

# **Coma - Sovellusprojekti**

**Hanna Alm**

**Kari Arkko**

**Pentti Nakari**

**Lassi Paavolainen**

**Tietotekniikan Sovellusprojekti  
Sovellussuunnitelma 1.0  
21.4.2004**

**JYVÄSKYLÄN YLIOPISTO  
TIETOTEKNIIKAN LAITOS**



Tekijät:	Hanna Alm, Kari Arkko, Pentti Nakari, Lassi Paavolainen
Työn nimi:	Sovellussuunnitelma
Työ:	Coma-Sovellusprojekti
Sivumäärä:	45
Tilaaaja:	Jyväskylän yliopisto, tietotekniikan laitos
Tiivistelmä:	Tämä dokumentti on Coma-projektin Sovellussuunnitelma, jossa kuvataan toteutettavien sovelluksien arkkitehtuuria ja käyttöliittymää. Dokumenttiin on kuvattu projektiin liittyvien sovelluksien ja tietokannan tekninen toteutus.
Avainsanat:	ECCOMAS2004, käyttöliittymä, proceedings, sekvenssikaavio

### Versiohistoria

Versio	Pvm	Kuvaus	Tekijä
0.1	19.02.2004	Rakenteen hahmottaminen	PN, KA
0.2	16.03.2004	v0.1 Sisältöä	KA, HA
0.3	18.03.2004	v0.2 Tietokanta englanniksi, julkaisun sivurakenne lisätty, sisältöä, hakemistorakenteen generointia	HA, KA, LP
0.4	24.03.2004	v0.3 dokumentin rakenteen korjaus, sisältöä	HA, LP
0.5	26.03.2004	v0.4 kirjoitusasu	HA
0.6	29.03.2004	v0.5 sisältöä luvut 4, 6 ja 7	KA, LP
0.7	02.04.2004	v0.6 oikeinkirjoitusta, kokouksessa esille tulleet korjausehdotukset, tietokannan päivitys, luvun 7 päivitystä	KA, HA
0.8	05.04.2004	v0.7 lisätty template lukuun 7, sisältö lukuun 5.2 ja 8.2	LP, PN, HA, KA
0.9	07.04.2004	v0.8 kokouksessa esille tulleet korjausehdotukset	HA
0.10	13.04.2004	v0.9 viimeistely	KA, LP, HA
1.0	21.04.2004	hyväksytty	HA

Taulukko 1. Versiohistoria.

### Tekijät

HA – Hanna Alm  
KA – Kari Arkko

PN – Pentti Nakari  
LP – Lassi Paavolainen

## Coma-projekti

Projektiryhmä:	Hanna Alm Kari Arkko Pentti Nakari Lassi Paavolainen	haalm@cc.jyu.fi ktarkko@cc.jyu.fi pjen@cc.jyu.fi lopaavol@cc.jyu.fi
Tilaaaja:	Jyväskylän yliopisto, Tietotekniikan laitos Kirsi Majava Pekka Neittaanmäki Tuomo Rossi	majkir@mit.jyu.fi pn@mit.jyu.fi tro@mit.jyu.fi
Vastaava ohjaaja:	Markus Inkeroinen	marink@mit.jyu.fi
Tekninen ohjaaja:	Tero Toivonen	tero.toivonen@cc.jyu.fi
Visuaalinen konsultti:	Jaana Rannanpää	jaelrann@cc.jyu.fi
Käytettävyyskonsultti:	Katja Kaario	kaario@mit.jyu.fi
Työtila:	Huone AgC 223.4, p. 014-260 4966	
Kotisivu:	<a href="http://sovellusprojektit.it.jyu.fi/coma/">http://sovellusprojektit.it.jyu.fi/coma/</a>	
Sähköposti:	coma@korppi.jyu.fi	
Laitteet:	Neljä PC-konetta, joissa kahdessa on käyttöjärjestelmänä Windows XP professional, ja kahdessa Linux.	
Tiivistelmä:	Coma-projekti on Jyväskylän yliopiston tietotekniikan laitoksella kevään 2004 aikana toteutettava Sovellusprojekti. Coma-projekti suunnittelee ja toteuttaa ECCOMAS2004-proceedingsin CD-ROM-muotoisena sekä yleiskäyttöisen sovelluksen konferenssi-proceedingsien tekemiseen.	

## Sisällys

1.	Johdanto.....	1
2.	Termit .....	2
3.	Yleiskatsaus .....	3
3.1.	Tietokanta .....	4
3.2.	Hakemistorakenteen parseroiva skripti .....	4
3.3.	Käyttöliittymä.....	4
3.4.	Kokoomajulkaisun generoiva sovellus.....	4
4.	Tietokanta.....	5
4.1.	Toteutus .....	5
4.2.	Rakenne.....	5
4.2.1.	Taulut .....	6
4.2.2.	Suhteet.....	10
4.3.	Tietokannan luominen .....	12
5.	Hakemistorakenteen parseroiva skripti.....	12
5.1.	Hakemistorakenteesta tietokantaan .....	12
5.2.	Tietokannasta hakemistorakenteeseen.....	13
6.	Käyttöliittymä .....	14
6.1.	Käyttäjien hallinta ja tunnistus .....	14
6.2.	Konferenssin hallinta.....	15
6.2.1.	Aihealueet ja sessiotyypit.....	16
6.2.2.	Sessio .....	17
6.2.3.	Artikkeleiden liittäminen sessioihin.....	18
6.2.4.	Aikataulu .....	19
6.2.5.	Ulkoasu .....	20
6.2.6.	Levyjako.....	20
6.2.7.	Siirrä artikkeli .....	22
7.	Kokoomajulkaisun generoiva sovellus.....	23
7.1.	Hakemistorakenteen generointi.....	23
7.2.	Sivujen generointi.....	24
7.2.1.	Conference Mainpage -sivu .....	25
7.2.2.	General Information -sivu(t) .....	26
7.2.3.	Proceedings Mainpage -sivu.....	26
7.2.4.	Article listing -sivut .....	26
7.2.5.	Author listing -sivut.....	27
7.2.6.	Schedule-sivu .....	27
7.2.7.	Session listing -sivut.....	27
7.2.8.	Date and Time -sivut .....	28
7.2.9.	Session-sivut.....	28

---

7.3.	Skriptit .....	29
7.3.1.	article.pl .....	30
7.3.2.	author.pl .....	31
7.3.3.	daati.pl .....	32
7.3.4.	generate.pl .....	32
7.3.5.	proceed.pl .....	33
7.3.6.	schedule.pl .....	33
7.3.7.	session.pl .....	33
7.3.8.	sessionl.pl .....	34
8.	Ohjelmointi .....	35
8.1.	Perl .....	35
8.2.	HTML ja CSS .....	36
9.	Testaus .....	38
10.	Yhteenveto .....	38
11.	Lähteet .....	39
12.	Liitteet .....	40

---

Taulukko 1. Versiohistoria. ....	I
Taulukko 2. Parametrit skripteille.....	29
Taulu 1. Sessiontype-table. ....	6
Taulu 2. Session-table.....	6
Taulu 3. Topic-table.....	6
Taulu 4. Article-table.....	7
Taulu 5. Users-table. ....	8
Taulu 6. Disk-table.....	8
Taulu 7. Keyword-table.....	8
Taulu 8. Usergroup-table.....	8
Taulu 9. HTTPsessions-table. ....	9
Taulu 10. Schedule-table.....	9
Taulu 11. Author relation-table. ....	9
Taulu 12. Present relation-table.....	9
Taulu 13. Associate relation-table. ....	10
Taulu 14. Organizer relation-table. ....	10
Kuva 1. Projektin tuottama kokonaisuus.....	3
Kuva 2. Tietokannan ER-kaavio. ....	5
Kuva 3. Käyttäjän tunnistus sivu.....	15
Kuva 4. Järjestelijän käyttöliittymän etusivu. ....	15
Kuva 5. Topic-sivu. ....	16
Kuva 6. Session-sivu. ....	17
Kuva 7. Articles in Sessions -sivu.....	18
Kuva 8. Schedule-sivu.....	19
Kuva 9. Layout-sivu. ....	20
Kuva 10. Disks-sivu.....	21
Kuva 11. Diskpartition-sivu. ....	22
Kuva 12. Move Article -sivu. ....	23
Kuva 13. Kokoomajulkaisun generoivan sovelluksen generoima hakemistorakenne. ....	24
Kuva 14. Sovelluksen generoima sivurakenne.....	25
Liite 1. Kokoomajulkaisun generoivan sovelluksen sekvenssikaavio.....	40
Liite 2. main.cgi-pääsivun ja topics_or_sessiontypes.pm-moduulin välinen sekvenssikaavio. ....	41
Liite 3. main.cgi-pääsivun ja session.pm-moduulin välinen sekvenssikaavio. ...	42
Liite 4. main.cgi-pääsivun ja disk.pm-moduulin välinen sekvenssikaavio. ....	43
Liite 5. main.cgi-pääsivun ja partition.pm-moduulin välinen sekvenssikaavio.....	44





## 1. Johdanto

Coma-projekti toteuttaa kevään 2004 aikana Jyväskylässä 24–28.7 järjestettävään ECCOMAS2004 (European Congress on Computational Methods in Applied Sciences and Engineering) -konferenssiin liittyvän kokoomajulkaisun eli proceedingsin CD-ROM-muotoisena. Lisäksi projektin on tarkoitus luoda konferenssisovellus, jota voidaan käyttää tulevaisuudessa konferenssi-proceedingsien tekemisessä. CD-ROM-proceedingsin on valmistuttava viimeistään heinäkuuhun mennessä, jotta se voidaan jakaa konferenssin aikana siihen osallistuville painettavan tiivistelmäjulkaisun välissä.

Projekti on aloitettu tilaajan tarpeesta saada kesän ECCOMAS2004-konferenssin kokoomajulkaisu eli proceedings CD-ROM-muotoisena. Samalla on mietitty voisiko projektin yhteydessä tehdä yleiskäyttöisen sovelluksen, jonka avulla proceedingsit voitaisiin luoda myös tulevaisuudessa järjestettävistä konferensseista ja tapahtumista melko helposti. Tämän takia projektiryhmältä on tilattu ECCOMAS2004-konferenssin CD-ROM-proceedings sekä konferenssisovellus.

Tässä dokumentissa kuvataan projektin vaatimusmäärittelyssä [1] määriteltyjen sovelluksien ominaisuuksien teknisen toteutuksen yksityiskohtia.

Sovellussuunnitelman luvussa 2 käydään läpi projektiin liittyviä oleellisia termejä. Luvussa 3 luodaan yleiskatsaus projektin tuottamiin sovelluksiin. Luku 4 käsittää tietokannan rakenteen ja kenttien tyypit. Projektin tuottamien sovelluksien sisältämiä toiminnallisia osia käsitellään luvuissa 5, 6 ja 7. Projektin käyttämiä ohjelmakoodin kommentointitapoja selvitetään luvussa 8 ja sovelluksien testauksesta mainitaan luvussa 9.

## 2. Termit

Luvussa esitellään projektin kannalta keskeisiä termejä.

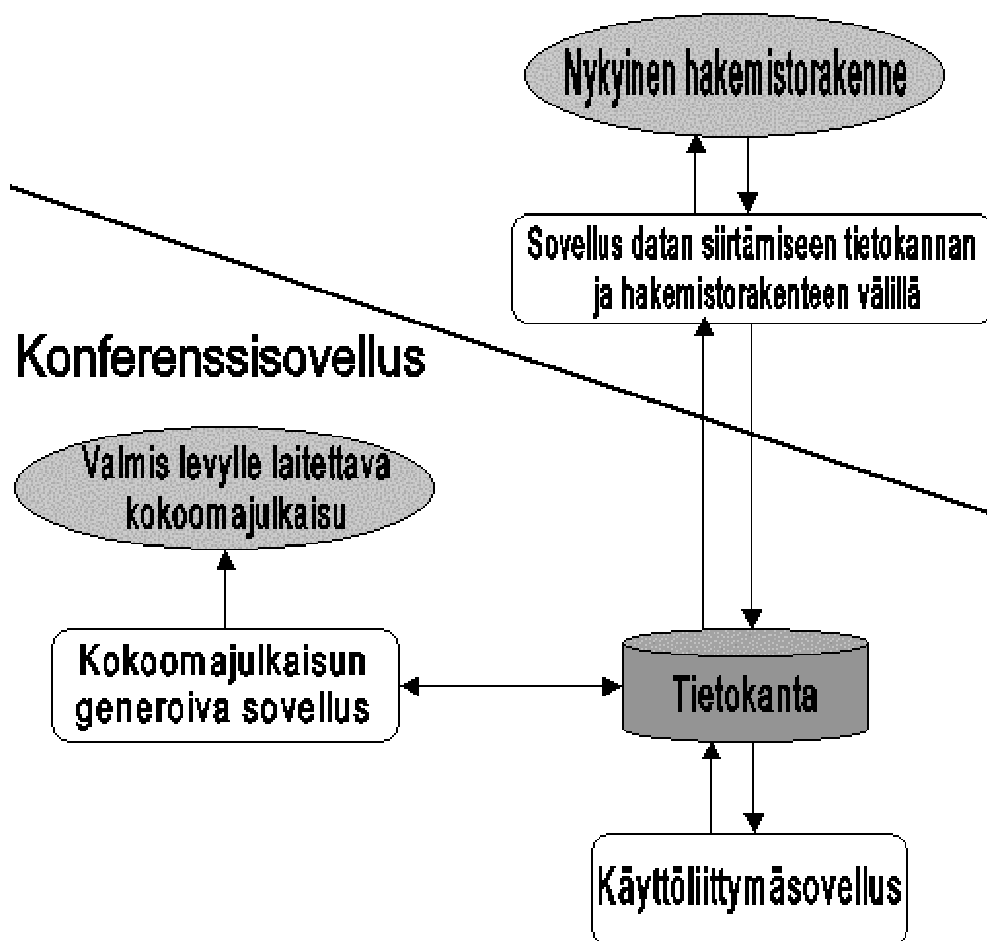
CSS	<i>Cascading Style Sheets</i> , menetelmä ja kieli HTML-dokumenttien ulkoasun määrittelyyn niin, että ulkoasu ja dokumenttien sisältö voidaan pitää erillään toisistaan. [5]
ECCOMAS	<i>European Community on Computational Methods in Applied Sciences</i> .
HTML	<i>HyperText Markup Language</i> , SGML-pohjainen kuvauskieli hypertekstidokumenttien laatimiseen. [4]
PDF	<i>Portable document format</i> , dokumenttimuoto joka on helppo siirtää eri koneympäristöjen välillä. [4]

Konferenssisovellukseen liittyviä termejä

HTTP	<i>HyperText Transfer Protocol</i> , protokolla WWW-asiakkaan ja -palvelimen keskinäiseen viestimiseen. [4]
Perl	<i>Practical Extraction and Report Language</i> , ajettaessa käännettävä vapaa ohjelmointikieli, joka sopii tekstitiedostojen tiedonhakuun, käsittelyyn ja raportointiin. [4]
PostgreSQL SQL	Avoin relaatiotietokantapalvelin. <i>Structured Query Language</i> , standardoitu kieli tietokantojen käsittelyyn. [5]
Template	Sivumalli, WWW-ohjelmoinnin perustekniikka jolla saadaan ulkoasuun yhtenäinen rakenne. [6]

### 3. Yleiskatsaus

Tämä luku kuvaa yleisesti toteutettavia sovelluksia. Projektissa suunnitellaan ja toteutetaan sovelluksien käyttöön tietokanta, ECCOMAS2004-konferenssin hakemistorakenteen ja tietokannan välille tietojen siirto-sovellus, konferenssijulkaisun generoiva sovellus sekä käyttöliittymä konferenssin hallintaan. Projektin tuottama kokonaisuus on kuvattu kuvassa, Kuva 1.



Kuva 1. Projektin tuottama kokonaisuus.

### 3.1. Tietokanta

Tietokanta pitää sisällään kaiken konferenssiin liittyvän tiedon. Tietokantaan on tallennettu tiedot artikkeleista, eri aihealueista, sessiotyypeistä sekä sessioista. Lisäksi tietokannassa on eri käyttäjiin liittyvät tiedot sekä kokoomajulkaisun levyjako.

### 3.2. Hakemistorakenteen parseroiva skripti

Datan siirtäminen nykyisestä hakemistorakenteesta tietokantaan ja takaisin toteutetaan skriptien avulla. Sovellus tarkistaa ja korjaa siirrettävän datan erikoismerkkien ja välilyöntien varalta. Ratkaisu on väliaikainen ja käytetään vain ECCOMAS2004-konferenssin yhteydessä. Tulevien konferenssien data kerätään suoraan tietokantaan.

### 3.3. Käyttöliittymä

Käyttöliittymän kautta hallitaan konferenssin aihealueiden, sessioiden, artikkelien ja muiden oleellisten tietojen syöttöä ja muokkausta. Levyjaon teko ja levyjen luominen tapahtuu käyttöliittymän kautta. Samalla voidaan valita levyille tuleva julkaisun ulkoasu.

Käyttöliittymällä on useita eri käyttäjäryhmiä, jotka omaavat eri oikeuksia. Käyttäjäryhmiä on hahmoteltuna viisi; *Administrator*, *Organizer*, *Contact person*, *Reviewer* ja *Author*. Näistä ainoastaan *Author*-ryhmään kuuluvat eivät käytä käyttöliittymää.

### 3.4. Kokoomajulkaisun generoiva sovellus

Sovellus generoi CD-ROM-levy(i)lle tulevan hakemistorakenteen tietokannan tiedoista. Hakemistorakenteeseen jaotellaan käyttöliittymältä annetun levyjaon mukaisesti artikkelit omiin hakemistoihinsa. Artikkeleiden lisäksi hakemistoon generoituu HTML-sivut, joissa on artikkelien selaukseen tarvittavat tiedot.

## 4. Tietokanta

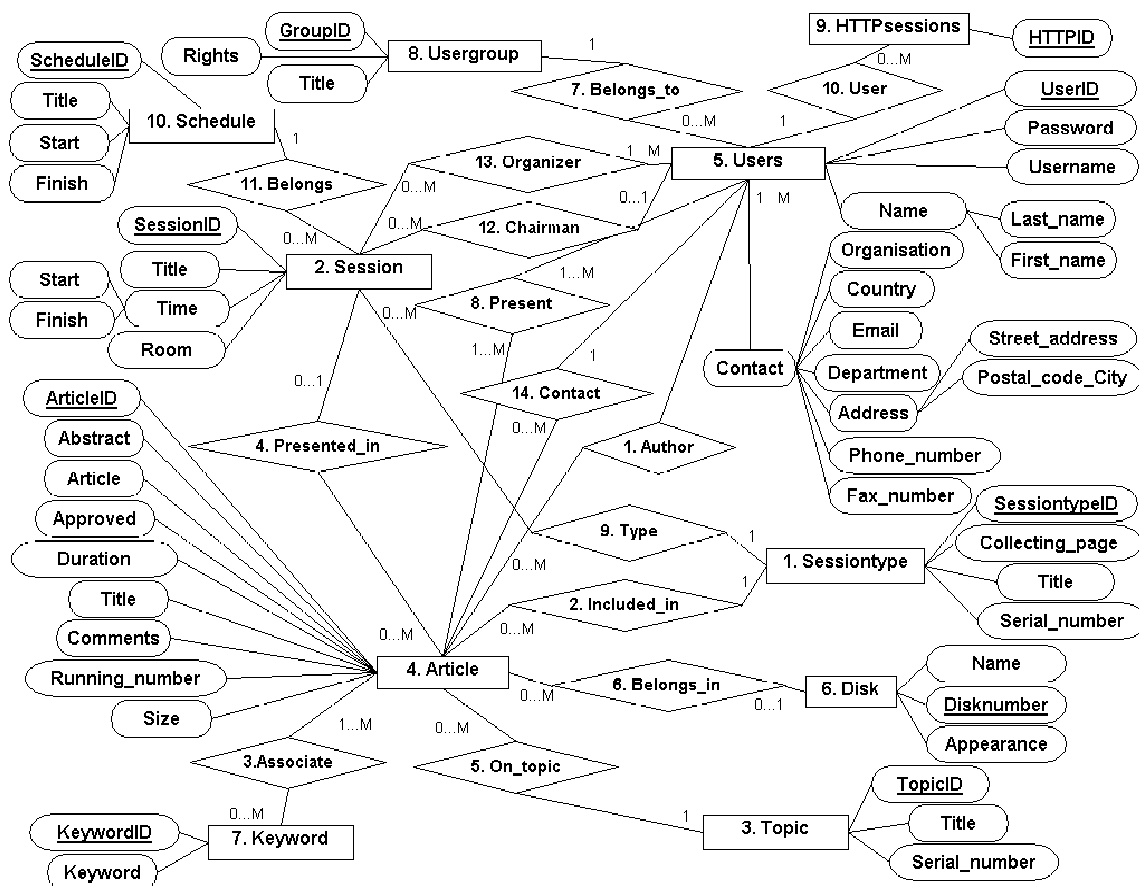
Tässä luvussa kuvataan projektin sovelluksiin liittyvä tietokanta, sen sisältämät taulut, taulujen väliset suhteet, kentät ja kenttien tyypit.

### 4.1. Toteutus

Tietokantapalvelimeksi on valittu PostgreSQL. Se on ominaisuuksiltaan parhaiten projektin käyttöön sopiva avoin tietokantapalvelin. Tietokanta luodaan käyttämällä SQL-lauseita.

### 4.2. Rakenne

Kuvassa, Kuva 2, on esitetty tietokannan ER-kaavio. Kaaviosta selviää tietokannan taulut 1-10, niiden kentät sekä taulujen väliset suhteet.



Kuva 2. Tietokannan ER-kaavio.

#### 4.2.1. Taulut

Tässä luvussa kuvataan tietokannan tauluja.

Property	Datatype	Mandatory	Default	Limits
<u>SessiontypeID</u>	Serial	Yes		Every Sessiontype has to have Unique ID
Title	Text	Yes		
Collecting_page	Text			
Serial_number	Integer	Yes		

Taulu 1. Sessiontype-table.

Property	Datatype	Mandatory	Default	Limits
<u>SessionID</u>	Serial	Yes		Every Session has to have Unique ID
Title	Text	Yes		
Start	Timestamp			
Finish	Timestamp			
Room	Text			
Type	Integer			Only values on Sessiontype-table SessiontypeID-field will do here. (UPDATE CASCADE), (DELETE RESTRICT)
Belongs	Integer			Only values on Schedule-table ScheduleID-field will do here. (UPDATE CASCADE), (DELETE RESTRICT)
Chairman	Integer			Only values on Users-table UserID-field will do here. (UPDATE CASCADE), (DELETE RESTRICT)

Taulu 2. Session-table.

Property	Datatype	Mandatory	Default	Limits
<u>TopicID</u>	Serial	Yes		Every Topic has to have Unique ID
Title	Text	Yes		
Serial_number	Integer	Yes		

Taulu 3. Topic-table.

Property	Datatype	Mandatory	Default	Limits
<u>ArticleID</u>	Integer	Yes		Every Article has to have Unique ID
Abstract	Text			
Title	Text	Yes		
On_topic	Integer	Yes		Only values on Topic-table TopicID-field will do here. (UPDATE CASCADE), (DELETE RESTRICT)
Included_in	Integer	Yes		Only values on Sessiontype-table SessiontypeID-field will do here. (UPDATE CASCADE), (DELETE RESTRICT)
Approved	Boolean		FALSE	
Article	Text			If approved is True (Reference)
Size	Integer			
Duration	Time			
Comments	Text			
Running_number	Integer			Can't be same as any other article on session
Presented_in	Integer			Only values on Session-table SessionID-field will do here. (UPDATE CASCADE), (DELETE RESTRICT)
Belongs_in	Integer			Only values on Disk-table Disknumber-field will do here. (UPDATE CASCADE), (DELETE RESTRICT)
Contact	Integer			Only values on Users-table UserID-field will do here. (UPDATE CASCADE), (DELETE RESTRICT)

Taulu 4. Article-table.

Property	Datatype	Mandatory	Default	Limits
<u>UserID</u>	Integer	Yes		Every User has to have Unique ID
First_name	Text	Yes		
Last_name	Text	Yes		
Organisation	Text	Yes		
Country	Text	Yes		
Email	Text	Yes		
Department	Text			
Phone_number	Text			
Fax_number	Text			
Street_address	Text			
Postal_code_City	Text			
Username	Text		NULL	Every User has to have Unique Username
Password	Text		NULL	
Belongs_to	Integer			Only values on Usergroup-table GroupID-field will do here. (UPDATE CASCADE), (DELETE RESTRICT)

Taulu 5. Users-table.

Property	Datatype	Mandatory	Default	Limits
<u>Disknumber</u>	Serial	Yes		Every Disk has to have Unique Disknumber
Name	Text	Yes		
Appearance	Text	Yes		

Taulu 6. Disk-table.

Property	Datatype	Mandatory	Default	Limits
<u>KeywordID</u>	Serial	Yes		Every Keyword has to have Unique ID
Keyword	Text	Yes		

Taulu 7. Keyword-table.

Property	Datatype	Mandatory	Default	Limits
<u>GroupID</u>	Serial	Yes		Every Usergroup has to have Unique ID
Title	Text	Yes		
Rights	Text	Yes		

Taulu 8. Usergroup-table.



Property	Datatype	Mandatory	Default	Limits
<u>HTTPID</u>	Text	Yes		Every HTTPsession has to have Unique ID
UserID	Integer	Yes		Only values on Users-table UserID-field will do here. (UPDATE CASCADE), (DELETE RESTRICT)

Taulu 9. HTTPsessions-table.

Property	Datatype	Mandatory	Default	Limits
<u>ScheduleID</u>	Serial	Yes		Every part of schedule has to have Unique ID
Title	Text	Yes		
Start	Timestamp	Yes		
Finish	Timestamp	Yes		

Taulu 10. Schedule-table.

Property	Datatype	Mandatory	Default	Limits
<u>UserID</u>	Integer	Yes		Only values on Users-table UserID-field will do here. (UPDATE CASCADE), (DELETE RESTRICT)
<u>Article</u>	Integer	Yes		Only values on Article-table ArticleID-field will do here. (UPDATE CASCADE), (DELETE RESTRICT)

Taulu 11. Author relation-table.

Property	Datatype	Mandatory	Default	Limits
<u>UserID</u>	Integer	Yes		Only values on Users-table UserID-field will do here. (UPDATE CASCADE), (DELETE RESTRICT)
<u>Article</u>	Integer	Yes		Only values on Article-table ArticleID-field will do here. (UPDATE CASCADE), (DELETE RESTRICT)

Taulu 12. Present relation-table.

Property	Datatype	Mandatory	Default	Limits
<u>Keyword</u>	Integer	Yes		Only values on Keyword-table KeywordID-field will do here. (UPDATE CASCADE), (DELETE RESTRICT)
<u>Article</u>	Integer	Yes		Only values on Article-table ArticleID-field will do here. (UPDATE CASCADE), (DELETE RESTRICT)

Taulu 13. Associate relation-table.

Property	Datatype	Mandatory	Default	Limits
<u>UserID</u>	Integer	Yes		Only values on Users-table UserID-field will do here. (UPDATE CASCADE), (DELETE RESTRICT)
<u>Session</u>	Integer	Yes		Only values on Session-table SessionID-field will do here. (UPDATE CASCADE), (DELETE RESTRICT)

Taulu 14. Organizer relation-table.

#### 4.2.2. Suhteet

Tässä luvussa kuvataan tietokannan taulujen välisiä suhteita.

##### 1. Author – Kirjoittaa

Artikkelilla täytyy olla yksi tai useampi käyttäjä, jotka ovat kirjoittaneet artikkelin. Käyttäjän ei tarvitse välttämättä kirjoittaa yhtään artikkelia, tällainen käyttäjä voi olla esimerkiksi ylläpitäjä.

##### 2. Included in – Sisältyy

Artikkeli sisältyy yhteen sessiotyyppiin ja sessiotyyppiin voi sisältyä nolla tai useampi artikkeli.

##### 3. Associate – Liittyy

Artikkeliin liittyy nolla tai useampi avainsana. Avainsana voi liittyä yhteen tai useampaan artikkeliin.

##### 4. Presented in – Esitetään

Artikkeli esitetään yhdessä sessiossa tai sitten ei ollenkaan. Sessiossa voidaan esittää nolla tai useampi artikkeli.

5. On topic – Aihealueeseen

Artikkeli on yhdestä aihealueesta ja yhdessä aihealueessa voi olla nolla tai useampi artikkeli.

6. Belongs in – Levylle

Artikkeli kuuluu yhdelle levylle tai ei millekään levylle. Levylle kuuluu nolla tai useampi artikkeli.

7. Belongs to – Ryhmään

Käyttäjä voi kuulua yhteen käyttäjäryhmään. Ryhmään voi kuulua nolla tai useampia käyttäjiä.

8. Present – Esittää

Käyttäjä voi esittää yhden tai useita artikkeleita. Artikkelin voi esittää yksi tai useampi käyttäjä.

9. Type – Tyyppiä

Sessio voi olla vain yhtä sessiotyyppiä. Yhdestä sessiotyypistä voi olla monta sessiota.

10. User – Käyttäjä

Yhdellä HTTP-istunnolla voi olla vain yksi käyttäjä. Yhdellä käyttäjällä voi olla nolla tai useampi HTTP-istunto.

11. Belongs – Kuuluu

Yksi sessio kuuluu yhteen aikatauluun. Yhteen aikatauluun voi kuulua nolla tai useampi sessio.

12. Chairman – Puheenjohtaja

Sessiolla voi olla nolla tai yksi käyttäjä puheenjohtajana. Käyttäjä voi olla puheenjohtajana nollassa tai useassa sessiossa.

13. Organizer – Järjestelijä

Sessiolla voi olla yksi tai useampi käyttäjä järjestelijänä, toimii vain ulkoistettujen sessioiden osalta. Käyttäjä voi olla järjestelijänä nollassa tai useassa sessiossa.

#### 14. Contact – Yhteyshenkilö

Artikkelilla on vain yksi yhteyshenkilö, mutta henkilö voi olla yhteyshenkilönä monessa artikkelissa.

### 4.3. Tietokannan luominen

Ylläpitäjä luo tietokannan komennolla:

```
createdb -U username databasename.
```

Ylläpitäjä luo kuvan, Kuva 2, mukaisen tietokannan taulut komennolla:

```
psql -U username -f path\comadb.sql databasename.
```

Tiedostossa `comadb.sql` sijaitsevat tietokannan luomiseen käytettävät SQL-lauseet.

Lopuksi ylläpitäjän täytyy luoda tiedosto `conn.ini`, jossa ovat seuraavat tiedot:

```
databasename= # Tietokannan nimi, johon otetaan yhteys käyttöliittymällä.  
host=         # Palvelimen nimi  
port=        # Portti  
user=        # Käyttäjätunnus  
password=    # Salasana
```

Tiedostossa on `tieto=arvo` pareja. Esimerkkitiedosto:

```
databasename=comadb  
host=localhost  
port=1234  
user=coma  
password=f83E3e
```

`Database.pm`-moduuli olettaa että `conn.ini`-tiedosto on samassa hakemistossa.

## 5. Hakemistorakenteen parseeroiva skripti

Luku käsittelee hakemistorakenteen parseeroivan skriptin toimintaa.

### 5.1. Hakemistorakenteesta tietokantaan

Parseri käy läpi artikkelihakemistot sisältävän hakemistorakenteen ja käsittelee kaikki `saved_[0-9]+`-muotoa olevat alihakemistot, jotka eivät ole tyhjiä.

Jos alihakemistosta löytyy `finish.txt`-tiedosto, sen sisältö luetaan. Mikäli artikkeli on yhteyshenkilön hyväksymä (tiedoston sisältö `accepted=1`), luetaan myös `topic.txt`-tiedosto jos se on olemassa. Jos ensimmäinen `author`-tiedosto (`author_1.txt`) löytyy, etsitään silmukalla myös mahdolliset muut `author`-tiedostot (`author_n.txt` saakka). Puuttuvista tiedostoista tulostetaan virheilmoitus ajon aikana terminaaliin.

Artikkelin ja kirjoittajien ID-numerot generoidaan parserilla, lopuksi ID-numeroiden automaattiset sekvenssit asetetaan alkamaan seuraavasta numerosta, jolloin jälkikäteen lisätyt artikkelit ja kirjoittajat saavat ID-numeron automaattisesti.

Tiedoissa olevat erikoismerkit, isoilla kirjaimilla kirjoitetut nimet, TeX-koodit ja muut poikkeamat korjataan. Tämän jälkeen tehdään vielä muita tarkistuksia ja lisätään arvot tietokantaan.

## 5.2. Tietokannasta hakemistorakenteeseen

Tietokannassa olevista tiedoista luodaan hakemistorakenne levyllä käymällä tietokanta järjestyksessä artikkeleittain läpi ja luomalla jokaiselle artikkelille hakemisto. Hakemiston nimi on muotoa `saved_1234` ja nimi saadaan tietokannan kentästä `Article.Article`. Kaikki tietokannassa mukana olevat artikkelit ovat hyväksytyjä, joten tähän hakemistoon luodaan tekstitiedosto `finish.txt` sisältönä yhdellä rivillä "`accepted=1`".

Artikkelin esitystiedot kirjoitetaan tekstitiedostoon `topic.txt`. Tiedosto sisältää rivit, joiden arvot saadaan artikkelin tietojen mukaisesti kentistä `Article.On_topic`, `Article.Included_in` ja `Article.Title`, avaimia vastaavat arvot saadaan tauluista `Sessiontype` ja `Topics`. Alla esimerkki:

```
presentationmode=contributedpaper
contributedpapertopic=1. Computational Solid and Structural
Mechanics
abstracttitle=Otsikko
```

Artikkelin esittäjien ja muiden kirjoittajien tiedot löytyvät tietokannasta `Article.Write` ja `Article.Present`-tauluista artikkelin `ArticleID:n` avulla. Kirjoittajat tallennetaan `UserID:n` mukaisessa järjestyksessä tiedostoihin `author_1.txt` ... `author_n.txt`. Mikäli kyseessä on esittäjä, kirjoitetaan `author_1.txt`-tiedoston ensimmäinen rivi muodossa "`presentingauthor_1=yes`", muussa tapauksessa "`presentingauthor_1=no`". Tiedot tallentuvat seuraavassa muodossa:

```
presentingauthor_1=yes  
lastname_1=Meikäläinen  
firstname_1=Matti  
institution_1=University of Technology of Mattila  
country_1=France  
email_1=matti.meika@dd.fi  
department_1=  
street_1=12 Meikäläisen katu 3 c 1234  
postcode_1=1234 Mattila  
phonenumber_1=1234567890  
faxnumber_1=11234567890
```

Mikäli tieto puuttuu tietokannasta, kirjoitetaan pelkkä avainsana. Avainsanojen perässä oleva numero vastaa tiedostonimen perässä olevaa numeroa (author\_1.txt tiedostossa presentingauthor\_1 jne.).

## 6. Käyttöliittymä

Luvussa käsitellään käyttöliittymän toteutusta käyttäjien ja konferenssin hallinnan osalta.

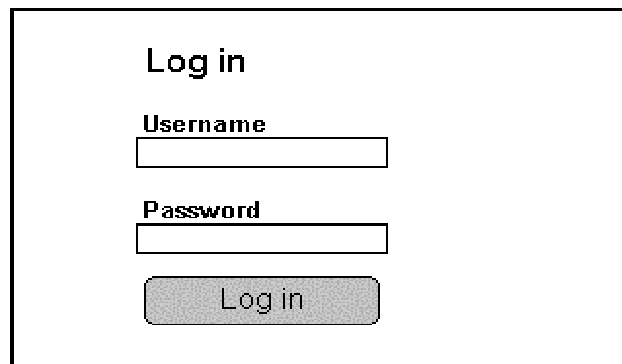
### 6.1. Käyttäjien hallinta ja tunnistus

Artikkelin yhteyshenkilö, *Contact person*, syöttää käyttäjätunnuksen ja sähköpostiosoitteen, jonka jälkeen yhteyshenkilölle lähetetään salasana sovellukseen. Muille käyttäjille salasanan lähettää järjestelijä, *Organizer*.

Käyttäjryhmän perusteella valitaan mitkä toiminnot käyttäjälle annetaan. Käyttäjryhmiä ovat *Administrator*, *Organizer*, *Contact person*, *Reviewer* ja *Author*. Jokaiselle käyttäjryhmälle on oma käyttöliittymänsä.

Käyttäjän tunnistus sivulla, Kuva 3, käyttäjä antaa tunnuksensa ja salasanasensa. Painamalla *Log in* -painiketta käyttäjä pääsee käyttöliittymän etusivulle edellyttäen, että käyttäjä tunnistetaan tunnuksen ja salasanan perusteella.

Käyttäjän selaimelle lähetetään eväste, jolla käyttäjä tunnistetaan myöhemmin saman istunnon aikana. Jos käyttäjää ei tunnisteta, niin palautetaan virheestä viesti ja kysytään uudelleen käyttäjä tunnusta ja salasanaa. Käyttäjän selaimen tulee tukea evästeitä.



**Log in**

**Username**

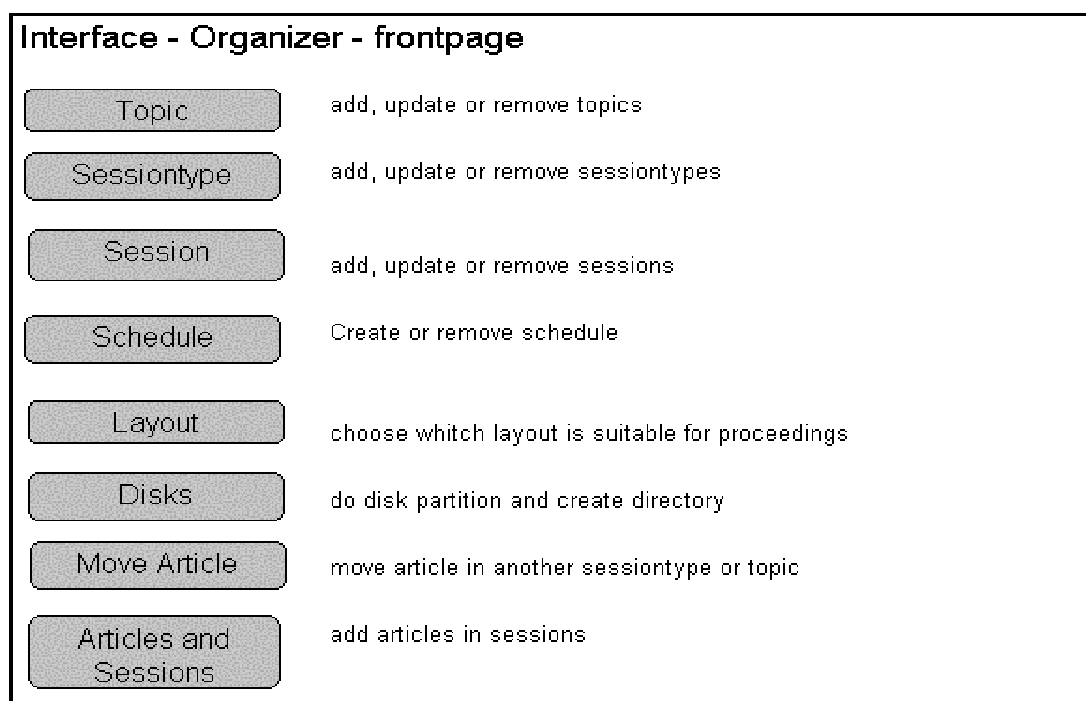
**Password**

Kuva 3. Käyttäjän tunnistus sivu.

## 6.2. Konferenssin hallinta

Järjestelijän käyttöliittymän etusivulta, Kuva 4, löytyvät linkit *Topic*, *Sessiontype*, *Session*, *Schedule*, *Layout*, *Disks*, *Move Article* ja *Articles and Sessions*.

Järjestelijän käyttöliittymä `main.cgi` keskustelelee moduuleiden kanssa, jotka palauttavat HTML-koodia. `main.cgi` luo perusrakenteen, kuten käyttöliittymän menun ja generoi moduuleiden avulla käyttäjän haluamat tiedot näkyviin. `main.cgi` tarkistaa myös käyttäjän evästeiden avulla.



**Interface - Organizer - frontpage**

<input type="button" value="Topic"/>	add, update or remove topics
<input type="button" value="Sessiontype"/>	add, update or remove sessiontypes
<input type="button" value="Session"/>	add, update or remove sessions
<input type="button" value="Schedule"/>	Create or remove schedule
<input type="button" value="Layout"/>	choose which layout is suitable for proceedings
<input type="button" value="Disks"/>	do disk partition and create directory
<input type="button" value="Move Article"/>	move article in another sessiontype or topic
<input type="button" value="Articles and Sessions"/>	add articles in sessions

Kuva 4. Järjestelijän käyttöliittymän etusivu.

### 6.2.1. Aihealueet ja sessiotyypit

Sekä *Topic*- että *Sessiontype*-sivulta löytyvät linkit *Alphabetic*, *Serial number*, *Add*, *Update* ja *Remove*. Molemmat sivut ovat saman näköisiä ja toimivat samoin, *Topic*-sivu on esitettyä alla, Kuva 5 .

*Topics\_or\_sessiontypes.pm*-moduulille viedään parametrina tietoa siitä onko kyseessä aihealue vai sessiotyyppi. Aihealueen tapauksessa moduuli hakee tietokannasta kaikki aihealueet. Moduuli generoi kuvan, Kuva 5, mukaisen HTML-koodin.

*Alphabetic*- ja *Serial number* -painikkeilla voidaan valita listautuuko aihealueet aakkosjärjestyksen vai järjestysnumeron mukaisesti. Aihealueen järjestysnumero listautuu aihealueen nimen eteen alavetovalikossa. Valikosta voi tarvittaessa myös muuttaa järjestysnumeroa. Kahdella aihealueella ei voi olla samaa järjestysnumeroa.

Tekstikenttään voidaan syöttää uuden aihealueen nimi ja *Add*-painikella se lisätään tietokantaan. Lisäys tulee heti näkyviin. Lisättäessä tarkistetaan onko kyseinen aihealue olemassa, jos aihealue on olemassa, huomautetaan käyttäjää ja evätään lisäys. Sovellus antaa automaattisesti lisäyksen yhteydessä aihealueelle seuraavan vapaana olevan järjestysnumeron.

**Topic**

Order Topics by:

	<b>Serial number</b>	<b>Name</b>	<b>Remove</b>
<input type="radio"/>	1 ▼	Topic1	<input type="checkbox"/>
<input checked="" type="radio"/>	2 ▼	Topic2	<input type="checkbox"/>
<input type="radio"/>	3 ▼	Topic3	<input type="checkbox"/>
<input type="radio"/>	4 ▼	Topic4	<input type="checkbox"/>

**Name**

Kuva 5. Topic-sivu.

Radionappulalla voidaan valita muutettava aihealue ja tekstikenttään syötetään teksti, jolla korvataan aikaisempi aihealueen nimi. Muokkaus tulee voimaan *Update*-painikkeella. Tarkistetaan onko kyseiseen aihealueeseen viitteitä, jos on



niin huomautetaan käyttäjää ja kysytään haluaako käyttäjä että muutos tulee voimaan.

Valintanappulalla valitaan poistettavat aihealueet ja *Remove*-painikkeella valitut aihealueet poistetaan. Tarkistetaan onko poistettaviin aihealueisiin viitteitä, jos on niin huomautetaan käyttäjää ja evätään poisto. *main.cgi*-pääsivun ja *Topics\_or\_sessiontypes.pm*-moduulin välinen sekvenssikaavio liitteenä, Liite 2.

### 6.2.2. Sessio

*Session*-sivulle, Kuva 6, listautuu jo luodut sessiot, niiden esityspaikka, päivä, alku- ja loppuaika sekä sessiotyyppi. Lisäksi sivulla on *Add*-, *Update*- ja *Remove*-painikkeet.

*Session.pm*-moduuli hakee tietokannasta sessiot ja niiden tiedot sekä kaikki sessiotyypit. Moduuli generoi kuvan, Kuva 6, mukaisen HTML-koodin ja palauttaa sen. Moduuli osaa myös lisätä, päivittää ja poistaa sessioita.

Sessio voidaan lisätä syöttämällä sivun kenttiin tarvittavat tiedot ja painamalla *Add*-painiketta. Sessiota ei voi lisätä jos jollakin toisella sessiolla on sama esityspaikka, päivä ja aika. Pakolliset tiedot ovat session nimi ja sessiotyyppi. Moduuli tarkistaa onko annetut tiedot oikeassa muodossa. Jos tiedot eivät ole oikeassa muodossa, käyttäjää huomautetaan ja ilmoitetaan oikea muoto. Muut tiedot tallennetaan tietokantaan.

Session							
	Name	Place	Date	Starts	Ends	Sessiontype	Remove
<input type="radio"/>	Session1	Aud1	24.7.2004	11:30	12:30	Minisymposium	<input type="checkbox"/>
<input checked="" type="radio"/>	Session2	Aud2	24.7.2004	11:30	12:30	Plenary	<input type="checkbox"/>
<input type="radio"/>	Session3	Aud2	24.7.2004	12:30	13:30	Plenary	<input type="checkbox"/>
<input type="radio"/>	Session4	Aud1	25.7.2004	11:30	12:30	Minisymposium	<input type="checkbox"/>

**Name**

**Place**  ▾

**Date**  ▾

**Starts**  ▾

**Ends**  ▾

**Sessiontype**  ▾

Kuva 6. Session-sivu.

Session muokkaus onnistuu valitsemalla muokattava sessio radiovalintanappulalla sivun listauksesta ja täyttämällä kenttien tiedot uudelleen. Tämän jälkeen tiedot päivittyy *Update*-painikkeella.

Session voi poistaa valitsemalla poistettava sessio valintanappulalla sivun listauksesta ja painamalla *Remove*-painiketta. Tarkistetaan onko poistettavaan sessioon viitteitä, jos on niin huomautetaan käyttäjää ja evätään poisto. `main.cgi`-pääsivun ja `Session.pm`-moduulin välinen sekvenssikaavio liitteenä, Liite 3.

### 6.2.3. Artikkeleiden liittäminen sessioihin

*Articles in Sessions* -sivulta, Kuva 7, voi valita sessiotyypin nimensä mukaisella painikkeella.

Sivun alalaitaan listautuu valittuun sessiotyyppiin kuuluvat artikkelit ja sessiot.

Artikkelin liittäminen tiettyyn sessioon tapahtuu alasvetovalikolla ja jokaisen artikkelin pitää kuulua johonkin sessioon. Sivun päivittyy *Update*-painikkeella.

Articles in Sessions		
<input type="button" value="Minisymposium"/>	<input type="button" value="Plenary"/>	<input type="button" value="Postery"/>
<input type="button" value="Keynote"/>		
Postery		
Article1	Session <input type="text" value=""/>	
Article2	Session <input type="text" value=""/>	
Article3	Session <input type="text" value=""/>	
Article4	Session <input type="text" value=""/>	
<input type="button" value="Update"/>		

Kuva 7. *Articles in Sessions* -sivu.

`Articles_in_sessions.pm`-moduulille viedään parametrina tieto siitä mihin sessiotyyppiin kuuluvia sessioita ja artikkeleita halutaan käsitellä. Moduuli hakee

tietokannasta valitun tyyppiset artikkelit ja sessiot. Moduuli generoi ja palauttaa kuvan, Kuva 7, mukaisen HTML-koodin. Moduuli osaa myös päivittää artikkelien ja sessioiden välisiä linkkejä.

#### 6.2.4. Aikataulu

*Schedule*-sivulle, Kuva 8, hahmottuu sen hetkinen aikataulu. Käyttäjä voi lisätä aikatauluun tauluja syöttämällä nimen, alkamisajan sekä loppumisajan ja painamalla *Add*-painiketta.

`Schedule.pm` tarkistaa ettei aikatauluun tule päällekkäisyyksiä. Päällekkäisyyksistä ilmoitetaan käyttäjälle ja evätään lisäys.

**Schedule**

	Monday 26 July 2004	Tuesday 27 July 2004	Wednesday 28 July 2004	Thursday 29 July 2004	Friday 30 July 2004
8:00		BOX2			
8:30					
9:00	BOX1			BOX5	
9:30					
10:00					
10:30		Lunch			
11:00					
11:30		BOX3			
12:00	Lunch				
12:30					BOX6
13:00					
13:30					
14:00			BOX4		
14:30					
15:00					
15:30					
16:00					

**Name**

**Start**

**Finish**

Kuva 8. Schedule-sivu.

### 6.2.5. Ulkoasu

*Layout*-sivulta, Kuva 9, avautuu WWW-lomake jolla määritellään ulkoasu. Sovelluksessa tulee olemaan valmiina muutamia vaihtoehtoja, joissa on mahdollisuus syöttää ulkoasuun konferenssin logo. Tyylin voi valita alasvetovalikolla ja logon saa haettua *Browse*-painikkeella. Tiedot päivittyvät *Update*-painikkeella, jonka jälkeen halutessaan tyyliä voi muokata ja tallentaa uudella nimellä. Ulkoasun esikatselu, *Preview*-painike, generoi levyille tulevat sivustot, joista nähdään levyjen ulkoasu sekä sivustojen viemä levytila. Tyylin muokkaamisessa auttavia vinkkejä löytyy *Help with CSS* -painikkeella avautuvalla sivulta.

Sivujen rakenne tehdään template-tekniikalla ja ulkoasu määritellään CSS-tyylitiedostolla.

**Layout**

**Upload CSS style**    **Style:**  ▼

**Logo:**    

**Edit Style**

```
/* esimerkkicss.txt - CSS example
 * 2004-03-30:18:26:00, pjen@cc.jyu.fi
 * Valid CSS version 2 - W3C CSS Validation Service
 */

/***** Edit this to change the main title font *****/

font.title, h1 {
    font-family: Verdana, Arial, Helvetica, Sans-Serif;
    font-size: 22px;
    font-weight: bold;
}
```

**Name for new style:**

Kuva 9. Layout-sivu.

### 6.2.6. Levyjako

*Disks*-sivulla, Kuva 10, on tiedot levyjaoista ja valintanappulat sekä *Add disk*-, *Remove disk*- ja *Update disk* -painikkeet. Lisäksi sivulla on *Diskpartition*- ja *Create disks* -painikkeet.

`Disks.pm`-moduuli hakee tietokannasta kaikkien levyjen tiedot ja laskee niille määrättyjen artikkeleiden määrän, artikkeleiden vaatiman levytilan, sivustojen vaatiman levytilan sekä yhteenlasketun tilan. Moduuli generoi kuvan, Kuva 10, mukaisen HTML-koodin ja palauttaa sen.

*Remove disk* -painike poistaa valitun levyn, valinta tapahtuu checkbox-nappulalla. Tarkistetaan onko levyyn viittauksia, jos on niin siitä ilmoitetaan käyttäjälle ja evätään poisto.

*Add disk* -painike lisää tekstikenttään syötetyn levyn.

*Update disk* -painike muokkaa radionappulalla valitun levyn nimen tekstikenttään annetun tekstin mukaiseksi.

*Create Disks* -painike luo levyjen hakemistorakenteen tietokannan tietojen perusteella. Liitteenä, Liite 4, `main.cgi`-pääsivun ja `Disks.pm`-moduulin välinen sekvenssikaavio.

				Remove
<input type="radio"/>	Disk1	includes 2 article	55 MB	<input type="checkbox"/>
<input checked="" type="radio"/>	Disk2	includes 30 article	300 MB	<input type="checkbox"/>
<input type="radio"/>	Disk3	includes 0 article	20 MB	<input type="checkbox"/>
<input type="radio"/>	Disk4	includes 15 article	100 MB	<input type="checkbox"/>
<input type="radio"/>	Disk5	includes 0 article	20 MB	<input type="checkbox"/>
<input type="radio"/>	Disk6	includes 22 article	200 MB	<input type="checkbox"/>

**Name**

Kuva 10. Disks-sivu

*Diskpartition*-sivulla, Kuva 11, on listaus artikkeleista tietoineen ja valintanappula, jolla valitaan levyille tulevat artikkelit. Levyjaon nopeuttamiseksi käyttäjä voi valita levyille tulevia artikkeleita suurempina kokonaisuuksina, aihealueen, session tai sessiotyyppin mukaan. Sivun listauksen voi päivittää *Article*-, *Sessiontype*-, *Topic*- tai *Session*-painikkeella. Artikkeleista näytettäviä tietoja ovat nimi, sessiotyyppi,

aihealue, sessio ja artikkelin koko. *Update*-painikkeella päivitetään levyjako ja *Remove*-painikkeella poistetaan levyjako.

`Partition.pm`-moduulille viedään parametrina tieto miten artikkelit lajitellaan. Moduuli hakee tietokannasta kaikki levyt ja tekee niille määrätyn levyjaon. Moduuli generoi kuvan, Kuva 11, mukaisen HTML-koodin ja palauttaa sen, jos artikkelit lajitellaan aakkosjärjetyksessä. *Update*-painikkeella `Partition.pm`-moduuli lisää tietokantaan muutetun levyjaon ja generoi sen jälkeen kuvan, Kuva 11, mukaisen HTML-koodin. Jos lajittelu on artikkeleittain, niin listataan kaikki artikkelit aakkosjärjestyksessä. Jos lajittelu on valittu sessioittain, aihealueittain tai sessiotyypeittäin, niin käyttäjä voi lisätä kaikki valittuun lajittelutyyppiin kuuluvat artikkelit yhdellä kertaa haluamalleen levyille. `main.cgi`-pääsivun ja `Partition.pm`-moduulin välinen sekvenssikaavio liitteenä, Liite 5.

Diskpartition							
Article		Sessiontype		Topic		Session	
1	2	3	Article1	Sessiontype1	Topic1	Session2	50KB
<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	Article2	Sessiontype1	Topic1	Session2	50KB
<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	Article3	Sessiontype1	Topic1	Session2	50KB
<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	Article4	Sessiontype2	Topic1	Session2	50KB
<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	Article5	Sessiontype2	Topic1	Session2	50KB
<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	Article6	Sessiontype2	Topic1	Session2	50KB

Kuva 11. Diskpartition-sivu.

### 6.2.7. Siirrä artikkeli

*Move Article* -sivulla, Kuva 12, valitaan siirretäänkö artikkeli sessiotyypistä tai aihealueesta toiseen.

Sivulla on listaus artikkeleista ja perässä alavetovalikot, joilla voi valita mihin aihealueeseen tai sessiotyyppiin artikkeli kuuluu. Sivulla voi valita artikkeleiden listauksen joko aakkosjärjestyksen, sessiotyyppin tai aihealueen mukaan. Tekemänsä muokkauksen saa päivitettyä *Update*-painikkeella.

**Move Article**  
  
Order Articles by:     
  

	<b>Sessiontype</b>	<b>Topic</b>
Article1	<input type="text"/>	<input type="text"/>
Article2	<input type="text"/>	<input type="text"/>
Article3	<input type="text"/>	<input type="text"/>
Article4	<input type="text"/>	<input type="text"/>

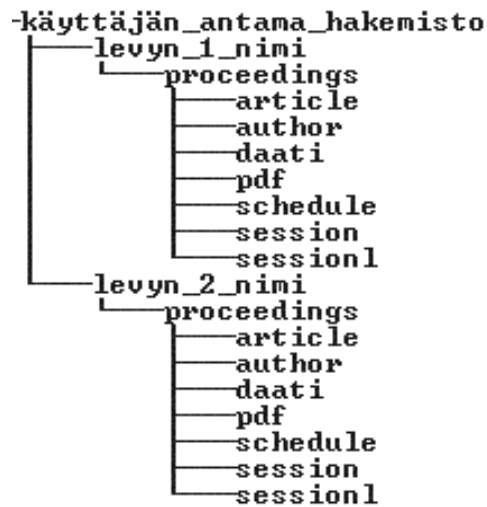
Kuva 12. Move Article -sivu.

## 7. Kokoomajulkaisun generoiva sovellus

Kokoomajulkaisun generoiva sovellus luo konferenssin proceedings-levyjen hakemistorakenteen. Sovellus luo jokaiselle luotavalle levyille oman hakemiston, joihin generoidaan HTML-sivut ja kopioidaan artikkelit PDF-muodossa. Hakemistoista luodaan CD-ROM-levyjen image-tiedostot.

### 7.1. Hakemistorakenteen generointi

Sovellus suorittaa ensin hakemistorakenteen generoinnin. Sovellus luo käyttäjän syöttämän hakemiston alle oman alihakemiston jokaiselle levyille, ja nimeää ne levyn nimen mukaan. Jokaisen levyn hakemistoon luodaan *Conference Mainpage* -HTML-sivu nimellä `index.html` sekä alihakemisto `proceedings`, johon tallennetaan proceedingsiin liittyvä materiaali. `Proceedings`-hakemiston alle luodaan hakemisto `pdf`, johon kopioidaan kaikki kyseiselle levyille tulevat artikkelit. Artikkelit nimetään artikkelin id-numeron mukaan muotoon "`#.pdf`" eli esimerkiksi `51.pdf`. Tämän jälkeen luodaan vielä jokaista eri selausmahdollisuutta varten oma hakemisto. Hakemistojen nimet on kuvattu luvussa 7.3.4. Kuvassa, Kuva 13, on esitetty generoitava hakemistorakenne.

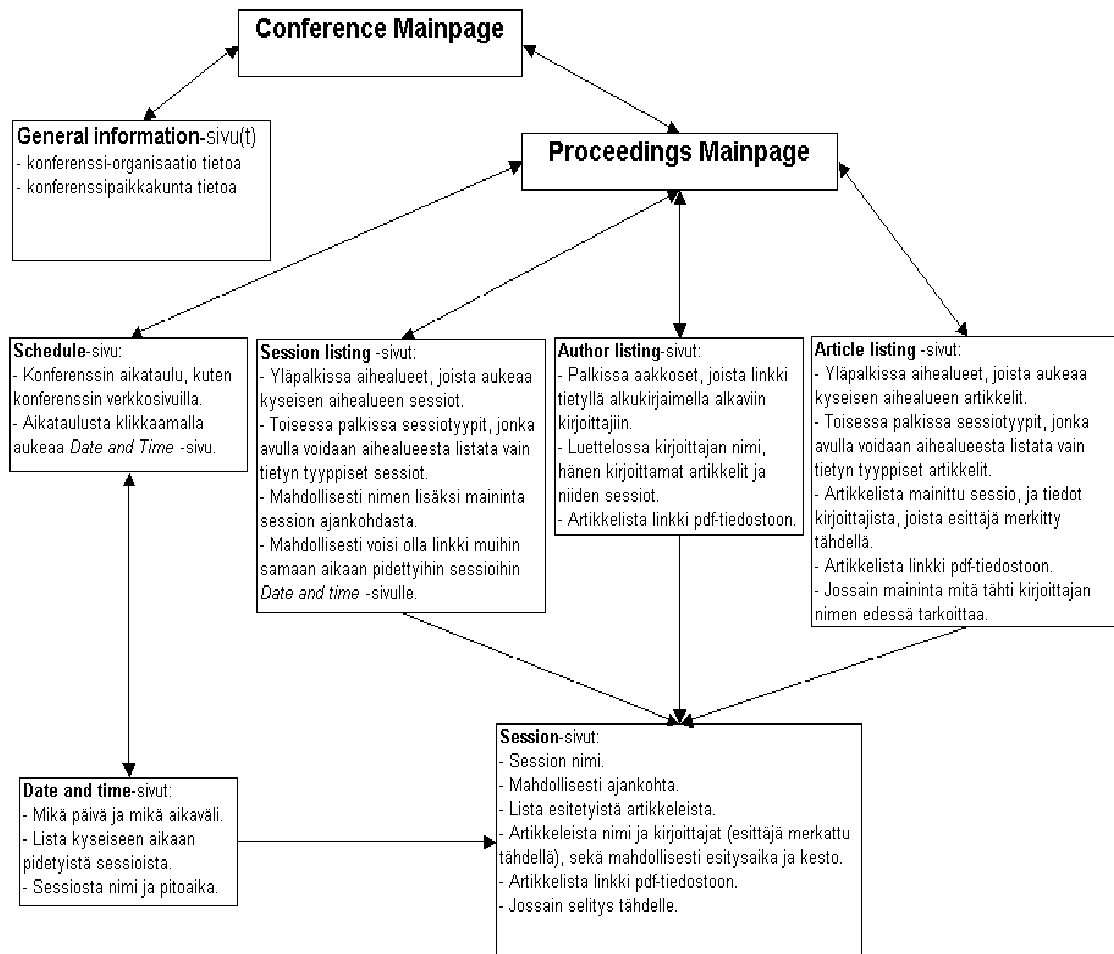


Kuva 13. Kokoomajulkaisun generoivan sovelluksen generoima hakemistorakenne.

## 7.2. Sivujen generointi

Hakemistorakenteen generoinnin jälkeen suoritetaan sivujen generointi. Kuvassa, Kuva 14, on esitetty levyille generoitava sivurakenne. Sivujen väliset linkit on esitetty nuolina. Lisäksi jokaiselta *proceedings*-sivulta päästään *Proceedings Mainpage* -sivulle, kuvassa nuolet on jätetty pois selkeyden säilyttämiseksi.





Kuva 14. Sovelluksen generoima sivurakenne.

### 7.2.1. Conference Mainpage -sivu

*Conference Mainpage* -sivu on ensimmäinen sivu, joka aukaistaan levyltä. Sivulla tulee olemaan linkit *Proceedings Mainpage* -sivulle sekä *General Information* –sivu(i)lle.

### 7.2.2. General Information -sivu(t)

*General Information* –sivu(t) sisältävät yleistä tietoa ECCOMAS2004-konferenssista. Sivu(i)lla on muun muassa:

- Information about Jyväskylä
- Information about ECCOMAS-organization
- Information about ECCOMAS2004-organization
  - Organizers
  - Organizing Committee
  - Scientific and Industrial Committees
  - International Corresponding Committee
- Information about this publication
  - Editors
  - Publisher
  - ISBN

Sivut nimetään info1, info2... nimisiksi html-tiedostoksi.

### 7.2.3. Proceedings Mainpage -sivu

*Proceedings Mainpage* -sivu on proceedings-julkaisun pääsivu. Tälle sivulle tulee linkit kaikkiin selausmahdollisuuksiin.

### 7.2.4. Article listing -sivut

*Article listing* -sivuille toteutetaan artikkeleiden selaus aihealueen mukaan.

Sivuille tulee kaksi palkkia, joista ensimmäinen sisältää aihealueen valinnan ja toisesta valitaan listataanko kaikki aihealueen artikkelit vai pelkästään tiettyyn sessiotyyppiin kuuluvat artikkelit.

Sisältöosassa on artikkelit listattu nimen mukaan aakkosjärjestykseen. Artikkelista mainitaan myös esityssessio sekä kirjoittajat, joista esittäjä on merkitty tähdellä. Artikkelin nimestä on linkki, josta aukeaa artikkeli pdf-muodossa ja sessiosta on linkki kyseisen session omalle sivulle. Jossakin kohtaa mainitaan mitä tähti kirjoittajan edessä tarkoittaa.

*Article listing* -sivut nimetään siten, että alkuun tulee "article", jonka perässä aihealueen numero ja lopuksi vielä tyypin numero tai 0 mikäli sivulla listattu kaikki

artikkelit. Aihealueen ja tyyppin välissä on t-merkki erotinmerkkinä. Esimerkiksi sivu, jolla on listattu aihealueen 1 tyyppiä 3 olevat artikkelit on muotoa `article1t3.html`.

### **7.2.5. Author listing -sivut**

*Author listing* -sivuille toteutetaan artikkeleiden selaus kirjoittajan mukaan.

Sivuille tulee palkki, jossa on aakkoset lueteltuna kirjain kirjaimelta. Kirjaimesta on linkki sivulle, jolla on listattu kaikki vastaavalla kirjaimella alkavat artikkelien kirjoittajien sukunimet.

Sisältöosassa on kirjoittajan nimi, jonka jälkeen on lueteltuna hänen kirjoittamansa artikkelit. Jokaisesta artikkelista on mainittu nimen lisäksi esityssessio sekä muiden kirjoittajien nimet. Muissa kirjoittajissa on esittäjän nimen eteen merkattu tähti, jonka merkitys kerrotaan myös sivulla. Mikäli kyseinen kirjoittaja on itse myös esittänyt artikkelin, tähteä ei merkitä ollenkaan. Muiden kirjoittajien mainitseminen saatetaan jättää pois. Artikkelin nimestä on linkki, josta aukeaa artikkeli pdf-muodossa ja sessiosta on linkki kyseisen session omalle sivulle.

*Author listing* -sivut nimetään siten, että alkuun tulee "author", jonka perään kyseinen alkukirjain. Esimerkiksi A:lla alkavien kirjoittajien listaussivu on muotoa `authorA.html`.

### **7.2.6. Schedule-sivu**

*Schedule*-sivulle toteutetaan konferenssin aikataulu, kuten ECCOMAS2004-verkkosivuilla. Sivun toteutetaan mikäli projektiryhmällä jää aikaa.

Aikataulu esitetään taulukossa. Jokaisesta taulukon ruudusta on linkki kyseiselle *Date and Time* -sivulle.

*Schedule*-sivu nimetään `schedule.html:ksi`.

### **7.2.7. Session listing -sivut**

*Session listing* -sivuille toteutetaan sessioiden selaus aihealueen mukaan.

Sivuille tulee kaksi palkkia, joista ensimmäinen sisältää aihealueen valinnan ja toisesta valitaan listataanko kaikki aihealueen sessiot vai vaan tietyn tyyppiset.

Sisältöosassa sessiot on listattu aakkosjärjestyksessä nimen mukaan. Sessiosta on esitetty nimi sekä mahdollisesti milloin se pidetään. Session nimestä on linkki kyseisen session sivulle. Session järjestysajankohdasta on linkki kyseiselle *Date and Time* -sivulle, mikäli aikataulu toteutetaan.

*Session listing* -sivut nimetään siten, että ensin tulee "sessionl", jonka perään aihealueen numero ja tyyppin numero. Aihealueen ja tyyppin numerot erotetaan toisistaan t-merkillä. Mikäli listataan kaikki sessiot piittaamatta tyylistä, niin tyyppin numeroksi tulee 0. Esimerkiksi sivu, jolla listataan aihealueen 3 kaikki sessiot on muotoa `sessionl3t0.html`.

### **7.2.8. Date and Time -sivut**

*Date and Time* -sivuille toteutetaan tiettyyn aikaan pidettyjen sessioiden listaus. Sivut toteutetaan vain mikäli konferenssin aikataulu toteutetaan julkaisuun.

Sivuilla kerrotaan mistä päivästä ja aikavälistä on kyse. Tämän jälkeen listataan kyseiseen aikaan pidetyt sessiot. Sessiosta ilmoitetaan nimi ja pitoaika. Session nimestä on linkki kyseisen session sivulle.

*Date and Time* -sivut nimetään siten, että ensin tulee "dt", jonka perään päivämäärä kuusinumeroisena lukuna muodossa pp-kk-vv, ja tämän jälkeen alku- ja loppuaika muodossa hhmm-hhmm. Esimerkiksi sivu, jolla on listattuna kaikki sessiot, jotka on esitetty 24.07.2004 aikavälillä 12:00–15:00, on muotoa `dt24-07-04-1200-1500.html`.

### **7.2.9. Session-sivut**

*Session*-sivulla esitetään kyseisen session tiedot.

Sessiosta on mainittu nimi sekä mahdollisesti pitoaika. Pitoajasta on linkki kyseiselle *Date and Time* -sivulle, mikäli ne toteutetaan. Tämän jälkeen on listattu session artikkelit esitysjärjestyksessä. Artikkelista on mainittu otsikon lisäksi kirjoittajat, joista esittäjä on merkitty tähdellä. Artikkelin otsikosta on linkki, josta aukeaa artikkeli pdf-muodossa. Jossakin kohtaa sivua on selitys mitä tähti tarkoittaa kirjoittajien yhteydessä.

Session-sivut nimetään siten, että ensin tulee "session", jonka perään kyseisen session id-numero. Esimerkiksi session 5 sivu tulee olemaan muotoa `session5.html`.

### 7.3. Skriptit

Hakemistorakenteen ja HTML-sivujen generointi toteutetaan skriptien ja template-sivujen avulla. Tarvittavia skriptejä on lueteltu seuraavassa:

- `article.pl`, luo *Article listing* -sivut.
- `author.pl`, luo *Author listing* -sivut.
- `daati.pl`, luo *Date and time* -sivut (jos ryhmä ehtii toteuttaa).
- `generate.pl`, hallinnoi koko hakemistorakenteen luontia.
- `proceed.pl`, luo *Proceedings Mainpage* -sivun.
- `schedule.pl`, luo *Schedule*-sivun (jos ryhmä ehtii toteuttaa).
- `session.pl`, luo *Session*-sivut.
- `sessionl.pl`, luo *Session listing* -sivut.

Kaikille edellä mainituille skripteille voidaan viedä samoja parametreja. Taulukossa, Taulukko 2, on kuvattu parametrien merkitys skripteille.

Parametri	Kuvaus
Ei mitään	Luodaan oletushakemistoon kyseinen sivu ja kaikki eri selausmahdollisuudet.
Hakemistopolku	Kyseinen sivu luodaan parametrin määrittämään hakemistoon kaikilla eri selausmahdollisuuksilla.
1	Luodaan <i>Article listing</i> -sivut.
2	Luodaan <i>Author listing</i> -sivut.
3	Luodaan <i>Session listing</i> -sivut.
4	Luodaan <i>Session</i> -sivut.
5	Luodaan <i>Schedule</i> -sivu.
6	Luodaan <i>Date and Time</i> -sivut.

Taulukko 2. Parametrit skripteille.

Eri parametreja voidaan myös yhdistää, jolloin voidaan esimerkiksi luoda haluttuun hakemistoon `proceedings`-pääsivu sekä tietyt selaussivut. Tällöin on aina määriteltävä hakemistopolku. Skriptien eri kutsumismuodot on kuvattu seuraavassa listassa käyttäen esimerkkinä `generate.pl`-skriptiä:

- `perl generate.pl`, generoi oletushakemistoon `proceedings`-julkaisun kaikilla eri selausmahdollisuuksilla.
- `perl generate.pl ./hakemistopolku`, generoi määrättyyn hakemistoon `proceedings`-julkaisun kaikilla eri selausmahdollisuuksilla.
- `perl generate.pl ./hakemistopolku 123`, generoi määrättyyn hakemistoon `proceedings`-julkaisun *Article listing*, *Author listing* ja *Session listing* -selauksilla.

Generoitaessa hakemistorakenne ja HTML-sivut, kutsutaan ensin `generate.pl`-skriptiä, jolle viedään halutut parametrit. Tämä puolestaan kutsuu muita skriptejä ja vie niille parametreissa tiedot hakemistopolusta sekä selausmahdollisuuksista. Liitteessä, Liite 1, on kuvattu kokoomajulkaisun generoivan sovelluksen sekvenssikaavio.

Skriptit generoivat HTML-sivut template-tekniikkaa hyväksikäyttäen. Tällöin skripti tuottaa itse enintään joitain linkkejä. HTML-dokumentti tehdään erilliseen template-tiedostoon, johon lisätään tarvittaviin paikkoihin template-tagit. Tagit korvataan skriptistä saatavilla muuttujilla lopulliseen HTML-dokumenttiin. Näin ollen sivun rakennetta voidaan muokata ilman, että tarvitsisi muuttaa skriptejä. Kokoomajulkaisun generoivan sovelluksen skriptit käyttävät hyväkseen `HTML::Template`-modulia HTML-sivujen generointiin.

Kaikkia muita skriptejä, paitsi `generate.pl`, varten luodaan oma template-sivu. Kyseiset sivut nimetään kuten skriptit, mutta `.tmpl`-päätteellä.

### 7.3.1. `article.pl`

Skripti generoi *Article listing* -sivut parametrina tuotuun hakemistoon käyttäen apunaan `article.tmpl`-tiedostoa.

Skripti hakee tietokannasta konferenssin aihealueet sekä sessiotyypit ja tallentaa nämä omiin taulukoihinsa. Tämän jälkeen luodaan jokaista aihealue-sessiotyyppi-yhdistelmää varten oma sivu.

Aihealueet viedään template-tiedostolle taulukossa nimeltä `topics` sekä sessiotyypit taulukossa `sessiontypes`. Taulukoissa on sijoitettu hajautustaulun

avaimen `title` perään kyseinen aihealueen tai sessiotyyppin nimi. `Sessiontypes`-tauluun lisätään alkuun alkio, jonka `title`-kentän arvo on All.

Sisältöosaa varten haetaan tietokannasta kyseisen aihealueen ja sessiotyyppin artikkeleiden tiedot. Nämä tallennetaan omaan taulukkoonsa. Lisäksi haetaan omiin taulukoihinsa kyseisen artikkelin kirjoittajien ja esityssession tiedot.

Sisältöosa viedään `template`-tiedostolle taulukossa nimeltä `articles`, jonka jokainen alkio sisältää hajautustaulussa seuraavat avaimet:

- `title`, artikkelin otsikko.
- `authors`, taulukko jossa avaimen `name` takana kirjoittajan nimi.
- `session`, artikkelin esityssessio.

### 7.3.2. `author.pl`

Skripti generoi *Author listing* -sivut parametrina tuotuun hakemistoon. Skripti luo yhtä monta HTML-sivua kuin eri kirjaimella alkavia kirjoittajia löytyy. Generoinnissa käytetään hyväksi `template`-tiedostoa `author.tmpl`.

Skripti hakee tietokannasta kaikkien kirjoittajien nimet. Tämän jälkeen luodaan taulukko, johon sijoitetaan kaikki löytyneet erilaiset sukunimen alkukirjaimet aakkosjärjestyksessä. Taulukon jokaisessa alkiossa on avaimen `alphabet` takana kyseinen aakkonen. Kyseinen taulukko viedään `template`-tiedostolle nimellä `alphabets`. Taulukko käydään läpi ja jokaista alkioita varten luodaan oma HTML-sivu.

Sisältöosaa varten skripti hakee tietokannasta kirjoittajien kirjoittamat artikkelit, näiden esityssessiot sekä artikkeleiden muut kirjoittajat. Nämä tiedot viedään `templatelle` taulukossa nimeltä `authors`, jonka jokainen alkio sisältää seuraavat avaimet:

- `article`, viittaus artikkeliin linkkiä varten.
- `name`, kirjoittajan nimi.
- `articles`, taulukko jossa avaimen `title` takana artikkelin otsikko.
- `sessions`, taulukko jossa avaimen `title` takana artikkelin esityssessio.
- `coauthors`, taulukko jossa avaimen `names` takana taulukko, joka sisältää jokaisessa alkiossa avaimen `name` takana toisen kirjoittajan nimen.

### 7.3.3. daati.pl

Skripti generoi *Date and Time* -sivut parametrina tuotuun hakemistoon käyttäen apuna template-tiedostoa `daati.tmp1`.

Skripti hakee tietokannasta konferenssin aikataulun ja luo jokaista aikataulun osaa varten oman HTML-sivun.

Template-tiedostolle viedään yleiset tiedot seuraavan nimisissä muuttujissa:

- `start`, ajankohdan alkuaika.
- `finish`, ajankohdan loppuaika.
- `date`, ajankohdan päivämäärä.

Tämän lisäksi haetaan tietokannasta kyseisellä ajankohdalla esitetyt sessiot. Sessioiden tiedot viedään template-tiedostolle taulukossa nimeltä `sessions`. Taulukon jokainen alkio sisältää kyseisen session nimen `title`-avaimen takana.

### 7.3.4. generate.pl

Skripti hallinnoi koko proceedings-julkaisun luomisprosessia.

Skripti hakee ensin tietokannasta tiedot kuinka monta eri levyä luodaan. Tämän jälkeen luodaan ensimmäiselle levyille oma alihakemisto. Kyseiseen alihakemistoon luodaan `proceedings`-hakemisto ja tähän hakemistoon edelleen alihakemisto nimeltä `pdf`, johon kopioidaan kaikki kyseiselle levyille tulevat artikkelit. Artikkelit nimetään artikkelin id-numeron mukaan muotoon `#.pdf` eli esimerkiksi `51.pdf`. Seuraavaksi luodaan vielä parametrien perusteella hakemistot selausmahdollisuuksia varten. Seuraavassa listassa on lueteltu hakemistot, joihin eri selausmahdollisuudet luodaan:

```
/proceedings/article, hakemistoon article listing -sivut.  
/proceedings/author, hakemistoon author listing -sivut.  
/proceedings/daati, hakemistoon date and time -sivut.  
/proceedings/schedule, hakemistoon schedule-sivun.  
/proceedings/session, hakemistoon session-sivut.  
/proceedings/session1, hakemistoon session listing -sivut.
```

Lopuksi kutsutaan muita skriptejä saatujen parametrien perusteella. Skripteille viedään parametrina edellä mainitussa listassa oleva hakemisto sekä tieto mitä muita selausmahdollisuuksia luodaan. Ainoa skripti jota kutsutaan parametreista välittämättä on `proceed.pl`, joka luo proceedings-julkaisun pääsivun. Skriptien kutsumisen järjestyksellä ei ole väliä.



### 7.3.5. **proceed.pl**

Skripti generoi *Proceedings Mainpage* -sivun parametrina tuotuun hakemistoon. Tämän lisäksi tutkitaan muut parametrit, joiden perusteella luodaan sivulle linkit eri selausmahdollisuuksiin.

Template-tiedostolle viedään eri selausmahdollisuudet seuraavissa muuttujissa:

- `article`, *Article listing* -sivujen sijainti.
- `author`, *Author listing* -sivujen sijainti.
- `session1`, *Session listing* -sivujen sijainti.
- `session`, *Session*-sivujen sijainti.
- `schedule`, *Schedule*-sivun sijainti.
- `daati`, *Date and Time* -sivujen sijainti.

Skripti tarkastaa luotavien levyjen määrän ja luo jokaiselle levyille oman ilmoitussivun. Sivulla on sisältönä ilmoitus; *The Article is in Disk number/name*. Luotaessa selaussivuja artikkeleiden, jotka eivät sijaitse juuri luotavan levyn hakemistossa, kohdalta linkitetään näihin ilmoitussivuihin.

### 7.3.6. **schedule.pl**

Skripti generoi *Schedule*-sivun parametrina tuotuun hakemistoon, käyttäen apunaan template-tiedostoa `schedule.tmpl`.

Skripti hakee tietokannasta konferenssin rakenteen ja tallentaa taulukon jokaiseen alkioon yhden päivän tiedot. Taulukon alkio pitää sisällään kyseisen päivämäärän `date`-avaimen takana sekä aikataululaatikot taulukossa avaimen `parts` takana. Tämän taulukon jokainen alkio sisältää tietoja seuraavien avaimien takana:

- `title`, Aikataulussa näkyvä teksti.
- `start`, osan alkuaika.
- `duration`, osan kesto puolina tunteina, eli 3 tarkoittaa 1,5 tuntia.

### 7.3.7. **session.pl**

Skripti generoi *Session*-sivut parametrina tuotuun hakemistoon. Skripti luo jokaiselle sessiolle oman HTML-sivun. Luonnin yhteydessä käytetään template-tiedostoa `session.tmpl`.

Skripti luo ensin taulukon, johon haetaan tietokannasta kaikki sessiot. Jokaiselle sessiolle luodaan HTML-sivu käymällä kaikki taulukon alkiot läpi.

Sessiota varten haetaan tietokannasta sessiossa esitettyjen artikkeleiden otsikot sekä jokaisen artikkelin kirjoittajien nimet. Session nimi viedään template-tiedostolle muuttujassa `title` ja session alku ja loppuaika muuttujissa `start` ja `finish`. Näiden lisäksi sessiossa esitetyt artikkelit viedään template-tiedostolle taulukossa `articles`, jossa jokainen alkio sisältää tiedot seuraavien avaimien takana:

- `article`, viittaus artikkeliin linkkiä varten.
- `title`, artikkelin otsikko.
- `authors`, sisältää taulukon, jossa jokaisessa alkiossa `name`-avaimen takana on kyseisen kirjoittajan nimi.

### 7.3.8. `sessionl.pl`

Skripti generoi *Session listing* -sivut parametrina tuotuun hakemistoon. Skripti luo oman HTML-sivun jokaista aihealue-sessiotyyppi-yhdistelmää varten. HTML-sivujen luonnissa käytetään apuna template-tiedostoa `sessionl.tmpl`.

Skripti hakee ensin tietokannasta kaikki aihealueet ja sessiotyypit ja tallentaa ne omiin taulukoihinsa. Tämän jälkeen luodaan jokaista aihealue-sessiotyyppi-yhdistelmää varten oma sivu.

Aihealueet viedään template-tiedostolle taulukossa nimeltä `topics` sekä sessiotyypit taulukossa `sessiontypes`. Taulukoissa on sijoitettu hajautustaulun avaimen `title` perään kyseinen aihealueen tai sessiotyyppin nimi. `Sessiontypes`-tauluun lisätään alkuun alkio, jonka `title`-kentän arvo on `All`.

Sisältöosaa varten haetaan tietokannasta kyseisen aihealueen ja sessiotyyppin sessioiden nimet ja pitoajat. Nämä tallennetaan omaan taulukkoon. Sisältöosa viedään template-tiedostolle taulukossa nimeltä `sessions`, jonka jokainen alkio sisältää hajautustaulussa seuraavat avaimet:

- `title`, session nimi.
- `date`, session pitopäivä.
- `start`, session alkuaika.
- `finish`, session loppumisaika.

## 8. Ohjelmointi

Kappaleessa määritellään Perl- ja HTML-koodaustyyli, muuttujien nimeäminen ja sisennykset.

### 8.1. Perl

Ohjelmointikielenä on Perl. Kaikki ohjelmakoodit ja kommentit kirjoitetaan ainakin englanniksi. Lisäksi kommentit voidaan kirjoittaa myös suomeksi. Ohjelman muuttujat nimetään selkeästi ja kuvaavasti.

Tiedoston alkuun tulee sijoittaa kommentti, joka kertoo mikä tiedosto on kyseessä, tekijä ja päivämäärä. Ohjelman alussa tulee olla lyhyt (mutta riittävä) kuvaus siitä, mitä ohjelma tekee. Lisäksi alussa tulisi näkyä ohjelman tarvitsemat osat, versionumerointi ja lyhyt kuvaus mitä kulloiseenkin versioon on lisätty. Versiohistoria liitetään tiedostoihin CVS:n avulla.

```
#####  
# parser.pl, <pjen@cc.jyu.fi>, 2004-02-26T13:06:00. #  
# #  
# Parser.pl transfers data from directory to database, and do some #  
# corrections. #  
# Parseri parsii hakemistorakennetta. #  
# #  
# Program needs these parts #  
# Ohjelman tarvitsemat osat #  
# #  
# Version history: #  
# v0.1, 2004-02-26T13:06:00, Pentti Nakari #  
# Basic frame #  
# v0.2, 2004-03-02T13:09:00, Pentti Nakari #  
# Topic-file #  
#####
```

Tiedostossa tulee kommentoida muuttujat ja parametrilista. Mihin niitä käytetään, muuttuvatko ne ja muita tärkeitä tietoja. Lisäksi voidaan kirjoittaa selventävä esimerkki.

Tiedostossa tulee kertoa välikommentein isompien kokonaisuuksien tehtävä tai algoritmi.

```
#####  
# PostgreSQL-database connection #  
#####
```

Lisäksi tulee kommentoida ohjelmakoodia väliin pienemmissä kokonaisuuksissa.

```
# Author-files
    if ($line =~ /author_[0-9].txt/) {
    }
# Topic-files
    if ($line =~ /topic.txt/) {
    }

    $i=5;
    $j=3;    # start loop in middle
```

Kommentointi tehdään, jos käytetyn koodin merkitys algoritmossa ei ole itsestään selvää.

## 8.2. HTML ja CSS

Käytettävä HTML-koodi on HTML 4.01 Transitional -standardin[3] mukaista ja validoidaan World Wide Web Consortiumin validaattorilla. [8]

HTML-tiedostojen pakolliset osat ovat:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">

<html>
<head><title>Sivun otsikko</title>

<LINK REL="STYLESHEET" HREF="style.css" TYPE="text/css">
<META HTTP-EQUIV="Content-Type" CONTENT="text/html; CHARSET=ISO-8859-1">

</head>
<body>
```

Tähän väliin tulee sivun sisältö.

```
</body>
</html>
```

Seuraavat määritykset header-osioon ovat optionaalisia:

```
<!-- Logo/icon to browsers addressbar and bookmarks-->
<!-- Logo/ikoni selaimen osoitepalkkiin ja bookmarkkeihin -->
<LINK REL="SHORTCUT ICON" HREF="logo.ico">

<!-- All commercial tags, which have been generated by MS Explorer, off-->
<!-- MS Explorering generoimat mainostagit pois -->
<META NAME="MSSmartTagsPreventParsing" CONTENT="TRUE">
```

```
<!-- Language definition, not mandatory, for search engines -->
<!-- Kielimäärittely, ei pakollinen, lähinnä hakukoneita varten -->
<META NAME="language" CONTENT="English">

<!-- Description, not mandatory, for search engines -->
<!-- Kuvaus, ei pakollinen, lähinnä hakukoneita varten -->
<META NAME="description" CONTENT="Proceedings CD-ROM">
```

Template HTML-tiedostoihin kirjoitetaan kommentteina tiedoston tekijän ja editoijien tiedot seuraavasti:

```
<!-- ----- -->
<!-- page_top.html - Page top template -->
<!-- Valid HTML 4.01 Transitional - W3C Validation Service -->
<!-- 2004-04-01T16:12:00, Pentti Nakari, <pjen@cc.jyu.fi> -->
<!-- ----- -->
```

Ajon aikana täydennettävät kohdat sivuista merkitään prosenttimerkkien väliin sijoitetuilla avainsanoilla: %author\_list%.

HTML-sivujen tyyli määritellään CSS Level 2 -standardin[2] mukaisilla CSS-tyylitiedostoilla. CSS-koodi validoidaan World Wide Web Consortiumin validaattorilla.[7]

CSS-tiedostojen alkuun kirjoitetaan kommentteina tiedoston tekijän ja editoijien tiedot seuraavasti:

```
/* example.css - CSS example
 * 2004-03-30:18:26:00, Pentti Nakari, <pjen@cc.jyu.fi>
 * Valid CSS Level 2 - W3C CSS Validation Service
 */
```

Tyylitiedoston osat erotellaan toisistaan kommenttiriveillä:

```
/****** Edit this to change the main title font *****/
```

Sisennyskäytäntö on seuraava:

```
font.title, h1 {
    font-family: Verdana, Arial, Helvetica, Sans-Serif;
    font-size: 22px;
    font-weight: bold;
    color: #006600;
    background-color: #eeffee;
    width: 98%;
    border: 1px solid black;
    margin-left: 10px;
    margin-right: 10px;
    margin-top: 10px;
```

```
margin-bottom: 10px;  
padding-left: 6px;  
padding-right: 2px;  
padding-top: 2px;  
padding-bottom: 2px;  
}
```

## 9. Testaus

Testaus suoritetaan projektissa erikseen laadittavan Testaussuunnitelman mukaisesti.

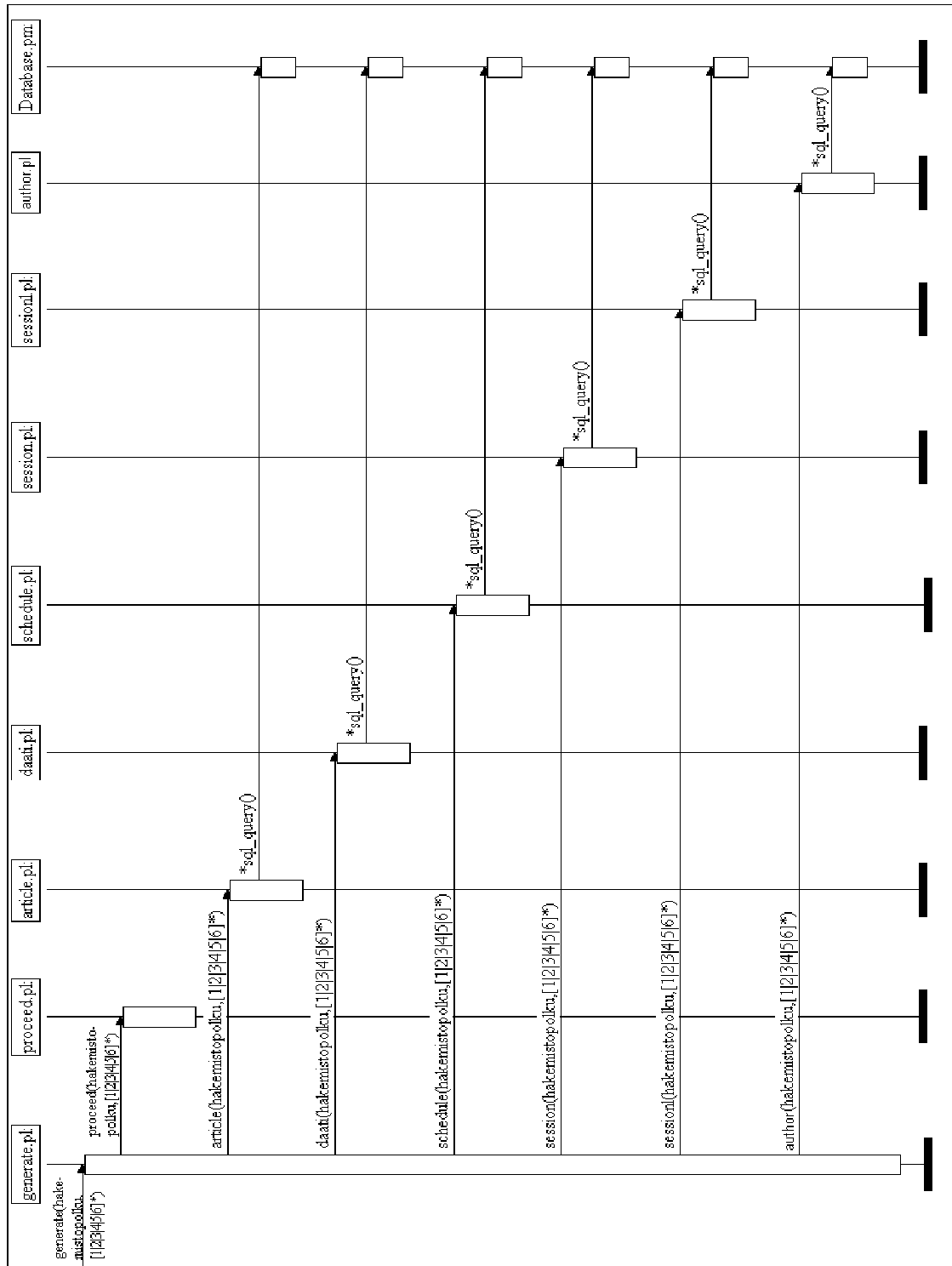
## 10. Yhteenveto

Tämä dokumentti sisältää Coma-projektin tuottamien sovellusten teknisen suunnittelun yksityiskohtia. Projektiin liittyvät sovellukset tulee toteuttaa dokumentin kuvaamalla tavalla.

## 11. Lähteet

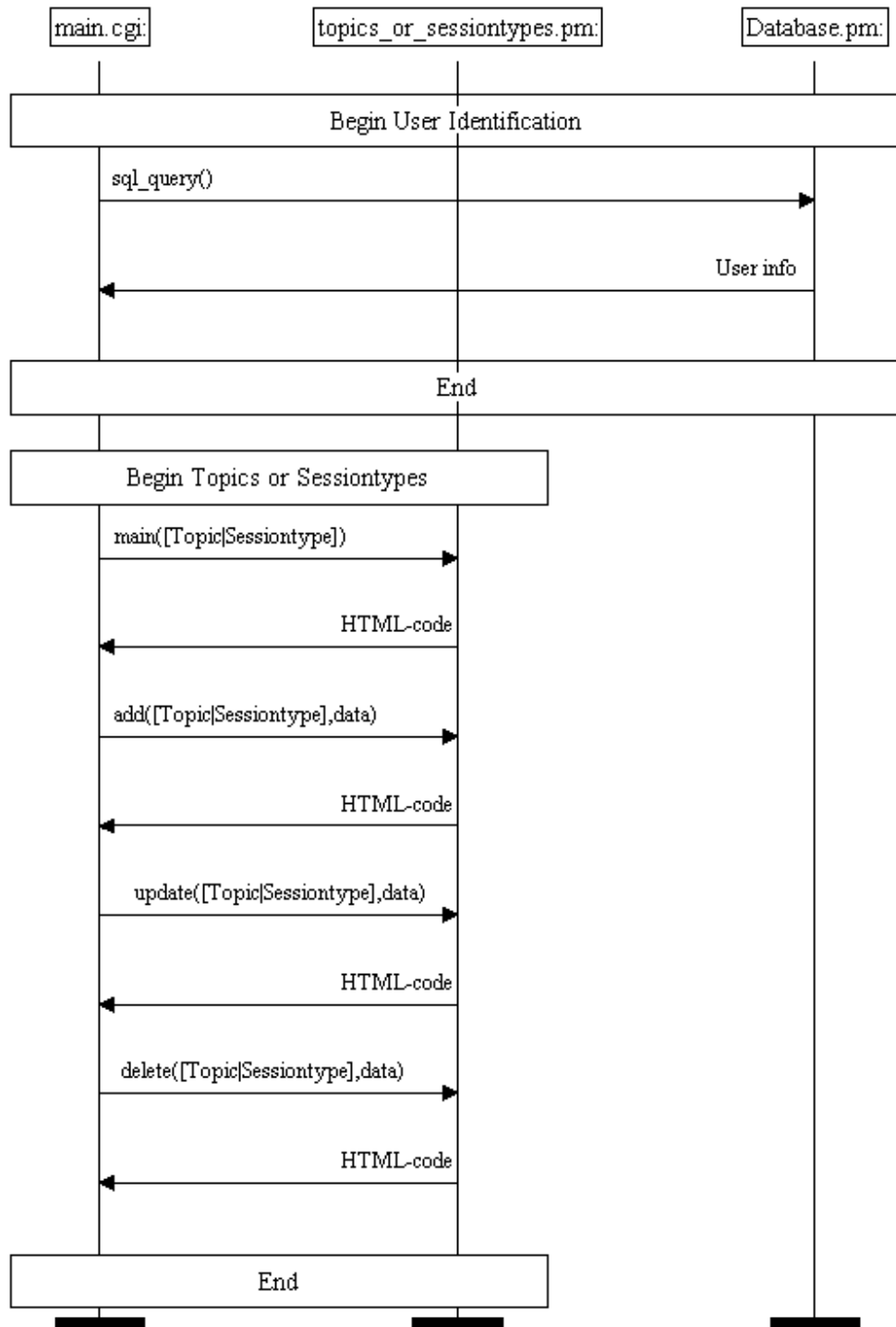
- [1] Alm Hanna, Arkko Kari, Nakari Pentti ja Paavolainen Lassi, *Vaatimusmäärittely*, Jyväskylän yliopiston tietotekniikan laitos, Jyväskylä, 2004.
- [2] Cascading Style Sheets, level 2 Specification <URL: <http://www.w3.org/TR/CSS2/>>, 5.4.2004.
- [3] HTML 4.01 Specification <URL: <http://www.w3.org/TR/html401/>>, 5.4.2004.
- [4] Jaakohuhta Hannu, *Suuri tietotekniikan tietosanakirja käsitteistö ja sanasto*, Suomen Atk-kustannus Oy, Helsinki, 1999.
- [5] Jaakohuhta Hannu, *IT Ensyklopedia*, Edita Oyj, Helsinki, 2001.
- [6] Rahkila Martti, *WWW-ohjelmoinnin perustekniikoita*, saatavilla WWW-muodossa <URL: <http://www.acoustics.hut.fi/u/mara/doc/hype/ohjelmointi.html>>, 13.2.2003.
- [7] W3C CSS Validation Service <URL: <http://jigsaw.w3.org/css-validator/>>, 5.4.2004.
- [8] W3C Markup Validation Service <URL: <http://validator.w3.org/>>, 5.4.2004.

## 12. Liitteet

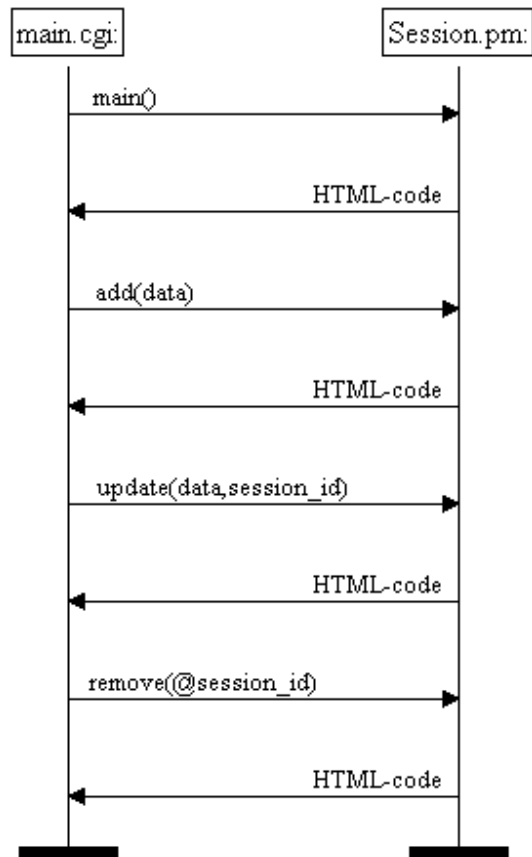


Liite 1. Kokoomajulkaisun generoivan sovelluksen sekvenssikaavio.

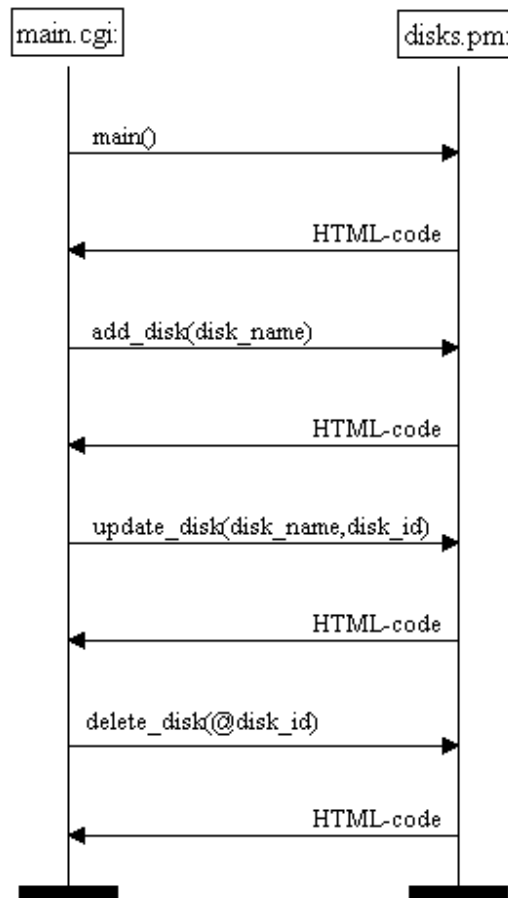




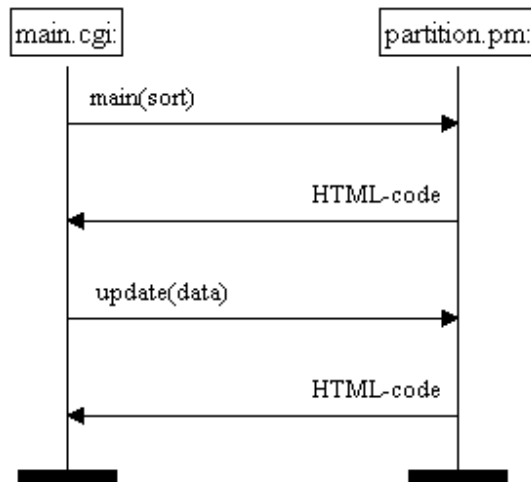
Liite 2. main.cgi-pääsivun ja topics\_or\_sessiotypes.pm-moduulin välinen sekvenssikaavio.



Liite 3. main.cgi-pääsivun ja session.pm-moduulin välinen sekvenssikaavio.



Liite 4. main.cgi-pääsivun ja disk.pm-moduulin välinen sekvenssikaavio.



Liite 5. main.cgi-pääsivun ja partition.pm-moduulin välinen sekvenssikaavio.