FIND Projects 4

Radio Wormholes for Wireless Label Switched Mesh Networks
RNA: A Recursive Network Architecture
Sensor-Internet Sharing and Search (SISS)
Service-Centric End-to-End Abstractions for Network Architecture
Session Layer Management of Network Intermediaries

Munyoung Lee
mylee@mmlab.snu.ac.kr
2007.11.28
Contents

• Radio Wormholes for Wireless Label Switched Mesh Networks
  • University of Colorado
• RNA: A Recursive Network Architecture
  • USC/ISI
• Sensor-Internet Sharing and Search (SISS)
  • USC/ISI, UCLA
• Service-Centric End-to-End Abstractions for Network Architecture
  • UMass Amherst
• Session Layer Management of Network Intermediaries
  • Stevens Institute of Technology
Radio Wormholes for Wireless Label Switched Mesh Networks

• **Title**
  – Radio Wormholes for Wireless Label Switched Mesh Networks

• **PIs**
  – Dirk Grunwald, Douglas C Sicker, Tim Brown and Peter Mathys

• **Institution**
  – University of Colorado

• **Project Wiki**
Radio Wormholes for Wireless Label Switched Mesh Networks

• Problem of Existing Mesh Network
  – Share properties of fixed multipoint wireless and ad hoc wireless networks → lower throughput, latency
  – No Quality of Service
  – No Traffic engineering

Decrease in bandwidth due to channel contention when using a omnidirectional antenna with different packet sizes (reflecting the impact of overhead)
Radio Wormholes for Wireless Label Switched Mesh Networks

- Approach
  - Building radio circuits that use cut through switching similar to optical lambda switching
  - Orthogonal frequency multiple access (OFDMA) serves as the underlying PHY layer for frequency switching
  - Adopt the network management and traffic engineering mechanisms implemented in Multiprotocol Label Switching (MPLS) and Generalized MPLS (GMPLS)

→ Proposed network design uses an existing PHY layer (OFDM) and an existing management plane (GMPLS)
Radio Wormholes for Wireless Label Switched Mesh Networks

- Cut through / Wormhole routing
Radio Wormholes for Wireless Label Switched Mesh Networks

• Schematic Diagram of a Wireless Label Switch
  - MPLS and GMPLS
    • Establish label switch paths (LSPs)
    • Those LSPs can operate on a variety of “labels” to reduce the routing and switching delay

Per-hop delay in label switch path
Radio Wormholes for Wireless Label Switched Mesh Networks

- Progress
  - Building a prototype hardware platform to validate this project. The prototype is based on an FPGA based software radio and a prototype 2.4Ghz RF front-end.

  The radio can transmit or receive in a 35Mhz band.

  The FPGA sampler board (a Nallatech XtremeDSP-II board) can transmit across a 80Mhz band and receive a ~50Mhz band.
Radio Wormholes for Wireless Label Switched Mesh Networks

- Expected improvement
  - Reduce communication latency
  - Improve throughput
  - Quality of service
  - Traffic engineering
RNA: A Recursive Network Architecture

• **Title**
  - RNA: A Recursive Network Architecture (One Protocol / Many Layers)

• **PI**
  - Joe Touch

• **Institution**
  - University of Southern California / ISI

• **Project Wiki**
  - http://www.isi.edu/rna/
RNA: A Recursive Network Architecture

• Motivation
  – Layers of a stack becoming more similar
    • Functions are recapitulated
    • Security, soft-state, retransmission
  – Not useful to implement all of these services at every layer of a protocol stack
  – Desire to support interlayer cooperation
    • Message boundary, congestion control
RNA: A Recursive Network Architecture

• Approach
  – Services are relative
    • E.g. Data formatting at link and presentation layer
    • No service is specific to a particular layer
    • Same protocol, with a variety of services, suffices at any layer
  – A template can avoid recapitulation

→ Use a single metaprotocol as a generic protocol layer
RNA: A Recursive Network Architecture

• **RNA Metaprotocol**
  - Generic protocol layer
  - Template of basic protocol service
  - Includes a number of basic services
    • Security, retransmission
  - Includes interlayer coordination
    • Identity management at various layers
    • Couple flow control

- Provide the building block
- Service is defined by context
  • Shared state exists at many layers, but supports different services
RNA: A Recursive Network Architecture

- **Features of RNA**
  - Each layer of the stack is an instance of the metaprotocol.
  - Consider four metaprotocol layers above a physical layer.
    - mp-1 might employ retransmission if the physical layer were wireless.
    - mp-1 might not employ if it were SONET (Synchronous Optical NETwork).
  - mp-2 might employ its own level of retransmission if any of the intermediate mp-1 hops had retransmission.

- **Layer Context Sensitivity**
  - Layers of the metaprotocol are thus context dependent, but based on the same protocol.
RNA: A Recursive Network Architecture

• Expected improvement
  – Reusing basic protocol operations across different protocol layers to avoid reimplementation
  – No service is specific to a particular layer
    • Same protocol, with a variety of services, suffices at any layer
Sensor-Internet Sharing and Search (SISS)

- **Title**
  - Sensor-Internet Sharing and Search (SISS)

- **PIs**
  - John Heidemann, Mark Hansen, Junghoo Cho

- **Institutions**
  - USC/ISI, UCLA

- **Project Wiki**
Sensor-Internet Sharing and Search (SISS)

• Motivation
  – Current sensor network
    • Operate in isolated patches
    • Use different mechanisms to deliver data to their users
    • Often have no formal methods to share data with others.
  – Future..
    • Large sensornets: formed of multiple wireless patches bridged by the Internet
    • Individual sensors: deployed by casual users or citizen scientists, with their data posted to sensor web logs or slogs.

  – It is increasingly becoming important to enable a common means to share data, preferably using the Internet as a communication platform
Sensor-Internet Sharing and Search (SISS)

• Approach
  – Build architectures and create protocols to enable users to
    • discover
    • process
    • republish data
    from thousands of independently operated sensors.
  – Architecture Components
    ▶ sensors (clustered into sensornets)
    □ sensor publishers that act as gateway
    ▪ sensor stores
    ☺ sensor search engines
    Connected using a common protocol: Sensor Data Stream Protocol (SDSP)
Sensor-Internet Sharing and Search (SISS)

- Sensors and Publishers
  - Sensors are gathered into sensornets, clusters of sensors that share a wireless network
  - Sensor publishers link sensornets to the Internet

- Sensor Stores
- Sensor Search Engines
  - Index and search sensor data
  - Ranking based on popularity

- Sensor Data Streaming Protocol
  - Link publishers to sensor stores and search engines
Sensor-Internet Sharing and Search (SISS)

- **Progress**
  - **Prototypes**
    - SensorBase (sensorbase.org)
      - Portal to sensor data that is used by over 50 projects
    - Sensor Search
      - Experimenting with providing a sensor search for SensorBase
Sensor-Internet Sharing and Search (SISS)

• Expected improvement
  – Define a new network architecture that blends sensor networks and the Internet
  – Enabling shared scientific sensor data and citizen-scientist-operated data
Service-Centric End-to-End Abstractions for Network Architecture

- **Title**
  - Service-Centric End-to-End Abstractions for Network Architecture

- **PI**
  - Tilman Wolf

- **Institutions**
  - UMass Amherst

- **Project Wiki**
  - http://www.ecs.umass.edu/ece/wolf/project_architecture.html
Service-Centric End-to-End Abstractions for Network Architecture

• Motivation
  – Network itself was kept relatively simple and provided just basic communication between the end-systems
    • Most of the complexity is implemented on the end-systems (e.g., retransmission of lost packets, congestion control based on end-system observations).
  – Next-generation networks will have to offer a much broader range of services
  – End-to-end argument poses a considerable limitation
    • Flexibility of the network architecture to adapt to new requirement
Service-Centric End-to-End Abstractions for Network Architecture

• Approach
  – Processing of data not limited to end-systems anymore
  – Data processing throughout the network
    • Network can provide best setup for end-to-end communication
    • Requires processing capabilities throughout the network
Service-Centric End-to-End Abstractions for Network Architecture

- Information Transfer and Data Services (ITDS) Architecture

- Data Services
  - Reliability, Privacy, Congestion control
  - Caching, Security, QoS, Multicast
Service-Centric End-to-End Abstractions for Network Architecture

• Examples of ITDS Scenarios
  – Services can be combined for end-to-end information transfer

• Example 1:
  – Reliable and private communication

• Example 2:
  – Content Distribution and Transcoding
Service-Centric End-to-End Abstractions for Network Architecture

- Progress
  - Prototype
    - Implemented a prototype of the network-service architecture
      - Support reliability, compression/decompression, and encryption/decryption.
  - Publications
Service-Centric End-to-End Abstractions for Network Architecture

• Expected improvement
  – Processing of data not limited to end-systems anymore
  – Data processing throughout the network
  – Semantics of information exchanges allows the network to provide data services that further improve the end-to-end communication
Session Layer Management of Network Intermediaries

- **Title**
  - Session Layer Management of Network Intermediaries

- **PI**
  - Daniel Duchamp

- **Institutions**
  - Stevens Institute of Technology

- **Project Wiki**
  - http://www.cs.stevens.edu/~djd/find/
Session Layer Management of Network Intermediaries

• Motivation
  – Original Internet: Homogeneous Internet
    • Intelligent hosts at edges of dumb routing fabric
  – In recent years both ISPs and end systems have deployed various forms of network *intermediaries*
    • Proxies, NATs, web caches, firewalls
  – The result of these deployments is a more complicated discrete Internet
    • Need a novel technical mechanism

• Approach
  – New definition of the *session layer*
    • Allows endpoints to become aware of and manage intermediate services
    • Session can survive a change to the underlying transport parameters
    • Verifying end-to-end delivery & offering a variety of semantics
Session Layer Management of Network Intermediaries

• Approach
  – Current method
    • Single end-to-end transport (L4) connection intercepted transparently by the intermediary.

  – Proposed method
    • End-to-end layer 5 (L5) session runs on top of one or more transport (L4) connections
    • Intermediaries explicitly addressed, each terminates an L4 connection
Session Layer Management of Network Intermediaries

- Expected improvement
  - Cleaner programming model for both applications and intermediate services
  - Reduced congestion and greater bandwidth utilization for the Internet as a whole
  - Improved performance of the individual session
Conclusion

• Radio Wormholes for Wireless Label Switched Mesh Networks

• RNA: A Recursive Network Architecture

• Sensor-Internet Sharing and Search (SISS)

• Service-Centric End-to-End Abstractions for Network Architecture

• Session Layer Management of Network Intermediaries
Thank you!
RNA: A Recursive Network Architecture

• RNA Metaprotocol
  – Generic protocol layer
  – Template of basic protocol service
  – Includes a number of basic services
  – Includes interlayer coordination
    • Identity management at various layers
    • Couple flow control.
Service-Centric End-to-End Abstractions for Network Architecture

• Tilman Wolf

• NeTS-NBD: Packet Spacing in Small-Buffer Networks
• http://www.nsf.gov/awardsearch/showAward.do?AwardNumber=0721790
• Service-Centric End-to-End Abstractions for Network Architecture
  Tilman Wolf, UMass Amherst

- Data semantics
  - Reliability
  - Privacy
  - Congestion control
  - Caching
  - Security
  - qos
  - Multicast
  - transcoding

- Information Transfer and Data Services Layer
  - processing
  - processing
  - processing
  - processing
(a) Example I: Reliable and Private Point-to-Point Communication

(b) Example II: Web Caching

(c) Example III: Content Distribution and Transcoding