



## **CAP.NET - Rich-Client-Plattform für .NET** **Komponentenmanagement und User-Interface-Integration**

Diplomaufgabe für Deepak Dhungana

Matrikelnummer 0057030

deepak.dhungana@students.jku.at

### **Hintergrund**

Neuere Softwaresysteme wie OSGi und Eclipse haben mit der Einführung von Plug-In-Architekturen neue Wege für den Bau von Anwendungen aufgezeigt. Typisch dabei ist, dass Anwendungen nicht als ein monolithischer Block aufgebaut sind, sondern aus einer beliebigen, erweiterbaren Menge von Komponenten (Plug-Ins) zusammengesetzt werden, die sich aber nahtlos zu einer homogenen Arbeitsumgebung integrieren. So wurde mit OSGi ein standardisiertes Komponentemodell und eine Plattform definiert, welche das Management und die Ausführung von unterschiedlichen Applikationen und Dienstprogrammen innerhalb einer Java VM erlaubt. Eclipse ist ein System, welches auf einer flexiblen Plug-In-Architektur aufsetzt und insbesondere im Bereich von Software-Entwicklungswerkzeugen die individuelle Gestaltung von Arbeitsumgebungen aus einer Vielzahl von Plug-Ins ermöglicht.

Obwohl die von diesen Systemen eingeführten Konzepte leistungsfähig und die Systeme damit entsprechend erfolgreich sind, fehlt bisher eine umfassende Analyse und Erforschung der Möglichkeiten und Prinzipien zur Gestaltung von Applikationen aus einer Menge von Komponenten. Im Projekt CAP.NET sollen diese grundsätzlichen Möglichkeiten für die Gestaltung von Rich-Client-Applikationen erforscht werden. Ziel des Projektes ist, auf der Basis der .NET-Technologie Prinzipien zu erforschen und Konzepte zu entwickeln, um die Entwicklung von Rich-Client-Applikationen möglichst gut zu unterstützen. Dabei steht die Individualisierung und Anpassbarkeit der Client-Applikation, bzw. Client-Application-Suite für den individuellen Benutzer im Vordergrund. Dazu gehören als Zielsetzungen:

- **Komponenten:** Die Zusammensetzung der Client-Application-Suite aus einer individuellen Auswahl von Komponenten (Plug-Ins)
- **Integration:** Die Integration der unterschiedlichen Komponenten zu einer integrierten, homogenen Arbeitsumgebung. Dabei muss die Integration auf unterschiedlichen Ebenen erfolgen:
  - **User Interface-Integration:** Die Komponenten stellen jeweils eigene User Interface-Elemente bereit, die zu einer homogenen Benutzeroberfläche integriert werden müssen.

- Service-Integration: Die Plug-Ins stellen bestimmte Funktionen in der Form von Service-Schnittstellen zur Verfügung. Diese müssen zu einer Gesamtfunktionalität integriert werden.
- Datenintegration: Die Plug-Ins arbeiten auf gemeinsamen Daten.
- Anpassbarkeit: Es soll die Möglichkeit geboten werden, die Application-Suite möglichst optimal an den Benutzer anzupassen. Dies beinhaltet die individuelle Gestaltung der Benutzeroberfläche als auch die Definition von für den Benutzer individuell gestalteten Arbeitsabläufen.

## **Aufgabenstellung**

Die Diplomarbeit wird im Rahmen des Gesamtprojektes CAP.NET erstellt, wobei folgende Teilbereiche sowohl konzeptionell zu erarbeiten als auch entsprechend umzusetzen sind:

### Komponentenmanagement:

Es sind in Anlehnung an die Konzepte von OSGi und unter Ausnutzung der Möglichkeiten von .NET Mechanismen für das Management von Komponenten zu entwickeln. Dazu gehören das Bereitstellen von Komponenten auf Installations-Servern, das Ausforschen der Komponenten durch einen Klienten, die Installation von Komponenten am Client, das automatische Update am Client bei neuen Versionen, entsprechende Sicherheitsmechanismen, die die Authentizität der Komponente sicherstellen, das dynamische Laden der Komponenten und ein dynamisches Reload bei Updates.

### Einstellungen:

Komponenten sollen durch eine Vielzahl von Einstellungen an die Bedürfnisse des Benutzers angepasst werden können. Dazu ist ein allgemeiner Mechanismus zu entwickeln, der die Beschreibung von Einstellungsmöglichkeiten bei Komponenten, das Verwalten von Einstellungen und die Eingabe von Einstellungen über entsprechende GUI-Komponenten unterstützt.

### User Interface-Integration:

Es sollen die Möglichkeiten der Integration von Komponenten auf der Ebene der Benutzeroberfläche erforscht, ein umfassendes Konzept für User-Interface-Integration entwickelt und unter konsequenter Ausnutzung der .NET-Technologie umgesetzt werden.

Beginn: März 2005

Betreuer: Dr. Herbert Prähofer