peer-to-peer and agent-based computing

JXTA

UNIVERSITY of ABERDEEN

Project JXTA

- Conceived by Bill Joy (SUN Microsystems)
  - Also the creator of "vi"
- Designed by small number of experts from academia and industry
- Stands for juxtaposed, i.e., side-by-side
  - a recognition that P2P is juxtaposed to client-server or Web-based computing, which is today's traditional distributed computing model
    - "A network programming and computing platform"
- Features:
  - Interoperability (different services)
  - Platform and language independence
  - Ubiquity (anything can be a peer)
  - Open standards (XML, HTTP, TCP/IP)
  - Open source (Java source available)

JXTA design constraints

Interoperability

- Software vendors tend to create specific code for their services e.g. file sharing, instant messaging etc
- Incompatible systems, duplication of effort
- JXTA give peers common language to talk

Platform independence – JXTA technology is designed to be independent of:

- Programming languages e.g. C or Java
- System platforms e.g. Microsoft Windows and UNIX
- Networking platforms (such as TCP/IP or Bluetooth)

Ubiquity

- Implementable on every device with a digital heartbeat e.g. PDAs, phones, sensors, etc.
- Avoid specific binding to platforms
- future proof e.g. such technologies should be extended to new platforms e.g. mobile phones etc e.g. using J2ME

JXTA implementations

JXTA Platform Current Implementations

- Java 2 Platform Standard Edition (J2SE) – the reference implementation
- Java 2 Platform Micro Edition (J2ME) – for cell phones, PDAs, and controllers
- PersonalJavaTM technology – for PDAs
- C, PERL, Python, Ruby, Mon (C#), Smalltalk

JXTA Transport Current Implementations

- TCP
- HTTP
- BEEP
- Reliable Multicast

Project JXTA

Goals/Purpose

- enable a wide range of distributed computing applications by developing a common set of general purpose P2P protocols
- achieve platform independence: any language, any OS, any hardware
- overcome the limitations found in many of today's P2P applications
- enable peers to be any device that has a digital heartbeat (desktop computers, servers, PDAs, cell phones, and other connected devices)

Project JXTA

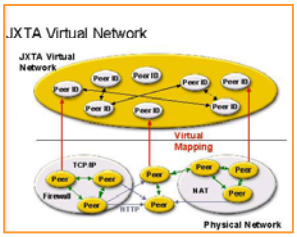
- In other words:
  - Interoperability (different services)
  - Platform and language independence
  - Ubiquity (anything can be a peer)
  - Open standards (XML, HTTP, TCP/IP)
  - Open source (Java source available)

Project JXTA: Technology (Cont'd)

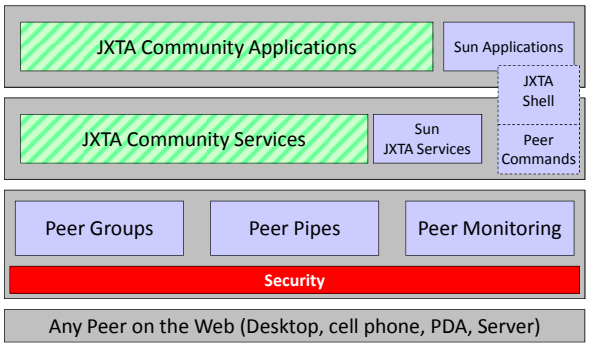
- Based on XML, Java, and key concepts of UNIX operating system
- Transmitted information packaged as messages. Messages define an XML envelop to transfer any kind of data
- Java is not required: JXTA protocols can be implemented in C, C++, Perl, or any other programming language

Project JXTA Technology (Cont'd)

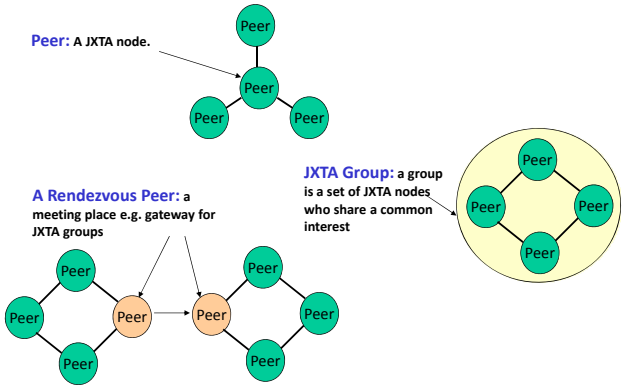
- Set of simple, open peer-to-peer protocols:
  - Any digital device that can be connected may communicate, collaborate and share resources
- JXTA peers create virtual, ad-hoc networks on top of existing networks, hiding details



Project JXTA Architecture



JXTA terms



Project JXTA Architecture (Cont'd)

- Core layer:
  - Peer groups – create, delete, join, advertise, discover, communication, security, sharing.
  - Peer pipes – transfer of data, content and code in protocol-independent way.
  - Peer monitoring – access control, priority setting, traffic metering, bandwidth balancing.
- Service layer:
  - Expand core and facilitate application development
  - Mechanisms for indexing, searching, resource sharing
- Application layer:
  - Built using peer services as well as the core layer
  - Emailing, auctioning, storage systems

JXTA technology concepts

- Lightweight specification:
  - Security is not specified (many ways to do it…)
  - Focus on mechanisms, rather than policies
- Entity:
  - A peer, a service, an advertisement, a pipe…
- Identifiers:
  - Every entity has an identifier
  - Uses UUID, a 128-bit datum
  - Within a local run-time environment, each entity has a unique identifier.

JXTA technology concepts (Cont'd)

- Resource:
  - A peer, a peer group, a pipe, or a service
- Advertisements:
  - XML structured document
  - Names, describes and publishes a resource
  - JXTA defines a basic set of advertisements
  - Other kinds of adverts via XML schemas
- Peer:
  - Any entity that speaks the protocols of a peer
  - A PDA, a sensor, a PC, a human user
  - No need to speak all 6 protocols!

JXTA technology concepts (Cont'd)

- Messages:
  - Asynchronous, unreliable, and unidirectional transport
  - Datagrams with envelope and stack of protocol headers with bodies
- Peer groups:
  - Virtual entity that speaks the set of protocols of a group of peers.
  - Collection of cooperating peers providing a common set of services
  - Spontaneous, not prescribed nor mandatory
  - Protocol available to discover peer groups
  - World Peer Group includes all JXTA peers

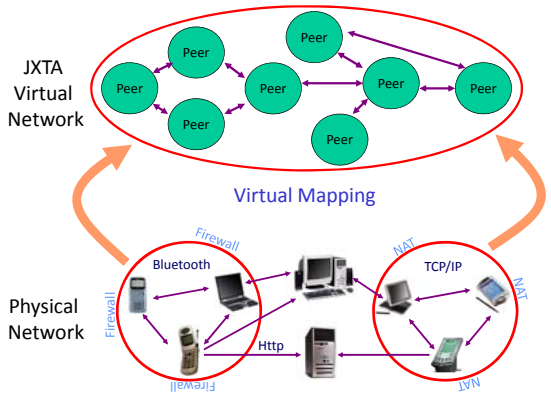
JXTA technology concepts (Cont'd)

- Pipes:
  - Communication channels to send/receive messages
  - Unidirectional, hence input and output pipes
  - Virtual: a pipe's endpoint may be connected to more than one peer endpoint
  - Pipe Binding Protocol binds pipe at run-time
  - Different protocols can be used within a pipe to transfer data

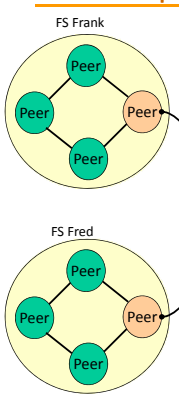
Virtual JXTA

- JXTA Virtual Network overlay
- JXTA Groups
- JXTA Virtual Pipes

JXTA Virtual Mapping



JXTA Groups

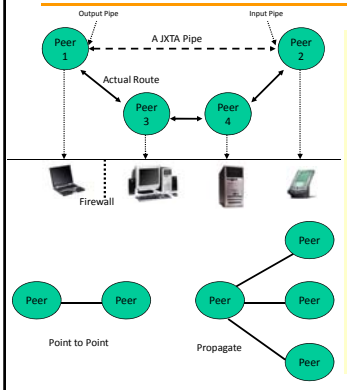


- Virtual entity - speak a set of peer group protocols
- Collection of cooperating peers providing a common set of services e.g. file sharing peer group, a CPU sharing peer group.
- Peer group boundaries define search scope
- Can be used to create a monitoring environment
- Can be password protected and implement local security policies
- One special group, called the *World Peer Group* (the default peer group a peer joins) that includes all JXTA peers.
- At least one rendezvous for a group – groups are the scoping environment for a rendezvous

JXTA Groups

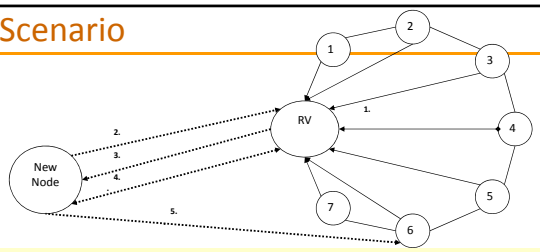
- Provide the grouping of services/users to provide:
  - Scope for searching
  - Security
  - Monitoring
  - And accountability etc.
- Provide a similar concept to VO's in Grid computing except:
  - VOs are more flexible
  - More fine-grained sharing rules
  - Low-level services
    - e.g. Job submission and file transfer vs services and pipe comms.

JXTA Pipes



- Virtual Communication Channels
  - Switchable e.g. TCP/IP, HTTP, Bluetooth
  - NOT point to point
- Pipe endpoints → *input pipe* (the receiving end) and the *output pipe* (the sending end).
- Asynchronous and unidirectional
- Messages flow from the output pipe into the input pipes.
- Support the transfer of any object, including binary code, data strings, and Java technology-based objects
- Two Types:
  - (End) Point to (End) Point
  - Propagate (multicast)

JXTA Scenario



1. Rendezvous node (RV) accepts connection for nodes 1-7 and stores advertisements locally
2. New node contacts Rendezvous using a discovery mechanism e.g. Unicast/multicast (PDP)
3. RV authenticates New Node and adds the New Node to the group (RVP)
4. New Nodes performs a file search query by contacting the RV find a match locally or propagates this query to all other members in the group. The file is found on node 6 (PDP)
5. New Node and node 6 communicate directly through a JXTA pipe. This connection is virtual and may actually traverse (route) through the RV node and node 7

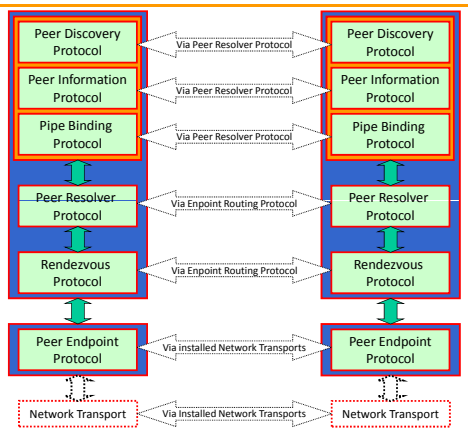
JXTA: Protocols

A set of 6 defined protocols:

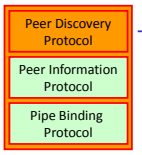
1. Peer Endpoint Protocol (PEP)
2. Rendezvous Protocol (RVP)
3. Peer Resolver Protocol (PRP)
4. Peer Discovery Protocol (PDP)
5. Peer Information Protocol (PIP)
6. Pipe Binding Protocol (PBP)

Protocols can be implemented in any programming language for any system

JXTA: Protocol Stack



Peer Discovery

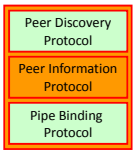


- A peer uses the PDP to discover a JXTA resource
- resources are described by advertisements e.g. can be services, pipes, peers, peer groups, or any other advertisements
- Note that the first word (peer) is the subject & not necessarily the object
- Using this protocol, peers can advertise their own resources, and discover the resources from other peers
- Peer resources are published using XML-based advertisements

Two levels of discovery:

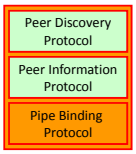
1. Joining a JXTA network
   1. Multicast
   2. Unicast
2. Discovering JXTA resource within a JXTA network.

Finding information about peers

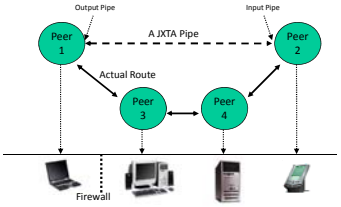


- allows peers to learn about the capabilities and status of other peers e.g. uptime, traffic load, capabilities, state etc
  - e.g. one can send a *ping* message to see if a peer is alive.
- also query a peer's properties where each property as a name and a value string
- useful for implementing monitoring

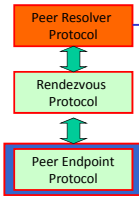
Binding pipes



- Allows a peer to establish a virtual communication channel (i.e. a pipe) between peers
- Allows the binding of the two or more ends of the pipe endpoints forming the connection
- A peer binds a pipe advertisement to a pipe endpoint thus indicating here messages actually go over the pipe
- Bind occurs during the open operation, whereas unbind occurs during the close operation.

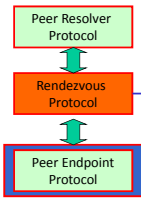


"The" Resolver



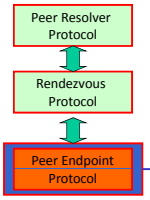
- Enables a peer to implement high-level search capabilities
- Allows a peer to send and receive generic queries to find or search for peers, peer groups, pipes, and other information

Rendezvous



- Allows a Peer to send messages to all the listeners of the service
- The rendezvous protocol defines how a peer can subscribe or be a subscriber to a propagation service allowing larger communities to form
- A rendezvous nodes' scope is a peer group
  - e.g. the rendezvous protocol is used by the peer resolver protocol and by the pipe binding protocol in order to propagate messages.

Routing those messages



- allows a peer to find information about the available routes for sending a message to destination peer
- i.e. pipes are often not directly connected to each other
- allows the implementation of routing algorithms into JXTA
- Peers implementing the endpoint routing protocol respond to queries with available route information giving a list of gateways along the route.

JXTA: security considerations

- No security mechanisms in place yet…
- Security is a big challenge:
  - Peers can be clients and servers
  - Anonymity of peers

JXTA: potential applications

- Search the entire web and all its connected devices (not just servers) for needed information
- Save files and information to distributed locations on the network
- Connect game systems so that multiple people in multiple locations
- Participate in auctions among selected groups of individuals
- Collaborate on projects from anywhere using any connected device
- Share compute services, such as processor cycles or storage systems, regardless of where the systems or the users are located

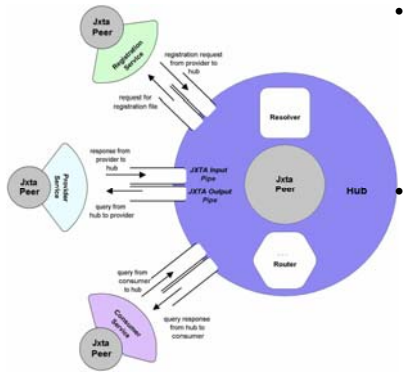
JXTA Search Overview

- Started in June 2000 by Infrasearch as an idea to distribute queries to network peers best capable of answering them.
- Now it is the default searching methodology for the JXTA framework in the form of JXTA Search.
- Communication via an XML protocol called Query Routing Protocol (QRP).
- Network components: providers, consumers, hubs
- Capable of providing both wide and deep search
  - deep search shows the most benefits
- Design goals:
  - Simplicity, structure, extensibility, scalability

JXTA Search Benefits

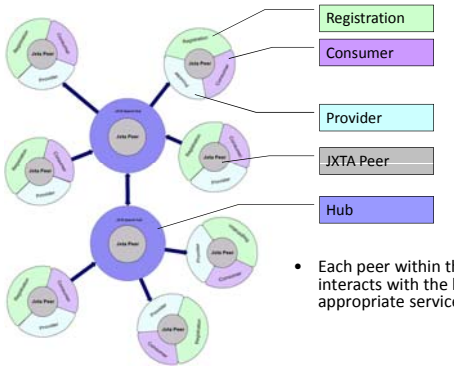
- Speed of update:
  - especially noticeable in deep search, where large data in databases are accessed directly without a need to create a central index.
- Access:
  - in crawling based approach many companies are resilient to grant access to web crawlers. In distributed approach the companies can serve the data as they feel appropriate.
- Efficiency:
  - no need to create a centrally placed and maintained index for the whole web.

JXTA Search Architecture



- Each JXTA peer can run instances of
  - Provider,
  - Consumer, and
  - Registration services on top of its JXTA core.
- Each peer interacts with the JXTA Search Hub Services, which is also running on top of the JXTA core.

JXTA Search Architecture (Cont'd)



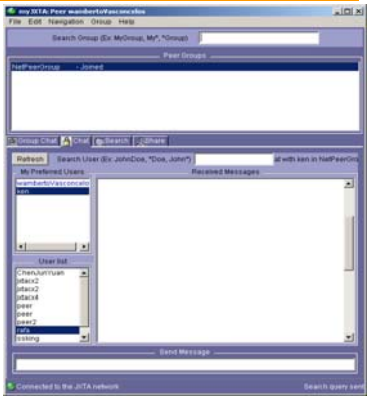
- Each peer within the network interacts with the hub using its appropriate service

Collaboration

- Over 25 companies developing JXTA projects
- Core (7 projects)
  - security, juxta-c, juxtaperl, pocketjxta
- Services (20 projects)
  - search, juxtaspaces, p2p-email, juxta-grid, payment, monitoring
- Applications (12 projects)
  - shell, jnushare, dfwbase, brando
- Other projects (5)
  - demos, tutorials, etc.

JXTA: myJXTA

- Peer interface
  - Windows
  - Unix/Linux
- Basic P2P services
  - Search
  - Chat
  - Download



Reading list

- JXTA Project. https://jxta.dev.java.net/
- Gnutella. http://www.gnutella.com

Books:

- Free downloadable book available at http://www.brendonwilson.com/projects/jxta-book/
- JXTA: P2P Computing with Java, Sing Li, 2002
- JXTA, Brendon J. Wilson, 2002
- JXTA: Java P2P Programming, Daniel Brookshire et al, 2002
- Mastering JXTA Development, Joe Gradecki, 2002
- Java P2P Unleashed, Robert Flenner et al, 2002
- JXTA in a Nutshell, Scott Oaks et al, 2002