

Rich Internet Applications – Technologien

Leif Hartmann
INF-M3 – Anwendungen 2 - Wintersemester 2007/2008
08. Januar 2008

Inhalt

- ▶ Einleitung
 - ▶ Kategorisierung
 - ▶ Technologien
 - ▶ E c h o
 - ▶ G o o g l e W e b T o o l k i t
 - ▶ O p e n L a s z l o
 - ▶ X M L 1 1
 - ▶ N e x a w e b
 - ▶ G o o g l e G e a r s
 - ▶ Zusammenfassung
-

Einleitung

- ▶ Vorstellung der Technologien bezüglich:
 - ▶ Sprachen
 - ▶ Laufzeitumgebungen
 - ▶ Offline-Funktionalität
 - ▶ Testing
 - ▶ Verteilung der Logik
 - ▶ IDEs
 - ▶ Architektur

Kategorisierung

Stand-alone

Java:

- ▶ Java Web Start
- ▶ *Nexaweb*
- ▶ Droplets

Plug-in

Flash:

- ▶ Flex
- ▶ *OpenLaszlo*

Java:

- ▶ Java Applets
- ▶ AltioLive
- ▶ *Nexaweb*

AJAX

- ▶ JavaScript-Bibliotheken
- ▶ *Echo*
- ▶ Eclipse RAP
- ▶ *GWT*
- ▶ *Nexaweb*
- ▶ *OpenLaszlo*
- ▶ ZK
- ▶ *XML11*

Angelehnt an [1]

Echo

- ▶ Open Source (Mozilla Public License / LGPL)
- ▶ AJAX-basiert
- ▶ Entwickelt wird in Java
- ▶ Logik: Schwerpunkt Serverseite
- ▶ Vollwertige IDE
 - ▶ Beliebige Java IDE
 - ▶ EchoStudio (kommerziell)
- ▶ Keine eigene Offline-Funktionalität
- ▶ Testing: Java, Browser-Emulator

Button Components	
Button	Next → → Next
CheckBox	☐ Jumping ☐ Running ☐ Laughing
RadioButton	☒ Orangutan ☐ Chimpanzee ☐ Gorilla
Text Components	
TextField	The quick red fox jumps over the lazy brown dog
PasswordField	*****
TextArea	The quick red fox jumps over the lazy brown dog.

Echo

► Application Framework

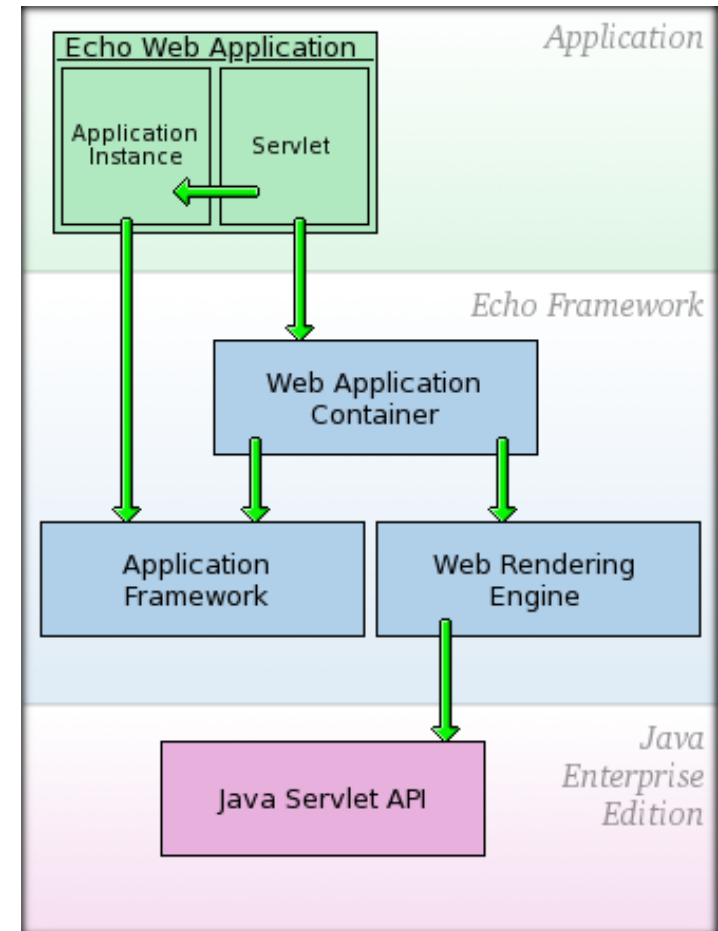
- Verwaltet den Zustand einer Anwendung und des UI

► Web Rendering Engine

- Server Engine: Java
- Client Engine: JavaScript
- Austauschbar

► Web Application Container

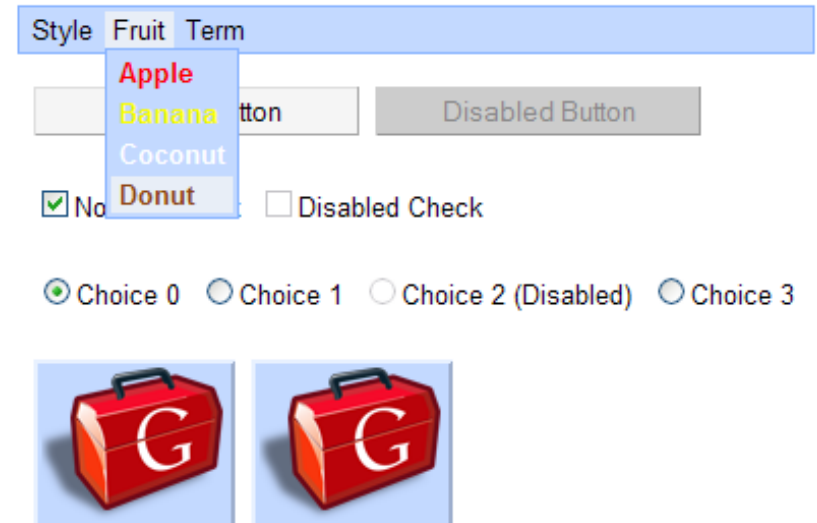
- Synchronisiert den Zustand des UI mit dem Webclient



Quelle: [3]

Google Web Toolkit

- ▶ Open Source (Apache License, Version 2.0)
- ▶ AJAX-basiert
- ▶ Entwickelt wird in Java
- ▶ Logik: Schwerpunkt Clientseite
 - ▶ Explizite RPC-Aufrufe für Serverlogik
- ▶ Vollwertige IDE
 - ▶ Beliebige Java IDE
 - ▶ Eclipse-Plugins
- ▶ Keine eigene Offline-Funktionalität
- ▶ Testing: Java (JUnit + GWT Integration), Browser-Emulator



Google Web Toolkit

► Hosted Mode

- Schnelle Code/Test-Zyklen
- Gute Debug-Möglichkeiten
- Ausgeführt als GWTShell (Java-Anwendung)

► Web Mode

- GWTCompiler
 - Java to JavaScript
 - Source-Code muss Java 1.4 kompatibel sein
- Ausgeführt in J2EE-Server

OpenLaszlo

- ▶ Open Source (CPL 1.0)
- ▶ Flash oder DHTML (AJAX)
- ▶ Entwickelt wird in
 - ▶ Client: XML, JavaScript
 - ▶ Server: Java (auch andere)
- ▶ Logik: Schwerpunkt Clientseite
 - ▶ Explizite RPC-Aufrufe für Serverlogik
 - ▶ JavaRPC
 - ▶ XML-RPC
 - ▶ SOAP

The screenshot displays a web application interface with several interactive elements:

- A vertical stack of buttons: "Submit", "Cancel", "OK", "Revert", and a disabled button labeled "disabled".
- A list of three checkboxes with labels: "I want to take a weekend flight", "Also search airports within 70 miles", and "I prefer non-stop flights".
- Two text input fields: the first contains "text entry here" and the second is disabled and contains "disabled".
- A dropdown menu currently showing "pistachio".
- A group of five radio buttons labeled: "tension", "distance", "delerium", "froth", and "persuasion".
- A horizontal slider control with a range from 0 to 100.
- A window titled "Frosty the Snowman" containing a text area with the lyrics: "Frosty the Snowman, was a jolly With a corncob pipe and a button Frosty the Snowman, is a fairytale He was made of snow, but the children say There must have been some magic For when they placed it on his head Ohhhhhhhh, Frosty, the Snowman and the children say he could laugh just the same as you and me. Thumpety thump, thump, thump".

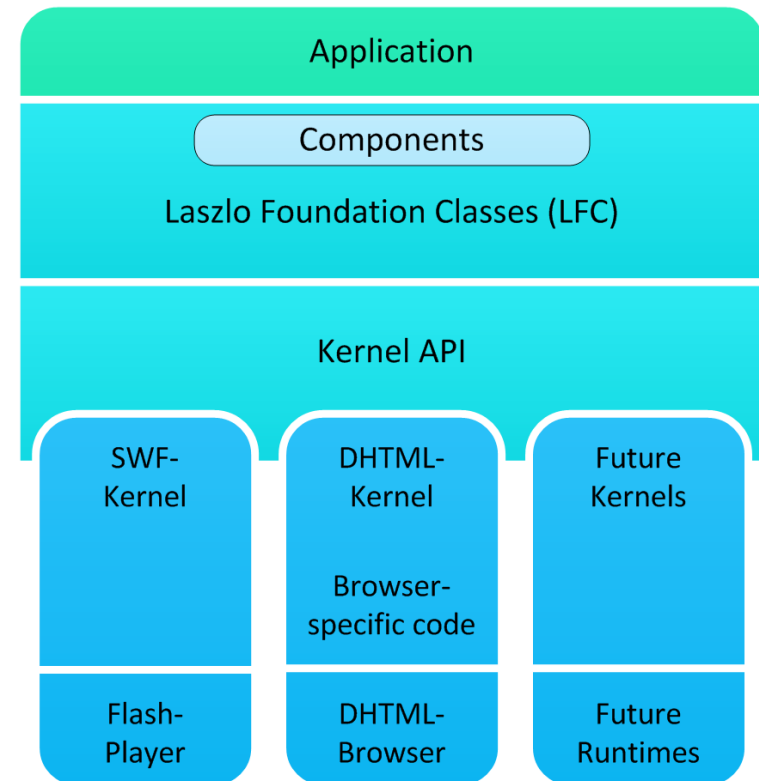
OpenLaszlo

► IDEs

- Spket IDE (Plug-in für Eclipse 3.2)
 - XML + JavaScript
- IDE for Laszlo (Plug-in für Eclipse)
 - Eingestellt

► Offline-Funktionalität: ?

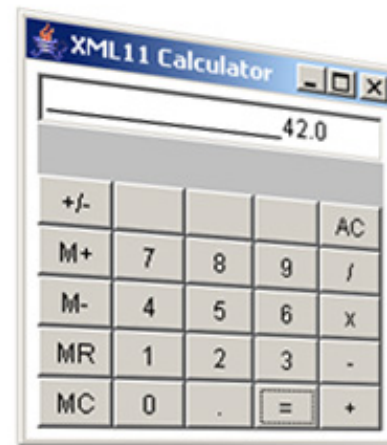
► Testing: LzUnit, Browser-Emulator



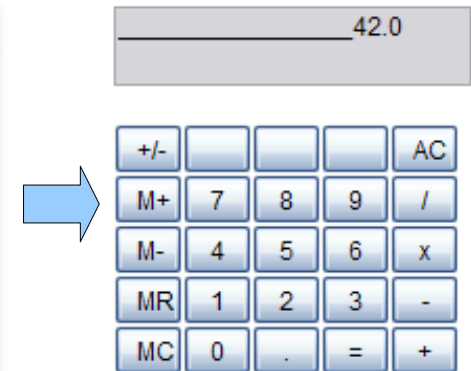
Nach [4]

XML11

- ▶ Open Source (GPL)
- ▶ AJAX-basiert
- ▶ Entwickelt wird in Java oder .NET
- ▶ Client/Server-Logik aufgeteilt
- ▶ Vollwertige IDE
 - ▶ je nach Sprache
- ▶ Keine Offline-Funktionalität
- ▶ Testing: je nach Sprache, Browser-Emulator

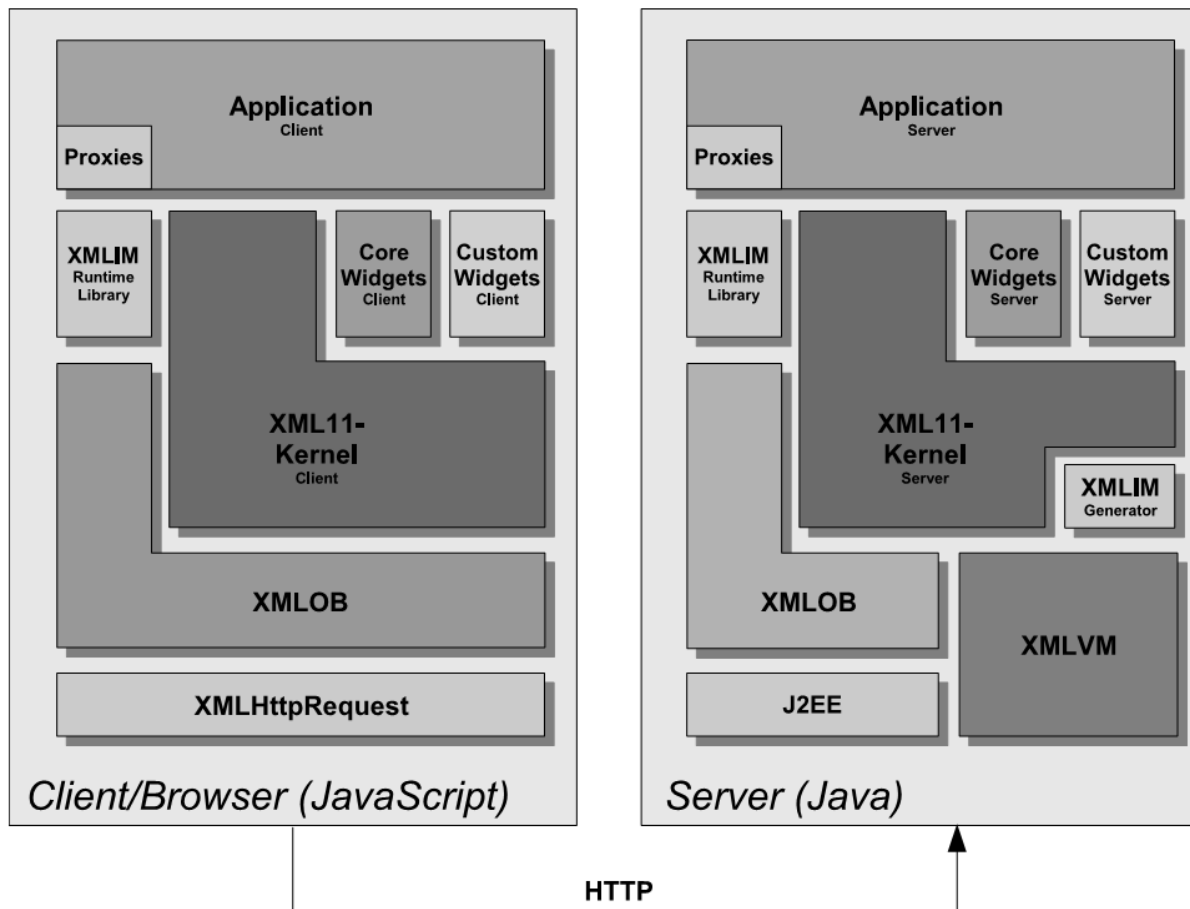


AWT



AJAX

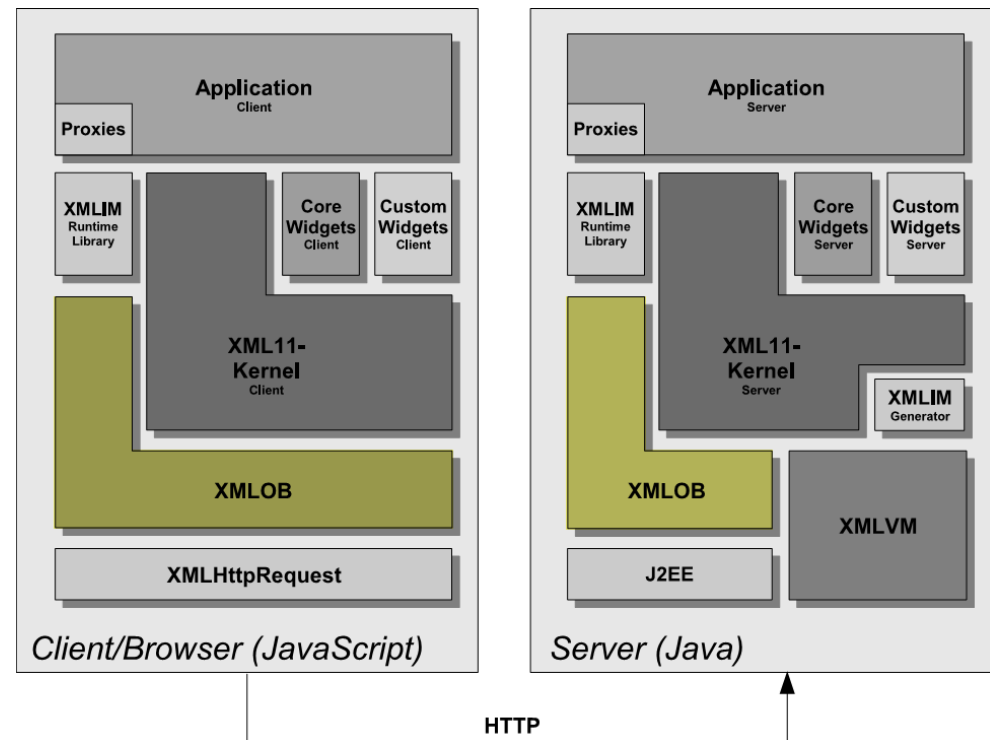
XML11



Quelle: [2]

XML1 1: XMLOB

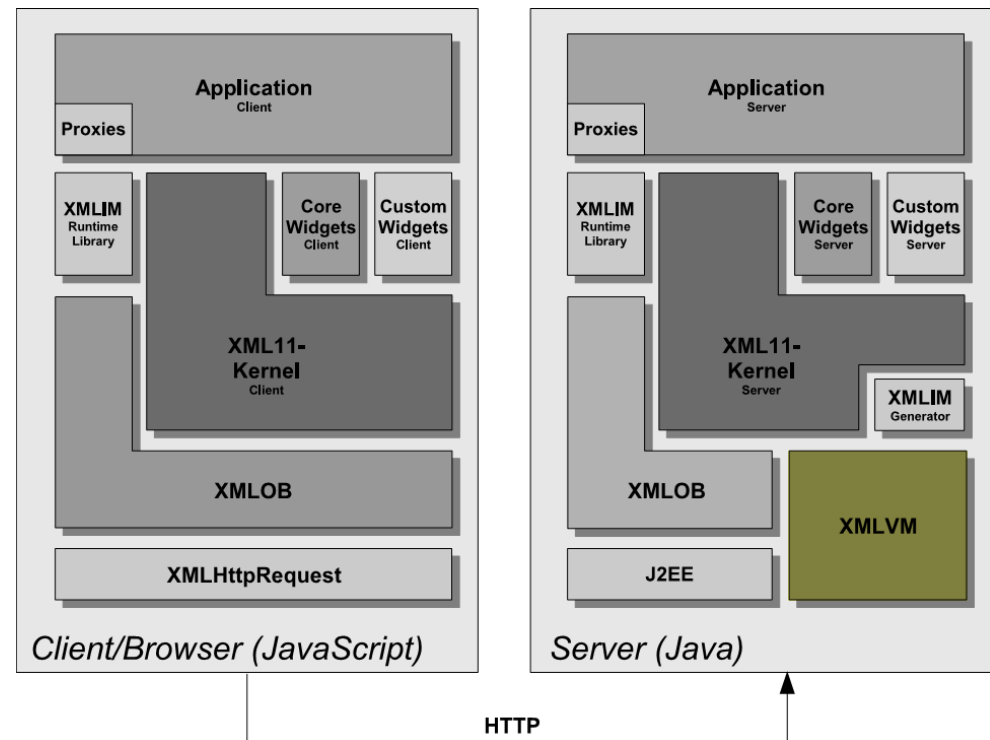
- ▶ XML Object Broker
- ▶ „light-weight“ Middleware
- ▶ (Un-)Marshalling
- ▶ Message-based transport
 - ▶ XMLCodec
 - ▶ JSONCodec
- ▶ Clientseitig: JavaScript
- ▶ Serverseitig: Java



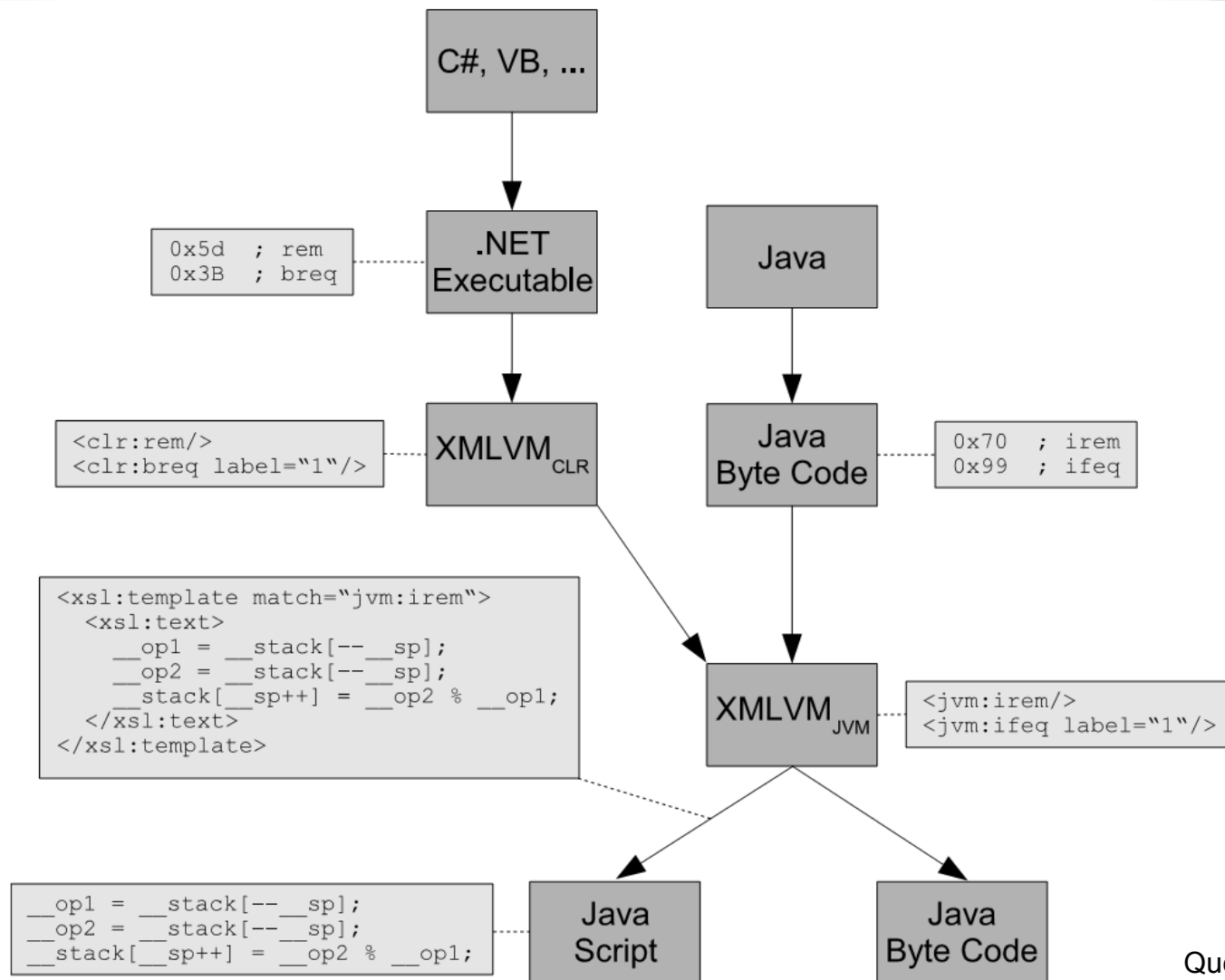
Quelle: [2]

XML11: XMLVM

- ▶ XML-based virtual machine model
- ▶ Übersetzen von Java- oder .NET-Byte-Code in JavaScript
- ▶ Analyse auf Byte-Code-Ebene
- ▶ Verwendet eine stack-based machine



Quelle: [2]



Quelle: [2]

Nexaweb

- ▶ Kommerzielle Lizenz
- ▶ Java Applet oder AJAX
- ▶ Entwickelt wird in
 - ▶ Client: X A L (X M L), J a v a S c r i p t oder J a v a
 - ▶ Server: J a v a
- ▶ Logik: Beliebig aufteilbar
 - ▶ X M L
 - ▶ J S O N
 - ▶ B i n a r y
 - ▶ S e r i a l i z e d J a v a O b j e c t s

Purchase Price \$ with % down for years

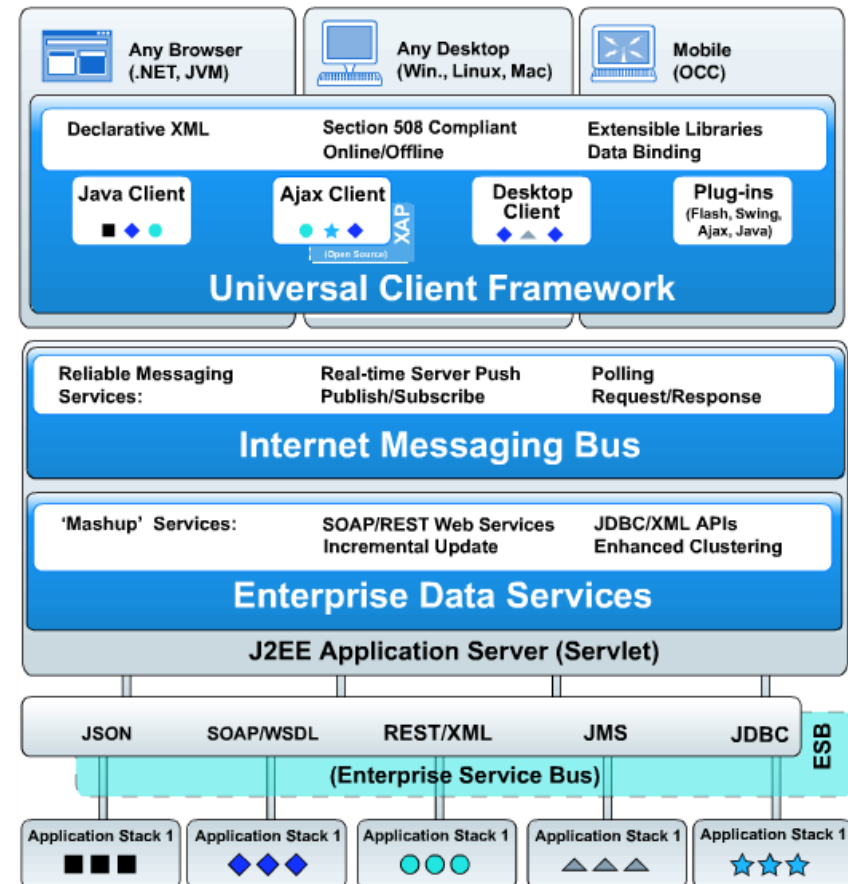
Interest Rate % Starting

[Click to Add Extra Payments](#)

Adding an extra \$ every month to your payment

Nexaweb

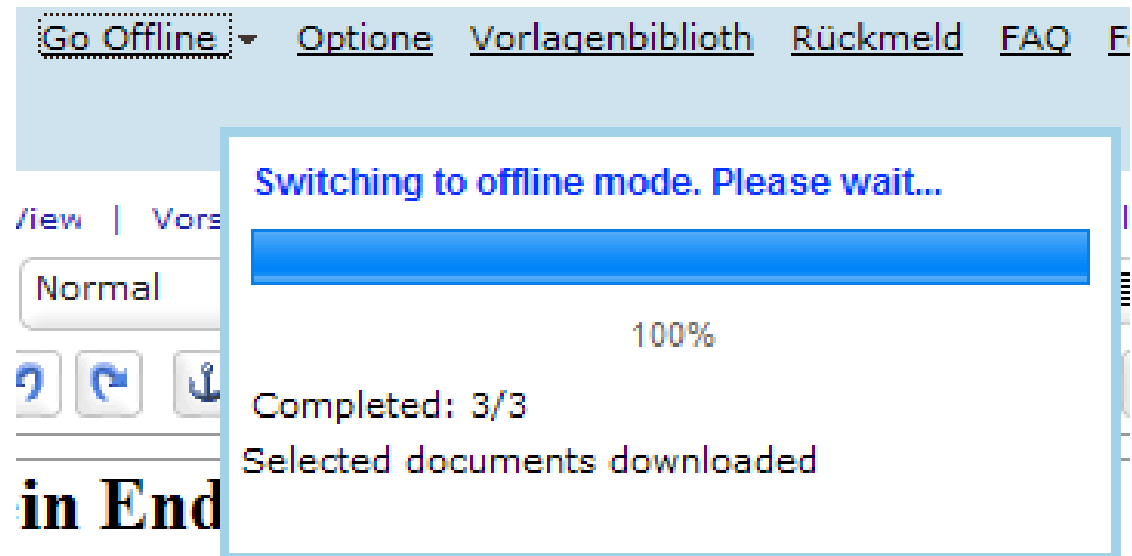
- ▶ **Vollwertige IDE**
 - ▶ Eclipse-basiert
 - ▶ WYSIWYG-Editor
- ▶ **Offline-Funktionalität**
 - ▶ Via Internet Messaging Bus
 - ▶ In-memory caching für Daten (auch mit AJAX-Client?)
 - ▶ Client-Persistenz nur mit Desktop-Client
 - ▶ Benötigt Installation
 - ▶ z.B. Java Web Start
- ▶ **Testing: Java, SOA Testing Tools**



Quelle: [5]

Google Gears

- ▶ Open Source (New BSD License)
- ▶ Offline-Funktionalität für AJAX-Anwendungen
- ▶ Plug-in-basiert
- ▶ JavaScript-APIs
 - ▶ LocalServer
 - ▶ Database
 - ▶ WorkerPool



Google Gears

LokalServer

- ▶ Speichert Dateien des Server lokal
 - ▶ „O f f l i n e C a c h e “
- ▶ Zuordnung von URLs zu lokalen Dateien
- ▶ Via Manifest-File
- ▶ Nur „komplette“ Seiten offline verwenden
- ▶ Offline- und Online-Version vergleichbar



Google Gears

Database

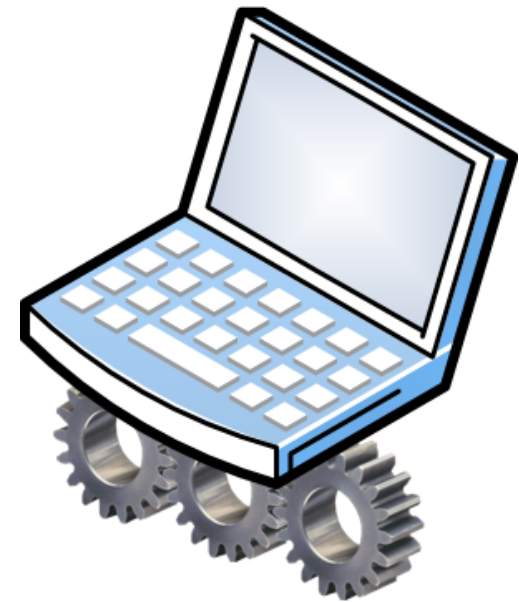
- ▶ Vollwertige SQL-Datenbank
- ▶ Verwendet SQLite
 - ▶ Volltextsuche via FTS-Modul
- ▶ Einfache JavaScript-API
 - ▶ „bind parameters“
- ▶ SDK bringt ein Query-Tool mit



Google Gears

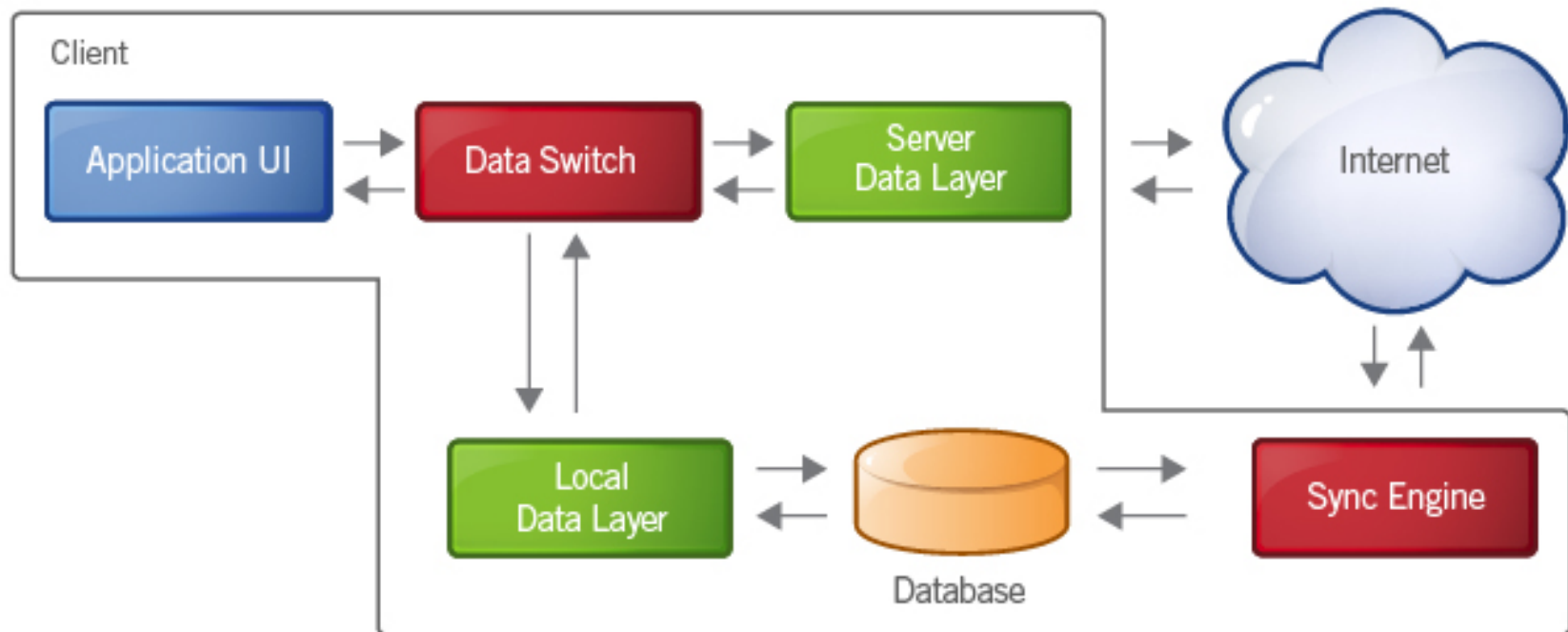
WorkerPool

- ▶ „Threads“ für JavaScript
- ▶ Blockieren des Browsers vermeiden
- ▶ Worker führt beliebigen JavaScript-Code im Hintegrund aus
- ▶ Kommunikation der „Threads“ via Strings
 - ▶ z.B. J S O N



Google Gears

Beispiel: Background sync architecture



Quelle: [7]

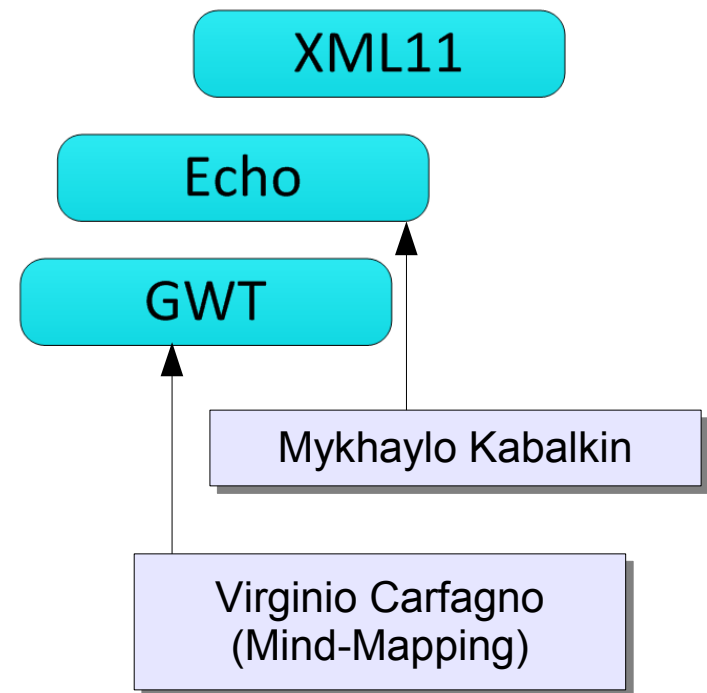
Zusammenfassung

Was ändert sich für Entwickler?

Web-Entwicklung



Desktop-Entwicklung



► Andere Fragestellungen

- W a r t b a r k e i t ?
- P r o d u k t i v i t ä t ?
- R o b u s t h e i t ?

Quellen

- [1] J. Farrell, G. S. Nezlek - *Rich Internet Applications: The Next Stage of Application Development* - In Proceedings of the ITI 2007 29 Int. Conf. on Information Technology Interfaces - 2007
 - [2] Puder, A. 2007 - *A cross-language framework for developing AJAX applications* - In Proceedings of the 5th international Symposium on Principles and Practice of Programming in Java (Lisboa, Portugal, September 05 - 07, 2007) - PPPJ '07, vol. 272 - ACM, New York, NY, 105-112 - <http://doi.acm.org/10.1145/1294325.1294340>
 - [3] NextApp - *NextApp . Echo . Technical Overview . Fundamentals* - <http://nextapp.com/platform/echo2/echo/doc/hltov/fundamentals.html> (Verifiziert: 03.01.2008)
 - [4] Laszlo Systems Inc - *Software Engineer's Guide to Developing OpenLaszlo Applications* - <http://www.openlaszlo.org/lps4/docs/developers/> (Verifiziert: 05.01.2008)
 - [5] Nexaweb Technologies Inc - *How Nexaweb works* - <http://nexaweb.com/home/us/index.html@cid=2295.html> (Verifiziert: 05.01.2008)
-

Quellen

- [6] Nexaweb Technologies Inc - *How Nexaweb Solves Connectivity* -
<http://www.nexaweb.com/home/us/index.html@cid=1836.html>
(Verifiziert: 05.01.2008)
- [7] Google - *Architecture - Google Gears API - Google Code* -
<http://code.google.com/apis/gears/architecture.html> (Verifiziert: 03.01.2008)
- [8] Virginio Carfagno - *Evaluation des Google Web Toolkits durch Entwicklung einer ajaxbasierten Mind-Mapping-Anwendung* - HAW Hamburg, Diplomarbeit - 2007

Fragen?

Leif Hartmann
INF-M3 - Seminar/Ringvorlesung - Wintersemester 2007/2008
07. Dezember 2007