



Perlen Vortrag: User-Toolkits - Softwareentwicklung 2.0

Kontakt:

Dr. Peter Dornbusch

dornbusch@fireflow.de

Phone: +49 (89) 210 2377 11

Fax: +49 (89) 210 2377 19

Mobil: +49 (162) 906 1974

Agenda

- Web 2.0 und Softwareentwicklung
- Unterschied: Web 1.0 Projekt und Web 2.0 Projekt
- Web 2.0 Entwicklerperspektive
- Web 2.0 Kundenperspektive
- User-Toolkits und Innovationsprozesse
- Beispiele
 - Mozean
 - Game Creator
 - Yahoo-Pipes
- Ausblick und weitere Entwicklung
- Herausforderungen



Web 2.0

- Stärkere User-Ausrichtung
- Vereinfachungen und Verbesserung der Usability (Eigene Website vs. Blog)
- Sehr starke Vernetzung
- Höhere Kommunikation
- Erweiterte Kreativität: User-generated Content (YouTube, etc.)
- Bei genauer Betrachtung kein diskreter Schritt von Web 1.0 zu Web 2.0, sondern kontinuierliche Entwicklung

- Web 2.0 „Entwicklungsprozess“ (sehr stark orientiert an Innovationsprozessen)



Web 1.0 Projekte vs. Web 2.0 Projekte

- Innovative Idee (PPT, Prototype)
 - Viel Geld eingesammelt
 - Erster Release
 - Viel Geld für Marketing ausgegeben
 - Entweder
 - Erfolgreich
 - Oder
 - Problem mit der Technik/Logistik/Sales/After-Sales
 - Business Modell funktioniert nicht
- Innovative Idee
 - Beta online; Kunden Feedback vorhanden
 - Kleineres Investment/Eigenfinanzierung (- 100 €)
 - Markteintritt und Verbreitung (ausschließlich) über Mundzumund-Propaganda
 - Ausbau auf Basis von Kundenwünschen/-Feedback
 - Bei kontinuierlichem Erfolg -> nächste Finanzierungsrunde



Web 2.0 „Entwicklungsprozess“

- Kleines Start-Team für die erste Version (max. 3 Personen)
- Limitationen führen zu kreativen Lösungen
- Skalierungsfragen am Anfang ausklammern
- Feature Request sollten ausschließlich von (potentiellen) Kunden kommen (wichtig: frage auch nach Dinge, welche die Leute nicht wollen)
- Früher Prototyp (Anfangs HTML-Mock-up) um möglichst schnell Kunden-Feedback zu bekommen
- Entwicklung erfolgt in (kurzen) Iterationen (Daily Build)



Web 2.0 „Entwicklungsprozess“ - II

- Einfache Software „Keep your code as simple as possible“
- Konzept der offenen Plattform: RSS, APIs, etc.
- Keine Verwendung einer funktionalen Spezifikation
- Enge Zusammenarbeit zwischen Entwicklung und Support
- Größeres Update 30 Tage nach erstem Release
- Priorisiere Bugs (ggf. ignoriere Bugs)



Web 2.0 - Kundenperspektive

- Long Tail – Verhältnis von Produkten zum Absatz -> Nischenmärkte werden bedient
- Selbstdarstellung
- Jeder bringt nur einen kleinen Teil ein, alle partizipieren am „großen Ganze“
- Kunde ist Teil der Entwicklung des Produkts
- Wikipedia bis Open-Source



Rückblick:

Kundennahe Entwicklungswerkzeuge für den Entwurf mobiler Geschäftsprozesse



Kundennahes Entwicklungswerkzeug (User-Toolkits)

Toolkits

- Werkzeug, welches es Kunden unmittelbar erlaubt eigene Produkte zu entwerfen und direkt zu verwenden
- Keine Erfragung von Kundewünschen (im Gegensatz zu klassischen Marktforschungsmethoden)
- Entwicklung eines eigenen Produkts innerhalb einer ausgesuchten Domäne
- „sticky“ Information werden implizit übertragen

Toolkit-Eigenschaften und Anforderungen

- Trial-and-Error-Lernprozess
- Aufspannen eines Lösungsraum innerhalb einer Domäne
- Benutzerfreundlichkeit
- Modulbibliotheken
- Abbildung auf Produktionssystem



Mobile Geschäftsprozesse und Herausforderungen

Mobiler Geschäftsprozess

- ausschließliche Betrachtung von elektronisch abbildbaren Geschäftsprozessen
- Informell: mobiler Geschäftsprozess sobald für einen Teilprozess
 - Unsicherheit des Ortes vorliegt, Ort extern determiniert ist oder externer Zugriff auf Ressourcen des Unternehmensnetzwerkes nötig ist
- Konkrete Betrachtung:
 - auf Webservices abbildbar Geschäftsprozesse
 - Neue Geschäftsprozesse sind komponierte Webservices
 - Bausteine: mobile Telekommunikationsdienste (MMS, SMS, Voice, usw.)

Herausforderungen

- Interaktion von Telekommunikationsdiensten kann zu unerwartetem Verhalten führen (Feature Interaction)
- Kunde modifiziert unmittelbar System des Anbieters
- Explizit formulierte Eigenschaften (Anbieter) fließen in Gesamtentwurf ein
- Zusammenführung der imperativen Prozessbeschreibung (Kunden) und deklarativen Anforderungsbeschreibung (Anbieter)



Ergebnisse der Arbeit

- Kosteneffiziente Entwicklung neuer mobiler Geschäftsprozesse
- Markt für mobile Geschäftsprozesse steht unter hohem Kosten- und Innovationsdruck
- Identifizierung bzw. Befriedigung von Kundenbedürfnissen
 - Marktforschung kann extrem heterogene und schwer erfassbare Bedürfnisse nicht identifizieren
 - Probleme: „Sticky“ Information, Innovatorparadoxon (Funktional fixedness)
- Toolkits Erstmalige Übertragung des Konzepts auf einen Bereich des Software-Engineering
 - Mobile Geschäftsprozesse
- Entwicklung von Geschäftsprozessen durch den Kunden unmittelbar auf der Plattform des Anbieters
 - Anforderungen bezüglich der Sicherheit und Stabilität



Web 2.0 und Toolkits

- Toolkits dienen dazu Kunden in den Innovationsprozess zu integrieren
- Web 2.0 versucht Kunden in die Wertschöpfung zu integrieren
- Toolkits als Web 2.0 Werkzeug
- Kunden bauen (innovative) Anwendungen; Betreiber aggregiert (Nischen-) Applikationen

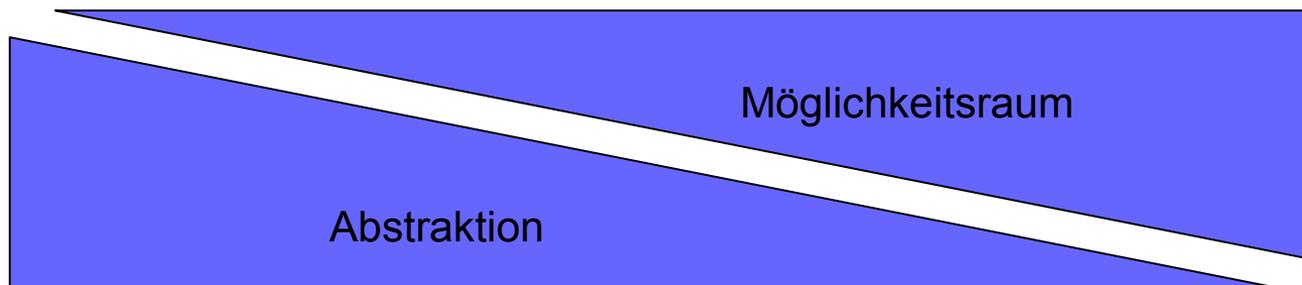


Beispiele



Ausrichtung auf anvisierte Zielgruppe

- Abstraktion muss sich an den unterschiedlichen Fähigkeiten der Nutzer orientieren
 - Toolkit sollte verschieden qualifizierte Usergruppen ansprechen
- Wichtig:
 - Wahl der richtigen Abstraktion der Basisdienste
 - Wahl der richtigen Abstraktion der Kombinationslogik
- Ausreichende und intuitive Dokumentation
- Technischer Support und Community-Unterstützung durch andere User



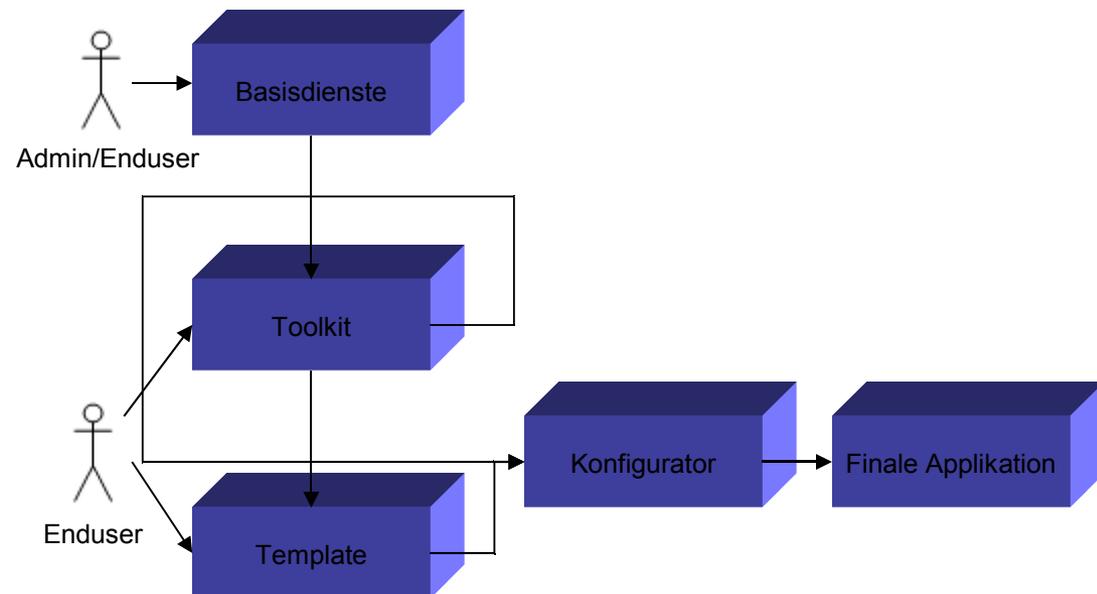
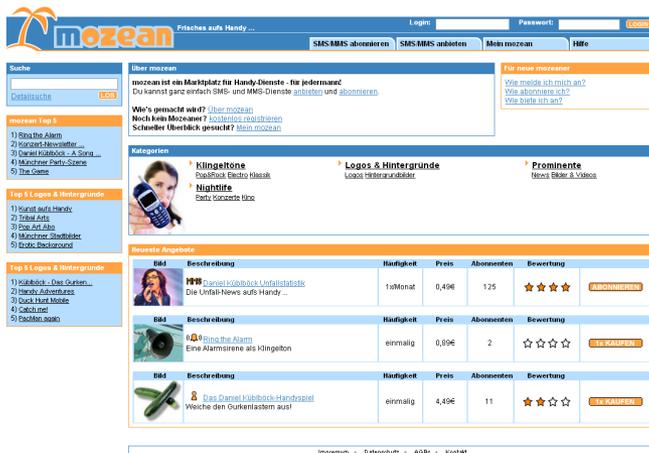
Leichte und intuitive
Bedienbarkeit

Großer gestalterischer Freiraum



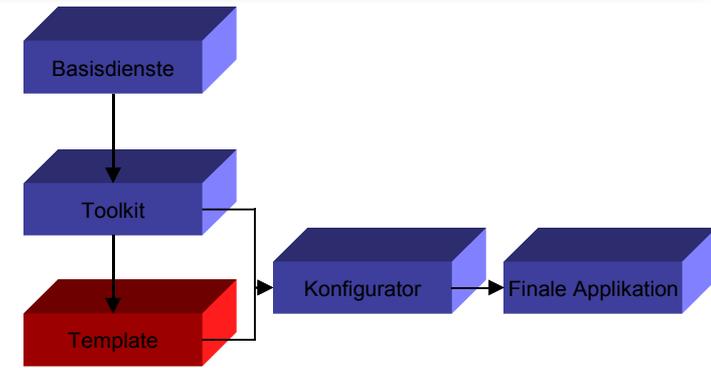
Fallstudie: Mozean

- Portal zum Anbieten von eigenen mobilen Inhalten und Services
 - Jeder kann Dienste anbieten und abonnieren/kaufen
 - Neben Standarddiensten (Klingeltöne, SMS-Newsletter, etc.), die durch Vorlage kreiert werden, lassen sich auch komplett neue Dienste entwickeln



Templates

- Schnelle Erstellung von neuen Diensten
- Rückgriff auf Know-how anderer Benutzer
- Niedrige Einstiegsschwelle
- Stark eingeschränkter Möglichkeitsraum



SMS/MMS anbieten > Schritt 1: Art des Angebots >

Hier kannst Du anderen mozean-Nutzern Handy-Dienste zugänglich machen. Wähle zunächst die **Art des Dienstes** aus, den Du anlegen willst:

SMS [SMS](#)
(Du versendest regelmäßig Text-Nachrichten an alle Nutzer des Dienstes)

MMS [MMS](#)
(Du versendest regelmäßig MMS-Nachrichten an alle Nutzer des Dienstes)

Klingelton
(Du stellst anderen Nutzern einen Klingelton zum Abruf zur Verfügung)

Logo [Logo](#)
(Du stellst anderen Nutzern ein Bild zum Abruf zur Verfügung)

Spiel
(Du stellst anderen Nutzern ein Handy-Spiel zum Abruf zur Verfügung)

Einmalige SMS
(Du stellst anderen Nutzern eine Text-Nachricht zum Abruf zur Verfügung)

Einmalige MMS
(Du stellst anderen Nutzern eine MMS-Nachrichte zum Abruf zur Verfügung)

Infos für neue Anbieter

[So bietest Du an mehr zum Thema Preise unzulässige Inhalte](#)

Dienst beschreiben

Dienst näher beschreiben:

Titel mit max. 180 Zeichen ein:

URL deines Dienstes ein:

am besten zu deinem Dienst passt:

keinen Dienst auswählen und bei Bedarf neue Logos hochladen.

kein Logo wählen LOGO WÄHLEN

Passwortschutz:

Hier kannst Du den Nutzerkreis deines Dienstes einschränken, indem Du ihn mit einem Passwort schützt. So können nur Leute den Dienst abonnieren, denen Du das Passwort gibst.

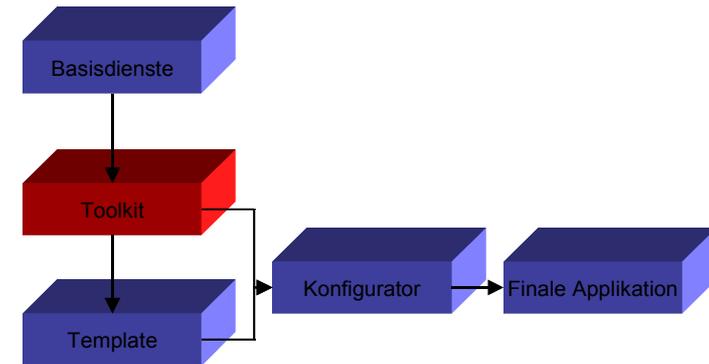
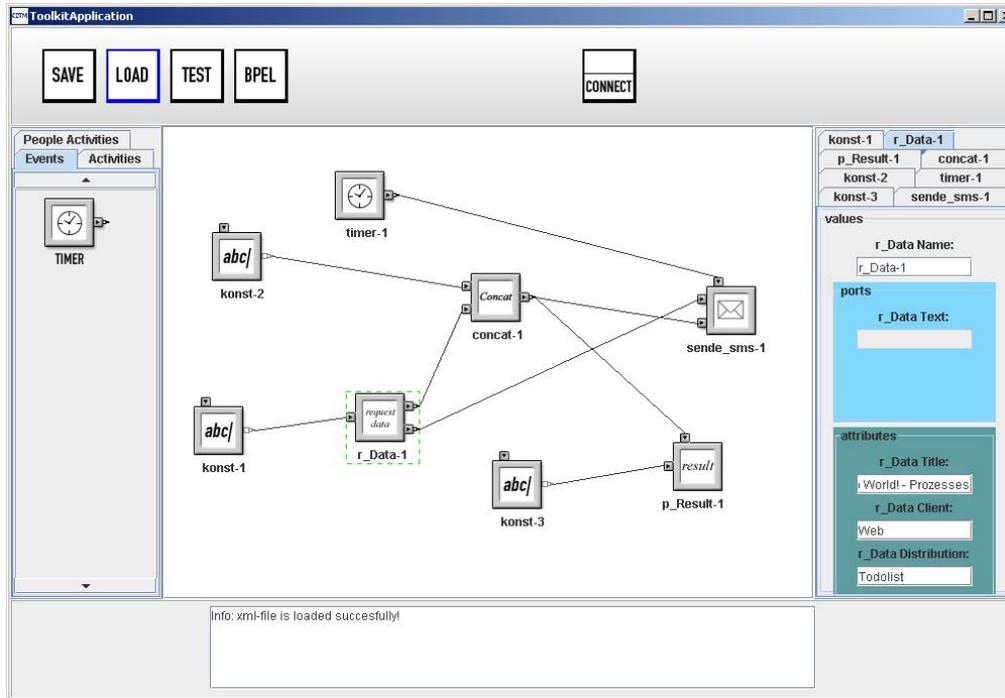
Passwort:

Premium SMS:

Wenn nicht bei mozean registrierte Nutzer deinen Dienst abonnieren wollen, müssen sie ein Codewort an den Anbieter Premium SMS schicken. Gib hier das Codewort ein, mit dem man deinen Dienst über Premium SMS bestellen soll!

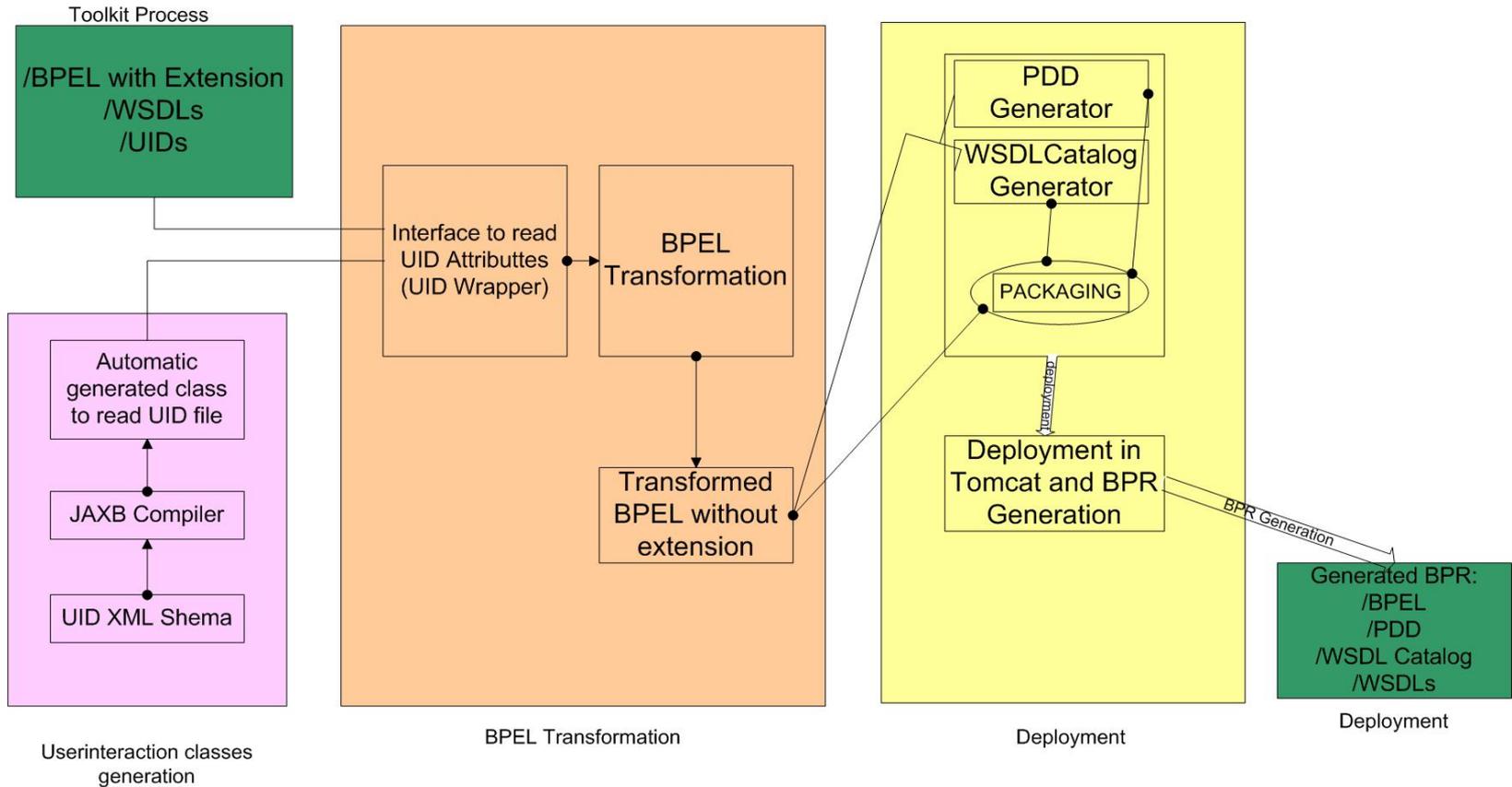
Codewort:

Toolkit



- Formale Spezifikation der Basisdienste
- Zusätzlich globale Regeln/Restriktionen
 - Problem von ausführbaren Prototypen -> Risiko muss abgeschätzt werden-> Nachweis von Eigenschaften
 - Fehlerfreie Ausführung
 - Sicherstellung von korrekter syntaktischer und semantischer Kombination der Basisdienste

Technische Realisierung



Globale Regeln/Restriktionen (Beispiele/informell)

Ausführung von Anwendungen verlangt nach Einhaltung von Regel, um Gesamtstabilität nicht zu gefährden

Im konkreten Fall zum Beispiel:

- Keine Zyklen
- Startereignis/-zustand
- Endereignis/-zustand
- Ein User darf innerhalb eines Dienstes nie mehr als 3 SMS zugestellt bekommen

Game Creator

- Handy Spiele Editor
- Verschiedene Entwicklungsstufen
 - Von Grafikvariation bis Code-Eingabe
- Spiele Templates
- Sicherheit hier nur untergeordnetes Problem, da Spiele auf dem Handy ausgeführt werden



The screenshot shows the FireFlow Game Creator website interface. At the top, there is a navigation bar with links for 'Neues Spiel', 'Spiele', 'MyGameCreator', 'Forum', 'Feedback', 'Login', 'Hilfe', and 'Impressum'. Below the navigation bar, there is a banner for 'powered by GameCreator' and a row of featured game thumbnails. The main content area is divided into three sections: 'Shooter', 'GcP@c', and 'Puzzle'. Each section contains several game templates with 'auswählen' buttons. A 'Lora vs. Doves' game preview is shown on the left, featuring a mobile game interface with a score of 997 and a score of 0. A 'Hilfe' section at the bottom left provides instructions on how to test a game on a mobile device. At the bottom of the page, there is a link for 'Informationen zum Datenschutz / AGBs'.

Yahoo Pipes

untitled*

Layout Expand All Collapse All Back to My Pipes New Save Publish...

► Sources

► User inputs

▼ Operators

- Content Analysis
- Count
- Filter
- For Each: Annotate
- For Each: Replace
- Location Extract
- Regex
- Rename
- Sort
- Split
- BabelFish
- Truncate
- Union
- Unique

► Url

► String

Location Builder

This module creates a location object from a string value. It understands most common address formats including zipcodes and many major airport codes.

Example: [Using the](#)

Location Builder

Location: Munich

Content Analysis

Pipe Output

Time taken: 0.285471s Refresh

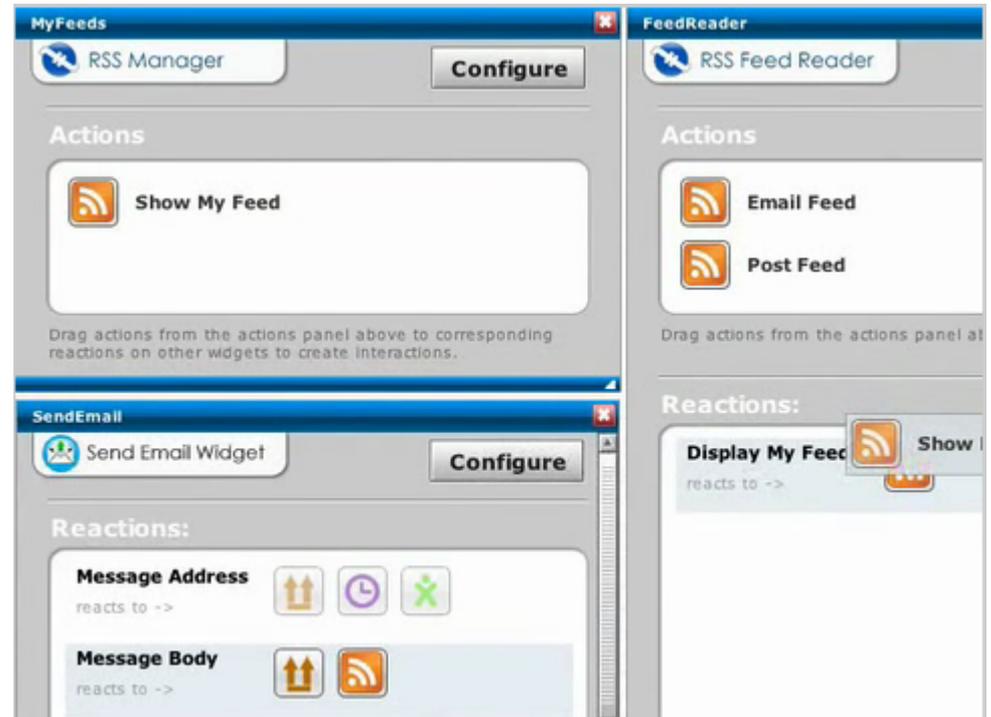
Debugger: Content Analysis (0 items)

- Online Editor für die Erstellung von eigenen Anwendungen
- Basisdienste sind Webservices (vorgegeben)
- Eigene Dienste (Pipes) können als Template für andere Dienste dienen oder aber Basisdienst für weitere Dienste



Weitere Online Editoren

- www.teqlo.com
- <http://www.dapper.net/>
- <http://www.coghead.com/>



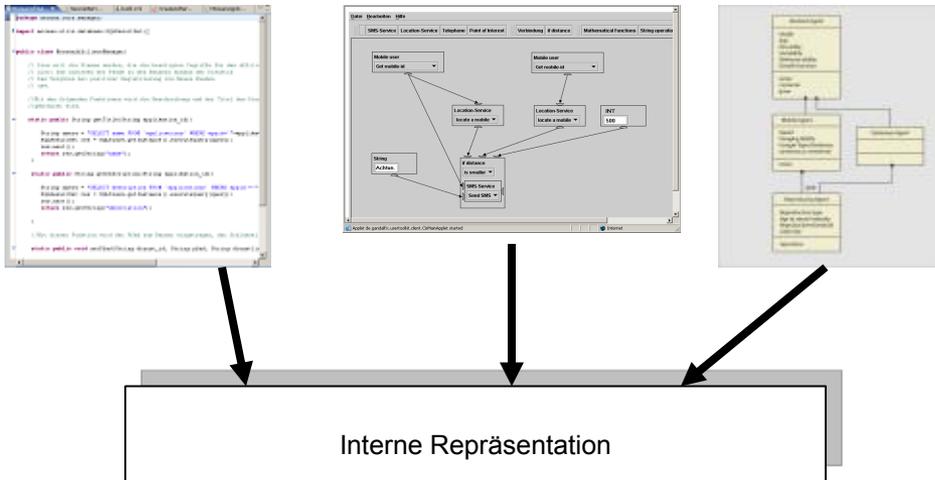
Forschungsrichtungen

- Beispiel: Wikipedia
 - „Solide Information über fast alles“
 - Fokus auf Wissen/Content
- Wikipedia für Anwendungen
 - Durch Tools können User Anwendungen entwickelt, diskutieren und erweitern
- Visuelle Programmierung
- Beschreibungssprachen (WSDL/BEPL)
- Sicherheitskonzepte (Server Sandbox), Ausführungsumgebungen



Visuelle Programmierung und Meta-Sprache

- Darstellungsschicht muss Kunden- und Domänen-abhängig sein
- Abbildung auf Meta-Sprache
- BPEL alleine nicht ausreichend



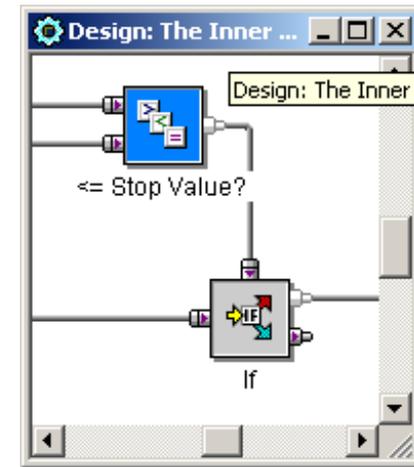
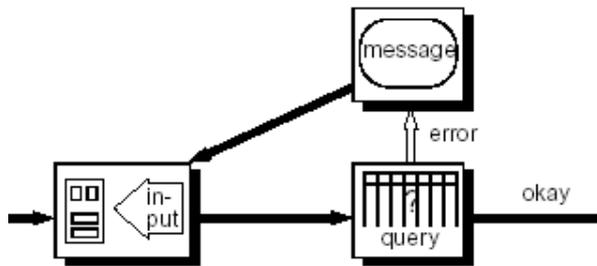
Darstellungsebene

Interne Repräsentation

Formale Beschreibung

Interdisziplinäre Herausforderung

- Aufgabe für Informatiker, Psychologen, Domänen-Experten
- Symbolauswahl/Visuelle Darstellung (Kognition)



- Evaluierung und Auswahl verschiedener Konzepte
- Kunden und Domänen abhängige Sprachen

Herausforderungen

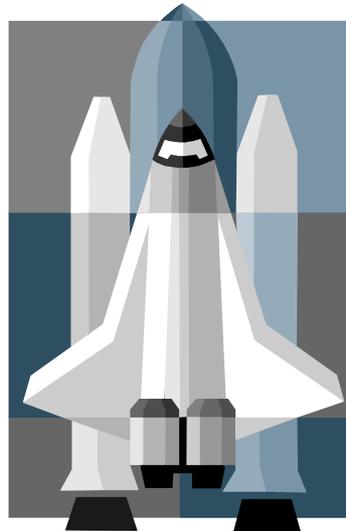
- Sicherheit

- Brücke schlagen zwischen „normalen“ Nutzern und Entwicklern

- Domänen abhängige Tools und Sprachen
 - Visuelle Programmierung
 - Meta Tools

- Webservice
 - Beschreibungssprachen für Komposition
 - BPEL und vergleichbares sind erster Schritt
 - Webservice umfassen nur syntaktische Beschreibung (WSDL)
 - Semantische Beschreibung ist notwendig





Danke ... Fragen?

Kontakt:

Dr. Peter Dornbusch

dornbusch@fireflow.de

Phone: +49 (89) 210 2377 11

Fax: +49 (89) 210 2377 19

Mobil: +49 (162) 906 1974