Extending Web Applications with Client and Server Plug-ins

Jahn, Wolfinger, Mössenböck
{jahn, wolfinger, moessenboek}@ase.jku.at
Johannes Kepler Universität, Linz
**Goals**

*Instead of a monolith ...*

... we want a thin core that can be extended by plugins

- Every user just loads what he/she needs
- Dynamic reconfiguration for different usage scenarios
- Dynamic plugging without restarting the application
- Dynamic unplugging without restarting the application
- Plug & Play extensions without programming or configuration
- **Component-based web applications**
- **Multi-user composition model**
- **User-specific web extensions**
Plux.NET component model

Extensions, Slots and Plugs

Slot and Plug in detail
Logger sample

public class MyApp : IStartup {
    ILogger logger = null;       // the logger extension
    string timeFormat;            // parameter of the logger extension

    public void AddLogger(object s, PlugEventArgs args) {
        logger = (ILogger) args.Extension;
        timeFormat = (string) args.GetParamValue("TimeFormat");
    }

    public void RemoveLogger(object s, PlugEventArgs args) {
        logger = null;
        timeFormat = null;
    }

    public void Run() {
        ...
        if (logger != null)
            logger.Print(DateTime.Now.ToString(timeFormat));
    }
}

public class ConsoleLogger : ILogger {
    public void Print(string msg) {
        Console.WriteLine(msg);
    }
}
Extending Web Applications with Plux.NET
Remote Plugging
The **Plux.NET** web runtime

The web-runtime...

... is distributed

... replicates composition state on all runtime nodes

... manages multiple user sessions

... persists composition state during round-trips
### Case Study - Time Recorder

**User:** Markus Jahn

**Date:** 08.04.2009

**Begin:** 07:32

<table>
<thead>
<tr>
<th>Date</th>
<th>Start</th>
<th>End</th>
<th>Presence</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mi 01.04.2009</td>
<td>07:24</td>
<td>17:26</td>
<td>10h 02m</td>
</tr>
<tr>
<td>Do 02.04.2009</td>
<td>07:58</td>
<td>16:15</td>
<td>08h 17m</td>
</tr>
<tr>
<td>Fr 03.04.2009</td>
<td>07:36</td>
<td>12:18</td>
<td>04h 42m</td>
</tr>
<tr>
<td>Mo 06.04.2009</td>
<td>07:45</td>
<td>18:17</td>
<td>10h 32m</td>
</tr>
<tr>
<td>Di 07.04.2009</td>
<td>07:14</td>
<td>15:37</td>
<td>08h 23m</td>
</tr>
</tbody>
</table>

**Sum:** 17h 56m
Time Recorder Architecture

Environment

Visibility Scope

<table>
<thead>
<tr>
<th>Server</th>
<th>Silverlight</th>
<th>Client</th>
</tr>
</thead>
<tbody>
<tr>
<td>Global</td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>Group</td>
<td></td>
<td></td>
</tr>
<tr>
<td>User</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Rich Internet Time Recorder

Environment

Visibility Scope

- Server
- Silverlight
- Client
  - Global: ✓ ✓
  - Group
  - User
Server-side Extensions

Environment

Visibility Scope

Server
Silentlight
Client

Global ✓ ✓ ✓
Group ✓ ✓
User

Time Recorder

User: Markus John
Date: 08.04.2009
Begin: 07:32

Plus.NET Server

- Server Runtime
- Framework
- Besprechung
- Präsentation

ASE
- Dissertationseminar
- BMD
- Mitarbeiter

Date
Start
End
Prüfung

Mi 01.04.2009
07:24
17:26
10h 02m

Do 02.04.2009
07:58
16:16
09h 17m

Fr 03.04.2009
07:36
12:18
04h 42m

Mo 06.04.2009
07:45
18:17
10h 32m

Di 07.04.2009
07:14
15:37
08h 23m

Lehre 17h 56m

Server (NET)

Client (Silverlight)
Server-side Extensions

Environment

Visibility Scope

<table>
<thead>
<tr>
<th>Server</th>
<th>Silverlight</th>
<th>Client</th>
</tr>
</thead>
<tbody>
<tr>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
</tbody>
</table>

Global

Group

User
Client-side Extensions

Environment

Visibility Scope

- Global: ✓ ✓ ✓
- Group: ✓ ✓ ✓
- User: ✓ ✓ ✓
Client-side Extensions

Environment

Visibility Scope

- Global ✓ ✓ ✓
- Group ✓ ✓ ✓ ✓
- User ✓ ✓ ✓ ✓
Client-side Extensions

Environment

Visibility Scope

Server  Silverlight  Client

Global  ✔  ✔  ✔

Group  ✔  ✔  ✔

User  ✔  ✔  ✔
http://ase.jku.at/plux