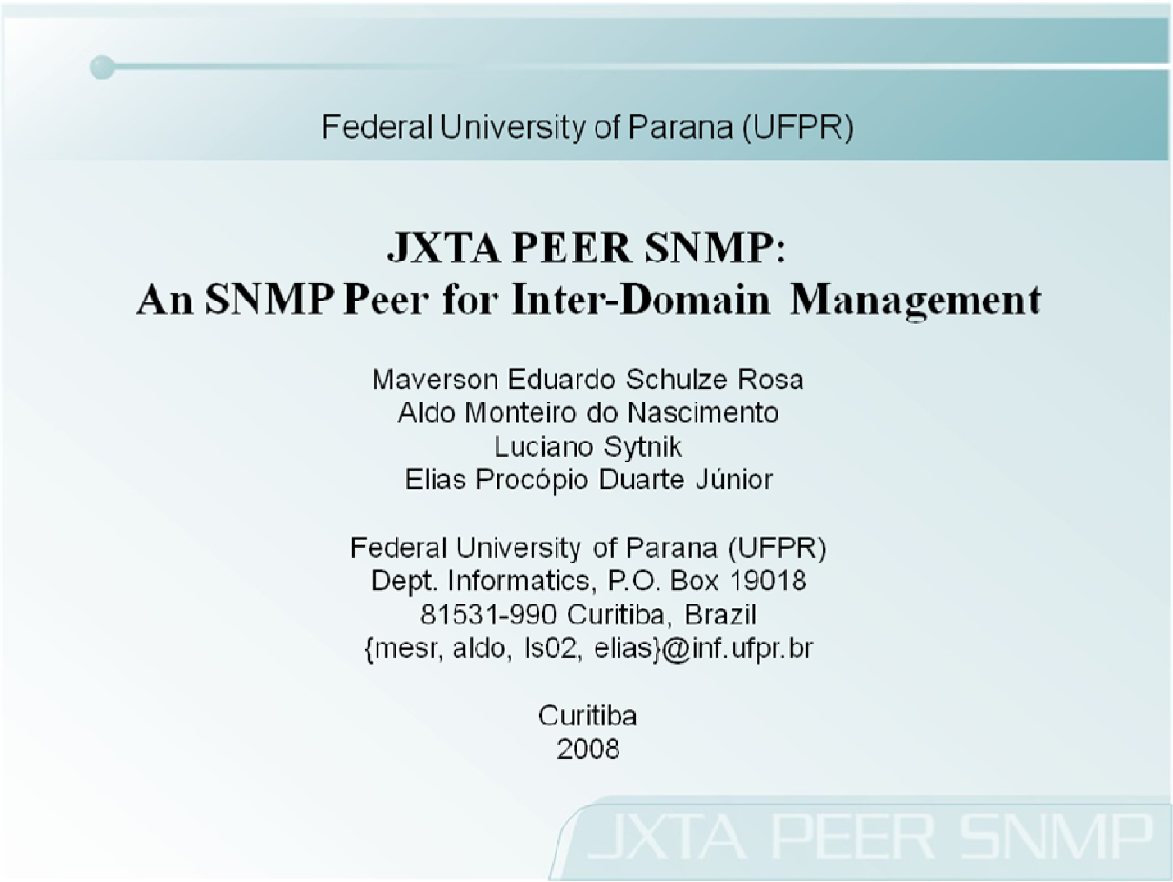


Federal University of Parana (UFPR)

JXTA PEER SNMP: An SNMP Peer for Inter-Domain Management

Maverson Eduardo Schulze Rosa
Aldo Monteiro do Nascimento
Luciano Sytnik
Elias Procópio Duarte Júnior

Federal University of Parana (UFPR)
Dept. Informatics, P.O. Box 19018
81531-990 Curitiba, Brazil
{mesr, aldo, ls02, elias}@inf.ufpr.br

Curitiba
2008

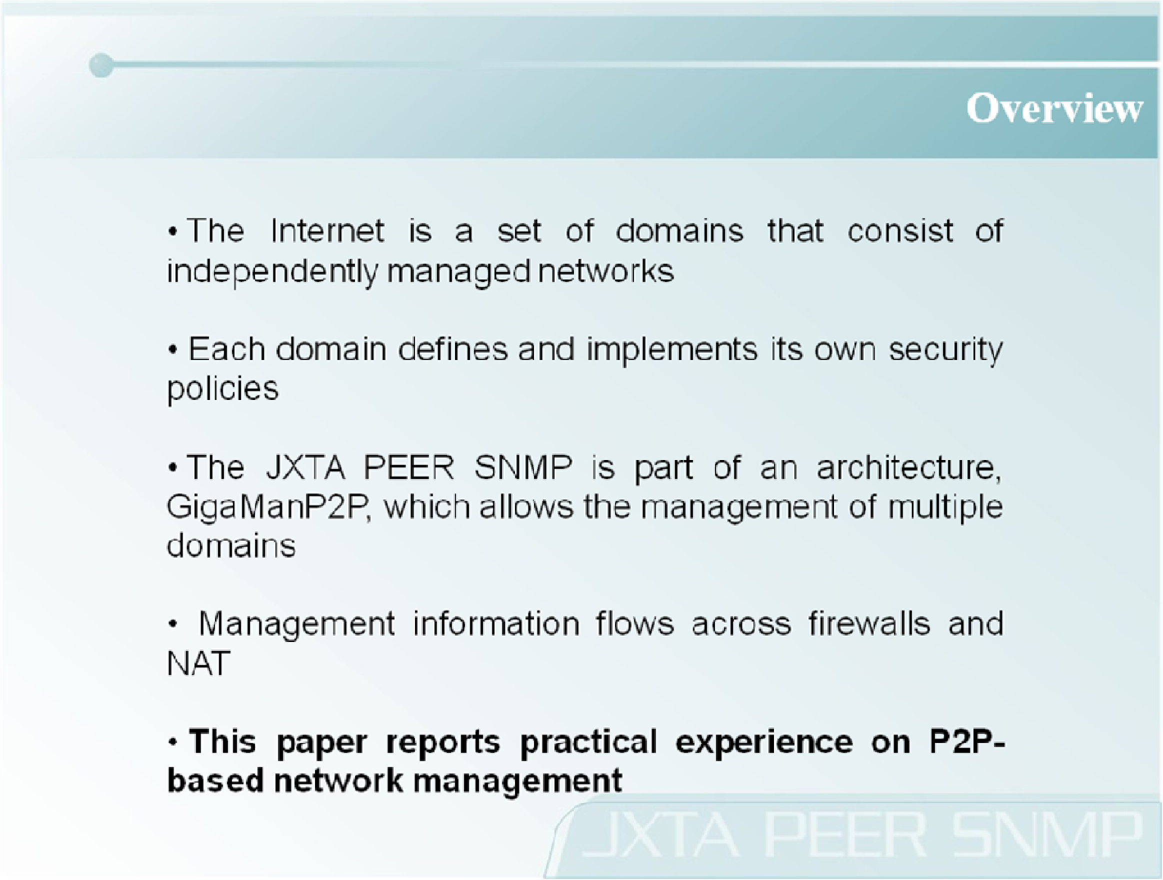
JXTA PEER SNMP

Keywords: Network Management, P2P Networking, QoS Management, Inter-Domain Management.

Abstract

The Internet standard Simple Network Management Protocol (SNMP) presents challenges when the monitored system consists of multiple domains or Autonomous Systems. These independent systems are protected by firewalls and other security devices. This problem can be solved using SNMP over a Peer-to-Peer (P2P) network, which allows the communication of management information generated with SNMP, even among autonomous systems. This work describes the implementation of a management peer, JXTA PEER SNMP, and the construction of secure peer groups within the GigaManP2P framework. The management peer was implemented with the JXTA P2P platform, using both the Net-SNMP agent and NetSNMPj management applications. Experimental results comparing the performance of JXTA PEER SNMP and Net-SNMP's native implementation are presented.

978-1-4244-2066-7/08/$25.00 ©2008 IEEE



Overview

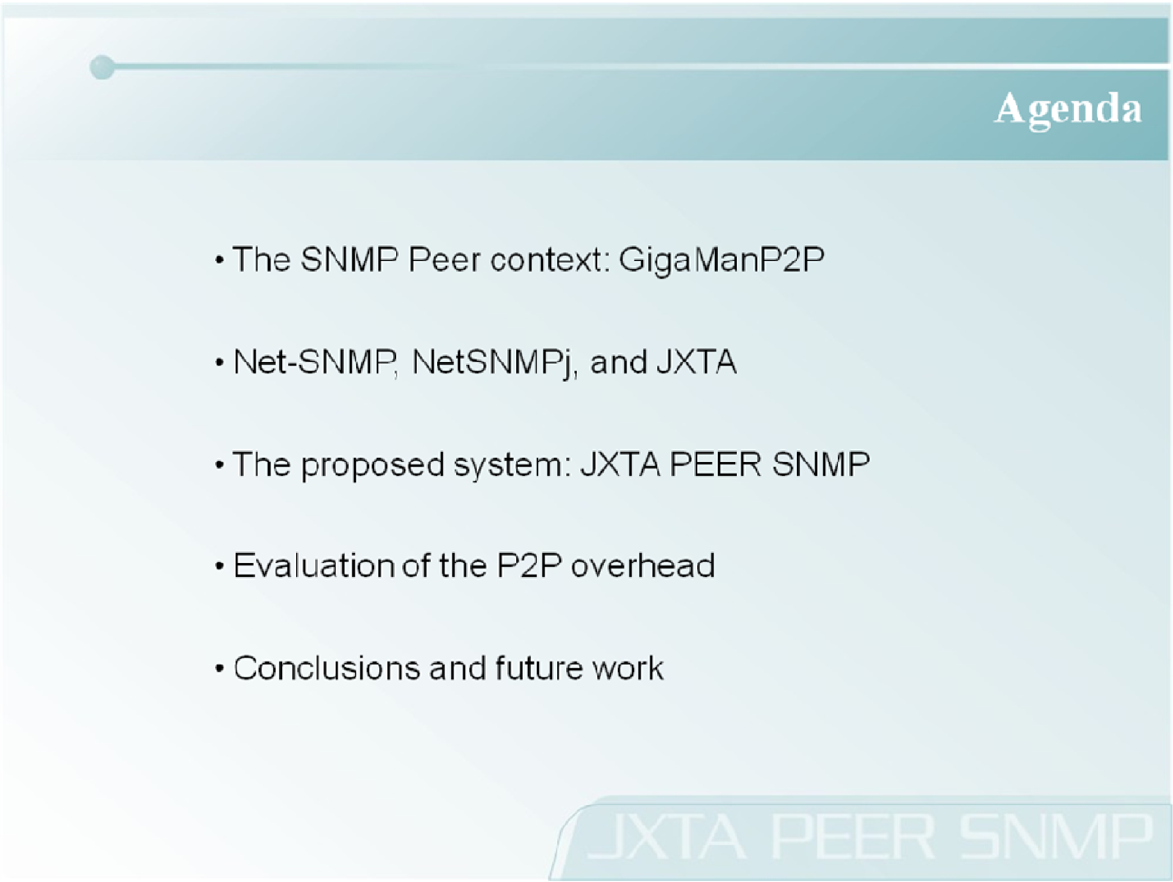
- The Internet is a set of domains that consist of independently managed networks
- Each domain defines and implements its own security policies
- The JXTA PEER SNMP is part of an architecture, GigaManP2P, which allows the management of multiple domains
- Management information flows across firewalls and NAT
- **This paper reports practical experience on P2P-based network management**

The Internet is a set of Autonomous Systems (AS's) which consists of independently managed networks. Monitoring and controlling resources that are shared by several AS's is often a difficult task, due to the fact that each AS defines and implements its own security policies.

Recently, a management architecture called GigaManP2P [6] was proposed to allow the management of optical backbones that span several AS's. GigaManP2P allows the execution of tasks, such as QoS configuration, that requires the system to be configured as a whole, and not as a set of independent pieces.

Several approaches have been proposed to implement AS resource management with P2P technology. P2P allows the construction of distributed systems that are not based on the traditional client/server architecture: each peer behaves, at the same time, as client and server.

In this work the implementation of an SNMP peer, called JXTA PEER SNMP, is presented. This is a management peer developed for the GigaManP2P architecture, which allows devices on multiple AS's to be monitored and configured from any point of the network, even if security devices such NAT (Network Address Translation) and firewalls are employed.



Agenda

- The SNMP Peer context: GigaManP2P
- Net-SNMP, NetSNMPj, and JXTA
- The proposed system: JXTA PEER SNMP
- Evaluation of the P2P overhead
- Conclusions and future work

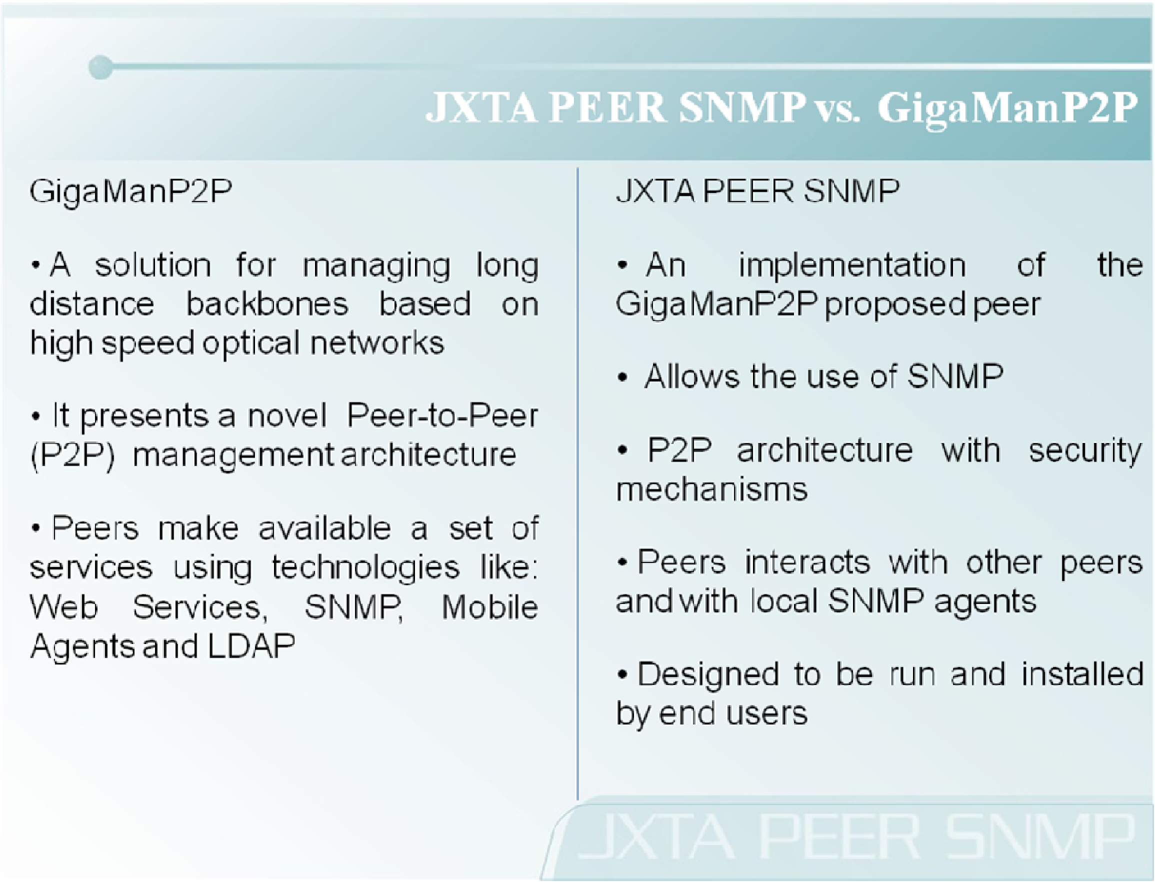
JXTA PEER SNMP

This paper initially presents GigaManP2P, the framework for which the proposed JXTA PEER SNMP was developed. An overview of technologies such as Net-SNMP, NetSNMPj, and JXTA is then given.

A section describing JXTA PEER SNMP follows, which defines of the structure of the proposed peer, its communication facilities, and some implementation details.

An evaluation based on experimental results of the overhead imposed by the P2P infrastructure on SNMP is then given.

Finally the conclusions and future works are presented.



JXTA PEER SNMP vs. GigaManP2P

GigaManP2P

- A solution for managing long distance backbones based on high speed optical networks
- It presents a novel Peer-to-Peer (P2P) management architecture
- Peers make available a set of services using technologies like: Web Services, SNMP, Mobile Agents and LDAP

JXTA PEER SNMP

- An implementation of the GigaManP2P proposed peer
- Allows the use of SNMP
- P2P architecture with security mechanisms
- Peers interacts with other peers and with local SNMP agents
- Designed to be run and installed by end users

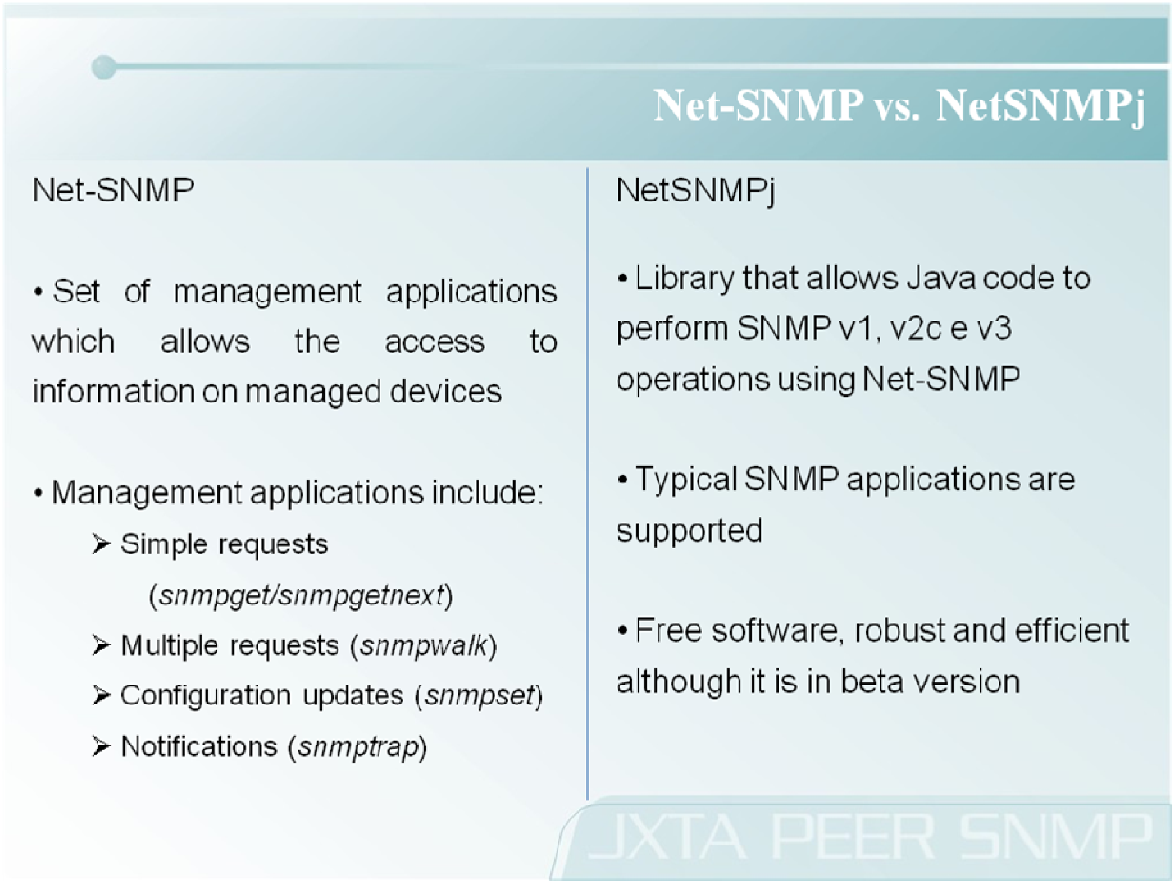
JXTA PEER SNMP

The GigaManP2P architecture [6] is a management solution that is responsible for deploying the requirements of user applications on the set of facilities offered by the underlying optical networks. With GigaManP2P it is possible to monitor and control resources from any point of the network.

GigaManP2P presents a novel P2P management architecture to overcome the hindrances posed by the use of different security policies in the various AS's covered by the backbone. In this architecture, each peer uses a series of technologies including Web Services, SNMP, Mobile Agents, among others.

JXTA PEER SNMP is an implementation of an SNMP peer for the GigaManP2P architecture. JXTA PEER SNMP interacts both with local SNMP agents in its AS, and with other JXTA peers, which can be run either within or outside its own domain. The proposed implementation guarantees the secure communication among peers, allows management information to flow across domain boundaries, and provides facilities for effectively configuring and monitoring devices that belong to several domains.

The main goals of the proposed implementation are to provide easy installation, configuration and usage by end users. It is developed according to the "plug and play" idea, i.e, it is not necessary adjust numerous parameters to run the peer.



Net-SNMP vs. NetSNMPj

Net-SNMP

- Set of management applications which allows the access to information on managed devices
- Management applications include:
  - Simple requests (*snmpget/snmpgetnext*)
  - Multiple requests (*snmpwalk*)
  - Configuration updates (*snmpset*)
  - Notifications (*snmptrap*)

NetSNMPj

- Library that allows Java code to perform SNMP v1, v2c e v3 operations using Net-SNMP
- Typical SNMP applications are supported
- Free software, robust and efficient although it is in beta version

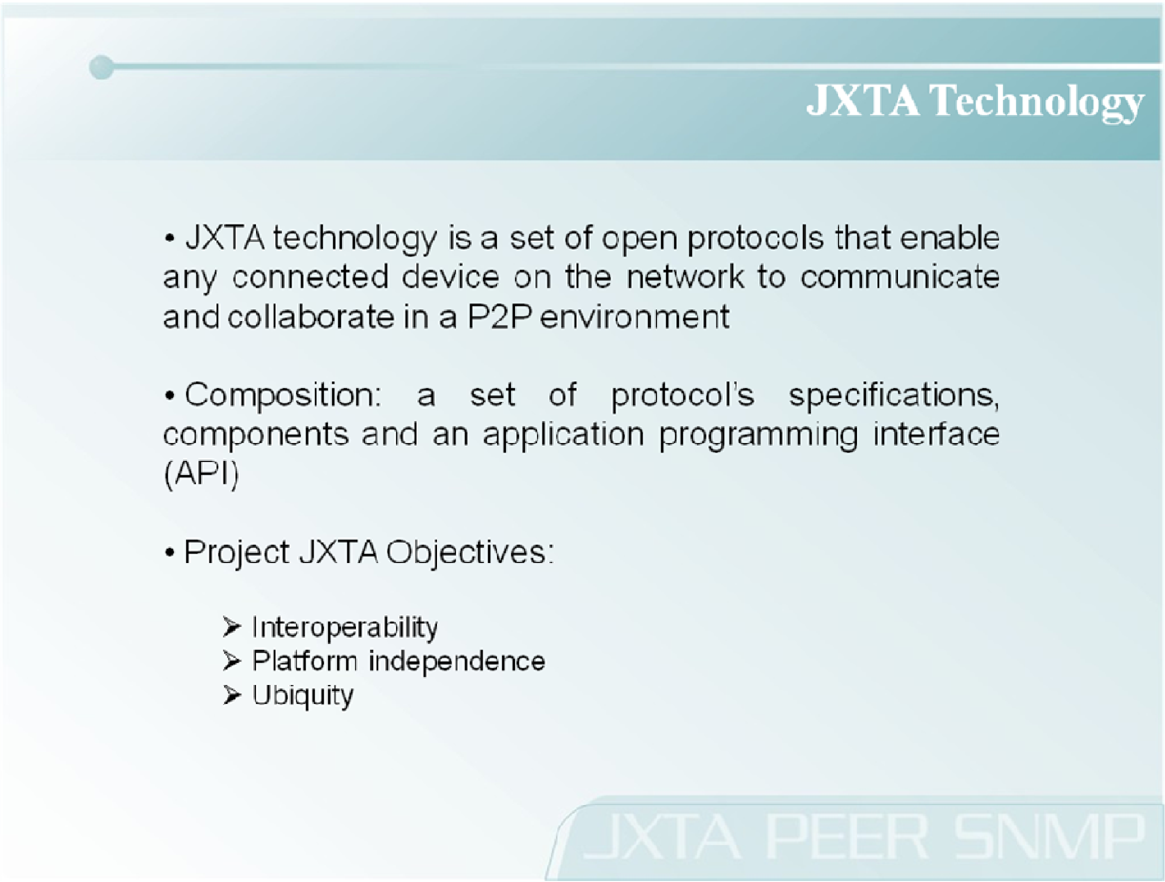
JXTA PEER SNMP

The Simple Network Management Protocol (SNMP) is a widely used framework for monitoring network equipment (eg. routers), computer equipment and even simple devices. Net-SNMP [2] is a suite of applications used to implement SNMP v1, SNMP v2c and SNMP v3 using both IPv4 and IPv6.

The suite includes applications to: retrieve information from an SNMP-capable device, either using single requests (*snmpget*, *snmpgetnext*), or multiple requests (*snmpwalk*, *snmptable*, *snmpdelta*); manipulate configuration information on an SNMP-capable device (*snmpset*); retrieve a fixed collection of information from an SNMP-capable device (*snmpdf*, *snmpnetstat*, *snmpstatus*); a daemon application for receiving SNMP notifications (*snmptrapd*); among others.

NetSNMPj [3] is an open source Java library that allows java code to perform SNMP v1, v2c and v3 operations using the NET-SNMP applications. It is composed of both java and native code elements being currently supported in Windows, Linux and Solaris(sparc). It is also developed as open source software.

NetSNMPj is a Java library that allows the execution of SNMP operations through a Java code. However, there are other alternatives to this task, like SNMP4J [4]. Net-SNMP needs to be installed on the device which will run the JXTA PEER SNMP. The SNMP agent can be installed in any device reachable to the JXTA PEER SNMP device.



JXTA Technology

- JXTA technology is a set of open protocols that enable any connected device on the network to communicate and collaborate in a P2P environment

- Composition: a set of protocol's specifications, components and an application programming interface (API)

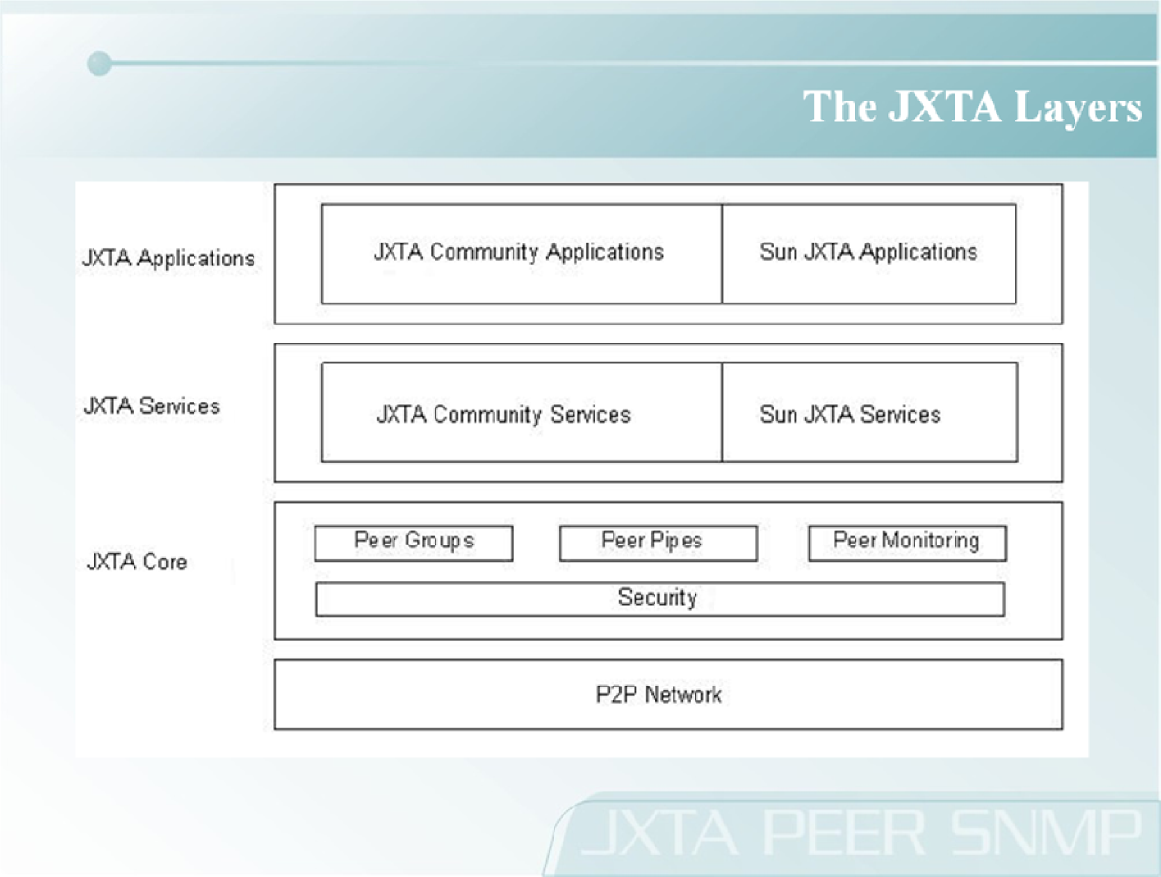
- Project JXTA Objectives:
  - ➤ Interoperability
  - ➤ Platform independence
  - ➤ Ubiquity

JXTA PEER SNMP

Introduced by Sun Microsystems, Inc., the JXTA technology [1] is a set of open, generalized peer-to-peer protocols, components and its application programming interfaces (API's) that allows any connected device on the network to communicate and collaborate in a P2P environment.

JXTA peers create a virtual network where any peer can interact with other peers and resources directly, even when some of the peers and resources are behind firewalls and NAT's.

The project JXTA objectives aims: the interoperability across different peer-to-peer systems and communities; platform independence, in which multiple/diverse languages, systems, and networks can be used to deploy a unique system; ubiquity, i.e., it can be run on any IP (Internet Protocol) host.



The JXTA Layers

JXTA Applications

JXTA Community Applications

Sun JXTA Applications

JXTA Services

JXTA Community Services

Sun JXTA Services

JXTA Core

Peer Groups

Peer Pipes

Peer Monitoring

Security

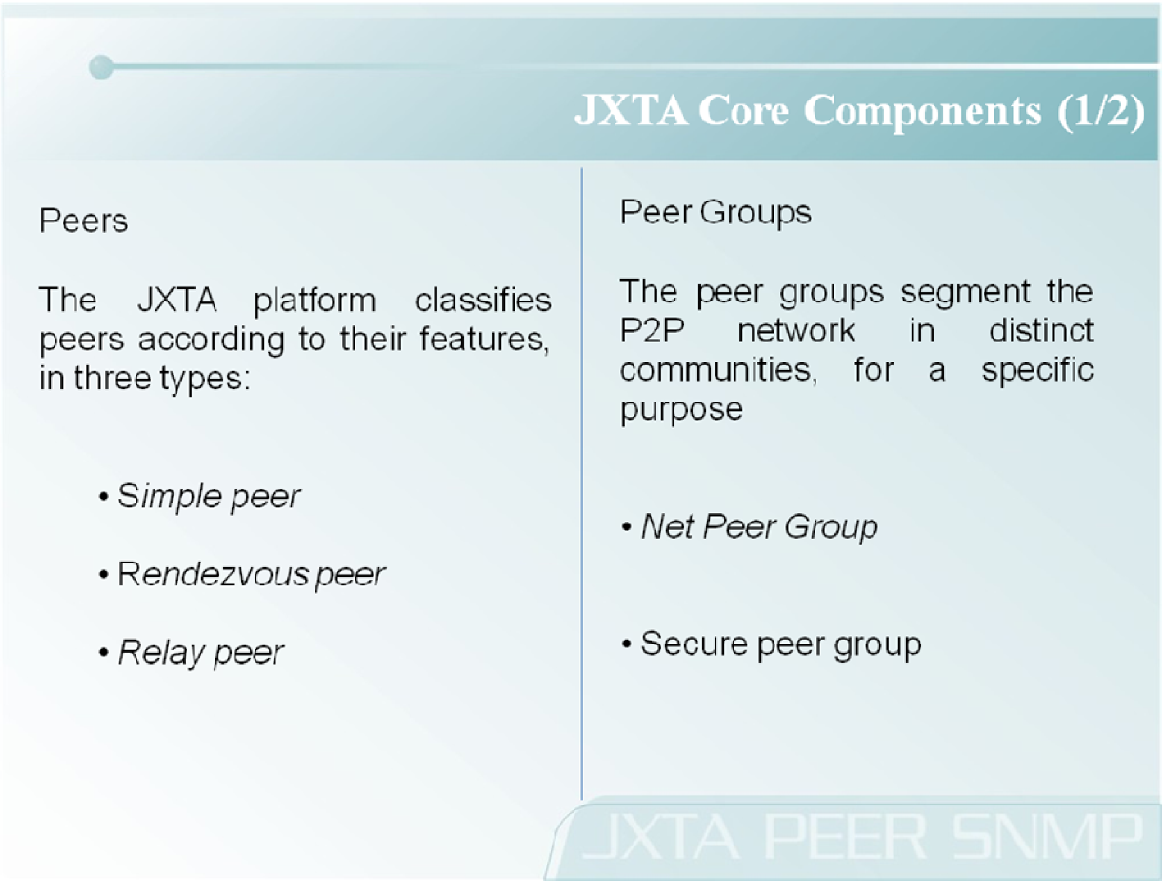
P2P Network

JXTA PEER SNMP

The JXTA platform is structured in three layers, as shown above. Each layer is based on the services of the lower layer, adding features and complexity. These layers are described in the following paragraphs.

The top layer, application layer, includes the P2P applications, for instance, instant messaging. The intermediate layer, called service layer, provides services that help but are not mandatory to the development of P2P applications. Example services include searching, file sharing, peer authentication, among others.

The bottom layer, which is called the kernel, provides the essential elements to the implementation of each JXTA P2P application. The main elements of this layer are peers, peer groups, advertisements, discovery, communication, monitoring, and primitives for authentication and security. This layer is the core of the JXTA solution, and is described in the following paragraphs.



JXTA Core Components (1/2)

Peers

The JXTA platform classifies peers according to their features, in three types:

- *Simple peer*
- Rendezvous *peer*
- *Relay peer*

Peer Groups

The peer groups segment the P2P network in distinct communities, for a specific purpose

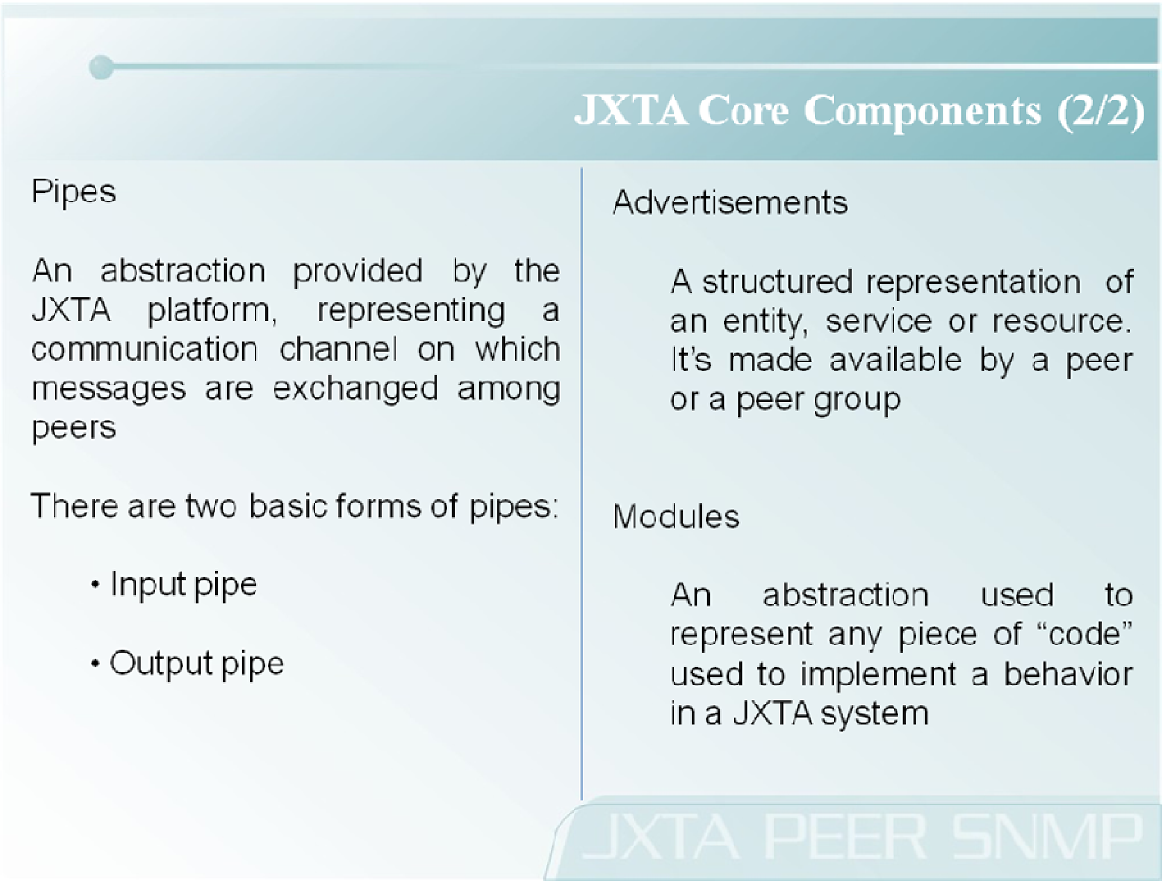
- *Net Peer Group*
- Secure peer group

JXTA PEER SNMP

The most basic concept of a P2P network is the peer itself, and the JXTA platform classifies peers according to their features, there are three different types of peers: simple peers, rendezvous peers, and relay peers. Each type represents a different set of responsibilities. Every peer in the network can be classified as belonging to one or more of these types.

Simple Peers are meant to serve a single user, allowing the user to provide services and to obtain services from other peers in the same P2P network. Possibly this type of peer is located behind a firewall, and peers located outside are probably not capable of connecting directly to them. Rendezvous peers provide an abstraction which represents a meeting point for peers. A rendezvous allow peers, resources and content to be searched. This type of peer is usually placed outside a private network, but it can also be located behind a firewall. In this case, the rendezvous needs a way to communicate through a firewall; this can be done employing a relay peer which is placed outside the firewall. Finally, Relay Peers implement methods to allow the communication of peers separated by firewalls or NAT.

JXTA also defines the so-called peer groups. Groups are an approach to segment the space of a P2P network into distinct peer communities, each organized for a specific purpose. Peers of several networks can join a peer group; each peer can be in more than one peer group simultaneously. Every JXTA peer belongs to a global peer group called *Net Peer Group*. This group allows each peer in the network to be visible to all others. Additionally, it is possible to implement secure peer groups with authentication of incoming peers. Secure peer groups are peer groups that define a set of logins and passwords, restricting the access to the group in such a way that only peers that have the right password can join into the group.



JXTA Core Components (2/2)

Pipes

An abstraction provided by the JXTA platform, representing a communication channel on which messages are exchanged among peers

There are two basic forms of pipes:

- Input pipe
- Output pipe

Advertisements

A structured representation of an entity, service or resource. It's made available by a peer or a peer group

Modules

An abstraction used to represent any piece of "code" used to implement a behavior in a JXTA system

JXTA PEER SNMP

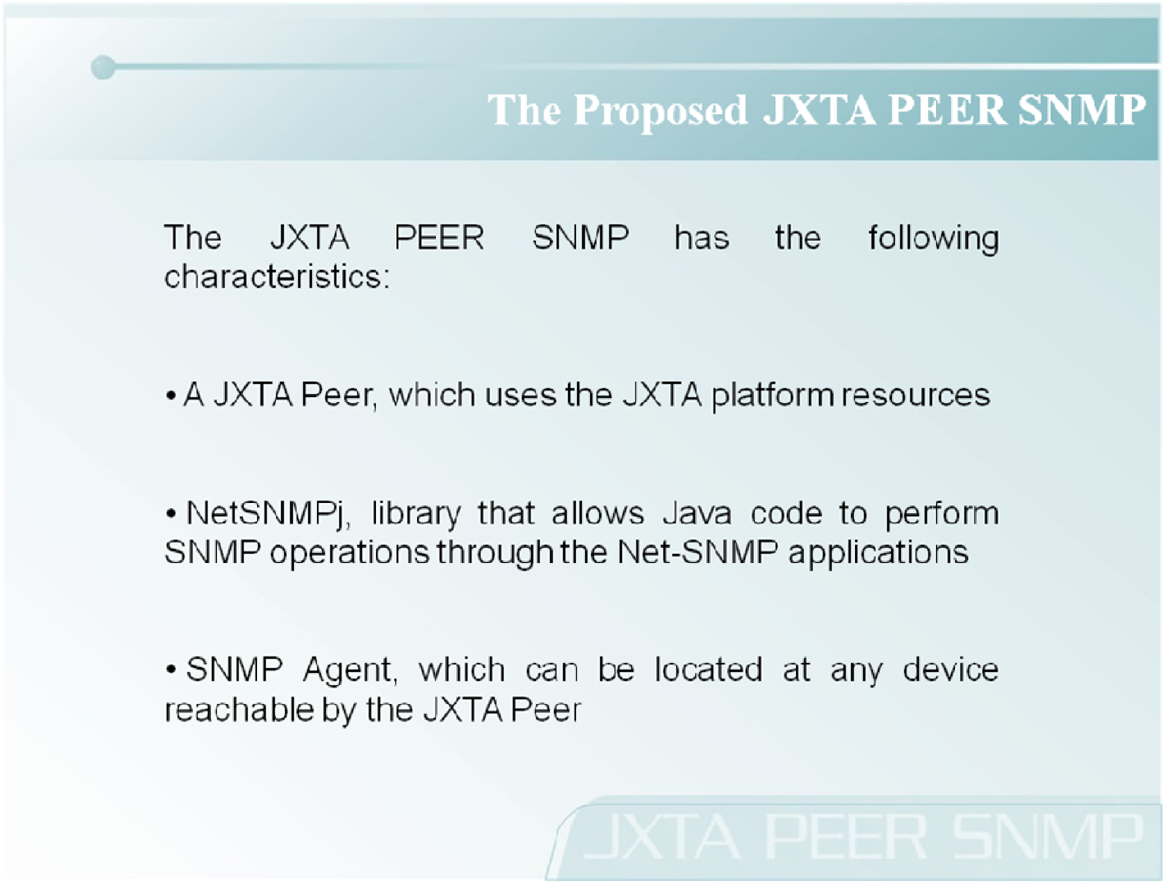
A pipe is abstraction provided by the JXTA platform to represent a communication channel on which messages are exchanged among peers. Basic pipes are asynchronous, unidirectional and unreliable. Nevertheless, other types of pipes with stronger service guarantees are also provided by the platform, which are constructed on top of a simple pipe. Each pipe is either an input pipe or an output pipe. An input pipe is used to receive messages while an output pipe is used to send messages.

A resource, service or entity, is made available to other peers through an abstraction called an advertisement. Every advertisement has a predetermined lifetime. This lifetime can be extended as needed, before the expiration time.

Another important concept is the module. A module is a feature of a peer that can be distributed. A module can be initialized, started or interrupted. The native modules provided by the JXTA platform are responsible for several JXTA features, including the peer groups, as well as services and applications provided by the peer groups.

JXTA modules are an abstraction used to represent any piece of "code" used to implement a behavior in the JXTA world. For instance, modules can be used to represent different implementations of a network service on different platforms.

A module provides a generic abstraction to allow a peer to instantiate a new behavior. As peers browse or join a new peer group, they may find new behaviors that they may want to instantiate. For example, when joining a peer group, a peer may have to learn a new search service that is only used in this peer group. In order to join this group, the peer must instantiate this new search service.



The Proposed JXTA PEER SNMP

The JXTA PEER SNMP has the following characteristics:

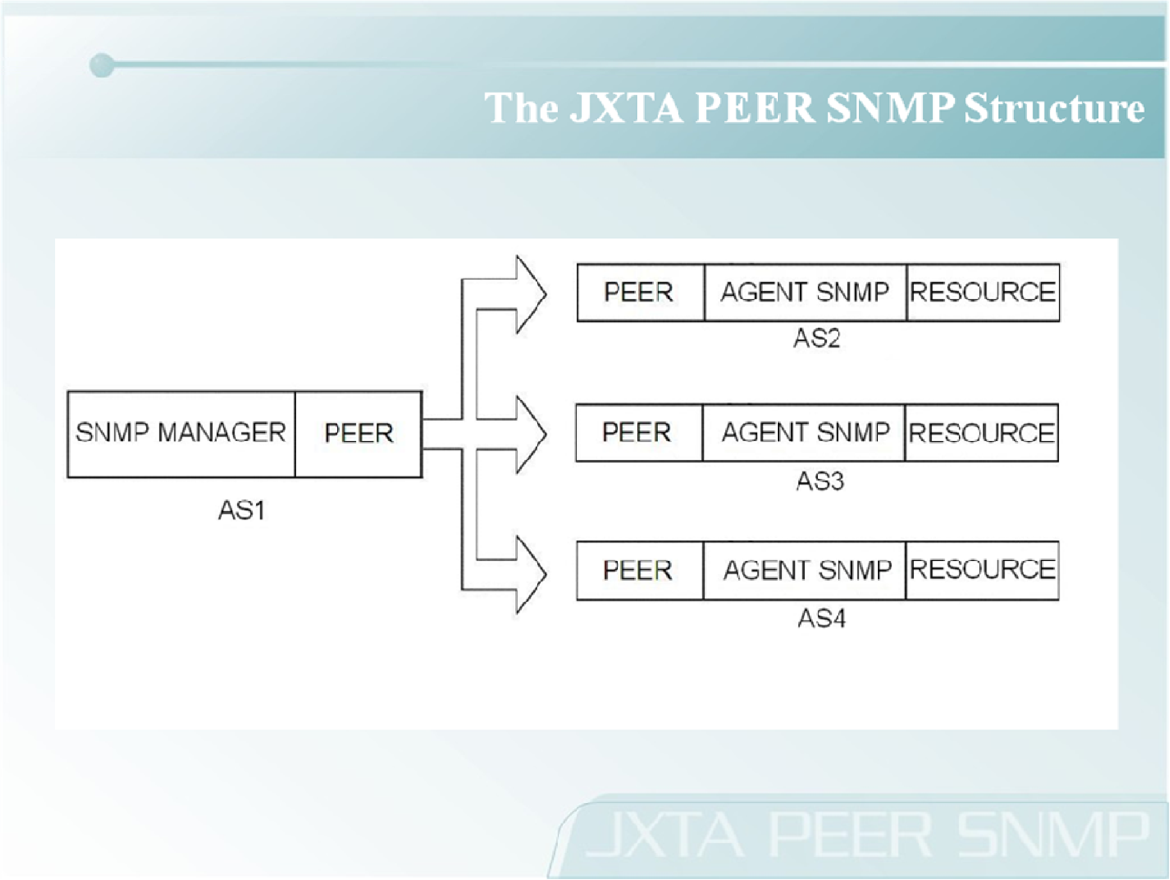
- A JXTA Peer, which uses the JXTA platform resources
- NetSNMPj, library that allows Java code to perform SNMP operations through the Net-SNMP applications
- SNMP Agent, which can be located at any device reachable by the JXTA Peer

JXTA PEER SNMP

The proposed JXTA PEER SNMP consists basically of a JXTA peer implemented in Java. This peer uses a series of resources of the JXTA platform to perform several tasks. These tasks include the creation of secure peer groups, access the rendezvous peer, utilization of authentication mechanisms, and so on.

The JXTA peer, in order to obtain management information, is integrated with the Java-based NetSNMPj library. The NetSNMPj module, within the peer, interacts with the Net-SNMP management applications, which need to be installed on the same device as the JXTA peer.

These Net-SNMP applications communicate with local SNMP agents, running on any device reachable to the JXTA peer.



The JXTA PEER SNMP Structure

SNMP MANAGER | PEER

AS1

PEER | AGENT SNMP | RESOURCE

AS2

PEER | AGENT SNMP | RESOURCE

AS3

PEER | AGENT SNMP | RESOURCE

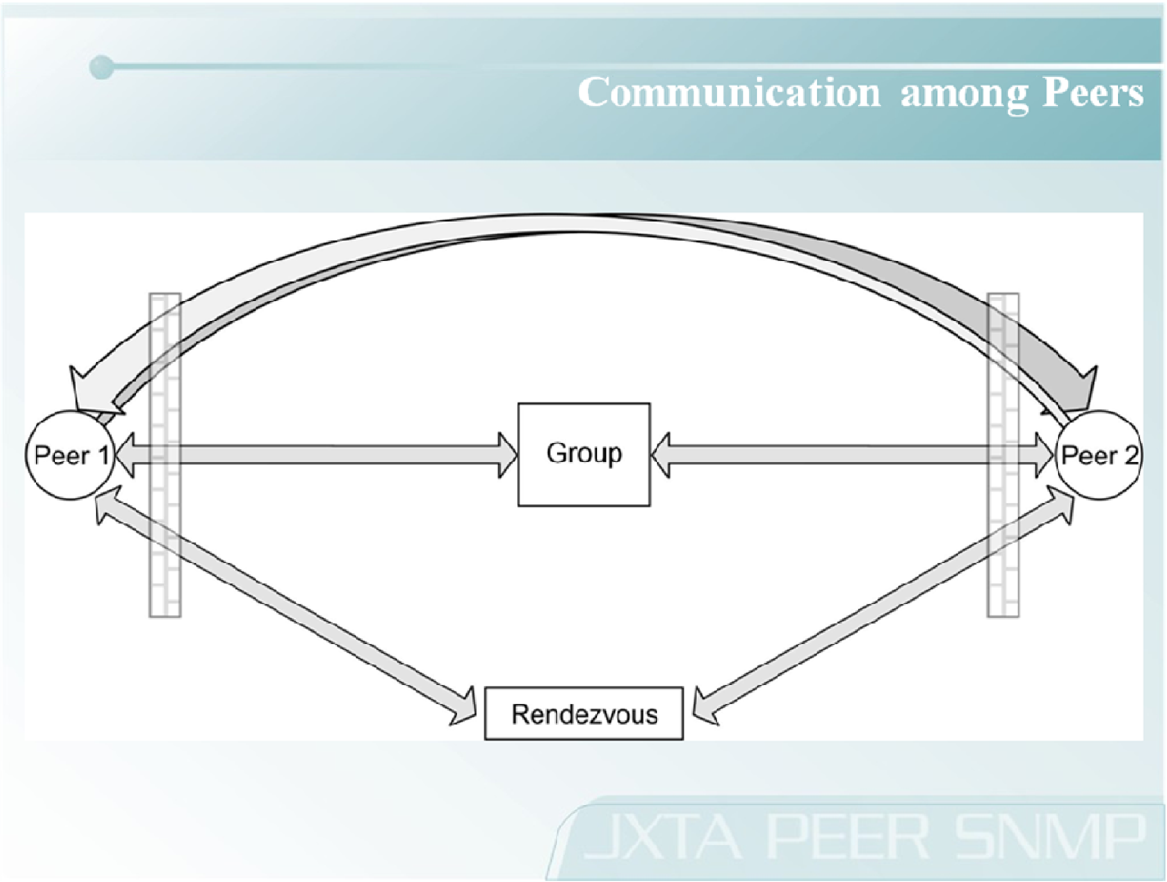
AS4

JXTA PEER SNMP

Using this system, a network administrator can manage resources in several AS's within a unified platform. The system is also secure, in the sense that it assures that only a JXTA PEER SNMP with permission (login and password) can have access to a group created by another JXTA PEER SNMP and be able to communicate with other peers in the group.

In order for a given peer to obtain management information from a given AS, a local SNMP agent running on that AS must be accessed through a peer also running on the target AS. Then the management information response is exchange between the peers.

The current JXTA PEER SNMP implementation provides command-line interface, where requests and SNMP operations are specified. The figure above shows the process in which resources spread in several AS's are accessed. Initially, the manager employs a peer that connects to other peers. Those peers make requests to their corresponding SNMP agents.



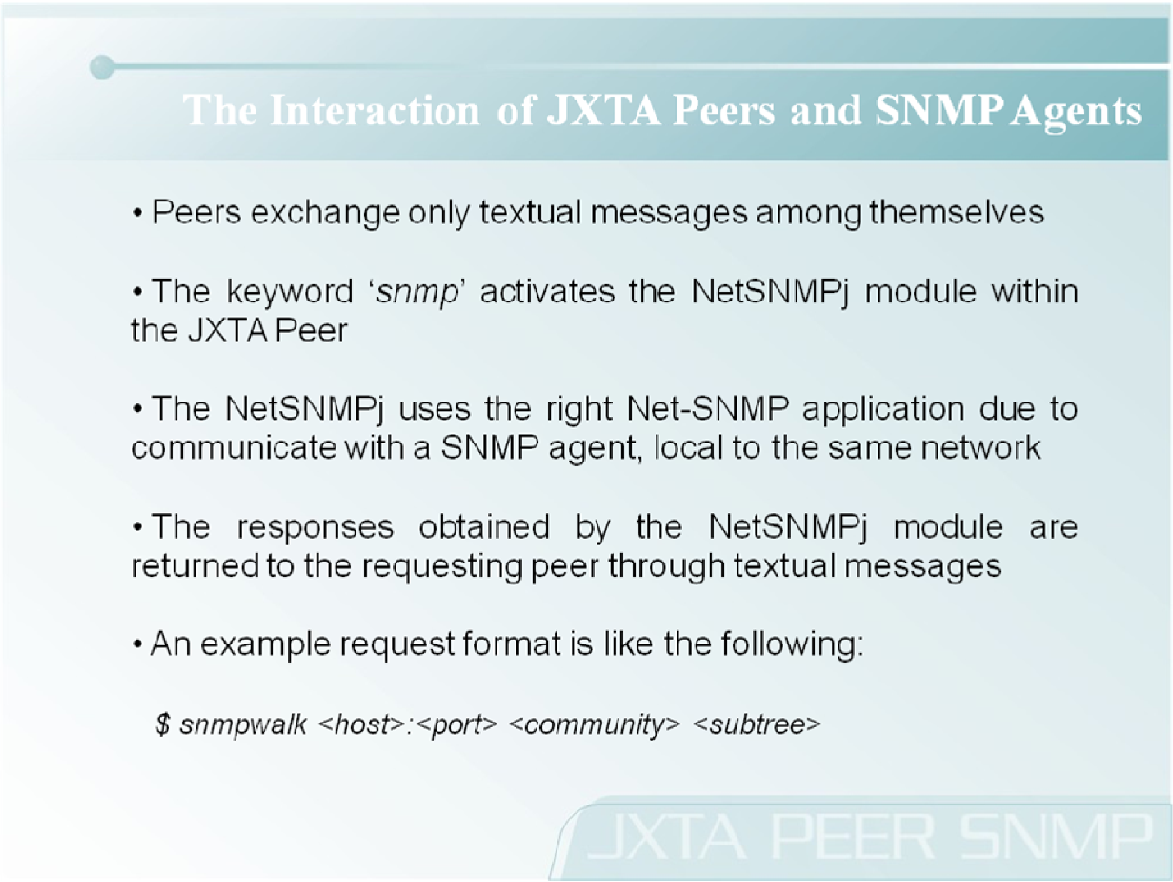
Communication among Peers

JXTA PEER SNMP

Each JXTA PEER SNMP needs, first of all, to connect with a rendezvous, the super-peer of the JXTA platform. After this initial connection is established, it is required to publish a peer group advertisement. The system performs this task automatically on demand. After a secure peer group is created, as described in the next section, each peer must join the group and, once inside the group, publish a pipe advertisement.

Another peer that needs to communicate with the owner of a group (the peer that published the advertisement for that group) or any other peer that had previously joined the group, takes steps similar to those above but, it is not necessary to create the group again, only to join it. Before joining a group, a peer searches peer group advertisements. The search is first executed locally (advertisements in the local network) and then remotely (advertisements in the rendezvous). Once the advertisement is obtained, it is possible to join into the group.

Again, after joining a group, a peer must publish its pipe advertisement. Now, in order to communicate a peer is required to find the pipe advertisement published by the target peer. A communication channel is established between both peers when the advertisement is found. Thus, each peer will have a channel, i.e. a pipe, actually one to receive messages (input pipe) and another one to send messages (output pipe). Figure above depicts the communication between two peers, highlighting that firewalls can be present in the way.



The Interaction of JXTA Peers and SNMP Agents

- Peers exchange only textual messages among themselves
- The keyword '*snmp*' activates the NetSNMPj module within the JXTA Peer
- The NetSNMPj uses the right Net-SNMP application due to communicate with a SNMP agent, local to the same network
- The responses obtained by the NetSNMPj module are returned to the requesting peer through textual messages
- An example request format is like the following:

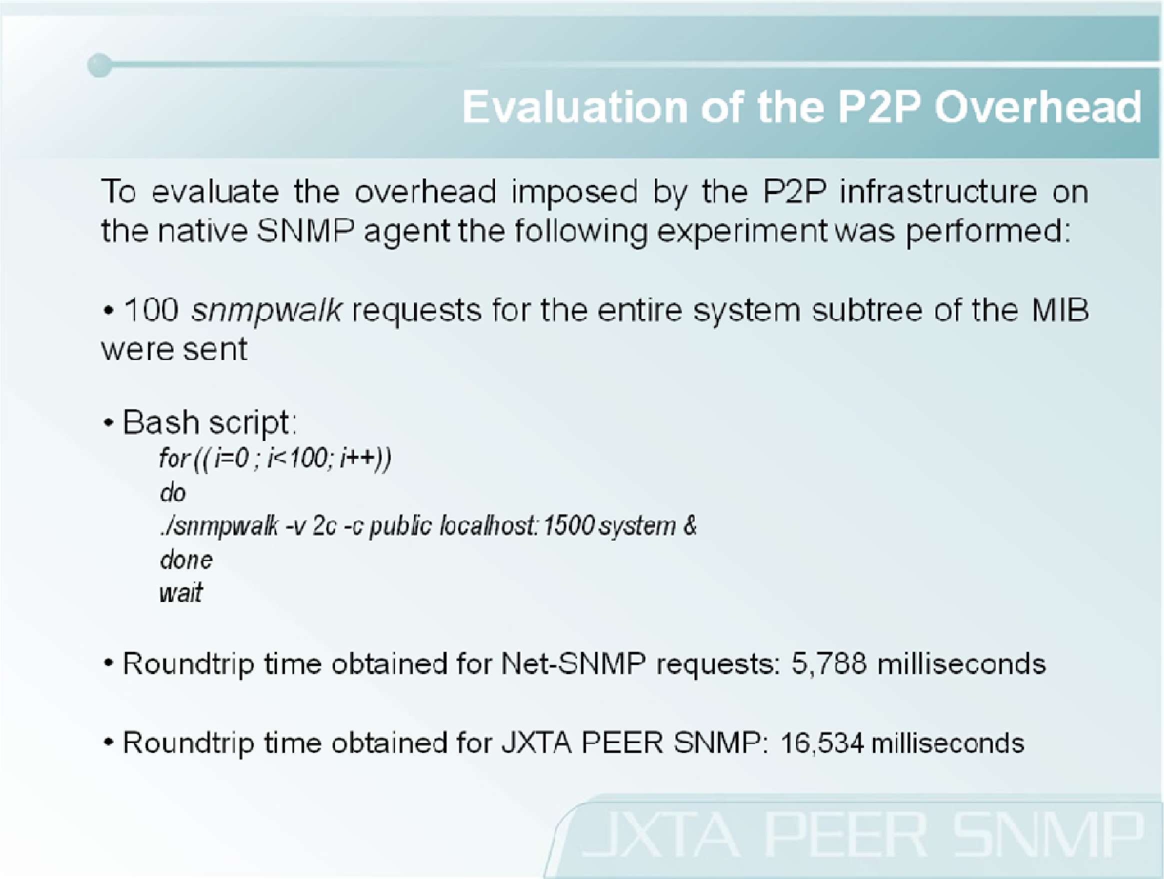
```
$ snmpwalk <host>:<port> <community> <subtree>
```

JXTA PEER SNMP

The communication between peers employs textual messages. By default, a message beginning with the 'SNMP' keyword activates the routines of the NetSNMPj module. An example of the syntax used by the peers to request for management information on a desired device is shown above.

NetSNMPj communicates with local SNMP agents running on one device, or reachable through the network, according to the parameters supplied by the message above.

NetSNMPj implements SNMP operations in particular ways. For example, the *snmpwalk* command is implemented as multiple *getNext* requests. Thus, when a peer sends a message requesting objects in a subtree, a request is sent for the first object in the subtree, and then, another is sent for the next object, and so on, until the end of the subtree is reached. The SNMP agent accesses the MIB on the target device and returns the corresponding responses to the NetSNMPj module on the JXTA peer. Then, these responses are transmitted to the requesting peer also using textual messages. It is important to point out that, in the current implementation, each response is sent to the requesting peer in a single message. Further improvements are possible for this case.



Evaluation of the P2P Overhead

To evaluate the overhead imposed by the P2P infrastructure on the native SNMP agent the following experiment was performed:

- 100 *snmpwalk* requests for the entire system subtree of the MIB were sent

- Bash script:

```
for (( i=0 ; i<100; i++))
do
./snmpwalk -v 2c -c public localhost:1500 system &
done
wait
```

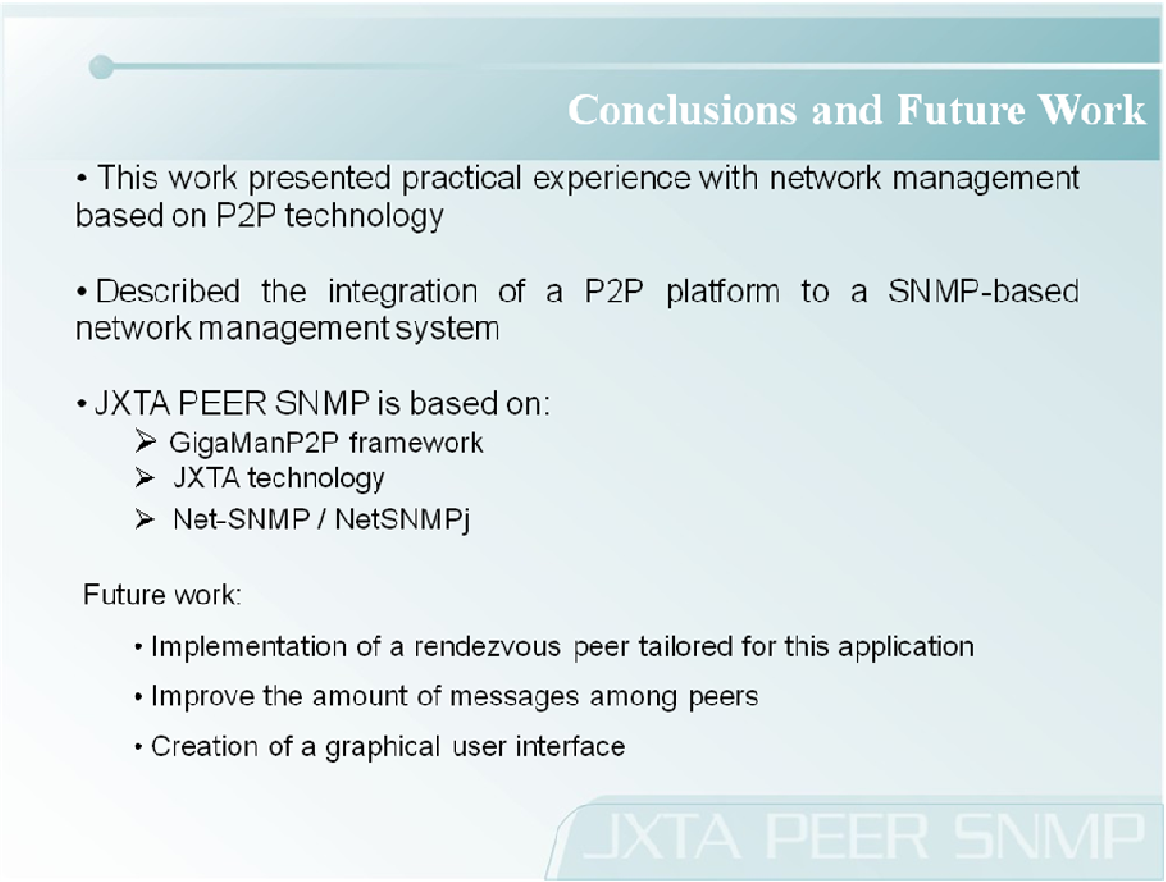
- Roundtrip time obtained for Net-SNMP requests: 5,788 milliseconds

- Roundtrip time obtained for JXTA PEER SNMP: 16,534 milliseconds

JXTA PEER SNMP was implemented, and experimental results were obtained to evaluate the overhead imposed by the P2P structure on the native SNMP agent, when management information is obtained. The main experiment consisted of executing the *snmpwalk* command 100 consecutive times. In each time the entire system subtree of the MIB (Management Information Base) [5] was returned.

In order to measure the overhead, in the first part of the experiment, the delay of employing the native Net-SNMP implementation was measured. After the experiment was repeated several times, an average delay of 5,788 milliseconds was obtained. In the second part of the experiment, the delay for running *snmpwalk* 100 consecutive times was measured, this time through the JXTA PEER SNMP. A peer sent 100 requesting messages to another peer. When the target peer receives a message, it sends a *getNext* request for each object contained in the subtree, to a local SNMP agent. Each response obtained from the SNMP agent is returned as a single textual message to the requesting peer. In this case, the elapsed time, between the sending of the first message and the receiving of the last response, was of 16,534 milliseconds in average.

Comparing the results, it is possible to conclude that the Net-SNMP agent without the JXTA PEER SNMP is, as expected, faster for this case study. This time difference is due, in part, to the fact that responses are sent, by JXTA PEER SNMP, as single messages, one for each object found in the subtree. When the number of objects in the subtree is too large, this represents a hindrance to JXTA PEER SNMP's performance. Despite this seemingly large overhead of JXTA PEER SNMP in comparisons with the native SNMP agent, it is important to point out that the obtained delay is acceptable for most network management applications.



Conclusions and Future Work

- This work presented practical experience with network management based on P2P technology

- Described the integration of a P2P platform to a SNMP-based network management system

- JXTA PEER SNMP is based on:
  - ➢ GigaManP2P framework
  - ➢ JXTA technology
  - ➢ Net-SNMP / NetSNMPj

Future work:

- Implementation of a rendezvous peer tailored for this application
- Improve the amount of messages among peers
- Creation of a graphical user interface

JXTA PEER SNMP

This work introduced JXTA PEER SNMP, a SNMP peer for managing resources spread across several autonomous domains, defined within the GigaManP2P management architecture. JXTA PEER SNMP employs P2P technology, specifically the JXTA framework, also using both the Net-SNMP agent and NetSNMPj management applications. The system allows the construction of secure peer groups which provide member authentication. Peers in these groups can communicate across firewalls. Experimental results comparing the performance of JXTA PEER SNMP and Net-SNMP's native implementation are presented.

Future work includes the implementation of a rendezvous peer specific for the JXTA SNMP architecture. A graphical interface for the system and improvements in the messages exchanging is also to be developed.

References:

1] Project JXTA, http://www.jxta.org, accessed April, 2006.

2] NET-SNMP, http://net-snmp.sourceforge.net, accessed April, 2006.

3] NetSNMPj, http://netsnmpj.sourceforge.net, accessed April, 2006.

4] SNMP4J, http://www.snmp4j.org, accessed October, 2007.

[5] J. Case, K. McCloghrie, M. Rose, S. Waldbusser, "Management Information Base (MIB) for the Simple Network Management Protocol (SNMP)," Request for Comments (RFC), 3418, 2002.

[6] SOUZA, J. N., GRANVILLE, L. Z., PIRMEZ, L., DUARTE Jr., E. P., ANDRADE, R.C., TAROUCO, L. R., CORREIA, R. B., LAGES, A. "GigaManP2P - A Peer-to-Peer Infrastructure for Managing Optical Networks", Capetown, South Africa,03-06 May 2005, IEEE ICT 2005.