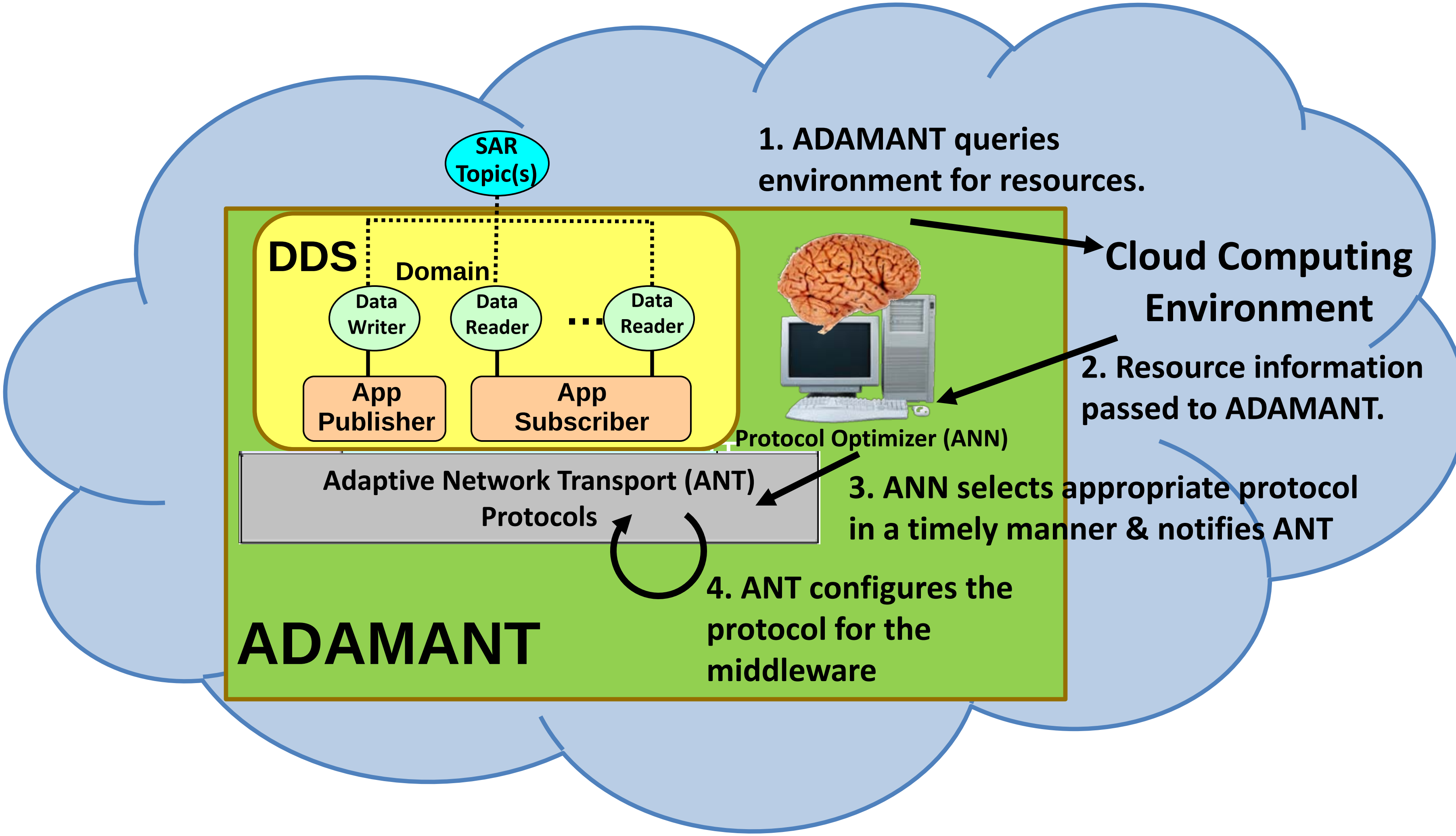
PRISMTECH OpenSplice™ DDS

Adaptive Network Transports (ANT) framework

- Transport protocol framework
- Composable modules
- Fine-grained protocol control



ADAMANT Architecture & Control Flow



Key:

Control interaction between subsystems

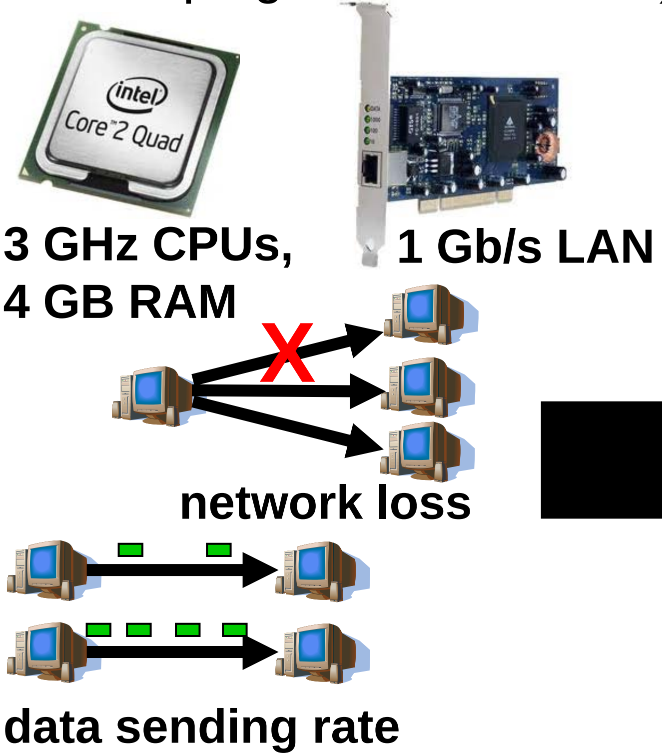
Assoc. between reader/writer &topic

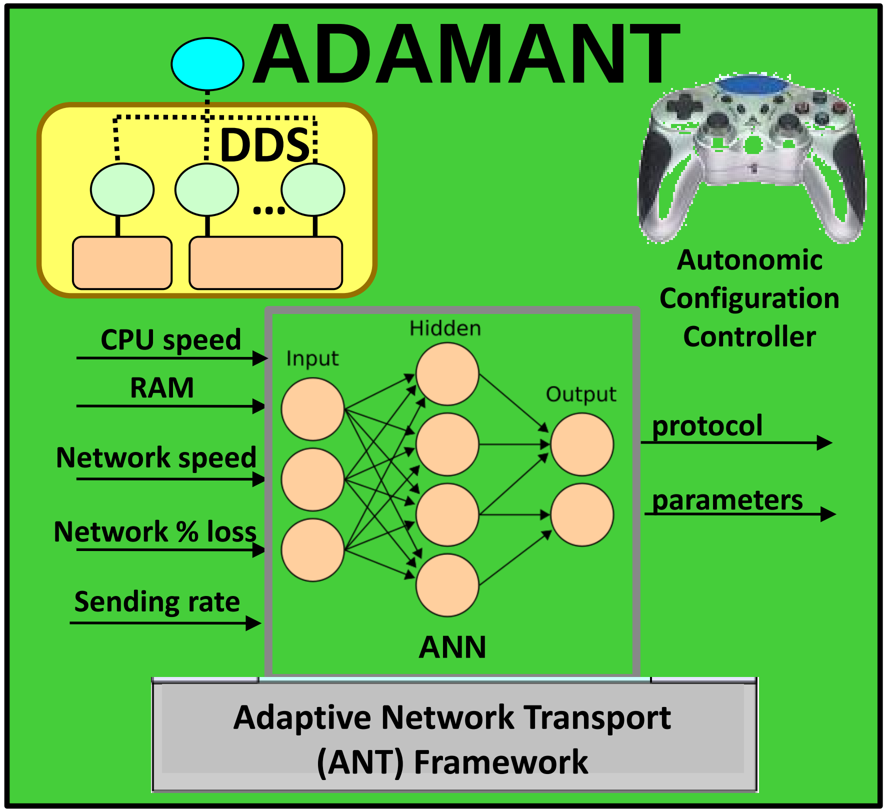
Addressing Challenges for Datacenter in Cloud Envs. (1/3)

ADAMANT addresses challenge 1 (development complexity) via ANNs to manage protocol selection & implementation transformation

ANNs manage the development complexity of protocol management:

- Automatically manage inherent complexity of relationships between environment and protocols
- Used directly in implementations (*i.e.*, avoids accidental complexity of developing implementation)



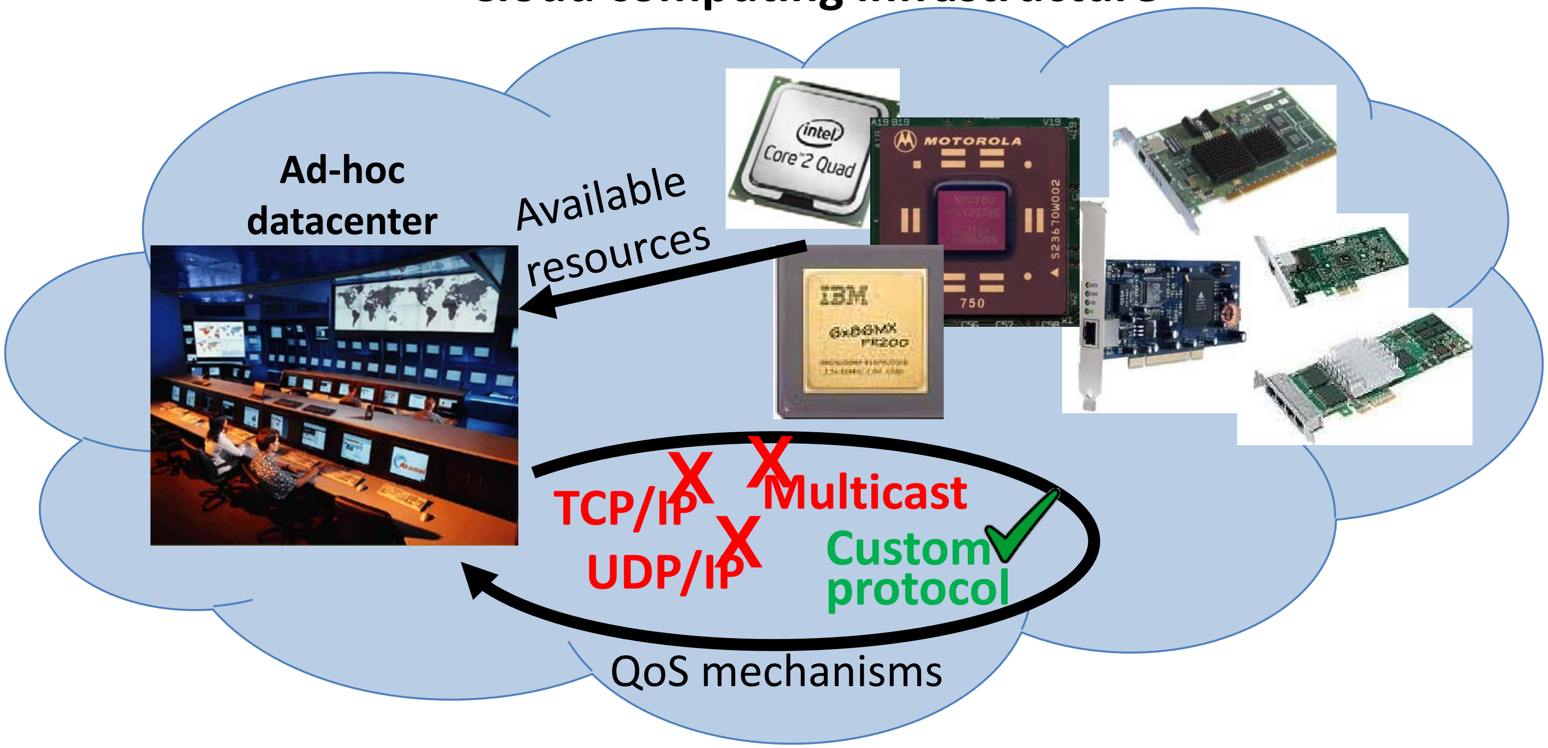


Addressing Challenges for Datacenter in Cloud Envs. (2/3)

ADAMANT addresses challenge 2 (accurate configuration) by overfitting ANN to data

Overfitting data increases ANN's accuracy for selecting appropriate protocol

Cloud computing infrastructure



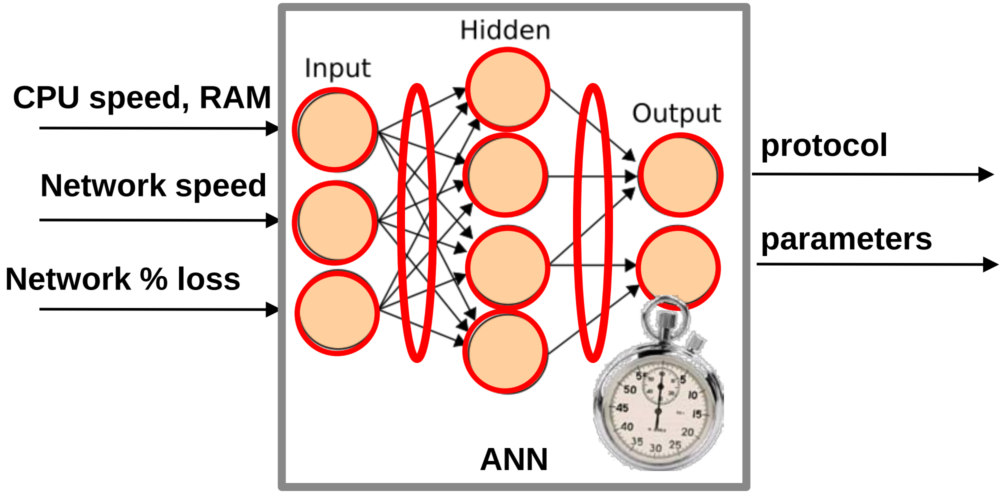
ADAMANT accurately selects correct protocol

Addressing Challenges for Datacenter in Cloud Envs. (3/3)

ADAMANT addresses challenge 3 (timely configuration) via ANN w/ bounded constant-time response

ANNs are equation based:

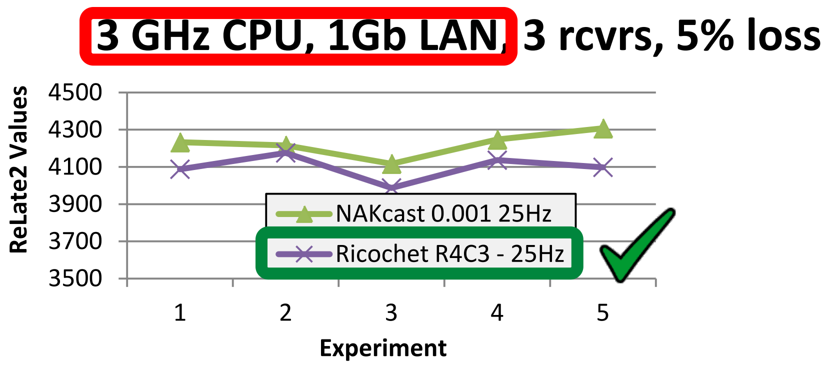
- Equations based on nodes and connections
- Fixed number of inputs, hidden nodes, outputs (determined at off-line training time)
- Constant # of connections (determined at training time)

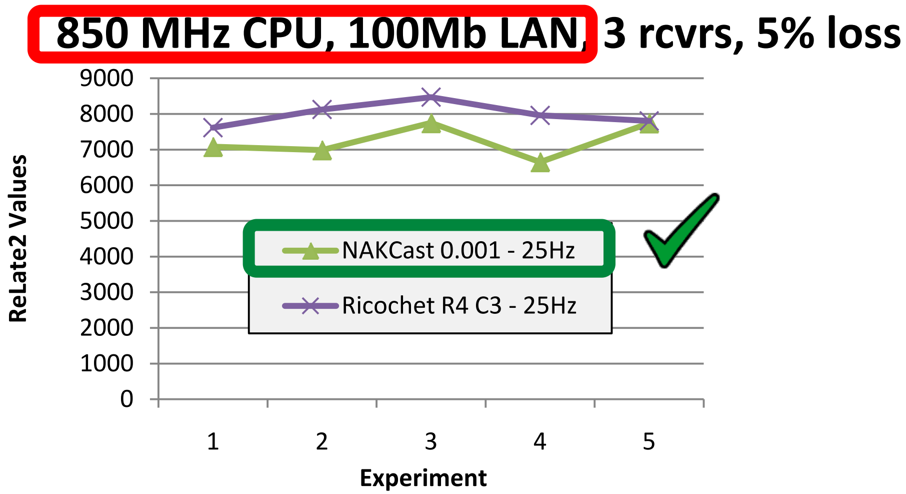


Empirical Results – Different Hardware → Different Protocols

Experimental environment:

- Using protocols that balance reliability and low latency
  - IP Multicast w/ NAKs (NAKcast)
  - Modified FEC (Ricochet)
- Varied CPU speed, network bandwidth
- Conducted several training runs



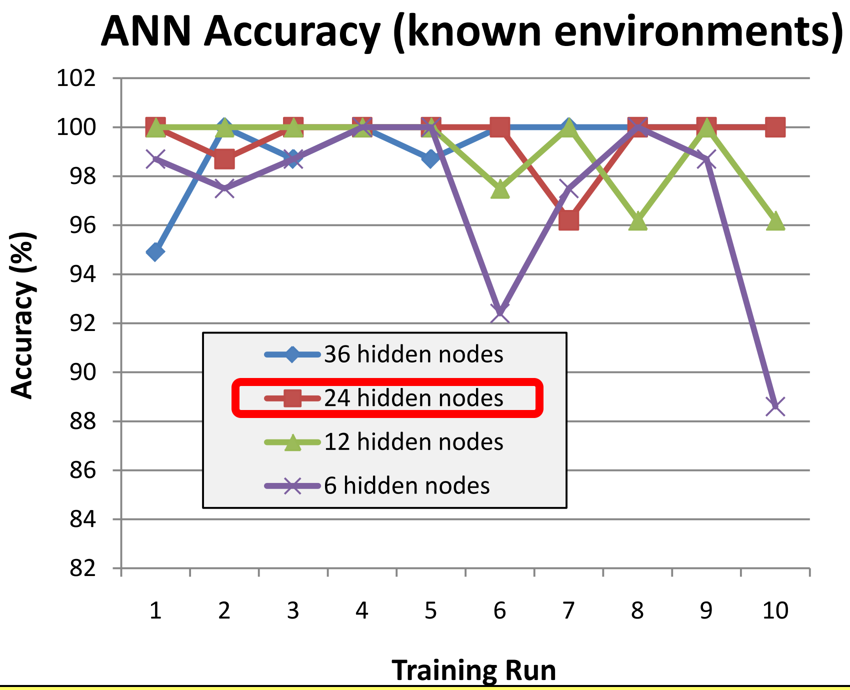


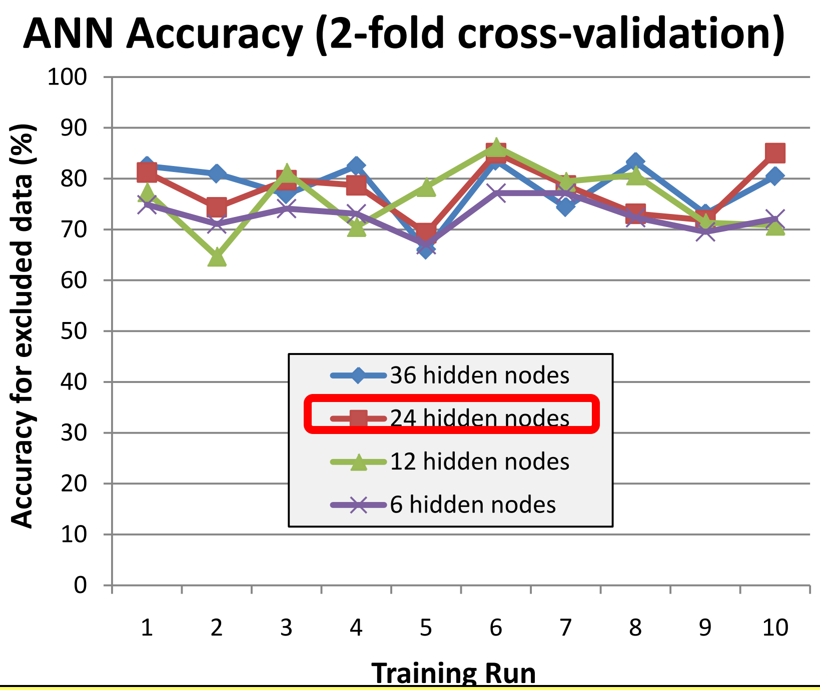
<u>Difference in hardware triggers a difference in appropriate transport protocol</u>

Empirical Results - Accuracy

Experimental environment:

- 394 operating environments
- Varied CPU speed, network bandwidth, # of data receivers, sending rate
- Conducted several training runs
- ANN outputs tested against known correct responses



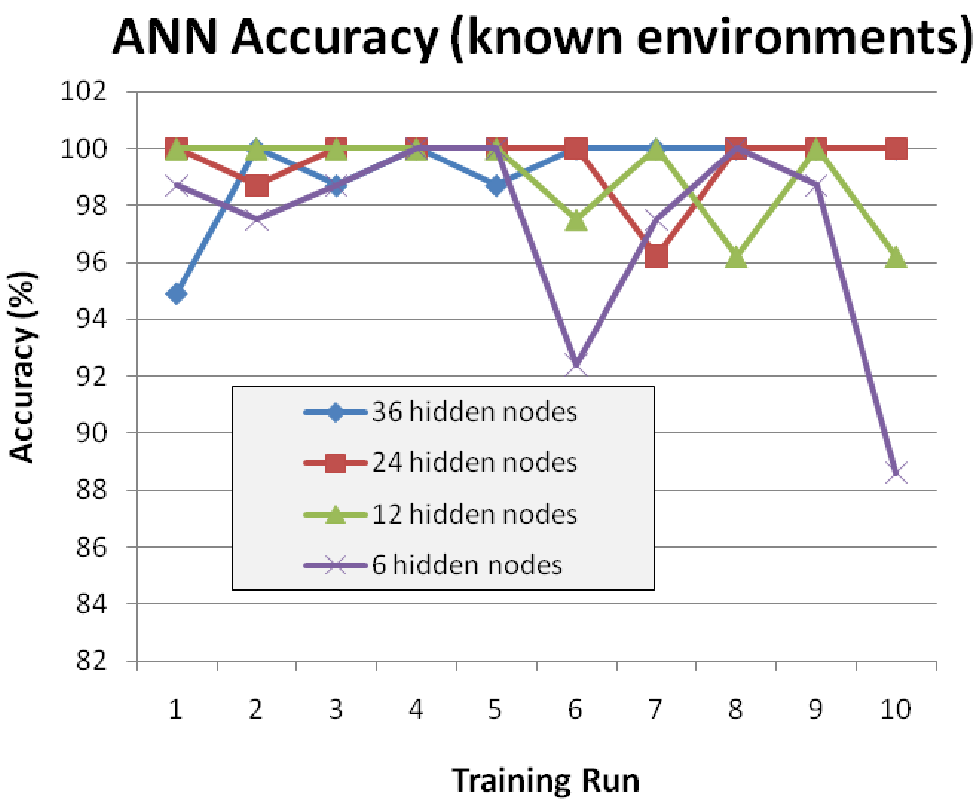


ANNs w/ 24 nodes provide most instances of 100% accuracy, highest average accuracy with 2-fold cross-validation (78%)

Qualifying Exam Hypothesis for ADAMANT

Evaluation Criteria	Description
(H1) Adjust for known environment	Hypothesize that ADAMANT will provide adjustment improvement for **known** environments at least 85% of the time

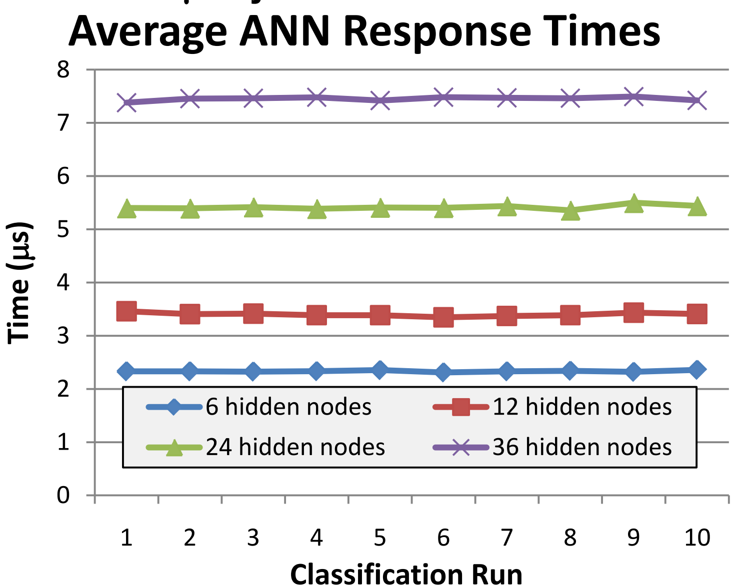
Using ANNs, ADAMANT provides 100% accurate adjustment for known environments

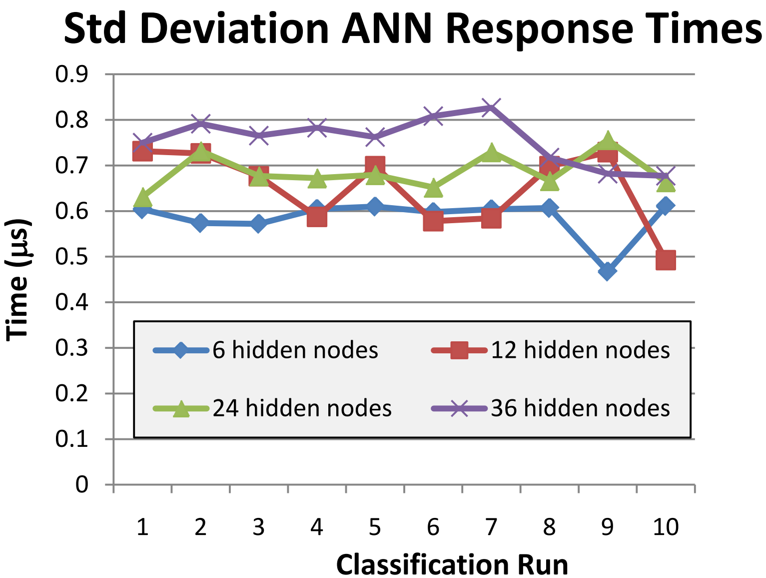


Empirical Results – ANN Timeliness

Experimental environment:

- 394 operating environments
- Emulab: 3 GHz CPU, 2GB of RAM, Fedora Core 6 w/ real-time patches
- Sub 10 μs average response times for all ANN configurations
- Sub μs jitter for all ANN configurations





ADAMANT addresses the challenges of

(1) Development complexity via machine learning to determine protocols,

(2) Configuration accuracy via overfitted supervised machine learning &

(3) Configuration timeliness via equation-based machine learning

ADAMANT Related Publications & Presentations

Conference Publications

1. Hoffert, J., Schmidt, D., & Gokhale, A. (2010, November). Adapting Distributed Real-time and Embedded Publish/Subscribe Middleware for Cloud-Computing Environments, Proceedings of the ACM/IFIP/USENIX 11th International Middleware Conference (Middleware 2010), Bangalore, India.
2. Hoffert, J., & Schmidt, D. (2010, October). Evaluating Supervised Machine Learning for Adapting Enterprise DRE Systems, Proceedings of the 2010 International Symposium on Intelligence Information Processing and Trusted Computing (IPTC 2010), Huanggang, China.

Workshop Publications

3. Hoffert, J., Schmidt, D., & Gokhale, A. (2010, April). Adapting and Evaluating Distributed Real-time and Embedded Systems in Dynamic Environments,The 1st International Workshop on Data Dissemination for Large scale Complex Critical Infrastructures (DD4LCCI 2010), Valencia, Spain.

First Author

Focus Area 4: Adapting DRE Systems in Dynamic Envs.

Focus Area 1 → Focus Area 2 → Focus Area 3 → Focus Area 4

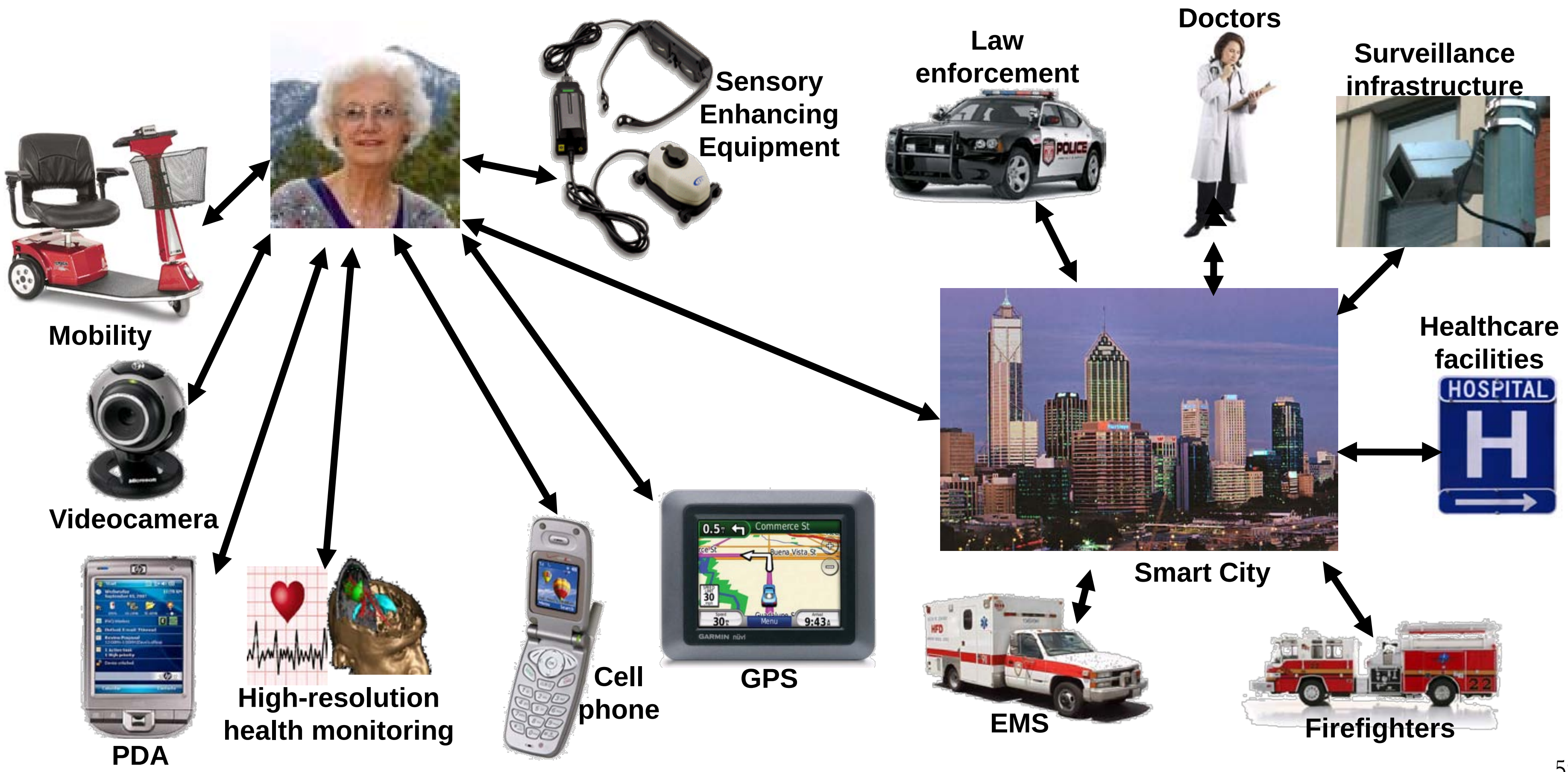
Motivating Example: Smart City Ambient Assisted Living (SCAAL)

- Aging population increasing, # of health care workers decreasing

Motivating Example: SCAAL Application (cont.)

Scenario

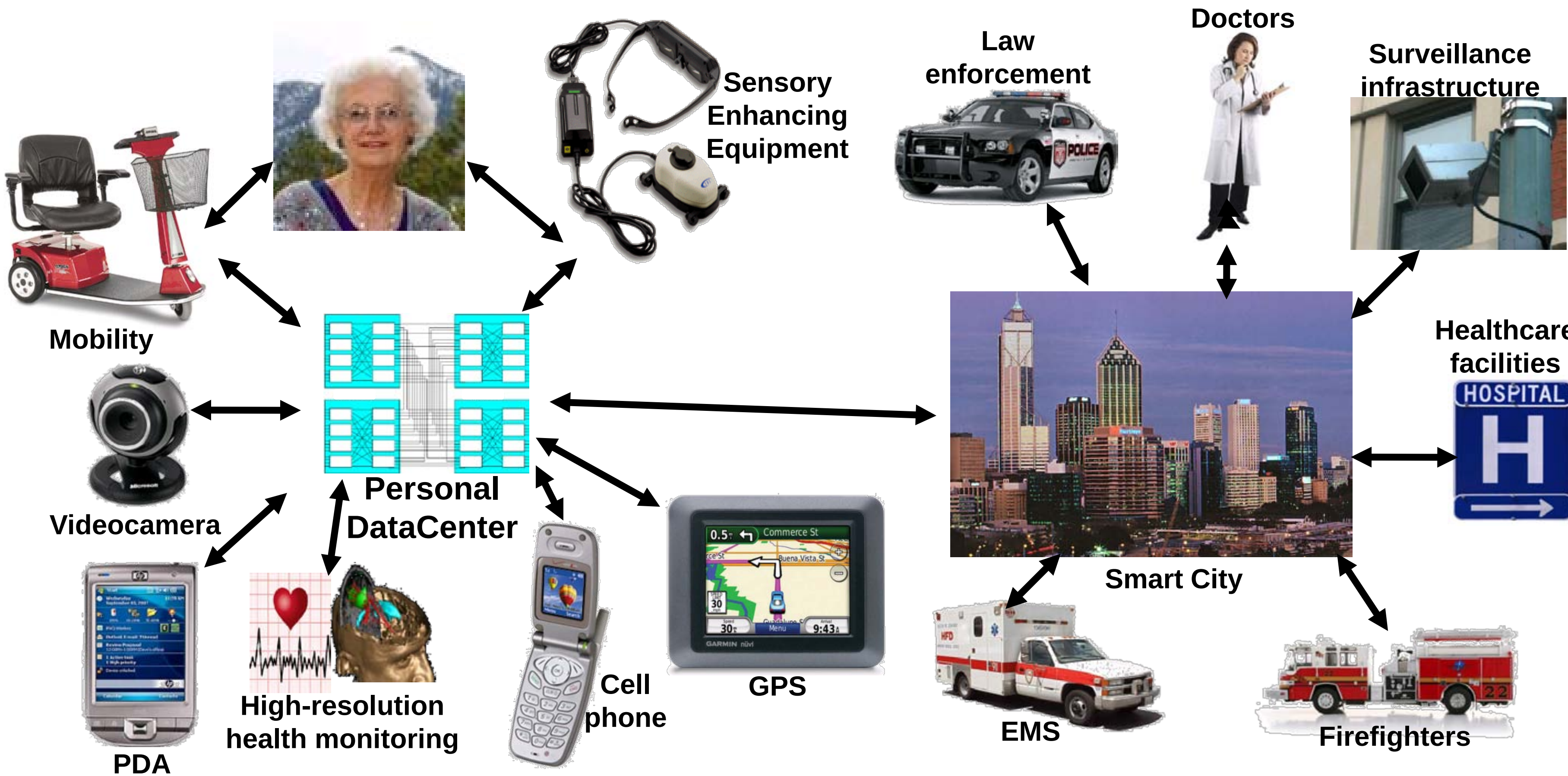
- Aging population increasing, # of health care workers decreasing
- Increase elderly autonomy in urban areas via coordination of personal equipment & sensing/aware "smart cities"



Motivating Example: SCAAL Application (cont.)

Scenario

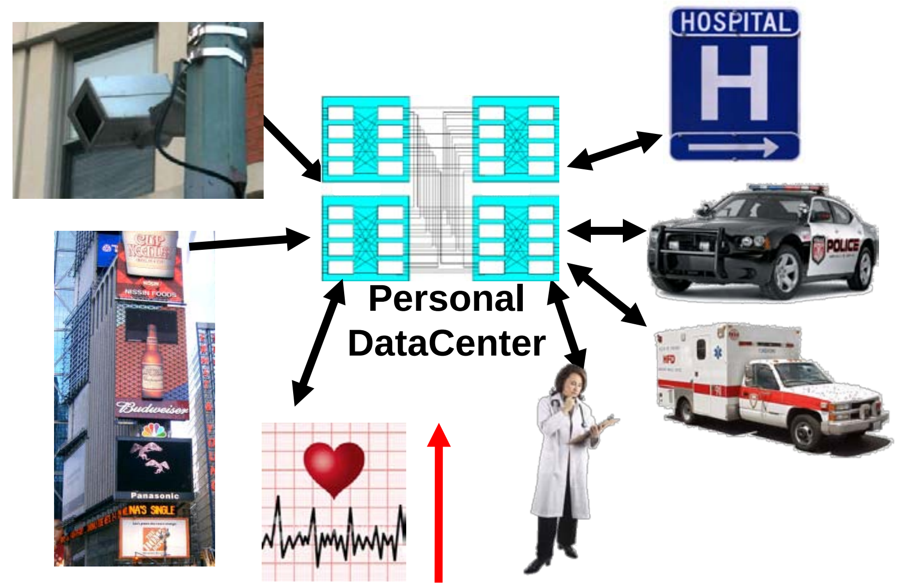
- Aging population increasing, # of health care workers decreasing
- Increase elderly autonomy in urban areas via coordination of personal equipment & sensing/aware "smart cities"
- Utilize personal data center to manage personal & environment data



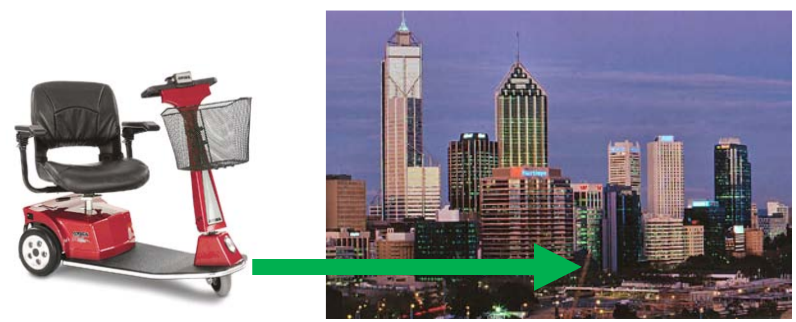
Motivating Example: SCAAL Application (cont.)

Requirements

- Operate in dynamic environment, e.g.,
  - Varying # of senders, receivers
  - Varying network bandwidth, loss
  - Varying data sending rates (e.g., more updates for critical data)



- Support QoS as environment changes
  - Reliability & latency
    - e.g., high resolution health monitoring
    - Multimedia data



Challenges for QoS in Dynamic Environments (1/2)

Challenge 1: Environment monitoring & update dissemination

As environment changes, updates need to be propagated throughout the application.

Normal health information,
Low update rate required

Low sending rate

Doctor detects health anomaly,
update rate increased

High sending rate,
More detailed information

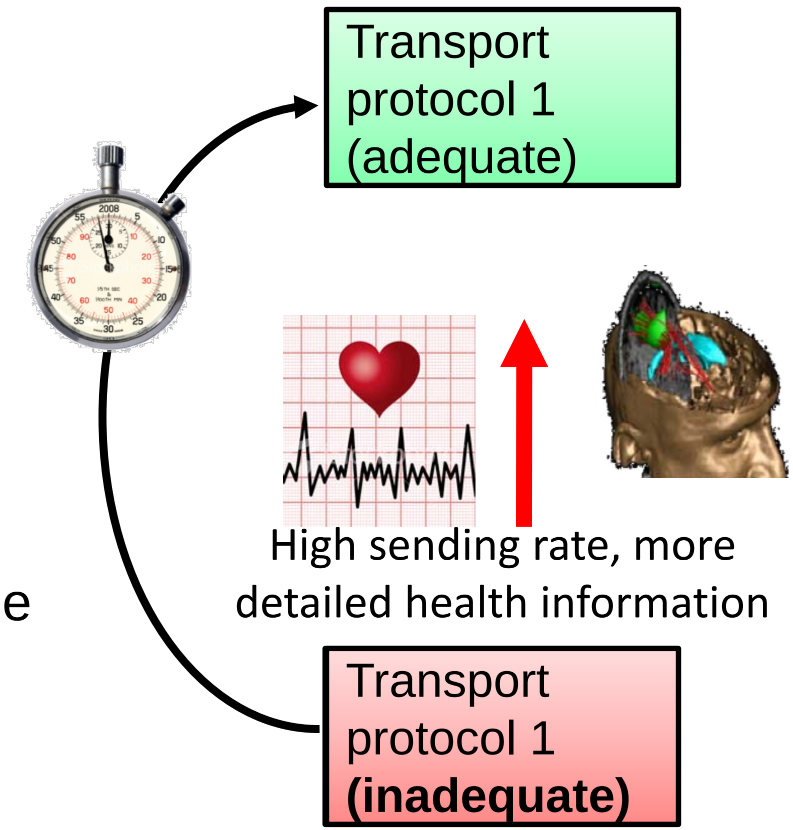
(initial environment) … (environment modification)

Elderly person traveling through smart city

Challenges for QoS in Dynamic Environments (2/2)

Challenge 2: Optimal accuracy for unknown environments while maintaining timeliness

- Inaccurate adjustment could lead to reduced health or death
- Ideally want accuracy for unknown environments to rival accuracy for known environments
- Timeliness concerns need to be addressed while selecting an adequate QoS mechanism



QoS in Dynamic Environments: Related Research

Related Research
PrismTech’s Tuner application, http://www.opensplice.com
Real-Time Innovations’ RTI Analyzer, RTI Scope, & RTI Protocol Analyzer, http://rti.com/products/developer_platform
M. Caporuscio *et a* “Design and Evaluation of a Support Service for Mobile, Wireless Publish/Subscribe Ap ations”, *IEEE Transactions on Software Engineering*, vol. 29, no. 12, pages1059–1071.
P. Grace *et al*. “Deep Middlew e Divergent Grid”, *Proceedings of the ACM/IFIP/USENIX 20* November 2005, Grenoble, France
P. Vienne & J.-L. Sourrouille. “A Middleware for Autonomic QoS Management Based on Learning. *Proceedings of the 5th International Workshop on Software Engineering & Middleware*, September 2005, Lisbon, Portugal
C. Hersenns *et al.* "Context-driven Autonomic Adaptation of SLA“, *6th International Conference on Service Oriented Computing,* December 2008, Sydney, Australia

Good for ***manually checking*** the run-time QoS status

QoS in Dynamic Environments: Related Research

Related Research
PrismTech’s Tuner application, http://www.opensplice.com
Real-Time Innovations’ RTI Analyzer, RTI Scope, & RTI Protocol Analyzer, http://rti.com/products/developer_platform
M. Caporuscio *et al*. “Design and Evaluation of a Support Service for Mobile, Wireless Publish/Subscribe Applications”, *IEEE Transactions on Software Engineering*, vol. 29, no. 12, pages1059–1071.
P. Grace *et al*. “Deep Middleware for the Divergent Grid”, *Proceedings of the ACM/IFIP/USENIX 2005 International Conference on Middleware*, November 2005, Grenoble, France
P. Vienne & J.-L. Sourrouille. “A Middleware for Autonomic QoS Management Based on Learning. *Proceedings of the 5th International Workshop on Software Engineering & Middleware*, September 2005, Lisbon, Portugal
C. Hersenns *et al.* "Context-driven Autonomic Adaptation of SLA“, *6th International Conference on Service Oriented Co…* … Australia

Good for ***developing*** adaptation applications

QoS in Dynamic Environments: Related Research

Related Research
PrismTech's Tuner application, http://www.opensplice.com
Real-Time Innovations' RTI Analyzer, RTI Scope, & RTI Protocol Analyzer http://rti.com/products/develo…
M. Caporuscio *et al*. "Design a… …eless Publish/Subscribe Applications", … on Software Engineering, vol. 29, no. 12, pages1059–1071.
P. Grace *et al*. "Deep Middlewar… …e Divergent Grid", *Proceedings of the ACM/IFIP/USENIX 2005 Int…ational Conference on Middleware*, November 2005, Grenoble, France
P. Vienne & J.-L. Sourrouille. "A Middleware for Autonomic QoS Management Based on Learning. *Proceedings of the 5th International Workshop on Software Engineering & Middleware*, September 2005, Lisbon, Portugal
C. Hersenns *et al.* "Context-driven Autonomic Adaptation of SLA", *6th International Conference on Service Oriented Computing,* December 2008, Sydney, Australia

Good for adaptation when ***timeliness is not a driving concern***

Properties for QoS Support in Dynamic Environments

What properties help us assess research to support QoS in dynamic environments?

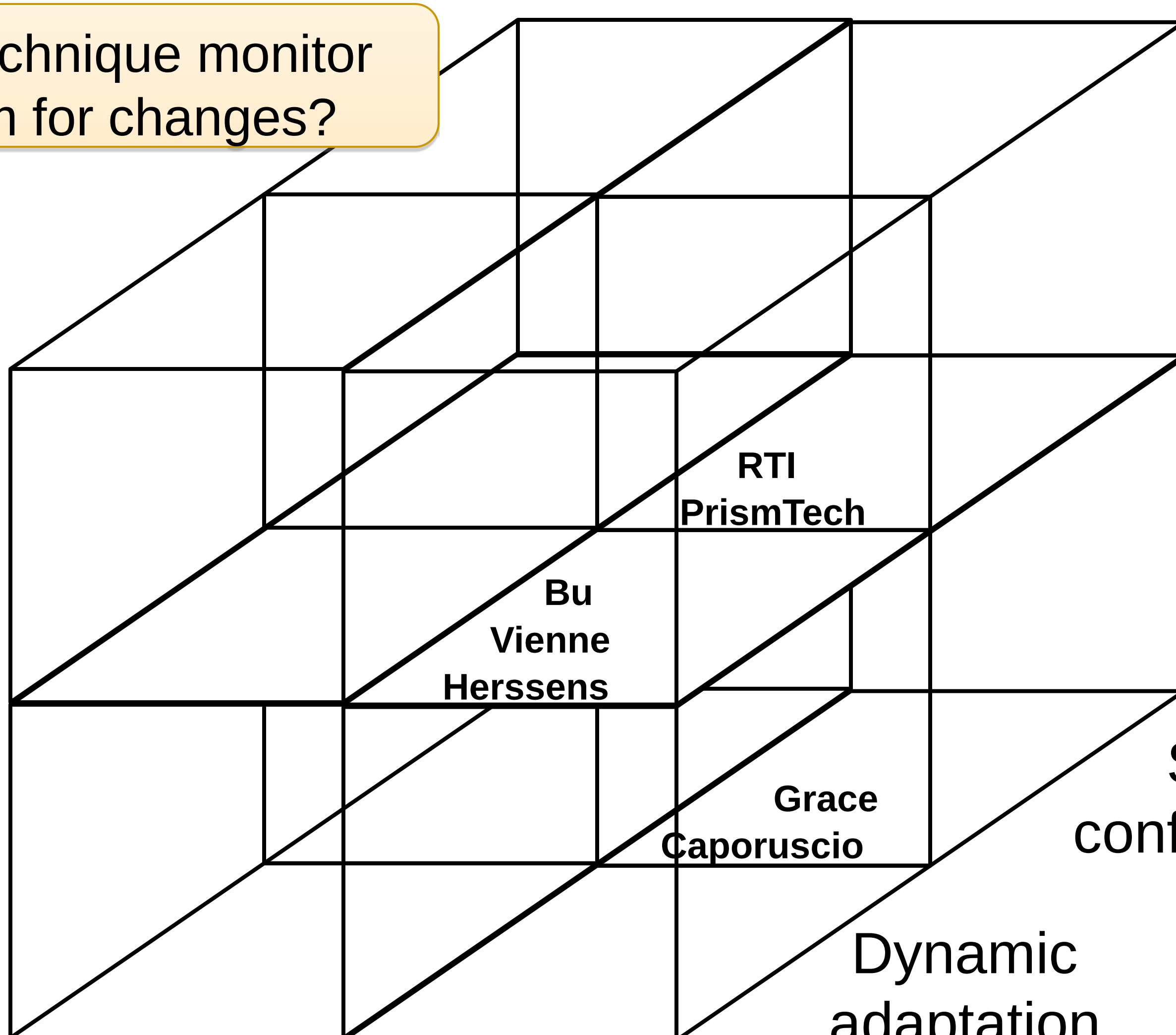
Property	Description
Monitor environment	Does the technique know when the environment has changed?
Dynamic adaptation	Does the technique perform adaptation adjustments while system is running?
Timely adaptation	Can the technique change to a more appropriate protocol in a timely manner?

QoS in Dynamic Environments: Related Work

Does the technique monitor the system for changes?

Monitor environment

No monitoring



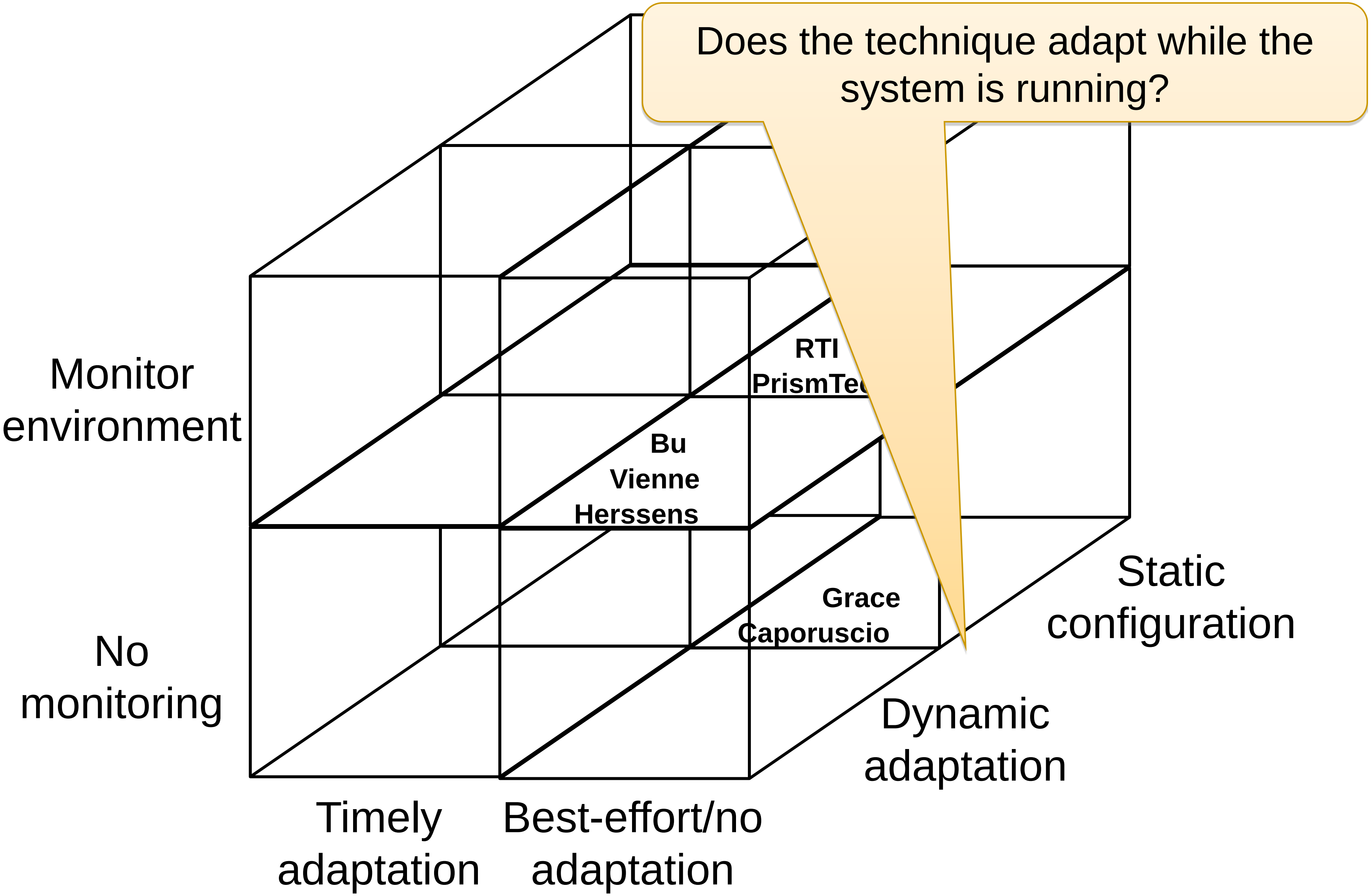
Static configuration

Dynamic adaptation

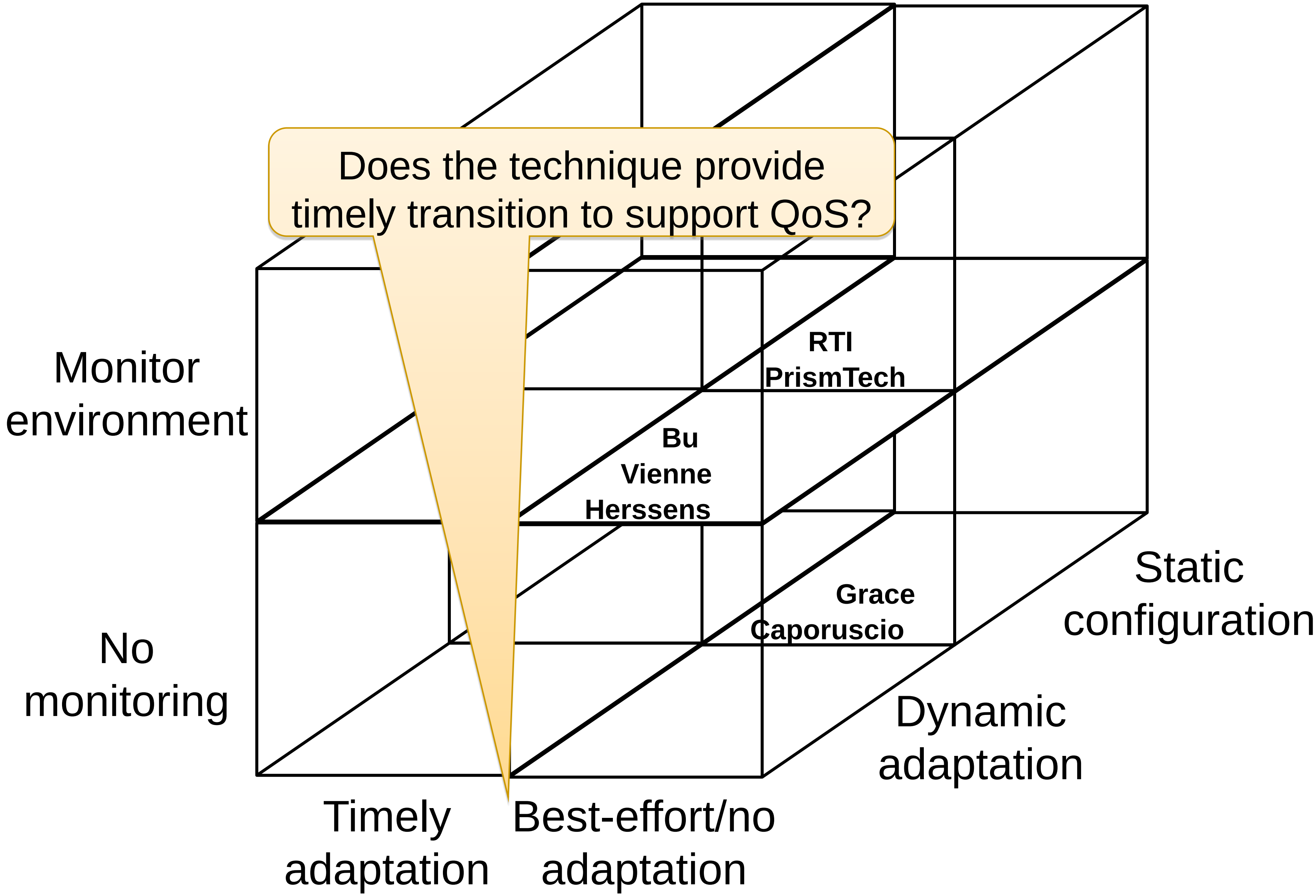
Timely adaptation

Best-effort/no adaptation

QoS in Dynamic Environments: Related Work

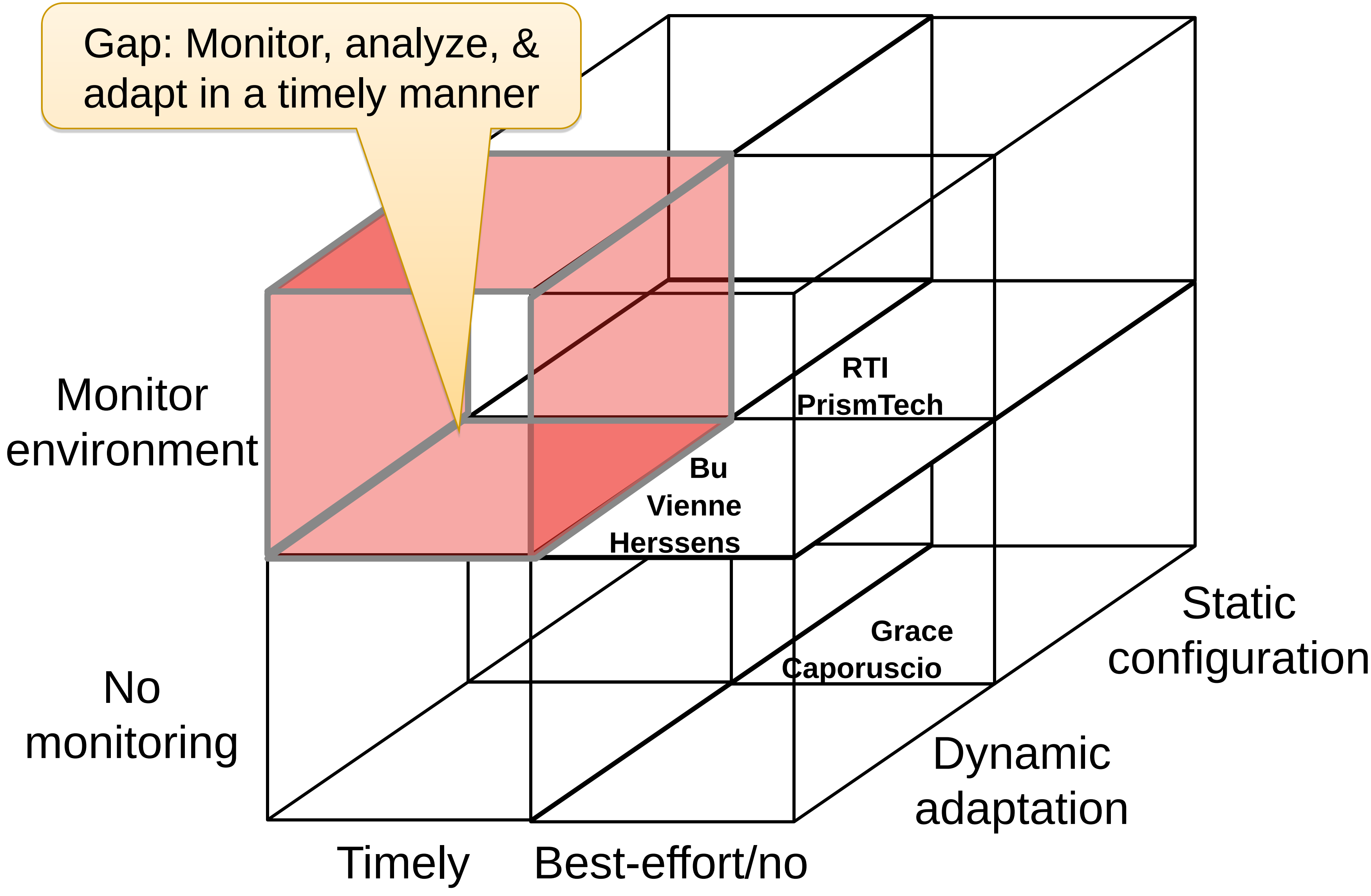


QoS in Dynamic Environments: Related Work



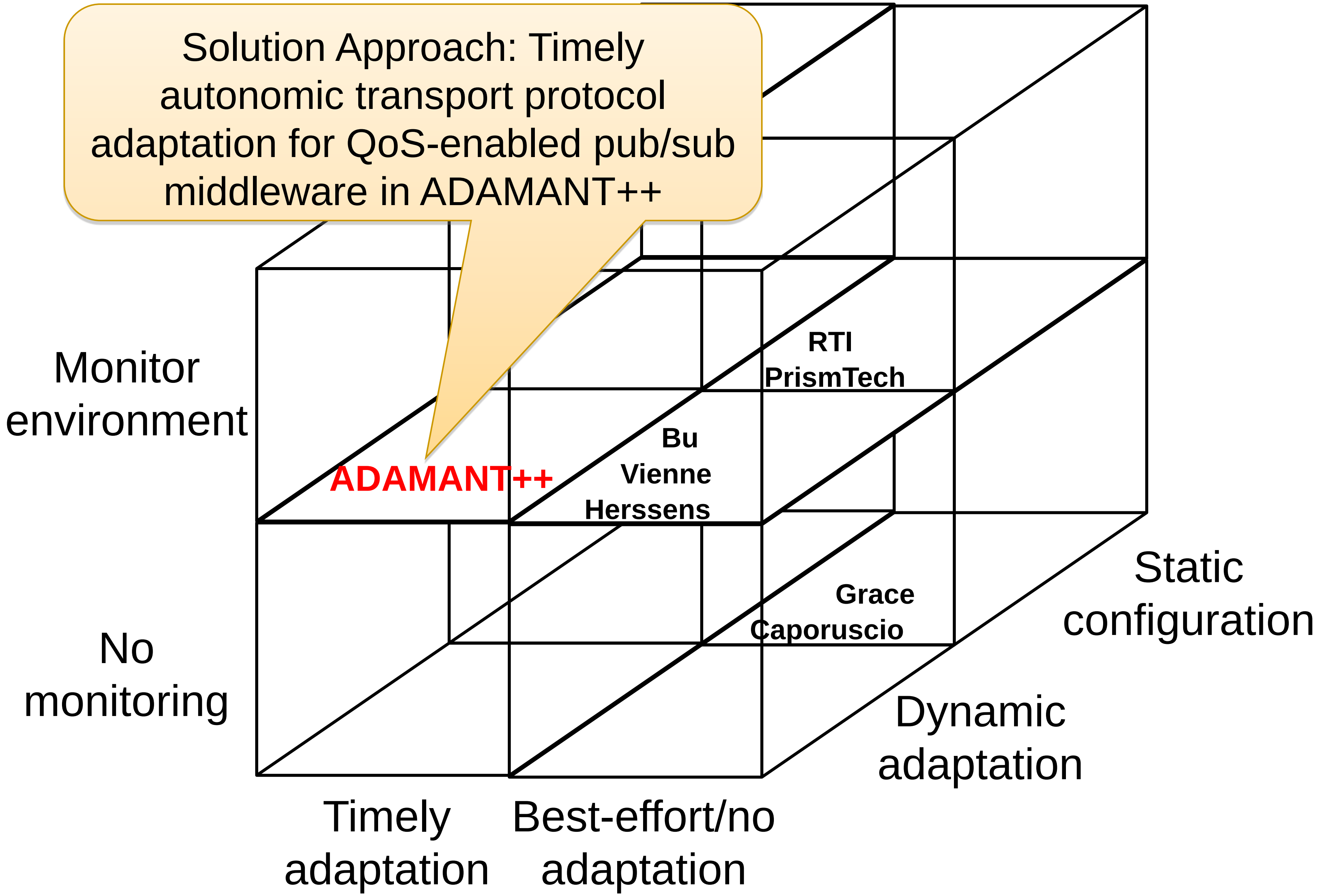
QoS in Dynamic Environments: Open Issues

Gap: Monitor, analyze, & adapt in a timely manner

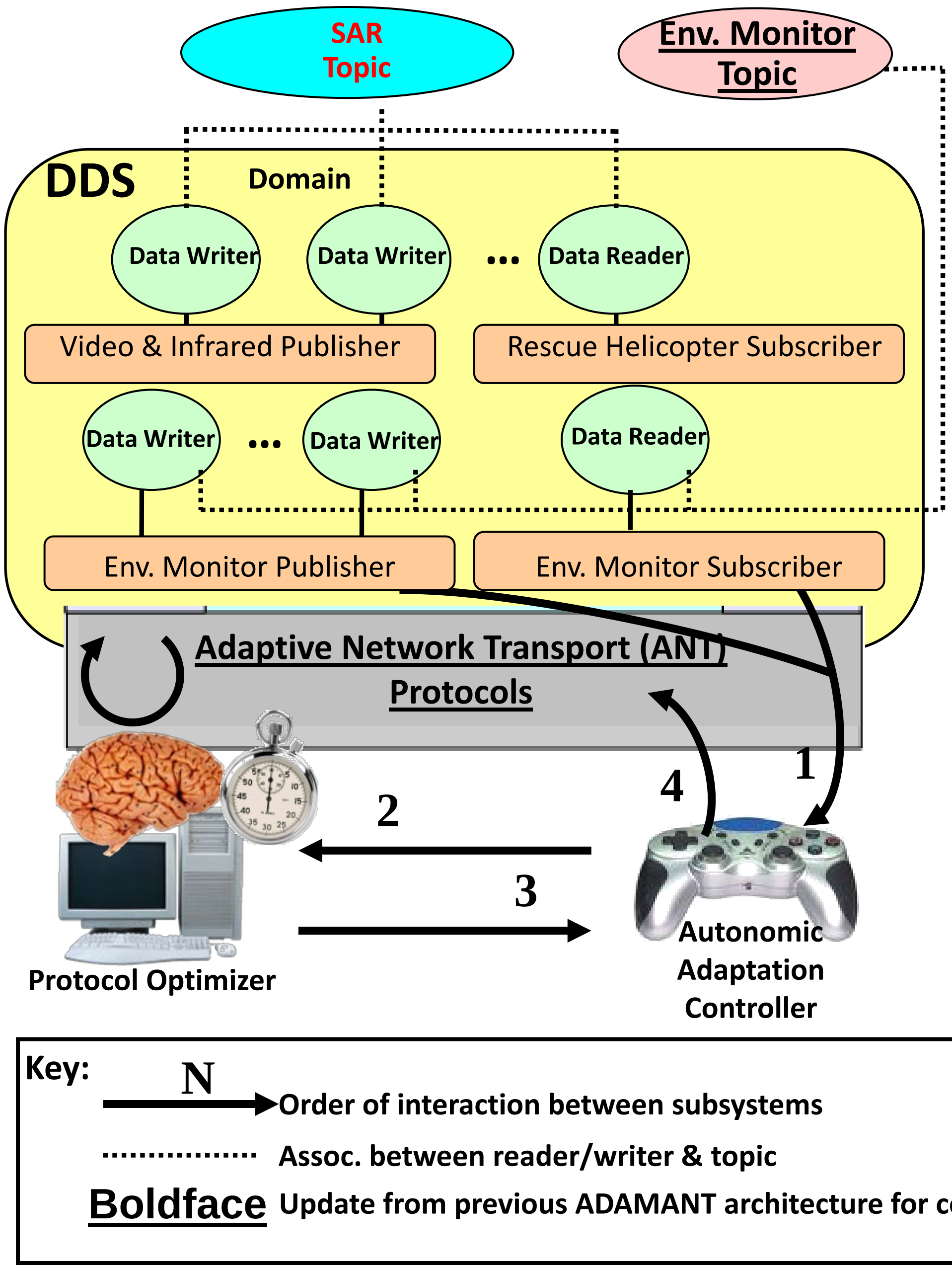


Current gap makes it *hard* for DRE systems in dynamic environments to adapt QoS in a timely manner

Solution Approach: Adaptive M/W & Network Transports++



ADAMANT++ Architecture & Control Flow



1. Middleware disseminates environment feedback

2. Controller monitors feedback, sends to optimizer

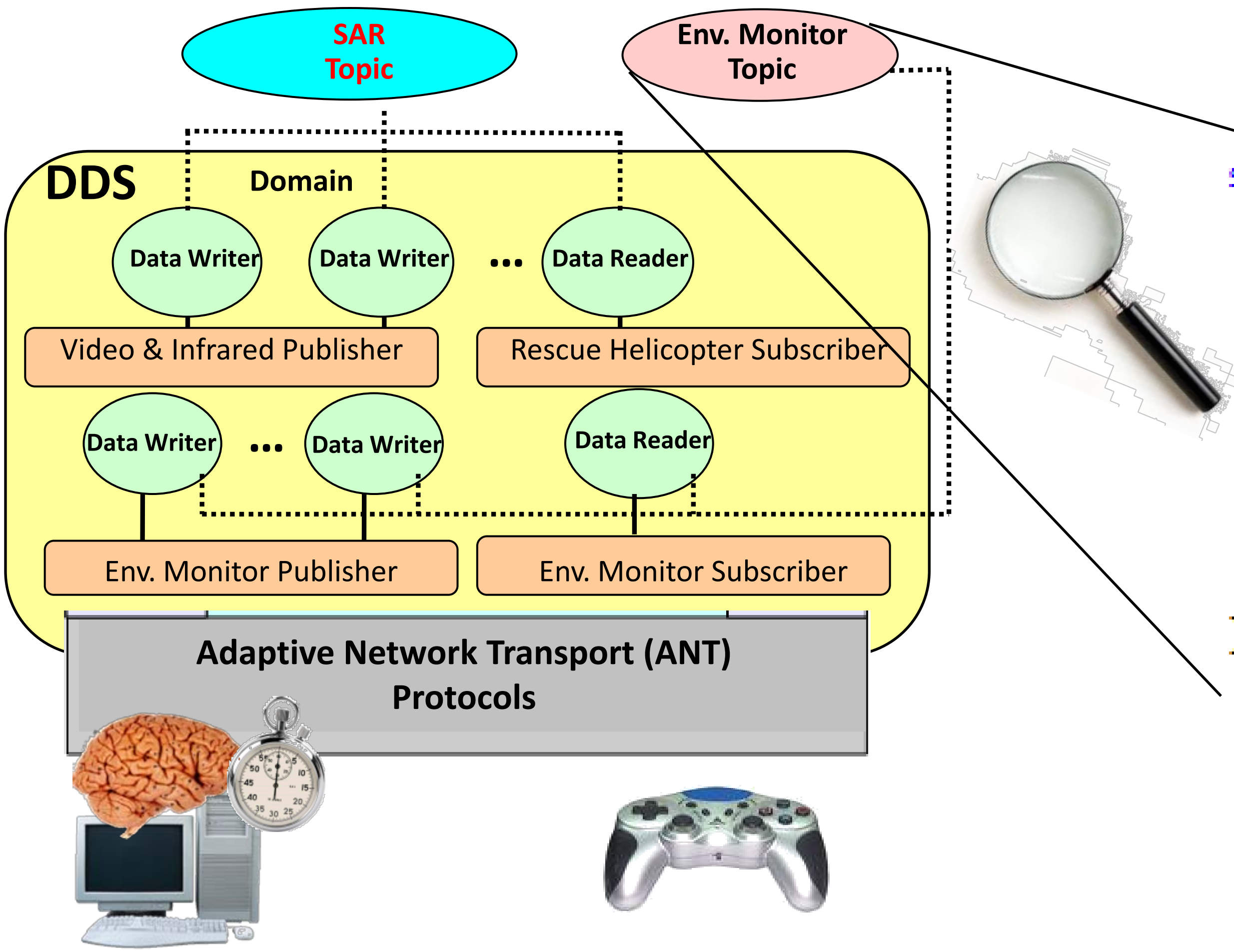
3. Optimizer determines optimal protocol & settings **(leveraging multiple machine learning techniques)**, returns to controller

4. Controller compares current & optimal settings, notifies ANT as needed

5. ANT **dynamically** updates the protocol and/or settings to maintain QoS

Addressing Challenges for QoS in Dynamic Envs. (1/2)

ADAMANT++ addresses Challenge 1 (disseminating updates) via environment monitoring topic



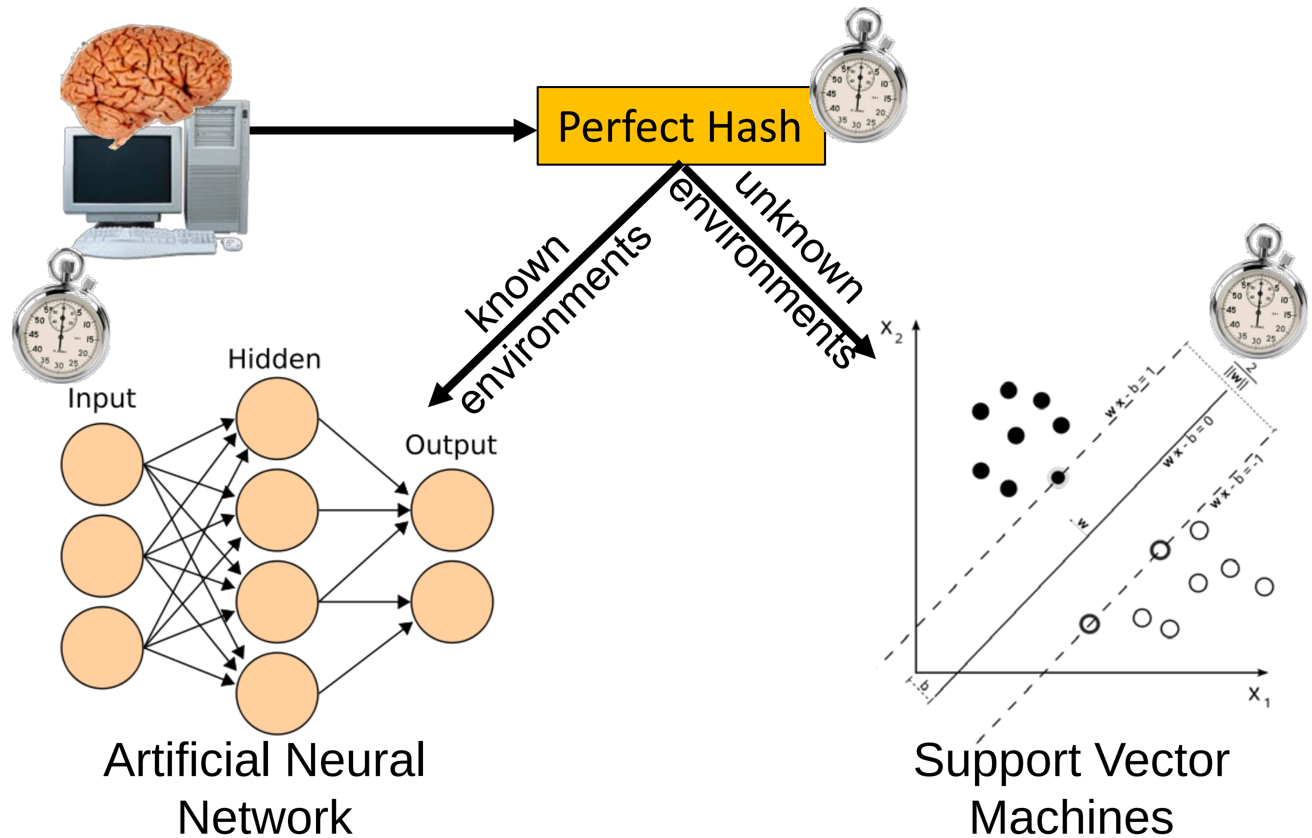

```
struct QosMonitoring {
  long receiver_count;
  long percent_network_loss;
  long send_rate_in_Hz;
  string cpu_speed;
  string ram;
  string network_speed;
  string dds_impl;
  string composite_metric;
};
```

ADAMANT++ leverages DDS to disseminate updates; QoS policies apply to monitoring topic

Addressing Challenges for QoS in Dynamic Envs. (2/2)

ADAMANT++ addresses Challenge 2 (increasing accuracy) via Timely Integrated Machine Learning (TIML)

Environment update



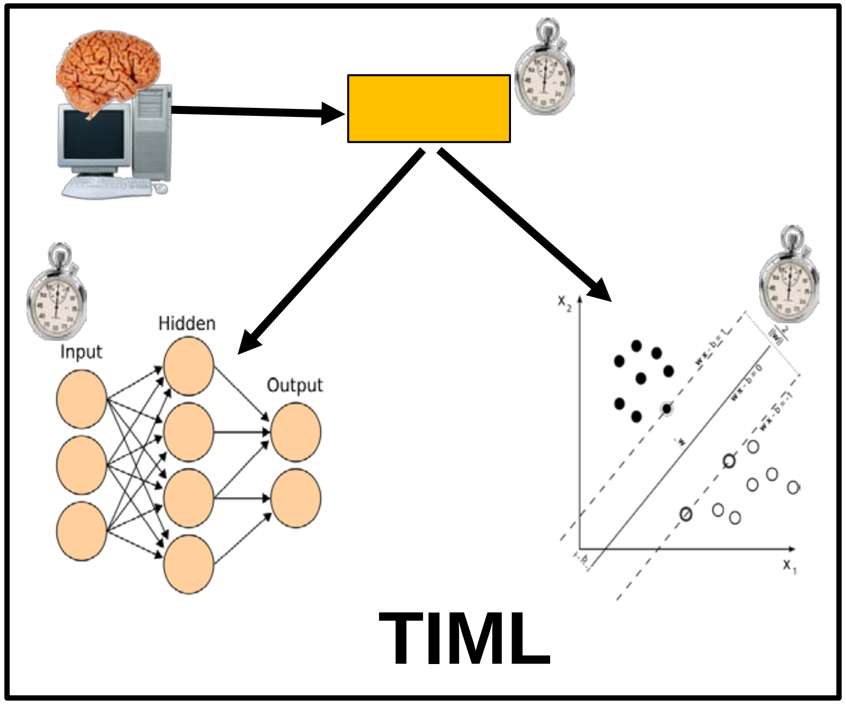
Timely Integration of Machine Learning (TIML)

TIML yields 8.6% accuracy increase for unknown environments (compared to just ANN), maintains timeliness

Proposed Experiment: ADAMANT & Dynamic Environments

Evaluation Criteria	Description
(H2) Adjust for unknown environment	Hypothesize that ADAMANT will provide adjustment improvement for **unknown** environments more than 50% of the time

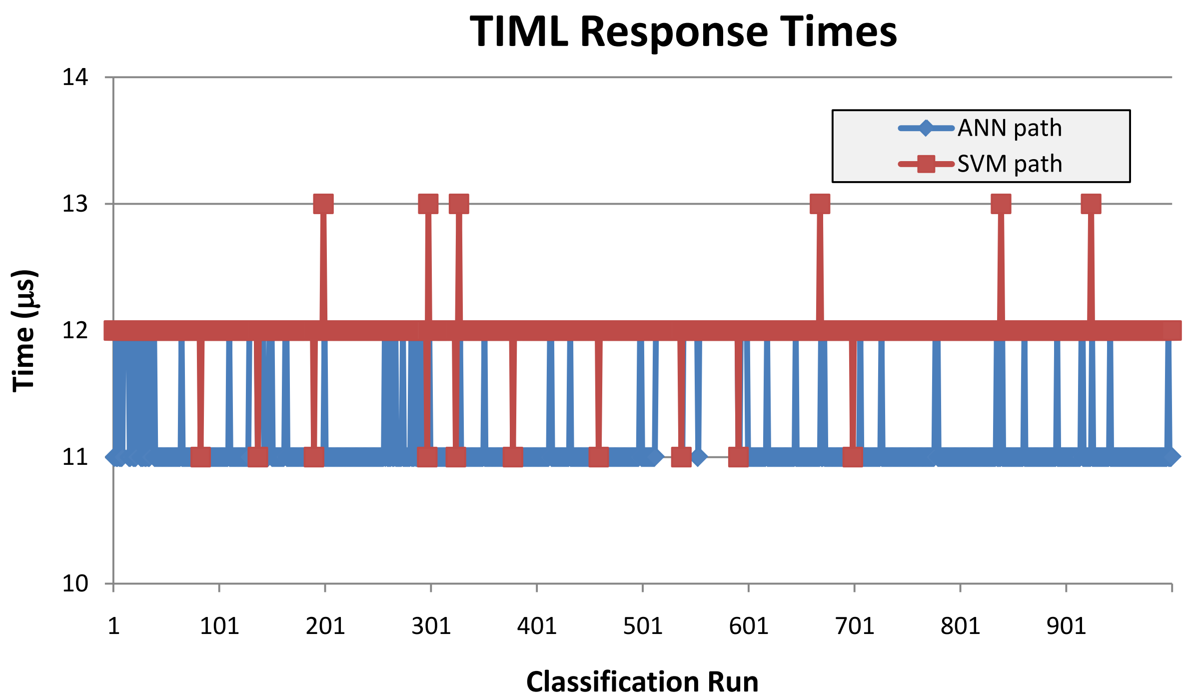
Leveraging TIML, ADAMANT++ provides 86% accuracy for unknown environments.



Empirical Results – TIML Timeliness

Experimental environment:

- 394 operating environments
- Emulab: 3 GHz CPU, 2GB of RAM, Fedora Core 6 w/ real-time patches
- 12 μs response times for determining to use ANN or SVM
- Jitter within timestamp resolution for ANN and SVM paths (*i.e.*, +/- 1 μs)

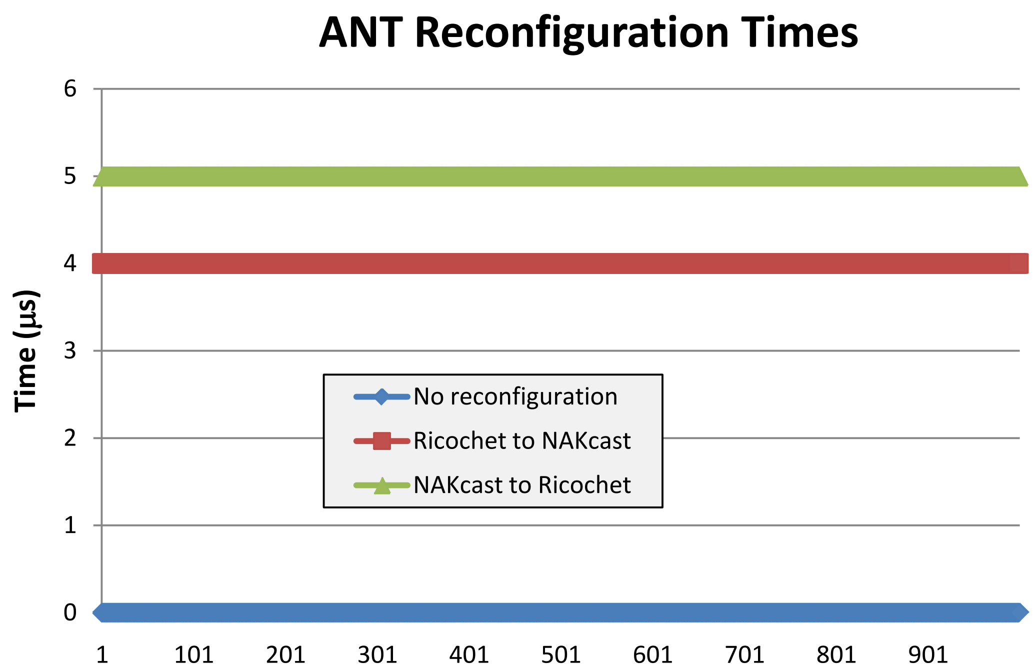


<u>TIML provides the predictable adaptation timeliness needed for DRE systems</u>

Empirical Results – ANT Timeliness

Experimental environment:

- 394 operating environments
- Emulab: 3 GHz CPU, 2GB of RAM, Fedora Core 6 w/ real-time patches
- Sub 10 μs response times for switching between NAKcast and Ricochet
- No jitter for all ANT reconfigurations



ADAMANT++ addresses the challenges of

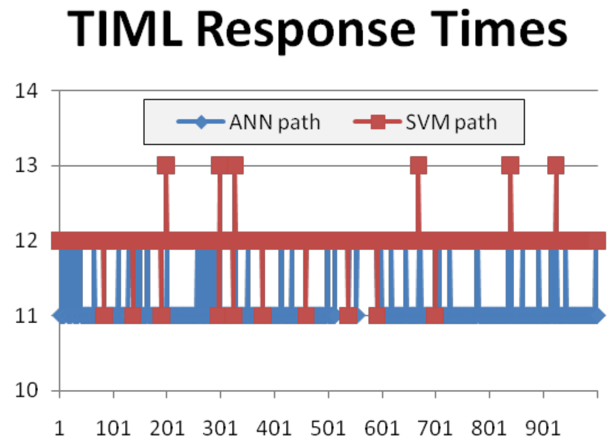
(1) Disseminating environment updates,

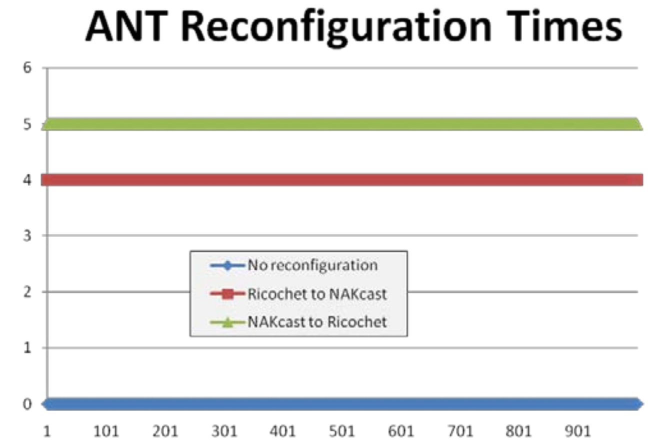
(2) Maximizing accuracy while maintaining timeliness

Proposed Experiment: ADAMANT & Dynamic Environments

Evaluation Criteria	Description
(H3) Provide bounded, constant time adaptation	Hypothesize that ADAMANT will adjust to new operating environment in bounded constant time (*i.e.*, O(1))

Leveraging equation-based machine learning and ANT, ADAMANT++ responds to new operating environments in constant time.





ADAMANT++ validates the three hypotheses from my qualifying exam:

(1) > 85% accuracy for known environments (achieved 100%),

(2) > 50% accuracy for unknown environments (achieved 86%),

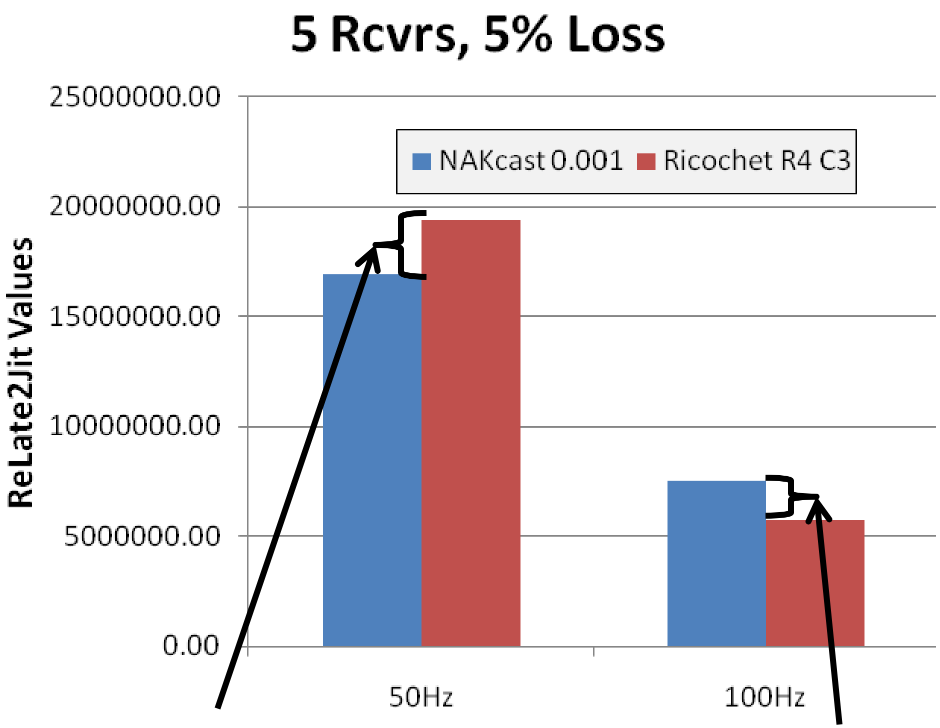
(3) Constant-time response

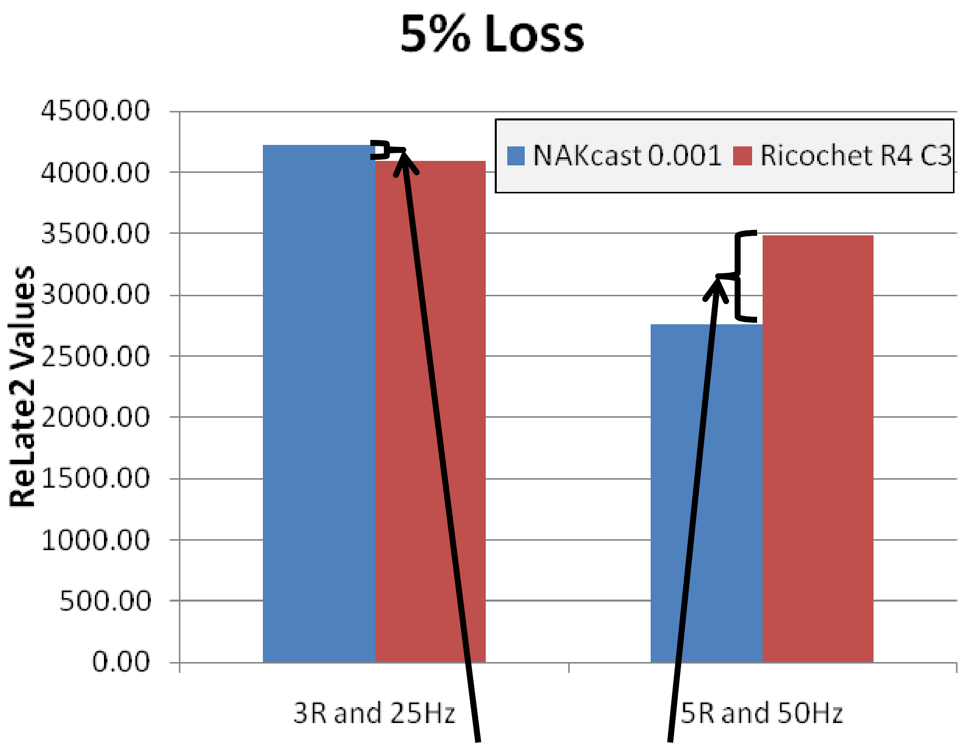
ADAMANT++ Sensitivity Analysis

Should adaptation ***always*** occur to get better QoS?

Are there times when the adjustment doesn't warrant the adaptation?

How can we analyze the value of adapting?





13% QoS Increase **24% QoS Increase** **3% QoS Increase** **21% QoS Increase**

Using threshold of 10% increase, we would reject adaptation for only 3% increase.

These decisions could be made at an application level & obeyed by ADAMANT++

Doctoral Research Contributions

Enhancing Productivity & Flexibility for QoS-enabled Pub/Sub DRE Systems

Focus Area	Challenge	Approach	Contribution
Valid QoS Design	• Design-time QoS Validation	• DSML that validates QoS configuration & generates implementation artifacts	• DQML
Evaluation of QoS Mechanisms	• Run-time QoS Guidance & Flexibility	• Pub/sub middleware with flexible protocol framework • Composite metrics & empirical analysis	• FLEXMAT • ReLate2 metrics
Autonomic Configuration for QoS	• Manage QoS in Flexible Environments	• Autonomic protocol config. in flexible resource envs • Timely adaptation based on supervised learning	• ADAMANT
Autonomic Adapation for QoS	• Manage QoS in Dynamic Environments	• Autonomic adaptation of protocols in dynamic envs • Increased accuracy via integration of supervised learning integration	• ADAMANT++ • TIML

www.dre.vanderbilt.edu/~jhoffert/research

Summary of Publications & Presentations

Journal Publications

1. Hoffert, J., Mack, D., & Schmidt, D. (2010) Integrating Machine Learning Techniques to Adapt Protocols for QoS-enabled Distributed Real-time and Embedded Publish/Subscribe Middleware, *International Journal of Network Protocols and Algorithms*, Vol. 2, No. 3.
2. Hoffert, J., Schmidt, D., & Gokhale, A. (2011) Evaluating Timeliness and Accuracy Trade-offs of Supervised Machine Learning for Adapting Enterprise DRE Systems in Dynamic Environments, (In submission to) *International Journal of Computational Intelligence Systems.*
3. Hoffert, J., Gokhale, A., & Schmidt, D. (2011) Autonomic Adaptation of Publish/Subscribe Middleware in Dynamic Environments, (In submission to) *International Journal of Adaptive, Resilient and Autonomic Systems.*

Conference Publications

4. Hoffert, J., Jiang S., & Schmidt, D. (2007, April). A Taxonomy of Discovery Services & Gap Analysis for Ultra-Large Scale Systems. *Proceedings of the 45th Annual Southeast Regional Conference*, Winston-Salem, NC
5. Hoffert, J., Schmidt, D., & Gokhale, A. (2007, June). A QoS Policy Configuration Modeling Language for Publish/Subscribe Middleware Platforms. *Proceedings of the Inaugural International Conference on Distributed Event-Based Systems*, Toronto, Canada.
6. Hoffert, J., Schmidt, D., & Gokhale, A. (2008, November). DQML: A Modeling Language for Configuring Distributed Publish/Subscribe Quality of Service Policies. *Proceedings of the 10th International Symposium on Distributed Objects, Middleware, & Applications*, Monterrey, Mexico.

First Author

Summary of Publications & Presentations (cont.)

Conference Publications (cont.)

7. Hoffert, J., Schmidt, D., & Gokhale, A. (2009, November). Evaluating Transport Protocols for Real-time Event Stream Processing Middleware & Applications. *The11th International Symposium on Distributed Objects, Middleware, & Applications*, Algarve, Portugal.
8. Hoffert, J. & Schmidt, D. (2009, July). Maintaining QoS for Publish/Subscribe Middleware in Dynamic Environments. *3rd ACM International Conference on Distributed Event-Based Systems*, Nashville, TN.
9. Hoffert, J., & Schmidt, D. (October, 2010). Evaluating Supervised Machine Learning for Adapting Enterprise DRE Systems, *International Symposium on Intelligence Information Processing and Trusted Computing*, Huanggang, China.
10. Hoffert, J., Schmidt, D., & Gokhale, A. (November, 2010). Adapting Distributed Real-time and Embedded Publish/Subscribe Middleware for Cloud-Computing Environments, *ACM/IFIP/USENIX 11th International Middleware Conference* , Bangalore, India.

Book Chapters

11. Hoffert, J., Schmidt, D., & Gokhale, A. Productivity Analysis for the Distributed QoS Modeling Language. *Model-Driven Domain Analysis & Software Development: Architectures & Functions.* Ed. Dr. Janis Osis & Dr. Erika Asnina, IGI Global.

First Author

Summary of Publications & Presentations (cont.)

Workshop Publications

12. Hoffert, J., & Schmidt, D. (2008, July). Supporting Scalability & Adaptability via Adaptive Middleware & Network Transports. *Proceedings of the OMG's Workshop on Distributed Object Computing for Real-time & Embedded Systems*, Washington, D.C., USA.
13. Hoffert, J., Schmidt, D., Balakrishnan, M., & Birman, K. (2008, September). Supporting Large-scale Continuous Stream Datacenters via Pub/Sub Middleware & Adaptive Transport Protocols. *Proceedings of the 2nd Workshop on Large-Scale Distributed Systems & Middleware*, Yorktown, NY.
14. Balakrishnan, M., Hoffert, J., Birman, K., & Schmidt, D., (2008, September). Rethinking Reliable Transport for the Datacenter. *Proceedings of the 2nd Workshop on Large-Scale Distributed Systems & Middleware*, Yorktown, NY.
15. Hoffert, J., & Schmidt, D. (2009, July). FLEXible Middleware & Transports (FLEXMAT) for Real-time Event Stream Processing (RT-ESP) Applications. *Proceedings of the OMG's Workshop on Distributed Object Computing for Real-time & Embedded Systems*, Washington, D.C., USA.
16. Hoffert, J., Mack, D., & Schmidt, D. (2009, December). Using Machine Learning to Maintain Pub/Sub System QoS in Dynamic Environments. *Proceedings of the 8th Workshop on Adaptive & Reflective Middleware*, Urbana Champaign, IL.
17. Hoffert, J., Schmidt, D., & Gokhale, A. (2010, April). Adapting and Evaluating Distributed Real-time and Embedded Systems in Dynamic Environments,1st International Workshop on Data Dissemination for Large scale Complex Critical Infrastructures, Valencia, Spain

First Author	Second Author

Summary of Publications & Presentations (cont.)

Poster Publications

18. Hoffert, J., Dabholkar, A., Gokhale, A., & Schmidt, D. (2007, March). Enhancing Security in Ultra-Large Scale (ULS) Systems using Domain-specific Modeling. *Spring 2007 Conference for Team for Research in Ubiquitous Secure Technology (TRUST)*, Berkeley, CA.
19. Hoffert, J., Gokhale, A. & Schmidt, D. (2007, September). QoS Management in Publish/Subscribe Systems using Domain-specific Modeling. *ACM/IEEE 10th International Conference on Model Driven Engineering Languages & Systems (MoDELS)*, Nashville, TN.
20. Hoffert, J., Schmidt, D., Balakrishnan, M., & Birman, K. (2008, April). Trustworthy Conferencing via Domain-specific Modeling & Low Latency Reliable Protocols. *Spring 2008 Conference for Team for Research in Ubiquitous Secure Technology (TRUST)*, Berkeley, CA.
21. Hoffert, J. (April, 2010). Evaluating and Adapting QoS for Distributed Real-time & Embedded Systems in Dynamic Environments, *EuroSys 2010 Conference*, Paris, France.

Tutorials

22. Hoffert, J. (October 2010). Intelligent Event Processing in Quality of Service-Enabled Publish/Subscribe Middleware, *The 2010 International Symposium on Intelligence Information Processing and Trusted Computing*, Huanggang, China,

First Author

Thank you for your time & attention.

Questions?

Soli Deo Gloria!