

Developing a Shared Presentation Viewer



Retreat 2003

Overview

- Review Interfaces
- Conceptual Overview
- Design Overview
- Code Walkthrough
- See it Work



Retreat 2003

Review Interfaces

- Application Factory
 - Found in each Virtual Venue
- Application Object
 - Found in Virtual Venues



Retreat 2003

Venue Application Factory

def CreateApplication(name, description, mimeType, eventService):

Name – Application Name

Description – Application Description

mimeType – Application Mime-Type

eventService – Reference to the Event Service for the Venue Server.

- Creates Application
 - Creates a Data Channel for the Application
- Makes it Available in the Venue as Web Services



Retreat 2003

Application Object

- Provides Web Services Interface
 - Join(publicId)
 - Leave(token)
- Provides Data Channel among clients
 - GetDataChannel(token)
- Can hold state for the Application
 - SetData(token, key, value)
 - GetData(token, key, value)



Conceptual Overview

- Everyone in a Venue can participate in a Shared Presentation
- There is only one Master who can send valid events
 - usually the speaker (but not always)
- Have to be able to get the presentation
- If you come in during the talk you want to find out what slide everyone is on



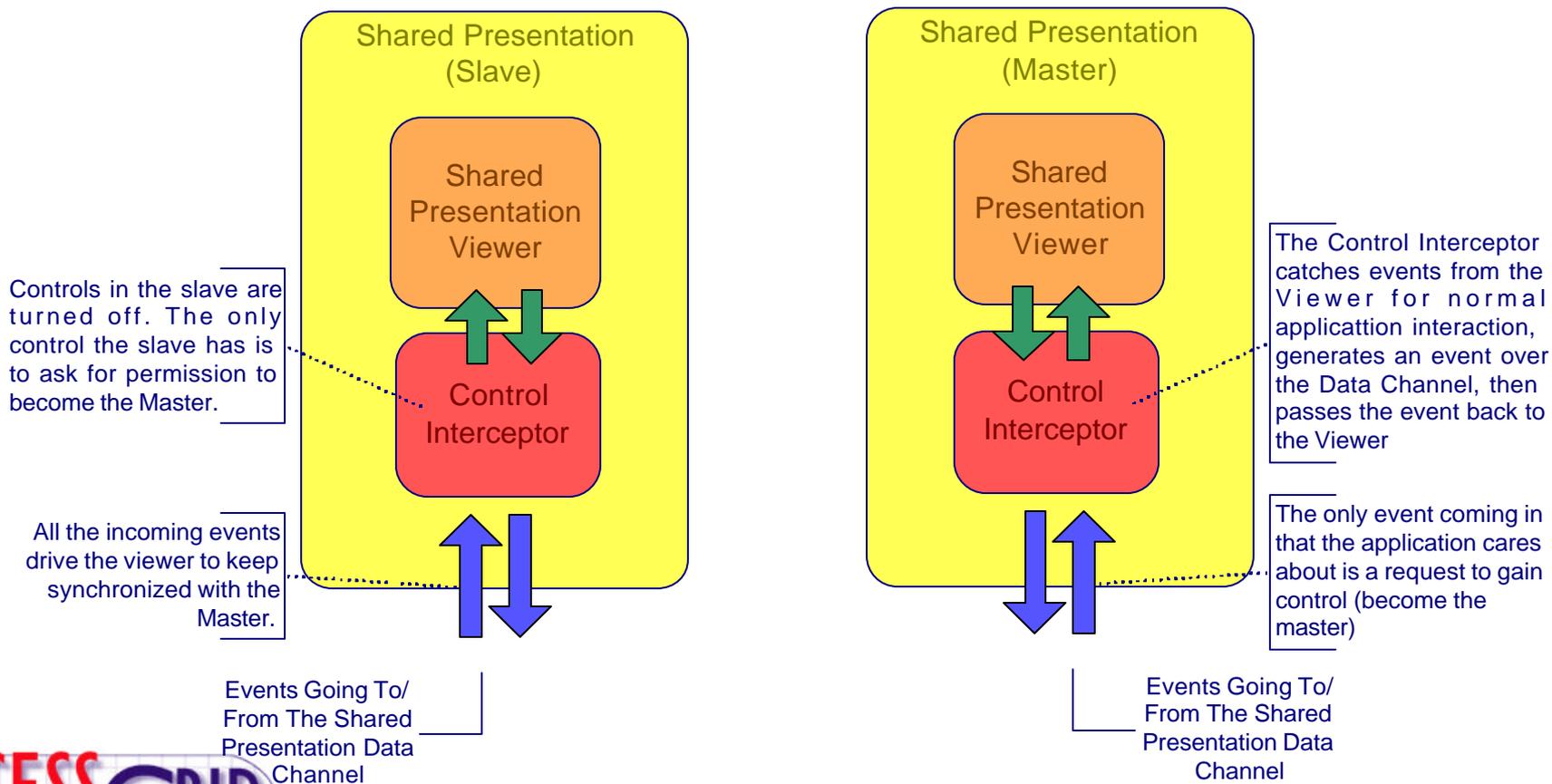
Design Overview

- Ideal Design
- Practical Design
- What the parts do
- How they talk to each other

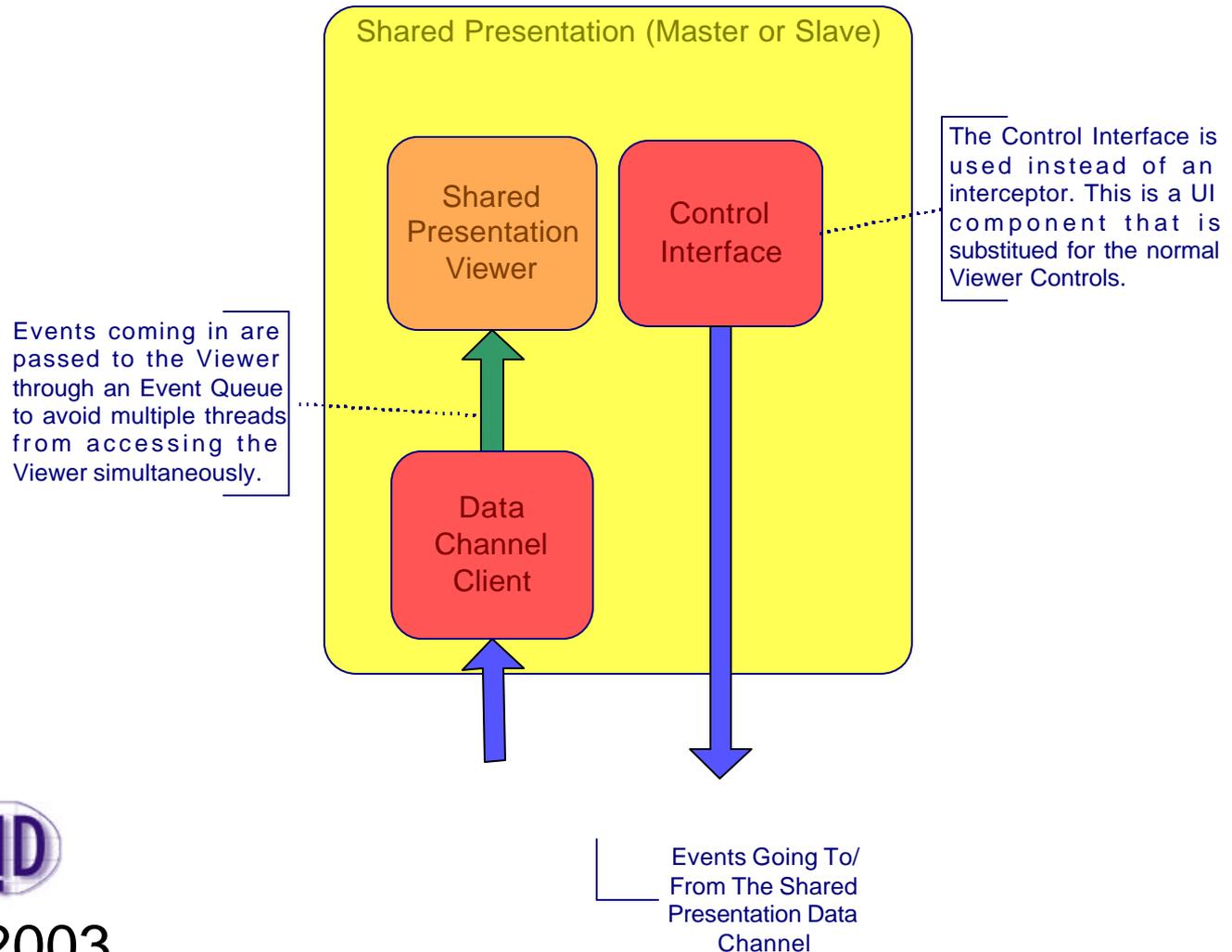


Retreat 2003

Ideal Design



Practical Design



Shared Presentation

- The Shared Presentation implements the Shared Application Framework.
- During Initialization it:
 - registers callback functions for events
- Then it creates:
 - A Shared Presentation Viewer
 - A Controller Interface
 - An Event Queue to pass events between threads
- Then the main thread spins processes events pending on the event queue.



Data Channel Client

- Every Shared Application has a Data Channel.
- The Data Channel is used as like a “shared bus” for data among the Application “clients”
- To use the Data Channel the Shared Application registers Callback functions with the Data Channel
- The callbacks are registered with the events they correspond to.



Shared Presentation Viewer

- The Shared Presentation Viewer is the most platform dependant part of the Shared Presentation Application.
- On Windows we are using the PowerPoint Application (via the python/COM interface)
- On Linux this could be an instance of the OpenOffice presentation application (via XML-RPC or CORBA)
- This Viewer needs to be able to read in the data that is specified by the Application Object residing in the Venue.



Control Interface

- The Control Interface is a substitute for the actual controls on the Viewer.
- It provides basic functionality for:
 - Next
 - Previous
 - Go To slide #
 - Load presentation
 - Quit



See it Work!



Retreat 2003