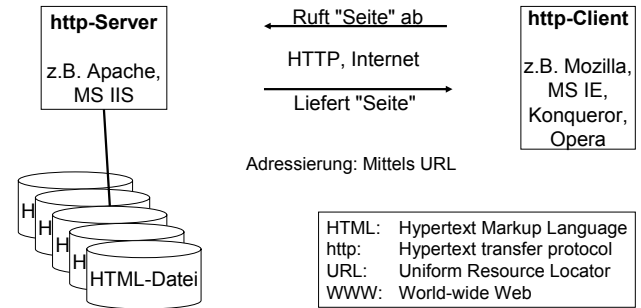


Ruby: Ausblick

Interessante Anwendungsgebiete
Aktuelle Trends

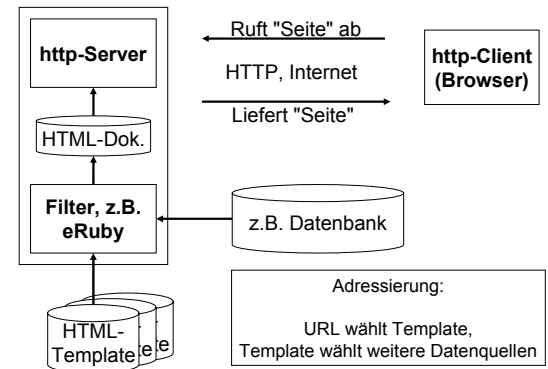
Grundfunktion eines Zugriffs im WWW auf statische Inhalte:



Ruby? Programmierung generell? Hier (noch) nicht benötigt!

- SSI (Server-Side Includes)
 - Dynamische Erzeugung spezieller Seitenteile
 - Beispiel: Datum, Copyright-Vermerke
 - Ansatz: Spezielle HTML-Kommentare zur Steuerung
 - Proprietär (typisch für Apache) und wenig flexibel
- Der Template-Ansatz
 - HTML-Dateien mit "Platzhaltern", die im Moment des Abrufs zu ersetzen sind mit aktuellen Inhalten, z.B. aus Datenbanken
 - Naheliegend z.B. für tabellarische Ergebnisse
 - Verallgemeinerung hin zur dynamischen Generierung ganzer Seiten ist fließend
 - Typisch für PHP!
 - Ruby-Alternative: eRuby (vgl. "Programming Ruby", ch. 14, p. 150f)
 - Kommerzielles Beispiel: SAP ITS, mit "AGate" und "WGate"

Prinzip der Seitengestaltung mit templates



Internet-Programmierung: CGI

- **CGI:** Delegation an spezielle Programme
 - Schaffung des "Common Gateway Interface (CGI)"
 - http-Server startet einen separaten (lokalen) Prozess
 - IPC über stdin, stdout sowie mittels einer Reihe von Umgebungsvariablen.
 - Vollwertige HTML-Dokumente mit http-Headern unter Beachtung spezieller "escaping"-Regeln sind zu beachten.
- Vorteil: Flexibilität
- Nachteil: Prozess-Overhead
 - Jede Anfrage startet einen eigenen CGI-Prozess!
- Werkzeuge:
 - Mit CGI gewann Perl enorm an Bedeutung
 - Ruby: Übernahme + Weiterentwicklung von Perl's Modul "CGI"
 - Auswege bei Performance-Engpässen: Interpreter in http-Server integrieren (mod_perl, mod_ruby), oder speicherresident (fast-cgi).

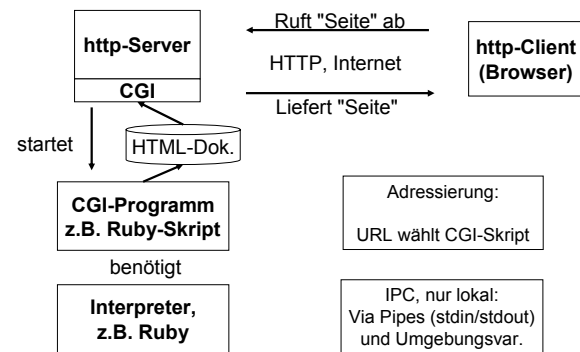
06.01.2004

H. Werntges, FB Informatik, FH Wiesbaden

5

Internet-Programmierung: CGI

Seitenaufbau mit CGI



06.01.2004

H. Werntges, FB Informatik, FH Wiesbaden

6

Internet-Programmierung

- Auswege aus CGI'S Performance-Engpässen:
 - Interpreter in http-Server integrieren (mod_perl, mod_ruby), oder speicherresident arbeiten (fast-cgi).
- Hinweise
 - Hier nicht näher behandelt
 - FCGI wird in "Programming Ruby" nicht behandelt, wohl aber von "Ruby Developer's Guide"
 - FastCGI ist generell eine sehr interessante Alternative, auch wenn sie eher selten erwähnt wird.
 - Besonders interessant z.B. bei zeitaufwändigen, aber nur einmal notwendigen Schritten wie dem Anmelden bei einer Datenbank.
 - Weitere Informationen unter <http://www.fastcgi.com>

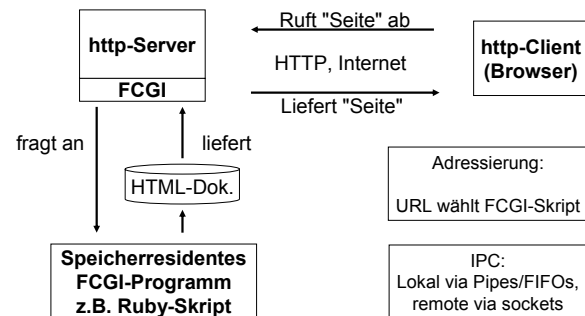
06.01.2004

H. Werntges, FB Informatik, FH Wiesbaden

7

Internet-Programmierung: FastCGI

Seitenaufbau mit FastCGI



06.01.2004

H. Werntges, FB Informatik, FH Wiesbaden

8

Internet-Programmierung

- Active Server Pages (ASP) und Java Servlets:
 - Proprietäre Antworten auf CGI's Performanceproblem
 - ASP (nur MS IIS) bzw. Servlets erfüllen ähnliche Aufgaben wie CGI-Skripte, aber innerhalb des Prozessraums des http-Servers.
 - Der http-Server muss dazu (proprietär) erweitert werden, z.B. um einen VB-Interpreter oder eine Java VM aufzunehmen.
- Ruby:
 - Hier noch nicht gesichtet: Wer proprietär erweitert, bleibt in der Regel innerhalb der so gewählten Technologie.

RDoc und Stdlib Doc

RDoc: Ruby Documentation Project

- **Konzept:**
 - Wiki-artiger Markup in Kommentarblöcken
 - Parsen von Ruby-Quellcodes mittels "rdoc", dabei
Extraktion der Markups
Analyse der Klassen und Methoden
 - Erzeugung von Output in verschiedenen Formaten (Default: HTML)
- **Demo:**
 - lotto+rdoc.rb
- **Anwendung:** Das Stdlib-Projekt
 - Demo, aktueller Stand (31.12.03) - Version für Ruby 1.8 +

Weiterführendes Material

- RAA - Ruby Application Archive
 - Analog zu Perl's "CPAN"
 - Zentrale Sammelstelle für Ruby-Module
<http://raa.ruby-lang.org/>
- RubyGarden
 - Diskussionsforen, Einführungsmaterial und vieles mehr rund um Ruby:
www.rubygarden.org,
www.rubygarden.org/ruby?HomePage
 - Besonders empfohlen: "Coding in Ruby"
www.rubygarden.org/ruby?CodingInRuby

Die Zukunft von Ruby

- Aktuell: Ruby **1.8.1**
 - Erschienen 25. 12. 2003
 - Nur noch maintenance updates erwartet
- Zwischenschritt: Ruby 1.9.x
 - Experimentelle Version, nur für Tester & Entwickler
 - M17N (Unterstützung zahlreicher nationaler Zeichensätze), neue Regexp-Engine "Oniguruma", "generational" GC
- Großes Ziel: Ruby 2.0 "Rite"
 - Noch keine Zeitaussage, Hinweise unter
<http://www.rubyist.net/~matz/slides/rc2003/>
 - Einschließlich Bytecode-Engine & noch schnellerem GC
 - Nicht vollständig abwärtskompatibel !



- Tipps zum Wiederholen
 - Kapitel "Ruby Crystallized"!
 - Welche Kap. von "Programming Ruby" haben wir behandelt?
 - "Ruby in a Nutshell"
 - Betonung der Praktikumsübungen!
- Fragen, Anregungen, Kritik
 - Vorlesung
 - Praktikum
- Bereits erkannter Bedarf (Evaluation, Dialoge):
 - Frühere Bereitstellung der neuen Übungen
 - 2. Teil der Registry-Aufgabe