Laborationer i kursmomentet Datoranvändning E1

Laboration nr 7: Mer om FrameMaker och lite annat

1996, 1997 Magnus Bondesson
98-11-02 Thomas Hallgren

1 Introduktion

I denna laboration gör vi först några nya övningar i FrameMaker och Netscape. Sedan följer några övningar som tar upp komprimering av filer. Sist finns några övningar på UNIX-kommandon och några finurliga sätt att sätta ihop existerande kommandon till nya användbara kommandon.

Som förberedelse inför denna laboration kan det vara lämpligt att läsa de avsnitt i Gula Boken som det finns hänvisningar till i de olika uppgifterna.

2 Uppgifter

Som tidigare markerar symbolen ☐ en konkret arbetsuppgift.

Skriv ner svaren på frågorna och visa för en handledare för att bli godkänd. Handledaren vill förmodligen också titta på FrameMaker-dokument och andra filer du skapat under arbetets gång.

2.1 FrameMaker

☐ I Laboration nr 4 ägnade vi oss åt FrameMaker och du producerade ett dokument baserat på mallen *Rapport normal*. Nu jobbar vi vidare lite på detta, så ta fram ditt dokument och spara det under ett nytt namn, t ex `Rapport7.doc`, innan du går vidare. (När senare uppgifter refererar till `Rapport7` är det ditt dokument som menas.)

Uppgift 1. Egen styckeutformning

Egna utformningar tas upp i Gula Boken i avsnitt 7.7.

☐ Det händer då och då att man vill ha en styckeutformning som inte finns färdig i styckekatalogen. Din uppgift här är att skapa en sådan och placera den i styckekatalogen. Den nya utformningen skall heta *Citat*, ha kursiv stil och vara indragen 3 cm i förhållande till vänstermarginalen. Använd menyn **Utforma/Stycken/Utformning** för att få tillverkningsdialogen och sedan den dialogens **Kommandon/Ny utformning** för att få resultatet att hamna i styckekatalogen. Prova sedan att utformningen fungerar som den skall.

Uppgift 2. Egen teckenutformning

Att göra en teckenutformning och placera den i teckenkatalogen går till på ungefär samma sätt. I teckenkatalogen för ditt dokument finns antagligen *Betoning* som kursiverar markerade tecken. Gör en ny teckenutformning för nyckelord, och låt dem t ex visas med röd text och fetstil. Kalla den nya teckenstilen *Nyckelord* och lägg den i teckenkatalogen. Använd menyn **Utforma/Tecken/Utformning** för att få tillverkningsdialogen och sedan den dialogens **Kommandon/Ny utformning** för att få resultatet att hamna i teckenkatalogen.

Uppgift 3. Ändra stil på hela dokumentet

Om man har utformat sitt dokument genom att konsekvent använda de namngivna utformningarna i stycke- och teckenutformningskatalogerna kan man ganska lätt ändra stil på hela dokumentet genom att importera utformningar från ett annat dokument.

Tillvägagångssätt: man väljer **Arkiv/Importera/Utformningar...** från menyn. I fönstret som visas finns en meny där man kan välja vilket dokument man vill importera från. Här visas andra dokument som är öppna, så man måste alltså öppna det dokument man vill importera utformningar från först. Nyskapade dokument visas inte förrän efter man har sparat dem första gången. Det finns också en mängd knappar för att ställa in vad som ska importeras. Alla alternativ är påslagna från början och det är väl lämpligt om man vill man ändra stil helt och hållet på sitt dokument.

Prova att importera utformningarna från standardmallen `RapportNumerisk` eller `RapportKantrubruk` i ditt eget dokument. (För att se finessen med den senare bör man ha några stycken med utformningen *Rubrik2* i sitt dokument.)

Uppgift 4. Numrerade figurer

I rapporter är det vanligt att man numrerar figurer och hänvisar till dem från ett eller flera ställen i den löpande texten.

Det finns inget färdigt verktyg för att skapa numrerade figurer i FrameMaker. I dokumentmallen *Rapport Normal* finns det en styckeutformning som heter Bild, som det är tänkt att man ska använda när man skapar en numrerad figur. Själva figuren gör man plats för genom att sätta in en kopplad ram i stycket. Ramens placering ska vara **Under aktuell rad**. Prova att göra en figur på detta sätt.

En vanlig stil när det gäller placering av figurer är att de läggs under den rad där man refererar till figuren, om det finns plats. Annars låter man den löpande texten fortsätta och lägger figuren överst på nästa sida. Denna metod får man om man kryssar för alternativet **Flytande** när man skapar kopplade ramar i FrameMaker.

Nackdelen med ovanstående metod för numrering av bilder är att bildtexten inte följer med figuren om figuren hamnar på något annat ställe än under aktuell rad. Prova t ex att markera ramen du nyss gjorde och välj placeringen **Överst i spalten**.

Ett annat sätt numrera figurer är att använda tabeller istället för kopplade ramar. Dokumentet `~hallgren/Intro/Figurer.doc` innehåller ett exempel på hur man kan göra. Öppna det och ta en titt.

☐ Importera stycke- och tabellutformningarna från `Figurer.doc` i ditt eget dokument och skapa en numrerad figur.

Uppgift 5. Generera PostScript-filer

Hur gör man om man vill göra sina FrameMaker-producerade dokument tillgängliga för andra via webben? Ett sätt är förstås att kopiera FrameMaker-filen till sin webbplats, men då måste den som vill titta på dokumentet ha FrameMaker, och det är det ju lång ifrån alla som har. Ett bättre alternativ är att skapa en PostScript-fil.

PostScript är det format FrameMaker använder för utskrift på skrivare, så att skapa en PostScript-fil är en variant på att göra en utskrift. Man väljer alltså **Skriv ut...** i **Arkiv**-menyn och kryssar i rutan **Skriv till fil**. FrameMaker föreslår ett filnamn, men man kan ändra på det om man vill.

☐ Skapa en PostScript-fil för ditt dokument.

Uppgift 6. Generera PDF-filer

Nyare versioner av FrameMaker (t ex 5.5) kan generera filer i PDF (Portable Document Format), en variant på PostScript som har blivit populär för elektronisk spridning av dokument. För att titta på PDF-filer kan man använda Acrobat Reader, som är gratis och finns för flera olika datorsystem.

Den version av FrameMaker som finns på E:s datorer är 5.1, så tyvärr kan vi inte öva på att skapa PDF-filer, men vi kan titta på ett färdigt PDF-dokument för att se hur det kan se ut.

☐ Ta en titt på PDF-filen `~hallgren/Intro/Laborationer.pdf`. Unix-kommandot för att starta Acrobat Reader heter `acroread`. Notera att man snabbt kan hoppa till olika delar i dokumentet genom att klicka i listan till vänster och att man kan klicka på trianglarna i listan för att ändra listans detaljnivå. Prova också att klicka på korsreferener i texten.

2.2 Netscape

Uppgift 7. Surfa lokalt

Den vanligaste användningen av Netscape är väl att surfa på webben, men man kan faktiskt surfa bland sina egna filer också. Vi tittade på lokala filer redan i Laboration nr 1. Faktum är att man även titta på kataloger. Filer och underkataloger blir då länkar, så istället för att använda `cd`, `ls`, `more` och andra kommandon för att bläddra runt och titta på filer kan man enkelt klicka sig fram med Netscape.

☐
- Titta på labbkontots hemkatalog med Netscape genom att skriva in `file:` följt av gruppens hemkatalog i **Location**-rutan (Vilket det är kan man ta reda på med kommandot `pwd`. `~`-notationen fungerar inte, tyvärr). Passa på att sätta ett bokmärke, om du lätt vill kunna göra detta fler gånger.

  Anm. Du märker säkert att Netscape visar filnamn som börjar med `.`, vilka i normala fall inte brukar visas.

- I katalogen finns förmodligen diverse olika slags filer. Netscape känner förstås inte till *alla* filformat. HTML-filer visas förstås i Netscapes fönster, men vad händer om man klickar på

a. en enkel textfil (t ex Berling.txt)? ____________________

b. en GIF-bild? ______________________________________

c. en JPEG-bild? _____________________________________

d. en PostScript-fil? __________________________________

e. en m-fil från MATLAB? ______________________________

f. ett FrameMaker-dokument? __________________________

Uppgift 8. Ändra Netscapes inställningar

Som du upptäckte i förra uppgiften startar Netscape ibland andra program för att visa filer av typer som Netscape inte kan visa själv. Det finns inställningar, som man kan ändra på om man vill, som avgör hur olika filtyper hanteras. I Netscape version 3 hittar man dessa genom att i menyn **Options** välja alternativet **General Preferences...** och i fönstret som dyker upp klicka på **Helpers**. Då ser man en lista med filtyper.

Vi provar att ändra inställningen för hur GIF-bilder hanteras.

- Leta upp filtypen `image/gif` i listan. Klicka på den och sedan på knappen **Edit...**. Det dyker upp ett fönster med inställningar för GIF-filer. I rutan **Handle By** kan man välja bland annat låta Netscape själv visa filen (alternativet **Navigator**) eller att starta ett annat program för att visa filen (alternativet **Application**). Förmodligen är det senare alternativet valt (dvs GIF-bilder visas av något annat program), så prova att ändra så att Netscape själv visar GIF-bilder. Om det redan var den inställningen som var vald, prova att välja **Application** och skriva in kommandot `xv %s` i rutan. (När Netscape sedan startar xv kommer `%s` att ersättas med ett filnamn.) Tryck på **OK** (i två fönster) och prova att klicka på någon GIF-bild i din hemkatalog och kolla att det fungerar.

- (Om du får tid över) Om man tycker att texten på webbsidorna i Netscape är svårläst, för stor eller för liten kan man ändra inställningarna under **Options/General Preferences.../Fonts**. Personligen tycker jag att New Century Schoolbook är ett trevligare typsnitt än Times, som brukar vara det förvala.

2.3 Komprimering

När man lagrar information kan man spara plats genom olika former av komprimering. I Laboration nr 6 tittade vi på olika filformat för bilder, varav några (GIF & JPEG) hade inbyggd komprimering. Det finns även program som kan komprimera godtyckliga filer. `gzip` är ett ofta använt sådant i Unix (i Windows är WinZip ett populärt program. På Macintosh finns t ex StuffIt och Compact Pro).

Generella komprimeringsprogram fungerar alltså på godtyckliga filer. Kan man verkligen komprimera information utan att veta något om dess uppbyggnad? Vi tar reda på det genom att göra några komprimeringsexperiment. `gzip` används enligt:

`gzip` *`filnamn`*

där *`filnamn`* är namnet på den fil man vill komprimera. Denna ersätts då med en komprimerad fil som får tillägget `.gz` sist i namnet. Man kan dekomprimera en fil med kommandot

`gunzip` *`filnamn.gz`*

där *`filnamn`*`.gz` är namnet på den komprimerade filen. Filen ersätts då med en dekomprimerad fil med `.gz` borttaget ur namnet. Innehållet i denna är exakt samma i den ursprungliga okomprimerade filen, dvs komprimeringen är förlustfri.

Uppgift 9. Komprimeringsexperiment

☐ Prova `gzip` på några filer och fyll i tabellen (om du inte redan har filerna kan du kopiera dem från `~hallgren/Intro/`):

Fil	Ursprunglig storlek (s0)	Komprimerad storlek (s1)	Kompressions-faktor (s0/s1)
`Berling.txt`			
`Propaganda.txt`			
`blomvas.ppm`			
`screendump1.ppm`			
`blomvas.jpg`			
`blomvas.gif`			
`lab_7.doc`			
`lab_7.ps`			
`lab_7.pdf`			

☐ Klarar Netscape av att visa en komprimerad fil? Prova att klicka på t ex den komprimerade versionen av `Propaganda.txt` (dvs `Propaganta.txt.gz`). Vad händer? ________________________________

☐ Med utgångspunkt från dessa experiment kanske du kan våga dig på att säga något om vad de filer som går bra att komprimera på detta sätt har gemensamt?

____

2.4 UNIX-kommandon, omdirigering och rörledningar

Uppgift 10. Några användbara UNIX-kommandon: Folk och datorer

☐ Pröva följande tre kommandon. Se manualblad eller Gula Boken för mer information.

- `finger` *användarnamn* eller *förnamn* eller *efternamn*
  Ger information om alla användare med aktuellt namn
- `rwho`
  Ger information om alla påloggade användare vid E-systemet
- `v`
  Lokalt UNIX-kommando, som ger information om lediga/upptagna datorer.

Uppgift 11. Omdirigering

Normalt hamnar resultaten av ett UNIX-kommando i det fönster i vilket kommandot givits, vilket brukar kallas "standard-ut". Men man kan också dirigera resultaten så att de i stället hamnar i en vanlig fil, med (> kan utläsas "från kommandot till")

`kommando >`*`filnamn`*

Läs mera om omdirigering i avsnitt 1.16.1 i Gula Boken.

☐ Se till att resultatet av `ls -l` hamnar som filen `LISTA`. Kontrollera med t ex `more` att `LISTA` innehåller det den skall.

☐ Programmet `~hallgren/Intro/slumpa` genererar en massa slumptal. Kör programmet. Kör det sedan så att resultaten i stället hamnar som filen `slumptal`.

På motsvarande sätt kan ett kommando som normalt hämtar information från tangentbordet ta den från en fil. Man använder då i stället tecknet < (”till kommandot från”). Vi avstår från någon uppgift på detta.

Uppgift 12. Några användbara UNIX-kommandon: Filer

Låt oss titta på tre kommandon `wc`, `grep` och `sort` som alla kan användas på två olika sätt. Antingen skriver man ett filnamn tillsammans med kommandot eller också skriver man in information på raderna efter kommandot och anger indataslut med CTRL-D. Mer information på vanligt sätt (manualblad resp sid 58).

☐ Tag med hjälp av `wc` reda på antal rader, ord och tecken i filen `Berling.txt`.

☐ Kommandot `grep` används för att söka efter teckenföljder i en fil. När `grep` funnit en rad som innehåller den sökta textsträngen så skriver `grep` ut denna rad. Skriv ut de rader i filen `Berling.txt` som innehåller teckenföljden `hon`. Ta också reda på vilka som innehåller ordet hon. (Tips: läs vad `grep -w` betyder i manualbladet för `grep`.)

☐ Kommandot `sort` används för att sortera text eller tal. Sortera raderna i `Berling.txt` alfabetiskt.

Uppgift 13. Rörledningar (pipes)

Läs om hopkoppling av kommandon i Gula Boken, sid 23.

☐ Ange hur man med hjälp av hopkoppling av kommandon på ett smidigt och enkelt sätt kan se hur många hemkataloger det finns på datorsystemet som tillhör E-teknologer inskrivna 1998 (katalogen `/u1/e98`; använd kommandona `ls` och `wc`): __________

☐ Hur många kataloger finns det? ______________

☐ Det tidigare nämnda programmet `slumpa` producerar osorterade slumptal. Skriv en rörledning som gör att de kommer ut sorterade i växande ordning på skärmen (*sort* behöver kanske en extra flagga).

Uppgift 14. Kommandofiler

När man startar ett terminalfönster startas normal kommandotolken `tcsh`, som läser de kommandon man matar in på tangentbordet och utför dem. En naturlig fråga är nu: kan `tcsh`, liksom kommandona vi tittade på i Uppgift 12, även läsa indata från en fil? Svaret är ja! Har man en fil som innehåller kommandon kan man få `tcsh` att utföra dem genom att skriva `tcsh` *`filnamn`*.

Filer som innehåller kommandon kallas inte helt oväntat för **kommandofiler** (se sidan 29 i Gula Boken). En kommandofil är alltså en textfil (tillverkas med t ex *NEdit*) bestående av ett antal kommandorader.

☐ Gör med *NEdit* en fil med innehållet

```
date
echo $1 $USER
```

Spara den under namnet `kommando1`. Provkör kommandofilen genom att mata in t ex

```
tcsh kommando1 Hejsan
```

Vad skrivs ut? ____

Förklaring: `$1` är en sorts variabel som ersätts med det ord som skrivs efter `kommando1` och `$USER` ersätts med användarnamnet. Kommandot `echo` skriver ut en textföljd, dvs kommandofilanropet kommer att skriva ut datumuppgifter och sedan `Hejsan` *`användarnamnet`*.

Det går även att göra kommandofiler lika lätta att använda som vanliga kommandon (dvs man slipper skriva namnet på kommandotolk varje gång man använder dem). `kommando1` görs körbar med UNIX-kommandot `chmod` enligt

```
chmod +x kommando1.
```

Därefter används kommandofilen med t ex

```
kommando1 Hejsan
```

☐ Prova detta!

Anm. Kommandofiler som används på detta sätt tolkas av kommandotolken `sh`, inte `tcsh`, som på E är den normala kommandotolken för det man själv skriver in i ett terminalfönster. I grunden fungerar båda på samma sätt, men lite mer avancerade saker skrivs på olika sätt. Vill man ange explicit vilken kommandotolk en kommandofil ska tolkas av kan man ange detta på första raden i filen. För att få `tcsh` kan man skriva
`#! /usr/local/pd/bin/tcsh -f`

Uppgift 15. e8rwho

☐ Tillverka ett nytt kommando som fungerar som `rwho`, men utskriften begränsas till användare vars namn börjar på `e8`. Kalla kommandot `e8rwho`.

3 Avslutande kommentarer

I år är det meningen att alla projektgrupper ska göra en hemsida på webben för sina projekt och göra projektrapporten tillgängliga via den. Lämpliga format att använda för rapporten är då HTML, förstås, eller PDF. Version 5.5 av FrameMaker har stöd för generering av såväl HTML som PDF. Tyvärr har E-sektionen bara version 5.1, så vi kommer förmodlingen att få nöja oss med att generera PostScript-versioner av rapporterna.

Kommentarer till FrameMaker: man kan lära sig mycket genom att själv experimentera och utforska möjligheterna. Det vi hinner ta upp allt på laborationerna är ju bara ett litet smakprov.

Kommentar till att använda Netscape för att bläddra bland egna filer: de som tidigare har använt Macintosh eller Windows på PC är förmodligen vana att enkelt kunna bläddra runt bland sina filer (med Finder respektive Utforskaren) och tycker kanske det verkar fånigt att använda Netscape för detta. Faktum är att det finns program för Unix som presenterar filer och kataloger ungefär som på Mac/Windows, men inget sådant finns installerat på E:s datorer, så vitt jag vet...

3.1 Information på webben

Det är företaget Addobe som gör både FrameMaker och Acrobat Reader. Adobe har webbsidor på `www.adobe.se` och `www.adobe.com`. Man kan hitta information om de två programmen på t ex sidorna

`http://www.adobe.com/prodindex/acrobat/`

`http://www.adobe.se/produktinfo/framemaker/`

Information om Netscape 3.0 finns på (fås också med **Help/Handbook** i menyn):

`http://home.netscape.com/eng/mozilla/3.0/handbook/`

Komprimeringsprogrammet gzip har en egen webbadress: `www.gzip.org`! Man kan hitta information komprimering och fler komprimeringsprogram i Yahoo-katalogen:

`http://dir.yahoo.com/Computers_and_Internet/Software/System_Utilities/Compression/`