Motivation

- Framework for distributed VR
- Well known programming model
  - scene graph API
  - data flow network for event propagation
- Transparent support for distribution
- Extensibel
- Maximum performance
System overview

Application

Avocado

Performer  Maestro  Elk Scheme

Linux / SGI IRIX

Object representation

- Based on SGI Performer
- Object attributes as 'Fields'
- Objects are Field-Container
- All objects support:
  - container-interface
  - persistence-interface

fpFile
Filename
Matrix
Children
Object class hierarchy

- fpFieldContainer
  - fpNode
    - fpGroup
    - fpDCS
    - fpSwitch
    - fpLOD
    - fpSequence
  - fpFile

Event model

- 'Field connections'
  - Fields of similar type
  - 'Copy-on-write' semantics
- Data flow graph
  - Evaluated once per frame
- Object notification
  - fieldHasChanged(fpField&)
  - fieldHasChangedLocalSideEffect(fpField&)
  - evaluate()
  - evaluateLocalSideEffect()

<fpFC.> = instance of a class derived from <fpFieldContainer>
Typical application structure

Distribution

- Goals
  - Known programming model
  - Transparent support of distributed application
  - Dynamic group membership
  - Distribution of all object attributes
    - not only transforms
- Distributed scene graph
Client/Server

- Process 1
- Process 2
- Process n

Network

Database Process

Replication

- Process 1
- Process 2
- Process n

Local Database Copy

Network
Distribution in groups

Distribution Group

Process A
- a
- b
- c
- d

Process B
- a
- b
- c
- d

Replicated Objects
Local Objects

Object replication

Process A

fpObject
Field 1
Field 2

Network

Process B

fpObject
Field 1
Field 2

fpObject:Field1=Value1:Field2=Value2
fpObject:Field1=Value1:Field2=Value2
Distributed scene graph

Event distribution
Total message ordering

<table>
<thead>
<tr>
<th>Process A</th>
<th>Process B</th>
<th>Process C</th>
<th>Process D</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>3</td>
<td>2</td>
</tr>
</tbody>
</table>

Consistence

- setValue()
- distributeValue()
- notify()
- receiveValue()
- sendMsg()
- receiveMsg()
Atomic state transfer

Time

Process A  Process B  Process C

Join Operation

Atomic state transfer

Time

Process A  Process B  Process C

Leave Operation
Group communication

- Maestro
  - Toolkit for group communication
- reliable communication
  - IP unicast, IP multicast
  - ATM
- Dynamic group membership
- Total message ordering