

HELSINGIN YLIOPISTO

TIETOJENKÄSITTELYTIETEEN LAITOS

OHJELMISTOTUOTANTOPROJEKTI

**HABA 2004**

Suunnitteludokumentti

Ahti Kare  
Antti Mäki  
Marko Lähde  
Mika Stenberg  
Riina Glinskih  
Teemu Virtanen

## Versiohistoria

<b>Versio</b>	<b>Päivämäärä</b>	<b>Kommentit/muutokset</b>
0.1	3.11.2004	Ensimmäinen versio.

## Sisältö

1	Johdanto .....	1
1.1	Järjestelmän yleiskuvaus.....	1
1.2	Laite- ja ohjelmistoympäristö .....	1
1.2.1	Ajoympäristö.....	1
1.2.2	Kehitysympäristö .....	2
2	Arkkitehtuuri.....	2
2.1	Yleiskuvaus.....	2
2.2	Osajärjestelmien tehtävät ja rajapinnat .....	4
2.3	Järjestelmän toiminta .....	5
3	Käyttöliittymä .....	6
3.1	Yleiskuvaus.....	7
3.2	JSP-sivut .....	7
3.2.1	Yleiset ominaisuudet.....	7
3.2.2	Kentät.....	8
	Sidemenu.jsp.....	10
	Initialview.jsp.....	10
	Contractview.jsp .....	12
	Companyview.jsp.....	14
	Changeconsequencies.jsp.....	15
	Changereport.jsp .....	17
	Leavingconsequencies.jsp.....	18
	Leavingreport.jsp .....	19
	Roleview.jsp.....	20
	Searchresults.jsp.....	21
	Sessionview.jsp.....	22
	Roleprocessview.jsp.....	23
	networkimage.jsp .....	24

4	Luokat ja rajapinnat .....	25
4.1	Actioncontroller-pakkaus.....	25
4.1.1	ActionParameters .....	25
4.1.2	ActionController .....	25
4.2	Datahandler-pakkaus.....	25
4.2.1	DataHandler.java.....	26
4.3	Messagehandler-pakkaus .....	31
4.4	Visualizing-pakkaus.....	34
4.5	Habautils-pakkaus .....	35
5	Lähdeluettelo.....	36

# **1 Johdanto**

## **1.1 Järjestelmän yleiskuvaus**

Haba 2004 on Helsingin yliopiston Tietojenkäsittelytieteen laitoksen ohjelmistotuotantoprojekti. Projektissa toteutetaan käyttöliittymä liiketoimintaverkostojen osapuolten yhteistoimintaa helpottavaan järjestelmään. Järjestelmän kautta käyttäjät pääsevät tarkastelemaan verkoston tilatietoja sekä rajoitetusti muuttamaan ja hallinnoimaan niitä. Käyttäjille esitetään verkoston rakenne graafisesti, mikäli se ko. tilanteessa on mahdollista.

Projektin pohjana toimii asiakkaan toteuttama liiketoimintaverkostojen hallintasovelluksen ydin, ns. sopimusvarasto. Nyt tuotettavan käyttöliittymän avulla on tarkoitus päästä hyödyntämään pohjalle rakennettua sovellusta hajautetusti www:n välityksellä.

Huomattavaa on, että asiakkaan järjestelmä ei ollut Haba2004- projektin alkaessa vielä lainkaan valmis vaan sitä on kehitetty ja muokattu useita kertoja käyttöliittymän määrittely- ja suunnitteluvaiheen aikana.

## **1.2 Laite- ja ohjelmistoympäristö**

Tässä uvussa kuvataan sovelluksen ajo- ja kehitysympäristö.

### **1.2.1 Ajoympäristö**

Projektissa tuotettavat HTML- sivut noudattavat HTML 4.0 spesifikaatiota ja vaativat näin ollen käyttäjiltään selaimen, joka on yhteensopiva sen kanssa. Samoin CSS- tyylimäärittelyt vaativat toimiakseen selaimen, joka tukee CSS:n versio 2:sta. Kehitysvaiheessa sivuja on testattu IE 6.0 sekä Mozilla 1.5 selaimilla.

JSP- sivut vaativat ympäristökseen jonkun JSP:tä tukevan palvelinohjelmiston.

Java- koodi vaatii kääntyäkseen vähintään Javan versio 1.4:n

Ohjelmisto on pitkälti laitteistoriippumaton. Java vaatii toimiakseen jonkin moniajtoa tukevan käyttöjärjestelmän. Lisäksi järjestelmä vaatii käyttöönsä verkkoyhteyden, jotta yhteys sopimusvarastoon voidaan muodostaa.

Suorittimen tehoon tai muistinkäyttöön liittyviä vaatimuksia ei ole.

## **1.2.2 Kehitysympäristö**

Sovellus toteutetaan Java-ympäristössä käyttäen Java J2SE 1.4.2. teknologiaa. Tuotettavan käyttöliittymän käyttäjälle näkyvät osat toteutetaan Java Server Pages (JSP) 2.0 –teknologiaa käyttäen, joka sisältyy Javan J2EE 1.4 spesifikaatioon. Samaisesta spesifikaatiosta käytetään niin ikään Servlet-versiota 2.4.

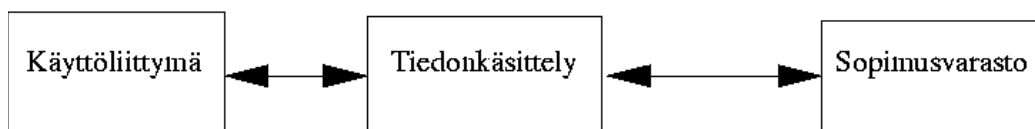
Toteutusvaiheessa käytetään Java-koodin tuottamiseen Eclipse-kehitysympäristön versiota 3.0, sekä J2EE-teknologian vaatimaa sovelluksen lisäkirjastoa. Kehitysympäristö tukee CVS-versionhallintaa, jota käytetään ohjelmakoodin säilyttämiseen ja julkaisemiseen.

Tuotantopalvelimena toimii Tietojenkäsittelytieteen laitoksen Apache Tomcat-sovelluspalvelin. Toteutettavan käyttöliittymän tietosisältö saadaan asiakkaan kehittämästä järjestelmästä Web Services –rajapinnan kautta. Välitysdokumenttien käsittelyyn käytetään Apache Axis –projektin tuottamaa kirjastoa.

## **2 Arkkitehtuuri**

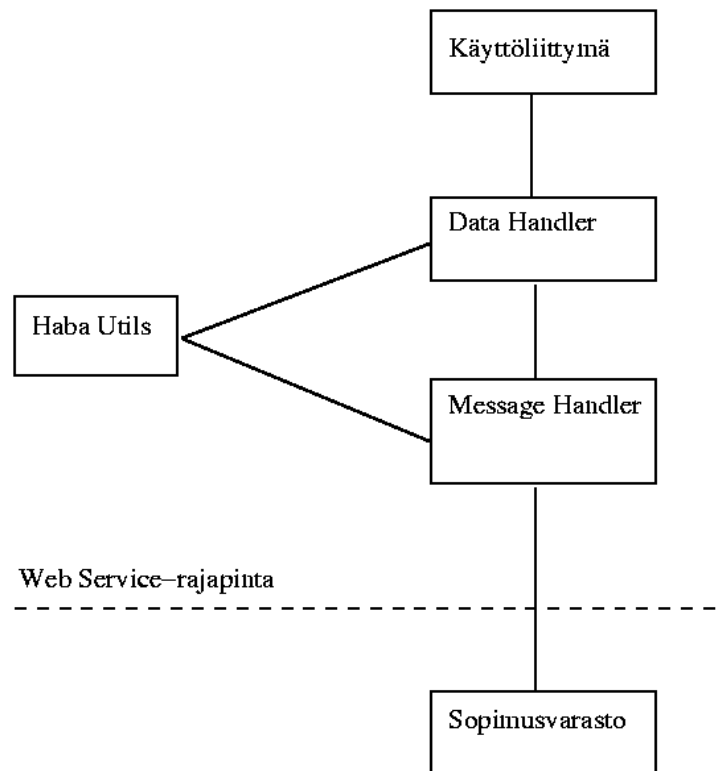
### **2.1 Yleiskuvaus**

Toteutettava liiketoimintaverkoston hallintapaneeli voidaan jakaa kolmeen osaan: Selaimen avulla käytettävään käyttöliittymäosaan, käyttöliittymän hyödyntämään tiedonkäsittelyosaan ja taustalla olevaan sopimusvarasto-järjestelmään. Kuvassa 1. on esitelty järjestelmä yleinen kolmikerrosmalli. Tässä tarkastellaan kahden ensimmäisen toteuttamiseen liittyviä kysymyksiä. Sopimusvarasto-järjestelmän toteuttaa asiakkaana oleva organisaatio. Sopimusvarasto on hallintapaneelin käytössä Web Service-rajapinnan kautta.



Kuva 1: Järjestelmän kolmikerrosmalli

Käyttöliittymä ja tiedon käsittely voidaan edelleen jakaa kolmeen pääosaan. Tässä osassa yleinen arkkitehtuuri vastaa MVC-suunnittelumallia. Selaimessa näkyvä osa, liiketoimintaverkoston hallintapaneeli tuotetaan jsp-sivuilla. Ne muodostavat yhdessä tarvittavien staattisten kuvien sekä html- ja css-tiedostojen kanssa yhden kokonaisuuden. Jatkossa osaa kutsutaan käyttöliittymäksi. Kokonaisuus vastaa MVC-mallin näkymä (view) osiota. Sivujen tiedonkäsittely osan välissä on kotrolleri, joka ohjaa pyynnöt oikealle käsittelijälle. Se vastaa MVC-mallin controller osaa. Sivujen taustalla, tiedonkäsittely-osassa toimivat DataHandler ja MessageHandler. DataHandler huolehtii käyttöliittymän tarvitsemien tietojen keräämisestä ja koostamisesta. MessageHandler huolehtii kommunikaatiosta sopimusvarasto-järjestelmään. Yhdessä ne muodostavat MVC-mallin data (model) osan. Näiden lisäksi on joukko sekalaisia apuluokkia. Niiden avulla hoidetaan esimerkiksi hallintapaneelin lokien kirjoitus ja tarvittava tila-automaatin visualisointi. Osaa ei ole kuvassa 1. Kutsutaan osaa jatkossa Haba utils-osaksi. Kuvassa 2. on liiketoimintaverkoston hallintapaneelin osajärjestelmät ja niiden väliset liitokset.



Kuva 2: Liiketoimintaverkoston hallintapaneelin osajärjestelmät ja niiden väliset liitokset

## 2.2 Osajärjestelmien tehtävät ja rajapinnat

Käyttöliittymä-osajärjestelmään kuuluvat kaikki käyttäjälle näkyvät komponentit ja niiden välittömään tuottamiseen tarvittava toiminnallisuus. Tarvittava toiminnallisuus toteutetaan jsp-tekniikalla. Muita käyttäjälle näkyviä osia, suoraan tai välillisesti, ovat staattiset html- ja css-tiedostot ja käyttöliittymässä käytettävät staattiset kuvat. Käyttöliittymässä näytettävän dynaamisen tietosisällön osajärjestelmä saa alla olevalta DataHandler-osajärjestelmältä. Käyttöliittymä ei käytä muita osajärjestelmiä suoraan, eikä sillä ole omaa julkista palvelurajapintaa.

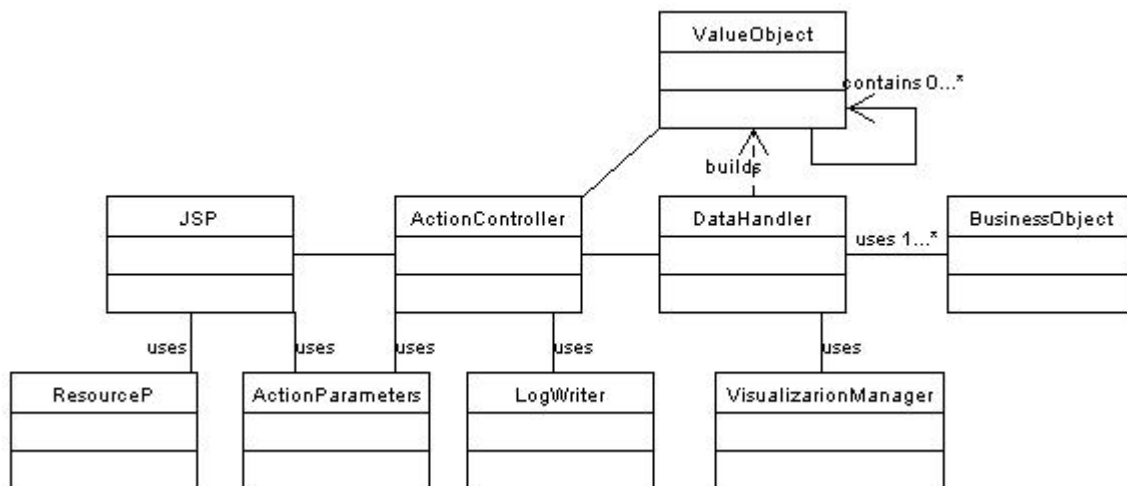
DataHandler-osajärjestelmä huolehtii näytettävän tietosisällön koostamisesta. Se palauttaa halutun tiedon yhdessä oliossa käyttöliittymälle. Kaikki näytettävä tieto on alun perin haettu sopimusvarasto järjestelmästä. Monissa käyttötapauksissa tarvitaan tietoja, joita ei saada suoraan sopimusvarasto-järjestelmästä, vaan haluttu tieto tulee koostaa eri kyselyillä saaduista tiedoista. Osajärjestelmä noudattaa arvo olion koostaja (value object assembler) suunnittelumallia. DataHandler-osajärjestelmässä on käyttötapauskohtaiset käsittelijät palvelupyynnöille, sekä



joukko niiden tuottamisessa tarvittavia apuluokkia. Data Handler välittää käyttöliittymälle myös dynaamisesti tuotetut kuvat, jotka muodostetaan Haba Utils-osajärjestelmässä. Osajärjestelmällä on julkinen rajapinta, joka on tarkoitettu käyttöliittymän käyttöön.

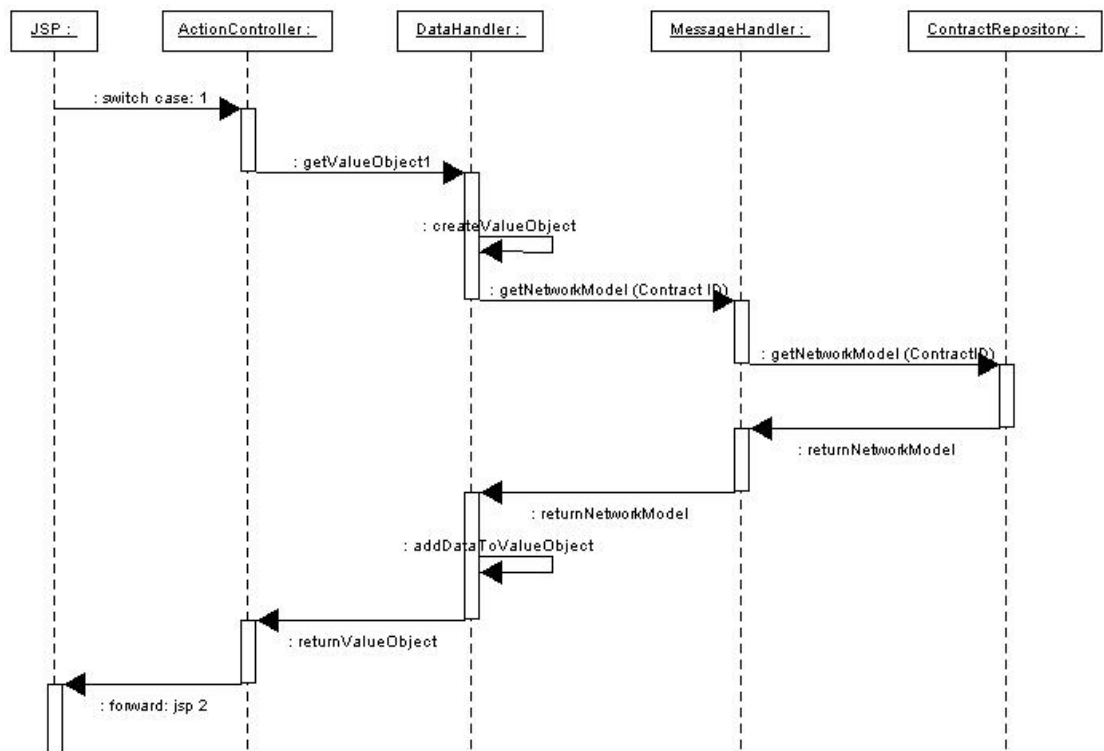
MessageHandler-osajärjestelmä huolehtii Web Service-rajapinnan kautta tapahtuvasta kommunikaatiosta sopimusvarasto-järjestelmään. Sen julkista rajapintaa käyttää ainoastaan MessageHadler.

Haba utils-osajärjestelmään kuuluvat kaikki muut toteutettavat osat. Ne ovat luonteeltaan sekalaisia apuluokkia, jotka eivät kuulu oleellisesti mihinkään muuhun osaan. Tällaisia ovat tarvittavien lokien pitäminen, asennus-tiedostojen käsittely ja tarvittavien dynaamisten kuvien tuottaminen. Osajärjestelmän julkista rajapintaa käytävät Data Handler ja Message Handler osajärjestelmät.



Kuva 3: Luokkakaavio järjestelmän keskeisimmistä luokista

### 2.3 Järjestelmän toiminta



Kuva 4: Osajärjestelmien toiminta tyypillisessä tapauksessa

Kuvassa 4 on hahmoteltu osajärjestelmien toimintaa tyypillisessä tapauksessa. Käyttöliittymältä tulee palvelupyyntö Data Handlerille. Tapahtuman käsittelevä Hadler olio pyytää tarvittavan tiedon Message Handler osajärjestelmältä, joka hakee sen edelleen Sopimusvarasto-järjestelmästä. Kun kaikki pyynnössä haluttu tieto on koossa Data Hadler palauttaa sen yhdessä oliossa käyttöliittymälle. Tiedot saatuaan jsp-sivu näyttää ne käyttäjälle. Käyttäjälle näkyvä ulkoasu määräytyy staattisten html- ja css-tiedostojen mukaan.

### 3 Käyttöliittymä

Järjestelmän käyttöliittymä on kuvattu käyttötapauskohteisesti dokumentin liitteenä olevassa Antti Mäen PowerPoint-esityksessä [Mäk04].

Käyttöliittymä muodostuu JSP-sivujen generoimista näytöistä. JSP-sivut ja niiden väliset siirtymät sekä toiminnot ja kentät esitellään kappaleessa 3.2.

## 3.1 Yleiskuvaus

Tähän tulee.

## 3.2 JSP-sivut

JSP-sivut on kuvattu esittelemällä ensin kaikille sivuille yhteiset yleiset ominaisuudet luvussa 3.2.1. JSP-sivukohtainen käyttöliittymän kenttien kuvaus parametritietoineen esitellään luvussa 3.2.2.

### 3.2.1 Yleiset ominaisuudet

Jokaisella sivulla määritellään JSP:n error page -tagilla käytettäväksi yleinen controlpanelerror.html-virhesivu. Sivun reagoi kaikkiin käyttöliittymätasolle päätyneisiin poikkeuksiin. Tarkemman tason virheisiin ei varauduta käyttöliittymässä, vaan niiden käsittely hoidetaan sovelluksen muissa kerroksissa. Controlpanelerror.html on staattinen sivu, joka ilmoittaa käyttäjälle järjestelmässä tapahtuneesta virheestä.

```
<error-page>
  <exception-type>java.lang.Exception</exception-type>
  <location>/controlpanelerror.html</location>
</error-page>
```

Käyttöliittymässä esitettävät staattiset tekstimuotoiset otsikkokentät on lokalisoitu termistön muuttumisen varalta Javan ResourceBundle-luokan avulla:

```
<%@ page import="java.util.*;"%>

<% ResourceBundle myResources =
ResourceBundle.getBundle("haba2004.MyResources");%>

<%= (myResources.getString("SOPIMUS")) %>
```

Jokainen JSP-sivu käyttää järjestelmän toimintojen indeksointiin ActionParameters-luokasta JSP:n use bean-tagilla luotavaa istuntokohtaista ilmentymää:

```
<jsp:useBean id="actionParameters" scope="session"  
class="general.ActionParameters" />
```

Sivuvalikon HTML:n generoinnista vastaa jokaiselle JSP-sivulle sisällytettävä sidemenu.jsp:

```
<jsp:include page="/sidemenu.jsp"></jsp:include>
```

Sivuihin liittyvät tyylit on kuvattu erillisessä styles.css-tyylitiedostossa.

### 3.2.2 Kentät

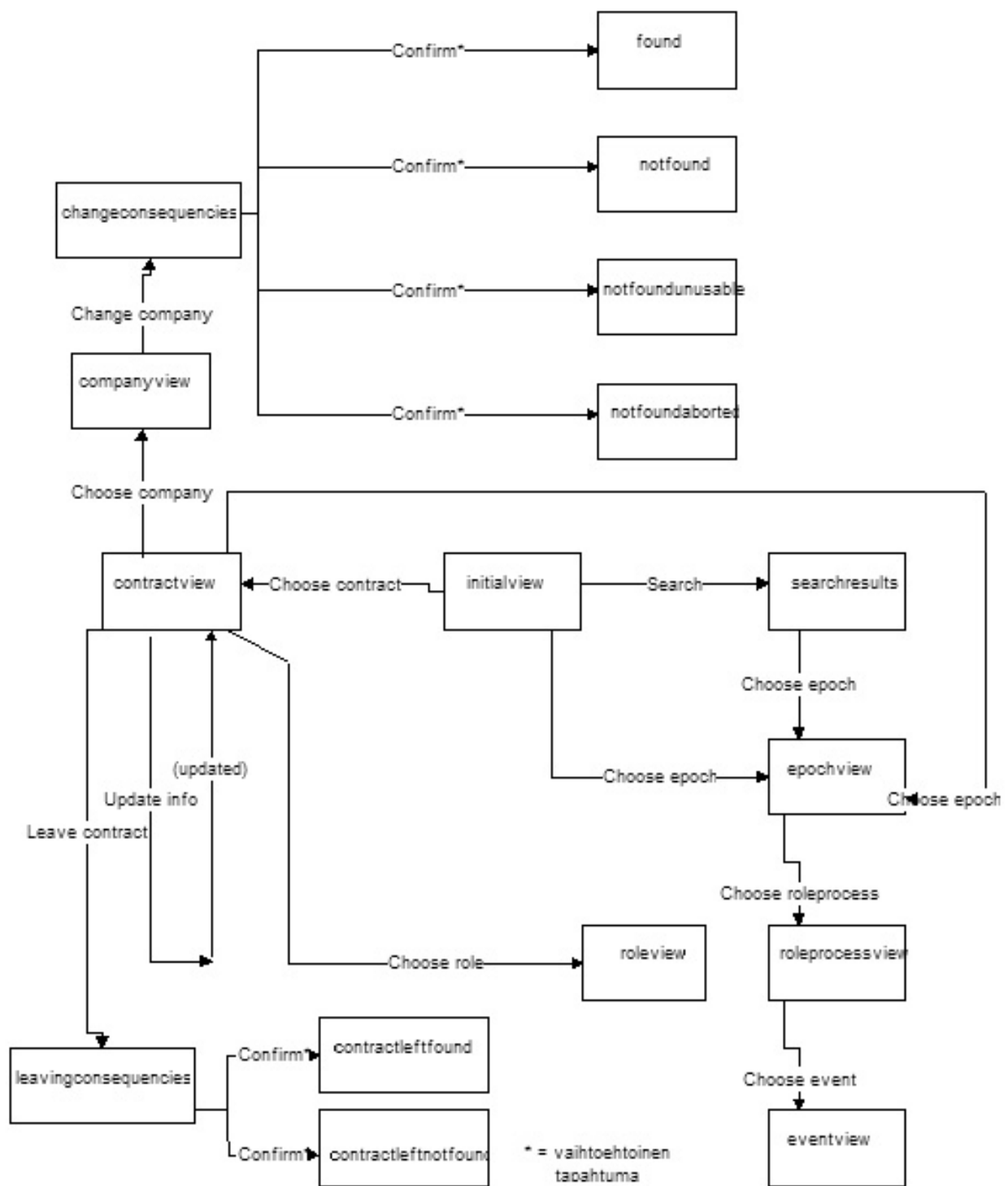
JSP-sivujen parametrien nimet on esitelty kuhunkin sivuun liittyvässä taulukossa. Parametreja on kahta tyyppiä. Parametri tietoa esitettäessä tarkoittaa parametrin nimeä, jolla saadaan esimerkiksi sopimuksen käyttöliittymässä esitettävä nimi:

```
<%= request.getParameter("contract_name") %>
```

Toinen parametrityyppi liittyy järjestelmän toimintoihin. Tähän parametriluokkaan kuuluvat toimintolinkkeihin generoitavat parametrit, kuten toiminnon identifioiva ns. action-parametri ja kyseisen toiminnon suorittamiseen tarvittavat mahdolliset muut parametrit:

```
<a href="ActionController?action=  
<%=actionParameters.VIEW_CONTRACT_DETAILS %>&  
contract_id=<%= request.getParameter("contract_id") %>  
Sopimus 1</a>
```

JSP-sivujen nimeäminen ja kuhunkin toimintoon liittyvä seuraavaksi kutsuttava sivu käy ilmi oheisesta näyttötilakaaviosta (Kuva 5).



Kuva 5: Näyttötilakaavio

## Sidemenu.jsp

Sidemenu.jsp sisällytetään jokaiseen näkymään JSP-sivukohtaisesti include page –tagilla.



Kuva 6: Navigointivalikko

Kenttä	Parametrin nimi tietoa esitettäessä	Toimintoparametri (action)	Toiminnon parametrit	Huomautukset
Update-linkki		initiate_session		Tämä suoritetaan myös esim. järjestelmän käynnistyksen yhteydessä.
Contract-linkki	contract_name	choose_contract	contract_id (ContractContent.contractID)	
Session-linkki	session_name	choose_session	session_id (ContractSession.sessionID)	
Search for a	-	search_company	company_identifier	Tämän parametrin

company- tekstikentt ä				arvo on osa yrityksen nimestä.
Quit-linkki				Linkki vie suoraan tyhjälle lopetussivulle. Tähän ei liity toimintoja.

Taulukko 1: Navigointivalikon parametrit

## Initialview.jsp

### Existing contracts

- [Contract 1, established](#)

- short description
- startDate and (possible) endDate



- [Contract 2, in-negotiation](#)

- short description
- startDate and (possible) endDate



- [Contract 3, terminated](#)

- short description
- startDate and (possible) endDate



Kuva 7: Perusnäyttö

Kenttä	Parametrin nimi tietoa esitettäessä	Toimintoparamet ri (action)	Toiminnon parametrit	Huomautuks et
--------	--	-----------------------------------	----------------------	------------------





Näytössä esiintyvä kuva verkoston rakenteesta Epoch-linkkeineen sijaitsee erillisessä kehyksessä. Kyseiseen osaan liittyvät toiminnot parametreineen on kuvattu networkimage.jsp-luvussa, joten niitä ei käsitellä muiden näyttöjen yhteydessä.

Kenttä	Parametrin nimi tietoa esitettäessä	Toimintoparametri (action)	Toiminnon parametrit	Huomautukset
Leave contract -painike		leave_contract	contract_id (ContractContent.contractID)	
short description	contract_description			
startDate	contract_start_date			
endDate	contract_end_date			
Roles (customer-sarake)	role_name	choose_role	role_name	
Roles (My company-sarake)	company_name	choose_company	company_name (ParticipantInfo.name)	
Channels (=Bindings)				Tämä tieto esitetään alalaidan kuvassa.
Allowed sessions	number_allowed_sessions			

Taulukko 3: Sopimuksen tiedot –näytön parametrit

# Companyview.jsp

**Company 1**

[Contract 1](#)

**Replace company**

---

- Name  
- Role

**Company information**

Address

Phone

E-mail

Contact

**Update info**

---

**Name of the business network model**

Epoch 1

```

graph TD
    Customer <--> Supplier
    Bank --> Customer
    Bank --> Supplier
        
```

**View**

[Epoch 2](#)

[Epoch 3](#)

Kuva 9: Yrityksen tiedot -näyttö

Kenttä	Parametrin nimi tietoa esitettäessä	Toimintoparametri (action)	Toiminnon parametrit	Huomautukset
Contract-linkki	contract_name	choose_contract	contract_id (ContractContent.contractID)	
Replece company-painike		change_company	participant_signature (ParticipantInfo.signature)	
Name	company_name			
Role	role_name			
Address	contact_addresses			
Phone	phone			
E-mail	contact_mail			

Contact	contact_person			
Update info - painike		update_info	company_name (ParticipantInfo.name), contact_address (ParticipantInfo.address), phone ParticipantInfo.phone, contact_mail (ParticipantInfo.contactMail), contact_person (ParticipantInfo.contactPerson )	

Taulukko 4: Yrityksen tiedot –näytön parametrit

## Changeconsequencies.jsp

### Company 1 replace consequences

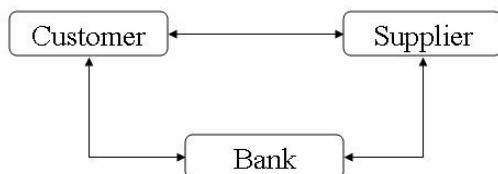
#### Contract 1

- list of roleprocesses terminated per each session
  - name of the roleprocess
  - other participants
  - description
- list of recovery processes
  - name of the recovery process
  - other participants
  - description

**Confirm**

#### **Name of the business network model**

Epoch 1



#### **View**

Epoch 2

Epoch 3

Kuva 10: Osallistujan vaihtamisen seuraukset -näyttö

Kenttä	Parametrin nimi tietoa esitettäessä	Toimintoparametri (action)	Toiminnon parametrit	Huomautukset
Company-linkki	company_name	choose_company	company_name (ParticipantInfo.name)	
Contract-linkki	contract_name	choose_contract	contract_id (ContractContent.contractID)	
name of the roleprocess	role_process_name			
other participants of the role process	role_process_other_participant			Näitä voi olla useita/rooliprosessi.
description of the roleprocess	role_process_description			
name of the recovery process	recovery_process_name			
other participants of the recovery process	recovery_process_other_participant			Näitä voi olla useita/recovery process.
description of the	recovery_process_description			

recovery process				
Confirm -painike		confirm _change_comp any	participant_signature (ParticipantInfo.signature)	

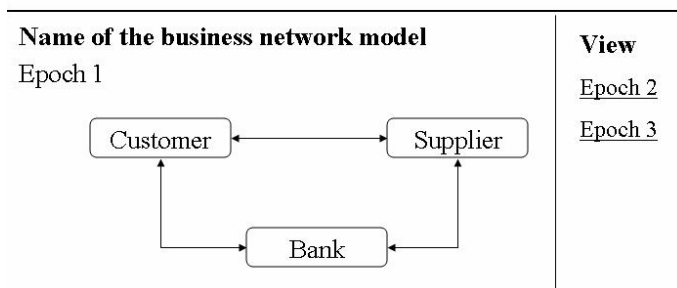
Taulukko 5: Sivun parametrit

## changereport.jsp

### Company 1 replaced

[Contract 1](#)

- details of the new company
- details of new processes
- state of contract after the replacement
- report of changes



Kuva 11: Raportti osallistujan vaihtamisesta

Kenttä	Parametrin nimi tietoa esitettäessä	Toimintoparametri (action)	Toiminnon parametrit	Huomautukset
Contract-linkki	contract_name	choose_contract	contract_id (ContractContent.contractID)	

details of the new company (Name)	new_company_name	choose_company	company_name (ParticipantInfo.name)	Tästä pääsee käsiksi uuden yrityksen tarkempiin tietoihin.
details of the new company (Role)	new_company_role	choose_role	role_name	
details of new processes	new_process_details			Tieto on yhdessä String-muuttujassa.
state of contract after the replacement	new_contract_state			

Taulukko 6: Sivun parametrit

## Leavingconsequencies.jsp

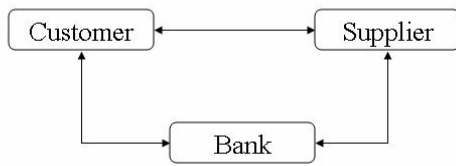
## Contract 1 leaving consequences

- list of roleprocesses terminated
  - name of the roleprocess
  - other participants
  - description
- list of recovery processes
  - name of the recovery process
  - other participants
  - description

Confirm

### Name of the business network model

Epoch 1



### View

Epoch 2

Epoch 3

Kuva 12: Sopimuksesta lähtemisen seuraukset -näyttö

Näkymän tietosisältö on sama kuin changeconsequences.jsp-sivulla. Vain Confirm-painikkeen parametrit eroavat näkymien välillä. Ne on kuvattu taulukossa.

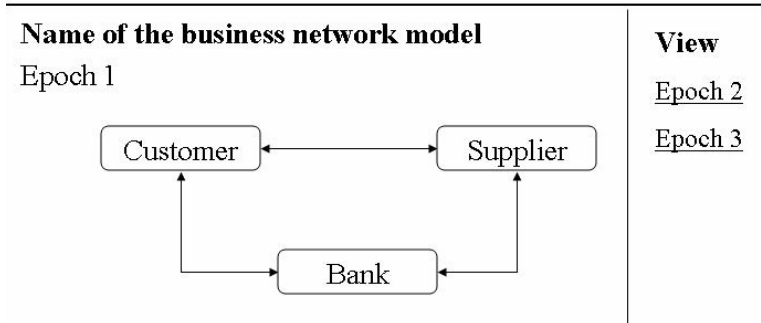
Kenttä	Parametrin nimi tietoa esitettäessä	Toimintoparametri (action)	Toiminnon parametrit	Huomautukset
Confirm-painike		confirm _leave_contract	participant_signature (ParticipantInfo.signature)	

Taulukko 7: Sivun parametrit

## Leavingreport.jsp

## Contract 1 left

- new company not found
- network is still operational
- list of roleprocesses terminated per each session
  - name, other participants, description
- list of recovery processes
  - name, other participants, description
- report of changes



Kuva 13: Raportti sopimuksesta lähtemisestä

Tietosisältö on pääosin sama kuin changereport.jsp-sivulla. Poikkeavat osat on listattu taulukossa.

Kenttä	Parametrin nimi tietoa esitettäessä	Toimintoparametri (action)	Toiminnon parametrit	Huomautukset
selitys (new company not found...)	report_text			Tämä tieto kulkee yhdessä String- muuttujassa.

Taulukko 8: Sivun parametrit

## Roleview.jsp

Kuva lisätään.

Kuva 14: Roolinäkömä

Kenttä	Parametrin nimi tietoa	Toimintoparametri	Toiminnon	Huomautukset
--------	------------------------	-------------------	-----------	--------------



	esitettäessä	(action)	parametrit	
	role_process_name			
	role_process_description			
	role_process_image_url			

Taulukko 9: Sivun parametrit

## Searchresults.jsp

Search results

---

- **Company 1**
  - [Contract 1, established](#)
    - [Session 1](#)
    - [Session 2](#)
    - [Session 3](#)
  - [Contract 3, terminated](#)
    - [Session 1](#)
    - [Session 2](#)


---

Kuva 15: Yrityshaun tulokset




Kenttä	Parametrin nimi tietoa esitettäessä	Toimintoparametri (action)	Toiminnon parametrit	Huomautukset
Company-linkki	company_name	choose_company	company_name (ParticipantInfo.name)	
Contract-linkki	contract_name	choose_contract	contract_id (ContractContent.contractID)	
Session-linkki	session_name	choose_session	session_id (ContractSession.sessionID)	

## Sessionview.jsp

Last update  
24.11.2004  
18:35

 Update

**Menu**

- [Contract 1](#) 
  - [Session 1](#)
  - [Session 2](#)
  - [Session 3](#)
- [Contract 2](#) 
  - [Session 1](#)
  - [Session 2](#)
- [Contract 3](#) 
  - [Session 1](#)
  - [Session 2](#)
  - [Session 3](#)
  - [Session 4](#)

Search for a company

[Quit](#)

Session 2, established

[Contract 1](#)

---

**Roles**

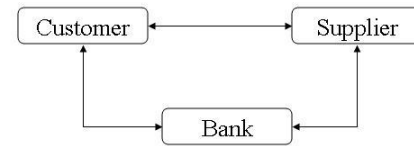
- **Supplier** – My company
- Customer – Company 3
- Deliverer – Company 4

**Channels**

---

**Name of the business network model**

Epoch 1 (current epoch)



**View**

[Epoch 2](#)

[Epoch 3](#)

Kuva 16: Sessionäyttö

Kenttä	Parametrin nimi tietoa esitettäessä	Toimintoparametri (action)	Toiminnon parametrit	Huomautukset
Session	session_name			
Tila (esim. established)	session_state			
Contract-linkki	contract_name	choose_contract	contract_id (ContractContent.contractID)	
Roles (supplier-sarake)	role_name	choose_role	role_name	
Roles (My)	company_name	choose_company	company_name	

company - sarake)	e		(ParticipantInfo.name)	
----------------------	---	--	------------------------	--

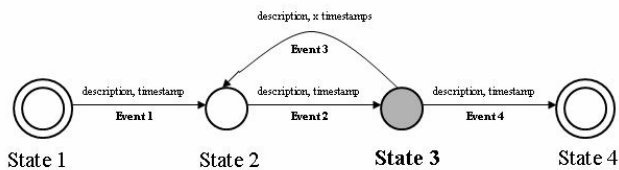
Taulukko 11: Sivun parametrit

## Roleprocessview.jsp

### Roleprocess 1, established

[Contract 1, Session 2](#)

- Roles**
- **Customer – My company** - Event 2
  - Supplier – Company 1 - Event 3
  - Event 1
  - Event 4



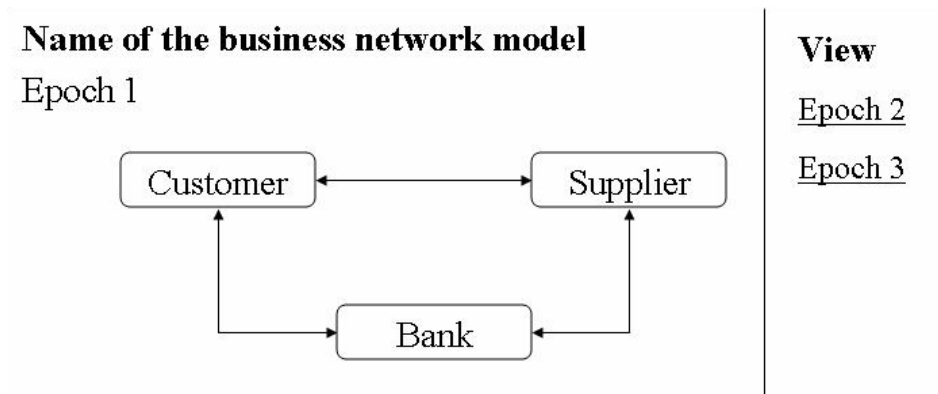
Kuva 17: Rooliprosessinäkymä

Kenttä	Parametrin nimi tietoa esitettäessä	Toimintoparametri (action)	Toiminnon parametrit	Huomautukset
Roleprocess	role_process_name			
state (esim. established)	role_process_state			
Contract- linkki	contract_name	choose_contract	contract_id (ContractContent.contra ctID)	
Session-linkki	session_name	choose_session	session_id	

			(ContractSession.session ID)	
Roles (customer-sarake)	role_name	choose_role	role_name	
Roles (My company - sarake)	company_name	choose_company	company_name (ParticipantInfo.name)	
Rooliprosessin visualisointikuva	role_process_image_url			

Taulukko 12: Sivun parametrit

## networkimage.jsp



Kuva 18: Verkoston rakennetta esittävä kuva

Kenttä	Parametrin nimi tietoa esitettäessä	Toimintoparametri (action)	Toiminnon parametrit	Huomautukset
Name of the business network model	network_model_name			

Verkoston visualisointikuva	network_image_url			
Epoch-linkki	epoch_name	choose_image_epoch	epoch_id	

Taulukko 13: Sivun parametrit

## 4 Luokat ja rajapinnat

Tässä luvussa esitellään toteutettavan järjestelmän sisäiset rajapinnat ja niihin liittyvät pakkaukset ja luokat.

### 4.1 *Actioncontroller-pakkaus*

Pakkaus sisältää luokat, joita käytetään JSP-sivujen kutsujen tulkintaan.

#### 4.1.1 ActionParameters

ActionParameters-luokka sisältää listauksen kaikista järjestelmän toimintoparametreista. Luokkaa käyttävät sekä JSP-sivut, että ActionController-luokka.

#### 4.1.2 ActionController

ActionController on Java Servlet –luokka. Se ottaa vastaan JSP-sivujen parametrit ja tulkitsee mitä DataHandlerin rajapinnan metodia kunkin toiminnon yhteydessä kutsutaan. Toiminto selvitetään ActionParameters-luokan toimintoparametrien indeksien avulla käyttäen swith-case-lausetta.

ActionController purkaa DataHandler-kutsun paluuarvona saatavan Value Objectin kentät HttpRequest-olioon käyttäen setParameter-metodia. Tämän jälkeen luokka kutsuu toimintoa seuraavaa JSP-näkymää näyttötilakaavion (kuva 5) mukaisesti.

### 4.2 *Datahandler-pakkaus*

Datahandler-pakkaukseen kuuluvat DataHandler-rajapinta apuluokkineen. Jokaista järjestelmän JSP-sivua kohden on yksi Value Object luokka, joka kapseloi kyseisen näkymän tietosisällön.

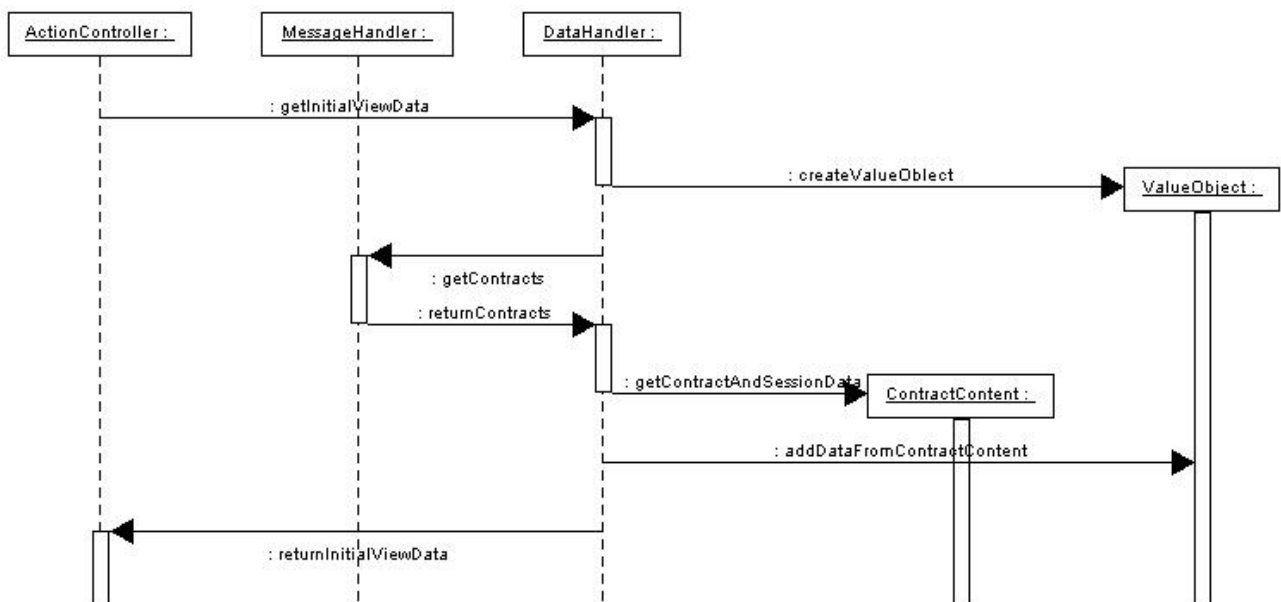
## 4.2.1 DataHandler.java

DataHandler tarjoaa rajapintansa ActionController-luokan käyttöön järjestelmän toimintojen suorittamiseksi. Lisäksi DataHandler pitää kirjaa kulloinkin käyttöliittymässä käsiteltävänä olevasta sopimuksesta ja sessiosta.

Toimintoihin liittyvät keskeiset sekvenssit on kuvattu metodikohtaisissa sekvenssikaavioissa rajapintakuvauksen yhteydessä.

### Rajapinta

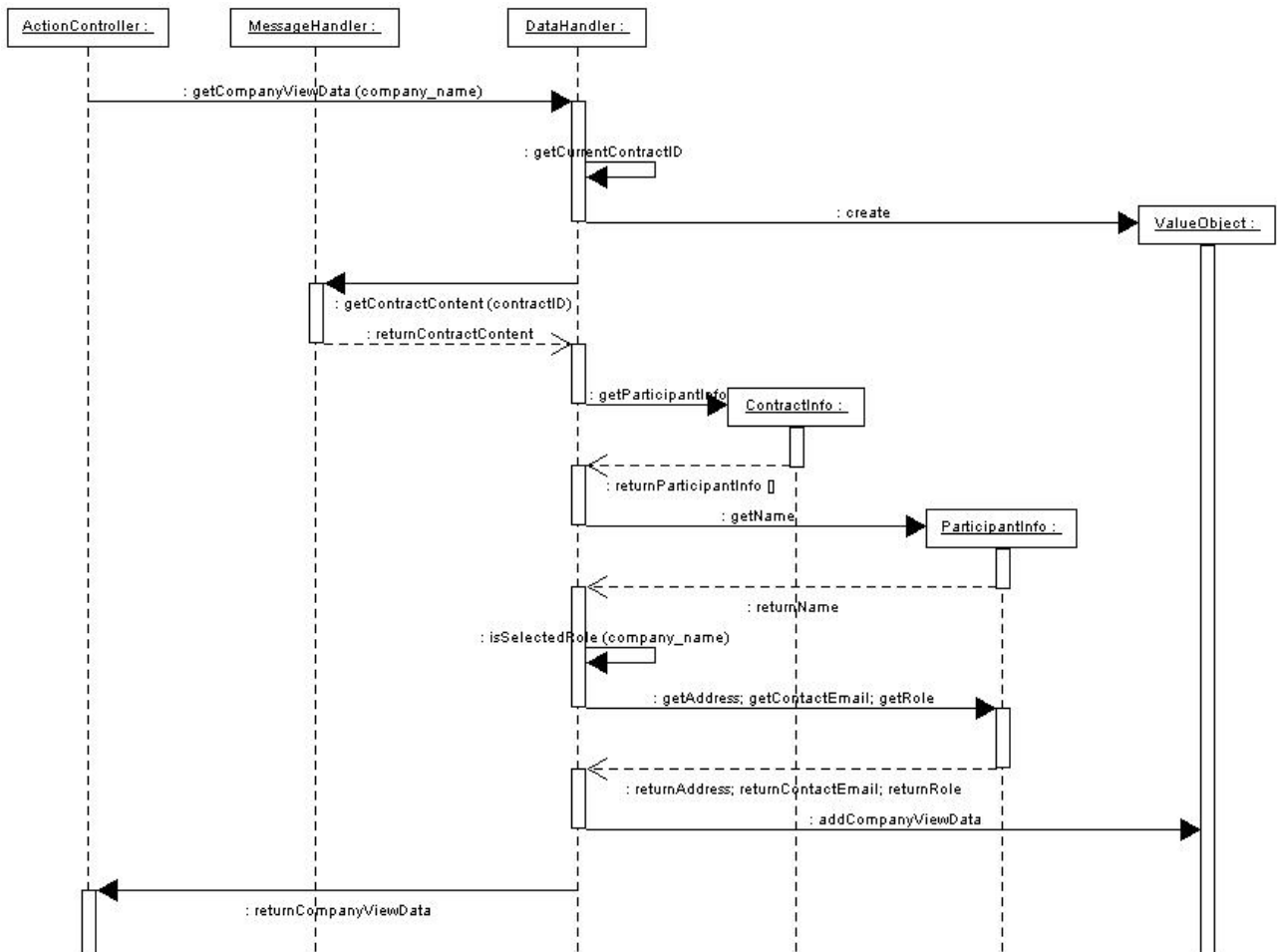
<i>getInitialViewData</i>	
Kuvaus	Hakee perusnäkyvän tietosisällön
Parametrit	
Parametrin tyyppi	
Paluarvon tyyppi	InitialViewValueObject-olio



Kuva 19: Sekvenssikaavio, getInitialViewData

<i>getContractViewData</i>
----------------------------

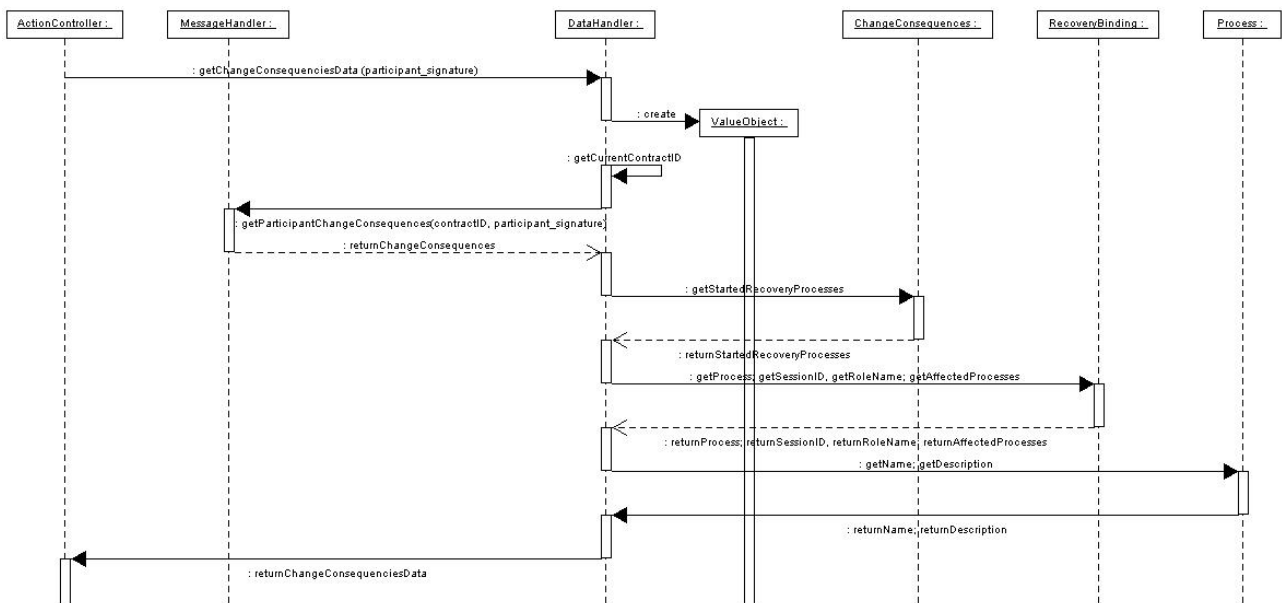
Kuvaus	Hakee sopimusnäytön tietosisällön
Parametrit	contract_id (ContractContent.contractID)
Parametrin tyyppi	String
Paluuavon tyyppi	ContractViewValueObject-olio



Kuva 20: Sekvenssikaavio, getContractViewData

<i>getCompanyViewData</i>	
Kuvaus	Hakee yrityskohtaisen tietosisällön
Parametrit	company_name (ParticipantInfo.name)
Parametrin tyyppi	String
Paluuavon tyyppi	CompanyViewValueObject-olio

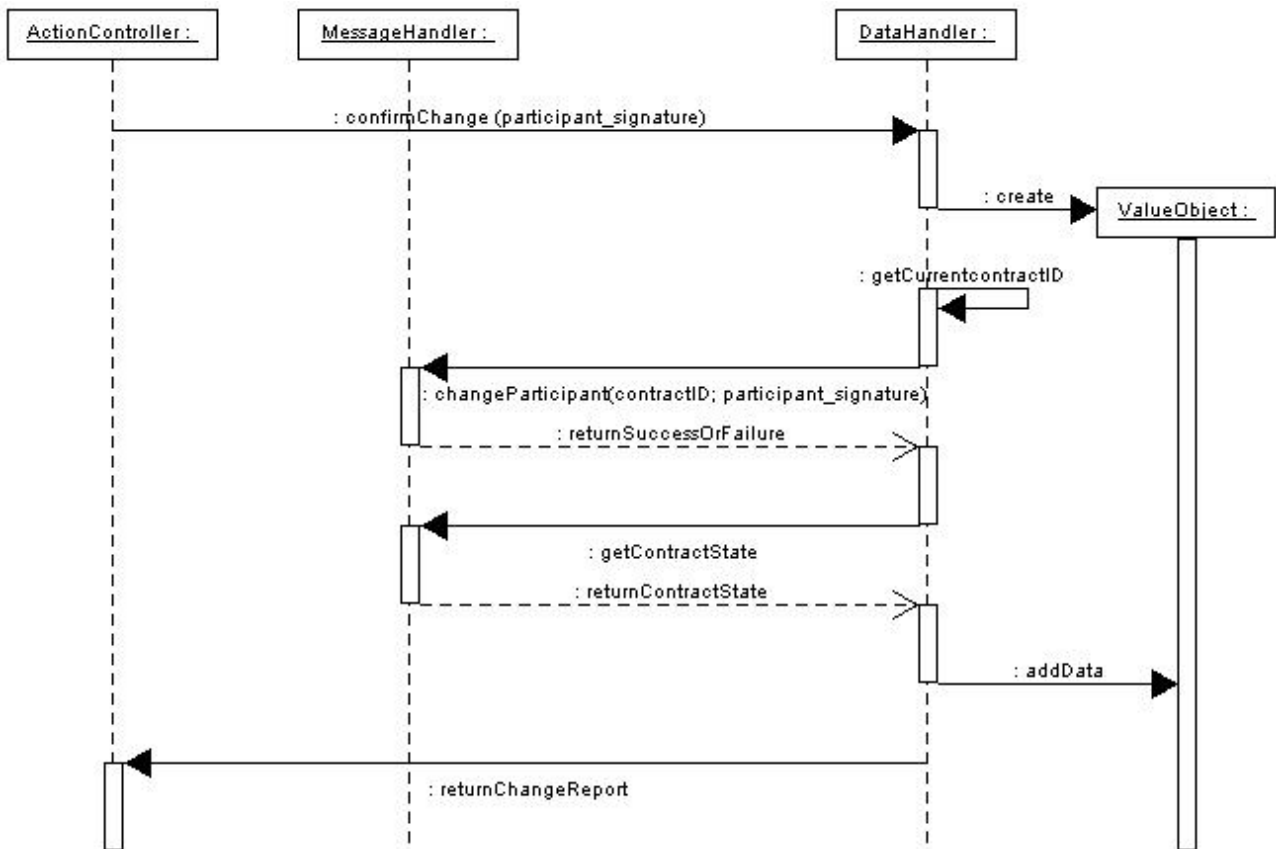
<b><i>getChangeConsequenciesData</i></b>	
Kuvaus	Hakee toimijan vaihdon seurauksiin liittyvän tietosisällön.
Parametrit	participant_signature (ParticipantInfo.signature)
Parametrin tyyppi	String
Paluuavon tyyppi	ChangeConsequenciesValueObject-olio



Kuva 21: Sekvenssikaavio, getChangeConsequenciesData

<b><i>confirmChange</i></b>	
Kuvaus	Vahvistaa toimijan vaihdon.
Parametrit	participant_signature (ParticipantInfo.signature)
Parametrin tyyppi	String
Paluuavon tyyppi	ChangeReportValueObject-olio





Kuva 22: Sekvenssikaavio, confirmChange

<b><i>getLeavingConsequencesData</i></b>	
Kuvaus	Hakee sopimuksesta poistumisen seurauksiin liittyvän tietosisällön.
Parametrit	participant_signature (ParticipantInfo.signature)
Parametrin tyyppi	String
Paluuavon tyyppi	LeavingConsequencesValueObject-olio
<b><i>confirmLeaving</i></b>	
Kuvaus	Vahvistaa sopimuksesta poistumisen.
Parametrit	participant_signature (ParticipantInfo.signature)
Parametrin tyyppi	String
Paluuavon tyyppi	LeavingReportValueObject-olio
<b><i>getSearchResultsData</i></b>	

Kuvaus	Hakee yrityshaun tuloksena kaikki yrityksen sopimukset ja niihin liittyvät sessiot.
Parametrit	company_identifier (substring yrityksen nimestä)
Parametrin tyyppi	String
Paluuavon tyyppi	SearchResultsValueObject-olio
<b><i>getSessionViewData</i></b>	
Kuvaus	Hakee sessionäyttöön liittyvät tiedot.
Parametrit	session_id (ContractSession.sessionID)
Parametrin tyyppi	String
Paluuavon tyyppi	SessionViewValueObject-olio
<b><i>getRoleProcessViewData</i></b>	
Kuvaus	Hakee rooliprosessinäyttöön liittyvät tiedot.
Parametrit	company_role1 (ParticipantInfo.role), company_role2 (ParticipantInfo.role)
Parametrin tyyppi	String
Paluuavon tyyppi	RoleProcessViewValueObject-olio
<b><i>getNetworkImageDataForCurrentEpoch</i></b>	
Kuvaus	Hakee verkostoa esittävän kuvan tiedot koskien oletusepokkia.
Parametrit	
Parametrin tyyppi	
Paluuavon tyyppi	String[] business_network_model_name, URL
<b><i>getNetworkImageDataForSpecifiedEpoch</i></b>	
Kuvaus	Hakee verkostoa esittävän kuvan tiedot koskien määriteltyä epokkia.
Parametrit	epoch_id
Parametrin tyyppi	String
Paluuavon tyyppi	String[] business_network_model_name, URL
<b><i>getActiveContractId</i></b>	

Kuvaus	Palauttaa tällä hetkellä seleiltavan sopimuksen id:n.
Parametrit	
Parametrin tyyppi	
Paluuavon tyyppi	String active_contract_id
<b><i>getActiveSessionId</i></b>	
Kuvaus	Palauttaa tällä hetkellä selailtavan session id:n.
Parametrit	
Parametrin tyyppi	
Paluuavon tyyppi	String active_session_id

Metodeihin getSessionViewData ja getRoleProcessViewData liittyvät sekvenssit on kuvattu dokumentin liitteenä olevassa sekvenssikaaviossa [Gli04].

### 4.3 Messagehandler-pakkaus

Messagehandler-pakkaukseen kuuluvat MessageHandler-rajapinta apuluokkineen.

Message Handler huolehtii kommunikaatiosta sopimusvarastojärjestelmään. Message Handlerin rajapinnan tarjoamat palvelut on tarkoitettu Data Handler -osajärjestelmän käyttöön. Seuraavassa esitellään rajapinnan tarjoamat palvelut, ja niiden tarvitsemat parametrit. Rajapinnan toteuttavan luokan tai luokkien yksityiskohtiin ei oteta tässä yhteydessä kantaa.

#### Rajapinta

<b><i>getContract</i></b>	
Kuvaus	Hakee yhden sopimuksen tietosisältön
Parametrit	Halutun sopimuksen tunnus
Paramerin tyyppi	String

Paluuavon tyyppi	ContractContent-olio
<b><i>getContracts</i></b>	
Kuvaus	Hakee kaikkien sopimusten tietosisältön
Parametrit	
Paramerin tyyppi	
Paluuavon tyyppi	ContractContent-taulukko
<b><i>getContractBNM</i></b>	
Kuvaus	Hakee sopimuksen liiketoimintaverkon mallin
Parametrit	Sopimuksen tunnus
Paramerin tyyppi	String
Paluuavon tyyppi	BusinessNetworkModel-olio
<b><i>getContractArcitectur</i></b>	
Kuvaus	Hakee sopimukseen liittyvän arkitehtuurin annetussa epokissa
Parametrit	Sopimuksen tunnus, epokki
Paramerin tyyppi	String
Paluuavon tyyppi	CommunityArcitecture-taulukon
<b><i>getContractMatrix</i></b>	
Kuvaus	Sopimuksen muodostama verkosto matriisin muodossa
Parametrit	Sopimuksen tunnus
Paramerin tyyppi	String
Paluuavon tyyppi	String-matriisi, jossa yhdellä rivillä solmupari joiden välillä kaari
<b><i>getContractPCC</i></b>	
Kuvaus	Sopimuksen toimijan vaihdon seuraukset
Parametrit	Sopimuksen tunnus, toimijan tunnus (signature)
Paramerin tyyppi	String, string
Paluuavon tyyppi	ChangeConsequenses-taulukko

<b><i>getContractPTC</i></b>	
Kuvaus	Sopimuksen toimijan poistumisen seurakset
Parametrit	Sopimuksen tuunus, toimijan tunnus (signature)
Paramerin tyyppi	String, string
Paluuavon tyyppi	ChangeConsequenses-taulukko
<b><i>execContractPC</i></b>	
Kuvaus	Suoritetaan toimijan vaihto
Parametrit	Sopimuksen ja toimijan tunnus
Paramerin tyyppi	String, string
Paluuavon tyyppi	Boolean
<b><i>execContractPT</i></b>	
Kuvaus	Suoritetaan toimijan poisto
Parametrit	Sopimuksen ja toimijan tunnus
Paramerin tyyppi	String, string
Paluuavon tyyppi	Process-taulukko
<b><i>execParticipantInfoUpdate</i></b>	
Kuvaus	päivitetään yrityksen tiedot
Parametrit	Tiedot yrityksestä, sopimus
Paramerin tyyppi	ParticipantInfo, string
Paluuavon tyyppi	boolean
<b><i>getProcessMatrix</i></b>	
Kuvaus	Prosessin esitys matriisin muodossa
Parametrit	Arkkitehtuurin ja roolin tunnus
Parametrin tyyppi	String, molemmat
Paluuarvon tyyppi	String-matriisi, jossa yhdellä rivillä lähtösolu, siirtymä ja maalisolmu
<b><i>getContractPRP</i></b>	

Kuvaus	Palauttaa kaikki sopimukseen liittyvät rooliprosessit
Parametrit	Sopimuksen tunnus
Parametrin tyyppi	String
Paluarvon tyyppi	Process-taulukko

#### **4.4 Visualizing-pakkaus**

Ohjelman käyttöliittymä tarjoaa myös visualisoinnin yleisesti tarkasteltavan verkoston tilasta (epokkikohtaisesti) sekä rooliprosessitilasta. Näistä ensimmäisen kohdalla tila esitetään ns. suuntaamattomana verkkona. Jälkimmäisen tila puolestaan esitetään ns. suunnattuna verkkona (automaattina).

Koko visualisointi toteutetaan käyttäen hyväksi verkosta ladattavaa GNU-lisenssin alaista JUNG – kirjastoa [Jun04].

Visualisoinnin tuottamista varten julkisessa rajapinnassa on kaksi metodia: yksi molempia eri malleja varten (suunnattu, suuntaamaton verkko). Tätä rajapintaa käyttää DataHandler muodostaessaan näkymäkohtaista tietosisältöä JSP-sivuja varten.

Molemmat metodit saavat parametrinaan String – taulukon, jossa esitetään verkon rakenne matriisina. Jokainen matriisin alkiopari muodostaa solmuparin, joka on yhdistetty verkon esityksessä kaarella. Kahdensuuntaiset kaaret esitetään toistuvina yhdensuuntaisina kaarina.

Jungin avulla tuotettu verkoston rakennetta esittävä kuva luetaan sen piirtoalustasta (Canvas) Javan omilla metodeilla, ja siitä muodostetaan JPG- tai PNG- muotoinen staattinen kuva. Tuotettu kuva tallennetaan joko kiinteästi levyille tai istuntokohtaiseen TEMP- muuttuun. Kuvan nimeämiseen käytetään satunnaisfunktiota tms.

Funktio palauttaa generoidun kuvan URL- osoitteen, jonka avulla visualisointi myöhemmin sijoitetaan JSP- sivulla haluttuun kohtaan.

## Rajapinta

<i>getNetworkImageURL</i>	
Kuvaus	Luo verkoston tilaa esittävän kuvatiedoston ja palauttaa tähän URL-viitteen
Parametrit	network_matrix
Paramerin tyyppi	String[][]
Paluuavon tyyppi	URL
<i>getRoleProcessImageURL</i>	
Kuvaus	Luo rooliprosessin tilaa esittävän kuvatiedoston ja palauttaa tähän URL-viitteen
Parametrit	network_matrix
Paramerin tyyppi	String[][]
Paluuavon tyyppi	URL

## 4.5 Habautils-pakkaus

Habautils-pakkaus pitää sisällään järjestelmän lokin kirjoituksen, ja tarvittavien asetustiedostojen lukemisen sekä rajapinnan näiden tietojen käyttöä varten.

Lokin kirjoitus hoidetaan writeLog-metodilla, joka saa parametrinaan lokiin kirjoitettavan viestin. Kirjoitettava viesti tulee yhdelle riville. Lokiin kirjoitettavia tietoja ei määritellä tässä yhteydessä.

Asennustiedostosta luettuja asetuksia voidaan käsitellä eri getSetting-metodeilla. Metodi palauttaa asennustiedostossa muuttujalle määritellyn arvon.

## 5 Lähdeluettelo

[Jun04]

Jung-grafiikkakirjaston kotisivu, <http://jung.sourceforge.net> (3.11.2004)

[Mäk04]

Antti Mäen PowerPoint-esitys käyttöliittymästä,  
<http://www.cs.helsinki.fi/group/haba2004/docs/UI/kali8.ppt>

[Gli04]

Riina Glinskihin piirtämä sekvenssikaavio liittyen DataHandler-luokan metodeihin getSessionViewData ja getRoleProcessViewData,  
<http://www.cs.helsinki.fi/group/haba2004/docs/su01/SequenceDiagram-getSessionRolesAndParticipants-getRoleProcess.jpg>