
Bemnet Tesfaye Merha and Benoit Landrieu

{merha, landrieu}@stud.ntnu.no

Technical Essay in TTM3 Design of Self-Adaptive Systems
Norwegian University of Science and Technology
December 2, 2008

1 Introduction

JXTA (Juxtapose) is a hybrid Peer-to-Peer (P2P) platform for developing interoperable P2P applications. It’s an *open source* project initiated by Sun Microsystems. Unlike most P2P systems (1) (2) (3) and (4), which are mainly developed to for specific purpose, JXTA provides a generic framework for developing P2P systems on top of it. The framework consists of set of protocols which involve exchanging XML based messages making it possible to implement it in diversified programming languages. Current implementations are available using Java, C/C++, C# and J2ME (5). JXTA enable us to create a virtual overlay network which allows peers to communicate regardless of the underlying transport protocol. JXTA can be implemented on top of different transport protocols including TCP/IP, HTTP, Bluetooth and other protocols.

Current version of JXTA, 2.0, exists as a specification (5). The JXTA Specification was submitted to the IETF in 2002 in anticipation of participating with a working group. The IETF declined to assign JXTA to a working group, but instead referred JXTA to their sister organization Internet Research Task Force (IRTF). Since its fist introduction in 2001, JXTA has evolved and has gained significant assonance in the open source community. Although there are some performance issues (6), developers can benefit a lot by focusing on their application specific logic and leave the common P2P issues, like forming peer groups, discovering peers and services and traversing over firewalls and NATs, for the JXTA framework. This essay will take a closer look at the JXTA technology in general and focus on the dynamic nature of the technology enabling it to self-adapt to changes in the network.

The rest of the essay is organized as follows. In Section 2 we will present the basic concept of P2P networking. In section 3 we will review the architecture of the JXTA technology. In section 4 major components of a JXTA network will be presented. Section 5 addresses the JXTA core and standard service protocols. A unique search and discovery technique called SRDI will be explained in section 6. In section 7 we focus on two self-adaptability aspect of JXTA network. Section 8 presents evaluation of the performance of JXTA protocol under different message types and different transport protocol. Finally we will conclude with some final remarks.

2 Background

Peer-to-peer (P2P) networking differs from the conventional client-server networking in that participants of the network communicate directly with each other. In the traditional client-server architecture service is provided by a powerful central server, whereas in P2P based networking peers collaborate with each other to provide services. An important goal of P2P network is all peers in the network provide services and
share resources such as storage and computing power. Due to this collaborative behavior the performance and fault tolerance of a P2P network increase as the number of peer increase in contrary to client-server network (7).

As the number of peers in P2P network increase, there comes a need to effectively collaborate the contribution by different peers. P2P network should have a mechanism to uniquely identify peers and services provided by those peers. There is also a need for a mechanism to enable peers dynamically discover each other. Depending on the mechanism used for discovering peers, P2P networks can be categorized as pure or hybrid systems. Pure P2P networks don’t define any entity that centrally coordinates the discovery of peers. Peers in such network adopt decentralized flooding technique to discover peers. Typical examples of pure P2P systems are Gnutella (1) and Freenet (2). In both systems every node has a servant role (both client and server roles) and can equally communicate with any other connected node. On the other hand, in order to manage the scalability of the P2P network, hybrid systems uses a centralized router to coordinate the communication between peers. For example Napster (3) and Pointera (4) use a central server node to keep index of file shared by set of users. Compared to pure P2P systems, hybrid systems have better performance because some functionalities like searching and directory lookup are more effective if performed in centralized manner (8). JXTA is categorized under the hybrid systems, benefiting from centralizing some of its functionalities on its rendezvous peers.

3 The JXTA Architecture

The JXTA protocol consists of three layers namely the JXTA core layer, the service layer and the application layer. The JXTA core layer is where basic P2P networking functionality is implemented. It defines mechanisms for create peers and peer groups and building blocks to discover peers. It also handles transport protocol issues like firewall and NAT (9). The JXTA specification mandates that the functionalities specified in the core layer are expected to be implemented by all entities participating in the JXTA network.

The service layer includes services that are consumed by JXTA applications. Those services are not required for the protocol to operate but they are used by most P2P systems. Examples of such services include file sharing, distributed file systems, indexing and directory systems and authentication and Public Key Infrastructure (PKI) services. Figure 1 bellow shows how those services fit in to the three-layered architecture of JXTA.

The application layer consists of set of applications that can be developed on top of the infrastructure laid by the service layer and the core layer. Applications consume the services provided by the service layer and also add domain specific functionality. Examples of JXTA applications include P2P Instant Messaging (IM), P2P E-mail and distributed auction system.
JXTA protocol consists of number of peers which can be any device participating in the network. JXTA specification does not put any restriction on the type of transport protocol but mostly TCP/IP, HTTP, Bluetooth and GSM are used to build P2P applications. Peers in JXTA network collaborate to provide some kind of service and/or to share resources. Services are mostly actively running programs that are consumed by the other peers. Service in JXTA can be provided by a single peer or group of peers. An important aspect of the JXTA network is that services provided by group of peers are independent of each other and do not affect each other (10). If unexpected behavior is experienced in the service provided by specific peer, the peers in this peer group are responsible to handle this situation without affecting other services. In this sense we can say JXTA networks are self-healing. Compared to services, shared resources are owned by peer and are considered to be static in content. For example in a file sharing P2P application the files owned by the peers can be considered as a shared resource.

Due to lack of a centralized location containing list of services and due to the dynamicity of a P2P network, JXTA introduced an important concept of advertisement of services and resources. Advertisements are used to let peers discover new services and recourses. In JXTA, advertisements are XML documents that describe the services and resources and how can peers connect to those services. Advertisements describe peer information such as name, peer ID, properties, and services. The core advertisements types in JXTA are peer, peer group, pipe, service, content, endpoint and user-defined advertisements (9). Peers discover resources by searching for their corresponding advertisements, and may cache any discovered advertisements locally (9). The XML document that contains the advertisement is encapsulated in to a message which contains some more information to enable flow of advisements in the JXTA system. Among them is the envelope of a JXTA message which contains routing information, credential of information and the XML body. (Basic description of the JXTA messages is provided in section 4.2). Using advertisements peers can find out new services and can bind to them in order to
consume the service. This makes a JXTA network to easily adapt to inclusion of new services and requires less effort in manual configuration when setting up new services.

Messages in JXTA are communicated between peer using pipes and sockets. Sockets are reliable two way connections that provide communication between peers. Due to the overhead of establishing a socket communication, mostly JXTA uses pipe to send messages to one another. Pipes are unreliable unidirectional communication mechanism. Pipes have ending points that enable peers to send and receive messages. Pipe endpoints correspond to an available network interface for example TCP port number can be considered as an end point of a pipe for a peer on IP network (9).

4 Components of JXTA Network

In order to understand how JXTA network function and adapts itself to changes related to its participating peers, we have to first understand the basic components that make up a JXTA network. In this section we will present the major components of the JXTA network and then on the next section we will describe the actual protocols that give the self-adaptive nature of a JXTA system.

4.1 Peer Groups and Peer Group Services

Peer is a participant in a JXTA network and can be uniquely identified using Peer ID. Peers implement one or more JXTA protocols, to provide services and share resources. Those protocols require an addressing mechanism to communicate between peers. Therefore peers provide network address to identify peers end point. The peer end point address is not dependent on the underplaying transport protocol addressing, making the protocol implementation transparent. JXTA network consists of large number of peers that actively communicate in a P2P fashion. Considering the entire connected peers in the universe as one huge p2p network is impractical due to the number of peers and the available bandwidth and hence some sort of partitioning is required. Peer groups are used as a partitioning technique dividing the whole JXTA network into small manageable portions.

Peers can be self organized to form a peer group, if they agree on common set of services they provide. Similar to peers, peer groups use a peer group ID to uniquely identify themselves. Using peer groups in JXTA network enable us to easily enforce policies. For example one can define set to security policies that must be meet before a peer joins a group. The JXTA specification does not dictate how to maintain membership but rather provides a generic membership service as part of the core JXTA service as explained in section 3. The main advantage of using peer groups is to improve the scalability of the overall network. Peer groups enable segment the network in to abstract boundaries making it fairly easy to search the content of the group.

4.2 Messages

JXTA uses messages to communicate between services and applications. Messages are data units that are passed in the JXTA network when publishing, discovering and binding to services and resources. Message consists of a fixed format envelop and an arbitrary size body. The structure of a JXTA message is shown in the diagram below. The envelop of a message consists of a header, source endpoint, destination endpoint and message digest. The

![Figure 2: Structure of JXTA Message](image-url)
message digest is an optional part of the envelop which is used for security purposes. Depending on the context of usage and the security policies that are being enforced, the message body can also contain other attributes like credentials of the sender in order to provide enhanced security.

The main reason for using such a generic message format in JXTA is to allow flexibility in implementation. The specification does not enforce using transport protocol specific attributes in the message header, making the messages to be as generic as possible so that the protocol can be implemented on top of any transport protocol. Furthermore JXTA uses a standard message encoding of messages using XML. JXTA takes the advantage of the universal accessibility and programming ease of XML, which means JXTA can be implemented on different platforms with different programming languages.

4.3 Advertisements

Advertisements are also XML documents that describe peers, peer groups, pipes and services in JXTA network. JXTA protocols create advertisements to describe existence of resources so that peers can be able to search and possibly locally cache them. For example, a peer that obtained an advertisement for a peer group can use the advertisement to join the group. In order to control life time of advertisement without using a centralized control mechanism, advertisements have a life time associated with them. This let us to delete obsolete advertisement or extend the lifetime before they expire. Every advertisement consists of XML elements which can contain other elements or just data as shown below. The code snippet below shows an advertisement of a pipe. It is specified as a unicast pipe named pipe1. The pipe, like any other resource in JXTA is identified using a unique identifier using Universal Unique Identifier (UUID).

```xml
<?xml version="1.0"?>
<!DOCTYPE jxta:PipeAdvertisement>
<jxta:PipeAdvertisement xmlns:jxta="http://jxta.org">
  <Id>urn:jxta:uuid-9616261646162614A78746150325033E3A0244B79B848C3AE1A9DC5A015999EB03</Id>
  <Type>JxtaUnicast</Type>
  <Name>Pipe1</Name>
</jxta:PipeAdvertisement>
```

Figure 3: Advertisement of Unicast Pipe

Advertisement and any other messages between peers is transferred though pipes. Pipes are asynchronous, unidirectional, non-reliable virtual communication channel used to transfer messages between end points that do not have direct physical link. Pipe end points can for example be a TCP port with a corresponding IP address. Pipes in JXTA can take two forms; point-to-point pipes or propagate pipes (9). Point-to-point pipes connect exactly two pipe end points, where as propagates connect one output pipe end point to multiple input pipes. The figure below shows both kinds of pipes. As shown in Figure 6, propagate pipes are usually used along with rendezvous peers where there are multiple point-to-point links.
4.5 JXTA IDs
In order to avoid dependency on network address and naming services, JXTA identifies resources in the network using a global unique identifier. Peers, peer groups, pipes and contents are all identified using IDs. IDs provide addressing mechanism independent of the physical address of resources making it easily adapt to changes in the physical location of the peer. For example, a peer being assigned a new IP address by DHCP server every time it boots up can always have the same unique peer ID regardless of the different IP address it may be assigned by the server (11).

5 The JXTA Protocols
The JXTA protocols are used to let peers discover each other, discover network resources, communicate with peers and to route message in the P2P network. The protocols can be categorized in to two group namely core specification protocols and standard service protocols. A total of six protocols are grouped under these categories, but we will discuss two from each group in the following sub sections. The protocols are not applications by themselves, but instead they define some basic mechanisms that are common to most P2P applications making it easy to develop application on top of those protocols. The following sub sections describe the protocols in brief.

5.1 Core Specification Protocol
JXTA is designed to be implemented on diversified platforms ranging from main frame computers to small devices with low computing power. The mandatory functionality of the JXTA protocols is defined in the core specification and must be implemented by all peers participating in the network. However implementation of the core functionality alone does not guarantee interoperability with other peer. In order to be sure that we have full interoperability with other systems, implementation must include the other specific behaviors defined as standard services (11). Every JXTA peer must implement the following two protocols.

5.1.1 End-Point Routing Protocol (ERP)
ERP is basic routing protocol used to find sequence of routes to a destination peer. If peer A wants to send a message to peer C and if it has no direct route to C, then it will need to use ERP to find the route sequence to reach C. Route sequence consists of peer ID of relay peers that can be used to relay messages to a destination peer.
5.1.2 Peer Resolver Protocol (PRP)
PRP is a mechanism to send generic query to one or more peers and receive responses. Queries can be sent to a specific peer through unicast messages or can be propagated to multiple peers using rendezvous services. Each service running in a peer group that is willing to respond to a query will register its query handler in the peer group’s resolver service. When the resolver service receives query request, it will be forwarded to the defined handler to reply to the originator of the query (9). The functionality provided by PRP is a replacement to the resource resolution operations found in traditional distributed systems like DNS for resolving peer name to IP address, LDAP for binding a socket to a port and for locating services using a directory.

5.2 Standard Service Protocols
To ensure full interoperability of JXTA systems, peers should implement additional set of protocols. The standard service protocol defines four protocols including Rendezvous Protocol (RVP), Peer Discovery Protocol (PDP), Peer Information Protocol (PIP) and Pipe Binding Protocol (PBP). We will only present the first two protocols and the detail for the last two can be found in JXTA specification (5).

5.2.1 Rendezvous Protocol (RVP)
RVP enables peers to subscribe and to propagate messages to other peers in a peer group. RVP provides a mechanism to propagate messages to be performed in a controlled and efficient way. The message propagation mechanism is implemented by all rendezvous in a peer group. Another use of this protocol is to create consistency between rendezvous. Rendezvous peers in a peer group keep track of each other using the PeerView protocol to self organize themselves, maintaining consistent common view (11).

5.2.2 Peer Discovery Protocol (PDP)
PDP is a mechanism by which peers advertise about themselves and receive advertisement from other peers. Advertisements can belong to resources including peers, peer groups, pipes and services. Unless the peer group defines its own discovery mechanism, PDP is the default discovery mechanism. Current implementation of JXSE uses combination of IP multicast in local subnet and rendezvous maintained in the Distributed Hash Table (DHT). The technique of dynamically resolving advertisement queries is called Shared Resource Distributing Index (SRDI) and will be presented in the next section in detail.

6 Search and Discovery in JXTA using SRDI
An important aspect of JXTA network is the ability to propagate advertisements of resources without need for centralized control. Shared Resource Distributing Index (SRDI) provides a mechanism that allows rendezvous peers to maintain an index of advertisements so that they can reply to query requests from edge peers. When an edge peer has a resource to advertise; it publishes the index of the advertisement on to its rendezvous peers using the SRDI service (9). In previous implementation of JXTA, rendezvous peers were required to maintain cache of advertisements from edge peers. However this poses a problem with regard to managing out-of-date cache entries. As the number of advertisement grow managing expired entries was observed to degrade the overall system performance. Therefore the use of advertisement cache was replaced with SRDI in JXTA 2.0.
6.1 Advertisements using SRDI

By using SRDI significant improvement in performance can be achieved. Because advertisements are only propagated by rendezvous peers, queries requesting for advertisements are sent only to those peers, reducing the number of peers involved (12).

In the initial phase of SRDI, the edge peer computes a hash value of the advertisement, and it will be indexed with the Peer ID of the edge peer. The index value will be published on the rendezvous peer to help it map queries to the originator of advertisements. Upon receiving the advertisement, the rendezvous uses DHT to map the index to other rendezvous in its local list and propagate the index to those peers as shown in Figure 6. Rendezvous peers maintain list of their known rendezvous with which they share advertisements. In the above example, R2 will propagate the advertisement to R3, R4 and R5. This list is initialized using seeding and bootstrapping and is also updating by randomly exchanging the list within a peer group.

6.2 Queries using SRDI

After the advertisement is propagated throughout the rendezvous peers, edge peers can start looking for resources. The peer will compose a query containing the specification of the advertisement the peers are interested in and send it to the local rendezvous peer. The rendezvous will compute the hash value of the advertisement in order to map the query to the original rendezvous that forwarded the advertisement. In the example presented below, peer P3 forwards the query to its rendezvous R5 and it will forward the query to the original rendezvous, in this case, to R2, which will in turn send the query to the peer P1. Because the query contain the peer ID of P3, P1 is now able to respond directly to P3 providing the required information to utilize the advertised resource as shown in Figure 7.
7 Self-Adaptability Aspect of JXTA

The ability to self-organize, self-configure and self-heal are important aspects of JXTA network. One interesting aspect is the formation of peer groups. The JXTA specification outlines that peer groups are formed and self-organized based on the mutual interest of the peers (5). By using peer group advertisement, peers of the same interest will organize themselves into a peer group without the need for external control. We have mentioned that services in JXTA can be installed as either peer or peer group services. If the peer providing a peer service goes down, the whole service will be terminated. However, if the service is provided as peer service, multiple instances of the same service can be installed on multiple peers in a peer group (13). We can say that JXTA network is self-healing because this fault tolerance and high availability of JXTA is achieved through the capability of peers to self-organize into peer groups.

Another self-healing aspect of JXTA network is the use of rendezvous peers. In section 7 we presented the importance of rendezvous peers in propagating advertisement and queries using the SRDI protocol. If the rendezvous peer in a peer group goes down it will be impossible to propagate advertisements and queries. JXTA addresses this issue by enabling promotion of edge peers to act as rendezvous peers, if a dedicated rendezvous peer is not available. This is done using the rendezvous protocol in order to ensure peer groups which use the rendezvous service will always have a rendezvous peer.

8 Performance Evaluation of JXTA

Over the years the JXTA protocols have undergone a series of changes with the aim of improving its performance and scalability. Careful study of those improvements is valuable for developers in order to help them decide whether or not to adopt the new features in their distributed applications. However, studying the performance of JXTA protocol requires setting up a complex benchmark simulating different traffic loads, scalability in different group size and reliability and failure recovery (12). Here, we review
the performance measures done on mainly focusing on (6) estimating the Round Trip Time (RTT) of different JXTA messages and over transport protocols.

The performance study presented by E. Halepovic and R. Deters (6), uses a benchmark with peers running on 800MHz Athlon CPU with 512MB connected with each other using 100Mbit LAN. The study tries to understand the behavior of JXTA pipes under different message types and transport protocols. The Round Trip Time (RTT), the time it takes to send a message and receive an acknowledgement, is used as a means of comparison. The RTT was measured when transmitting message with payload ranging from 1KB to 10MB through unicast and propagate pipes. All the three pipes showed a linear increase in RTT as message size increases. To the contrary, in the new version of JXTA 2.0, significant performance degradation is observed with message size 1MB to 10MB over unicast and propagates pipes up to factor of two. This degradation in performance is claimed to be due to the new message composition technique introduced in JXTA 2.0 (6). Therefore this is a very good indicator that a careful study should be performed identifying the expected performance degradation and then determining if the application would befit from the newly introduced features before major upgrade is performed on deployed systems.

As presented in section 1, JXTA is transport protocol is transparent. TCP/UDP is mainly used when there is a direct connection between peers, and HTTP is used when edge peers are behind NAT/Firewall. The performance of peers differ depending on which protocol is used. For example, TCP/UDP configuration showed the lowest RTT when used between peers directly connected without any rendezvous or relay peers used. Whereas when HTTP and TCP with IP multicast disabled is used between two peers communicating through a relay RTT, increased for all the pipes. This is mainly due to the overhead of the relay peer. Compared to unicast and secure pipes, the latency in propagate pipes increased by a factor of four. This is due to the absence of IP multicast and it showed that the use of IP multicast has significant improvement for propagate pipes. Although JXTA is promoted as transport protocol independent, the choice of transport significantly affects the implementation of distributed applications. Therefore, one should do performance analysis before choosing a particular transport protocol.

9 Conclusion

Due to the distributed nature of P2P systems, there is an interest for a common framework using which we can develop services. Such a framework need to be self-adaptive to the dynamic nature of P2P network. JXTA is such a framework that provides building blocks to develop P2P systems on top of it. Unlike most protocols, JXTA is designed to be platform, language and transport protocol independent. This make it possible to develop P2P systems that can be executed on devices ranging from large super computer to small hand held devices. In this essay we have described the building blocks that are required to develop such a system and described the various protocols that are required for the JXTA network to function. We have also addressed how peers can organize themselves in to a peer group in order to provide reliable redundant service. Although it is being improved currently, the introduction of SRDI protocol, in JXTA 2.0, showed a significant performance improvement in propagating resource advertisements and queries. The JXTA community is currently working on improving the limited walker algorithm, enhancing the overall performance of the JXTA system (14). All in all we can for see that the use of JXTA in developing interoperable P2P application will be a natural choice in the near future.
References


