JaDE: A JXTA Support for Distributed Virtual Environments

Luca Genovali, Laura Ricci
Università degli Studi di Pisa

IEEE Symposium on Computers and Communications
Marrakesh, July 2008
DISTRIBUTED VIRTUAL ENVIRONMENTS

Real-Time Distributed Virtual Environments (DVE):

• provide to geographically distributed end-users the illusion of being immersed in a unique shared virtual world

• real time interactions among users and/or among users and computer controlled entities

• Examples: distributed multiplayer games, military simulations

• Architectural models: central server, P2P, hybrid

Multiplayer Games:

• a set of entities (avatars, monsters, tanks,...) populate a virtual world

• each entity communicates its state (change of position, colour), or virtual world modifications to other participants
DVE: DESIGN CHALLENGES

Design challenges for a scalable DVE implementation

- mechanisms to guarantee the consistency of the virtual world
- synchronization, state replication
- **real time constraints**: the action performed by an entity should be visible to other entities within a bounded interval of time
- scalability
  - consistency requires the exchange of a huge amount of information among end users
  - high bandwidth, CPU load
  - optimizations required to minimize the exchanged information
DVE: INTEREST MANAGEMENT

• Interest Management: reduce DVEs communication requirements.
• Area of Interest (AOI) of an entity E = portion of the virtual world including entities that may interact with E
  – a player interacts with active entities (players, monsters) and/or passive objects located in its surroundings, e.g. in the same room.
• The definition of the AOI of E depends upon the semantics of the application, e.g. the sight capability of E
• E is interested in receiving information from entities in its AOI only
• AOI implementation:
  – Multicast groups
  – Publish-subscribe systems
  – JXTA groups
JaDE: DESIGN CHOICES

• JaDE: A Distributed Virtual Environment (DVE) support exploiting JXTA constructs
  • Manages different kind of events:
    ▪ position updates (heartbeats)
    ▪ passive objects modifications
  • Statically decomposes the DVE into a set of regions
  • Position updates: each peer periodically
    ▪ sends its position to any other peer in its region
    ▪ receives the position of the other players in its region
  • Passive objects management. Challenges
    ▪ consistency
    ▪ persistence
    ▪ replication
JaDE: AOI DEFINITION

• JaDE statically partitions the DVE into a set of regions

• The shape and size of the regions depends upon the application

• Currently, each region includes a central zone (C) and a set of peripheral zones (N, NW, NE,...)

• The AOI of a peer P overlaps the region R where P is located (peer=black circle) \( \text{AOI}(P) = R \)

• Peripheral regions are introduced to implement prefetching
JaDE: AOI DEFINITION

- When a peer enters a new region $R$ it initializes its state with active/passive entities $\in R$.
- Prefetching avoids a delay in the acquisition of the new state. When $P$ is approaching the border of its region,
  - a set $S$ of neighbour regions are included in $\text{AOI}(P)$
  - before entering a region $R \in S$, $P$ starts to collect the state of $R$ and to notify its presence to peers in $R$.

Example: The AOI of the peer (black circle) located in the southern peripheral region is augmented by the southern neighbour region.
JaDE: STATE PREFETCHING

<table>
<thead>
<tr>
<th>NW</th>
<th>N</th>
<th>NE</th>
</tr>
</thead>
<tbody>
<tr>
<td>W</td>
<td>W</td>
<td>E</td>
</tr>
<tr>
<td>SW</td>
<td>S</td>
<td>SE</td>
</tr>
<tr>
<td>NW</td>
<td>N</td>
<td>NE</td>
</tr>
</tbody>
</table>

- prefetching of smaller region may be defined
- any region may be paired to a JXTA group
PASSIVE OBJECTS: REPLICATION

- Each peer holds a map of the whole DVE including immutable objects (landscapes, trees,...) and objects which may be dynamically modified.
- The first peer that updates an object O activates the object:
  - allocates a data structure to store the state of O
  - replicates the state of O to each peer in its AOI. Each peer stores the objects of the region in a local cache.
- Passive Object Replication improves:
  - DVE reliability. An object replicated on a set of peers is not lost if one of them crashes.
  - DVE responsiveness. Object state is available in the local cache and should not be requested to a single peer.
- A set of mechanisms should be defined to guarantee the consistency of concurrent updates.
PASSIVE OBJECTS: PERSISTENCE

- **Object persistence**
  - do not lose the state of objects in regions which are not inhabited
  - JaDE defines a **Backup Peer** which takes charge of the objects in regions that are not inhabited

- **The Backup Peer of a region $R$** is
  - the last peer $P$ leaving $R$
  - $P$ delegates the management of passive objects of a region to a peer $P'$, if it leaves the DVE. $P'$ may be chosen randomly among the peers of the DVE and takes charge of the objects of $R$.

- A set of **Backup Peers** may be defined to avoid object loss in presence of abrupt peer crashes
PASSIVE OBJECT: PERSISTENCE

- The identity of the BackUp peer of a region \( R \) should be notified to any peer of the DVE

- When a peer \( P \) enters a region \( R \)
  - if \( R \) is inhabited, \( P \) contacts a peer \( P' \in R \) randomly choosen to receive the objects of \( R \)
  - otherwise \( P \) contacts the Backup Peer of \( R \) and takes charge of the object of the region

- JXTA constructs allow a straightforward implementation of the BackUp mechanisms.
PASSIVE OBJECTS: CONSISTENCY

- **Update Area (UA)** of an object O - portion of the DVE where a peer P must be located to update O
- The size of this area depends upon the game semantics and on object/peer characteristics
  - a peer must be close to a potion to drink it.
  - a peer may be far from a bottle if it throws a stone to break it.
- **Concurrent Updates** may occur when several peers are simultaneously present in the update area of an object O
- Notification of modifications occurring in the UA of an object may not reach a peer P located outside UA before P enters UA, because of
  - High network delay
  - High peer speed
JaDE: CONFLICT AREA

- **Conflict Area (CA)** of an object $O$: zone of the DVE centered at the object where concurrent updates may occur.

- The radius of Conflict Area $CA$ is defined so that
  
  *the time to reach the UA of $O$ from any point in $CA$ is smaller than the time to notify an update to any peer of the region.*

- **iff** the conflict area of $O$ does not include further peers $A$, $P$ may update its local copy of $O$ and notify the updated value of $O$ to other peers in its AOI (in its region).

- otherwise a **consistency mechanism** is required.

- **JaDE** consistency protocol
  - exploit the spatial location of the peers and the knowledge of the network delays to detect concurrent updates
  - define a **coordinator** for each object to serialize concurrent updates
JaDE: OBJECT CONSISTENCY

- Coordinator of an object: peer which creates or activate an object
  - A timestamp mechanism resolves the conflicts to acquire the coordination

- When a coordinator exits a region, it delegates the coordination of the object to another peer

- The coordinator
  - serializes concurrent updates on an object
    - resolves the conflicts among the peer in the conflict area
A JXTA SUPPORT FOR JaDE

- JXTA: a distributed platform for the development of P2P applications generally exploited for classical applications like file sharing.
- Our goal:
  - to exploit JXTA protocols to support JaDE
  - to investigate the effectiveness of JXTA for the development of DVE
  - to evaluate different JXTA protocols in order to evaluate the best one

- discovery protocol
- pipe binding protocol
- resolver protocol
A JXTA SUPPORT FOR JaDE

- Notification of events is restricted to peers belonging to the static region by pairing each static region of the DVE with a different JXTA group.
- Alternative JXTA solutions to support heartbeats notification:
  - Propagate pipes:
    - A pipe advertisement is published by the first peer entering a region R. The pipe is exploited to exchange heartbeats.
    - Each peer entering R discovers the pipe through JXTA Discovery Protocol.
  - JXTA resolver protocol:
    - Heartbeat notification is sent as an XML query.
- Experimental results show the propagates pipes overcome resolver protocol.
A JXTA SUPPORT FOR JaDE

- JXTA caches store the state of replicated objects

- When a peer activates/modifies an object $O$, it publishes an advertisement describing $O$

- Peers periodically search for advertisement notifying object modifications through the JXTA Discovery protocol

- A low expiration time is associated to the object advertisements in order to avoid repeated fetch of unmodified objects
Implementation of persistence mechanism

- the last peer P (backup peer) leaving a region R copies the objects of R into its local cache
- P publishes a backup pipe advertisement identified by the name of R and binds the corresponding pipe in input
- any peer of the DVE may contact the backup peer by discovering the backup pipe advertisement
- JXTA protocol exploited by the persistence mechanism
  - JXTA Discovery Protocol to find out the Backup pipe
  - Pipe binding Protocol to bind the Backup pipe and contact the Backup peer
A single propagate pipe is exploited by 20 peers to notify their position.
Frequency of notification: 200 ms.
One peer implements a ping-pong mechanism by acting as a reply peer.
Round trip Time measured by one peer: 8.57 ms.
Heartbeat notification latency is compatible with the real time requirements of the DVE.
RESOLVER PROTOCOL EVALUATION

• The Resolver Protocol is exploited to notify heartbeats
• Frequency notification: 200 ms
• Average latency = 17.43 ms, maximum latency 802 ms
• The pipe binding protocol outperforms Resolver Protocol
OBJECT DISCOVERY EVALUATION

- a region $R$ includes 20 peer and 20 objects
- each peer
  - discovers all the objects of $R$ through the discovery protocol
  - evaluates the average time required to discover all the objects
- average Discovery time: 2.7 seconds
- Prefetching mechanism may hide the discovery delay
A preliminary evaluation on a wide area network: 3 peer exploiting a propagate pipe to notify their position

Average latency: 400 ms

Acceptable latency for a DVE
CONCLUSIONS

- JaDE: a JXTA Support for DVE
- At best of our knowledge, the first proposal to exploit JXTA technology for DVE
- Future works
  - definition of dynamical area of interest
  - hierarchical solutions: static region with dynamic AOI within each region
  - Evaluation on a WAN