

# Entwicklung eines Patientendaten Verwaltungs-Prototypen mit dem besonderen Fokus: Datenintegration

Sascha-Ulf Habenicht

Kolloquium  
Universität Oldenburg  
Department für Informatik

26.02.2010

# Inhalt des Kolloquium

- 1 Cochlear Implant (CI)
- 2 Eclipse Rich Client Platform
- 3 Datenintegration
- 4 Ergebnisse

# Implant und Sprachprozessor

Implant



Sprachprozessor

# Was ist ein Cochlear Implant (CI) ?

## Informationen zu CI

- Häufige Ursache einer Hörstörung liegt im Innenohr
- Hörsinneszellen verstärken akustische Informationen und leiten diese an den Hörnerv
- CI ersetzt Hörfunktion des Innenohres, reizt elektrisch den funktionsfähigen Hörnerv
- CI soll Hörempfindung und Sprachverstehen ermöglichen
- **Erstmalig Ersatz für ein Sinnesorgan**

# Aufgaben eines CI-Zentrums

## CI-Versorgung erfolgt in hochspezialisierten CI-Zentrum

- interdisziplinäre Teams
- Voruntersuchungen und umfangreiche Beratungen
- medizinischer, pädagogischer und audiologischer Sicht
- individuelle Anpassung des Sprachprozessors beginnt ca. 4 Wochen nach der Implantation
- ambulante Hör-Sprach-Therapie wird über 2 bis 3 Jahre einmal wöchentlich durchgeführt
- lebenslanger Behandlungszeitraum

# ECLIPSE RICH CLIENT PLATFORM

## Warum als Eclipse Rich Client Platform (ERCP) ?

- Alternativen denkbar, z.B. Implementierung als Website:
  - ① Protokoll http ist zustandslos
  - ② Ressourcen sind begrenzt, z.B. bei einer Website  
Datenvolumen und Verarbeitungszeit
- **Vorteile:**
  - ① Gut bedienbare Benutzeroberflächen möglich
  - ② Einsetzbar ohne Netzwerkverbindung
  - ③ *Einsatz vieler Technologien möglich*
- **Nachteil:**
  - ① Aufwendig zu erlernen, Literatur notwendig
  - ② *Einsatz vieler Technologien notwendig*

## ERCP basiert auf OSGi-Framework

- Klassen werden zur Laufzeit in Bundles nachgeladen
- Ermöglicht dynamische Verwaltung von Java Komponenten
- Kein Zugriff über Typumwandlungen möglich
- Definiert: exportierende Elemente und Schnittstellen

## Konfiguration

- 1 **Dependencies:** Abhängigkeiten, die zur Laufzeit benötigt werden.
- 2 **Runtime:** Regelt die Sichtbarkeit der Klassen zu anderen Plug-ins.
- 3 **Extensions:** Schnittstellen, die außerhalb des Plug-in benutzt werden können.
- 4 **MANIFEST.MF:** Konfiguration des OSGi-Manifest.
- 5 **plugin.xml:** Klassen, die bei Bedarf zur Laufzeit geladen werden, Plug-in-Manifest.

## Workbench Lebenszyklus

- 1 **Activator**: Vor Starten der Anwendung
- 2 **Application**: Eintrittspunkt
- 3 **ApplicationWorkbenchAdvisor**: Workbench
- 4 **ApplicationWorkbenchWindowAdvisor**: Fenster
- 5 **ApplicationActionBarAdvisor**: Aktionen

# DATENINTEGRATION

# orthogonale Dimensionen der Datenintegration

- ① **Verteilung:**
  - physisch (geografisch getrennt)
  - logisch (mehrere Orte zur Speicherung)
- ② **Autonomie:** Entscheidungen über Daten wie Struktur, Zugriff
  - Schnittstellenautonomie: Anfragesprachen, Protokolle
  - Designautonomie: Art und Weise, Zugriff, Datenformat, Datenmodell, ...
- ③ **Heterogenität:** Datenmodelle selten exakt gleich, unterschiedliche Methoden, unterschiedliche Zugriffe

# ETL-Prozess

- 1 **Extraktion:** Informationen werden aus einer Datenquelle ausgelesen. Dazu werden z.B. SQL-Skripte, aber auch Export- und Reportingwerkzeuge benutzt.
- 2 **Transformation:** Transformation der Informationen in das Format und die Struktur der Datenbank. Dazu gehören auch die *Umformung von Datumswerten* oder die *Umrechnung von Einheiten*, wie z.B. Dollar in Euro.
- 3 **Laden (Load):** Beim Laden werden die Informationen in einen Arbeitsbereich geladen und von dort aus in die Datenbank gespeichert.

# Health Level Seven (HL7)

## HL7

- In den Kliniken kein HL7 Version 3, nur Version 2 im Einsatz
- HAPI: ein Parser von HL7 2.x Nachrichten

### Auszug einer A03 HL7-Nachricht:

```
MSH|^ &|ISH|0001|Orbis|0001|20090806064858||[...]<CR>  
EVN|A03|20090803100345||NP97I0|EHLERS<CR>  
PID|||5824882^8^M11||Mustermann^Manuela[...]<CR>
```

# Gerätedatentransfer (GDT)

## Gerätedatentransfer (GDT)

- Wird oft bei der Kommunikation mit medizinischen Geräten verwendet
- Dateibasiert
- Erste 3 Stellen geben die Länge (inkl. <CR><CF>) an (16),
- nächsten 4 Stellen geben die Feldkennung an (3101) und
- bis zum <CR> befindet sich der Inhalt (Schmidt).

### Auszug einer GDT-Nachricht:

```
0163101Schmidt<CR><LF>
```

# ERGEBNIS

# Ergebnisse

## Ergebnis der Bachelor-Arbeit

- Übersichtliche Benutzerschnittstellen möglich
- Anforderungen an Wartung und Pflege sind in ERCP berücksichtigt
- Plug-In Technologie erlaubt Zusammenarbeit mehrerer Anbieter auf einer Plattform

# Ergebnisse

## Ergebnis der Bachelor-Arbeit

- Wenn möglich, Kommunikation über den KOMMSERV (HL7)
- Senden und Empfangen von HL7 Nachrichten kann mit dem Open-Source Parser HAPI umgesetzt werden
- GDT hat *immer* individuelle Feldkennungen des Herstellers
- GDT ist nicht für die Verarbeitung durch Benutzer geeignet
- Konzentration von GDT-Nachrichten auf Serverebene

# Ergebnisse

## Ergebnis der Bachelor-Arbeit

- **Langfristig** ist eine Mehrschichten-Architektur notwendig
- Im Uniklinikum Köln werden neue Rechner angeschafft
- Aufwand des Projekts ist nun in Teilaufgaben zerlegbar
- Benutzeroberflächen vereinfachen Kommunikation

# Vielen Dank für Ihre Aufmerksamkeit!

*(Unterlagen auf [www.Sascha.Habenicht.name](http://www.Sascha.Habenicht.name))*