

# Databasestøttet Webpublicering

ved

IT-højskolen i København

## Databasestøttet Webpublicering, forår 2002

### Forelæsning 1

- Generel Kursusinformation
- Ugentlige øvelser og eksamen
- Kursusindhold og motivation
- Web-publicering
- Web-baserede Services
- Udvikling af Statiske Sider
- Versionskontrol
- Detaljerede Kursusindhold
- HTML på 21 minutter
- Nogle Web-design Regler
- Eksempel på SQL
- CGI-scripts og Appletter m.m.
- Nogen om Programmeringssprog
- Eksempler på Tcl

### Generel Kursusinformation

- **Kursusmål:** Lær at anvende avancerede web-teknikker til at udvikle realistiske, brugbare web-sites, der involverer udveksling af information med en database.
- **Lærer:** Martin Elsmann(mael@it.edu, kontor 2.42)
- **Hjælpeværere:** Frank Bjerløge (bjergoe@it.edu)
- **Kursets hjemmeside** er <http://www.it.edu/courses/W2/F2002>
- **Kursusforløb,** se forelæsningsplanen: <http://www.it.edu/courses/W2/F2002>
- Hjemmesiden indeholder udlånte materialer, sidste nyt, ...
- Forelæsningsplanen indeholder forelæsningsplanerne
- Forelæsningsplanerne udkommer (senest) dagen før forelæsningen
- **I skal holde øje med hjemmesiden hele semestret igennem**
- **Hvornår:** forelæsning mandag formiddag. Øvelser mandag eftermiddag.
- Stil endelig spørgsmål under forelæsningerne

### I får brug for

- **Lærebog:** "Philip and Alex' Guide to Web Publishing", der enten skrives ud fra nettet (<http://www.arsdigita.com/books/panda/>), bestilles hos Amazon (<http://www.amazon.dk/>) eller købes i boghandel. Pris ca. 500 kroner.
- Noter, der udskrives fra hjemmesiden
  - *A Smooth Introduction to Tcl for Web Nerds*, Martin Elsmann.
  - *Tcl for Web Nerds*, Hal Abelson, Philip Greenspun, and Lydia Sandoz.
  - *SQL for Web Nerds. Tutorial for the SQL language*, links into online Oracle docs for completeness. Philip Greenspun.
  - *SlideExtractor*, the engineering behind a tool for extracting slides from WimpyPoint, a web-based slide-constructor. Niels Hallenberg.
- Relaterede Noter
  - *HTML overview*, Peter Sestoft.
  - *HTML-codes for unusual characters and symbols*, MITW3C.
  - *Four Part Article about AOLserver, tcl, and Oracle*. In *LinuxWorld*, Philip Greenspun.
  - og meget andet - se kursets hjemmeside.
- Konto på IT-højskolens maskiner og på kursets web-server.

## Ugentlige Øvelser

### Forudsætninger:

Grundlæggende *Web-design* eller tilsvarende forudsætninger. Derudover skal du være motiveret for at *lære at programmere*.

- Man lærer kun for alvor ved at prøve selv.
- Læsevejledning: Læs lærebog og noter forud for hver forelæsning. **Løs opgaver** i god tid. Læs også de nemme ekstra opgaver hvis nødvendigt!
- Øvelser: 4 timer pr. uge pr. studerende (startende i anden uge)
- Øvelserne foregår på IT-højskolen, Glentevej 67
- Øvelsesidspunkter: mandag 12.30–16.30.
- Ugentlige opgaver til elektronisk aflevering. Eksempel:
  - Opgaver stilles mandag på løbesejden
  - Opgaverne regnes ved øvelserne mandag eftermiddag
  - Besvarelserne afleveres senest ugen efter
  - Vi retter besvarelserne den efterfølgende uge

IT-højskolen

Databasestøttet Webpublicering, forår 2002

Side 1-5

## Kursusindhold og Motivation

Web-publicering kræver generalister:

tekstbehandling, fotografiering, publicering, system administration, databasekendskab, forståelse for brugergrænseflader, programmeringsferfaring, kendskab til designmetodikker, . . . .

Udgangspunkt: Statistiske sites (HTML)

Howdan kan man konstruere mere interessante sites:

- Sites som er programmer (dynamisk HTML)
    - Beregning af skat
    - Bill Gates personal wealth clock (<http://db.photo.net/wealthclock.html>)
  - Sites som er databaser (dynamisk HTML med databasetilgang)
    - IT-C's nyhedsfløjestre
    - CourseGrader (<http://hug.it.edu:8002/vu/index.tcl>)
    - Amazon (<http://www.amazon.com>)
- Populære og brugbare sites kan bygges med få midler!

IT-højskolen

Databasestøttet Webpublicering, forår 2002

Side 1-7

## Eksamen, Pensum og Øvelsesindhold

- Ud at semestrets 11 løbesejler vil de 8 bedste indgå med 50% af den samlede karakter.
- Skriftlig eksamen indgår med 50% af den samlede karakter.
- **Skriftlig eksamen og løbesejler skal beståes hver for sig for at bestå kurset**
- Løbesejlerne afleveres elektronisk - mere information herom på den første løbesejdel.
- Retlejerne gøres synlige i CourseGrader: <http://hug.itu.dk:8002/vu/index.tcl>
- Løbesejlerne løses individuelt:
  - Løbesejlerne dækker:
    - HTML
    - Sempel programmering med TCL
    - Sempel databaseprogrammering med SQL
    - Web-sites konstrueret med HTML, TCL og SQL
- Pensum vil være lærebog, og noter om TCL og SQL, løbesejler, ekstra opgaver samt forelæsningsplanerne. Det forventes, at man behersker HTML. Justering og præcisering af pensum kan ske gennem semestret, og den sidste løbesejdel vil indeholde en præcis pensumliste.

IT-højskolen

Databasestøttet Webpublicering, forår 2002

Side 1-6

## Web-publicering

Web-publicering er ofte kendetegnet ved et magasin agtigt indhold, f.eks. produktkataloger, avisfløjestre m.m.

Web-services er kendetegnet ved at løse et problem for en bruger, f.eks. huskeliste, kommentarservice, bogkøb, billetbestilling m.m.

Start med at spørge hvad en bruger kan være interesseret i!

Hvad kan du gøre for at berige en bruger?

- Undgå
- user-modeling, entry tunnel, exit tunnel
  - flashy langsom grafik
- Understøt flere syn på indholdet (eng. *Multiple Views*).
- Tilfød brugerkommentarer med f.eks. ArsDigita's gratis services.

IT-højskolen

Databasestøttet Webpublicering, forår 2002

Side 1-8

### Web-baserede Services

Fra desktop applikationer til web-baserede programmer

- WimpyPoint (<http://wimpy.arsdigita.com>)
- RemindMe
- Kalendersystem
- Tekstbehandling/versionskontrol
- CourseGrader
- Kursusevaluering
- Kursusbog / tilmeldingssystem

Føj i web-baserede programmer kræver og tillader rettelser hurtigt!

### Udvikling af Statistiske Web-sites

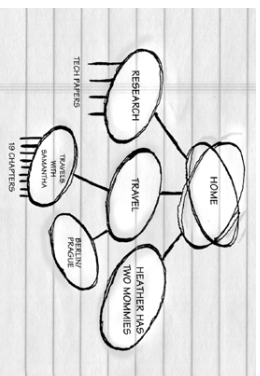
- Tegn en site-oversigt (eng. *site map*)
- Saml og strukturer indhold; giv navne til filer.

Overvej hvorledes sekundære filer såsom billeder m.m. skal organiseres.

- Konstruer et "kun-tekst" site
- Ansæt en grafisk designer!
- Konstruer en vedligeholdelsesplan. Hvorledes opdateres indhold?

Kan man uddelegere dette ansvar til dem, der skriver indholdet, eller skal man have en central hjemmesidedeling.

- Lav periodiske brugertests.



### Versionskontrol

Problem

- Joe henter Version A af et dokument kl. 09.00 fra web-sitet og bruger en dag på at editere det.
- Mary henter Version A kl. 12.00 og retter en typo. Mary skriver dokumentet tilbage til serveren kl. 12.10 (Version B).
- Joe skriver sit dokument tilbage til serveren kl. 17.00 (Version C).

Mary's rettelser er glemt! Version C indeholder ikke rettelser i Version B!

Løsninger: "låsnings" af filer med emails, Emacs, RCS, CVS, .....

CVS: <http://www.cvshome.org/>

### Detaljeret Kursusindhold

- HTML (Hyper Text Markup Language)
- TCL (Tool Command Language)
- SQL (Structured Query Language)

#### Hvorfor lære programmering?

- Bedre forståelse for muligheder og begrænsninger i udvikling af dynamiske web-sites
- Nødvendig betingelse for udvikling af avancerede web-sites
- Bedre muligheder for at forestille sig nye typer af web-sites

#### Hvorfor lære TCL?

- TCL er velegnet til udvikling af dynamiske web-sites (ligesom Java, ASP, PHP3, Standard ML, ...)
- TCL er et lille sprog med relativt få programkonstruktioner.
- TCL er godt til behandling af strenge (og derfor godt til web-programmering)
- `tclsh` gør det let at afprøve programmer
- TCL er fuldt integreret med en web-server (AOLserver)

## Specielle programmeringssprog: HTML og SQL

**HTML:** Beskriv hypertextens struktur og (til dels) layout

Eksempel (hello.html):

```
<html>
<head><title>
Hello
</title></head>
<body>
Hello
</body>
</html>
```

Browsere er normalt ikke pedantiske overfor korrekt HTML – men derfor skal I skrive korrekt HTML alligevel!

**SQL:** Sender data frem og tilbage mellem et program og en database.

```
select navn, adr from navne order by navn;
insert into navne (navn, adr) values ('Iris', 'Gallevej 435');
```

Der findes et utal af SQL-dialekter - men det er normalt trivielt at forstå forskellene.

## HTML på 21 minutter

Et lægalt HTML dokument (legalt.html):

```
<html>
<head>
<title>Mit Size</title>
</head>
<body bgcolor=white>
<h2>Mit Size</h2>
Se siden for
<a href="http://www.it-c.dk/courses/W2/E2001/">Database-baseret
Web-publicering</a>
</body>
</html>
```

Det kan være svært at sætte HTML kode prænt op – men prøv at være konsistent!

PT. skriver vi HTML i hånden – senere skal vi lave programmer som genererer HTML kode som resultat.

**Skriv HTML koden selv!**

## Generel Side Layout (lb1-ekstra.html)

Du bør kende følgende tags i HTML:

- overskrifter, dvs. <h1>, </h1>, ..., <h4>, </h4>.
- skillelinier såsom <hr>
- afsnit og ny linie såsom <p>, </p> og <br> og </br>
- quotes, <blockquote> og </blockquote>.
- centrerung <center> og </center>
- fed <b> og </b>
- kursiv <i> og </i>
- understregning <u> og </u>
- ordrede <ol>, </ol> og uordrede <ul>, </ul> lister samt <li>.

## Hyperreferencer (lb1-ekstra.html)

- hyperreferencer <a href="link">navn</a>
- lokale hyperreferencer med etiket <a name="etiket"> og reference til etiket <a href="index.html#etiket">navn</a>
- emailreferencer <a href="mailto:nh@it-c.dk">nh@it-c</a>
- tabeller med <table>, </table>, <tr>, </tr>, <th>, </th>, <td> og </td> samt de mest almindelige attributter
- billeder <img>
- farver f.eks. bgcolor

### Find fem fejl

i et HTML dokument uden konsistent indryk

```
<html><head><title>Hello</title></head>Hello<center>great page<p>
this is in <b>bold<i> and italics</html>
```

eller det samme dokument med indrykning

```
<html>
<head>
  <title>Hello</title>
</head>
Hello
<center>
  great page
<p>
  this is in <b>bold<i> and italics
</html>
```

Netcape viser dette dokument uden fejl Eller gør den??

### Simpelt er Godt

Problemer med fancy-tags, såsom FONT:

- Gamle browsere ignorerer dem
- Nye browsere ignorerer dem
- Når du udskifter den gratiske designer skal du ændre 10.000 htm1-sider!
- Istedet: brug cascading style sheets.

Modern-brugere kan ikke lide for megen grafik!

### Mere HTML dokumentation

Der er masser af henvisninger på kursets hjemmeside til små tekster om HTML:

- Peter Sestofts HTML-oversigt
- Den legendariske A Beginner's Guide to HTML
- HTML-koder for usædvanlige tegn og symboler
- Per B. Rasmussens Fravækoder i HTML

### Flere HTML Regler

- Minimer brug af JavaScript (understøttes ikke af de fleste browsere).
- Undgå store tabeller, til f.eks. side-layout (giver problemer med hastighed).
- Undgå redefinition af linkfarver (folk forbinder blå med et link).
- Sørg for at din side kan udskrives på printer (f.eks. udskriver Netscape hvid tekst på blå baggrund som hvid tekst på hvid baggrund!)
- Træk på din brugers skærm som en knap ressource.
- Undgå frames (virker dårligt sammen med email, printer og bogmærker).
- Husk at give dine HTML sider en titel (til bl.a. bogmærke).

En målsætning kunne være, at både Netscape 1.1 (eller 2.0) samt Explorer kan vise din side.

Hvad med WebTV, Palm Pilots, mobil telefoner, ...

### Flere Web-design Regler

- Informationen skal være brugergrænsefladen: ved klik på informativ ord, bringes man til uddybende information (undgå "her"-links)
- Giv brugere et bredt, flat overblik af information, istedet for et sekventielt informationsforløb
- Organisér dit site efter forventede brugerinteresser, istedet for efter den interne organisation.
- Hvorfor bruge ikoner til at navigere, når ord er mere sigende og fylder mindre?
- Længere dokumenter fremfor brede.
- Husk at linke fra dine sider med indhold tilbage til en hovedside (indeks), bl.a. p.g.a. søgemaskiner.
- Udstyr dine sider med kontaktinformation
- En vertikal menu i siden optager mere plads på skærmen fremfor en horizontal menu

### Eksempel på SQL

Givet en tabel student med nogle studentedata

efternavn	fornavn	studienummer
Olesen	Peter	L2143
Hansen	Erika	J00007
Funder	Ulrik	Hg0014

Vi kan forespørge på fornavn og efternavn eksempelvis—sorteret stigende:

```
select fornavn, efternavn  
from student  
order by efternavn;
```

Vi kan indsætte en ny række:

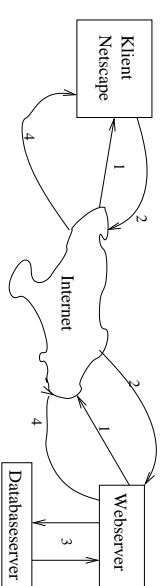
```
insert into student (efternavn, fornavn, studienummer)  
values ('Fong', 'Joe', 'JF0032');
```

Vi kan slette en række:

```
delete from student  
where efternavn = 'Fong';
```

... og meget mere

### Brug af CGI-scripts (TCL der afvikles på web-server)



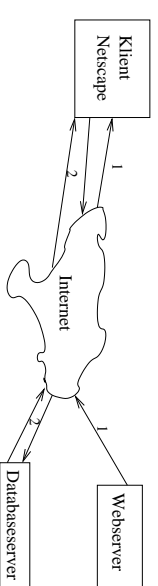
1. Klient henter HTML side fra web-server
2. Klienten indtaster data i HTML form, og sender data tilbage til web-server
3. Web-server afvikler et TCL program, som behandler data fra klienten, bl.a. ved at opdatere en database
4. Web-server sender en HTML side tilbage til klienten

Hvis HTML koden er simpel, og bl.a. ikke indeholder for megen JavaScript, kan den vises i de fleste browsere. Loader normalt hurtigt, men brugergrænsefladen er ikke som et normalt desktop program.

I dette kursus kører web-server (AOLserver) og database-server (Oracle) på høg . i t - c . dk.

I har alle jeres egen AOLserver og Oracle server på høg!

### Brug af Appletter til Opdatering af Databaser



1. Klient henter HTML side og applet fra web-server
  2. Klienten udfører appletten, som er et program; forespørgsler og opdateringer sendes direkte til database
- Appletter virker kun godt på nogle nyere browsere, og tager lang tid at downloade. Til gengæld er de mere interaktive.

Vi diskuterer ikke appletter og andre lignende teknologier (Flash, •Net m.m.) i dette kursus.

## Hvad er et programmeringssprog?

Et programmeringssprog er en notation for instrukser til datamaten, dvs. for programmerer.

Der findes mange forskellige programmeringssprog: Tcl, Perl, PHP, Standard ML, Java, C#, C, C++, Erlang, Haskell, Pascal, Forth, Fortran, Prolog, Ada, ...

Man skal være meget omhyggelig med programmeringssprog.

En computer er normalt meget pedantisk.

Dette er specielt til gene og frustration for begyndere.

Eksempel: Korrekt TCL program:

```
puts "I'm alive!"
```

Helt forkert TCL program:

```
Put "Im alive!"
```

Det kræver øvelse at spotte fejl.

## Et simpelt TCL program (name.tcl)

```
set firstrname Martin
set lastname Elsmann
set age 30
set email mael@itu.dk
puts "My name is $firstrname $lastname and my email address is $email."
```

Programmet består af

- en variabel `firstrname` med værdien `Martin`.
- en variabel `lastname` med værdien `Elsmann`
- en variabel `age` med værdien `30`
- en variabel `email` med værdien `mael@itu.dk`
- kommandoen `set` anvendt fire gange, som *tildele* en variabel en værdi
- kommandoen `puts`, som udskriver en streng (hvor variablerne er nævnt) på skærmen.

Programmet afvikles fra første linje og ned efter.

## Redigering og udførelse af et TCL program

### Linux

```
[23:15-S1011# tc1sh
% source name.tcl
My name is Martin Elsmann and my email address is mael@itu.dk.
%
```

Starter TCL fortolkeren `tclsh` og indlæser samt kører programmet med `source`.

Alternativt starter man fortolkeren og indtaster kommandoen en linje af gangen.

### Windows

I stedet for `tclsh` hedder TCL fortolkeren `wi.sh`.

I kan installere TCL fortolkere hjemme, både under Windows og Linux.

## Grundbegreber: Variabler og tildeling

En variabel kan indeholde en værdi, f.eks. tallet 30 eller strengen "Hello"

En variabel svarer til et sted i datamaterns lager.

Du navngiver selv dine variabler.

Du skaber en variabel med kommandoen `set`, som også *tildele* variabelen en værdi

```
set firstrname "Martin"
```

Når du skaber en variabel, så afsættes plads til variabelen i datamaterns lager.

Du kan overskrive indholdet af en variabel med en ny værdi:

```
set firstrname "Hans Otto"
```

Man kan tegne datamaterns lager og se hvad der sker når et program kører.

### Et simpelt TCL program der genererer HTML (hello.tcl)

```
puts "<html>
<head>
<title>
Hello
</title>
</head>
<body>
Hello
</body>
</html>"
```

### I Løbet af Ugen...

Alle deltagere

- bliver oprettet på hug . it . edu
- får tilsendt email med password til hug . it . edu
- bliver oprettet i CourseGrader (til brug for øvelsesaflevering)

Løs eventuelt ekstraopgaverne til løbeseddel 1.

Læs i bogen!

### Et mere avanceret TCL program der genererer HTML (loop.tcl)

Programmet, som I på nuværende tidspunkt ikke forventes at forstå:

```
proc mul_tab {n} {
  puts "<html><body><table>"
  puts "<tr><th>n</th><th>n*3</th><th>n*n</th></tr>"
  for {set i 0} {$i<n} {incr i} {
    puts "<tr><td>$i</td><td>[expr $i * 3]</td>
      <td>[expr $i * $i]</td></tr>"
  }
  puts "</table></body></html>"
}

og uddata:
23:46-S101]# tclsh
% source loop.tcl
% mul_tab 3
<html><body><table><tr><th>n</th><th>n*3</th><th>n*n</th></tr>
<tr><td>0</td><td>0</td><td>0</td></tr>
<tr><td>1</td><td>3</td><td>1</td></tr>
<tr><td>2</td><td>6</td><td>4</td></tr>
</table></body></html>
```