

Overview

- Shared Applications
- Shared Browser



Retreat 2003

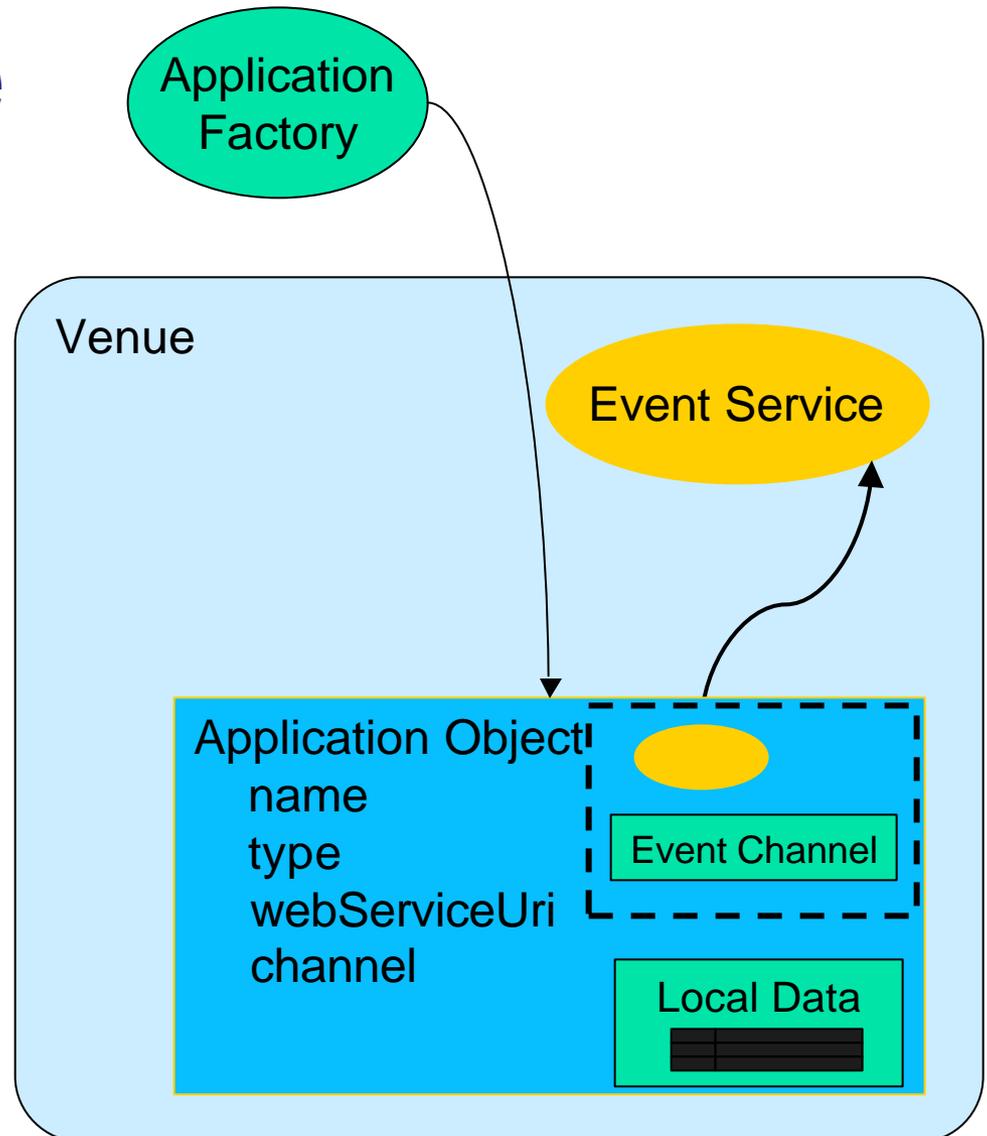
Why Shared Applications

- Past application-sharing success
 - DPPT
 - Big win for remote collaboration
 - Not integrated with the venue
- Goals
 - Enable users in a Venue to share applications
 - Integrate startup with the venue client
 - Allow storage of application state in the venue
 - Provide coherent state through event distribution



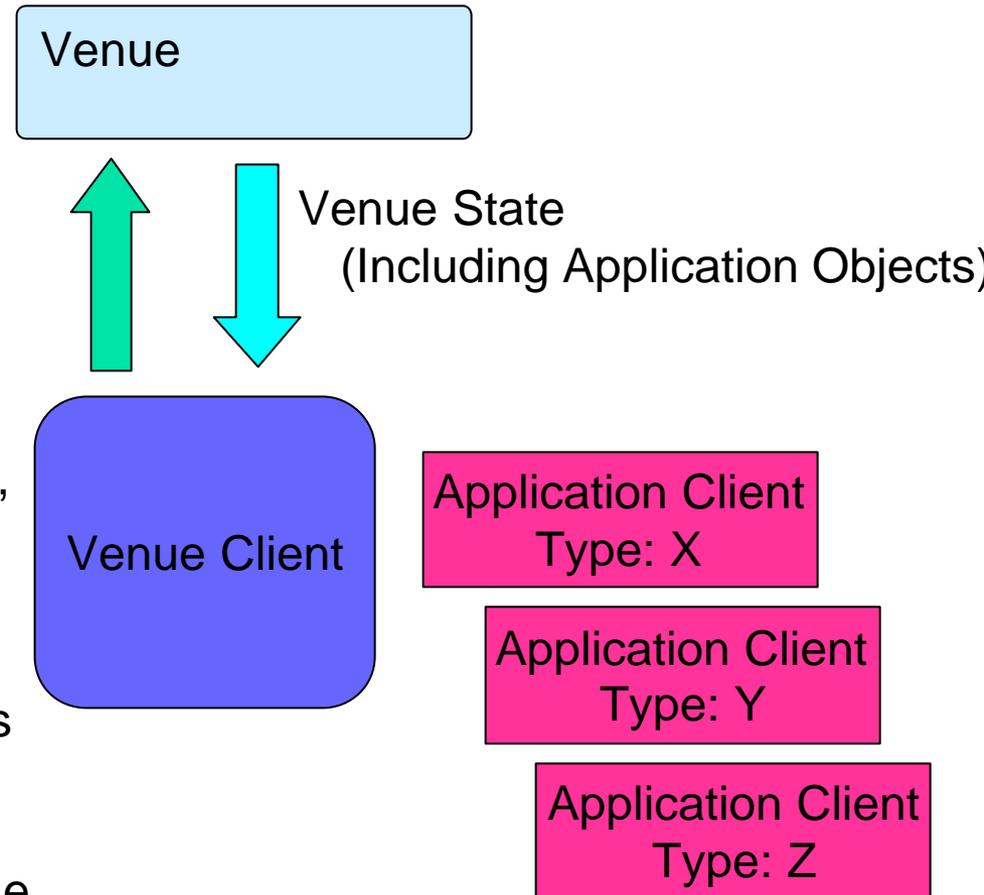
Application Architecture: Venue Side

- An *Application Factory* creates Venue-resident applications.
- Each application is represented in the venue by an *Application Object*
- An application object can store *local data* and can have one *event channel*
- Event channels utilize a Venue-based *Event Service*



Application Architecture: User Side

- On the user side the Venue Client is the key.
- The user can install applications, which are then available to the Venue Client.
- When a user enters a Venue, if there are application objects, the Venue Client looks for applications that are of the same type.
- The Venue Client also enables the user to start a local application, and create the Application Object in the Venue



Getting of Initial State (Optional)

- Application Objects that maintain state can provide access to that state via a Web Services Interface.
- The Application Client is responsible for storing/retrieving the state via the Application Object.

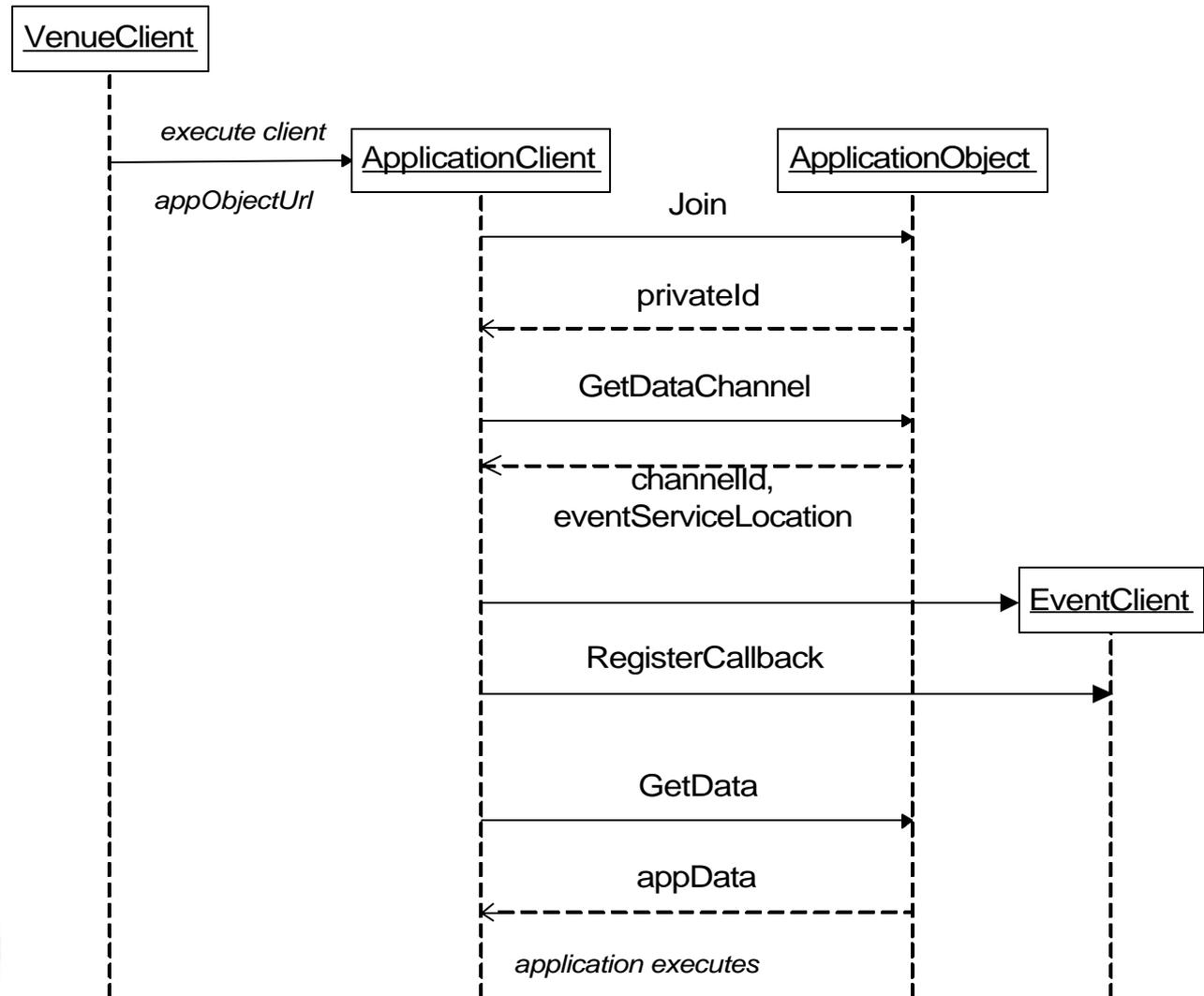


Applications Interface

- Venue
 - CreateApplication : appObjUrl
 - DestroyApplication
- ApplicationObject
 - Join : privateId
 - Leave
 - GetDataChannel (channelId, eventServiceLocation)
 - GetData
 - SetData



Typical application startup

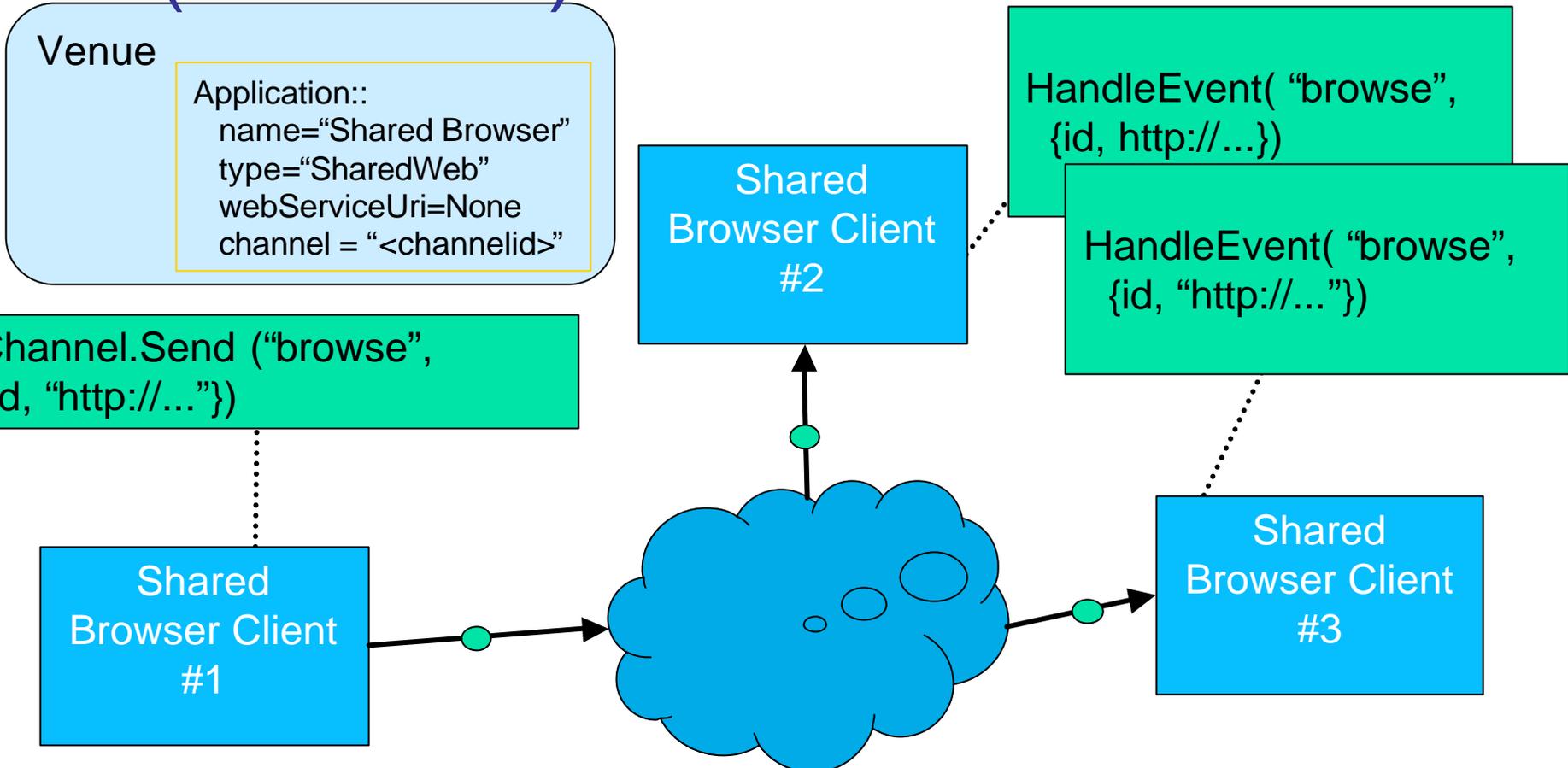


Shared Web Browser

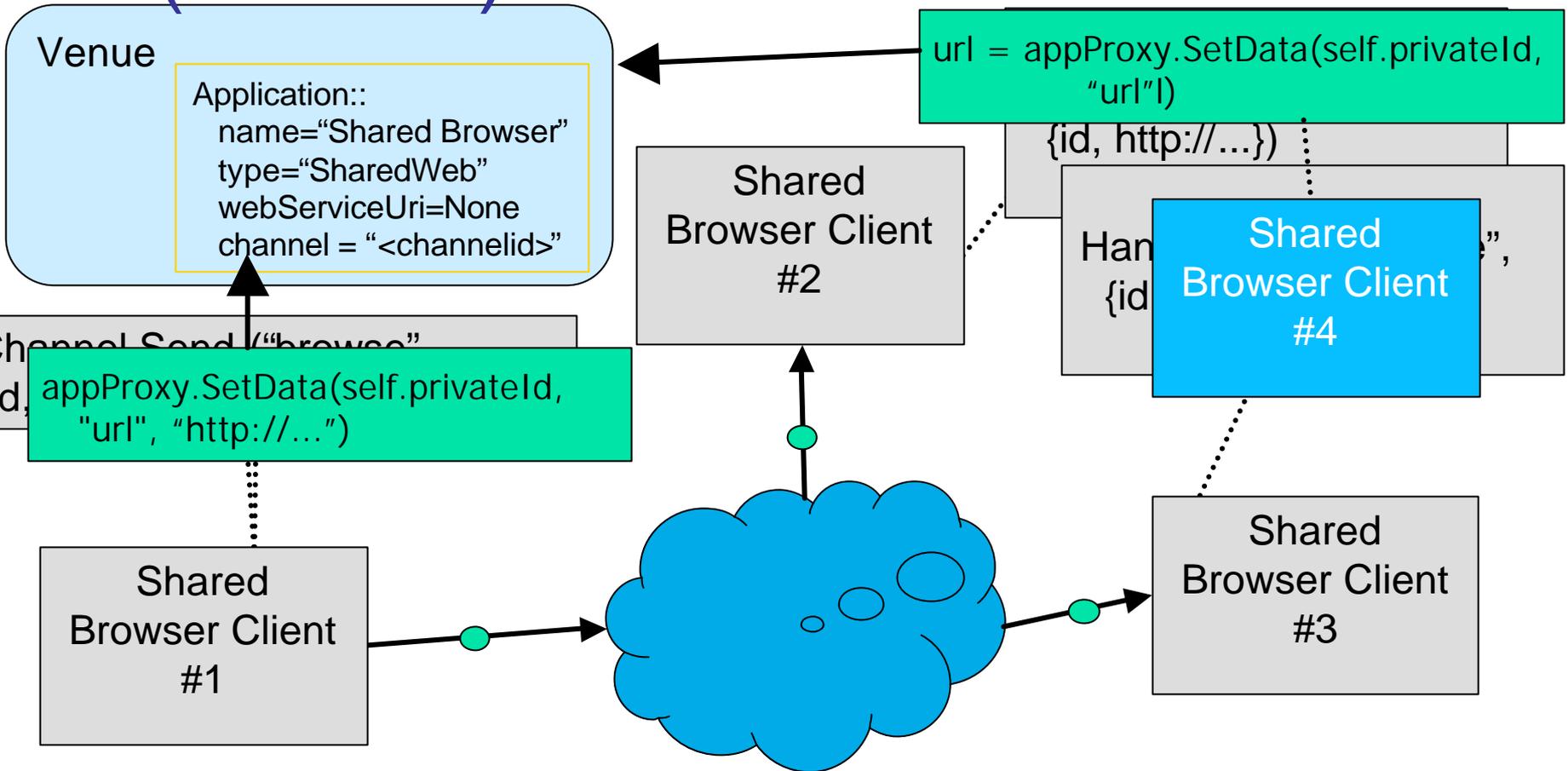
- Application task: Web browsing
- Goal: All users see the same page
- The Venue serves as a rendezvous mechanism
- Application state: webpage URL
- State is distributed; that is, there is no central server maintaining the state
- With each state change, an event is distributed to all interested clients



Shared Web Browser Architecture (stateless)



Shared Web Browser Architecture (stateful)



Shared Browser sample

Browser -- Futures Laboratory
Location: <http://www.unimcs.anl.gov/>

Computing and Communications Infrastructure

Futures Laboratory

The Argonne Futures Lab performs basic and applied research in advanced communications, collaboration, and visualization technologies (e.g., teleimmersion) to enable the development of wide-area collaborative computational science. Our focus is on the development and evaluation of high-end technologies and systems that extend and complement commercially available tools and resources.

[news](#)
[research](#)
[publications](#)
[people](#)

Futures Laboratory MCS ANL

For more information: fl-info@mcs.anl.gov



The Point is...

Shared application development is easy.

- Assuming the existence of a suitable interface, one need only write a small amount of glue code
- Among the AG2 deliverables is a document describing this development

