

Entwicklung eines Portlet Abonnement Plugins für den IBM Workplace Managed Client

**Seminararbeit
Torben Rasche**

Betreuer: Ingo Erdmann

University of Paderborn
Business Computing 2 – Information Management & Office Systems
Faculty of Business Administration, Business Computing & Economics
Prof. Dr. Ludwig Nastansky
Warburger Str. 100, D-33098 Paderborn
Tel.: +49--5251--60-3368
<http://gcc.upb.de>



Table of Contents

Einführung

Konzeption

Implementierung

Analyse

- ➔ **Einführung**
 - ➔ Ziel der Arbeit
 - ➔ Eclipse Plugin
 - ➔ IBM Workplace Managed Client

- ➔ **Konzeption**
 - ➔ WSRP-Spezifikation
 - ➔ Szenario

- ➔ **Implementierung**

- ➔ **Analyse**



Table of Contents

Einführung

Konzeption

Implementierung

Analyse

➔ Ziel der Arbeit

- ➔ Motivation: Anbieter von Remote Portlets machen deren Inhalte und Funktionalität frei verfügbar
- ➔ Vorteil: Anwendungen müssen nicht selbst entwickelt werden (Beispiel: Börsenportlet)
- ➔ Implementierung in Form eines RCP-Plugins
- ➔ Dieses RCP-Plugin soll im IBM Workplace Manged Client lauffähig sein



Table of Contents

Einführung

Konzeption

Implementierung

Analyse

→ Eclipse Plugin

- Ein Plugin beschreibt eine Funktionalität die in ein bestehendes System integriert werden kann
- Eclipse Plattform aufgebaut aus Plugins

→ IBM Workplace Managed Client

- Integrierte Verfügbarkeit
- Mail-, Instant Messaging- und Kalendaranwendungen
- Basiert auf dem Eclipse RCP-Framework



Table of Contents

Einführung

Konzeption

Implementierung

Analyse

→ Ausgangssituation Portalumfeld

- Es existieren viele Service-Anbietern
 - Verwendung von proprietären APIs
 - Portlets müssen für jedes Portal angepasst werden
 - Verhindern universellen Einsatz in konkurrierende Portalsystemen
- Wie lassen sich Portlets ohne großen Aufwand in ein Portal integrieren?
- Antwort: Standardisierung der Portaltechnologie



Table of Contents

Einführung

Konzeption

Implementierung

Analyse

→ Java Specification Request 168 (JSR 168)

- JCP-Konsortium unter Beteiligung von Sun, Oracle, IBM...

- Portlets sollen vom Aufbau uniform sein

- Auf Basis bestehender J2EE Konzepte (Servlet-Technologie)

- Variable Darstellungsformen
 - Fenster in unterschiedlichen Zustände (min, max, normal)
 - Unterstützung verschiedener Modi (View, Edit, Help)



Table of Contents

Einführung

Konzeption

Implementierung

Analyse

➔ Web Services for Remote Portlets (WSRP)

- ➔ Standard der OASIS Gruppe mit beteiligte Firmen IBM, Apache, Microsoft etc.
- ➔ Ziel: Standard für Remote Portlets zu etablieren
- ➔ Ansteuern von JSR-168-konformen Portlets und via Web Services in das eigene Portal integrieren
- ➔ Neben der Übertragung von Inhalt und Geschäftslogik wird auch eine Präsentationsschicht mitgeliefert

 OASIS

→ WSRP Architektur in einer Portalumgebung

- Producer bietet Portlets als WSRP-Service an
- Portal implementiert einen WSRP Consumer um den WSRP-Service zu nutzen
- Consumer aggregiert Portlets nach den Bedürfnissen des End-Nutzers und stellt sie als Portalseite zur Verfügung

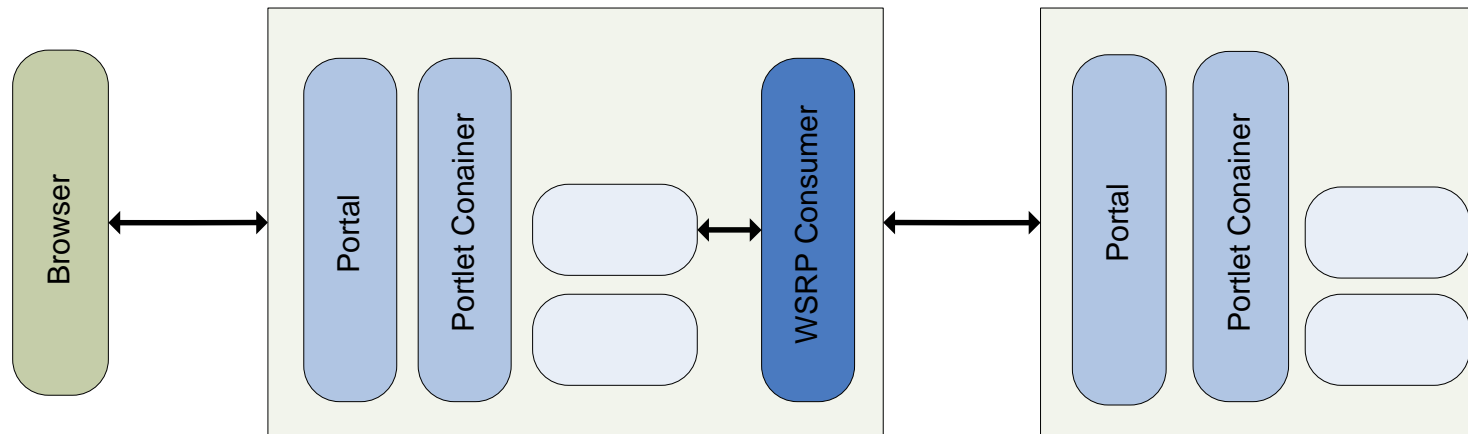


Table of Contents

Einführung

Konzeption

Implementierung

Analyse



Table of Contents

Einführung

Konzeption

Implementierung

Analyse

→ Architektur Portlet Abonnement Plugin

- Das Eclipse Plugin übernimmt die Rolle des Consumers
- Der Consumer agiert als Standalone-Applikation mit der Aufgabe die vom Producer angebotenen Portlets zu integrieren

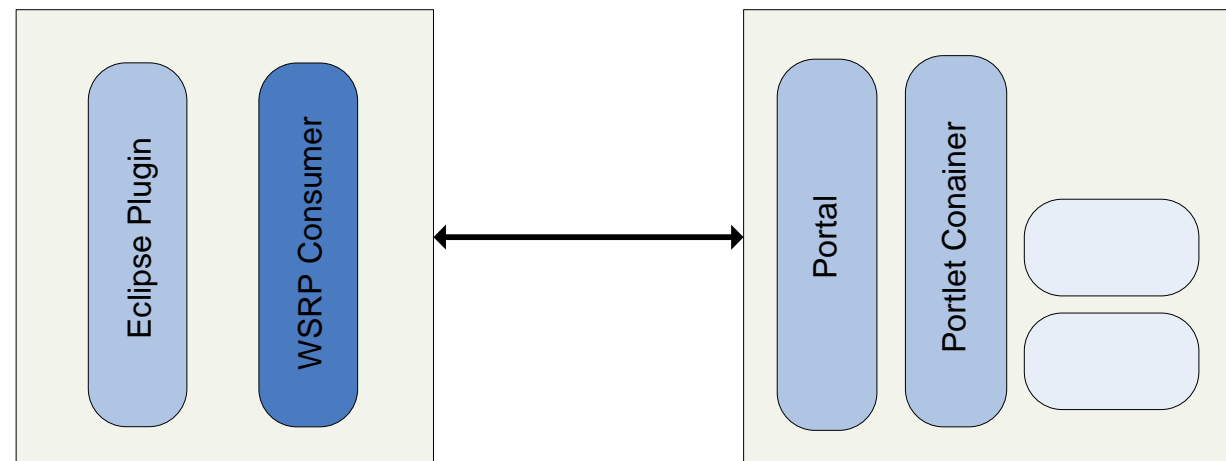




Table of Contents

Einführung

Konzeption

Implementierung

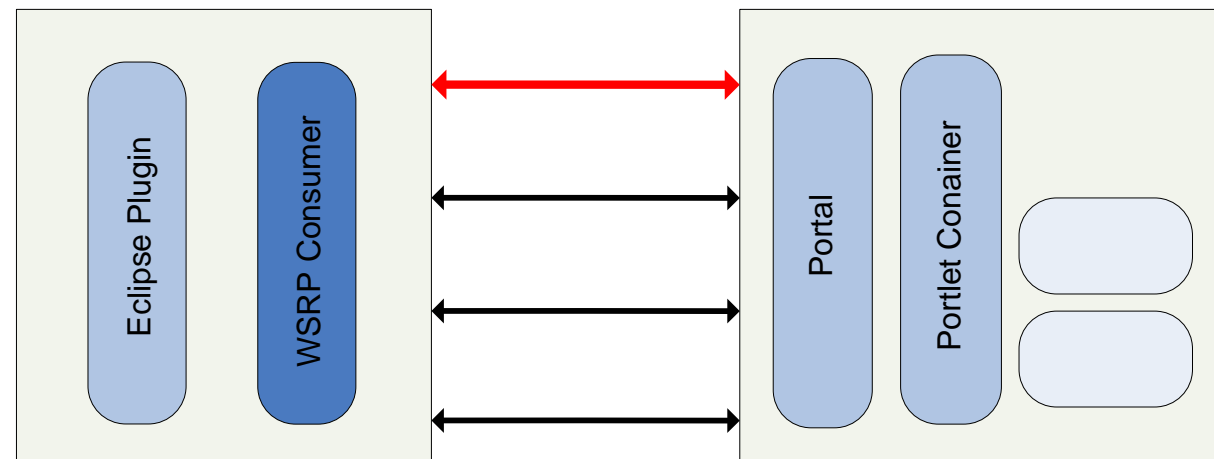
Analyse

→ WSRP Schnittstellen (Interfaces)

- Um die Interaktion zwischen Producer und Consumer zu standardisieren und vereinfachen stehen vier Schnittstellen zur Verfügung
 - Service Description Interface
 - Registration Interface
 - Markup Interface
 - Portlet Management Interface
- Über die vier Interfaces läuft die gesamte Kommunikation zwischen Producer und Consumer
- Der Producer muss diese Schnittstellen implementieren und der Consumer die entsprechenden Methoden bereitstellen um diese anzusteuern.

→ Szenario Step 1

- Consumer fordert über das ServiceDescription Interface eine Dienstbeschreibung des Producers (Metadaten)
- Der Producer sendet eine Antwort an den Consumer in der u.a. die Information enthalten ist, ob der Consumer sich beim Producer registrieren muss
- Falls keine Registrierung erforderlich ist bleibt der Consumer anonym



→ Szenario Step 2

- Ist eine Registrierung notwendig erfolgt diese durch Aufrufen des Registration Interfaces
- Definition eines Gültigkeitsbereichs und Senden eines *registrationHandles*

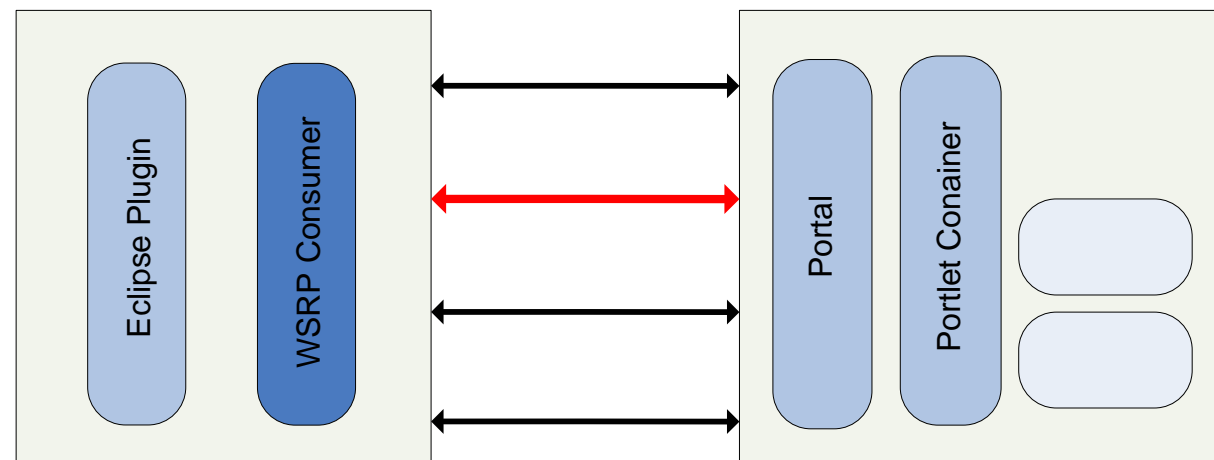


Table of Contents

Einführung

Konzeption

Implementierung

Analyse

→ Szenario Step 3

- Zweite Anforderung der Service Description
- Authentifizierung des Consumers durch Übermittlung des *registrationHandles*
- Producer gibt Auskunft über die angebotenen Portlets, Sprachen, Modi, Window States usw.
- Ein verfügbares Portlet wird über einen eindeutigen *portletHandle* referenziert

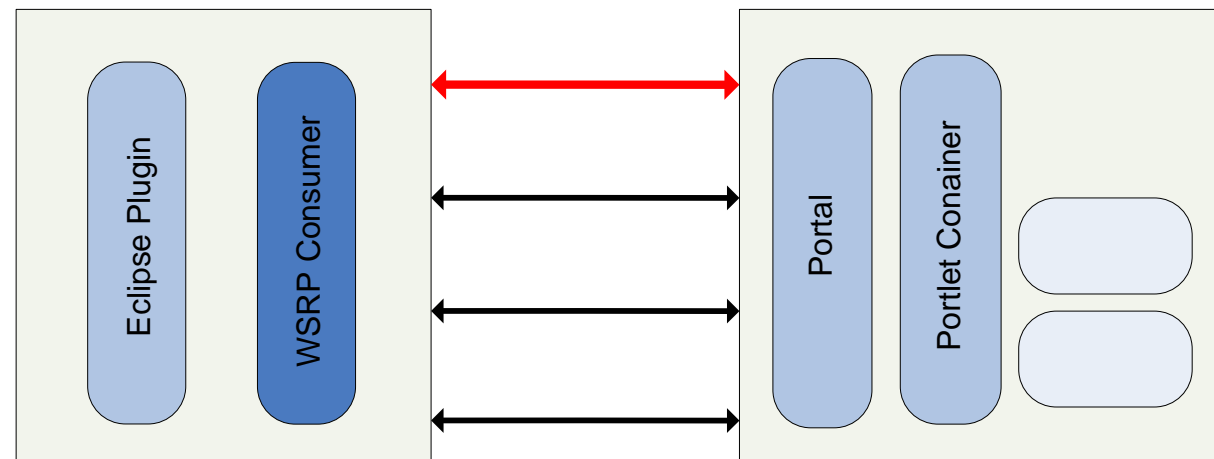


Table of Contents

Einführung

Konzeption

Implementierung

Analyse



Table of Contents

Einführung

Konzeption

Implementierung

Analyse

→ Szenario Step 4

- Um die Portletseiten zu generieren, muss der Consumer das Markup der Portlets vom Producer erhalten
- Das Markup Interface muss bei jeder Interaktion neu angesteuert werden, z.B. beim Betätigen eines Links

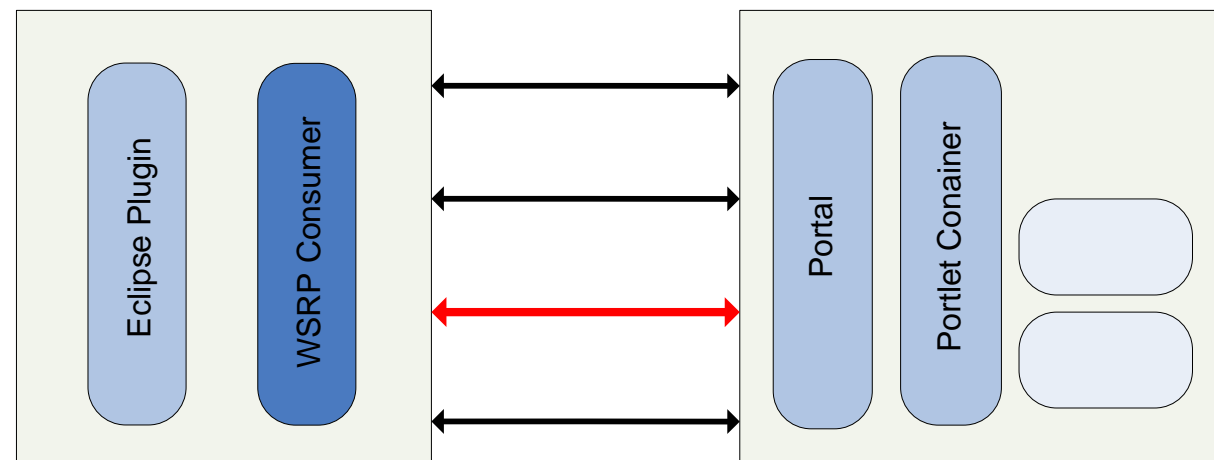




Table of Contents

Einführung

Konzeption

Implementierung

Analyse

→ Szenario Step 5

- Von Benutzerseite kann es nötig sein, die Portlets an persönlichen Bedürfnisse anzupassen
- Das *Portlet Management Interfaces* ermöglicht den persistenten Status von Objekten und den Lebenszyklus zu steuern
- *Consumer-Configured* Portlets
- Beispiel: Portlet mit Wetterbericht

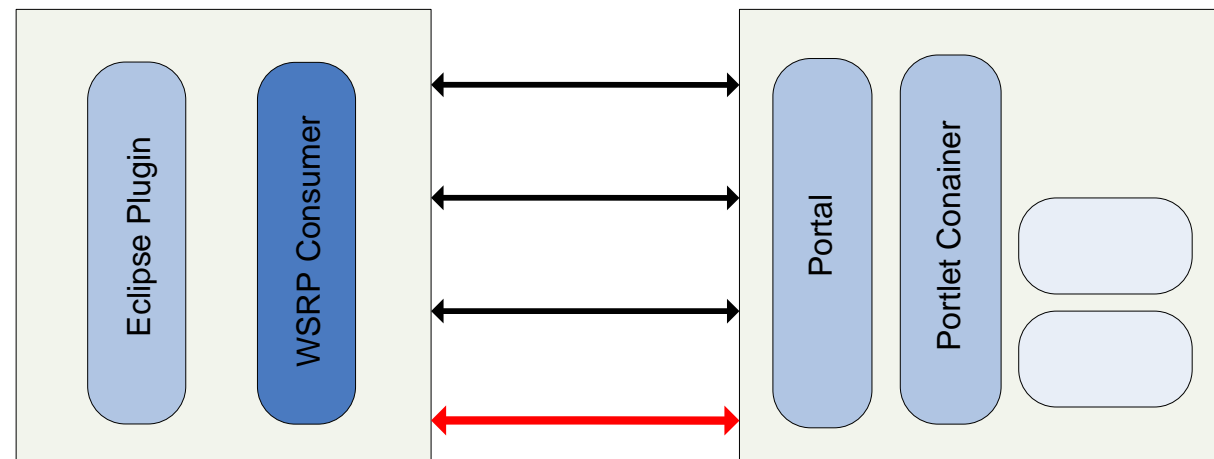




Table of Contents

Einführung

Konzeption

Implementierung

Analyse

→ WSRP4J

- WSRP-Referenzimplementierung der Apache Software Foundation (ASF)
- Ins Leben gerufen durch IBM
- Stellt ein *Proxyportlet* und einen *Swing Consumer* zu Verfügung
- Projektstatus *Incubated*

→ Portlet Abonnement Plugin

- Verwenden des Quellcodes des *Swing Consumers*
- Anpassungen und Erweiterungen des bestehenden Sourcecodes





Table of Contents

Einführung

Konzeption

Implementierung

Analyse

Demonstration des Portlet Abonnement Plugins

→ Probleme

- WSRP4J unterstützt keine *Forms*
- Referenzimplementierung nicht fehlerfrei und stabil
- Kein WSRP4J-Projektfortschritt
- Es sind nur sehr wenige WSRP-Producer frei verfügbar



Table of Contents

Einführung

Konzeption

Implementierung

Analyse

→ Fazit

- Aufgrund der vorhandenen Schwächen eingeschränkte Funktionalität
- Falls Probleme in naher Zukunft in WSRP4J nicht gelöst werden, empfiehlt es sich einen eigenen Lösungsansatz zu entwickeln
- WSRP Standard bietet neue Möglichkeiten
- Grundgedanken aus der WSRP-Spezifikationen sind in der Implementierung umgesetzt
- Ansteuern verschiedener Portlets nun möglich



Table of Contents

Einführung

Konzeption

Implementierung

Analyse



Table of Contents

Einführung

Konzeption

Implementierung

Analyse

➔ **Ausblick**

- ➔ Einbinden einer UDDI-Schnittstelle
- ➔ Steigende Anzahl von WSRP-Service Anbietern
- ➔ WSRP Spezifikation 2.0 für Mitte 2005 angekündigt
 - ➔ Portlet-to-Portlet Kommunikation



Table of Contents

Vielen Dank für ihre Aufmerksamkeit!

Fragen ???