

Self-Constructing P2P Networks on the Project JXTA Platform

Bill Yeager

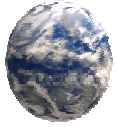
Chief Technologist

Project JXTA

Sun Microsystems, Inc.

Highlights

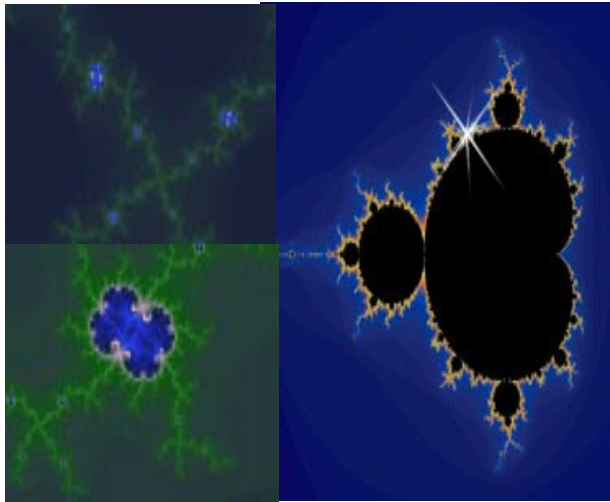
- Some weekend musings ...
 - What is P2P
 - Why P2P Now
- Fundamentals of the JXTA Virtual Network



What is P2P (people to people)?



- Server/Service Centric Internet
 - Host communities are isolated satellites
 - Habitable == access + bandwidth
 - Cold on the edges
 - Not all satellites are equal
 - If you are an Eskimo, you live on pluto



- P2P Internet
 - Hot edges
 - spontaneous growth and activity
 - Silent, powerful infrastructure
 - Edge-to-edge connectivity
 - Edge services
 - Location based
 - Content + aggregators, etc ...

From Fractal Journeys by Katherine McGuire

Why P2P Now?

- Current Internet
 - As the number of **devices** increases
 - Shorter / Fatter network is required
 - 10mbps, 100mbps, 1gbps, 10gbps ...
 - More powerful servers
 - Same service and sometimes less QOS
- We are not “knocking” the technology
 - It’s magnificent!
 - Rather when limits are reached history shows
 - disruptive technology arrives
 - It is scary and there is resistance
 - The economy always benefits

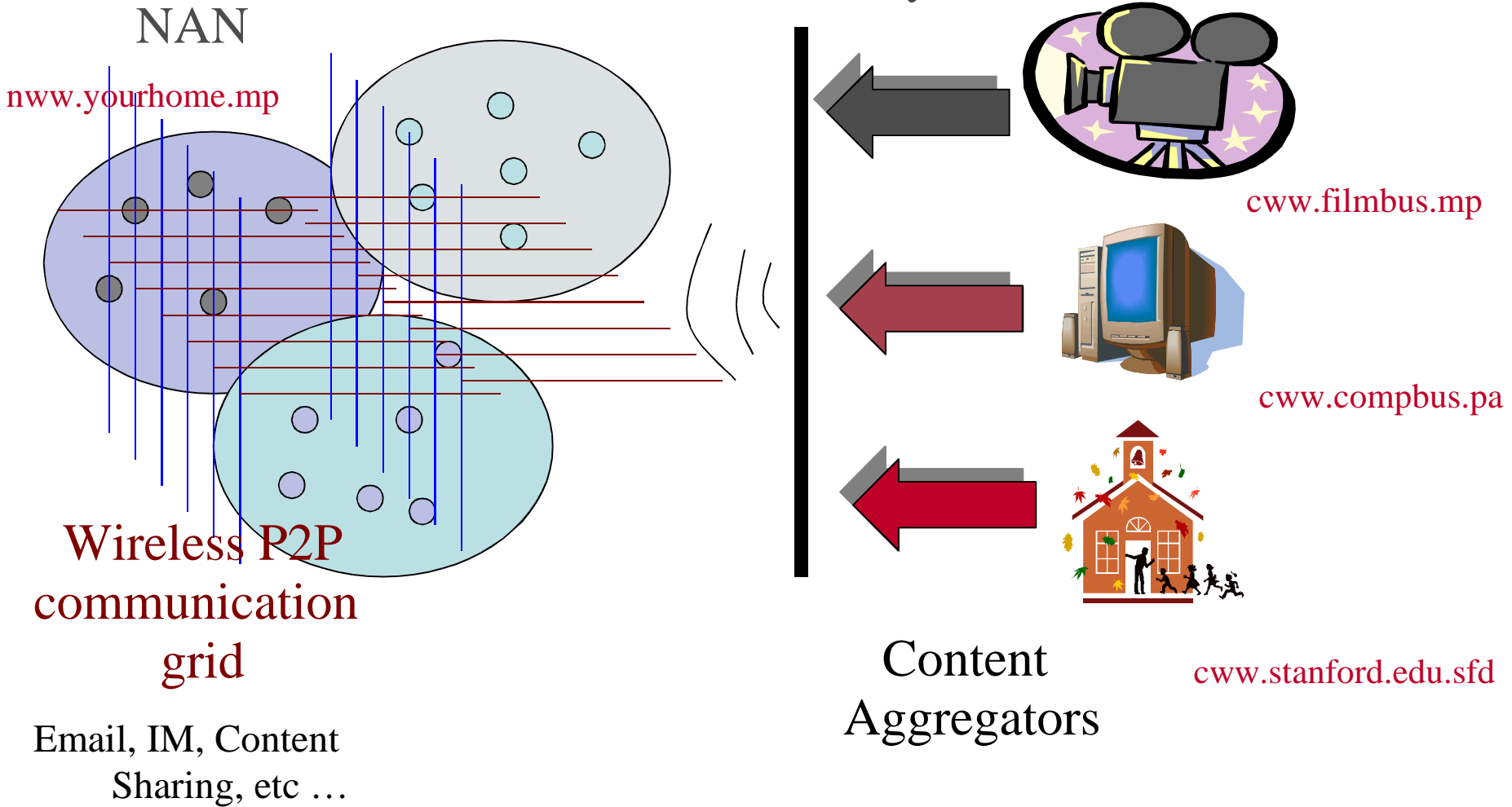
Why P2P Now Continued

Imagine that P2P, Edge technology arrived
Before
The Centralized Internet

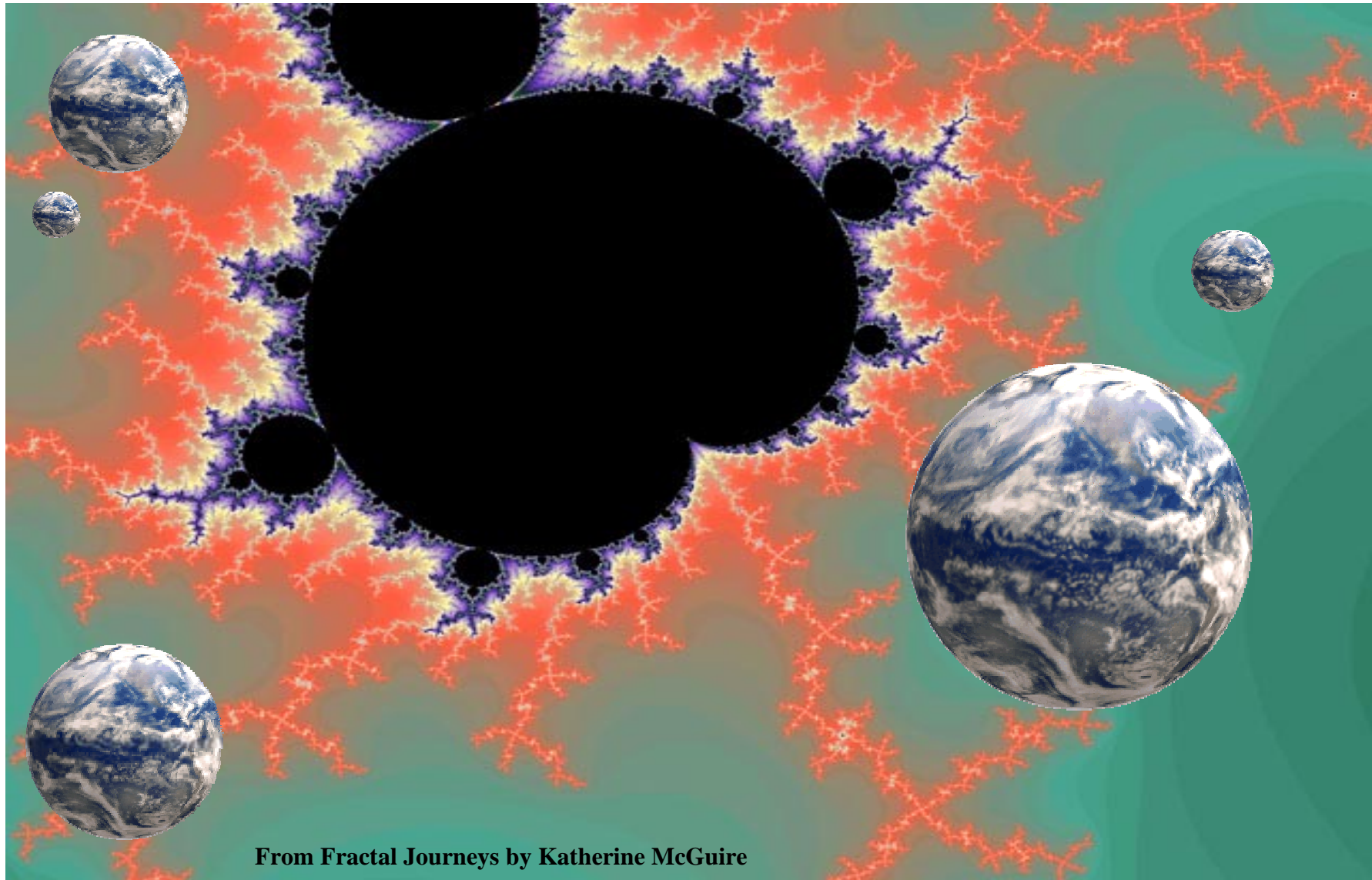


How would the “center” evolve?

From the Outside Inward



The P2P Internet



From *Fractal Journeys* by Katherine McGuire

The Fundamentals of the JXTA Virtual Network

Overview

- The “Stack”
- Peers and Peergroups
- Network Abstractions
- JVNet Definitions
- Protocols
- Security
- Implementation Status

**Security
Is behind it all**

The “Stack”

P2P Applications: IM, Content Sharing, email (Morpheus)

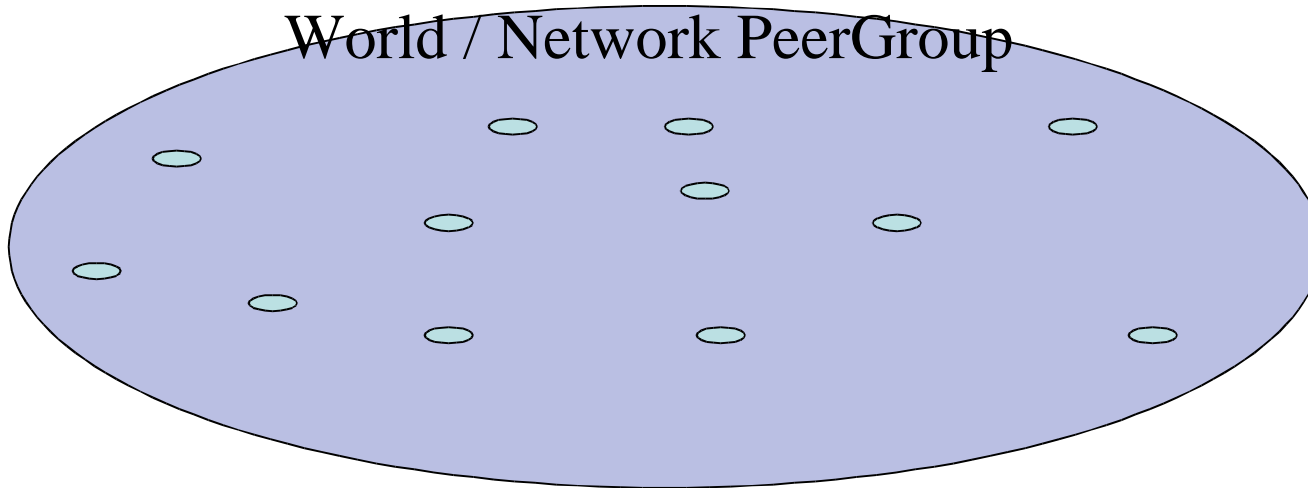
Services: Pipe, (Gnutella, CSN, CAN, DRM)

Peers, Pipes, Endpoints, Advertisements, PeerGroups,
Rendezvous

Protocols: Discovery, Resolution, Group Membership

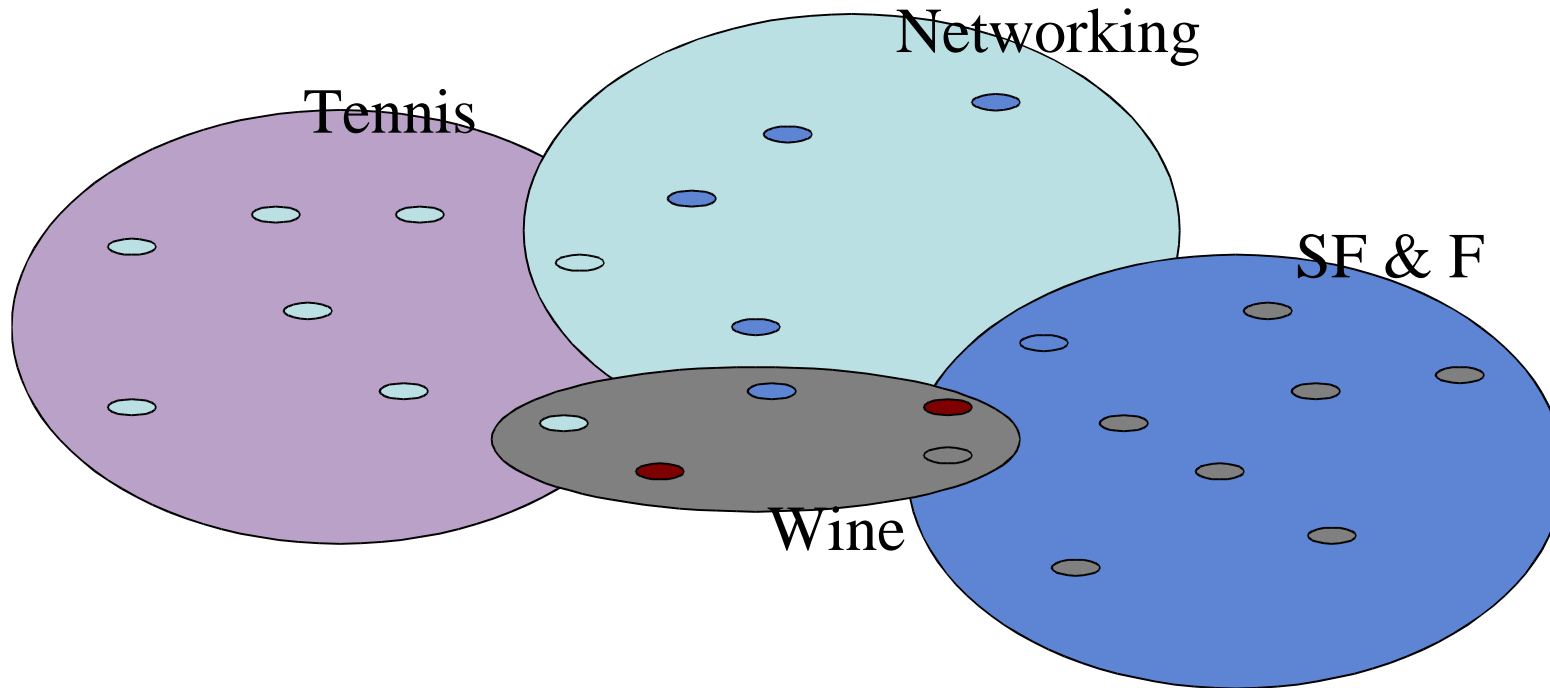


Peers and PeerGroups



- When you boot JXTA you
 - are a member of the World/Network PeerGroup
 - can discover and communicate with other peers in that PeerGroup

Peers and PeerGroups

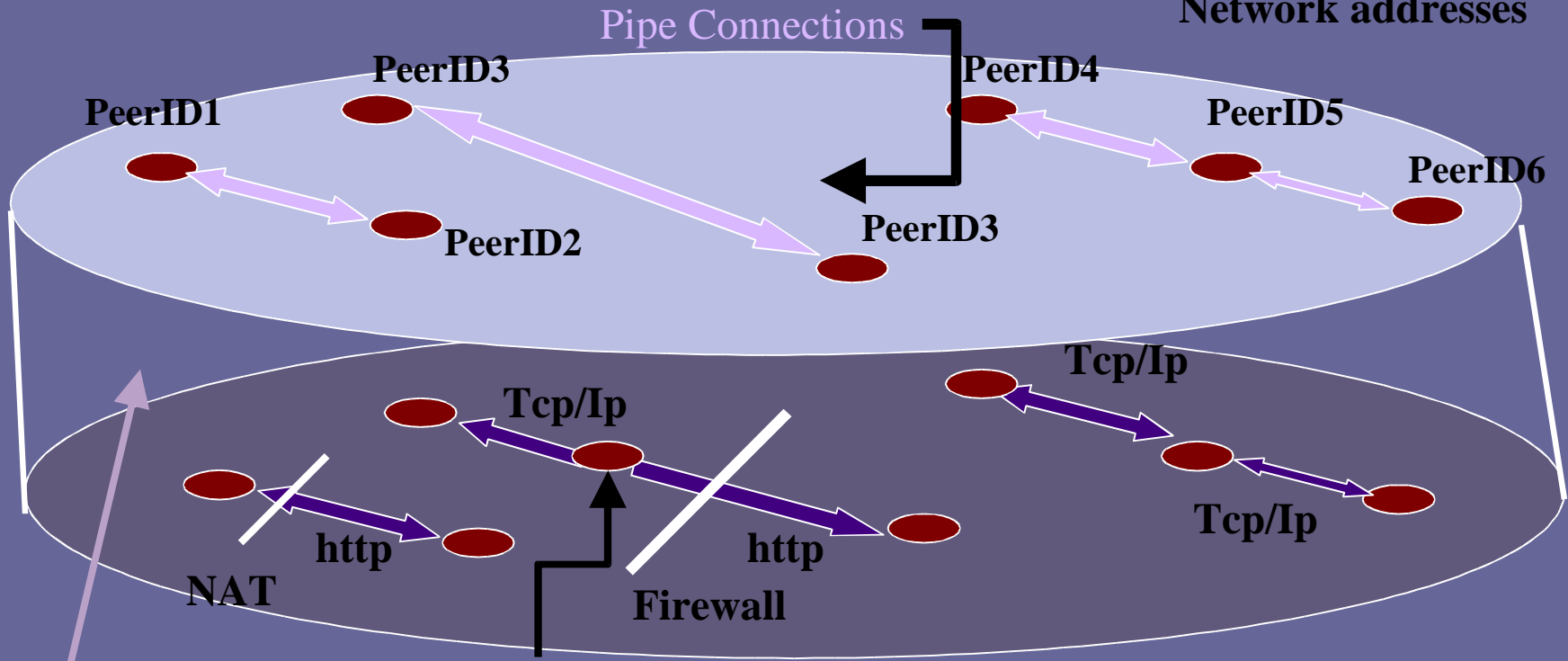


- To give scope to “search” and permit group specific policies
 - JXTA has peerGroups
 - Any peer can create, join and leave PeerGroups
 - A peer can be a member of more than one PeerGroup
 - All searches are applied to the peer’s current PeerGroup

Network Abstractions

JXTA Virtual Network
JVNet

PeerIDs are virtual
Network addresses



JXTA core
software inside

Rendezvous-relay

Real Network

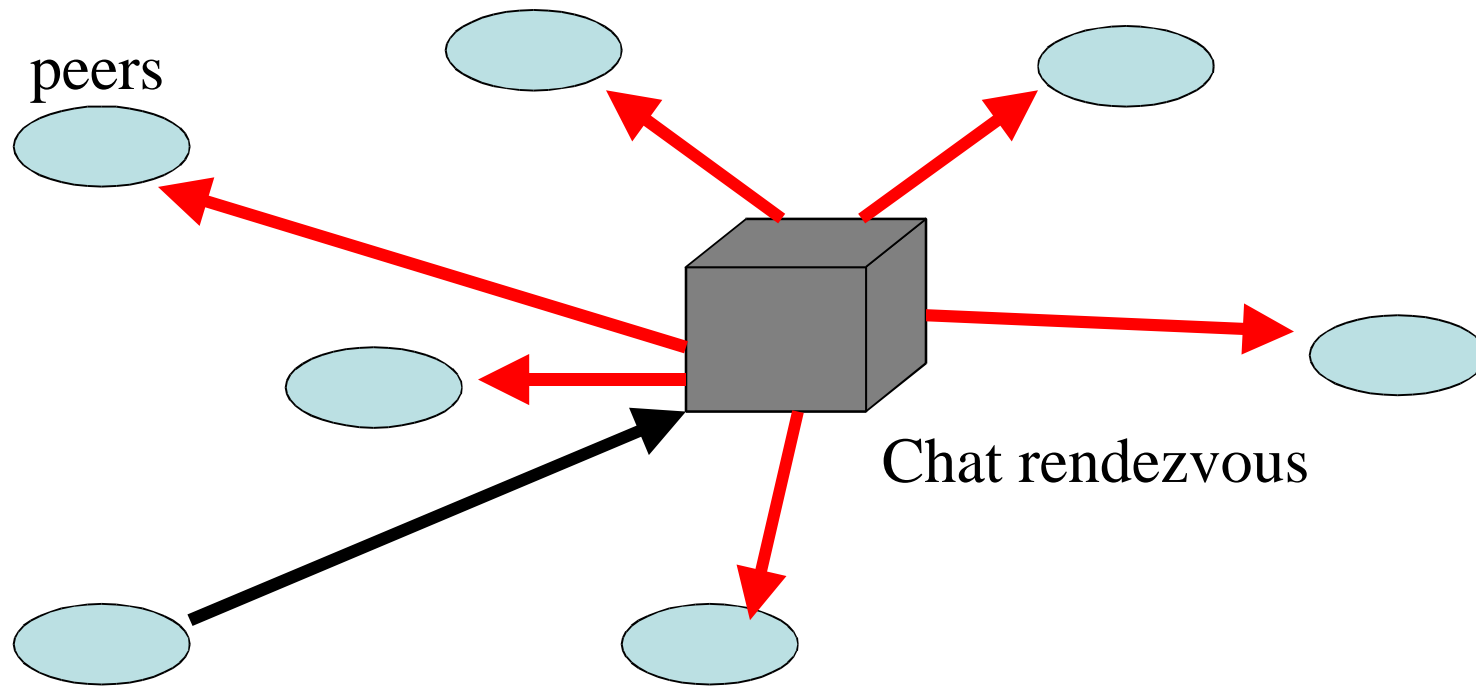
JVNet Definitions#1

- **Peers**
 - **Any device on the jxta virtual network**
 - **PeerName**
 - **Each peer has a human readable text string name**
 - **PeerIDs**
 - **These are virtual network addresses**
 - **Bound to a peer**
 - **urn:uuid-<320 unique bits>**

JVNet Definitions#2

- **Pipes**
 - Used to pass data between peers
 - uni-directional
 - uni-directional secure
 - Multicast listener
 - Only requires registration and is “connection-less.”
 - PipeID
 - These are virtual port numbers
 - Application specific ID
 - » urn:uuid-<320 unique bits>
 - Each port has a human readable application name string
 - Input pipes are created as listeners
 - Output pipes attempt to connect to these “listeners” except for the Multicast pipe.

Group Chat Multicast Pipe Example



— Unicast message
— Multicast Message

N peers sending a message to N-1 peers is always an N^2 problem on the virtual network.

Without Multicast:
If a rendezvous is required, it is $2N^2$ on the real transport.

JVNet Definitions#3

- **Endpoints**
 - **Virtual network socket**
 - **peerID.pipeID**

Unix netstat on a JXTA peer would yield something like

```
Jxta> netstat -a | grep LISTEN
```

```
*.imPipe          LISTEN
```

```
*.MobAgentPipe    LISTEN
```

```
*.secureftpPipe   LISTEN
```

JVNet Definitions#4

And for established pipe connections on the virtual network:

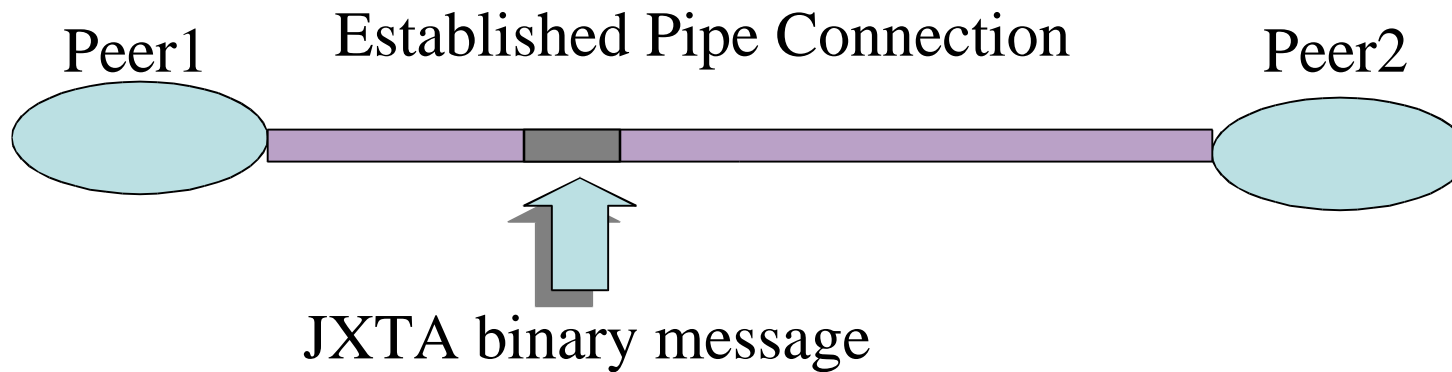
```
Jxta> netstat -a | grep ESTABLISHED
```

localPeerName.im	remotePeerName.remPipeID1	ESTABLISHED
localPeerName.MobAgent	remotePeerName.remPipeID2	ESTABLISHED
localPeerNamesecureftp	remotePeerName.remPipeID3	ESTABLISHED

```
Jxta> netstat -a -n | grep ESTABLISHED
```

localPeerID. PipeID1	remotePeerID1.remPipeID1	ESTABLISHED
localPeerID. PipeID2	remotePeerID1.remPipeID2	ESTABLISHED
localPeerID. PipeID3	remotePeerID2.remPipeID3	ESTABLISHED

JVNet Definitions#5



The JXTA binary message is the IP packet analog

- 1) Source and destination endpoint addresses
- 2) Payload as application namespace defined elements
 - a) Mime typed payload

Binary Message Example

- **Email application payload**
 - **Namespace == jxtaEmail**
 - **jxtaEmail:from**
 - **Text/plain**
 - » “william yeager” <wyeager@peerID1>
 - **jxtaMail:to**
 - **Text/plain**
 - » “chen yu” <cyu@peerID2>
 - **jxtaMail:subject**
 - **Text/plain**
 - » **Stanford networking seminar**
 - **jxtaMail:body**
 - **Text/plain**
 - » **It’s on for next week.**

Advertisements

- JXTA uses advertisements to create descriptions of peers
 - XML Documents
 - Peer Advertisement
 - Pipe Advertisement
 - PeerGroup Advertisement
 - Rendezvous Advertisement
 - Protocols are used to publish and subscribe to these document types

Advertisements: Quick Overview#1

- Peer Advertisement
 - Peer nameS
 - Bill@MTV29
 - Peer ID is 64 bytes
 - urn:uuid-<20 group + 20 peer unique bytes>+padding+format
 - Group ID
 - **Default is urn:jxta:jxta-NetGroup**
 - Endpoint address parameters
 - peerID (jxta virtual network)
 - Tcp://ipaddress (tcp transport)
 - Jxtatls:peerID/TlsTransport/jxta-WorldGroup (tls transport)
 - Rendezvous True or False
 - Root x509.v3 certificate

Advertisements: Quick Overview#2

- Pipe Advertisement
 - PipeID is 64 bytes
 - urn:uuid-<20 group + 20 pipe unique bytes> + padding + format info
 - Type
 - JxtaUnicast
 - JxtaUnicastSecure
 - JxtaPropagate (multicast listener)
 - Name
 - ApplicationString.human-readable-text
 - EG: JxtaMobileAgent.wjyNAN@menloOaks

Advertisements: Quick Overview#3

- Group Advertisement

- groupID

- urn:uuid-<20 unique bytes> + format + padding
 - urn:uuid:jxta-NetGroup (well know group name)

- Name

- Text String

- For example: Networking-seminar

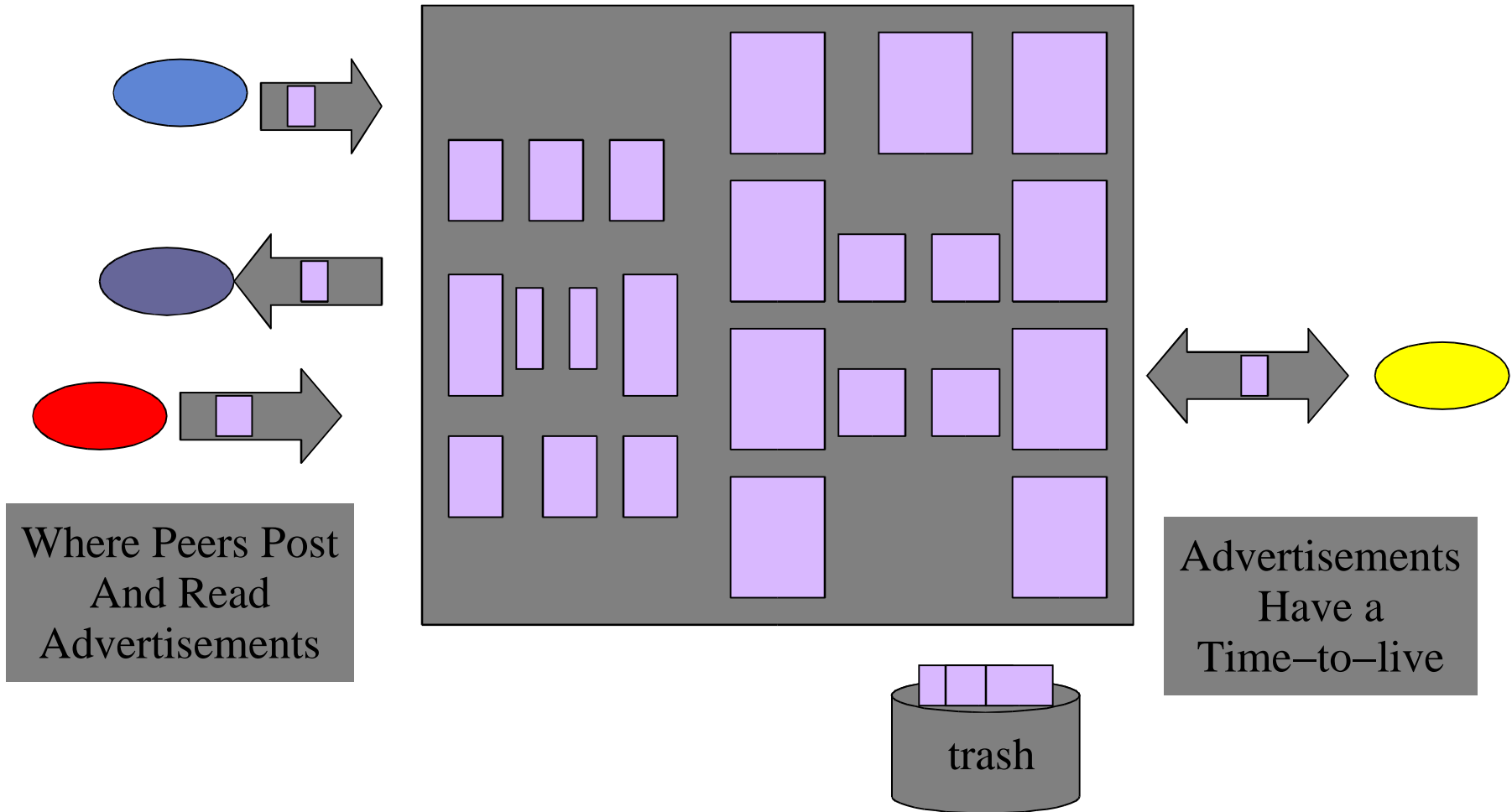
- Module specification id

- Defines the version of the API's used by the group

- Permits multiple implementations of group policies, eg, authentication

- » urn:uuid-<numeric-id> + padding

Rendezvous – The JXTA Bulletin Board



Protocols

- **Discovery Group Services**
 - Publication of advertisements
 - Locally and remotely
 - Upon advertisement creation they are usually published
 - Subscription to published advertisements
 - Retrieve locally and remotely published
 - These are searches and are scoped to the peerGroup
 - One can refine the search for pipe adv to the application
 - » Search for MobileAgent listening pipes in the peerMonitoring group for example

Protocols

- **JXTA Pipe Binding Protocol (JPBP)**
 - Pipe advertisements are published without locality, i. e., not bound to a peer
 - First one discovers a pipe advertisement
 - 2nd one must find the peer to whom it belongs
 - **Ask “if this is your pipe adv, then please send me your Peer advertisement”**
 - **The JPBP does this much like ARP (IP address resolution protocol) finds MAC addresses belonging to IP addresses.**
 - **Searches for pipe owners’ peer advertisements are again restricted to the current peerGroup.**

Pipe / Binary Message Code Example

Assume the object pipeService is an instance of PipeService:

```
String from = "\"Wei-Li\" <weili@peerID1 >";
```

```
// create binary message
```

```
Message msg = pipeService.createMessage();
```

```
// add the payload
```

```
msg.setBytes("jxtaMail:From", from.getBytes());
```

```
:
```

Assume adv is the input pipeAdvertisement we want to resolve:

```
// create and resolve the output pipe
```

```
OutputPipe pipeOut = pipeService.createOutputPipe(adv, 30000);
```

```
// and send the message
```

```
pipeOut.send(msg);
```

Group Membership Protocol

- Peers can
 - Create groups
 - Creates and publishes advertisement
 - Provide the code for the membership policies
 - Authentication and authorization for example
 - Join groups
 - Automatically changes the groupID in the peer advertisement
 - Leave groups

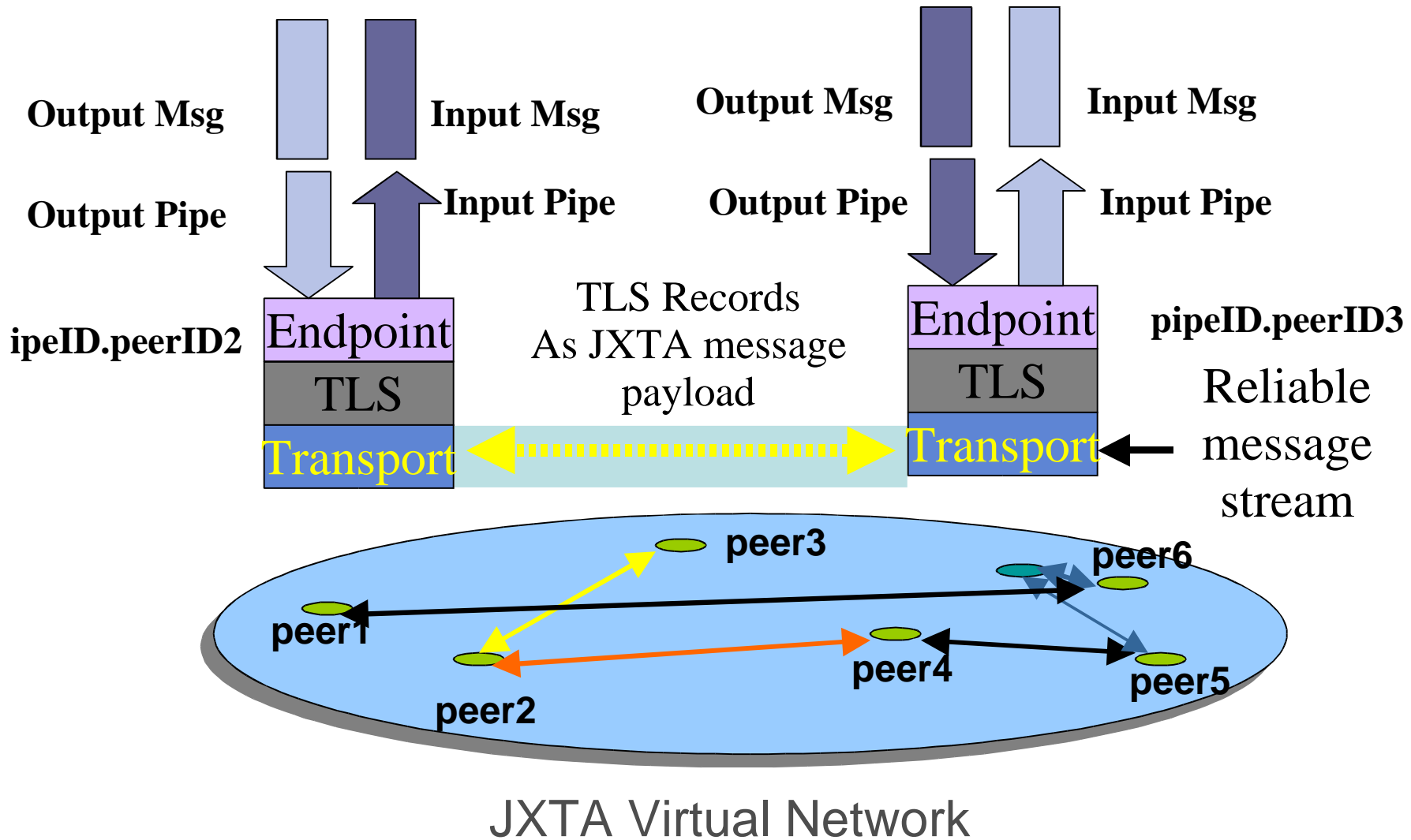
JXTA's Virtual Network Core Implementation

- **The JXTA Virtual Network hides**
 - The complexity of the real underlying transports and their associated protocols
 - Just what is your IP address?
 - The infrastructure barriers
 - NAT, Firewalls
 - The physical transports
 - JXTA works end-to-end across physical layers
- from the P2P application programmer**

JXTA Security

- Poblano “degrees of security model”
 - \$0 cost is least secure (every peer it’s own Certificate Authority)
 - Use standard Web CA’s for most secure
 - <http://www.jxta.org/docs/trust.pdf>
 - by Bill Yeager and Rita Chen
- Full implementation of Transport Layer Security [rfc2246] This is SSL.V3+
 - Default Cipher suite
 - RSA1024, 3DES EDE CBC, SHA-1
 - X509.v3 certificates
 - <http://www.jxta.org/project/www/docs/SecurityJXTA.PDF>
 - by Bill Yeager

TLS Transport



Implementation Status

- Full J2SE implementation is complete
- Full Java Mobile phone implementation is complete
- C implementation is in progress with full completion targeted for May.
- All implementations are edge-to-edge interoperable!
 - We can discover peers, maintain “buddy lists,” and chat from a mobile phone to a laptop or desktop and vice-versa.

Questions?

Project JXTA is open
source

<http://www.jxta.org>