

Interreg



EUROPÄISCHE
UNION

Österreich-Tschechische Republik

Europäischer Fonds für regionale Entwicklung

INFORMATIK

Grundlagen der Webanwendungen



UNIVERSITY
OF APPLIED SCIENCES
UPPER AUSTRIA



EUROPÄISCHE UNION

Inhalt

1. Kommunikation, Netzwerke, Protokolle.....	3
1.1. Kommunikationsprotokolle.....	3
1.2. TCP/IP.....	4
1.2.1. Anwendungsschicht.....	5
1.2.2. Physikalische Schicht.....	5
1.2.3. Datenverbindungsschicht.....	5
1.2.4. Internet-Schicht.....	6
1.2.5. Transportschicht.....	6
1.3. ICMP.....	7
1.3.1. Paketgröße.....	7
1.4. IPv4.....	8
1.5. IPv6.....	9
1.6. ICMP.....	9
1.7. TCP.....	9
1.8. UDP.....	10
1.9. SCTP.....	10
2. Sprachen für die Präsentation von Webinhalten.....	11
2.1. Einführung in HTML.....	11
2.1.1. Geschichte von HTML.....	11
2.1.2. Erstellung von HTML-Seiten.....	11
2.1.3. In HTML verwendete Begriffe.....	11
2.1.4. HTML.....	13
2.2. CSS.....	14
2.2.1. Geschichte von CSS.....	14
2.2.2. Anwendungen von CSS.....	14
2.2.3. Syntaxe.....	16
2.2.4. CSS-Stile.....	18
2.3. DHTML.....	19
2.3.1. CSS-Stile und -Klassen, Bezeichner und Stil.....	19
3. Client-seitige Logik - JavaScript.....	20
3.1. Was ist JavaScript.....	20

3.1.1.	Die Sprachmerkmale	21
3.1.2.	Erläuterung des Skripts	21
3.1.3.	Skripterstellung.....	23
4.	Web-Architektur	25
4.1.	Webarchitektur-Design.....	25
4.2.	Programmierung von Webanwendungen.....	26
4.3.	Technologie unserer Webanwendungen	29
4.4.	Was ist eine Webarchitektur?	29
4.4.1.	Wie man eine perfekte Architektur schafft.....	31
5.	PHP – basics	35
5.1.	Was ist PHP?.....	37
6.	Verarbeitung von http-Requests.....	40
6.1.	Anfragemethoden	40
6.1.1.	Request Headers	41
6.1.2.	Response-Header	42
6.2.	Grundlegende Merkmale des HTTP-Protokolls	42
6.2.1.	HTTP-Protokoll Anforderungsformat	43
7.	HTTP-Protokoll- Anforderungsmethoden	45
7.1.	GET-Methode	45
7.2.	POST-Methode.....	46
7.3.	Verwendung von Datenquellen	47
7.3.1.	Was ist ODBC?.....	47
8.	Datenquellen	49
9.	Datenbasis Ansatz.....	52
10.	Strukturierte und unstrukturierte Daten.....	54
10.1.	Strukturierte Daten	54
10.2.	Unstrukturierte Daten	55
10.2.1.	Technologische Merkmale von Data Warehouses	56
10.2.2.	Logische Struktur des Data Warehouse.....	56
11.	AJAX	58
12.	Frameworks	63
12.1.	Was ist ein Framework?.....	65
13.	Die 10 besten PHP-Frameworks für Entwickler	67

I. KOMMUNIKATION, NETZWERKE, PROTOKOLLE

I.1. Kommunikationsprotokolle

Im Kontext von Computernetzwerken sehen wir oft Kommunikationsprotokolle. Sie definieren genau die Art und Weise, wie die Kommunikation eine bestimmte Funktion über alle Ebenen hinweg erfüllen kann. Es gibt Protokolle für das Senden von Daten, das Erstellen sicherer Kanäle und das Suchen nach IP-Adressen, die dem Domännennamen entsprechen, oder das Zustellen von E-Mails, etc. Das Protokoll ist beiden Kommunikationspartnern bekannt und beschreibt genau, welcher Inhalt, welche Reihenfolge und welches Timing für die Übertragung verwendet wird. Jede Abweichung von dieser strukturierten Kommunikation kann als Fehler interpretiert werden.

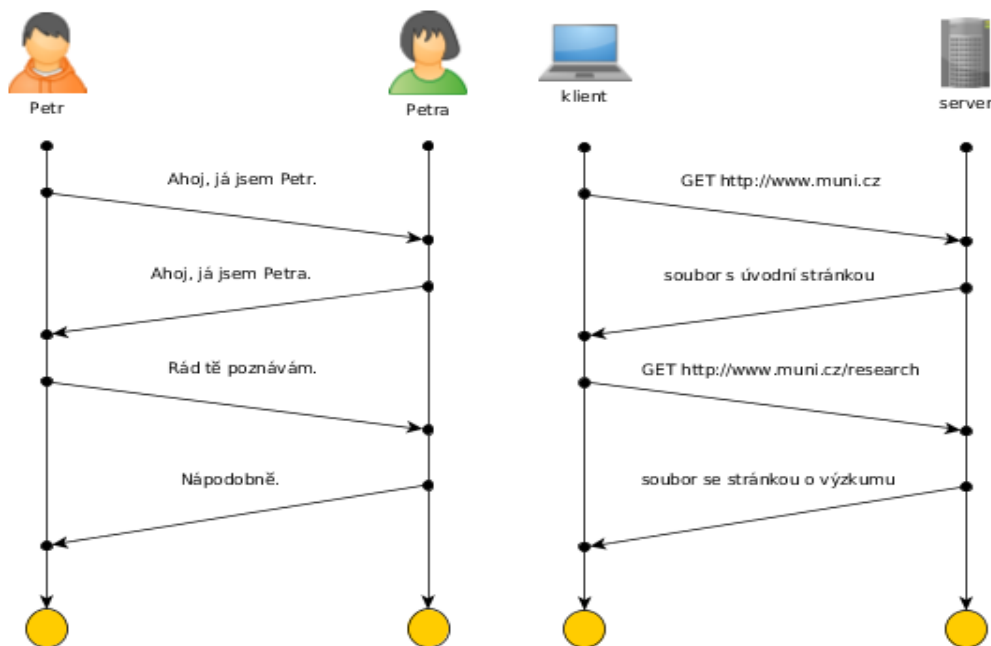


Abb.: Darstellung des Kommunikationsprotokolls

1.2. TCP/IP

Das wesentliche Prinzip der Computernetzwerkarchitektur ist die Aufteilung der Kommunikation in Schichten durch Abstraktion. Jede Schicht ist dafür verantwortlich, die Übertragung zu beschreiben, die die Anwendungsebene über physikalische Datenverbindungskommunikation startet. Das TCP/IP-Netzwerkmodell ist der Eckpfeiler aller bestehenden Netzwerke sowie des gesamten Internets. Der Name leitet sich von zwei wichtigen Protokollen ab, die das Datenrouting und den Transport zwischen den Knoten sicherstellen. Das IP-Protokoll beschreibt die Adressierung von Knoten, paketweise zerlegte Daten und deren Weiterleitung innerhalb des Netzwerks.

Die Kommunikation zwischen den gleichen Schichten zweier verschiedener Systeme wird durch das Kommunikationsprotokoll unter Verwendung einer Datenverbindung gesteuert, die von der benachbarten unteren Schicht hergestellt wird. Die Architektur erlaubt es, Protokolle einer Schicht auszutauschen, ohne die anderen zu beeinflussen. Die TCP/IP-Architektur ist in vier Schichten unterteilt (im Gegensatz zum OSI-Referenzmodell mit sieben Schichten):

- Anwendungsschicht
- Transportschicht
- Internet-Schicht
- Netzwerkschnittstelle

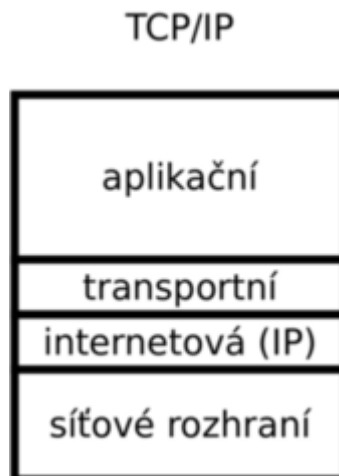


Abb.: TCP/IP-Netzwerkmodellschichten

TCP ist ein Transportprotokoll, das einen "verknüpften" Dienst auf IP-Übertragung aufbaut: Es stellt die Steuerung der erfolgreichen Übertragung und potenziellen Weiterleitung von fehlenden oder beschädigten Teilen sicher und wählt eine optimale Übertragungsgeschwindigkeit.

Die TCP/IP-Protokollfamilie, die derzeit für die absolute Mehrheit der Netzwerkkommunikation verwendet wird, wurde zwischen den 70er und 80er Jahren mit dem Ziel entwickelt, ein robusteres Modell der Netzwerkkommunikation zu schaffen, das bis zu einem gewissen Grad in der Lage ist, Verluste von Teilen des Netzwerks zu bewältigen. Die entscheidende (und damals revolutionäre) Idee ist die Paketvermittlung: Daten werden nicht als kontinuierlicher Strom übertragen, sondern in separaten Blöcken (Paketen) und jeder Netzwerkknoten allein bestimmt, in welche Richtung Pakete gesendet und weitergeleitet werden.

In der Praxis basiert diese Idee auf einer Reihe von Protokollen, die die notwendigen Funktionen implementieren. Protokolle werden typischerweise in mehrere Ebenen (Layer) unterteilt. Anstelle des abstrakten ISO/OSI-Modells, das mit sieben Schichten arbeitet, wird für die TCP/IP-Interpretation meist ein vereinfachtes Fünf-Schichten-Modell verwendet (einige Protokolle im TCP/IP-Modell implementieren die Funktion mehrerer Schichten des ISO/OSI-Modells).

I.2.1. Anwendungsschicht

Die Anwendungsschicht ist die oberste Schicht, die sich auf die Anwendungsprotokolldaten der einzelnen Netzwerkdienste bezieht. Es beinhaltet Netzwerkanwendungsprotokolle: elektronische Post, HTTP (Websites), DNS (Domain-Service) und andere. Es gibt eine Vielzahl solcher Protokolle, z.B. HTTP, SMTP, FTP, NTP sind die am häufigsten bekannten. Für TCP/IP sind das Daten, die an einen Zielempfänger übertragen werden müssen. Die Implementierung ist bis zu den unteren Schichten möglich.

I.2.2. Physikalische Schicht

Die physikalische Schicht ist die unterste Schicht im Modell. Im Gegensatz zu den oberen Schichten ist es keine Softwareschicht (Protokoll) und bezieht sich auf ein bestimmtes physikalisches Medium zur Datenübertragung. Ein Beispiel kann ein Twisted Pair, die Verkabelung in den meisten lokalen Ethernet-Netzwerken, Koaxialkabel, Glasfaser oder eine Telefonleitung sein. Das Medium muss nicht materiell sein - z.B. bei drahtlosen Netzwerken im Mikrowellenband (Wi-Fi, Breezenet) oder optischen Freiraumverbindungen (FSO).

I.2.3. Datenverbindungsschicht

Die Datenverbindungsschicht ist die niedrigste der Softwareschichten. Es ist das niedrigste Kommunikationsprotokoll, das für die Datenübertragung auf einem physikalischen Medium verwendet wird. Dieses Protokoll ist meist eng mit einem bestimmten Medium verbunden, aber diese Korrespondenz muss nicht 1:1 sein, z.B. ist Ethernet nicht nur auf Twisted-Pair-Verkabelung implementiert, sondern Sie sehen auch Implementierungen mit Koaxialverkabelung oder Glasfasern. Ein weiteres Beispiel für ein Datenverbindungs-

schichtprotokoll ist PPP, das für die Implementierung von DFÜ-Verbindungen oder seriellen Computerverbindungen verwendet wird. Das wichtige Merkmal von Data Link Layer Protokollen ist, dass sie sich nur mit der Kommunikation zwischen direkt verbundenen Knoten (daher der Name) befassen.

I.2.4. Internet-Schicht

Die Internet-Schicht ist für die globale Adressierung und Weiterleitung zuständig, in der Praxis am häufigsten durch ein IP-Protokoll (in zwei Versionen IPv4 und IPv6) implementiert. Obwohl es Adressen in Link-Layer-Protokollen und bei Ethernet-MAC-Adressen gibt, sind sie global eindeutig (oder sollten es zumindest sein), können sie jedoch nicht für das Paketrouting verwendet werden, da sie das Ziel nicht anzeigen. IP-Protokolladressen (IP-Adressen) sind jedoch hierarchisch aufgebaut, so dass die Delegation einzelner Bereiche die Netzwerktopologie widerspiegelt. Daher können Sie anhand der Ziel-IP-Adresse erkennen, wohin Sie ein Paket weiterleiten sollen, d.h. mindestens den folgenden Hop auf der Route. Zusätzlich zu dieser Grundfunktion ermöglicht das IP-Protokoll beispielsweise die Fragmentierung (Zerlegung zu großer Pakete in mehrere kleinere Datagramme) oder die Paketbezeichnung nach Art des Dienstes (ToS).

I.2.5. Transportschicht

UDP und TCP sind die am häufigsten verwendeten Protokolle der Transportschicht. UDP (User Datagram Protocol) ist ein minimal nachrichtenorientiertes Transportschichtprotokoll, das nur das Konzept des Ports als spezifische Prozessadresse (besser gesagt als Socket) innerhalb des Zielknotens einführt. Aber UDP ist immer noch ein zustandsloses Protokoll (keine Verbindung im wahrsten Sinne des Wortes), das sich nicht mit dem Problem der verlorenen Pakete oder ihrer Reihenfolge beschäftigt. Dennoch wird es häufig verwendet, weil es einfach und kostengünstig ist, insbesondere dort, wo diese Fragen nicht behandelt werden müssen.

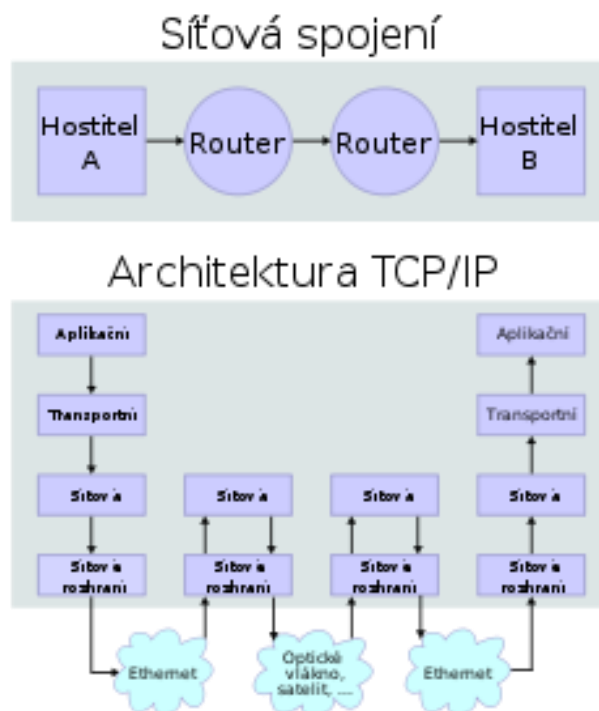
Der umgekehrte Ansatz wird durch TCP (Transmission Control Protocol) repräsentiert. Im Gegensatz zu UDP stellt es Verbindungen zwischen zwei Ports auf den Endknoten her. Aus Sicht von Client-Anwendungen ist das Verhalten dieser Verbindung vergleichbar mit einer Kommunikationsleitung zwischen zwei Prozessen (aber im Gegensatz zur Leitung ist TCP eine Zwei-Wege-Verbindung). Es wird sichergestellt, dass ein von der einen Seite versandter Stream (Bytefolge) auf der anderen Seite im gleichen Format empfangen wird. Das TCP-Protokoll ermöglicht das Erkennen und erneute Senden von verlorenen Daten sowie die Neuordnung von Daten aus Paketen, die in der falschen Reihenfolge ankommen. Da TCP-Transportprotokolle einen relativ hohen Komfort für die Anwendungsschicht gewährleisten, werden sie derzeit für den Großteil der Kommunikation verwendet. Der Nachteil ist ein höherer Overhead und eine relativ geringe Reaktion auf Ausfälle, weshalb UDP für bestimmte Zwecke (z.B. die meisten DNS-Abfragen oder VoIP) bevorzugt wird.

1.3. ICMP

ICMP (Internet Control Message Protocol) geht etwas über die Layerstruktur hinaus. Pakete (Nachrichten) dieses Protokolls werden direkt über ein IP-Protokoll übertragen. ICMP wird jedoch nicht als Transportprotokoll betrachtet, da es nicht für die Übertragung von Anwendungsdaten verwendet wird. Dieses Protokoll wird für Diagnose- und Servicezwecke verwendet. Die Beispiele für ICMP-Anwendungen sind paketbestimmte, nicht erreichbare Nachrichten, ICMP-Echo und Echo-Antwortpakete oder einige Dienstypen von Nachrichten (Redirect).

1.3.1. Paketgröße

Die maximale (theoretische) Größe des IP-Pakets beträgt 65535 B, aber die Datenverbindungsschicht ist oft ein limitierender Faktor. Da die meisten Pakete mindestens einmal über das Ethernet (oder das Äquivalent) laufen, wird die Größe der Pakete oft nach ihrer Grenze (1536 B) ausgewählt, daher ist der typischste Wert 1500 B. Aber natürlich ist dies der Maximalwert und die Pakete sind sehr oft viel kleiner, insbesondere für interaktive Anwendungen. Die Größe der IP- und TCP-Header beträgt 20-60 B (typischerweise nahe der unteren Grenze), UDP- und ICMP-Header sind 8 B und Ethernet-Header sind 14 B (zusätzlich ist 2 B am Ende des Pakets die Prüfsumme).



Die TCP/IP-Schichten, die den Datentransfer zwischen zwei Hosts über zwei Router ermöglichen.

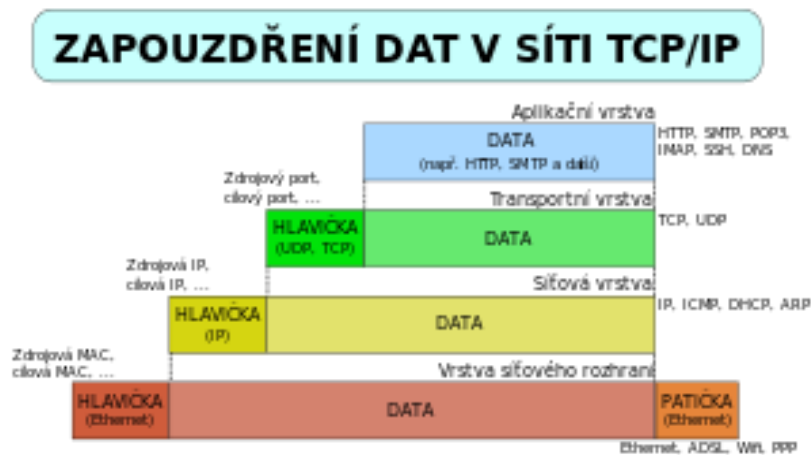


Abb.: Übersicht über die Kapselung von Anwendungsdaten auf TCP/IP-Schichten.

Angesichts der Problemkomplexität wird die Netzwerkkommunikation in sogenannte Schichten unterteilt, die die Hierarchie der Aktivitäten darstellen. Der Informationsaustausch zwischen den Schichten ist genau definiert. Jede Schicht verwendet Dienste der unteren Schichten und stellt Dienste für die oberen Schichten bereit.

Das Internetprotokoll ist das Basisprotokoll der Internetschicht und des gesamten Internets. Es generiert Datagramme basierend auf Netzwerk-IP-Adressen, die in ihren Headern enthalten sind. Es stellt den verbindungslosen Netzwerkdienst für die oberen Schichten zur Verfügung.

Am weitesten verbreitet ist derzeit das IP-Protokoll Version 4. Die neue Version 6, die den Mangel an Adressen in IPv4 und Sicherheitsprobleme behebt und andere Eigenschaften des IP-Protokolls verbessert, wird nur von ein paar Prozent der weltweit mit dem Internet verbundenen Geräte genutzt, aber die Zahl steigt rasant an.

I.4. IPv4

- Internet-Protokoll Version 4
- 32-Bit-Adressen
- App. 4 Milliarden IP-Adressen, jetzt nicht mehr ausreichend
- Format: xxx.xxx.xxx.xxx.xxx.xxx wobei xxx eine beliebige Zahl ist 0 - 255 (8 Bit)

1.5. IPv6

- Internetprotokoll Version 6
- 128 Bit Adressen
- Sicherheitsunterstützung
- Unterstützung für mobile Geräte
- QoS - Quality of Service Funktion
- Paketfragmentierung - Verteilung
- Nicht kompatibel mit IPv4

Das Address Resolution Protocol wird verwendet, um nach einer physikalischen MAC-Adresse mit einer bekannten IP-Adresse zu suchen. Bei Bedarf sendet das Protokoll ein Datagramm mit den gesuchten IP-Adressinformationen und adressiert diese an alle Stationen im Netzwerk. Der Knoten mit der gesuchten Adresse antwortet und gibt seine MAC-Adresse ein. Wenn sich der gesuchte Knoten nicht im gleichen Segment befindet, antwortet der jeweilige Router und gibt seine Adresse zurück.

1.6. ICMP

Das Internet Control Message Protocol wird für die Übertragung von Steuernachrichten in Bezug auf Fehlerzustände und besondere Übertragungsbedingungen verwendet. Es wird z.B. für Computerverfügbarkeitstests im Ping-Programm oder für die Verfolgung von Paketrouten zu einem anderen Knoten durch das Traceroute-Programm verwendet.

1.7. TCP

Das Transmission Control Protocol schafft eine virtuelle Verbindung zwischen den Endanwendungen, d.h. eine zuverlässige Datenübertragung. Die Protokollmerkmale sind:

- Zuverlässiger Transportservice, der alle Daten verlustfrei und in der richtigen Reihenfolge an den Empfänger liefert.
- Verbindungsdienst mit Phasen des Verbindungsaufbaus, der Datenübertragung und des Verbindungsabbaus.
- Transparente Übertragung beliebiger Daten.
- Vollduplexverbindung, parallele Zweiwege-Datenübertragung.
- Unterscheidung zwischen Anwendungen, die Ports verwenden.

I.8. UDP

Das User Datagram Protocol bietet einen unzuverlässigen Transportdienst für Anwendungen, die keine Zuverlässigkeit des TCP-Protokolls benötigen. Es fehlt die Verbindungsaufbau- und -abbauphase und bereits das erste UDP-Segment enthält Anwendungsdaten.

I.9. SCTP

Ein zuverlässiges Protokoll für die Übertragung von Datagrammen in mehreren Streams. Sie wird in erster Linie für die Telekommunikation eingesetzt. Es hat einige zusätzliche Funktionen, die TCP nicht bietet:

- Multihoming - der kommunizierende Knoten kann mehrere IP-Adressen haben.
- Aufspaltung des Datenstroms in Datagramme.
- Verwendung mehrerer Datenströme - zur Reduzierung der Blockierung der Kommunikation aufgrund eines fehlenden Datenblocks, der bei TCP auftreten kann.
- Routenauswahl und -verfolgung - Verwendung einer Alternative, wenn die Hauptadresse Verfügbarkeitsprobleme hat.

2. SPRACHEN FÜR DIE PRÄSENTATION VON WEBINHALTEN

2.1. Einführung in HTML

HTML ist eine einfache Auszeichnungssprache für die Erstellung von Websites, die nur gewöhnliche Textdateien sind, die einen Text und mehrere HTML-Tags enthalten, die die Bedeutung und das Aussehen der einzelnen Website-Teile bestimmen.

Neben html gibt es noch mehrere andere Sprachen, die für die Erstellung von WWW-Seiten verwendet werden: css, php, javascript, etc.

HTML ist eine Grundlage, wenn Sie etwas über CSS-Stile (CSS-Stile bestimmen das Aussehen von Webseiten) und später über PHP (bezogen auf die Programmierung) erfahren möchten.

2.1.1. Geschichte von HTML

Web erschien 1989 und wurde seither kontinuierlich entwickelt. Derzeit ist HTML 5 als Entwickler tätig.

2.1.2. Erstellung von HTML-Seiten

Wir können einen einfachen Texteditor, Notepad oder spezielle Editoren mit HTML-Codeunterstützung verwenden. Diese Programme heben die Syntax hervor und ermöglichen es uns, zwischen einem Code und einer Vorschau zu wechseln. Die Arbeit mit einem solchen Editor ist viel effizienter als die Arbeit mit einem herkömmlichen Notizblock.

2.1.3. In HTML verwendete Begriffe.

- Tag** ist eine grundlegende HTML-Markierung, geschrieben als `<tag>`.
- Attribut** wird innerhalb eines Tags geschrieben und setzt seine Eigenschaft. Es ist geschrieben als: `<tag attribute="value">`.
- Element** Schreiben einer Überschrift: `<h1>heading</h1>`.

Erste Seite

HTML-Seiten sind nur gängige Textdateien mit Tags.

Tags

Tags werden zwischen Klammern < > geschrieben, einige von ihnen sind gepaart, andere nicht.

Schreiben eines nicht gepaarten Tags:

```
<tag>
```

Schreiben eines gepaarten Tags:

```
<tag> einiger Text </tag>
```

Im Falle eines gepaarten Tags ist es wichtig, einen Schrägstrich zu schreiben, da der Explorer sonst nicht verstehen würde.

Attribute

Attribute werden direkt in den Tag geschrieben.

```
<tag attribute="value">  
<tag attribute="value">Gemeinsames Tag mit attribute.</tag>
```

Es ist verboten Tags zu überkreuzen

Tags können ineinander verschachtelt werden, dürfen sich aber nicht überkreuzen.

```
<b><i>fett kursiv</b></i>
```

Richtige Schreibweise:

```
<b><i>fett kursiv</i></b>
```

Größe der Zeichen

In html spielt die Größe der Zeichen keine Rolle, daher ist es möglich, sowohl <TAG> als auch <tag> zu schreiben, während in xhtml, der neuesten Version von html, weder Tags noch Attribute in Großbuchstaben geschrieben werden können.

In der URL ist es notwendig, die Größe der Zeichen zu beachten, d.h. FILE.html ist nicht dasselbe wie file.html.

2.1.4. HTML

HTML steht für Hypertext Markup Language. Hypertext-Markup-Sprache war Entwickler von SGML und wurde zur am weitesten verbreiteten Sprache für die Erstellung von Webseiten. In der Vergangenheit waren die am häufigsten verwendeten Versionen HTML 2.0, HTML 3.2, HTML 4.01 und HTML 5. Von HTML wurde auch XHTML (extended hypertext markup language) als Anwendung von XML entwickelt, was meiner Meinung nach nicht von höherer Bedeutung ist. Im Jahr 2010 wurde HTML 5 in Betracht gezogen; die meisten seiner Innovationen wurden bereits eingesetzt.

Struktur einer html-Datei

Die am häufigsten verwendete "Vorlage" einer Seite:

```
<!DOCTYPE HTML>
<html>
  <head>
    <meta charset="windows-1250">
    <title>Name</title>Name</title>
  </head>
  <body>

Seitentext

  </body>
</html>
```

2.2. CSS

2.2.1. Geschichte von CSS

CSS erschien um 1997. Es ist eine Sammlung von Methoden zur grafischen Gestaltung von Webseiten. Die Abkürzung steht für Cascading Style Sheets. Sie werden als Kaskadierung bezeichnet, da Definitionen eines Styles überlagert werden können, aber nur die letzte ist gültig. Dies ist jedoch im Moment nicht wichtig.

Es wurden auch CSS 2, verbesserte und ausgefeiltere Stilformen entwickelt, die jedoch im am häufigsten verwendeten Internet Explorer nicht gut funktionieren.

Wann Sie CSS verwenden sollten

Im Jahr 2015 könnte man sagen, dass das gesamte Web mit CSS formatiert ist. Aus dem bisher verwendeten HTML wurden nur noch Fetttheit und Kursivschrift verwendet. Daher ist es gut, einige Kenntnisse in CSS zu haben, wenn Sie Webseiten erstellen wollen. Zunächst einmal ist es notwendig zu wissen, wie HTML funktioniert. Ohne Grundkenntnisse ist es nicht empfehlenswert, mit CSS zu arbeiten. CSS ist zu untersuchen, wenn:

- Sie möchten formatierte Websites haben - Farben, Rechtfertigung des Textes, Spaltenlayout, etc.
- Sie schreiben oft Texte für das Internet, ohne Zeit mit der Formatierung zu verlieren,
- Sie beschäftigen sich mit Skripten, insbesondere mit Javascripten.
- Web mit vielen Seiten verwalten (oder planen), die ein ähnliches Design haben sollen

2.2.2. Anwendungen von CSS

Es gibt drei Möglichkeiten, CSS zu verwenden

Der Stil kann auf drei Arten bezeichnet sein (siehe die Beispiele unten). Es genügt, mindestens eine der folgenden Methoden zu kennen:

Im Quelltext für ein formatiertes Element über das Attribut `style="..."`. Dies wird als direkter Stil bezeichnet. Es ist völlig ungeeignet.

Mit Hilfe von "Style Sheet" im Kopf der Seite. Style Sheet ist eine Liste von Styles, die Informationen über die Formatierung liefert (z.B. grüne Überschriften). In einer Seite wird ein Style Sheet zwischen den Tags `<style>` und `</style>` geschrieben.

Mit Hilfe des externen Stylesheets - es ist eine Datei *.css, mit der die Seite mit einem Tag <link> verknüpft ist. Die Datei enthält ein Stylesheet. Der Hauptvorteil ist, dass viele Seiten mit einer Datei verknüpft werden können und alle Seiten ein ähnliches Design haben.

Beispiele

Erstellen Sie ein Paragraf in roter Schrift mit CSS. Es stehen drei Möglichkeiten zur Verfügung:

Direkter Stil

In der Quelle wird diese Absatzerklärung geschrieben:

```
<p style="color: red">Dieser Absatz wird rot sein.</p>
```

Erklärung: <p> ist eine Markierung, die einen Absatz bezeichnet. Das Attribut "style" ist ein allgemeines Attribut, das für jedes Element verwendet werden kann.

Style Sheet

In der Dokumentenüberschrift wird ein Stylesheet zwischen den Tags <style></style> geschrieben:

```
<style>
  p {Farbe: rot}
</style>
```

Absätze werden im Textkörper der Seite geschrieben:

```
<p>Dieser Absatz wird rot angezeigt. </p>
<p>Dieser Absatz wird auch rot sein, da alle Absätze rot sein
werden.</p>
```

Wenn wir wollen, dass nur einige Absätze rot sind, verwenden Sie "classes" und "identifier".

Durch externe CSS-Datei

Es wird eine Datei erstellt, die z.B. *styly.css* heißt. Es wird nur den folgenden Text enthalten:

```
p {color: red}
```

In der Überschrift eines html-Dokuments, das wir nach Stil ändern wollen, muss der folgende Link geschrieben werden:

```
<link rel="stylesheet" type="text/css" href="styly.css">
```

Im Textkörper des Dokuments werden alle Absätze rot dargestellt.

2.2.3. Syntaxe

Wie Sie vielleicht bemerkt haben, ist CSS kein Teil von HTML, daher sind sie anders geschrieben. Wenn die folgende Tabelle zu theoretisch erscheint, beachten Sie bitte nur die Beispiele im unteren Teil.

Direct style

```
<tag style="characteristics">styled element</tag>
```

Style sheet

```
<style>
  tag {characteristics
  2nd tag {characteristics}
</style>
```

Vereinfachte Charakteristiken:

```
characteristics: value; 2nd characteristics: 2nd value
```

Allgemeine Charakteristiken:

```
Characteristics: value [, value2] [; another characteristics]
```

Beispiele

Direct style

```
<p style="color: red;">text of red paragraph</p>
```

Style sheet

```
<style>
  p {color: red}
  body {background-color: yellow;}
</style>
```

Einfache Schreibweise der Charakteristiken:

```
color: red
```

Komplexe Schreibweise der Charakteristiken:

```
font-family: Arial, Arial CE, sans-serif; color: red;
```

Es ist zu beachten, wenn Anführungszeichen, Doppelpunkte, geschweifte Klammern, Semikolons und Kommas verwendet werden. Beispiel für eine korrekte Schreibweise:

```
h2 {color: green; background-color: yellow}
```

Leerzeichen und Zeilenenden spielen keine Rolle, sie können hinzugefügt und weggelassen werden. Die Zeichengröße spielt dabei keine Rolle. Es steht eine Liste der Merkmale und deren Werte zur Verfügung.

Der Explorer ignoriert die Werte, die er nicht erkennt.

In Style Sheets werden Kommentare ähnlich wie in Java zwischen `/* a */`. Zwei Schrägstriche funktionieren nicht.

Beispiel mit einer Überschrift

Es ist ganz einfach in Stylesheets oder externen CSS-Dateien.

```
<style>
    h1 {color: green;}
    h2 {color: blue;}
</style>
```

Auf diese Weise enthält das gesamte Dokument grüne Überschriften der ersten Ebene und blaue Überschriften der zweiten Ebene, wobei jedoch nur angenommen wird, dass die Tags `<h1>` und `<h2>` zum Schreiben eines Textes verwendet wurden. Mit anderen Worten, Stylesheets können nur in gut strukturierten Texten verwendet werden.

2.2.4. CSS-Stile

CSS-Stile sind kaskadierend, sie werden verwendet, um einen Stil einer Webseite zu erstellen (Farbe, Schriftart, Schriftgröße). Mit CSS kann eine Datei das Design des gesamten Webs beeinflussen.

Veraltete Methoden

Vor CSS-Stilen wurde für einen Webseiten-Stil das Element `` verwendet, das nicht mehr verwendet wird. Gegenüber CSS hat es folgende Nachteile:

- Wenn Sie den Stil eines Textes oft geändert haben, erschien dieser Tag sehr oft im Quellcode, was die Seite verlangsamte.
- Es ermöglicht, nur Schriftart, Farbe und Größe zu ändern. ``

2.3. DHTML

Vielleicht sind Sie schon auf die Abkürzung DHTML oder Dynamic HTML gestoßen. Diese Sprache wird fast ausschließlich aus JavaScript, VBScript (Sprache mit ähnlichen Eigenschaften wie JavaScript) und CSS-Stilen erstellt. Diese Sprache nutzt die Vorteile von HTML, JavaScript und CSS und schafft so ein perfektes Design und gut aussehende Seiten.

2.3.1. CSS-Stile und -Klassen, Bezeichner und Stil

CSS - Cascading Style Sheets - Cascading Styles wurden von Microsoft erstmals 1996 in Internet Explorer 3.0 implementiert. Der CSS-Stil entfernt `` vollständig und führt `<style>` ein. Mit Hilfe von CSS-Stilen ist es möglich, Farbe, Schriftart, Größe und viele andere zu definieren (Box, Unterstreichung, Kühnheit, Welligkeit, Display, Aufzählungen, Ränder...).

CSS-Stile werden hauptsächlich über Klassen und Identifikatoren angewendet. Diese ermöglichen es, einen CSS-Stil nur mit einem Attribut zu erstellen und der Benutzer muss einen Code nicht zehnmal wiederholen. Darüber hinaus ist es auch möglich, den Stil der Elemente (h1, Tabelle, etc.) über Selektoren zu definieren. So wird beispielsweise jedes Element `<input>` immer einen roten Text haben - dies ist mit einer CSS-Zeile möglich.

3. CLIENT-SEITIGE LOGIK - JAVASCRIPT

3.1. Was ist JavaScript

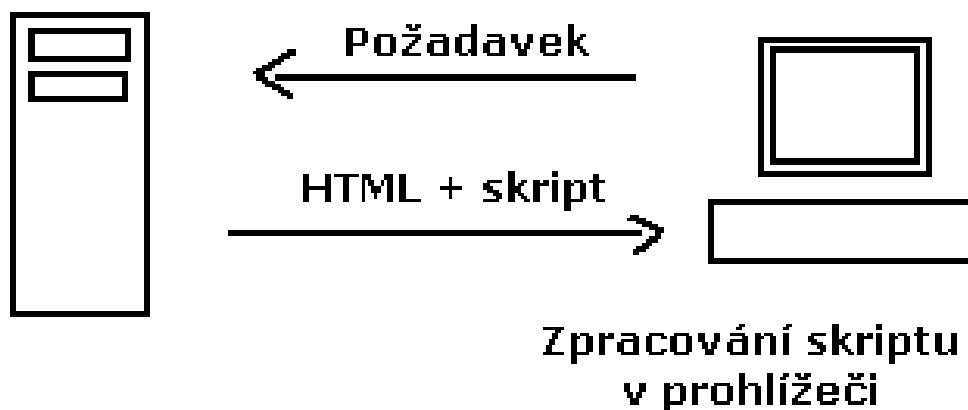
JavaScript ist eine Programmiersprache, die auf Websites verwendet wird. Es wird direkt in HTML-Code geschrieben, was aufgrund seiner Einfachheit ein großer Vorteil ist.

JavaScript ist ein Client-Skript. Das bedeutet, dass das Programm mit einer Seite an den Client (in den Explorer) gesendet und dort ausgeführt wird (im Gegensatz zu Serverskripten, die auf einem Server ausgeführt werden und der Client nur die Ergebnisse erhält).

Klientský skript

Server

Klient (čtenář)



Legende: klientský skript - Kundenskript, klient (čtenář) - Kunde (Leser), požadavek - Anforderung, zpracování skriptu v prohlížeči - Verarbeitungsskript im Explorer

Es gibt auch andere Sprachen von Client-Skripten, z.B. VBScript. Es wird jedoch so selten verwendet, dass, wenn man "Skripte" erwähnt, dies meist "JavaScripte" bedeutet.

JavaScript ist nicht Java

JavaScript wird oft mit Java verwechselt. Java ist eine separate Programmiersprache. Es hat nur eine ähnliche Syntax wie JavaScript.

Erforderliche Fähigkeiten

- HTML, Grundlagen von HTML mindestens
- Grundlagen der Programmierung

3.1.1. Die Sprachmerkmale

JavaScript ist eine Sprache

- interpretiert - es muss nicht kompiliert werden.
- beanstandet - es verwendet das Objekt eines Explorers und eingebaute Objekte.
- abhängig vom Explorer - funktioniert aber in den meisten Explorern
- Groß-/Kleinschreibung beachten - die Größe der Zeichen im EINTRAG ist entscheidend.
- seine Syntax ist ähnlich wie bei C, Java und dergleichen.
- Einschränkungen der Sprache
- JavaScript funktioniert nur innerhalb eines Explorers.
- Benutzer können JavaScript blockieren
- Es gibt verschiedene Versionen der Sprache und der Explorer, was zu häufigen Fehlern führt.
- es kann nicht auf Dateien (außer Cookies) oder andere Systemobjekte zugreifen.
- Es können keine Daten (außer Cookies) gespeichert werden.
- Dies macht es nur zu einer Sekundärsprache, die nur für HTML-Seiten gilt.

Wie man mit JavaScript umgeht

Nach der Beherrschung der Grundlagen empfiehlt es sich, Skripte auf anderen Websites zu beachten. Die meisten Skripte werden direkt in den Quellcode der Webseiten geschrieben, so dass es möglich ist, sie zu kopieren (einige Codes befinden sich in externen Dateien, aber auch diese können heruntergeladen werden).

3.1.2. Erläuterung des Skripts

Das Skript wird in HTML zwischen den Tags `<script>` und `</script>` geschrieben. Alles, was zwischen den Tags geschrieben wird, ist ein Programm, das in der Sprache Javascript geschrieben wurde.

Das Beispiel verwendet den Befehl `document.write()`. Dies ermöglicht das normale Schreiben in den Dokumentenstrom. Der geschriebene Text wird sofort im Explorer angezeigt.

Normaler Text muss zwischen Anführungszeichen geschrieben werden (im Gegensatz zu Variablen). Eine Zeile darf nicht zwischen den Anführungszeichen umgebrochen werden.

Jeder JavaScript-Befehl wird mit einem Semikolon oder einem Zeilenumbruch abgeschlossen.

Wie man das erste Skript erstellt

Alles, was in JavaScript erstellt wird, wird als Skript bezeichnet. Es kann auf der Seite platziert werden oder es ist möglich, einen Link darauf zu erstellen. In diesem Fall wird die Seite in der JavaScript-Seite hochgeladen. Separate Dateien, die in JavaScript geschrieben wurden, haben die Endungen .js oder .jse. Die Erweiterung .js ist gebräuchlicher. Das Einzige, was Sie für die Erstellung eines Skripts benötigen, ist ein Quelltexteditor (PSPad, Texteditor oder ein beliebiger HTML-Editor). Zum Surfen benötigen Sie einen Explorer (mindestens Internet Explorer und Mozilla Firefox, damit Sie Skripte in diesen am häufigsten verwendeten Explorern überprüfen können).

Einfügen eines Skripts in eine Seite

Das Skript wird zwischen den Tags `<script>` und `</script>` geschrieben. Sie können zwischen dem Abschnitt "Körper" oder "Kopf" eingefügt werden (abhängig vom Zweck eines Skripts).

```
<html>
  <head>
    ...
    <script type="text/javascript">
      javascript script body ..
    </script>
    ...
  </head>
  <body>

document body
  <script type="text/javascript">
    .javascript script body
  </script>

document body
  </body>
</html>

Tag <script>
```

Die Syntax des Tags `<script>` lautet wie folgt:

```
<script type="text/javascript" src="url of external file">
  <!--
      javascript script content
  //-->
</script>
```

Der Attributtyp bezeichnet einen Typ eines Skripts (im Falle von JavaScript "text/javascript"). Da es Explorer gibt, die JavaScript nicht verstehen, wird empfohlen, `<!--` am Anfang des Skripts und `//-->` am Ende des Skripts zu schreiben, sonst würde der Explorer das Skript als normalen Text schreiben (jetzt betrachtet er den Text als Kommentar und zeigt ihn nicht an).

3.1.3. Skripterstellung

In JavaScript ist es wichtig, zwischen Groß- und Kleinschreibung zu unterscheiden, daher ist `document.write` nicht dasselbe wie `DOCUMENT.write`. Diese Regel muss beachtet werden, sonst funktioniert das Skript nicht.

JavaScript ist eine plattformübergreifende, objektorientierte Skriptsprache, deren Autor Brendan Eich von der Firma Netscape ist.

Nun wird JavaScript in der Regel als interpretierte Programmiersprache für WWW-Seiten verwendet, die oft direkt in einen HTML-Seitencode eingefügt wird. Es steuert in der Regel verschiedene interaktive Elemente GUI oder erstellt Animationen und Bildeffekte.

JavaScript war ursprünglich ein Handelsname einer Implementierung von Netscape, die ursprünglich unter dem Namen Mocha, später LiveScript, entwickelt wurde. Es wurde im Dezember 1995 mit der Firma Sun Microsystems als Ergänzung zu den Sprachen HTML und Java angekündigt. Für die Version der Firma Microsoft wird der Name JScript verwendet. Dies wird von der Plattform.NET unterstützt.

Im Gegensatz zu anderen interpretierten Programmiersprachen (z.B. PHP und ASP), die noch vor dem Download aus dem Internet serverseitig gestartet werden, wird ein Programm in JavaScript in der Regel nach dem Herunterladen einer WWW-Seite aus dem Internet (auf Client-Seite) gestartet. Dies impliziert bestimmte Sicherheitseinschränkungen. Beispielsweise kann JavaScript nicht mit Dateien arbeiten, da es sonst die Privatsphäre des Benutzers gefährden könnte.

JavaScript kann auch auf der Seite eines Servers eingesetzt werden. Die erste serverseitige Implementierung von JavaScript war LiveWire der 1996 gegründeten Firma Netscape. Heutzutage gibt es mehr Möglichkeiten, einschließlich Open-Source-Implementierung Rhinola auf Basis von Rhino, gcj, Node.js und Apache.

Neben DHTML wird JavaScript zum Schreiben von Erweiterungen für viele Anwendungen verwendet, z.B. Adobe Acrobat.

JavaScript kann auch in den Betriebssystemen Windows mit dem Programm Windows Script Host verwendet werden und ersetzt so Batch-Dateien MS-DOS.

4. WEB-ARCHITEKTUR

4.I. Webarchitektur-Design

Wir beginnen nie damit, ein Web mit grafischem Design zu erstellen. Zunächst werden die Ziele der einzelnen Seiten definiert und deren Platzierung im Web festgelegt.



Warum ist es wichtig?

Dank des Designs der Webarchitektur ist es möglich, sich vorzustellen, wie das resultierende Web aussehen wird und wie seine Funktionen aussehen werden.

Auf diese Weise ist es möglich, potenzielle Fehler zu identifizieren und zu beheben und so Geld zu sparen, das für die spätere Reparatur programmierter Anwendungen erforderlich wäre.

Webstruktur-Design

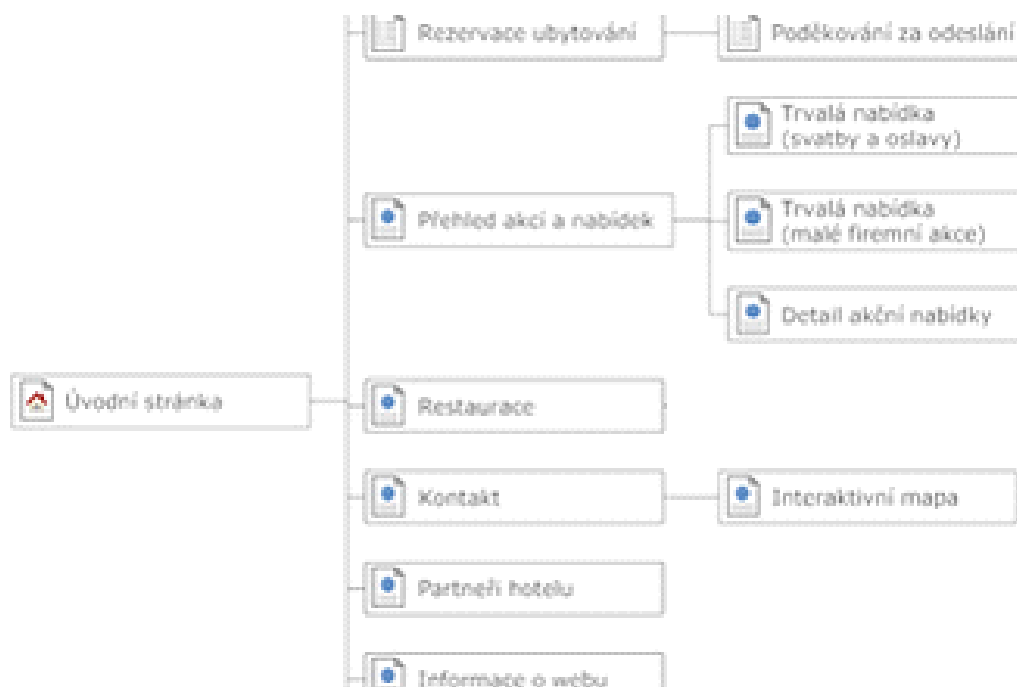
Im Strukturdesign müssen wir alle Webseiten, deren gegenseitige Verknüpfungen und übliche Benutzerszenarien definieren. Es ist notwendig, sicherzustellen, dass die Informationen angemessen strukturiert und einfach zu bedienen sind und zu Zielerreichungen führen.

Web-Navigationsdesign

Eine geeignete Navigation ist ein wichtiger Bestandteil eines Webs. Es ermöglicht einem Besucher, schnell zu finden, was er sucht. Diesem Aspekt ist bei der Gestaltung eines Webs besondere Aufmerksamkeit zu schenken.

Drahtgittermodellierung

Sobald wir den Inhalt des Webs kennen, kann mit der Wireframe-Modellierung begonnen werden, die die Grundlage für das grafische Design des Webs bildet.



4.2. Programmierung von Webanwendungen

Webanwendung bedeutet in der Regel ein serverseitiges Skript (ein Code, der die Programmfunktion sicherstellt). Es ist oft mit einer Datenbank verbunden, einem System, das Daten von Webanwendungen speichert (vereinfacht gesagt, kann man sich eine Datenbank als MS Excel-Datei vorstellen). Die Skriptausgabe ist eine Webseite, die zur Anzeige an den Explorer übergeben wird.

Schéma webové aplikace



Legende: schéma webové aplikace - Webaplikationsgraph, Skript - Skript, www stránka - www page, databáze - Datenbank

Die Aufgabe von Webanwendungen besteht vor allem darin, die Interaktion der Internetpräsentation mit ihren Besuchern zu erhöhen oder die Webadministration zu erleichtern, d.h. repetitive Arbeit bei der Erstellung von WWW-Seiten zu ersparen.

Abhängig von den Anforderungen an die Funktionalität kann eine Webanwendung aus nur wenigen Zeilen Code bestehen (z.B. beim Versenden von Kontaktformularen), aber es gibt auch Webanwendungen mit Tausenden von Zeilen.

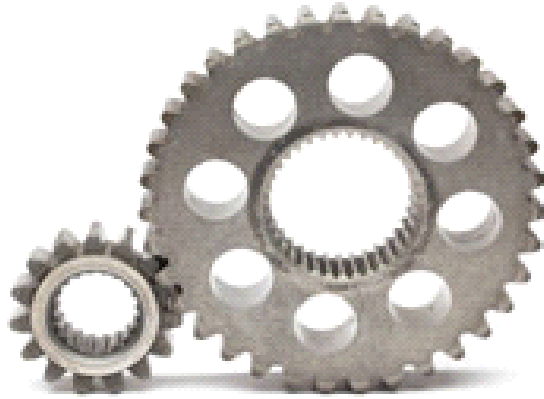
Komplexere Webanwendungen sind oft mit anderer Software innerhalb eines Unternehmens verknüpft, z.B. mit Bestellsystemen, Buchhaltungsprogrammen, etc. Dies ermöglicht es, Kosten für die menschliche Arbeit zu sparen. Es ist auch möglich, die Anwendung an Online-Zahlungssysteme anzubinden.

Beispiele für einfache Webanwendungen

- Kontaktformular
- Gästebuch
- Diskussionen oder Chat
- Kataloge und Preislisten
- Wörterbücher
- Bannersysteme

Bei größeren Präsentationen ist es besser, die gesamte Präsentation dynamisch anzugehen, z.B. über Vorlagen oder den Einsatz des Content Management Systems. Eine spezifische Form einer solchen Anwendung ist der Internet-Shop. Aufgrund ihrer vielen Vorteile werden Weblogs sowie verschiedene Intranets und Extranets immer beliebter. Alle gehören zu den speziellen Arten von Webanwendungen.

Bei der Programmierung aller Webanwendungen sind die folgenden Aspekte zu berücksichtigen:



Sicherheit - ist eine Priorität für jede Webanwendung, da immer die Gefahr besteht, dass Daten verloren gehen oder zerstört werden. Es besteht auch die Gefahr, dass die Daten oder Informationen von einer unbefugten Person gestohlen werden oder dass der Webserver mittels einer Anwendung gehackt wird (Gefahr des Verlustes des Firmenimages).

Nutzung der verfügbaren Ressourcen - bei der Programmierung jeder Webanwendung versuchen wir, fertige Codestücke aus anderen Ressourcen zu verwenden; zu diesem Zweck besitzen wir ein umfangreiches Skriptarchiv. Auf diese Weise kann etwas Arbeit, die für die Entwicklung der Anwendung notwendig ist, und damit auch Geld und die gesamte Zeit für die Realisierung des Auftrags eingespart werden.

Skalierbarkeit - wenn sich herausstellt, dass die Webanwendung in der Praxis gut funktioniert, beginnt sie in der Regel modifiziert, verbessert und erweitert zu werden. Werden diese Änderungen bei der Anwendungsgestaltung berücksichtigt, ist die Einbindung viel einfacher und kostengünstiger. Aus diesem Grund werden die komplexesten Webanwendungen modular angegangen.

Geschwindigkeit - langsame Webanwendungen sind nicht sehr anwendbar, sie sind auch ein Problem für Explorer. Daher versuchen wir, alle Skripte so zu optimieren, dass sie schnell sind, und deshalb wird auch empfohlen, eine fertige Anwendung auf unseren Servern zu installieren. Da auch die neuesten Entwicklungstools installiert sind, die auf vielen Servern für kommerzielles Webhosting fehlen, ist die Entwicklung der Webanwendung auch schneller und kostengünstiger.

Die maximale Belastung der Webanwendung ist ein Begriff, der mit hohem Traffic verbunden ist. Wenn ein Web von einer höheren Anzahl von Besuchern besucht wird (z.B. wenn Sie erwarten, dass ein Produkt präsentiert wird), ist der Server oft nicht in der Lage, sie zu bedienen (wir sagen, dass der Server "abgestürzt" ist). Die Fähigkeit einer Webanwendung, der hohen Belastung standzuhalten, erfordert die Auswahl geeigneter Tools, Datenbankoptimierung, Berechnungen und andere spezielle Techniken wie das Pre-Caching. Aufgrund der Erfahrung können wir sagen, dass der geringe Lastwiderstand ein

Beweis für die geringe Leistungsfähigkeit der Programme ist, die die Webanwendung erstellt haben.

Testen - Vor dem Start einer Webanwendung sollen alle ihre Funktionen auf einem Entwicklungsserver getestet werden. Dieser hat die gleiche Konfiguration wie der eigentliche Server, was eine Reduzierung möglicher Probleme während der eigentlichen Inbetriebnahme ermöglicht.

4.3. Technologie unserer Webanwendungen

Die derzeit am weitesten verbreitete Skriptsprache für die Programmierung von Webanwendungen ist PHP und die Datenbank MySQL. Diese Kombination mit einem Webserver (Programm) Apache wird Triade genannt. Es hat sich als nützlich für seine Flexibilität erwiesen. Weitere Vorteile dieses Systems sind die Zugänglichkeit von Funktionen und Codefragmenten sowie die kontinuierliche Weiterentwicklung dieser Programme.

Wenn es Gründe für eine Webanwendung oder eine Vereinfachung ihrer Entwicklung gibt, werden andere Programmiersprachen verwendet, z.B. Perl, Python oder die Datenbank PostgreSQL. Jeder von ihnen ist für die spezifischen Anforderungen einer bestimmten Webanwendung geeignet.

Der größte Vorteil aller genannten Technologien ist ihre Einbindung in Open Source. Es bedeutet, dass sie frei sind (was einer der wesentlichen Faktoren für ihre Popularität und Verbreitung ist).

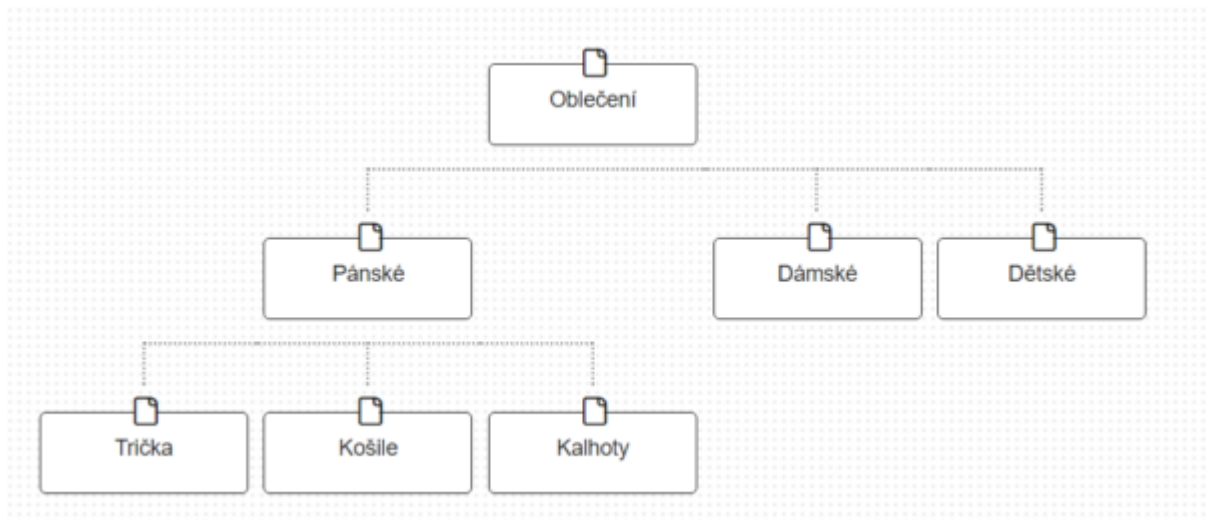
4.4. Was ist eine Webarchitektur?

Die Informations(inhalt)webarchitektur ist eine Methode, um die Webinformationen in ein logisches Ganzes zu bringen.

Es ist klar, dass die Informationen nicht auf der gleichen Seite sein werden, aber das Web wird mehr Seiten enthalten, die in verschiedenen Ebenen angeordnet sind, von der allgemeinen bis zu den Details.

Beispiel: Eine Web-Selling-Kleidung ist wie folgt angeordnet: Männer Kleidung > Hemden > Kurzarm. Die Art und Weise, wie die Informationen innerhalb eines Webs angeordnet sind, wird als Webarchitektur bezeichnet.

Weitere Informationen finden Sie in der folgenden Abbildung:



Legende: oblečení - Kleidung, Pánské - Männer, dámské - Frauen, dětské - Kinder, trička - T-Shirts, Košile - Hemden, Kalhoty - Hosen

Warum man sich mit Webarchitektur beschäftigen sollte

Eine gute Webarchitektur ist aus den folgenden Gründen wichtig:

- Es ermöglicht einem Benutzer, sein Ziel (den gewünschten Webinhalt) zu erreichen.
- Es schafft eine Logik des gesamten Webs, die sowohl von Nutzern als auch von Forschern und Suchmaschinen wahrgenommen und genutzt wird.
- Es ermöglicht die Anordnung der Webinhalte so, dass sie in Suchmaschinen leicht zu finden sind.
- Bei der Vorbereitung eines neuen Webs oder E-Shops wird die inhaltliche (Informations-)Architektur eines Webs innerhalb des grundlegenden Webdesigns behandelt (vor dem Grafikdesign und der Realisierung des Webs - siehe 4 Schritte eines professionellen Webdesigns).
- Eine gute Webarchitektur ermöglicht es, alle wichtigen Themen richtig anzuordnen. Dies spiegelt sich in einer einfacheren Orientierung sowie in einer besseren Nachvollziehbarkeit durch gezielte Keywords in Suchmaschinen wider. Wie bereits bekannt ist, ist der Traffic von Suchmaschinen kostenlos und hoch relevant.

4.4.I. Wie man eine perfekte Architektur schafft

Innerhalb von MD Webdesign wird die Webarchitektur auf Basis der Keyword-Analyse erstellt. Das bedeutet, herauszufinden, nach welchen Keywords die Nutzer in den Suchmaschinen von Google und Seznam suchen, sie zu analysieren und dann zu kategorisieren, um einen besseren Einblick zu erhalten.

Wenn wir durch ein bestimmtes Keyword nachvollziehbar sein wollen, ist es empfehlenswert, eine Webseite zum Thema Keyword und Keyword selbst zu haben.

Die Keyword-Analyse ermöglicht es, herauszufinden, woran die Nutzer der Suchmaschinen interessiert sind. Die Einbindung dieser Themen (Keywords) in die Webarchitektur erhöht die Chancen auf bessere Positionen bei der Suche erheblich.

Beim Aufbau der Informationsarchitektur wird auch der Inhalt der einzelnen Seiten beschrieben. Die Gestaltung der Webinhalte erfolgt auf der Grundlage der Analyse von Kunden / Klienten in Form einer sogenannten Person (weitere Informationen zu Personen siehe Wikipedia). Dadurch ist es möglich, ein Web zu entwerfen, das den Erwartungen der konkreten Kunden entspricht. Und wir sind in der Lage, die benötigten Informationen bereitzustellen.

Erstellung von Webseiten: beginnend mit der Informationsarchitektur

Informationsarchitektur ist ein Plan zum Sortieren von Informationen im Web. Für ein gutes Funktionieren der Webseiten muss der Schaffung einer klaren Informationsarchitektur große Aufmerksamkeit geschenkt werden.

Als Architekt, der an einem Hausdesign arbeitet, arbeitet er mit Raum, Licht und Formen, während der Informationsarchitekt mit Informationen, Strukturen und Prioritäten arbeitet.

Generell muss eine gut gestaltete Informationsarchitektur auch in der Textform verständlich sein. Die Verwendung von Farben oder Pfeilen hilft nicht, wenn die Struktur der Informationsseite nicht verständlich ist.

Die Hauptgründe für die Verwirrung des Webs sind: die verwendete Terminologie, eine Vereinbarung, die Fachkenntnisse erfordert, eine unlogische Anordnung von Informationen.

Um die oben genannten Probleme zu vermeiden, muss das Design der Informationsarchitektur die folgenden Schritte umfassen:

- Identifizierung der Zielgruppe
- Sammeln von Informationen
- Gruppierung von Informationen
- Priorisierung
- Erstellung einer Informationsarchitektur

Die einzelnen Schritte:

1. Identifizierung einer Zielgruppe

Die Zielgruppe wird vor dem Entwurf der Informationsarchitektur identifiziert, da sich die Sichtweise auf die veröffentlichten Informationen dramatisch ändert.

Im Falle von Websites der öffentlichen Verwaltung ist der Zielnutzer fast jeder; daher wird es schwierig sein, das Wissen oder die Gewohnheiten zu definieren, die die meisten Benutzer teilen werden. Wir können jedoch davon ausgehen, welches Wissen die Benutzer NICHT haben werden.

Die Praxis zeigt, dass die Nutzer, die von Städten und Gemeinden aus auf die Websites zugreifen, in der Regel den Unterschied zwischen den Begriffen "Stadt" und "Behörde" nicht kennen. Sie kennen die Terminologie nicht und es ist für sie schwierig, die Organisationsstruktur einer Behörde zu verstehen. Die Situation ist noch schlimmer, da die gleiche Agenda in der Regel von verschiedenen Abteilungen in verschiedenen Städten verwaltet wird.

Daraus folgt, dass die interne Struktur einer Behörde kein gutes Beispiel für die Schaffung einer der Öffentlichkeit präsentierte Informationsarchitektur ist.

2. Sammeln von Informationen

Der zweite Schritt bei der Gestaltung der Informationsstruktur besteht darin, festzulegen, nach welchen Informationen der Besucher im Web suchen wird. Behörden sind verpflichtet, bestimmte Informationen zu veröffentlichen, aber es sind nicht alle Informationen, die die Website enthalten soll.

Die Webseiten der öffentlichen Verwaltung enthalten in der Regel eine Vielzahl von Informationen, die von Sitzungsprotokollen, Verordnungen, Budgets, Beschlüssen, Vorschriften bis hin zu aktuellen Ereignissen in der Gemeinde und einer Liste von Kultur- oder Sportveranstaltungen reichen.

Ein Teil dieses Schrittes ist auch eine Entscheidung darüber, welche Informationen NICHT im Web veröffentlicht werden. Es ist immer zu überlegen, welche Informationen der Web-Betreiber regelmäßig aktualisieren kann. Die Probleme mit veralteten Informationen können beim Design der Informationsarchitektur vermieden werden.

3. Gruppierung von Informationen

Wenn wir wissen, welche Informationen das Web präsentieren soll, werden thematische Einheiten geschaffen. Diese Einheiten müssen alle im vorherigen Schritt gesammelten Informationen enthalten, möglichst ohne Überschneidungen.

Zu diesem Zeitpunkt muss mindestens ein Vertreter der Öffentlichkeit in den Prozess einbezogen werden. Als Büroangestellter sind Sie sich der Struktur des Büros und des Managementprozesses bewusst. Für die Öffentlichkeit ist das anders; es ist nicht praktikabel, von einem Besucher zu erwarten, dass er sich der inneren Struktur des Büros in einem Web bewusst ist.

Der Besucher geht "von oben" durch das Web: Er geht von einer allgemeinen Ebene aus und spezifiziert seine Anfrage schrittweise mit weiteren Klicks, bis er die gewünschten Informationen findet. Beim Entwurf der Informationsarchitektur erfolgt der Prozess jedoch "von unten" - Informationen werden sortiert und in Gruppen und Einheiten eingeteilt. Diese Diskrepanz kann für einen Besucher irreführend sein, da er die Webinhalte nicht kennt und nicht weiß, ob die Informationen im Web vorhanden sind. Daher ist es unerlässlich, die Verständlichkeit der Informationsstruktur "von oben" zu testen.

4. Priorisierung

Bisher genügte der gesunde Menschenverstand und das Testen mit den Anwendern. Die Priorisierung im Design der Informationsarchitektur ist jedoch mit der Analyse verbunden. Daher ist es notwendig, über konkrete Zahlen zu verfügen, z.B. die Analyse der meistgesuchten Begriffe im Web. Die Prioritäten können sich auch im Laufe der Zeit ändern, z.B. im Sommer, es gibt häufige Anfragen zu Schwimmbädern etc., während im Winter die Nutzer z.B. an den Schneeverhältnissen interessiert sind.

Es gibt viele Methoden der Priorisierung: z.B. das Platzieren der Informationen oben in der Struktur, das Platzieren auf der Titelseite oder der Hauptseite der thematischen Einheit, das Hervorheben durch Kontrast, Farben, Größe, etc.

Beispielsweise sollen Seiten, die sich auf Siedlungsabfälle konzentrieren, Informationen über den aktuellen Preis und die Fälligkeit der Gebühr enthalten, ohne den Besucher zu zwingen, die pdf-Dateien mit einem Hinweis zu lesen. Da es sich um

häufig gesuchte Informationen handelt, können sie auch mit einer größeren Schriftart hervorgehoben werden.

5. Erstellen einer Informationsarchitektur

Beim Aufbau einer Informationsarchitektur kann ein geeignetes Werkzeug verwendet werden, aber ein Werkzeug selbst reicht nicht aus. Die Informationsarchitektur ähnelt einem Baum.

5.PHP – BASICS

PHP ist eine Programmiersprache, die auf der Seite des Servers arbeitet. PHP ermöglicht das Speichern und Ändern von Website-Daten. Die ursprüngliche Bedeutung war Personal Home Page. Es wurde 1996 gegründet und hat viele Veränderungen erfahren. Heutzutage steht die Abkürzung für Hypertext Preprocessor.

PHP-Sprache

PHP ist eine der am häufigsten verwendeten Programmiersprachen für die Erstellung von Webanwendungen. PHP ist eine Server- und Serverseite zum Erzeugen eines HTML/XHTML-Seitencodes, der später an den Explorer gesendet wird (im Gegensatz zu clientseitigem JavaScript, das erst nach der Anzeige im Explorer funktioniert).

Der Hauptvorteil von PHP ist seine Unabhängigkeit von der Plattform (Windows, Linux, Unix...). Weitere Vorteile sind ein breites Anwendungsspektrum. PHP kann mit Dateien und vielen verschiedenen Datenbanken arbeiten, es kann Grafiken generieren und bearbeiten, Mails senden und empfangen, PDF erstellen, alle wichtigen Internetprotokolle unterstützen.

Da PHP eine relativ freie Syntax hat (wie es geschrieben ist), ist es leicht zu erlernen, besonders wenn Sie Erfahrung mit anderen Programmiersprachen haben. Zusammen mit dem Webserver Apache und der Datenbank MySQL bildet sie die so genannte Triade, drei Programme, die am häufigsten für die Erstellung von Webseiten verwendet werden. Dies bringt einen weiteren Vorteil von PHP - eine große Anzahl von Fragmenten, benutzerdefinierten Funktionen und vorgefertigten Lösungen für häufige Probleme im Internet.

Möglichkeiten von PHP

PHP ist nicht schwer zu verstehen, und es genügt, die Grundlagen zu kennen. Es kann Daten speichern, ändern und löschen. Alles wird innerhalb eines Webserver erledigt (wo es Quellcodes von Webseiten gibt). PHP-Skript wird zuerst auf dem Server ausgeführt und sendet nur das Ergebnis an den Explorer (d.h. es berechnet zuerst 300/30 und sendet dann die Nummer 10 an den Explorer). Daher gibt es im Quellcode nur "10" (im Gegensatz zu JavaScript, das direkt im Explorer berechnet). Im Gegensatz zu JavaScript und HTML wird der PHP-Quellcode im Explorer nicht angezeigt.

PHP kann für die Erstellung eines Diskussionsforums, Gästebuchs, Counters, Meinungsumfrage, Grafik verwendet werden; mit einem einfachen Code ist es möglich, den gesamten Inhalt des Webs zu verwerten. Darüber hinaus ist es möglich, die Seiten mit Datenbanken, z.B. MySQL, zu verbinden.

Zweck von PHP

Es gibt mindestens eine Funktion, die jedes Web nutzen kann. Auf Webseiten werden bestimmte Teile oft wiederholt: Kopf mit Links, Menü, Fußzeile. Mit PHP ist es einfach, eine Vorlage für ein Web zu erstellen, in die die Dateien mit Menü und Fußzeilen eingefügt werden können. Auf diese Weise ist es möglich, das Menü nur einmal zu schreiben und in andere Seiten zu kopieren. Und das Wechseln des Menüs ist sehr einfach (siehe PHP-Menü).

PHP-Dateien

Webseiten mit PHP-Elementen haben meist die Erweiterung.php, aber es kann auch andere Erweiterungen geben, z.B..phtml, php3, php4, php5. Einige Hostings haben bestimmt, welche Version des PHP-Skripts basierend auf der Erweiterung gestartet werden soll (aktuelle Version ist 7). Dies ist jedoch ein Ausnahmefall; meistens genügt .php.

Installation

PHP ist eine Sprache, die nicht nur mit einer bestimmten Version eines Explorers funktioniert (im Gegensatz zu HTML oder JavaScript). Es muss auf dem Computer installiert sein. Grundlage sind ein Webserver und Bibliotheken. Um PHP zu unterstützen, ist es notwendig, es zu installieren und einen Server (meist Apache) zu konfigurieren. Es ist empfehlenswert, PHP Tread für die Installation des PHP-Programms zu verwenden.

Webhosting mit PHP

Nicht jedes Webhosting beinhaltet PHP-Unterstützung. Der Support für Webhosting ist ein überdurchschnittlicher Service gegen Aufpreis. Es ist jedoch möglich, kostenloses Webhosting mit PHP-Unterstützung zu erhalten (z.B. Webzdarma.cz, PHP 5). Bei der Auswahl von Webhosting für PHP-Seiten ist es notwendig, sorgfältig zu lesen, was das Angebot enthält.

PHP - Grundlegende Informationen

Dynamische Seiten (d.h. Seiten, die zuerst von einem Server generiert und dann an den Client gesendet werden) sind heutzutage ein wesentlicher Bestandteil jeder komplexeren Website. Die wichtigsten Skriptsprachen für die Erstellung solcher Seiten sind ASP (Active Server Pages) und PHP. Im folgenden Teil werden wir uns mit der Einführung in PHP beschäftigen.

5.1. Was ist PHP?

PHP ist eine serverseitige Skriptsprache, die in einen gemeinsamen HTML-Code eingefügt wird. Was bedeutet das? Jede Seite, die PHP-Skripte enthält, wird vom Server übernommen und alle Befehle auf der Seite in PHP aufgelistet, dann wird ein sauberer HTML-Code an den Client gesendet (was ein Ergebnis des Skripts ist). Theoretisch kann der Server in allen gesendeten Dateien nach PHP-Skripten suchen, aber meistens ist er so konfiguriert, dass er nur in den Dateien mit den Erweiterungen.php, .php3 oder .phtml nach ihnen sucht. PHP-Befehle werden direkt in den HTML-Code eingefügt und durch die Tags `<?>` (oder `<?php` und `?>`) getrennt.

Wofür ist PHP gut?

PHP ist eine sehr vielseitige Sprache, in der es relativ einfach ist, z.B. einen Newsserver oder einen virtuellen Shop zu programmieren. Die Daten können in gängigen Textdateien oder in einer Datenbank gespeichert werden (PHP ist mit fast allen gängigen Datenbanken, z.B. MySQL, kompatibel). Die Verarbeitung der Daten aus den Formularen ist sehr einfach, verschiedene einfache Online-Tests einschließlich der Erfolgsquote der Besucher können einfach erstellt werden; auch ein hochwertiges Werbesystem kann einfach programmiert werden. Die Stärke von PHP zeigt sich in der Verwendung auf den Servern Email.cz, Centrum.cz oder BillBoard.cz.

Was ist notwendig, um in PHP zu arbeiten?

PHP wird in HTML eingefügt, so dass jeder gängige HTML-Editor für die Erstellung von PHP-Skripten verwendet werden kann. Es ist sogar genug, um Notepad zu benutzen. Das Wichtigste ist, die erstellten Skripte platzieren und testen zu können. Dazu muss auf dem Server PHP-Unterstützung installiert sein (PHP, jetzt in Version 4, kann kostenlos unter www.php.net heruntergeladen werden). PHP ist als Modul des Linux-Systems am effizientesten, kann aber auch unter Windows verwendet werden (diese Informationen benötigen Sie nur, wenn Sie der Administrator Ihrer Webseite sind). Wenn Sie eines der kostenlosen Webhosting-Angebote nutzen, ist die PHP-Unterstützung nicht sehr wahrscheinlich, aber z.B. der Server www.kgb.cz bietet kostenloses Webhosting mit PHP-Unterstützung (mit einigen Einschränkungen, z.B. ist es nicht möglich, Datenbanken zu verwenden, etc.). Im Falle von Paid Webhosting gibt der Anbieter Auskunft über PHP - entweder kostenlos

oder gegen eine bestimmte Gebühr. Der Anbieter soll auch sagen können, welche Erweiterung für die Dateien mit PHP-Skripten notwendig ist (wie bereits erwähnt, meist sind dies .php, .php3 oder .phtml).

Erstes Skript in PHP

Hier ist das erste PHP-Skript, das die aktuelle Zeit schreibt:

```
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=windows-1250">
    <title>PHP - Beispiel 1</title>
  </head>

  <body bgcolor="#FFFFFF" text="#000000">
    <center><font face="Arial CE, Arial" size="5">
      Aktuelle Zeit: <?php echo Datum ("H:i:s"); ?>
    </font></center></center>
  </body>
</html>
```

Anweisungen: Speichern Sie im Server z.B. als Datei example1.php und betrachten Sie sie in einem Explorer. Damit ein Skript funktioniert, muss es zuerst von einem Server interpretiert werden, d.h. es konnte nicht offline von einer Festplatte aus untersucht werden.

Wie funktioniert das Skript?

Es ist zu beachten, dass es sich im Grunde genommen um eine klassische HTML-Seite handelt, die auch einen PHP-Befehl enthält - `echo Date ("H:i:s")`, der vom restlichen HTML durch die Tags `<?php` und `?>` getrennt ist. Der Server nimmt zuerst die erforderliche Datei example1.php, und da seine Erweiterung .php ist, lässt er sie durch den PHP-Interpreter laufen und führt alle Befehle aus - diese werden zwischen den Tags `<?php` und `?>` gesucht. Es stößt auf den Befehl `echo`, dessen Server zum Schreiben in die resultierende Datei dienen. Die Funktion `Date` gibt das Datum und die Uhrzeit zurück, in den Klammern gibt es die Parameter, die das Format bestimmen, in dem die Uhrzeit angezeigt werden soll (Stunden, Minuten und Sekunden getrennt durch Doppelpunkte - Funktion `Date` wird später behandelt). Das folgende Semikolon dient zur Trennung der einzelnen Befehle. Es ist in diesem Fall redundant, da es nur einen Befehl gibt, aber es wird empfohlen, sich an das Schreiben zu gewöhnen. Anstelle von PHP-Befehlen schreibt der PHP-

Interpreter die aktuelle Zeit und der Server sendet die Seite dann an den Besucher. Wenn Sie die Quelle des Skriptergebnisses anzeigen möchten, sehen Sie nur ein reines HTML, nicht PHP. Dies führt dazu, dass im Gegensatz zu clientseitigen Sprachen (JavaScript) niemand an Ihren Code herankommt und er daher nicht kopiert werden kann.

Zusammenfassung

- PHP-Befehle werden in einen gemeinsamen HTML-Code eingefügt, sie werden durch die Tags `<?php` und `?>` (oder nur `<?` und `?>`) getrennt.
- Um nur nach PHP-Befehlen zu suchen, muss die Datei die richtige Erweiterung haben - normalerweise `.php`, `.php3` oder `.phtml`.
- Der Besucher erhält nur einen reinen HTML-Code vom Server (d.h. den Code, nachdem alle PHP-Befehle ausgeführt wurden).

6. VERARBEITUNG VON HTTP-REQUESTS

6.1. Anfragemethoden

GET

Es ist die am weitesten verbreitete Methode. Es dient zum Abrufen eines Objekts (HTML-Datei, Bild,...) vom Server. Die Antwort ist "cacheable". Die GET-Anforderung wird daher von einer Reihe von Überschriften begleitet, in denen sie angegeben ist, wie alt das Dokument ist, ob es geändert wurde usw. Die GET-Anfrage hat in der Regel keinen Body.

POST

Mit dieser Methode ist es möglich, die Informationen vom Benutzer innerhalb eines Körpers an den Server zu liefern (POST wird oft zum Senden größerer Daten aus Webformularen, zum Hochladen von Dateien usw. verwendet).

HEAD

Ist ähnlich wie GET, aber der Körper wird in der Antwort nicht übertragen. Diese Anforderung kann z.B. genutzt werden, um herauszufinden, ob das Objekt wirklich existiert (zur Steuerung der Links innerhalb einer Seite).

PUT/DELETE

Erstellt/löscht das angegebene Objekt vom Server. Diese Methoden werden in der Praxis nicht sehr häufig eingesetzt.

OPTIONS

Es wird verwendet, um Informationen über einen bestimmten Kontext (oder "*" für den gesamten Server) zu erhalten. Der Client kann herausfinden, welche Anfragen er an den jeweiligen Kontext senden kann.

OPTIONEN * HTTP/1.1

Host: www.root.cz

Beispiel für eine implizite Einstellung des Servers

TRACE

Es wird zur Verfolgung des gesamten Anforderungspfades verwendet. Im Body einer Antwort erhält der Client die Anforderungen der einzelnen Systeme, über die die Anforderung ging. Diese Methode wird von Administratoren und Webprogrammen verwendet, die herausfinden möchten, warum der Server ein abgelaufenes Dokument zurückgibt, etc.

Headers

Das Protokoll HTTP Version 1.1 definiert eine große Anzahl von Headern für Anfragen und Antworten. Es gibt einige von ihnen.

6.1.1. Request Headers

Accept*

Header dieses Typs geben an, was der Client verarbeiten kann. Der Server wählt dann die am besten geeignete Alternative aus. Zu den Überschriften gehören Accept (MIME-Dokumenttypen, Accept-Charset (Zeichensatz, sehr wichtig in der tschechischen Umgebung), Accept-Encoding (Codierung der übertragenen Daten, wird meist zur Auswahl einer Kompression verwendet) und Accept-Language (Sprache des Dokuments).

Connection

Im Protokoll HTTP 1.1 ist der Parameter "close" definiert, der ein sofortiges Schließen der Verbindung nach der Übertragung des ersten angeforderten Dokuments erfordert.

Referer

Mit diesem Header gibt der Client die URL der Seite bekannt, von der aus der Link generiert wurde.

Host

Mit HTTP 1.1 wird die Unterstützung von sogenannten namensbasierten virtuellen Servern eingeführt. Diese Methode ermöglicht es, mehr virtuelle Server von einer IP-Adresse aus zu betreiben, aber der Client muss den Namen des Servers angeben, mit dem er kommunizieren möchte.

User-Agent

Dieser Header wird verwendet, um das Kundenprogramm zu identifizieren, entweder für statistische Zwecke oder um verschiedenen Forschern unterschiedliche Inhalte zur Verfügung zu stellen usw.

6.1.2. Response-Header

Content*

Überschriften, die den Inhalt (Körper) der Antwort beschreiben. Es kann z.B. die Länge des Inhalts, seinen MD5-Digest (Content-MD5), die Sprache (Content-Language), den Typ des Dokuments (Content-Type) und andere Attribute enthalten. Es ist zu beachten, dass diese Überschriften nicht nur in den Antworten verwendet werden. Wenn ein Body auch die Anforderung enthält (z.B. bei der POST-Methode), müssen diese ebenfalls verwendet werden.

Server

Dieser Header wird verwendet, um den Server zu identifizieren (normalerweise gibt es seinen Namen, seine Version und manchmal auch andere Informationen).

Verfällt

Ein Server kann diese Informationen verwenden, um den Ablauf eines Dokuments anzuzeigen. Nach dieser Zeit sollte der Client eine neue Version herunterladen.

Es gibt eine Reihe weiterer Header, die z.B. für die Steuerung des Dokumenten-Downloads verwendet werden können ("Download nur, wenn das Dokument seit... geändert wurde") oder um dem Server einen Benutzernamen und ein Passwort für den Zugriff auf die nicht öffentlichen Teile des Servers zur Verfügung zu stellen. Ebenso kann ein Server seine Antwort genauer beschreiben und den Client informieren, wann das Dokument zum letzten Mal geändert wurde oder ob sein Caching in öffentlichen oder privaten Caches erlaubt ist.

6.2. Grundlegende Merkmale des HTTP-Protokolls

Um die Informationen vollständig zu verstehen, ist es notwendig, die grundlegenden Merkmale des HTTP-Protokolls und seine Funktionsweise zu kennen. Das HTTP-Protokoll ist ein Protokoll auf Anwendungsebene für verteilte hypermediale Informationssysteme. In der Praxis bedeutet dies, dass dieses Protokoll im Allgemeinen im Internet nicht nur zur Datenübertragung zwischen einem Client und einem Server, sondern auch für viele andere Zwecke verwendet wird. Das HTTP-Protokoll ist zustandslos, d.h. es erkennt nicht die Clients, von denen Anfragen gestellt werden. Wenn ein Client eine Anfrage sendet und dann eine andere sendet, erkennt der Server nicht, dass es sich um den gleichen Client handelt.

HTTP existiert in 3 Versionen - 0.9, 1.0 und 1.1 . Das erste von ihnen, das als HTTP/0.9 bezeichnet wird, existierte als einfaches Protokoll, das Daten im Internet nur eingeschränkt übertragen konnte. Die Version HTTP/1.0 ermöglichte die Übertragung von Daten im MIME-Format, so dass sie auch Metainformationen über die übertragenen Daten enthalten konnte. Die wichtigste Verbesserung gegenüber der Version HTTP/1.1, die auch die neueste Version ist, war, dass alle Verbindungen dauerhaft wurden, was bedeutet, dass die Verbindung geschlossen wird, wenn entweder der Client oder der Server einen Header zum Schließen sendet. HTTP verwendet, um die Verbindung nach jeder Antwort vom Server zu schließen. Diese Verbesserung beschleunigte die Übertragung erheblich, da der Server nicht für jedes Bild, jeden Frame und jedes Applet eine neue Verbindung öffnen muss.

6.2.1. HTTP-Protokoll Anforderungsformat

Die Anforderung des HTTP-Protokolls hat folgendes Format:

```
METHODE URL DES HTTP-DOKUMENTS  
HEADER  
Leerzeile  
ANDERE DATEN nur bei der Methode POST
```

Die Anforderungsmethode gibt an, wie der Server die Anforderung verarbeiten soll. Die Methoden werden später behandelt. Header werden im folgenden Format gesendet:

```
NAME DES HEADERS: HEADER-WERT
```

Jeder Header muss in einer separaten Zeile stehen. Alle Zeilen müssen mit den Tags CRLF (`\r\n`) abgeschlossen werden. Am Ende aller Überschriften muss eine Leerzeile stehen, auch wenn keine weiteren Daten vorhanden sind.

Anfragen in PHP werden über sogenannte Sockets gesendet. Ein Socket ist im Grunde genommen eine Verbindung zwischen einem Client und einem Server. Um mit ihnen zu arbeiten, ist es notwendig, zuerst das Socket zu öffnen. Zum Öffnen wird die Funktion *fsockopen* verwendet:

```
fsockopen(server, port);
```

Beispiel:

```
$sock = fsockopen("www.interval.cz", 80);
```

Bei geöffneter Socket ist es möglich, die Anforderung über die Funktion *fputs* zu senden:

```
fputs(socket, request);
```

Beispiel:

```
fputs($sock, "GET /index.html HTTP/1.1\r\nHost: www.inter-  
val.cz\r\n\r\n")
```

Im folgenden Teil werden wir uns mit HTTP-Methoden befassen.

7. HTTP-PROTOKOLL-ANFORDERUNGSMETHODEN

In HTTP/1.1 gibt es sieben grundlegende Methoden von HTTP-Requests:

- GET
- POST
- HEAD
- OPTIONS
- PUT
- DELETE
- TRACE

Nach jeder Anfrage können einzelne Header folgen. In der Version HTTP 1.1 ist es in jeder Anfrage zwingend erforderlich, den Header Host zu verwenden, der den Host angibt. Nach den Überschriften muss eine Leerzeile folgen, wie bereits erwähnt.

7.1. GET-Methode

Die GET-Methode ist die einfachste und eine der grundlegenden Methoden. Jedes Mal, wenn eine Seite von einem Server geladen wird, ohne zuerst ein Formular mit der POST-Methode zu senden, wird diese Methode verwendet, um die Seite vom Server abzurufen. Das Ergebnis ist also eine Seite und ihre Überschriften, nach denen wir bei dieser Methode fragen. Das Format dieser Methode ist wie folgt:

LIEFERT URL DER SEITE DER PROTOKOLLVERSION
HEADERS
Leerzeile

Beispiel:

GET /index.asp HTTP/1.1.1
Host: www.interval.cz
Leerzeile

7.2. POST-Methode

Die POST-Methode arbeitet wie die GET-Methode, aber im Falle der POST-Methode ist es möglich, Daten an das Skript nach den Überschriften und der Leerzeile zu senden. Diese Methode wird zum Senden der Daten aus einem Formular mit der POST-Methode verwendet. Das Format dieser Anfrage ist wie folgt:

```
POST URL- DER SEITE DER PROTOKOLLVERSION
HEADERS
Leerzeile
DATEN AUS DEM FORMULAR
```

Beispiel:

```
POST /processdata.php HTTP/1.1
Host: www.formulare.cz
Inhaltslänge: 29
Content-Type: application/x-www-form-urlencoded
Leerzeile
```

```
array1=value1&array2=value2=value2
```

Bei dieser Methode gibt es Mroe Header, die noch nicht erwähnt wurden. Content-Length gibt die Länge der Daten aus dem Formular (in Bytes) an, und Content-Type: application/x-www-form-urlencoded bezeichnet den MIME-Typ der Daten aus dem Formular.

HEAD, OPTIONEN, PUT, DELETE, TRACE

Wenn Sie keinen Internet Explorer programmieren, werden Sie wahrscheinlich nicht auf diese Methoden stoßen. Die folgende Tabelle zeigt daher nur die grundlegende Bedeutung dieser Methoden.

HEAD	Es funktioniert als GET, aber es gibt nicht den Body der Seite zurück, sondern nur den Header. Dies wird z.B. verwendet, um festzustellen, ob sich die Seite gegenüber dem letzten Aufruf geändert hat.
OPTIONS	Es wird für Anfragen nach Servermöglichkeiten verwendet.
PUT	Es wird als GET bezeichnet, behält aber den Körper der Anforderung an einer Stelle, die durch die erforderliche URL vorgegeben ist. Es ist vergleichbar mit dem Senden von Dateien per FTP.
DELETE	Es entfernt das Dokument vom Server. Das zu entfernende Dokument erhält die URL der Anfrage.

TRACE	Es wird verwendet, um die Anfrage über alle Proxy-Server und Firewalls zu verfolgen, die die Anfrage durchläuft. Es ist vergleichbar mit dem Tool TraceRoute.
--------------	---

7.3. Verwendung von Datenquellen

Moderne und effiziente Datenverarbeitungsanwendungen speichern Informationen nicht direkt in ihren eigenen Dateien, sondern nutzen einige der externen Datenquellen. Es gibt eine Reihe von Datenquellen, angefangen von einer einfachen Textdatei über Dateidatenbanken z.B. im DBF-Format bis hin zu Datenbanken, die auf SQL-Servern gespeichert sind. Jede dieser Quellen verwendet in der Regel ein eigenes Format für die Speicherung von Daten und Methoden für den Abruf der gespeicherten Daten. Um das Problem mit der Vielfalt der Datenquellen zu vermeiden und die Notwendigkeit zu vermeiden, Anwendungen für jede dieser Quellen zu entwickeln, haben die Anwendungsprogrammierer ein Standardwerkzeug entwickelt, das den Zugriff auf verschiedene Datenquellen über eine standardisierte Plattform - ODBC - ermöglicht.

Eine solche standardisierte Plattform ermöglicht es den Nutzern, die Daten aus verschiedenen Quellen an verschiedenen Standorten komplex und einfach zu nutzen. So ist es beispielsweise möglich, die Daten aus Access in Word oder Excel zu verwenden, ohne sie exportieren und zuerst in die gewünschte Anwendung laden zu müssen.

7.3.1. Was ist ODBC?

ODBC steht für Open Database Connectivity. ODBC-Datenquellen sind für Anwendungen über einen relevanten Treiber zugänglich, der als Vermittler für die Kommunikation zwischen einer user's Anwendung und einer externen Datenquelle wahrgenommen werden kann. Die Anwendung sendet die Anfrage nach den Daten an ODBC. Der jeweilige Fahrer übersetzt die Anforderung so, dass die externe Datenquelle sie versteht und an die Datenquelle sendet. Die Antwort der Datenquelle wird ebenfalls über ODBC gesendet, das das Ergebnis in das Standardformular übersetzt und an die Anwendung zurückgibt.

Das Prinzip von ODBC ist Standard; daher kann jede Anwendung, die den ODBC-Treiber verwenden kann, auf jede externe Datenquelle von jedem Hersteller zugreifen, der den entsprechenden Treiber bereitstellt. ODBC ermöglichte es so, den Zugriff auf die Daten für die Anwendungen zu standardisieren, ohne zu lösen, wie die Datenquelle funktioniert.

Einerseits muss die Anwendung nur wissen, wie man mit ODBC arbeitet. Andererseits stellen die Hersteller verschiedener Datenquellen die relevanten ODBC-Treiber zur Verfügung, so dass ihre Quellen für die Anwendungen leicht zugänglich sind. Dies ist sowohl für die Anwendungsprogrammierer, die einen unabhängigen, standardisierten Zugriff auf Daten haben, als auch für die Benutzer, die die Anwendungen erhalten, die kostengünstiger und schneller auf verschiedene Daten zugreifen können.

Arbeiten mit ODBC unter Windows

ODBC-Setup-Steuererelemente sind bereits ein integraler Bestandteil des Betriebssystems Microsoft Windows. Der ODBC-Datenquellenadministrator befindet sich im Bedienfeld. Dabei ist zu beachten, dass die Verfahren und Bilder im Betriebssystem Windows 10 gültig sind, aber auch in anderen Versionen von Windows. In einigen Betriebssystemen kann der Zugriff auf den ODBC-Administrator jedoch auf eine andere Weise erfolgen.

8. DATENQUELLEN

Um die Mailinglisten und Vorlagen zur richtigen Zeit mit den richtigen Daten zu füllen, verwendet Mailkit XML- und RSS-Datenquellen. Ihre Nutzung ist je nach Art des Benutzerkontos begrenzt. Die Version Mailkit Base ist auf die Verwendung von XML-Datenquellen beschränkt, die nur für den Import der Empfänger in Listen bestimmt sind, während Mailkit Syndicate und Agency sowohl XML- als auch RSS-Datenquellen unterstützen, auch für das Laden von Daten in Vorlagen.

Verwendung von XML-Datenquellen für die Empfängerliste

Um eine neue Empfängerliste aus einer Datenquelle zu erstellen, ist es notwendig, eine neue XML-Datenquelle einzurichten.

- Name - Name der Datenquelle. Wenn die Datenquelle für die Empfängerlisten funktioniert, hat die Liste den gleichen Namen.
- Beschreibung - Beschreibung der Datenquelle.
- Quelle - URL-Adresse des XML oder RSS, das als Datenquelle dient.
- Autorisierung - es ist angekreuzt, wenn der Zugriff auf den Speicherort der Datenquelle passwortgeschützt ist.
- Typ - Sie können zwischen RSS- oder XML-Datenquelle wählen.
- Ziel - das Ziel der Datenquelle wird aus der Auswahlliste ausgewählt. Es wird entweder für die Liste der Empfänger oder für die Vorlage verwendet.
- Automatisches Update - wenn diese Option aktiviert ist, wird die Quelle automatisch aktualisiert, bevor die Kampagne gesendet wird.
- Verfallszeit - legt die Zeit fest, nach der die Datenquelle im Falle einer automatischen Aktualisierung aktualisiert werden soll.
- Letztes Update - es zeigt das Datum des letzten Updates der Datenquelle an.
- Leere Datensätze:
- Zurücksetzen - die Empfängerdaten werden entsprechend den leeren Datensätzen im Import gelöscht.
- Aktuellen Wert beibehalten - die Empfängerdaten bleiben in der gleichen Form wie vor dem Import.

Wie man die Datenquelle vorbereitet

Die Datenquelle hat keine definierte Struktur und kann in den Formaten XML oder JSON vorliegen, die voll gültig sein müssen. Die Datenquelldatei muss an eine URL gesendet werden, die von den Mailkit-Servern verfügbar ist, und gegen den Zugriff Dritter geschützt sein, da es sich um sensible Informationen handelt.

Da jeder Kunde ein anderes Informationssystem mit unterschiedlichen Möglichkeiten verwendet, ist das Datenquellen-System universell und setzt die Struktur der benötigten Daten nicht spezifisch fest. Es gibt jedoch einige technische Einschränkungen der Datenquellen:

- Vollständig valides XML oder JSON
- Attribute werden im XML-Format nicht unterstützt (z.B. Vorname="Jana" gender="f" country="cz").
- UTF8-Zeichenkodierung empfohlen
- Verwenden Sie "," nicht, um mehr Werte zu trennen - verwenden Sie "|".
- Der eindeutige Datensatzbezeichner und das einzige obligatorische Feld ist die E-Mail-Adresse. Wenn es mehr Datensätze mit derselben E-Mail-Adresse gibt, werden diese überschrieben.

Wie bereits erwähnt, sind nur E-Mails obligatorisch; andere Daten sind nicht obligatorisch, aber wichtig. Die allgemeine Regel ist "je mehr, desto besser", aber auch nichts darf übertrieben werden. Die Datenquelle ist wichtig für das Maximum der verfügbaren Empfängerdaten, die für bestehende und zukünftige E-Mail-Kampagnen verwendet werden können.

Import von Daten aus der Datenquelle

Um Inhalte zu erhalten, müssen die Quellzweige den Kontaktfeldern zugeordnet werden. Um eine Zuordnung vorzunehmen, klicken Sie auf den Namen der Datenquelle und klicken Sie auf die Schaltfläche View Structure. Nachdem Sie alle Felder zugewiesen haben, klicken Sie auf Speichern. Anschließend ist es möglich, mit Import für den aktuellen Datenimport fortzufahren. In diesem Moment wird eine neue Empfängerliste erstellt (sie hat den gleichen Namen wie die Quelle, die sie erstellt hat), und die Daten aus der Datenquelle werden entsprechend der vorherigen Zuordnung importiert.

Gleichzeitig haben die Benutzer des Syndikats- und Agenturkontos die Möglichkeit der automatischen Aktualisierung. Wenn die automatische Aktualisierung gewählt wird, wird die Datenquelle automatisch importiert und die Empfängerliste wird vor Beginn jeder Kampagnenauslieferung aktualisiert. Dieser Parameter wird für Datenquellen mit mehr als tausend Datensätzen nicht empfohlen, da die Aktualisierung einige Minuten dauern kann, was die Auslieferung der Kampagne verzögert. Für größere Datenquellen wird empfohlen, die Möglichkeit der regelmäßigen geplanten Aktualisierung durch den Kundensupport für eine bestimmte Tageszeit zu nutzen.

Verwendung von XML- und RSS-Datenquellen in Vorlagen

Die Einrichtung von XML- und RSS-Datenquellen für die Verwendung in Vorlagen ähnelt der Verwendung für Empfängerlisten, ohne jedoch jedem Feld die Bedeutung zuweisen zu müssen. Die Werte werden durch eine Reihe von Namen in der Vorlage bestimmt, daher ist es einfach, jede XML- oder RSS-Quelle einzurichten.

Oben gibt es ein Beispiel für einen Template Code, für den die RSS-Datenquelle EXAMPLE verwendet wird. Der Befehl FOREACH erstellt eine Schleife zum Parsen und Suchen aller Datensätze. Jeder der Standard-RSS-Tags ist einfach zu lösen und in den HTML-Code einzufügen, so dass die Daten in der Vorlage ausgegeben werden können. Weitere Informationen finden Sie unter E-Mail-Vorlagen.

Produktdatenquellen

Datenquellen können auch für die Übermittlung des Produktangebots an Mailkit und für die spätere Verwendung von Produktinformationen in Kampagnen genutzt werden. Hier zeigt sich die Leistungsfähigkeit von Datenquellen und programmierbaren Vorlagen, die es ermöglichen, Daten aus mehreren Quellen zu kombinieren und die Inhalte für die einzelnen Empfänger automatisch zu personalisieren.

Für Produktinformationen ist es möglich, eines der gängigen Produkt-Feed-Formate für Heureka, Zbozi, Google, etc. zu verwenden oder den eigenen Feed mit den notwendigen Informationen zu generieren. Da die Produktfeeds sehr umfangreich sind und die Geschwindigkeit der Arbeit mit den enthaltenen Daten wichtig ist, werden diese Datenquellen direkt in die SQL-Datenbanken übertragen, und es ist weiterhin möglich, mit ihnen zu arbeiten. Für die Einstellung der Produktdatenquelle wenden Sie sich bitte an den Kundensupport, der Ihnen bei der Implementierung und Nutzung hilft.

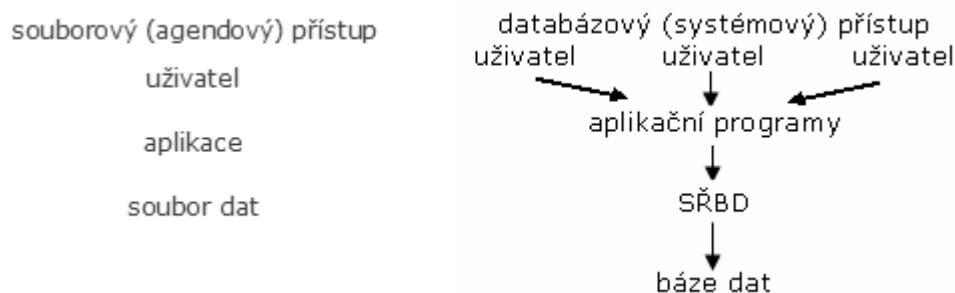
Zustell-Feeds

Delivery Feeds sind spezielle Datenquellen, die zur Weitergabe strukturierter Informationen für die Realisierung der Kampagnenauslieferung verwendet werden. Während eine Kampagne in der Regel die voreingestellte Empfängerliste verwendet, die für die Auslieferung der Kampagne nach den festgelegten Regeln verwendet wird, wird die Kampagne im Falle des Lieferfeed nur an die im Feed aufgeführten Adressen gesendet. Es ist eine Alternative zum API-Aufruf mailkit.sendmail_mass, d.h. der Weg, um beim Versenden der Kampagne an Mailkit hochstrukturierte Daten zu verarbeiten, z.B. aus Personalisierungssystemen oder CRM. Diese Feeds müssen eine streng definierte Struktur im XML-Format haben.

9. DATENBASIS ANSATZ

Anforderungen an das Datenbanksystem:

- Überprüfung der Datenkonsistenz - Die Datenbank muss in der Lage sein, die Einhaltung bestimmter Regeln der so genannten Integritätsbeschränkungen zu gewährleisten und die Daten vor möglichen Unfällen, die bei Transaktionen auftreten können, zu schützen.
- Transaktion - Reihenfolge der Manipulation mit Daten, die stattfinden muss, um sicherzustellen, dass die Daten ordnungsgemäß gespeichert werden, z.B. Übertragung von einem Konto auf ein anderes Konto (dies muss auf beiden Konten korrekt erfolgen).
- Große Datenmengen - im Vergleich zu den Möglichkeiten von Speichermedien muss die Datenbank in der Lage sein, ausreichende Datenmengen zu speichern.
- Datenmanagement - Entwicklungsstufen



Legende: datová hierarchie - Datenhierarchie, databáze - Datenbank, soubor - Datei, věta, Datensatz - Anweisung, Datensatz, Attribut, Attribut, Pol - Attribut, Array

Design einer strukturierten Datenbank

- Die Realität, deren Reflexion die entworfene Datenbank sein soll. Es besteht aus mehreren Objekten (Entitäten).
- Es können unterschiedliche Beziehungen zwischen den überwachten Einheiten bestehen (z.B. zwischen den Einheiten des gleichen Typs = rekursive Beziehung).

Kardinalität der Beziehung: symbolische Bezeichnung 1:1 (Professor XY´s Frau ist ZN); 1:N (Professor XY Lehrer Schüler); M:N (welche Schüler von welchen Professoren unterrichtet werden)

Integritätsbeschränkungen der Datenbank - alle Regeln, die die zulässigen Werte (eine Kombination von Werten) der Attribute definieren, Anzeigeformat

Relationales Datenmodell

Es wird davon ausgegangen, dass es einwertige Attribute gibt.

- Darstellung in Form einer relationalen Tabelle, in der ein Tupel einer Zeile entspricht und ein Attribut einer Spalte entspricht.
- Relationale Datenbank
- alle Daten liegen in Form von 1 oder mehreren Tabellen vor, die nach Spalten benannt sind.
- jede Spalte enthält Daten aus 1 Domäne (d.h. 1 Datentyp)
- die Elemente der einzelnen Spalten (mit Name und Typ) werden in der Regel als Elemente oder Arrays bezeichnet, und die Begriffszeile entspricht dem Datensatz (Statement).
- Im Sinne eines relationalen Modells beschreiben Sitzungen beide Einheiten und die Beziehungen zwischen ihnen.
- Basierend auf dieser dualen Nutzung ist es möglich, zwischen sogenannten Entity-Sessions (Sätze von arrangierten Tupelattributen, die die Entitäten beschreiben) und Relationship-Sessions (Sätze von arrangierten Tupeln) zu unterscheiden.
- Data Warehouses
- Warehousesprinzip - 2 Hauptziele
- Die Vereinheitlichung der Datenansicht in den einzelnen sogenannten Produktionssystemen bietet einen klaren Zugriff auf die Daten - unterschiedlich aufgerufen, aber die gleichen Objekte werden als ein Objekt betrachtet.
- unterschiedlich gemessen, aber die gleichen Variablen werden mit dem gleichen Messgerät gemessen.

10. STRUKTURIERTE UND UNSTRUKTURIERTE DATEN

10.1. Strukturierte Daten

Grundtypen (Klassifizierung zur Unterscheidung von erlaubten und unzulässigen Manipulationen und Werten)

- Text (Zeichenketten, die Informationen mittels eines Textcodes ausdrücken, bestimmte Mengen von Elementen, die aufgezeichnet werden können oder die eine Syntax definieren können)
- numerisch - reale rationale Zahlen
- Datum, Uhrzeit - Grenzwerte, die es erreichen kann (30. Februar, 27:00 Uhr)
- logisch - erfüllt die Bedingungen für das Vorhandensein oder Nichtvorhandensein eines Objekteigenschaften - 2 Werte (0 und 1, A und N)
- Kategorie - Wert der Merkmale, die aus einer Skala ausgewählt werden (oft wählt man, ermöglicht die Aufzeichnung der Werte nur mit Hilfe eines Codes).
- Strukturierung schafft eine Datenorganisation, die eine effiziente Speicherung, Verarbeitung und Abruf von Daten nach Bedarf ermöglicht → Strukturierte Daten erstellen Suchschlüssel (manchmal auch als Identifikationsschlüssel bezeichnet) - Schlüssel, die Datensätze eindeutig identifizieren, werden als private Schlüssel (Identifikationsschlüssel) bezeichnet - eine Grundvoraussetzung für Daten und Datenbanksystem.
- Informationen über etwas (Widerspiegelung der Realität) - Vor- und Nachname, Adresse, Alter, Telefonnummer, Gewicht, Preis,.... Punkteanzahl, Kategorie, Durchschnittsnote,... Stückzahl, Seitenzahl
- Operationen oder was getan werden kann - Addition, Rundung, Multiplikation, Verbindung (Name+Nachname), Verkürzung, Sortierung,....., Tag in einer Woche, Negation,
- Datentyp (muss definiert werden) - Anzahl, Text, Datum und Uhrzeit, logische Daten (ja/nein)

- Verschlüsselte Daten - verschiedene Kodierungen - Text, Buchstaben - verschiedene Codetabellen (ascii, ebdc,...) nationales Alphabet; Datum und Uhrzeit (wie wir ein Datum schreiben)

10.2. Unstrukturierte Daten

Datentyp: Freitext, Audio, Video, Grafik, Multimedia, etc.

- Bieten Sie mehr Informationen als strukturierte Daten.
- Problem: Es ist sehr schwierig, mit unstrukturierten Daten zu suchen. Lösung: Unstrukturierte Daten werden durch strukturierte Daten ergänzt (mp3-Titel).

Datenmengen - strukturierte und unstrukturierte Daten

- ascii Textseite (Notepad) 1,8 kB
- Worttext Seite 50 kB
- Vektorgrafik A4 30 kB
- Bitmap-Bild A4(jpg, rgb) 5 MB
- 1-minütige Audioaufnahme (WAV) 10 MB
- 90-minütige Videoaufzeichnung 3 GB

Data Warehouse (DWH) ist eine spezielle Form der relationalen Datenbank, die es ermöglicht, Aufgaben zu lösen, die sich hauptsächlich auf die analytische Abfrage über große Datensätze konzentrieren.

Fachliche Orientierung

Im Falle einer gemeinsamen relationalen Datenbank ist es üblich, die Redundanz der gespeicherten Daten zu minimieren, die durch ihre Normalisierung in die dritte Normalform und die interne Verknüpfung der einzelnen logischen Funktionseinheiten erreicht wird. Im Data Warehouse wird immer auf eine klare interne Trennung der einzelnen Funktionseinheiten geachtet. Das Ergebnis ist eine klarere Struktur für einen Benutzer (Manager, Business Analyst) auf Kosten der erhöhten Anforderungen an einen Speicherplatz.

Integration

Eine gemeinsame operative Anwendung (ein Programm) über eine relationale Datenbank löst eine bestimmte spezifische Art von Aufgaben über "ihre" spezifischen Daten. In einem Data Warehouse ist es notwendig, die Informationen aus vielen verschiedenen Quellen zu sammeln und nicht nach ihrer Herkunft, sondern nach ihrer logischen Bedeutung zu gruppieren (sie sind eng mit der Themenorientierung verbunden - alle Daten zu einem bestimmten Funktionsbereich müssen "auf einem Stapel" liegen, unabhängig davon, woher sie kommen).

Geringe Variabilität

Die Daten werden in der Regel in größeren Chargen (z.B. in Wochen- oder Tagesintervallen) in ein Lager hochgeladen und nicht nachträglich verändert.

Historisierung

Die Daten in einem Data Warehouse werden in der Regel in ihrer historischen Form gespeichert, nicht nur im aktuellen Zustand. Dies ist auf die Notwendigkeit zurückzuführen, Analysen durchzuführen, die sich auf die Entwicklung im Zeitablauf konzentrieren. Aus der Sicht von users´ ist in einer gemeinsamen relationalen Datenbank nur der aktuelle Zustand der Datenobjekte interessant.

10.2.1. Technologische Merkmale von Data Warehouses

Die Anforderungen an Data Warehouses implizieren ihre technologischen Eigenschaften:

- Ein Data Warehouse muss ein Werkzeug zum Hochladen von Daten aus verschiedenen Datenquellen enthalten. Diese Quellen können unterschiedliche Datenformate und unterschiedliche physische Standorte haben. Es müssen nicht nur relationale Datenbanken sein.
- Ein Data Warehouse speichert Daten nicht unter Berücksichtigung der besten Bearbeitungsbedingungen, sondern im Hinblick auf die beste und schnellste Ausführung komplexer Anfragen. Daher wird für die Datenspeicherung häufig die OLAP-Technologie eingesetzt.
- Es kann nicht gesagt werden, welche Anforderungen und Aufgaben die Benutzer in Zukunft haben werden (zum Zeitpunkt der Erstellung eines Data Warehouse kennen wir nur die Art der Aufgaben, nicht die einzelnen Aufgaben und Anforderungen).

10.2.2. Logische Struktur des Data Warehouse

Aus logischer Sicht (user´s) sind die Daten in einem Data Warehouse in Diagramme unterteilt. Jedes Diagramm entspricht einem analysierten Funktionsbereich.

Der Kern jedes Diagramms besteht aus einer oder mehreren Faktentabellen. Sie speichern die analysierten Daten - numerische und finanzielle Werte, die für analytische Berechnungen verwendet werden - Aggregation, Sortierung, etc. Der meiste Speicherplatz in einem Data Warehouse wird durch Faktentabellen belegt, die detaillierte Informationen aus allen Quellen enthalten, d.h. mehr als andere Tabellen.

Faktentabellen werden über Fremdschlüssel mit Dimensionen verknüpft. Dimensionen sind Tabellen, die Listen von Werten enthalten, die zur Kategorisierung und Sortierung von Daten in Faktentabellen verwendet werden.

Beispiel

Es ist notwendig, Informationen über alle Verkäufe aus den Registrierkassen von Hypermärkten in einem Data Warehouse zu speichern. Die Daten werden anhand des Verkaufszeitpunkts, des Geschäfts, der Warenart, des Lieferanten, der laufenden Marketingveranstaltungen und der Zahlungsmethode (mit Karte, in bar) weiter analysiert.

Das Verkaufsdiagramm enthält eine Faktentabelle - Verkaufsartikel, in der Informationen über die Art der verkauften Ware, den Preis und die Stückzahl (oder die verkaufte Menge) für jeden verkauften Artikel gespeichert werden.

Neben dieser Faktentabelle enthält das Diagramm auch Dimensionen für die Sortierung der Verkaufsartikel: Zeitdimensionen Datum und Stunde (innerhalb eines Tages), Dimensionen Shop, Warenart, bei der es für jeden Artikel eine Zeile gibt (z.B. Erdbeeryoghurt 250 ml), Warengruppe (Joghurt), Abteilung (Milchprodukte), Lieferant (Molkerei XX), etc.

Die Faktentabelle muss mit jeder dieser Dimensionen über einen Fremdschlüssel direkt oder indirekt verknüpft sein.

Es gibt zwei Möglichkeiten, hierarchische Dimensionen zu speichern:

- Aus der gesamten Hierarchie wird eine Dimensionstabelle erstellt, die die Daten für die höheren Hierarchieebenen redundant speichert. Auf diese Weise wird ein Diagramm erstellt, bei dem jede Dimensionstabelle direkt mit einer Faktentabelle verknüpft ist. Je nach Form wird dieses Diagramm als Sternschema bezeichnet.
- 3NF wird auf die hierarchische Dimension angewendet, so dass nur die Dimension auf der untersten Ebene der Hierarchie direkt mit der Faktentabelle verknüpft wird. Andere Dimensionen werden mit jeder der niedrigeren Dimensionen in der hierarchischen Struktur verknüpft. Dies wird als Snowflake-Schema bezeichnet.

II. AJAX

AJAX steht für Asynchronous JavaScript and XML. Es ist ein Oberbegriff für Technologien zur Entwicklung interaktiver Webanwendungen, die den Inhalt ihrer Seiten verändern, ohne sie vollständig hochladen zu müssen. Sie erfolgt durch asynchrone Verarbeitung von Webseiten mittels einer Bibliothek in JavaScript. Im Gegensatz zu klassischen Webanwendungen sind sie benutzerfreundlicher, erfordern aber die Verwendung moderner Webbrowser.

Diese Anwendungen werden mit den folgenden Technologien entwickelt:

- HTML (oder XHTML) und CSS zur Darstellung von Informationen;
- DOM und JavaScript zur Anzeige und dynamischen Änderung der präsentierten Informationen;
- XMLHttpRequest für den asynchronen Datenaustausch mit einem Webserver (typischerweise wird das XML-Format verwendet, aber es ist auch möglich, andere Formate wie HTML, Text, JSON oder EBML zu verwenden).
- Wie DHTML, LAMP oder SPA ist Ajax keine konkrete, unabhängige Technologie. Es ist ein Begriff, der sich auf mehrere Technologien und ein bestimmtes Ziel bezieht.

Vorteile

Zu den Vorteilen gehört der Wegfall der Notwendigkeit, die gesamte Seite in jedem Vorgang neu zu laden und zu zeichnen (im Gegensatz zum klassischen Modell der WWW-Seiten). Wenn ein Benutzer in einer Meinungsumfrage auf einen Vote-Button klickt, muss die gesamte Seite vom Server neu geladen werden, auch wenn nur die Vote-Ergebnisse aktualisiert werden und der Rest des Inhalts gleich bleibt. Mit AJAX wird die Abstimmung im Hintergrund gesendet, der Server sendet nur die Teile der Seiten, die sich geändert haben, und diese Teile werden aktualisiert und auf der Seite neu gezeichnet. Außerdem gibt es keinen unangenehmen Effekt, wenn nach der Aktivität ihre Blockelemente, Bilder usw. auf der kontinuierlich hochgeladenen Seite schrittweise angepasst und formatiert werden. Es kann auch ärgerlich sein, dass nach der spezifischen Aktivität, in der Mitte einer längeren Seite (nach unten gescrollt), die neu hochgeladene Seite nach oben gescrollt angezeigt wird. Mit AJAX arbeitet der Benutzer flüssiger und die Geschwindigkeit (insbesondere bei einer schnelleren Internetverbindung) ist fast gleich der Geschwindigkeit von gängigen Desktop-Anwendungen.

Dies hat auch das Potenzial, die Belastung der Webserver und des Web im Allgemeinen zu reduzieren. Da nicht für jeden Request das gesamte HTML-Dokument erstellt werden muss, sondern nur die vorgenommenen Änderungen markiert werden müssen, ist das Datenvolumen deutlich geringer und kann theoretisch auch die Auslastung der Datenbankserver oder anderer Backend-Systeme positiv beeinflussen.

Nachteile

Andererseits kann AJAX die Anzahl der ausgetauschten HTTP-Requests erhöhen, und obwohl sie ein geringeres Datenvolumen übertragen, sinkt die Last bei unsachgemäßer Implementierung nicht.

Zu den Nachteilen gehören vor allem die Änderungen im Paradigma der Webnutzung: Webseiten verhalten sich wie Anwendungen mit einer komplexen internen Logik, nicht wie eine Folge von Seiten, die auch mit den Schaltflächen Zurück und Weiter navigiert werden können. Ebenso ist es nicht möglich, die URL der Seite, auf der etwas "geklickt" wurde, mit der AJAX-Technologie zu übergeben. Moderne AJAX-Anwendungen sind in der Lage, das Browsen im Verlauf (zumindest teilweise) mit verschiedenen Techniken wiederherzustellen (z.B. mit einem Teil der Adresse nach dem # oder mit unsichtbaren IFRAMES). Dies macht jedoch das Design der Seiten komplizierter und zeitaufwendiger, sowie den Aufwand für die Implementierung mit AJAX.

Ein weiteres Problem von AJAX-Anwendungen kann die Netzwerklatenz sein: Der Bedarf an Internetkommunikation hat negative Auswirkungen auf die Antwortgeschwindigkeit und die Interaktivität der Benutzeroberfläche. Wenn der Benutzer nicht deutlich angekündigt wird, dass die Anwendung seine Anfrage bearbeitet (und mit dem Server im Hintergrund kommuniziert), registrieren sie lediglich eine verzögerte Antwort (die Benutzer können sogar versuchen, den Vorgang erneut zu starten, da sie denken, dass das System die Anfrage ignoriert hat, wodurch eine höhere Last für den Server erzeugt wird und / oder etwas verursacht wird, das sie nicht geplant haben, z.B. zehn Tickets statt zwei). Als geeignete Lösung empfiehlt es sich, irgendwie zu zeigen, dass die Anfrage user´s bearbeitet wird, z.B. mit Text oder animiertem Symbol.

Ein weiterer Nachteil von AJAX ist die Notwendigkeit, einen modernen Grafikbrowser zu verwenden, der die notwendigen Technologien unterstützt. Allerdings unterstützen alle derzeit verwendeten gängigen Browser diese Technologien zumindest grundsätzlich; das Problem liegt nur bei den Minderheitsbrowsern von Lynx oder bei hardware schwachen Browsern, z.B. bei einigen Handys und PDAs. Im Rahmen der Web-Zugänglichkeit ist es erforderlich, dass die Seiten auch ohne AJAX von Browsern aus zugänglich sind, was mehr Zeit und Arbeit für die Entwickler und höhere Kosten für die Käufer der Website bedeutet.

AJAX

Wie bereits erwähnt, steht AJAX für Asynchronous JavaScript and XML. AJAX ist eine moderne Technologie, die in den aktuellen Webanwendungen häufig verwendet wird. Es wird oft erwähnt, da es ein Teil von RIA ist, einem neuen Programmierstil, der zu höherem user´s Komfort und Funktionalität der Anwendungen führt.

AJAX ist keine neue Technologie, es ist nur eine Kombination aus bereits bekannten Technologien - HTML (oder XHTML), JavaScript, XML und XMLHttpRequest.

Warum ist AJAX so vorteilhaft? Die Anwendungen, die AJAX verwenden, können Daten von einem Server senden und abrufen, ohne die gesamte Seite neu laden zu müssen (im Gegensatz zu klassischen Links). AJAX kann daher für verschiedene Zwecke verwendet werden, z.B. für Autovervollständigungen (Formulare, die je nach gedrückter Taste automatisch ausgefüllt werden), AJAX Meinungsumfragen und andere komplexere Anwendungen, die die Arbeit eines Benutzers erleichtern können.

AJAX hat auch einige Nachteile, insbesondere bei unsachgemäßer Verwendung, es reduziert die Benutzerfreundlichkeit der Seiten erheblich. Wie bei anderen Technologien ist es daher notwendig, die AJAX-Anwendung sorgfältig zu planen und am Anwender zu testen.

Wie Sie bereits wissen, ist Javascript eine so genannte clientseitige Sprache, d.h. es wird auf der Seite des Webbrowsers client's durchgeführt und vom Javascript-Interpreter interpretiert. Es hat eine Vielzahl von Eigenschaften, die auch die Fähigkeit beinhalten, asynchrone Operationen / Aufgaben durchzuführen.

Worin besteht die AJAX-Asynchronie genau? Es ist die Fähigkeit von Javascript, eine Script- oder Element-API auf dem Server aufzurufen und nicht auf die Antwort zu warten. Stattdessen wird die Codeausführung fortgesetzt (z.B. kann der Benutzer die Seite sehen und mit ihr arbeiten). Wenn die Antwort kommt, wird die Ausführung des Code-Kontextes gestoppt (früher oder später, je nach Priorität der Aufgabe) und es gibt einen Rückruf (es handelt sich um eine Funktion, die ausgeführt wird, wenn der Server eine Antwort liefert). Ein solcher Code oder eine solche Funktion wird als "non-blocking" bezeichnet, da sie den Ablauf des Skripts nicht blockiert, indem sie auf die Antwort oder Verarbeitung wartet.

AJAX ist eine Methode zur Programmierung in Javascript. Es handelt sich nicht um eine neue Sprache oder ein neues Framework oder eine Bibliothek eines Drittanbieters. Es ist eine Methode, um die Daten in der Anwendung mit einer Datenbank, Server-Skripten (PHP, Java, ASP, etc.) auszutauschen, ohne die gesamte Seite aktualisieren / neu laden zu müssen. Technisch gesehen aktualisieren wir keinen Teil der Anwendung oder eine Seite (in Bezug auf die Aktualisierung) - das ist eigentlich die AJAX-Essenz.

Warum ist XML enthalten? Es ist eines der Datenformate, in denen AJAX die Informationen vom Server abrufen kann. Die bekanntesten und am häufigsten verwendeten sind JSON, XML, Text, Binärdaten. Jeder von ihnen kann auf eine andere Weise verwendet werden.

AJAX ermöglicht es, die vom Benutzer eingegebenen Daten zu überprüfen, noch bevor der Benutzer sie durch Bestätigen des Formulars sendet. Bis zu einem gewissen Grad kann dies auch per Javascript geschehen, aber bei einem Anmeldeformular mit bedingtem Passwort, Benutzername oder E-Mail, die innerhalb einer Tabelle oder Datenbank eindeutig sein müssen, sind dies genau die Situationen, für die AJAX die richtige Technologie ist. Es ermöglicht die gleichen Operationen, die auch durch den Aufruf von PHP-Skripten beim Umleiten nach dem Senden des Formulars ausgeführt werden können, aber im Gegensatz zu ihnen können Sie diesen Aufruf im Hintergrund ausführen.

AJAX Grundlagen

Die Grundidee von AJAX besteht darin, dass Teile einer Webseite asynchron hochgeladen werden, was den Inhalt der Seite verändert. Das bedeutet, dass eine Änderung des Inhalts nicht zum Hochladen der gesamten Seite führt, sondern nur des geänderten Inhalts. Zu diesem Zweck wird das Objekt XMLHttpRequest aus JavaScript aufgerufen. Diese Anforderung fragt den Server nach dem Inhalt, der typischerweise mit XML und JavaScript zurückgesetzt wird. Das Skript analysiert dann dieses XML. Die Daten aus dem XML-Skript übersetzen und ändern dann die Webseite mit DOM (Document Object Model) entsprechend.

Die Grundprinzipien der Arbeit mit AJAX werden an einem praktischen Beispiel beschrieben - einem Diskussionsforum, in dem die Diskussionen in HTML als Baum dargestellt werden. Wenn die Nutzer einen Beitrag lesen wollen, müssen sie auf dessen Überschrift klicken. Bei einem klassischen Modell wird die Anfrage an den Server gesendet und anschließend eine ganze Seite erstellt, die den Text des Beitrags und den gesamten "Baum" enthält. Im Falle des AJAX-Modells wird als Reaktion auf die Anfrage nur der Text des Beitrags gesendet. Das Hinzufügen des Textes zur ursprünglichen Webseite wird durch ein entsprechendes JavaScript-Programm sichergestellt.

Für ältere Internet Explorer-Versionen (ab Version 5.0) stand für diese Aufgabe ein ActiveX-Objekt zur Verfügung. Wie andere Entdecker stellt auch IEZ das entsprechende Objekt nativ zur Verfügung. Die aktuelle API XMLHttpRequest des Internet Explorers ist unter MSDN verfügbar; Mozilla-Projekt-API ist ebenfalls verfügbar.

AJAX's eigener Trick ist XMLHttpRequest - er kann asynchron verarbeitet werden, was in diesem Zusammenhang asynchron zur Anordnung des Restes der Seite sowie zu den laufenden Skripten bedeutet. Das bedeutet, dass die Daten über HTTP heruntergeladen werden können und dass die Daten heruntergeladen werden können, ohne die Interaktion des Benutzers mit der Seite zu beeinträchtigen. Die angeforderte Seite wird ebenfalls im Hintergrund erstellt.

Was ist AJAX? Einfach gesagt, ist AJAX eine Technologie, die Skripte verwendet, um die Kommunikation zwischen einer Seite und einem Webserver zu erleichtern. Auf diese Weise ist es möglich, den Seiteninhalt zu ändern, ohne ihn erneut hochladen zu müssen.

Theoretisch ist es möglich, dynamische Seiten ohne Serverskripte zu erstellen, d.h. eine Seite für das gesamte Web zu haben, die nur den Inhalt ändert. Dies ist jedoch nicht empfehlenswert - wie bereits gesagt wurde, handelt es sich um eine Skript-Technologie, so dass es vom Client abhängt und nicht gesichert werden kann, dass es funktioniert. Dennoch kann es als Add-on für Webseiten verwendet werden - z.B. Meinungsumfrage oder ein Suchassistent wie bei Seznam.cz.

Wenn Sie sicher sind, dass AJAX für die Besucher Ihrer Webseiten funktioniert, kann es auch für komplexere Anwendungen verwendet werden. Es kann z.B. für einen Chat verwendet werden, der nur neue Beiträge herunterlädt; wenn der Besucher allein ist, verlangsamt sich die Suche nach Updates, was viele übertragene Daten speichert.

12. FRAMEWORKS

Framework (Application Framework) ist eine Softwarestruktur, die als Unterstützung bei der Programmierung und Entwicklung und Organisation anderer Softwareprojekte dient. Es kann Rahmenprogramme, API-Bibliotheken, Unterstützung von Entwurfsmustern oder empfohlene Verfahren in der Entwicklung enthalten.

- Zweck
- Architektur
- Beispiele
- Ähnliche Artikel
- Externe Links

Zweck

Ziel eines Frameworks ist es, typische Probleme eines bestimmten Bereichs zu übernehmen und so die Entwicklung zu erleichtern, damit sich die Designer und Entwickler auf ihre Aufgaben konzentrieren können. So kann sich beispielsweise das Team, das Apache Struts für die Entwicklung von Webseiten für eine Bank einsetzt, auf den Bankbetrieb konzentrieren und nicht auch auf eine perfekte Navigation zwischen den einzelnen Seiten.

Es wird argumentiert, dass die Verwendung von Frameworks den Code langsam oder ineffizient macht und dass die durch die Verwendung eines fremden Codes eingesparte Zeit für das Studium des Frameworks aufgewendet werden muss. Durch die Wiederverwendung oder den Einsatz für ein Großprojekt wird jedoch eine erhebliche Zeitersparnis erzielt. Nach der Deinstallation des Frameworks können einige Anwendungen nicht mehr funktionieren.

Architektur

Frameworks bestehen aus sogenannten frozen spots und hot spots. Frozen Spots definieren die Gesamtarchitektur der Softwarestruktur, ihre Grundkomponenten und die Beziehungen zwischen ihnen. Diese Teile ändern sich in keiner Weise in der Rahmennutzung. Im Gegenteil, Hot Spots sind Komponenten, die zusammen mit dem Code programmer's eine ganz spezifische Funktionalität schaffen und daher jedes Mal anders sind.

In einer objektorientierten Umgebung besteht das Framework aus abstrakten und klasschen (nicht abstrakten) Klassen. Frozen Spots können durch abstrakte Klassen dargestellt werden und der Code (Hot Spots) selbst wird durch die Implementierung abstrakter Methoden hinzugefügt.

Framework ist eine Software-Struktur, die als Unterstützung bei der Programmierung und Entwicklung und Organisation anderer Softwareprojekte dient. Es kann unterstützende

Programme, API-Bibliotheken, Konstruktionsmuster oder empfohlene Verfahren in der Entwicklung beinhalten.

Ein weiterer sehr wichtiger Faktor bei der Wahl ist, wofür der Rahmen und sein Zweck benötigt werden. Im Allgemeinen können Frameworks in zwei Gruppen eingeteilt werden:

- Skriptsätze - Bibliotheken - für alle Arten von Bedürfnissen
- Skripte, die eine konkrete Webanwendung erstellen.

Vorteile

- Schnellere Entwicklung
- Weniger Code
- Universeller Code - Änderungen oder neue Funktionen werden viel einfacher hinzugefügt.
- Nette URL

Die Entwicklungsgeschwindigkeit ist höher. Framework erleichtert die Arbeit und vermeidet Programmerroutinen wie die Verbindung zu einer Datenbank, die Überprüfung der richtigen Option mit einem Abschnitt oder die Validierung eines Formulars. Cool-Url, XSS, separate Anwendungs- und Präsentationslogik, der Wechsel des DB-Servers von MySQL auf PostgreSQL wird ganz einfach sein.

Dank des Frameworks ist es möglich, sich nur auf die Entwicklung zu konzentrieren, aber es wird durch die Zeit eingelöst, die Sie benötigen, um zu lernen, mit Frameworks zu arbeiten. Wenn Sie lernen, etwas zu tun, werden Sie bald feststellen, dass es viel einfacher und schneller geht. Das Erlernen der Arbeit mit dem Framework kann eine Angelegenheit von einem Monat sein, aber auch eine Angelegenheit von mehreren Monaten oder einem Jahr. Schließlich werden Sie das Framework um Bibliotheken erweitern und die Programmierzeit deutlich verkürzen können. Framework bedeutet Zeit- und Kostenersparnis

Die Wahl eines Frameworks ist nicht einfach. Es wird empfohlen, mindestens zwei zu haben - eine für einfache und eine zweite für komplexere Aufgaben. Vor jedem Projekt muss entschieden werden, welches davon am besten geeignet ist. Natürlich ist das Beste, was man tun kann, diejenigen mit einer guten großen Gemeinschaft zu lernen, diejenigen, bei denen man sicher ist, dass ihre Entwicklung nicht in einer Woche abgeschlossen sein wird.

- **CakePHP:** ziemlich einfach zu erlernen; Qualitäts-Community; lange Entwicklung einer neuen Version
- **Zend Framework:** sehr effizient; deckt alle Anforderungen bei der Erstellung von Webanwendungen ab; große Community; anspruchsvoll; lange Titel;
- **Codelgniter:** klein, einfach; entwickelt von einer Firma, nicht von einer Community; besser als Cake in einigen Bereichen x "dumm" in anderen (es hat keine Layouts)

Monolithische Frameworks entkoppeln sich allmählich in einzelne Komponenten, was viele Vorteile mit sich bringt. War es früher schwierig und manchmal sogar unmöglich, nur eine Rahmenkomponente zu verwenden, ist es heute möglich, diese Komponente einfach zu installieren. Der Entwicklungszyklus der einzelnen Komponenten ist unterschiedlich. Sie können ihre eigenen Repositories haben, Issue Tracker, Entwicklungsteams.

Komponenten können keine neuen Versionen kontinuierlich aktualisiert werden, ohne auf eine andere Version des gesamten Frameworks zu warten. Es ist auch möglich, eine Komponente nicht zu aktualisieren, z.B. wegen eines BC-Bruchs.

Die Bedeutung des Begriffs "Framework" verschiebt sich also, und "Versionen" haben nun fast nicht mehr ihre ursprüngliche Bedeutung. Anstelle eines Frameworks XYZ, Version 2.3.1., wird ein Satz von Komponenten in verschiedenen Versionen verwendet, die zusammenwirken.

12.1. Was ist ein Framework?

Bei der Erstellung moderner Webanwendungen werden häufig Frameworks oder Entwicklungsrahmen verwendet. Es handelt sich um eine Reihe von Bibliotheken und Quellcode, die wiederholt verwendet werden können, um die Arbeit zu erleichtern, und deren Funktionalität einen Teil der erstellten Anwendung abdecken kann. Frameworks können Probleme lösen, die in irgendeiner Weise verallgemeinert werden können; sonst wäre es nicht sinnvoll, einen Rahmen zu schaffen. Dies ermöglicht es dem Anwendungsentwickler, sich auf die Funktionen zu konzentrieren, die für die Anwendung einzigartig und exklusiv sind und keine Routineprobleme lösen müssen.

Verwendung von Javascript-Frameworks

Frameworks können für die meisten Programmiersprachen erstellt werden. Javascript Framework ist somit ein Framework, das verwendet wird, um die Arbeit und Programmierung in Javascript zu erleichtern. Die Verwendung von Frameworks ist sehr umfangreich und jeden Tag ist das Portfolio der Probleme, die Javascript-Frameworks lösen können, größer. Generell kann man sagen, dass sie für das effektive Schreiben von Quellcode verwendet werden können; sie lösen weitgehend auch die Inkompatibilität zwischen Webbrowsern und einem geschriebenen Code, wodurch die Zeit auf programmer´s eingespart wird und es möglich wird, Elemente zu verwenden, die schwer zu programmieren wären. Ihre Stärke und Nutzung ist vor allem mit der Verwendung der AJAX-Technologie verbunden, da sie die Interaktion von Benutzer und Anwendung durch asynchrone Kommunikation zwischen Client und Server verbessern. Darüber hinaus ermöglichen sie den dynamischen und einfachen Zugriff und die Änderung der einzelnen Elemente einer Seite (DOM-Modell), GUI-Elemente, weisen Ereignisse zu und animieren sie. Last but not least

enthalten einige Frameworks vorgefertigte GUI-Komponenten, die entweder nicht mit einer anderen Technologie implementiert werden können oder zu schwierig und ineffizient sind. Diese Elemente sind einfach zu bearbeiten und in einem Framework mit nur mehreren Codezeilen zu implementieren.

13. DIE 10 BESTEN PHP-FRAMEWORKS FÜR ENTWICKLER

PHP, weltweit bekannt als die beliebteste serverseitige Skriptsprache, hat sich stark weiterentwickelt, da in statischen HTML-Dateien erste Inline-Codefragmente auftauchten.

Damals mussten Entwickler komplexe Webs und Webanwendungen erstellen; bei einer gewissen Komplexität war es zu zeitaufwändig und es erforderte zu viel Aufwand, immer bei Null anzufangen. Daher erschien die Notwendigkeit einer strukturierteren und natürlicheren Art der Entwicklung. PHP-Frameworks bieten Entwicklern eine adäquate Lösung für dieses Problem.

Warum man das PHP-Framework verwenden sollte

Zunächst werden wir uns mit den stärksten Gründen befassen, warum viele Entwickler PHP-Frameworks bevorzugen und wie diese den Entwicklungsprozess verbessern. Die Vorteile von PHP-Frameworks sind wie folgt:

- Ermöglicht eine schnelle Entwicklung
- Bereitstellung von gut organisiertem, wiederverwendbarem und wartbarem Code
- Ermöglicht Wachstum im Zeitablauf, da Framework-basierte Webanwendungen skalierbar sind.
- Keine Sorgen um Low-Level-Websicherheit
- Nach MVC (Model-View-Controller) Muster, das die Trennung von Darstellung und Logik gewährleistet.
- Förderung moderner Webentwicklungspraktiken einschließlich objektorientierter Programmierwerkzeuge

1. Larve

Obwohl Laravel ein relativ neues PHP-Framework ist (es wurde 2011 veröffentlicht), ist es laut der neuesten Online-Umfrage auf Sitepoint das beliebteste Framework bei Entwicklern. Laravel verfügt über ein riesiges Ökosystem mit einer Plattform, die sofort gehostet und bereitgestellt werden kann, und das offizielle Web bietet viele Tutorials in Form von Screencasts namens Laracasts.

2. Symfony

Die Komponenten des Frameworks Symfony 2 werden von vielen Projekten genutzt, wie z.B. dem System zur Verwaltung der Inhalte von Drupal oder der Software phpBB für laufende Foren. Es wird auch von Laravel verwendet. Symfony hat eine umfangreiche developers´ Community und viele Unterstützer.

3. Codelgniter

Codelgniter ist ein fast 10-jähriges, leichtes PHP-Framework (ursprünglich 2006 veröffentlicht). Codelgniter hat einen sehr einfachen Installationsprozess, der nur eine minimale Konfiguration erfordert. Es ist eine ideale Option, um Konflikte mit PHP-Versionen zu vermeiden, da es auf fast allen Shared und Dedicated Hosting Plattformen reibungslos funktioniert (derzeit benötigt es nur PHP 5.2.4).

4. Yii 2

Die Wahl von Yii-Frameworks gibt einer Seite einen Leistungsschub, da sie schneller ist als andere PHP-Frameworks. Es nutzt weitgehend die Technologie des "faulen Ladens". Yii 2 ist rein objektorientiert und basiert auf dem DRY (Don't Repeat Yourself) Codierungskonzept, so dass es eine klare und logische Codebasis bietet.

5. Phalcon

Das Phalcon-Framework wurde 2012 veröffentlicht und wurde bald bei den PHP-Entwicklern beliebt. Es soll so schnell wie ein Falke sein, wie es in C- und C++-Sprachen geschrieben wurde, um eine höchstmögliche Leistungsoptimierung zu erreichen. Für die Verwendung von Phalcon ist es nicht notwendig, die Sprache C zu lernen, da die Funktionalität als PHP-Klassen bereitgestellt wird, die in jeder Anwendung verwendet werden können.

6. CakePHP

CakePHP wird seit einem Jahrzehnt verwendet (die erste Version wurde 2005 veröffentlicht), aber es ist immer noch eines der beliebtesten PHP-Frameworks, da es immer bestrebt war, aktuell zu sein. Die neueste Version, CakePHP 3.0, bietet eine erweiterte Sitzungsverwaltung, verbesserte Modularität durch die Trennung mehrerer Komponenten und erhöhte die Möglichkeit, mehr autarke Bibliotheken zu erstellen.

7. Zend Framework

Zend ist ein robustes und stabiles PHP-Framework mit einer Reihe von Konfigurationsoptionen; daher wird es normalerweise nicht für kleinere Projekte empfohlen. Es ist jedoch großartig für die komplexeren. Zu den Zend-Partnern gehören z.B. IBM, Microsoft, Google und Adobe. Die kommende neue Version, Zend Framework 3, wird für PHP 7 optimiert, aber sie wird weiterhin PHP 5.5 unterstützen.

8. Schlank

Slim ist ein PHP-Mikroframework, das nur das bereitstellt, was benötigt wird. Das Design von Micro frameworks´ ist minimalistisch, sie sind ideal für kleinere Anwendungen, für die ein voll ausgestattetes Framework zu viel wäre. Der Slim Schöpfer wurde von einem Mikrorahmen Rubin namens Sinatra inspiriert.

9. FuelPHP

FuelPHP ist ein flexibles, voll funktionsfähiges PHP-Framework, das nicht nur MVC, sondern auch die entwickelte Version HMVC (Hierarchical Model-View-Controller) unterstützt. FuelPHP fügt eine optionale Klasse namens Presenter (früher ViewModel genannt) zwischen den Ebenen View und Controller hinzu, um die für die Erzeugung von Views erforderliche Logik einzubinden.

10. PHPixie

PHPixie ist ein brandneues Framework, das 2012 gestartet wurde, um ein leistungsfähiges Framework für schreibgeschützte Webs zu schaffen. Wie FuelPHP implementiert PHPixie das Entwurfsmuster HMC und wird mit Hilfe unabhängiger Komponenten erstellt, die auch ohne das Framework verwendet werden können. PHPixies-Komponenten werden zu 100% durch Unit-Tests getestet und erfordern nur ein Minimum an Abhängigkeiten.