# Bloggarkivering - en jämförelse av insamlingstekniker för långtidsbevaring av bloggar

Emil Sandin

Examensarbete

Medieteknik

2011

| EXAMENSARBETE | |
|---|---|
| Arcada | |
| | |
| Utbildningsprogram: | Informations- och medieteknik |
| | |
| Identifikationsnummer: | 3237 |
| Författare: | Emil Sandin |
| Arbetets namn: | Bloggarkivering – en jämförelse av insamlingstekniker för långtidsbevaring av bloggar. |
| Handledare (Arcada): | Johnny Biström |
| | |
| Uppdragsgivare: | Svenska Litteratursällskapet r.f. |
| | |


Sammandrag:

Bloggar är en vanlig uttrycksform på nätet idag, men den dynamiska uppbyggnaden av bloggar gör insamling och arkivering svårare än för statiska webbsidor. Detta examensarbete undersöker tillgängliga tekniker för insamling av bloggar för långtidsbevaring. Arbetet är gjort för Svenska Litteratursällskapet r.f.

Syftet med arbetet är att ge en översikt av de tekniker som kan användas för insamling av bloggar, samt skillnaderna mellan dessa. En jämförelse av olika tekniker görs med en litteraturkälla som grund för jämförelsen. Den teoretiska delen behandlar långtidsbevaring och de problem som finns, samt bloggens historia, format och tekniska uppbyggnad. Den praktiska jämförelsen görs genom att samla in två bloggar med de olika insamlingsteknikerna för att sedan jämföra det insamlade materialet.

De insamlingstekniker som behandlas i detta arbete är insamling direkt i server eller CMS, insamling med robot, insamling med webbläsare samt insamling av RSS-flöden.



| Nyckelord: | Bloggarkivering, långtidsbevaring, bloggar, insamlingstekniker |
|---|---|
| Sidantal: | 51 |
| Språk: | Svenska |
| Datum för godkännande: | 25.5.2011 |

| | |
|---|---|
| DEGREE THESIS | |
| Arcada | |
| | |
| Degree Programme: | Information- and mediatechnology |
| | |
| Identification number: | 3237 |
| Author: | Emil Sandin |
| Title: | Blog archiving – a comparison of capturing techniques for long-term preservation of blogs |
| Supervisor (Arcada): | Johnny Biström |
| | |
| Commissioned by: | The Society of Swedish Literature in Finland |
| | |


Abstract:

Blogs are a common way of expression on the web in today's world, but their dynamic structure makes collection and archiving harder than for static web pages. This thesis examines available techniques for collection of blogs for long-term preservation. The thesis is written for the Society of Swedish Literature in Finland.

The purpose of this thesis is to give an overview of the techniques that can be used for collection of blogs, and the differences between the techniques. A comparison of different techniques is made with a literature source as the basis for the comparison. The theoretical part of the thesis presents long-term preservation and its problems, and also the history, format and structure of blogs. The practical comparison is done by collecting two different blogs, using the different techniques in the comparison, and then comparing the result of the collections.

The different collecting techniques examined in this thesis are collection straight from server or CMS, collection with a crawler, collection within the web browser and collection of RSS-feeds.



| | |
|---|---|
| Keywords: | Blog archiving, long-term preservation, blogs, collection techniques |
| Number of pages: | 51 |
| Language: | Swedish |
| Date of acceptance: | 25.5.2011 |

# INNEHÅLL

## Figurer / Figures

# 1 INLEDNING

Bloggar är en mycket viktig uttryckssform för många privatpersoner och många organisationer. Det skapas tusentals nya bloggar på internet varje dag, men många försvinner också. Många av dessa bloggar har ett kulturellt värde som borde sparas för eftervärlden. Men hur går man tillväga för att spara en blogg för framtida bruk?

## 1.1 Bakgrund

Det här arbetet ger en översikt av de olika insamlingstekniker som kan användas för att samla in en blogg för långtidsbevaring i digitala arkiv. Skillnaderna mellan de olika teknikerna presenteras också så att man ska kunna använda detta arbete som grund när man väljer en insamlingsteknik för en specifik blogg.

Detta arbete är gjort inom ramarna för Arkarva, ett projekt mellan Arcada och Svenska Litteratursällskapet r.f. (härefter SLS). Eftersom SLS i första hand intresserar sig för findlandssvensk kultur och litteratur, så har findlandssvenska bloggar varit mest intressanta även för detta arbete. Detta syns till exempel på att YLE:s bondbloggen.fi används för att testa de olika insamlingsteknikerna.

## 1.2 Syfte och mål

Syftet med detta arbete är att ge en översikt av de olika tekniker som man kan använda för att samla in bloggar för långtidsarkivering. Jag kommer även att göra en jämförelse mellan ett antal verktyg som använder de olika teknikerna.

Målet med detta arbete är att hjälpa läsaren att avgöra vilken teknik som är bäst lämpad för att samla in en blogg. Det kommer kanske inte gå att säga att en viss teknik alltid är bäst, utan det kan variera från fall till fall beroende på vilka delar i en blogg man prioriterar mest.

De frågeställningar som behandlas i detta arbete är följande:

- Vilka tekniker för insamling av bloggar finns?
- Hur skiljer sig de olika teknikerna åt?
- Hur fungerar de olika teknikerna?
- Hur bra uppfyller de olika teknikerna kraven som ställs för långtidsbevaring?

## 1.3 Metod

Detta arbete är en jämförelse av olika insamlingstekniker av bloggar för långtidsbevaring. Studier av tidigare forskning samt diskussioner med personer på SLS har använts för att välja ut ett antal tekniker till jämförelsen. För att testa de olika teknikerna och kunna jämföra dem har ett antal verktyg som använder sig av de aktuella teknikerna valts ut för att användas i det praktiska testet. Endast gratis samt fritt tillgängliga verktyg har använts i testet.

Testet har utförts på två bloggar. Den första är bondbloggen.fi, en aktiv blogg som upprätthålls av svenska Yle. Denna blogg användes för testerna på rekommendation av SLS. Den andra bloggen skapades specifikt för dessa tester för att ha full kontroll på allt material som fanns på bloggen.

Testkriterierna har valts utgående från de riktlinjer som ges om skapande av bevaringsplaner i arbetet ”Systematic planning for digital preservation: evaluating potential strategies and building preservation plans” från 2009 av Christoph Becker med flera. (Becker, 2009)

## 1.4 Avgränsning

I detta arbete kommer jag att koncentrera mig på insamling av bloggar för långtidsarkivering. Jag kommer även att ta upp hur väl de insamlade filerna lämpar sig för långtidsbevaring. Däremot kommer jag inte att behandla hur det insamlade materialet arkiveras och hur man säkerställer bevaringen. Jag kommer inte heller att

behandla upphovsrättsliga frågor och problem som kan uppstå, utan utgår från att man har tillstånd att samla in bloggen.

Även om de tekniker som testas i detta arbete går att använda på de flesta typer av bloggar, så kommer jag i detta arbete främst testa utgående från SLS:s behov. Detta innebär bloggar skrivna av privatpersoner eller organisationer, som oftast fungerar som någon slags dagbok eller journal. Nyhetssidor på nätet kan ofta uppfylla definitionen på en blogg, men dessa rena nyhetssidor eller nyhetsbloggar kommer inte att behandlas i detta arbete.

## 1.5 Termer och begrepp

*Emulering och migrering*

För att den insamlade informationen ska vara läsbar även i framtiden måste man antingen se till att plattformen för att läsa materialet finns kvar (emulering) eller ändra innehållets format vartefter nya format uppkommer och gamla försvinner (migrering)

*Metadata*

Metadata kan enkelt beskrivas som ”information om informationen”. Den vanligaste typen av metadata är upphovsman, datum för skapandet, datum för insamling/arkivering, samt information om vilken slags material det är frågan om och vad det handlar om.

*Autenticitet*

Autenticitet betyder att man måste kunna spåra var materialet kommer ifrån. Detta betyder att man måste kunna säkerställa att materialet faktiskt är det samma som en gång i tiden blev arkiverat. Praktiskt löses detta i arkiv med nogranna loggar om vem som har gjort vad med materialet och när detta har skett.

*CMS*

CMS (Content Management System) eller innehållshanteringssystem, är ett system som används för att enkelt publicera innehåll på webben. Wordpress är ett innehållshanteringssystem som behandlas i detta arbete.

*XML*

XML (eXtensible Markup Language) är ett märkspråk som används för att utväxla information mellan olika informationssystem. Data skickas som ren text, som även kan förstås av människor.

# 2 LÅNGTIDSBEVARING

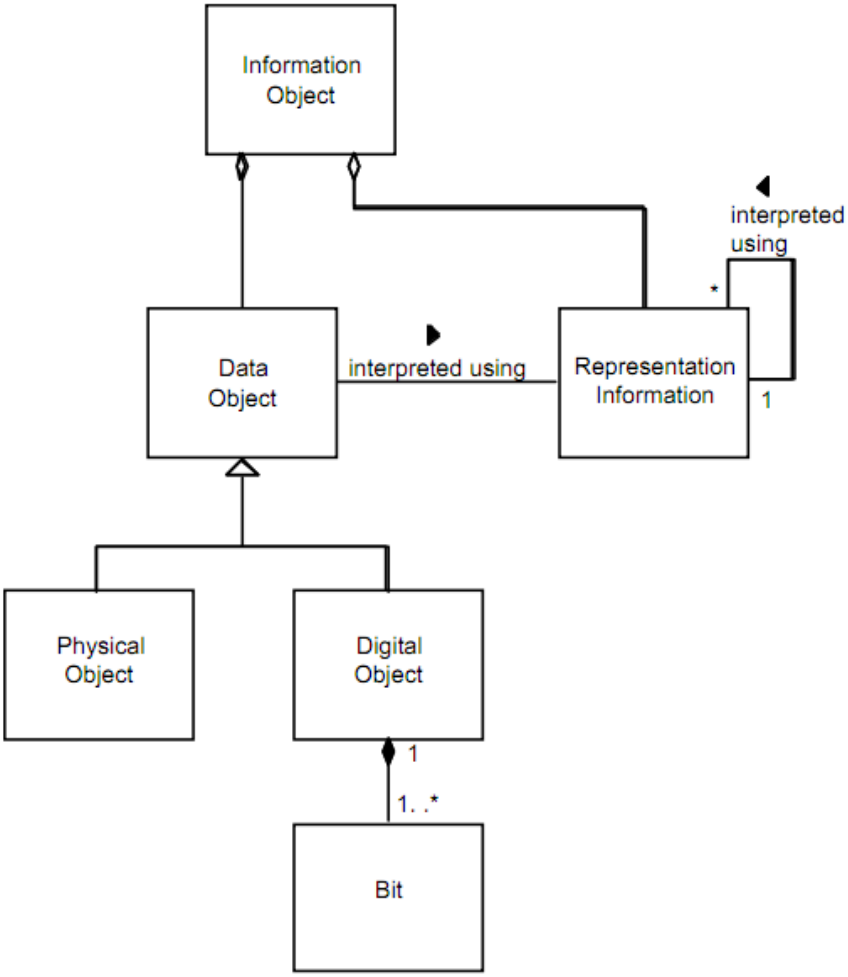
Begreppet långtidsbevaring är viktigt att definiera i detta sammanhang. Hur länge är en ”lång tid” när det gäller digitala arkiv? OAIS-standarden (Open Archival Information System) definierar en lång tid inom långtidsbevaring som en så lång tid att man måste ta i beaktande att tekniken kan ändras, t.ex. nya medie- och dataformat kan tillkomma och gamla format försvinna. (OAIS 2011)

Det här innebär att man måste ta i beaktande att program och plattformar som används idag inte finns allmänt tillgängliga i framtiden, vilket kan innebära problem att läsa det arkiverade materialet. För en blogg kan det t.ex. innebära att man inte längre har en fungerande serverplattform att köra bloggen på, eller att de videoformat som var allmänt i bruk inte längre går att spela upp.

## 2.1 OAIS – Open Archival Information System

OAIS är ett ramverk ursprungligen utvecklat av NASA för att hantera och lagra digital information. OAIS är mycket mångsidig och går att tillämpa på all lagring av digital information. Den används därför numera som en standard för elektroniska arkiv som hanterar digital information. (OAIS 2011)

Ett viktigt grundkoncept i OAIS referensmodellen är att information är en kombination av data och representationsinformation. Det här innebär att ett informationsobjekt består av ett dataobjekt som kan vara endera i fysisk eller digital form, samt representationsinformationen som möjliggör en full tolkning av dataobjektet till meningsfull information. För en blogg så skulle det insamlade materialet utgöra dataobjektet, medan metadata och loggar från insamlingen skulle utgöra en del av representationsinformationen. (CCSDS 2002 s. 4-19 f.)




*Figur 1. Principen för ett informationsobjekt i OAIS (CCSDS, 2002)*

## 2.2 Krav

Långtidsbevaring handlar inte bara om att säkerställa att samma filer som man arkiverar finns kvar efter hundra år. Man måste även kunna läsa informationen och veta filernas ursprung för att ha någon nytta av materialet. Man kan definiera två stora grupper som ställer krav på materialet som arkiveras. Den ena gruppen är den som har ansvar för arkiveringen, det vill säga arkivarierna. Den andra gruppen är användarna, alltså de som kommer att använda det arkiverade materialet i framtiden. Arkivariernas krav berör främst aspekter som har med arkivbeständighet att göra, det vill säga materialet ska kunna bevaras oförändrat och fortfarande vara läsbart i framtiden. Användarnas krav på

materialet handlar mera om användbarhet och tillgänglighet, det vill säga materialet bör vara enkelt att söka och hitta information i.

### 2.2.1 Metadata

Enligt OAIS-modellen måste data som arkiveras åtföljas av representationsinformation som möjliggör tolkning av datan till betydelsefull information. Det här innebär att man även måste samla in information om den data som samlas in. I DAVID - Archiving Websites (Boudrez & Van den Eynde 2002 s.42-44) finns en lista på olika typer av metadata som är önskvärd att samla in om en webbplats som arkiveras. En del av de saker som nämns där är:

ALLMÄNT

- Titel
- Skapare
- Ämne/abstrakt
- Funktion/syfte med sidan
- Besökarantal

WEBBSERVER

- Operativsystem
- Mjukvara och konfiguration för webbserver
- Server-script (bl.a. CGI, ASP, PHP)
- Loggfiler
- Dokumentation

INSAMLAT MATERIAL

- Arkiverad version
- Saknade filer

TEKNISKA DETALJER

- URL/IP-adress
- Antal filer och mappar
- Total storlek på materialet
- Eventuella lösenord
- Filformat som används
- Mjukvara som krävs för att visa materialet

HISTORIA

- Tidpunkt som materialet fanns tillgängligt
- Modifieringar/uppdateringar
- Datum för insamling

Boudrez och Van den Eynde konstaterar också att all denna information inte går att få ut från det insamlade materialet. De föreslår därför att man utarbetar ett webbformulär som de ansvariga för bloggen får fylla i, för att samla in den metadata som endast dessa personer kan bidra med. (Boudrez & Van den Eynde 2002 s.42-47)

## 2.3 Problem

När digital information ska lagras i långa tider uppstår flera problem som inte uppstår vid kortare lagring. Problemen skiljer sig även från de problem som uppstår i fysiska arkiv.

### 2.3.1 Läsbarhet

Om man jämför digitala arkiv med traditionella fysiska arkiv så hittar man flera problem som endast uppstår inom digitala arkiv. En bok kan stoppas undan i en bokhylla, tas fram 100 år senare och fortfarande vara fullt läsbar, förutsatt att ingenting

oförutsett som brand eller översvämmning skett. En digital fil måste däremot övervinna flera hinder för att samma sak ska vara möjlig:

*Hållbarhet på lagringsmedia*

Medan en bok kan utsättas för relativt mycket åverkan och ändå vara åtminstone delvis läsbar, så kan digitala lagringsmedier skadas mycket lättare. En CD-skiva kan bli oläslig av en enda rispa, en hårddisk kan raderas av ett magnetfält eller gå sönder av att tappas i golvet och ett flashminne kan förstöras av statisk elektricitet. (Digital Preservation Europe 2011b)

*Utdöd hårdvara*

Även om man lyckas bevara lagringsmediet utan att det skadas eller förstörs, så är det inte all säkert att man fortfarande har den hårdvara som krävs för att läsa infomationen på mediet. (Digital Preservation Europe 2011b)

*Utdöda eller förlegade filformat*

Slutligen måste man även kunna tolka filerna om man lyckas läsa dem från lagringsmediet. Man måste då ha program som klarar av att öppna och läsa filformatet som filen en gång sparades i. Detta problem kan redan idag uppstå på filer som endast är 10-20 år gamla. (Digital Preservation Europe 2011b)

Digitala arkiv kräver regelbunden översyn och uppföljning. Vartefter som tekniker blir föråldrade måste man avgöra om man migrerar materialet till en ny teknik, eller om man bevarar den gamla plattformen genom emulering.

### 2.3.2 Autenticitet

För att det arkiverade materialet ska kunna användas i framtiden måste man även veta vad materialet är och vad det handlar om. Man måste även kunna lita på att materialet är det samma som en gång arkiverades, och att det inte har ändrats under tiden i arkivet,

antingen avsiktligt eller oavsiktligt. Om materialet har ändrats avsiktligt, till exempel i samband med migrering, måste man veta när detta har skett och vad som har gjorts.

Eftersom webbsidor och därmed även bloggar renderas av en webbläsare utgående från html-kod, så kan webbsidor se olika ut beroende på vilken webbläsare som används. Om man i framtiden öppnar sidan med en modern webbläsare, kan informationen på sidan visas fel eller inte visas alls, beroende på hur webbläsaren renderar koden. Det kan därför vara viktigt att på något sätt kunna kontrollera att sidan ser korrekt ut, t.ex genom att använda en gammal webbläsare eller genom att redan i arkiveringsskedet spara en renderad version av sidan. (Boudrez & Van den Eynde 2002 s.26-29)

### 2.3.3 Dynamiskt och länkat innehåll

Ett problem som uppstår med bloggar och annat webbmaterial är hur man ska hantera dynamiskt och länkat innehåll. Medan material som bilder, böcker eller ljudklipp oftast är tydligt avgränsade objekt, kan en blogg vara svår att avgränsa på ett tydligt sätt. Hur gör man med inbäddat material i bloggen, som finns lagrat på en annan plats, t.ex. ett videoklipp från YouTube. Här uppstår flera probem som man måste ta ställning till. Går detta material att samla in? Är det viktigt att samla in? Har man rättigheter att samla in detta material? Samma frågor kan även uppstå om man råkar ut för ett blogginlägg som är utformat som en kommentar till ett annat, länkat inlägg från en annan blogg, så att inlägget på bloggen man samlar in saknar sammanhang och relevans utan det externa inlägget.

Det dynamiska sättet på vilket en blogg oftast är uppbyggd kan också ställa till problem. En blogg kan ha olika layout för olika webbläsare, till exempel för mobiltelefoner. Layouten kan också styras av till exempel datum, så att bloggen får ett jultema i december. När innehållet och utseendet på bloggen förändras dynamiskt på detta sätt kan det bli svårt att samla in allt, eller så kan det bli svårt att säkerställa att man faktiskt har samlat in det innehåll man tror att man samlat in. (Boudrez & Van den Eynde 2002 s.26-29)

# 3 BLOGGEN

För att kunna bedöma vad och hur man ska samla in information från en blogg är det viktigt att veta vad en blogg är, och vilka olika delar en blogg kan bestå av. Det är även viktigt att förstå vad det är som skiljer en blogg från ”vanliga” webbsidor.

## 3.1 Historia

Ordet blogg härstammar från ordet webblogg, som i sin tur är det svenska ordet för engelskans weblog. Även det engelska ordet har en kortare variant som ofta används, blog. Uttrycket weblog myntades av John Barger i december 1997. Peter Merholz började 1999 skriva weblog som ”we blog” (= vi bloggar). Snart föll ”we” bort, och webbloggarna kallades kort och gott ”blogs”, eller ”bloggar” på svenska. (Blood 2000)

I början debatterade man mycket om vad som egentligen krävdes för att en sida skulle kunna kallas för en blogg. Brigitte Eaton skapade en lista på alla för henne kända bloggar 1999. Hon hade endast ett kriterium för att en sida skulle kallas weblog: Sidan skulle bestå av daterade inlägg. Eftersom hennes lista var den mest kompletta listan över existerande bloggar så kvarstod hennes definition och blev mer eller mindre accepterad som standard. (Blood 2000)

De första bloggarna bestod av länkar till andra sidor som blogförfattaren hittat och tyckte var intressanta. En kort beskrivning av sidan kunde också ingå på bloggen. De som publicerade bloggar på den här tiden var oftast teknikintresserade, eftersom inga blogg-verktyg ännu existerade, så man var tvungen att manuellt uppdatera bloggens html-kod för att lägga till innehåll. (Blood 2000)

1999 började bloggandet ta fart på allvar. Detta berodde till stor del på att flera bloggverktyg lanserades då, som gjorde det möjligt för vem som helst att skapa en egen blogg utan några större tekniska kunskaper. De mest kända verktygen som lanserades 1999 var LiveJournal och Blogger. (Blood 2000)

Den idag mest populära bloggplattformen Wordpress lanserades 2003 som en vidareutveckling av den tidigare plattformen b2/cafelog. Wordpress är lanserad som öppen källkod under GPLv2-licensen. (WordPress 2011)

Enligt W3Techs användes WordPress på 13,8 % av de en miljon största sidorna på internet 21.3.2011. Enligt wordpress.org har den senaste versionen, 3.1, blivit nedladdad över tre miljoner gånger fram till 21.3.2011. (W3Techs 2011, Wordpress.org 2011a)

Sedan starten har antalet bloggar ökat mycket snabbt. Enligt tjänsten BlogPulse så fanns det den 21.3.2011 över 158 miljoner publika bloggar på internet. (The Nielsen Company 2011)

## 3.2 Definition

Att hitta en allmän och heltäckande definition på vad som egentligen definierar en blogg är svårt. Enligt Svenska Akademiens ordlista är en blogg en ”personlig dagbok på webben som uppdateras kontinuerligt och är öppen för kommentarer.” The Economist och många andra tidningar använder ofta en kortare definition, ”personlig dagbok på nätet.”(SAOL 2011 s.90, The Economist 2006)

För det här arbetet har jag valt att definiera en blogg som en personlig dagbok på nätet skriven av en eller flera personer privat eller i en organisation. Jag räknar även med att inläggen kan innehålla bilder och att det finns en funktion för att skriva kommentarer till inläggen.

## 3.3 Teknisk uppbyggnad

I början var bloggar oftast statiska HTML-sidor, men moderna bloggar är oftast databasdrivna och dynamiska. Den populära bloggplattformen WordPress använder MySQL för att lagra innehåll så som inlägg och kommentarer, PHP för att skapa webbsidor och CSS för layouten. Detta gör det möjligt att enkelt sortera poster till exempel enligt månad eller kategori.

*HTML*

HTML (HyperText Markup Language) är det språk som används för att visa webbsidor. HTML består av olika så kallade taggar, som anger hur innehållet mellan dessa ska visas. HTML-koden skickas av webbserverns till besökarens webbläsare, som renderar en sida som visas åt besökaren. När det är frågan om statiska sidor så lagras HTML-koden vanligen i en fil med ändelsen .html eller .htm. (Boudrez & Van den Eynde 2002 s.8-9)

*PHP*

PHP (Hypertext Preprocessor) är ett skriptspråk, som körs på en webbserver för att leverera webbsidor med dynamiskt innehåll. PHP härstammar från PHP/FI, skapat av Rasmus Lerdorf 1995. (PHP Group, 2011)

PHP är det som gör en blogg dynamisk. När någon besöker en sida på en blogg, så begär besökarens webbläsare denna sida av webbservern. Om sidan är dynamisk så består den av en PHP-fil på servern. Denna PHP-fil körs på servern och innehåller instruktioner för hur HTML-koden som skickas till besökarens webbläsare ska se ut. Dessa instruktioner kan för en blogg handla om att hämta ett blogginlägg från en databas för att presentera på sidan, eller inkludera en annan PHP-fil som sköter om att visa kommentarer för inlägget. Sidan som besökaren ser skapas alltså först när besökaren begär den av servern. (Boudrez & Van den Eynde 2002 s.10-11)

*MySQL*

MySQL är en relationsdatabas-hanterare som använder sig av frågespråket SQL (Structured Query Language). MySQL är fri programvara, licensierad under GNU General Public License. MySQL eller någon annan databashanterare används oftast för att lagra data för bloggen, så som inlägg och kommentarer. (MySQL 2011)

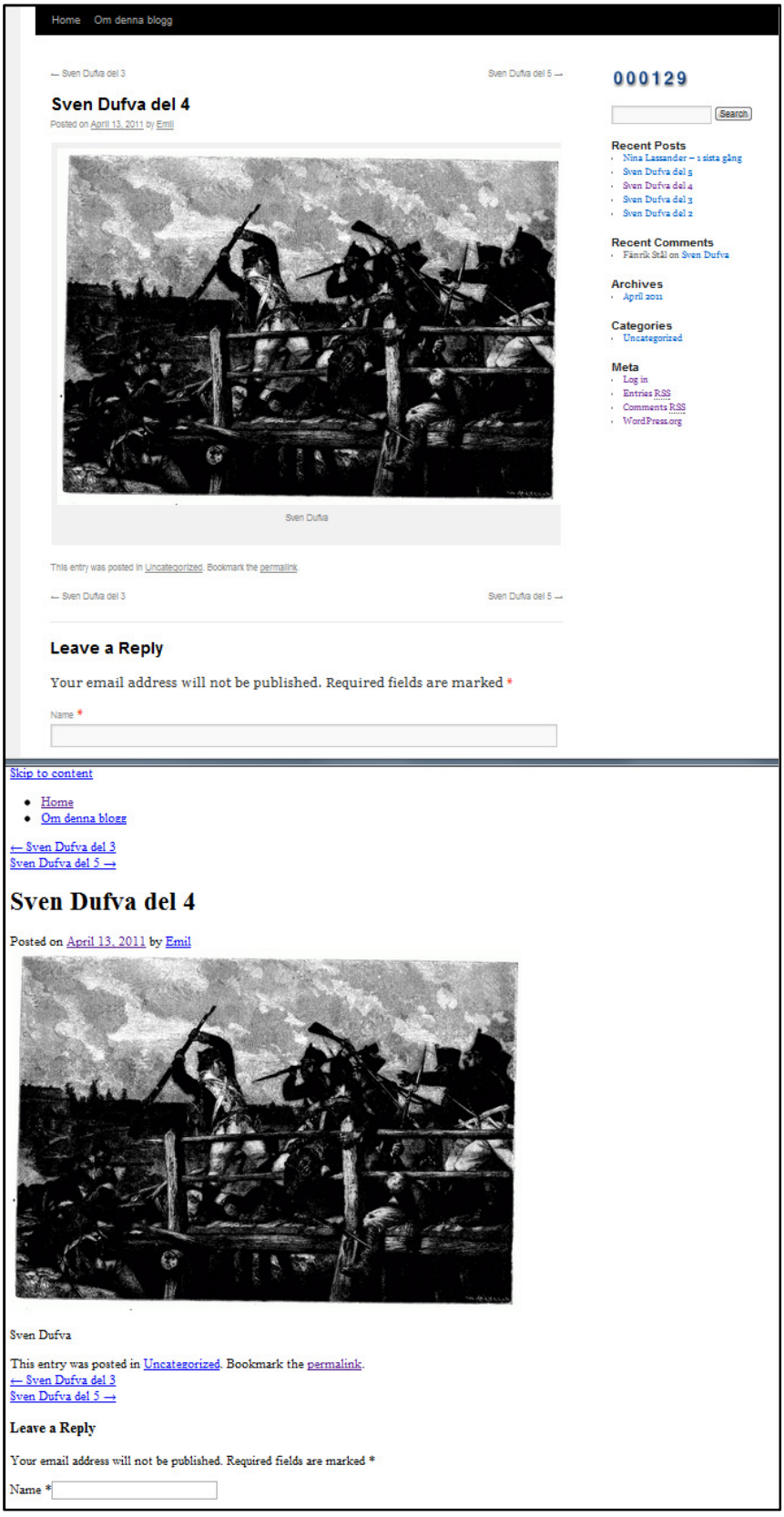
I databasen lagras innehållet i tabeller. En tabell kan till exempel innehålla alla inlägg för bloggen. Varje databaspost eller rad i denna tabell innehåller då ett inlägg. Varje kolumn innehåller då olika egenskaper för inläggen. Dessa kan vara ett unikt ID för

inlägget, rubriken, själva inlägget, datum för publicering, författare och kategorier. Dessa kan variera mellan olika bloggplattformar. Vanligen lagras endast material i textform i databasen, även om det är möjligt att även lagra t.ex. bilder i databasen. Istället lagrar man endast en sökväg till bilden i databasen, och lagrar sedan bilden som en fil på webbservern.

Man kan anta att informationen i en bloggs databas oftast är en sak som man vill bevara. Man måste ändå fundera på om det är ändamålsenligt att spara databasen i befintligt format, eller om man även behöver presentationen, det vill säga bloggstrukturen, för att använda den.

*CSS*

CSS står för ”Cascading Style Sheets”, eller på svenska stilmall. CSS är ett språk som används för att beskriva utseendet på ett dokument. Saker som anges i en stilmall kan vara typsnitt, bakgrundsfärg och placering av olika element på en sida. CSS-filen/filerna är viktiga att samla in om man vill bevara layouten på en blogg som man samlar in. (W3C 2011a)




*Figur 2. Testbloggen med CSS-fil (uppe) och utan (nere).*

*JavaScript*

JavaScript är ett skriptspråk som körs på klienten, det vill säga besökarens dator, i motsats till PHP som körs på servern. JavaScript kan användas för enklare animationer, bildgallerier och mycket annat. (W3C 2011b)

AJAX (Asynchronous JavaScript and XML) är en teknik som är ganska vanlig idag. AJAX kan bland annat användas för att hämta information i bakgrunden utan att ladda om hela sidan. (W3C 2011b)

Eftersom data kan hämtas först när den behövs så kan det hända att man vid en insamling av en blogg missar information. Om insamlingsmetoden man använder fryser all dynamisk funktionalitet på sidan kan det hända att man förlorar viktiga funktioner på bloggen. Man måste därför på förhand undersöka i vilken utsträckning den aktuella bloggen använder JavaScript.

*RSS-flöden*

RSS står för Really Simple Syndication, och är ett format för att syndikera webbinehåll. RSS filer måste följa XML 1.0-specifikationen. RSS-flöden används på bloggar för att ge besökaren möjlighet att ”prenumerera” på nya inlägg och kommentarer. RSS-flöden kan läsas med särskilda program, RSS-aggregatorer, men många nya webbläsare har även stöd för att läsa RSS-flöden direkt. (RSS Advisory Board, 2011)

RSS-flöden kan vara ett möjligt sätt att samla in en blogg, men man får då endast inlägg och kommentarer, och förlorar helt bloggens layout.

## 3.4 Komponenter

Genom att studera olika bloggar så märker man att en blogg kan delas upp i flera olika delar eller komponenter, som utgörs av olika typer av information. Genom att dela upp

bloggen i olika delar kan man göra en bedömning av vilka delar som är viktigast att bevara, och vilken insamlingsteknik som bäst bevarar alla viktiga delar.

*Inlägg*

Den viktigaste komponenten i en blogg är inläggen som den som har hand om bloggen skriver. Dessa består av text och är oftast daterade och kan även vara taggade med olika kategorier. Inlägg kan ibland ändras eller uppdateras, och vid insamling kan man behöva ta detta i beaktande. Man måste besluta om man vill bevara alla versioner av inlägget eller om man är nöjd med den första eller den senaste versionen.

*Media*

Förutom text innehåller inläggen ofta bilder. Även video- och ljudklipp kan förekomma. Ett problem som kan uppstå med media, främst video- och ljudklipp, är att de kan vara lagrade på en annan plats och endast inbäddade i inlägget. I sådana fall kan det vara svårt att få med klippet i insamlingen.

*Kommentarer*

Bloggar erbjuder ofta möjligheten för läsaren att kommentera inläggen. Dessa kommentarer kan ge ny information och leda till diskussioner som kan vara viktiga att bevara. En sak som kan göra insamling av kommentarer svårt är att det kan komma nya kommentarer även till gamla inlägg som redan blivit insamlade.

*Layout*

En blogg har ofta en unik, personlig layout som skiljer den från andra bloggar. Vid insamling måste man besluta om layouten är viktig att bevara. I vissa fall kan den vara viktig för presentationen av det övriga innehållet på en blogg, men det kan vara svårt att bevara layouten och i vissa fall kan bloggen se olika ut i olika webbläsare. Då måste man avgöra vilket utseende som är rätt, vilket kan vara svårt att avgöra.

I vissa fall kan man även föredra att inte bevara layouten, utan istället lagra innehållet på ett mer sökbart sätt.

*Insticksmoduler*

Flera blogg-plattformar, bland annat WordPress, erbjuder möjligheten att utöka funktionaliteten med olika insticksmoduler. Dessa insticksmoduler kan till exempel erbjuda möjligheten att infoga en rss-ström från en annan sida eller möjlighet att skapa bildgallerier. Det finns även många insticksmoduler som hjälper till med administrationen av sidan och inte alls syns för besökaren.

Om det finns aktiva insticksmoduler på en blogg måste man undersöka vilken funktionalitet de har, och avgöra om denna funktionalitet bör bevaras även i den insamlade bloggen.

# 4 KRAVSPECIFIKATION

För att kunna testa och jämföra de olika insamlingsteknikerna är det viktigt att bestämma vilka egenskaper som ska bedömas. För att kunna bestämma dessa egenskaper har jag dels granskat dokument om långtidsbevaring, och dels granskat rapporter om tidigare utförda bloggarkiveringsprojekt. Informationen i detta kapitel används som grund för jämförelsen av de olika insamlingsteknikerna.

## 4.1 Viktiga aspekter vid arkivering

I "Systematic planning for digital preservation: evaluating potential strategies and building preservation plans" (Becker et.al. 2009) presenteras de kriterier som är viktiga att ta i beaktande när man gör upp en bevaringsplan. De kan delas in i fyra olika delar:

*Objektegenskaper*

Objektegenskaperna är de visuella och kontextuella egenskaperna för objektet. Man refererar ofta till dessa som signifikanta egenskaper. Ett vanligt sätt att beskriva dem är genom att använda aspekterna innehåll, sammanhang, struktur, utseende och beteende/funktionalitet. Dessa egenskaper är de som existerar hos det ännu inte insamlade materialet, och som man vill bevara så mycket som möjligt av vid insamlingen. (Becker et.al. 2009)

Speciellt funktionaliteten är något man måste ta i beaktande när man arkiverar en blogg. Det finns många funktioner i en blogg, och man måste bestämma vad man vill göra med dem. Man delar vanligen in dem i tre kategorier: avaktivera, bevara eller frys. Ett exempel på funktionalitet som man vill bevara kan vara menynavigering, medan man ofta vill frysa t.ex. en besöksräknare, så att den inte fortsätter räkna när man studerar den arkiverade sidan i arkivet. (Becker et.al. 2009)

*Arkivobjektegenskaper*

Arkivobjektegenskaperna för objektet är de egenskaper som har att göra med hur det insamlade materialet lagras i arkivet. Till dessa egenskaper hör bland annat metadata, samt filstrukturen på de lagrade materialet. Länkning mellan filer är också en viktig sak. Dessa saker avgör om man kan referera till det arkiverade materialet på ett pålitligt sätt. (Becker et.al. 2009)

*Processegenskaper*

Processegenskaperna beskriver själva arkiveringsprocessen, det vill säga insamling och införande av materialet i arkivet. Det som man ska ta i beaktande här är bland annat användning av insamlingsverktyg, tidsåtgång, samt dokumentation. (Becker et.al. 2009)

Man måste alltså bedöma hur svårt insamlingsverktyget är att använda, om det är tillräckligt snabbt och om det klarar av att dokumentera insamlingsprocessen tillräckligt bra till exempel genom att skriva loggar över insamlingsprocessen.

*Kostnad*

Kostnader för insamling och arkivering måsta också tas i beaktande. Här ska man räkna med alla kostnader som uppstår under det arkiverade objektets livstid, det vill säga inte endast insamlingen och arkiveringen, utan även lagringskostnaden och underhållskostnaden till exempel för en migrering av materialet till en ny plattform. (Becker et.al. 2009)

## 4.2 Tidigare fallstudie

I sin rapport ”Approaches to Archiving Professional Blogs Hosted in the Cloud” beskriver Brian Kelly och Marieke Guy de tillvägagångssätt som använts för att bevara de bloggar som användes inom UKOLN. Det fanns flera olika typer av bloggar, både bloggar av anställda, bloggar för att stöda olika projekt samt bloggar för att stöda och främja olika evenemang. Vissa bloggar var dessutom publicerade på externa tjänster, som wordpress.com. De tekniker som behandlas i rapporten är insamling med sökroboten HTTrack, export av innehåll till XML-fil med inbyggd exportfunktion eller

via RSS-flöden samt skapande av ett pdf-dokument med blogginnehållet. (Kelly & Guy 2010)

I rapporten konstateras att man måste göra upp en plan för arkiveringen, där man ställer upp krav och önskemål på det arkiverade materialet. Därefter kan man välja en lämplig metod för insamlingen som uppfyller de ställda kraven. Man kan alltså inte välja en insamlingsmetod som man sedan alltid använder, och förvänta sig att det insamlade materialet alltid blir bra, utan detta måste avgöras från fall till fall. (Kelly & Guy 2010)

## 4.3 Sammanfattning

Det finns många egenskaper att ta i beaktande när man planerar insamling och arkivering av en blogg. Man måste fundera vad det är som är viktigt att bevara av bloggen, och därefter välja en insamlingsteknik som passar för ändamålet.

# 5 INSAMLINGSTEKNIKER

När man ska samla in en blogg så kan man göra det på olika punkter i distributionen mellan server och klient. Enligt The Preservation of Web Resources Handbook (JISC, 2008 s. 25) så är dessa punkter följande:

- Insamling direkt från server
- Insamling vid klient/webbläsare
- Insamling med ”Crawler”

Om man främst är intresserad av att samla in inlägg och kommentarer kan man lägga till ännu ett alternativ som Kelly & Guy nämner i sin rapport:

- Insamling via RSS-flöden (Kelly & Guy 2010)

Resultatet av insamling från de olika punkterna kan variera, och man måste därför fundera på vad som lämpar sig bäst för varje aktuellt fall. Vissa gånger är alla alternativ inte heller möjliga, om man t.ex. saknar direkt serveråtkomst eller bloggen inte publicerar RSS-flöden.

## 5.1 Insamling direkt i CMS eller server

Om man har direkt åtkomst till servern så kan detta vara ett bra alternativ. Genom att samla in alla filer direkt från severn kan man få en komplett kopia av innehållet på sidan. Nackdelen med denna metod är dock att man även behöver en lämplig serverplattform att köra sidan på för att kunna presentera den i ursprunglig form. Därför kan det även bli aktuellt att arkivera serverkonfiguration, vilket kan göra saken mer komplicerad. (JISC 2008 s. 24)

Ibland kan även bloggmotorn, t.ex. Wordpress, erbjuda möjligheter att göra en backup av hela bloggen. Detta kan även vara ett bra alternativ, men även då måste man ha en fungerande wordpressplattform för att kunna använda dessa backupfiler.

## 5.2 Insamling med webbläsare

Insamling direkt i webbläsaren är en metod som kan lämpa sig för mycket små eller enkla bloggar. Detta kan även vara en bra metod om bloggen har en layout som bara fungerar korrekt i en specifik webbläsare och man behöver bevara detta.

Fördelen med denna metod är att man oftast ser precis vad man sparar. Nackdelen är att det kan vara mycket tidskrävande att manuellt gå igenom en blogg och spara allt material. Man kan även använda sig av olika tillägg till webbläsaren för att delvis automatisera processen. Man kan spara enskilda sidor främst på två sätt. Det ena är att använda webbläsares ”Spara sida som”-funktion, som sparar sidan i html-format med tillhörande filer så som bilder. Det andra sättet är att spara sidan i pdf-format, vilket förenklat kan beskrivas som att man skriver ut sidan på virtuellt papper. Fördelen med denna metod är att man får en fil som inte ändrar utseende beroende på vilket program som används för att granska den. Nackdelen är att man förlorar all interaktivitet på sidan. Länkar fungerar visserligen, men kan inte styras om till arkiverade mål, utan leder till den aktiva sidan, som kanske inte finns kvar. (JISC 2008 s. 22)

För att senare kunna visa material som samlats in med en webbläsare behövs lämplig programvara för visning av materialet. För material som exporterats till pdf-format betyder detta en pdf-läsare. För material som sparats i HTML-format behövs en webbläsare för att visa materialet.

## 5.3 Insamling med robot

Vid insamling med robot (eng. web crawler) använder man en robot som automatiskt följer länkar på en sida och sparar allt material den hittar. Fördelen med denna metod är att den ganska långt går att automatisera. En nackdel är att det kan vara svårt att konfigurera roboten korrekt. Detta kan leda till att man samlar in för mycket material, eller alternativt missar en del av materialet. Vid insamling av mera omfattande bloggar kan insamlingen även ta ganska mycket tid. Man har inte heller samma kontroll på vad som samlas in jämfört med att göra det manuellt i en webbläsare. (JISC 2008 s. 22)

När man använder en robot för insamling så sparas materialet som samlas in som statiska html-sidor. Man kan beskriva det insamlade materialet som en ögonblicksbild av bloggen som den såg ut vid insamlingstillfället.

Insamling med robot lämpar sig bra när större mängder material ska samlas in. Därför används insamling med sökrobot ofta av större webbarkiv, t.ex. Wayback machine.

## 5.4 Insamling av RSS-flöden

Många bloggar erbjuder RSS-flöden av inlägg, och ibland även av kommentarer. Om man främst är ute efter att bevara inläggen kan RSS-flöden vara ett behändigt sätt att samla in dessa. Tyvärr finns även flera begränsningar med denna metod. RSS-flödenas omfattning är ofta begränsad. Till exempel kan RSS-flödet för en blogg innehålla endast de 100 senaste posterna, eller så innehåller flödet bara en kort bit av varje inlägg. En annan nackdel med att samla in bloggar via RSS-flöden är att man även förlorar det mesta av layouten.

# 6 JÄMFÖRELSE AV ALTERNATIV

I detta kapitel testar jag några alternativ för bloggarkivering. Jag presenterar de olika alternativen och kommenterar användningen av dem, samt jämför dem sinsemellan. Slutligen presenterar jag resultatet av jämförelsen.

## 6.1 Jämförelsemetod

För jämförelsen har jag använt två bloggar som jag samlar in. Jag använder de olika insamlingsteknikerna på båda bloggarna och studerar det material som insamlingsverktygen samlar in.

Jag har utgått från de resultat som presenterades i kapitel 4 när jag planerat jämförelsen. Jag har valt ut de kriterier som är viktiga att notera när man väljer ett insamlingsverktyg. Jag har valt att inte göra någon betygssättning eftersom man alltid måste välja verktyg utgående från egna behov. Istället har jag valt att kommentera skillnader mellan verktygen.

Dessa saker kommer att jämföras:

**Objektegenskaper**

*Innehåll* - har allt relevant inehåll samlats in, och vilka typer av innehåll klarar de naktuella tekniken av att samla in(poster, kommentarer, media o.s.v.)

*Struktur* - fungerar navigeringen mellan de olika sidorna i bloggen fortfarande? Hur fungerar länkar till externt material?

*Utseende* - Hur skiljer sig det insamlade materialet utseendemässigt från originalet? Saknas något i utseendet, eller ser något fel ut?

*Funktionalitet* - Hur har funktionella beteenden bevarats i det insamlade materialet? Fungerar fortfarande animationer och menyer gjorde med flash eller javascript? Hur beter sig en besöksräknare?

**Arkivobjektegenskaper**

*Metadata* - vilken metadata finns tillgänglig i det arkiverade materialet?

*Filer* - Hur ser det insamlade materialet ut på filnivå? Är det de ursprungliga filerna eller har det insamlade materialet annat format?

**Processegenskaper**

*Användbarhet* - Finns det speciella krav för att insamlingsverktyget/metoden ska kunna användas?

*Användarvänlighet* - Hur svårt är det att använda insamlingsverktyget?

*Tidsåtgång* - Går insamlingen av materialet att utföra inom en rimlig tid?

*Dokumentation och validering* - Skapar insamlingsverktyget några loggar som går visar hur insamlingen gått, vad som samlats in och vid vilken tidpunkt?

## 6.2 Testobjekt

För att kunna testa de olika insamlingsverktygen så användes två bloggar som testobjekt. För att kunna testa så mycket av funktionaliteten som möjligt hos de olika verktygen så användes en riktig blogg och en testblogg som skapades specifikt för insamlingstester. Testbloggen behövs främst för att man ska kunna undersöka hur mycket och vilka typer av material som verktygen kan samla in.

**Bondbloggen**

Bondbloggen.fi är en blogg som upprätthålls av Yle. På bloggen bloggar fyra olika bönder från olika ställen Österbotten och Nyland. Bloggen körs på WordPress-plattformen.


*Figur 3. Bondbloggen som den såg ut 12.4.2011*

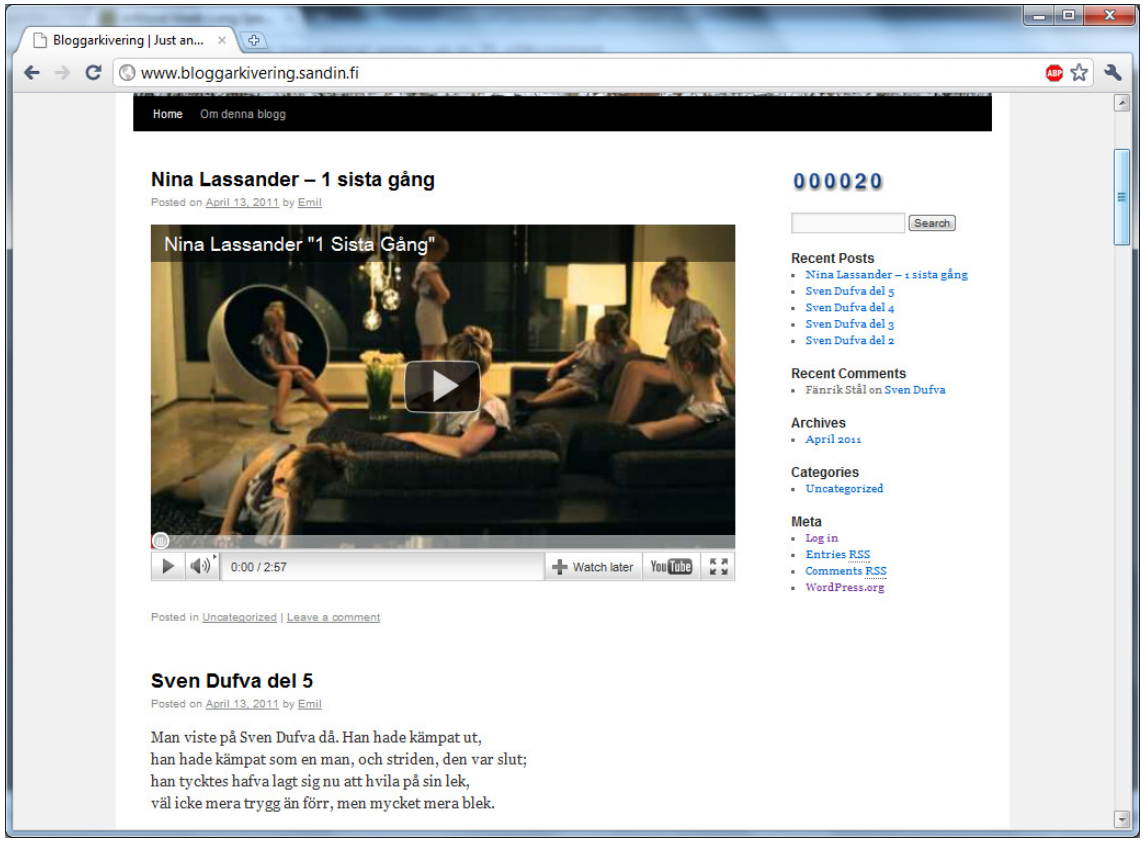
## Testbloggen

Den för insamlingstesterna skapade testbloggen skapades med ett par olika krav. För det första skulle den vara överskådlig, för att snabbt kunna granska resultatet av insamlingen. För det andra skulle den även innehålla typiska element som ofta återfinns

på en blogg. Dessa element är förutom inlägg, kommentarer och bilder även besöksräknare, videoklipp från Youtube, samt filer i pdf och mp3-format.

Testbloggen behövs också för att kunna testa insamling från CMS eller direkt från server. Dessa metoder går endast att göra om man har administratörsrättigheter till CMS eller servern.

Testbloggen gjordes med den mest populära blogplattformen Wordpress. Bloggen skapades på ett webbhotell med webhotellets installationsverktyg. Standardtemat som följer med installationen användes. Sedan lades innehåll till, både text, bilder samt videoklipp från Youtube. Även kommentarer på en del inlägg gjordes.




*Figur 4. Testbloggen som skapades för att kuna testa olika insamlingstekniker.*

## 6.3 Val av insamlingsverktyg

För att kunna göra en praktisk jämförelse mellan de olika teknikerna så har jag valt att testa ett antal olika verktyg som använder olika insamlingstekniker. Det kan visserligen finnas skillnader mellan olika verktyg som använder samma teknik, men syftet med detta test är att jämföra de olika teknikerna, inte själva verktygen.

## 6.4 Insamling i CMS eller direkt från server

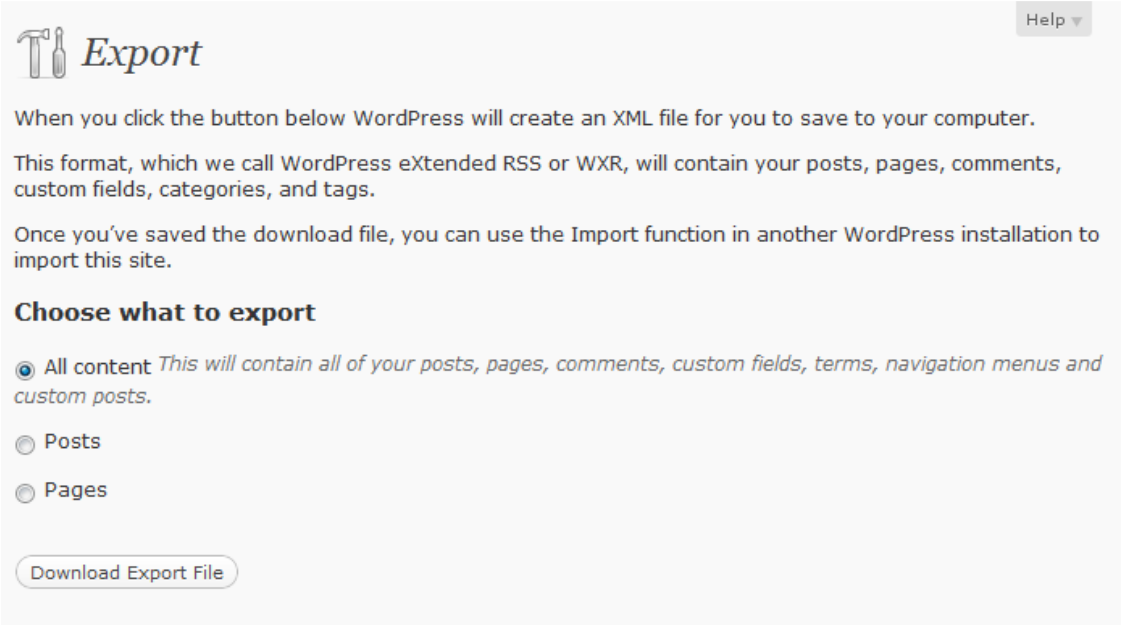
Denna metod går endast att genomföra om man har tillräckliga användarrättigheter till bloggen som ska samlas in. Man måste antingen ha direkt åtkomst till filerna på servern, t.ex. via ftp, eller alternativt ett användarkonto till bloggen med tillräckliga rättigheter.

*Insamling i CMS*

Insamling i CMS testas på testbloggen, som kör på WordPress-plattformen. WordPress har en inbyggd funktion för att exportera blogginnehåll till en XML-fil, som sedan kan importeras i en annan Wordpressinstallation. Denna fil innehåller alla inlägg, kommentarer, sidor taggar och kategorier, men inga mediafiler så som bilder och videoklipp.

*Insamling direkt från server*

Insamling direkt från server görs i två steg. Först kopieras alla filer på webbservern via FTP. I detta test görs det med programmet WinSCP. Sedan används PHPmyAdmin för att exportera databasen.




*Figur 5. Den inbyggda exporteringsfunktionen i WordPress*

## 6.5 Firefox med tillägg

Firefox är den mest använda webbläsaren i dagsläget. Förutom att använda den inbyggads ”Spara som”- funktionen så kan man även använda olika insticksmoduler för att snabba in insamlingsprocessen. Även export till pdf-format testas. Pdf anses vara ett arkivbeständigt format, så även om export till pdf innebär stora förändringar av materialet så kan det vara ett fungerande alternativ om förändringarna visar sig vara acceptabla.

För dessa tester användes Firefox 4 som webbläsare. För pdf-export användes gratisverktyget CutePDF Writer. Detta verktyg fungerar som en virtuell skrivare, som skriver ut till pdf-filer istället för att skriva ut på papper.

## 6.6 HTTrack

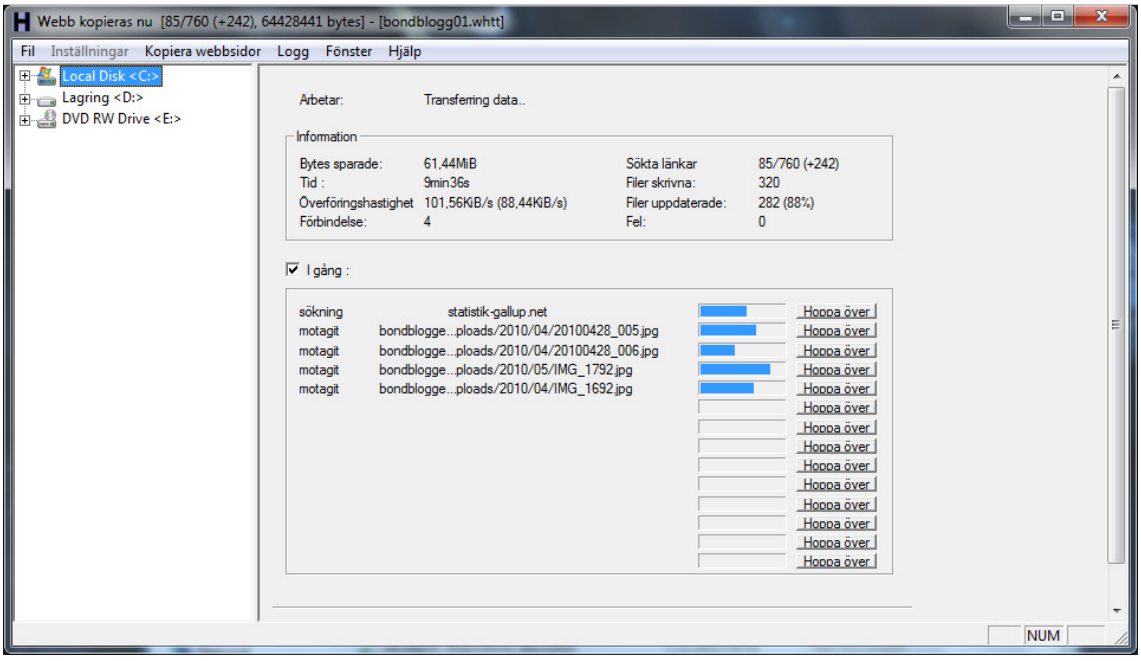
HTTrack är en sökrobot som är släppt under som öppen källkod under GPL. Den finns både för Windows och Linux/Unix/BSD-plattformar. Man kan använda programmet både från kommandotolk och med ett grafiskt användargränssnitt. För detta test

användes Windows-versionen WinHTTrack med grafiskt användargränssnitt på en dator med Windows 7 64-bit Professional.

HTTrack har en guide som hjälper användaren att skapa ett projekt. Man kan ställa in vilka filer som ska inkluderas i eller uteslutas ur insamlingen, samt hur djupt roboten ska gå både internt på webbplatsen och externt på andra webbplatser. Man kan även ställa in hastigheter och max tillåtna samtidiga anslutningar till servern, för att undvika att överbelasta servern vid insamlingen.

Vid insamling av bondbloggen.fi kunde konstateras att det är mycket viktigt att ställa in externt sökdjup rätt. Bondbloggen har många länkar till yles övriga sidor, så även om man har ett lågt värde på externt sökdjup så förlorar man snabbt kontroll på varifrån materialet samlas in.

Vidare kunde också konstateras att det tog ganska länge att samla in bondbloggen, eftersom den innehåller mycket material. Många bilder fanns dessutom länkade i originalupplösning, vilket ökade omfattningen på materialet mycket.




*Figur 6. WinHTTrack under en insamling av bondbloggen.fi*

## 6.7 Insamling av RSS-flöden

Det visade sig vara svårt att hitta ett lämpligt verktyg för att samla in RSS-flöden. Det finns flera så kallade RSS-aggregatorer, som är program för att läsa RSS-flöden. Dessa visade sig dock inte vara särsklit lämpade för arkiveringsbehov. Det vore önskvärt att ett verktyg för insamling av RSS-flöden skulle kunna köra kontinuerligt, och på så sätt arkivera nytt material vartefter det publiceras. När något sådant verktyg inte hittades, gjordes insamlingen av RSS-flöden istället via webbläsaren. I testerna användes webbläsaren Firefox 4, Men även de andra stora webbläsarna klarar av att hantera RSS-flöden.

## 6.8 Jämförelse

Jämförelsen av de olika teknikerna presenteras här för varje egenskap som testades. Alla egenskaper gick inte att testa med alla verktyg.

### 6.8.1 Objektegenskaper

*Innehåll*

HTTrack samlar in största delen av innehållet på en blogg, bara det är rätt inställt. Programmet kan även samla in länkat material från utomstående sidor om så önskas. En sak som dock inte samlades in var inbäddade videor från Youtube.

Insamling direkt från server via ftp samlar in allt material som finns på servern. Dock samlas inget material in som befinner sig någon annanstans, t.ex. bilder som finns på en annan server. Även databasen måste exporteras skilt. I testet användes PHPMyAdmin för att exportera databasen, vilket gav möjlighet att exportera databasen i ett flertal format, däribland XML. Wordpress inbyggda export-funktion gav även den en XML-fil, som sedan kan importeras i en annan WordPress-installation för att återställa en blogg.

Insamling av RSS-flöden samlar in allt material som skickas i RSS-flödet. Det fungerade bra på testbloggen, där det endast fanns ett litet antal inlägg, men på

bondbloggen så samlades endast de senaste inläggen in. Detta alternativ fungerar därför endast om man kontinuerligt samlar in en blogg.

Vid insamling med webbläsare så är det endast möjligt att samla in sidan man för tillfället befinner sig på. Det mesta av innehållet samlas in. Videoklippet från Youtube samlades inte in, men flash-filen för själva videospelaren sparades.

*Struktur*

HTTrack behåller strukturen på sidorna. Vid insamlingen ersätts de ursprungliga länkarna med länkar till insamlat material om det är möjligt. I annat fall behålls de ursprungliga länkarna. Det här innebär till exempel att ett inbäddat klipp från Youtube i en blogg även fungerar i det arkiverade materialet, om klippet fortfarande finns tillgängligt på samma adress. Om det istället rör sig om en insamlad bild så styrs länken om till den lokala kopian, så bilden finns arkiverad.

Vid insamling i servern så får man den filstruktur som finns på servern, samt en exporterad databas. Filerna från servern består främst av PHP-filer samt mediafiler så som bilder och videoklipp. Dessa filer är de som används för att rendera bloggen så som en besökare ser den, men för att kunna göra detta krävs en fungerande webbserver med PHP och MySQL installerat. Innehållet i bloggen finns lagrat i databasfilen.

Material insamlat med webbläsaren behåller samma struktur som originalet, så alla länkar pekar fortfarande på material på servern. Vid export till pdf så förlorades länkarnas funktion helt och hållet.

RSS-flödena som samlas in ser ut precis som de vanliga RSS-flödena. Det här innebär att man får en XML-fil med informationen precis som den ser ut i RSS-flödet.

*Utseende*

HTTRack lyckas bevara utseendet bra. Innehåll som finns lagrat på annan plats än bloggen tas dock inte med, men istället länkas till det ursprungliga materialet, så om det

fins tillgängligt så kommer det med. Material som berörs av det här kan vara bilder, videoklipp och reklambanners.

Vid insamling från server så lagras de ursprungliga filerna, vilket gör att man måste återskapa servermiljön för att kunna rendera en sida med det ursprungliga utseendet. För en WordPress blogg innebär det här en webbserver med PHP och MySQL installerat. Även här gäller att externt materialt inte samlas in.

Insamling med webbläsare till html ger ett utseende som är likadant som det man ser i webbläsaren vid insamlingen. Externt lagrat innehåll som material från Youtube samlas dock inte in. Vid export till pdf så förloras en stor del av layouten. Textformatering och bilder i inläggen finns kvar, men bakgrundsfärger och –bilder försvinner. Youtube-videon ersattes av en grå rektangel.

*Funktionalitet*

HTTRack bevarar funktionalitet som körs på klienten. Det här innebär främst funktionalitet som möjliggörs med javascript och flash. Det här innebär att till exempel menyer och bildspel fungerar bra. Funktionalitet som körs på servern så fryses däremot så som den var vid insamlingen. En besöksräknare stannar till exempel i det läge som den var när sidan samlades in. Annat dynamiskt innehåll som taggmoln ändras heller inte efter insamlingsögonblicket.

Vid insamling i servern så samlar man in allt som behövs för att återskapa all funktionalitet på bloggen, förutsatt att man lyckas skapa en servermiljö som motsvarar den ursrungliga.

Vid insamling i webbläsare så bevaras precis som vid insamling med HTTrack sådan funktionalitet som körs på klienten. Vid export till pdf så försvinner däremot nästan all funktionalitet. Till och med länkarna slutar fungera. Det är möjligt att ett mer avancerat pdf-verktyg skulle kunna bevara mer funktionalitet, men detta är inget som har undersökts närmare i detta arbete.

### 6.8.2 Arkivobjektegenskaper

*Metadata*

Ingen av de testade insamlingsteknikerna samlar in någon större mängd metadata. Största anledningen till detta är att det inte finns så mycket metadata som går att samla in direkt från bloggen, som dessutom går att skilja åt från resten av det insamlade materialet. En del metadata sparas i loggfilerna när man använder HTTrack, men dena berör mest själva insamlingen. Vid insamling från servern eller i CMS så får man en del metadata i XML-filerna med innehållet.

Det vore bra om insamlingsverktyget skulle kunna erbjuda en möjlighet att mata in metadata i samband med insamlingen, men inget av verktygen erbjuder denna möjlighet.

*Filer*

Vid insamling med HTTrack så sparas själva sidorna som html-filer som ser ut som det en webbläsare tar emot. De ursprungliga PHP-filerna som dessa sidor skapas utgående ifrån samlas dock inte in. Övriga filer som till exempel bilder samlas in i sitt ursprungliga format.

Vid insamling direkt från server är det de ursrungliga filerna som bevaras. Om dessa placeras på en server med PHP och databasen importeras så kan man återställa bloggen som den såg ut vid insamlingsögonblicket.

Vid insamling i webbläsare har man flera olika alternativ. Man kan antingen spara sidan i html-format eller exportera till pdf genom att skriva ut på en pdf-skrivare. Om man sparar sidan i html-format så får man en liknande filstruktur som om man skulle använda HTTrack, men endast för en sida åt gången. Om man däremot exporterar sidan till pdf så får man en pdf-fil med sidan som den skulle se ut när man skriver ut den på papper.

Insamling av RSS-flöde resulterar i en xml-fil med samma information som i RSS-flödet.

### 6.8.3 Processegenskaper

*Användbarhet*

HTTrack går att använda på alla websidor som man kommer åt med en vanlig webbläsare. Det betyder att HTTrack och insamling i webbläsare har samma användbarhetsgrad.

För att kunna använda den inbyggda exporteringsfunktionen i Wordpress måste man ha tillgång till admininstrationspanelen i Wordpress. Det här innebär att vem som helst inte kan använda det här på en blogg. Samma sak gäller även för insamling direkt från server, man måste ha direkt åtkomst till filerna på servern, samt åtkomsträttigheter till databasen där bloggens data lagras.

Insamling av RSS-flöden är beroende av hur RSS-flödena är formaterade. Vissa RSS-flöden innehåller till exempel endast början av inläggen, vilket sänker användbarhetsgraden avsevärt. Om man inte själv administerar bloggen som ska samlas in så kan man sällan påverka vilken inormation som skickas i RSS-flödena.

*Användarvänlighet*

HTTrack är ett i mitt tycke relativt enkelt program att använda. Dock finns det många olika inställningar att göra för insamlingen. Detta är bra eftersom man då kan ställa in insamlingen precis som man vill ha den, men vissa inställningar kan vara svåra att förstå, och en felinställning kan påverka resultatet av insamlingen negativt.

Wordpress inbyggda exporteringsfunktion är enkel att använda. Man väljer vad man vill exportera och trycker sedan på en knapp, så får man en XML-fil med det innehåll man vill ha.

Insamling direkt från server kräver lite mera kunskaper. Man måste även göra insamlingen i två steg, först filerna på webbservern och sedan databasen, vilket gör denna metod något krångligare än de andra.

Insamling i webbläsare innebär mycket manuellt arbete. Vid större insamlingsmängder blir arbetet även enformigt. Personen som samlar in måste dessutom själv hålla reda på vad som samlas in, eftersom inga automatiska loggfiler skapas. Export till pdf ger pdf-filer som är lätthanterliga.

Insamling av RSS-flöden via webbläsaren följer i strort sett samma procedur som vanlig insamling av webbsidor via webbläsaren. Med rätt insamlingsverktyg skulle insamling av RSS-flöden däremot kunna automatiseras ganska långt, vilket skulle kunna öka på användarvänligheten.

*Tidsåtgång*

Att samla in en blogg med HTTrack kan ta ganska länge. HTTrack söker efter länkar till andra sidor på bloggen och följer dessa för att hitta mer material att samla in. Det här innebär att HTTrack ofta får besöka samma sida på bloggen flera gånger men via olika länkar. HTTrack sköter dock insamlingen automatiskt när man har startat den, så arbetsåtgången bör ändå inte bli så hög.

Att samla in från servern eller i CMS går betydligt snabbare, eftersom samma innehåll inte laddas ner flera gånger. Däremot kan det krävas mer manuellt arbete ifall man måste exportera databasen och ladda ner mediefiler var för sig. Den tillgängliga bandbredden mellan server och insamlingspunkt är det som kan begränsa tiden som insamlingen tar.

Insamling i webbläsare kräver mycket manuellt arbete. Den här metoden är bara praktisk att använda då det rör sig om en blogg med lite material, eller om man endast är intresserad av att samla in en del av materialet. Om antalet inlägg i bloggen som ska samlas in är mer än något tiotal så går det snabbare att konfigurera en sökrobot att sköta insamlingen än att samla in materialet manuellt.

RSS-flöden innebär ofta en begränsad mängd filer att samla in, ganska vanligt är ett flöde för inlägg och ett för kommentarer. Om endast de senaste inläggen och kommentarerna skickas i RSS-flödena så måste man samla in kontinuerligt, vilket ökar tidsåtgången. Med de rätta verktygen skulle detta dock kunna automatiseras för att minska tidsåtgången.

*Dokumentation och validering*

HTTrack skapar loggfiler på vad som samlats in och vid vilken tidpunkt. Om det finns filer eller sidor som insamlingen misslyckats för så noteras även detta i loggen. Även inställningarna för insamlingen finns lagrade i loggfilen, samt en sammanställning i slutet på hur många filer som har samlats in och hur länge insamlingen tog. För en stor insamling kan loggfilen bli ganska lång och svårläst. Ett insamlingstest som kördes i tre timmar skapade en loggfil på över 26000 rader. Det är därför sällan praktiskt att kontrollera hela loggfilen. Däremot kan man söka efter felmeddelanden och varningar i filen och endast kontrollera dessa, vilket innebär en betydligt mera överkådlig datamängd att kontrollera.

Vid insamling i webbläsare skapas inga loggfiler. Om man exporterar sidorna till pdf kan man däremot lägga till ett sidhuvud med infomation om när exporteringen skedde, samt adressen till den insamlade sidan. Vid insamling av RSS-flöden skapas inte heller några loggar automatiskt.

## 6.9 Resultat

De olika insamlingsteknikerna skiljer sig ganska kraftigt åt både vad gäller insamlingens omfattning, filstruktur och metadata. Även faktorer som tidsåtgång och användarvänlighet skiljer de olika teknikerna åt.

Vid insamling med en robot så går insamlingen automatiskt när alla inställningar väl är gjorda. Insamlingen kan ta lång tid, men man behöver inte övervaka den hela tiden. Loggfiler skapas på vilka filer som samlas in och eventuella fel loggas också.

Webbsidorna sparas som html-filer. Mediefiler som bilder sparas i sitt ursprungliga format. Resultatet av insamlingen går att visa med en vanlig webbläsare.

Vid insamling direkt i server eller CMS så går insamlingen snabbt. Resultatet av insamlingen blir de ursprungliga filerna som fanns på servern, samt en databas exporterad i valbart format. Inga automatiska loggar skapas på insamlingen. Resultatet av insamlingen är svårt att använda direkt. För bästa resultat borde en servermiljö som motsvarar den ursprungliga servern sättas upp och materialet placeras på denna. Resultatet blir då en direkt kopia av den insamlade bloggen.

Insamling i webbläsare är en omständig och tidskrävande metod. Den kan vara smidig om det endast rör sig om ett fårtal enskilda sidor som ska samlas in. Om man sparar sidorna som html så kan man granska det insamlade materialet i en webbläsare. Om man exporterar till pdf så förlorar man en stor del av den ursprungliga layouten. Fördelen blir dock lätthanterliga filer, som kan vara tillräckligt användbara om det endast är texten som är det viktiga.

Insamling av RSS-flöden är omständigt att göra via webbläsaren. Med rätt verktyg skulle denna process konna automatiseras, vilket skulle göra kontinuerlig insamling av en aktiv blogg möjligt. Resultatet av insamling av RSS-flöden är filer i xml-format.

# 7 SAMMANFATTNING

Det finns flera olika tekniker som kan användas för insamling av bloggar. Resulatet av insamling med de olika insamlingsteknikerna skiljer sig åt ganska mycket. Man bör därför överväga följande frågor när man väljer insamlingsteknik:

- Vad vill man bevara?
- Hurudan tillgång har man till bloggen?
- Hur mycket tid kan man lägga ner på arkiveringen?

De vanligaste teknikerna för insamling av bloggar är insamling med robot, insamling direkt i server eller CMS, insamling i webbläsare eller insamling av RSS-flöden. De största skillnaderna mellan de olika teknikerna är hur stor del av innehållet som samlas in, var i distributionskedjan server-klient de samlar in materialet, i vilket format materialet lagras samt hur stor tidsåtgången för insamlingen är.

Det material som är svårast att samla in är inbäddat material, som inte finns lagrat på samma ställe som själva bloggen. Speciellt videoklipp från videotjänster som YouTube ställer till problem.

Alla de testade teknikerna går att använda för insamling av bloggar, men de två tekniker som gör den mest omfattande insamlingen på kortast tid är insamling direkt i server eller CMS samt insamling med robot. Av dessa två alternativ så är insamling med robot den teknik som oftare går att använda, eftersom insamling direkt i server eller CMS förutsätter tillräckliga användarrättigheter, vilket man inte alltid har.

Alla tekniker uppfyller kravet på läsbarhet. Filerna som samlas in är antingen identiska med de som finns på servern eller migrerade till standardiserade format som lämpar sig för långtidsbevaring, så som html, xml och pdf. För att kunna läsa information som samlas in direkt i server eller CMS krävs dock ofta att man även bevarar den ursprungliga miljön för att köra filerna. För att bevara läsbarheten i framtiden kan man då behöva använda sig av emulering.

## 7.1 Diskussion

Långtidsbevaring är ett i mitt tycke utmanande, men samtidigt intressant ämne. Det finns många saker man måste tänka på när man planerar för långtidsbevaring. När det gäller insamling och arkivering av bloggar så var det svårt att hitta tidigare forskning, men en del gick trots allt att hitta.

Den insamlingsteknik som jag skulle välja av de som jag testat är insamling med robot. Det kan visserligen vara lite knepigt att ställa in den rätt, och insamlingen tar tid, men sedan man startat insamlingen sköter den sig helt själv, och det insamlade materialet går att utforska direkt efter insamling med en vanlig webbläsare

Jag tror att intresset för arkivering och långtidsbevaring av digitalt material kommer att öka mycket bland olika organisationer i framtiden, eftersom mängden digitalt material som produceras ökar hela tiden. Jag tror även att privatpersoner mer och mer börjar intressera sig för insamling av sitt eget material på nätet, när de börjar inse hur lätt allt kan gå förlorat.

Inom ett par år tror jag att vi kommer att ha mycket bätter verktyg till vårt förfogande för insamling av material på webben. Samtidigt tror jag att webben kommer att utvecklas och ändras mycket under samma tid, så de ska bli intressant att se om insamlingsteknikerna hänger med i utvecklingen.

## 7.2 Nästa steg

I detta arbete har endast utvalda insamlingsverktyg testats. Det är möjligt att det finns verktyg som lämpar sig bättre för vissa typer av insamling än de som jämförts här. Ett exempel är Heritrix, ett alternativ till HTTrack som dock bedömdes vara för komplicerat för detta arbete. Det skulle även vara möjligt att närmare undersöka de optimala inställningarna för HTTrack eller en annan robot för att samla in till exempel en WordPress-blogg.

Det är svårt att säga hur länge bloggar kommer att användas, och hur de kommer att utvecklas i framtiden. En trend som jag personligen ser är att en stor del av den kommunikation som gjordes via bloggar för ett par år sedan har flyttat över till Facebook. Med detta i åtanke kan det löna sig att studera möjliga insamlingstekniker för material som publiceras där.

# KÄLLOR


Becker, Christoph; Kulovits, Hannes; Guttenbrunner, Mark; Strodl, Stephan; Rauber, Andreas, Hofman, Hans. 2009. *Systematic planning for digital preservation: evaluating potential strategies and building preservation plans*.
Tillgänglig: http://www.ifs.tuwien.ac.at/~becker/pubs/becker-ijdl2009.pdf
25 s. Hämtad 23.9.2010.

Blood, Rebecca. 2000. *Weblogs: a history and perspective*. Publicerad 7.9.2000.
Tillgänglig: http://www.rebeccablood.net/essays/weblog_history.html
Hämtad 21.3.2011.

Boudrez, Filip & Van den Eynde, Sofie. 2002. *DAVID - Archiving websites*
Tillgänglig: http://www.edavid.be/davidproject/teksten/Rapporten/Report5.pdf
95 s. Hämtad 14.5.2011

CCSDS. 2002. *650.0-B-1 Reference Model for an Open Archival Information System (OAIS).* Tillgänglig: http://public.ccsds.org/publications/archive/650x0b1.pdf
148 s. Hämtad 25.10.2010

Digital Preservation Europe. 2011a. *Considerations for the Preservation of Blogs*.
http://www.digitalpreservationeurope.eu/publications/briefs/preservartion_blogs.pdf
Hämtad 10.1.2011

Digital Preservation Europe. 2011b. *What is digital preservation?*
http://www.digitalpreservationeurope.eu/what-is-digital-preservation/
Hämtad 13.1.2011

JISC. 2008. *The Preservation of Web Resources Handbook*. Publicerad 5.11.2008.
Tillgänglig: http://jiscpowr.jiscinvolve.org/wp/files/2008/11/powrhandbookv1.pdf
104 s. Hämtad 13.1.2011

Kelly, B., Guy, M. 2010. *Approaches to archiving professional blogs hosted in the cloud.* Tillgänlig: http://opus.bath.ac.uk/20327/2/ipres-2010-kelly-guy-025.pdf
7 s. Hämtad 25.9.2010.

MySQL. 2011. Wikipedia. Tillgänglig: http://en.wikipedia.org/wiki/MySQL
Hämtad 21.3.2011

RSS Advisory Board. 2011. *RSS 2.0 Specification.*
Tillgänglig: http://www.rssboard.org/rss-specification
Hämtad 22.3.2011

SAOL.2006. *Svenska Akademiens ordlista.*13:e upplagan. Norstedts Akademiska Förlag. 1130 s.

The Economist. 2006. *A survey of new media: It's the links, stupid!* Publicerad 20.4.2006.Tillgänglig: http://www.economist.com/node/6794172?story_id=6794172
Hämtad 11.1.2011

The Nielsen Company. 2011. *BlogPulse*. Tillgänglig: http://www.blogpulse.com
Hämtad 21.3.2011.

W3C. 2011a. *HTML & CSS*. Tillgänglig:
http://www.w3.org/standards/webdesign/htmlcss.html
Hämtad 22.4.2011

W3C. 2011b. *Scripting and Ajax*. Tillgänglig:
http://www.w3.org/standards/webdesign/script
Hämtad 22.4.2011

W3Techs. 2011. *Usage of content management systems for websites*. Tillgänglig:
http://w3techs.com/technologies/overview/content_management/all
Hämtad 21.3.2011.

RSS. 2011. Wikipedia. Tillgänglig: http://en.wikipedia.org/wiki/RSS
Hämtad 16.1.2011

Wordpress.org. 2011a. *About WordPress*. Tillgänglig: http://wordpress.org/about/
Hämtad 21.3.2011.

Wordpress.org. 2011b. *WordPress Download Counter*. Tillgänglig:
http://wordpress.org/download/counter/
Hämtad 21.3.2011.

Wordpress.com. 2011. *WordPress Export*. Tillgänglig:
http://en.support.wordpress.com/export/ Hämtad 28.4.2011