Challenges in Distributed Adaptation

Mark Yarvis
UCLA
March 2, 2000

http://fmg-www.cs.ucla.edu/Conductor
Outline

- Intro to Adaptive Networking
- Distributed Adaptation
- Conductor Architecture
- Challenges
Intro to Adaptive Networking

- Applications: increasingly network dependent
  - Internet radio/movies, gaming, MS Office
  - Thin clients
  - Internet appliances
- Minimum level of service assumed
Intro to Adaptive Networking

- Networks: not always fast and free
  - Bandwidth, latency, jitter, security, $$, reliability

- Applications should provide gracefully degraded service
  - Research focus: last mile
Achieving Graceful Degradation

• Use different applications
  – PalmOS clipping applications
• Write adaptive applications
  – Odyssey [Nob97], Rover[Jos95]
  – RealPlayer
• Adapt protocols within the network
Distilling the Web

Trade: Quality for transfer time
Link Scheduling

Trade: Latency for battery power
Other Forms of Adaptation

- Application layer
  - Distill, compress, encrypt, cache
- Network and transport layers
  - Link scheduling, prioritization, FEC
  - Snoop [Bal95]
State of the Art

- Daedalus [Fox98]
- Protocol Boosters [Mal97]
- Transformer Tunnels [Sud98]
- Focus:
  - Last mile
  - Independent adaptation
Beyond the "Last Mile"

• Leaf nodes become leaf networks
  – Home/office wired nets
  – Home/office wireless nets
  – Personal area nets
• User-to-user services
• Multi-hop networks
• Network/server congestion
Possible Approaches

- Solve end-to-end
- Single proxy node
- Independent solutions
- Distributed adaptation

- ✔
- ✗
- ✗
- ✔
Factors in Adaptor Placement

- Placement of adaptation is restricted by
  - Access to link status/control
  - Adaptation conflicts
  - Topology
  - Trust
  - Node resources
  - Load balancing
Conductor Design Goals

- Application-level, connection-oriented protocol adaptation
- Support heterogeneous networks
- Application transparent
- Automatic, but user controllable
- Arbitrary adaptations
- Easy to deploy adaptations
Conductor Architecture

• Adaptation framework
  – Transparent interception and routing
  – Node/link status monitoring
  – Distributed planning and deployment
  – Adaptor runtime environment

• Adaptor modules
  – Operate on data stream
  – Frequently paired
A Conductor-Enabled Node
Challenges in Distributed Adaptation

- Reliable Transmission
- Automated Planning
- Secure Adaptation
“A distributed system is one in which the failure of a computer you didn’t even know existed can render your own computer unusable”

— Leslie Lamport, May 1987
Reliable Transmission

- Distribution introduces new points of failure
- End-to-end reliability typically assumes data immutability
  - Retransmission by byte or packet count
- Adaptation modifies data in transit
  - Need a new unit of retransmission
<img low src=b.gif src=a.gif> Retransmit at byte 9 <img src=a.gif>
- Allows adaptors to express recovery
  • Maintained by segment combination
  end-to-end

- Preserve semantic meaning of each segment
  • Dynamically, based on data type and adaptation

- Divide stream into semantic units

- Meaningful unit of retransmission: a semantically

Reliable Transmission
Reliable Transmission

• Rules of segmentation
  – Constrain each stream modification to one segment
  – Combine segments where necessary
    • Not reversible
    • New segment contains combined semantic meaning
  – Final delivery of complete segments only
Reliable Transmission

- Service guarantees:
  - Transaction-like adaptation (all or nothing)
  - Exactly-once delivery of some form of each semantic element

- Other reliability models are possible
Challenges in Distributed Adaptation

- Reliable Transmission
- Automated Planning
- Secure Adaptation
Automated Planning

- **Goal:**
  - Select *which* adaptors and *where* to put them

- **Based on:**
  - Link characteristics
  - Node resources
  - Available adaptors
Automated Planning

- Distributed planning
  - fast
  - non-optimal results
Automated Planning

- Distributed planning
- Distributed planning with incremental refinement
  - How constraining is the initial plan?
- Centralized planning
  - Round trip for information gathering and plan distribution
Automated Planning

Node 1

Node 2
Plan

Node 3
Plan

1 2 3

Plan

Plan
Automated Planning
Automated Planning

• Feasible plans may be hard to find
• Large search space
  – # of problems, # of adaptors, # of nodes
  – Adaptor ordering and composition
• Many constraints
  – Node resources and trust
  – Adaptor composition
• Limited time!
Challenges in Distributed Adaptation

✔ Reliable Transmission
✔ Automated Planning
• Secure Adaptation
Secure Adaptation

- Protect the infrastructure
  - The usual mobile code issues
  - Java is good enough

- Protect the data
  - Integrity and secrecy (when needed)
  - Allow adaptation, but only authorized adaptation
Secure Adaptation

• Mechanisms
  – Select trusted nodes
    • Implicitly trust endpoints
    • Endpoints select other trusted nodes
  – Protect planning
    • Digitally sign planning messages
  – Protect data
    • Distribute session keys to trusted nodes
Secure Adaptation

- Need a verified public key for each node
Secure Adaptation

- Complications
  - Connections span administrative domains
  - No ubiquitous public key architecture
  - Each connection may require a different level of trust

- Pluggable authentication mechanism
  - Requires secure agreement of mechanism
Secure Adaptation
Secure Adaptation

- Authentication scheme proposed in plaintext and verified via signature
  - No node can change the authentication type without notice
- Public key encryption is used for session key distribution
- Additional mechanisms are needed to prevent replay
Challenges in Distributed Adaptation

- Reliable Transmission
- Automated Planning
- Secure Adaptation
Concluding Remarks

- Applications must be adaptive
- In heterogeneous networks applications benefit from distributed adaptation
- Key issues
  - Reliability, automatic planning, security
  - Automatic component composition
- Conductor, a prototype of proposed solutions
References


References


