

# Eclipse RCP für Enterprise Lösungen

von

Stefan Lötscher, Inf. Ing. ETH



# Worum geht es?

- Eclipse RCP, die ideale Lösung
- Erweiterungen von Eclipse RCP für Enterprise Lösungen
- Dashboard – Ein Beispiel

# Ausgangslage

- Software Engineering Unternehmen
- Fokus auf mehrschichtige Client–Server Applikationen
- Erfahrung mit HTML- und Rich-Clients
- Kein eigenes Rich-Client Framework
- Schlechte Erfahrungen mit proprietären Frameworks

# Unsere Anforderungen an ein Java Client Framework

- “Normale” Anforderungen  
(Look & Feel des OS, Top-Level- und  
Kontext-Menüs, Standard GUI-  
Komponenten Shortcuts und Favoriten,  
Persistenz der Ansichten, Help-System  
usw.)
- Automatische Aktualisierung
- Start/Restart/Stop-Zyklus

- Rollenabhängige Steuerung von GUI Komponenten
- Modulare Entwicklung (Basis- und Erweiterungs-Module)
- Weite Verbreitung (grosser Bekanntheitsgrad, gute Dokumentation, Community)
- **Erweiterbarkeit**



# Eclipse RCP – Das ideale Fundament



# Anforderung einer

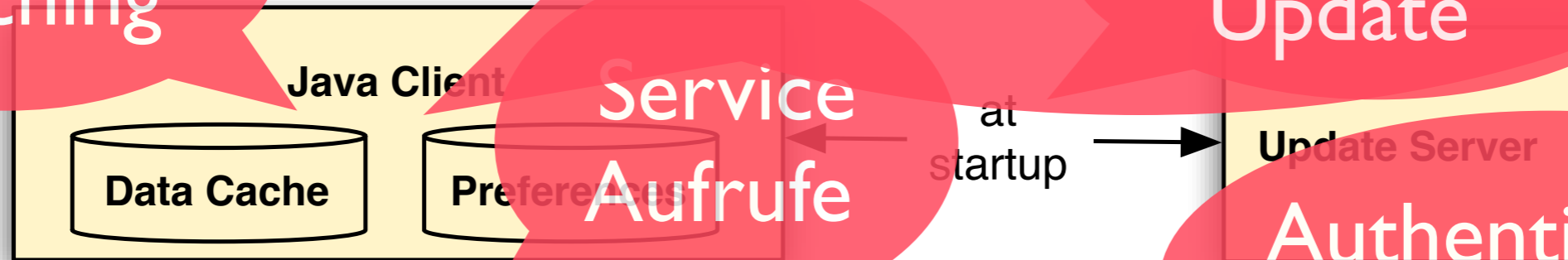
Rollenabhängiges GUI

Zusammenspiel zw. GUI

Komponenten und automatisches

Update

DataObject  
Caching



Service  
Aufrufe

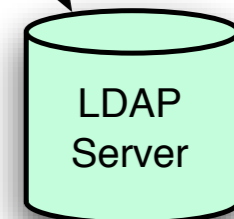
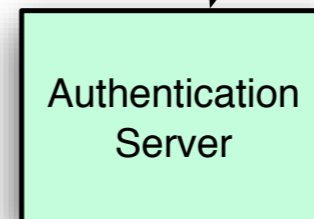
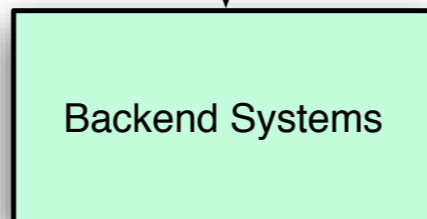
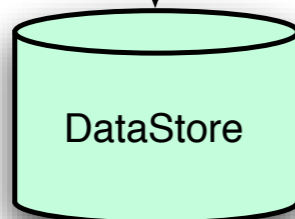
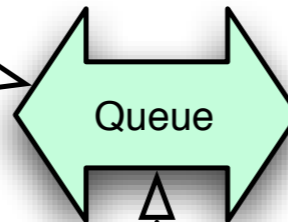
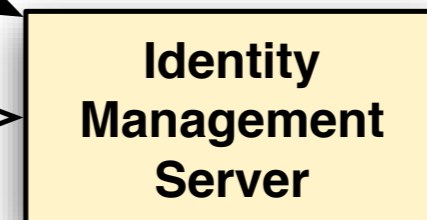
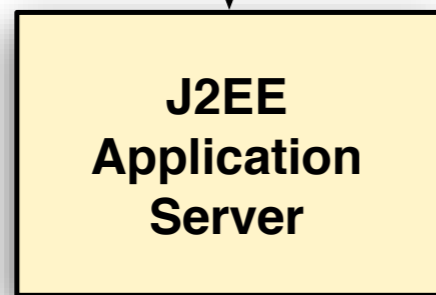
at  
startup

Update Server

Authentisierung  
Autorisierung

Communication 2  
(synchron/asynchron)

Communication 1  
(synchron)

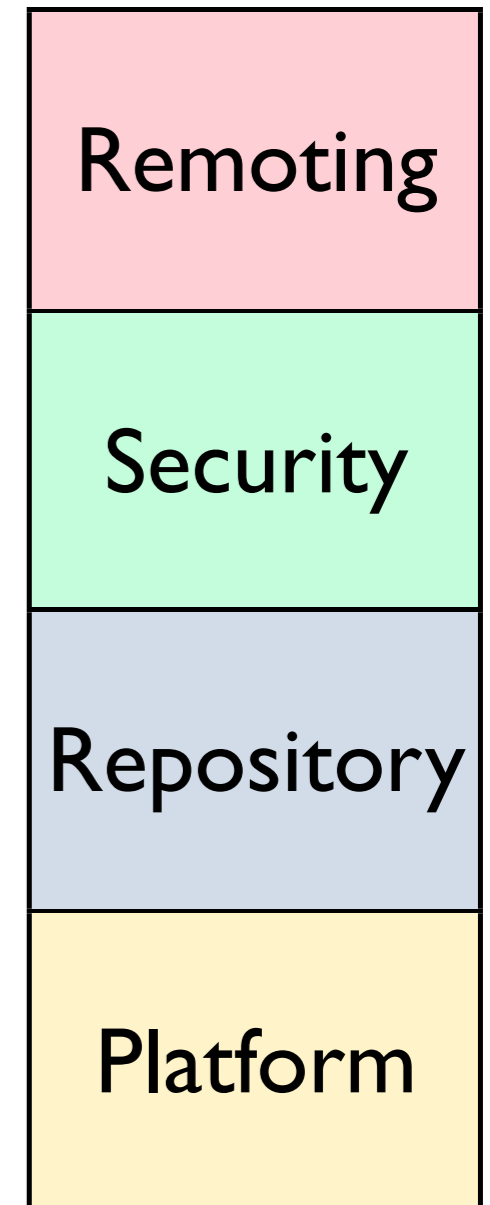


# Unsere Grundsätze für die Erweiterung von Eclipse RCP

- Verwendung von Eclipse Konzepten und Mechanismen, wo immer sinnvoll und möglich (z.B. Extension Points)
- Trennung von Konfiguration und Plugins
- Benutzer sind Entwickler
- Konfigurierbar bedeutet immer auch erweiterbar durch Plugin Entwickler

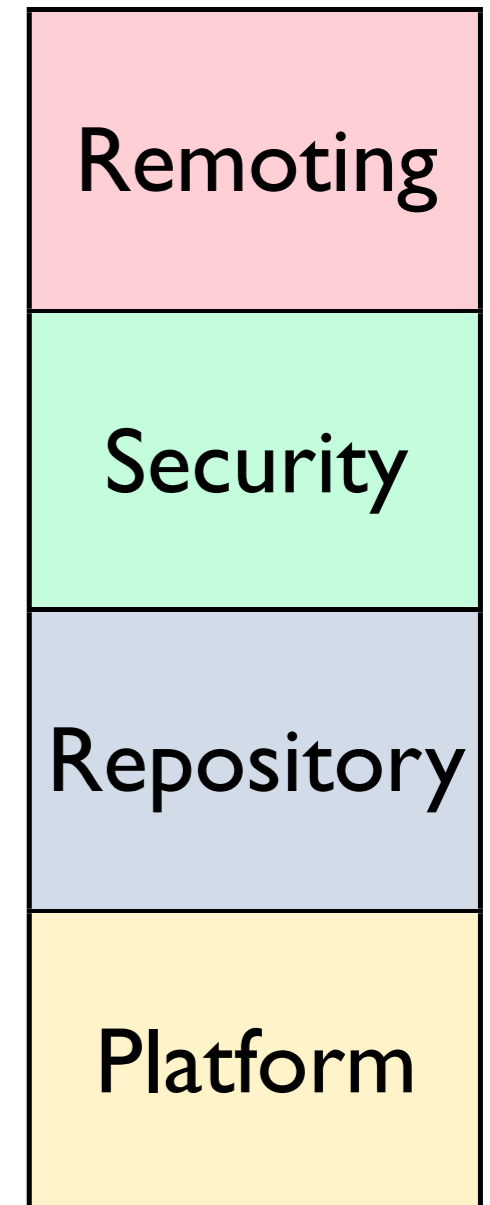
# Unsere Plugins

- Remoting Plugin
  - Synchrone / asynchrone Kommunikation
  - Mehre Kommunikationskanäle mit eigenen Security Stufen
  - Konfiguration von Protokoll und Handlern im Plugin
  - Konfiguration von Kommunikationsparametern in Konfigurationsfile
  - Sessionhandling mit Callback für Security Plugin



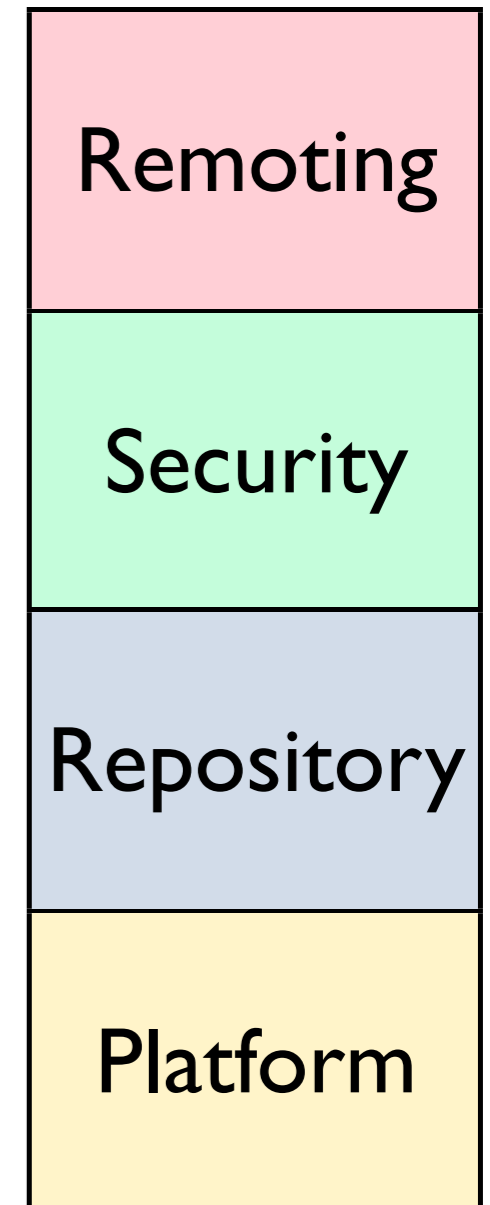
- Security Plugin

- Authentisierung GUI Komponenten für diverse Verfahren
- Einstufige- und Mehrstufige-Verfahren
  - User-Password
  - User-Password-(Secure-ID / TAN)
- Initialer Passwort Wechsel
- Erneutes Anmelden nach Time-Out
- Abmelden



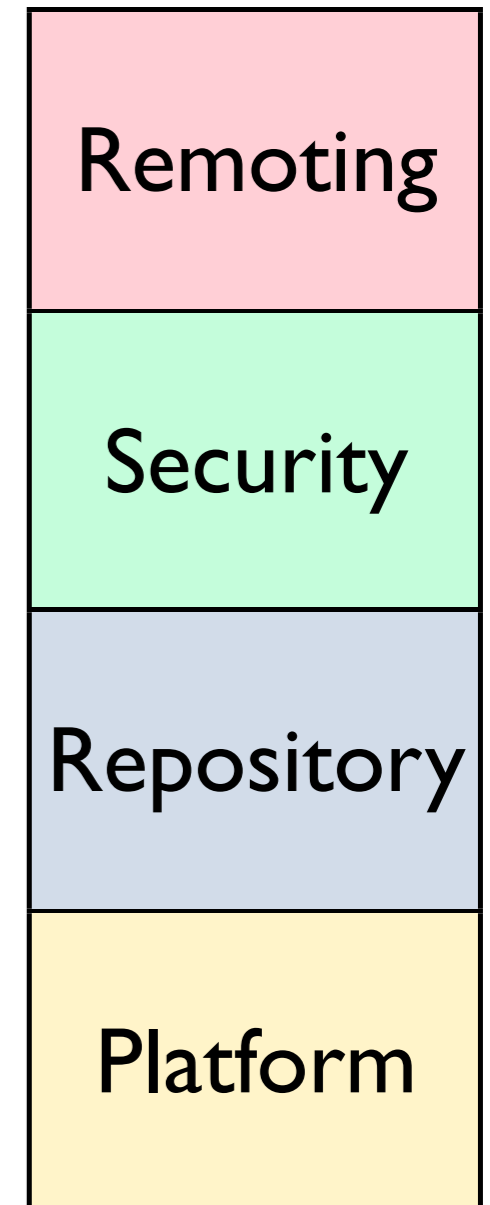
## ● Repository Plugin

- Intelligenter Cache für Daten-Objekte
- Automatische Analyse der Resultate von Service-Aufrufen und Update des Cache
- Erzeugen von Update-, Add- und Delete-Events (Daten-Objekte können untereinander verhängt sein, 1:1 oder 1:n)
- Eindeutige Identifikation eines Daten-Objektes (jedes Daten-Objekt kommt nur einmal auf dem Client vor)
- Schnittstelle um sich für Events zu abonnieren



## ● Platform Plugin

- Konfigurierbare Daten-Objekt-Bäume und -Tabellen
- Konfigurierbare Editoren Inhalte
- Wiederverwendbare Inhalte für Editoren und Wizards
- Konfigurierbare, dynamische Wizards
- Verknüpfung von Modell und Editoren Inhalte
- Konfigurierbare Konverter und Label-Providers für Daten-Objekte



# Was ist eine Client Plattform

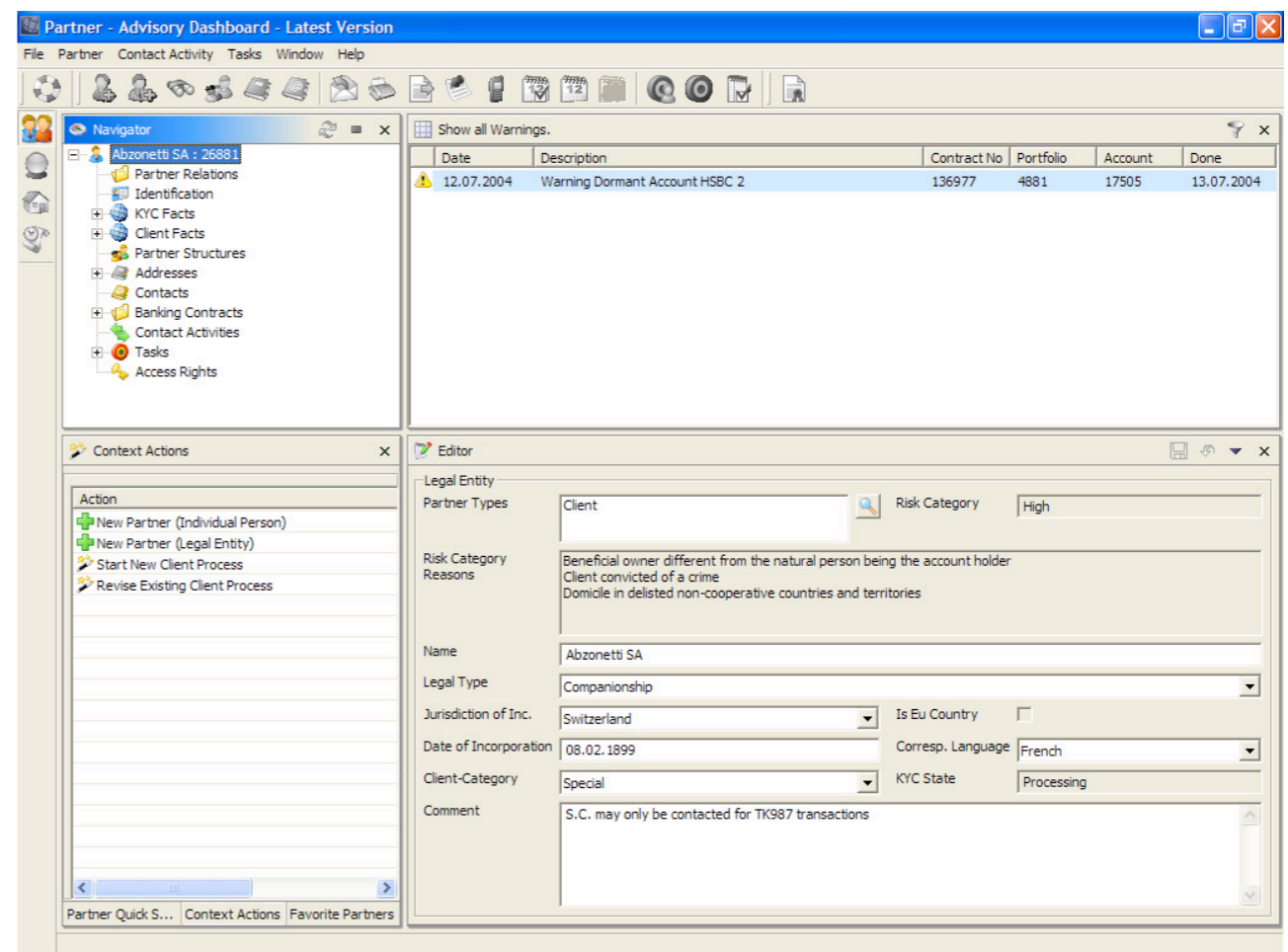
- Beispiele:
  - Eclipse IDE = Plattform für Tools im Bereich der Software Entwicklung
  - IBM Workplace: “Eine innovative Plattform zur vollständigen Integration von Mitarbeiter in bestehende Geschäftsprozesse.”
  - Microsoft Outlook – Mail, Kalender, Adressverwaltung
- Merkmal: Modulare Erweiterbarkeit

# Beispiel – Der “normale” Arbeitsplatz

- Pro Rolle eine Applikation
  - mehrfaches Anmelden
  - copy-paste von Applikation zu Applikation
  - verschiedene Datensichten
- Keine neue Applikationen, dafür zusätzliche Tasks für das IT-Department
- Verschiedene Look and Feel's - schwierige Schulung, kein einheitliches Konzept
- Verschiedene Technologien

# Dashboard – Der neue Arbeitsplatz

- Eine Applikation
- Funktionalität wird durch Rollen gesteuert
- Geringer Aufwand für zusätzliche Module
- Einheitliches Look & Feel
- Einheitliche Datensicht
- Vereinfachter Deployment- und Aktualisierungs-Prozess



# Fazit

- Mit Eclipse RCP als Fundament und den enterprisespezifischen Erweiterungen war es mit geringem Aufwand möglich eine individuelle Plattform für den Kunden zu entwickeln.
- Der Kunde besitzt nun auch auf dem Client eine mächtige Plattform um neue Business-Funktionalitäten effizient in die bestehenden Applikationen zu integrieren.

age of internet invention

**: inventage**

**Inventage AG**

**Lerzenstrasse 21**

**CH-8953 Dietikon**

**Switzerland**

**Tel: +41 43 343 20 20**

**[www.inventage.com](http://www.inventage.com)**

**[stefan.loetscher@inventage.com](mailto:stefan.loetscher@inventage.com)**