Open Distributed Networking Intelligence: A New Java Paradigm

Tal Lavian
Agenda

- **Openness** - Virtual community development, Domain experts
- **Architecture** and technology concepts
- **Intelligent** distributed applications
- **Demos:**
  - JEND’s JVM’s, ORE
  - Java Oplet
  - XML on router
- **Summary**
Vision

- Transfer Intelligence to network devices
- Programmable network devices
- Virtual community of developers
- Domain experts bundle Intelligent network devices as part of their solutions
- New era of innovation for startups
Change the Name of the Game

- Openness
- In the current routing market we have to change the name of the game
- It will enable technology leapfrog
- Programmable devices creates a Paradigm shift
- Open Architecture is an excellent technology and direction
- Request: Open Architecture
- Java appeals to the market & customers
Openness
Why an Open Architecture?

- Unparalleled customization capabilities
- Dynamic delivery of new services
- The network adapts itself to users needs, not the other way around
- 3rd party developers - create innovative services and applications
- HW & OS independent. can migrate to new HW and SW => Investment protection
Benefits

- Network PC-ification
- Intelligent vs Dumb Devices
- Open vs Proprietary
- 1,000,000 Java programmers
- Revolutionize the network
- IEEE 1520 - programmable networks: Service Providers, Enterprises, request programmability
Evolution vs. Revolution

Performance / Enhancement

Logarithmic

Time

Open Architecture

Evolution

Revolution

HW

Net SW

Intelligence Dynamic

Now

Net

10M

100M

1G

10G

E-Commences

XML

Last Mile

56k --> 10Mb

Static Net SW

Web

Performance / Enhancement
The Web changes everything

Need for programmable Intelligent network

- Large demand bandwidth from web and multimedia applications
- Huge LAN bandwidth compared to limited WAN bandwidth
- New direction of networked applications
- Global village - distributed business and computation environment
- More business processes become network-centric and web-centric
The new networked world

Need for Intelligent Network utilizing the network capabilities

Today, the network is used as transparent media
Users need a new kind of ‘net

- Personal networking
- Accelerating business velocity
- Mass customization
- **Enabled by a new infrastructure**

Need for device programmability:
- Distributed apps
- E-commerce
- Web centric
- Business apps

And a new kind of thinking...
Benefits

- Breakaway business strategy for creation of value over network elements
- Positioning of open standards versus proprietary designs - market acceptance
- Faster TTM for software development
- Unparalleled support and maintenance capabilities
- PC-ification - broad adoption
Community Openness

- Success stories by large community of developers

- Net-Based developers’ communities
  - Linux, GNU, Apache, BSD, X-Windows, Perl, Tk/Tcl
  - Netscape browser, NFS, JDK, JVM

- Linux success:
  - Compaq, HP, IBM, SUN and SGI
  - Intel, Sparc64, Alpha, PowerPC

- The Web Changes everything
  - Java, XML, E-Business
Technology
Technology Concept
“Reversed Applets”

Technology is based on the concept of Reversed-Applets

The JVM is in the Browser

The JVM is in the Device

Java Beans
Oplets
Reversed Applets

Web Server
Applet
Web Browser

Open Architecture
Why Aren’t The Current Interfaces Sufficient?

- There are two main management interfaces to most devices:
  - SNMP
  - CLI

- Web/HTTP is typically just a MIB viewer/manipulator
  - “Collapsed SNMP”
SNMP Is Data Driven

- SNMP = MIB + side effects + get/set protocol
- SNMP is not executable
  - As defined, you can’t really script it
  - You can script things like Perl and have them do SNMP for you
- You can only manipulate MIB objects that are supported
- SNMP apps can’t run on the device itself
  - Requires separate server
- SNMP requires polling
  - Eats up net bandwidth
- Example: You can’t write a custom CLI login authentication module easily using SNMP
  - Perhaps you could with a bunch of traps and such but it would be really ugly
SNMP and CLI Summary

- SNMP is not a Turing Machine
- CLI could be made into a Turing Machine, but the implementation is ugly and you still couldn’t implement all that you wanted very easily
Desired Solution

Router Run-time

<table>
<thead>
<tr>
<th>Customer-specific functionality</th>
<th>VAR provided functionality</th>
<th>extension functionality</th>
<th>3rd party functionality</th>
</tr>
</thead>
</table>
Possible Customer/VAR Applications

- Advanced SNMP monitoring and trap generation
  - Eliminate polling, generate custom traps or system log messages
- “Pluggable” remote authentication systems
  - People want different things: LDAP (who’s schema), Security Dynamics, TACACS, etc.
- Custom CLI commands or web management screens
- Custom accounting interface
- Custom asset management
- Generated MIB extensions for SNMP
- Runtime image file distribution for upgrades
- VARs can add a lot of system value tying things together
Possible Applications

- INM can download advanced network management functions dynamically
  - Might download a diagnosis module after determining a fault
- In-field upgrades
  - New algorithms
  - Dynamic downloads via the Internet
- Custom “glue” written by different divisions to create a more cohesive solution
Dynamic ORE Adapters and Oplets

ORE Manageable Service Beans
Enabling New Concepts
Java MIB API - Proxy Mode

- Uses SNMP loopback mechanism to target a remote network element
- API can be used to control devices that don’t have an embedded JVM
Collaboration with Business Applications

- New paradigm of distributed applications
- Network devices collaborating with business applications

Oracle DB
E-Commerce
Business apps
SAP, ERP, Optivity

RMI, XML, CORBA

Accelar
Passport
BayRS
Mobile Agents

- Jumping Java Beans across the network
Network Devices Collaboration

- Distribution application across network devices

Diagram showing network devices such as Accelar, Passport, and PBX, connected through JVM + ORE and using RMI, CORBA, and XML protocols.
Summary

- **Openness** - successfully proven paradigm
- **Domain experts** - virtual developers community
- Allows **innovations** and added value
- **Dynamic** Loading
- **Dynamic** agents vs **static** agents
- **Strong** Security
- New capabilities - **XML** example
- An **enabling-technology**
Appendix A: Strong Security in the New Model

- The new concept is secure to add 3rd party code to network devices
  - Digital Signature
  - “Certified Oplet”
  - No access out of the JVM space
  - No pointers to damage the work
  - Access only to the published API
  - Verifier - only correct code can be loaded
  - Class loader access list
    - Different Oplets with different access levels
  - JVM has run time bounds, type, and executing checking
Old model Security (C++)

- Old model - Not secure to add 3rd party code
  - Not recommended to add 3rd party code to network devices
  - Dangerous, C/C++ Pointers
    - Can touch sensitive memory location
  - Risk: Memory allocations and free
    - Allocation without freeing
    - Free without allocation (core dump !!!! )

- Limited security in SNMP
Appendix B: Java SNMP MIB API

- Portable across a range of network devices
- Extendible
- Simple and convenient for client use
- Consistent with SNMP model
- Hides unnecessary SNMP details
- Permits optimized access
- Re-use MIB documentation
MIB API Generation

- Most of the Java code is generated automatically
- ASN.1 MIB definitions are converted into Java classes
- Documentation and commentary in the MIB definitions is placed as Javadoc formal comments
- HTML documentation generated from Javadoc
MIB Objects

- The MIB data model is structured as a tree
- API represents MIB groups with Java classes
- MIB variables are represented with accessor methods
- Conceptual tables are represented with iterators
- API converts SNMP data values into standard Java types
API uses a MIB Map to dispatch requests to variable access routines
• Different parts of the MIB tree can be serviced by different mechanisms
• Two main schemes:
  • An ad hoc interface to the SNMP instrumentation layer
  • A generic SNMP loopback
Advantages of MIB map

- Allows immediate generic implementation of the entire MIB via the loopback scheme
- Enables optimized native implementation of key MIB variables for maximum efficiency
- Permits definition of pseudo-MIB variables for extending MIB dynamically
- Provides site for centralized access management