

# Sosiaalihuollon asiakastietojen ja -asiakirjojen tietomallinnus

---

*Prosessi, suunnitteluperiaatteet ja työkalut*

---

*4.5.2010*

---

Riikka Huttunen  
Konstantin Hyppönen  
Antero Lehmuskoski  
Virpi Hotti  
Jaana Nevalainen  
Päivi Tossavainen  
Erja Ailio  
Aki Miettinen

---

Versio	Päiväys	Kohdat	Muutoksen sisältö	Tekijät
0.1	21.7.2009	koko doku	Rakenteen kasaamista	RH
0.2	30.7.2009	koko doku	Sisällön lisäilyä	RH
0.3	15.8.2009	koko doku	Sisällön lisäilyä	RH
0.4	12.11.2009	koko doku	Rakenteen ja sisällön muokkausta	RH
0.6	10.2.2010	koko doku	Lisää tekstiä, muokkailua	RH
0.7	11.3.2010	koko doku	Rakenteen muokkausta	RH, VH
0.8	20.4.2010	koko doku	Lisää tekstiä ja muokkailua	kaikki
0.9	23.4.2010	koko doku	Rakenteen muokkausta	VH, RH
0.10	29.4.2010	koko doku	Kommentointi	VH
0.11	30.4.2010	Luvut 2 ja 5	Sisällön lisäilyä	AL, KH
0.12	4.5.2010	koko doku	Kommentointi ja tarkennukset	JN, KH, AL, VH

SOSIAALIALAN TIETOTEKNOLOGIAHANKE  
SOSIAALI- JA TERVEYSMINISTERIÖ

Suomen kuntaliitto  
Terveiden ja hyvinvoinnin laitos  
Itä-Suomen sosiaalialan osaamiskeskus  
Itä-Suomen Yliopisto

# Sisältö

---

1	Johdanto .....	5
2	Sosiaalihuollon käsitteelliset erityispiirteet .....	7
2.1	Sosiaalipalvelut ja asiat .....	7
2.2	Toimijat ja niiden roolit .....	9
2.3	Asiakas .....	11
2.4	Perhe ja vanhemmuus .....	13
3	Tietomallit .....	16
3.1	Tiedon mallinnuksen prosessi .....	18
3.2	Tietomallinnuksen tasot .....	20
3.3	Tietomallinnusmenetelmien vertailu .....	23
3.3.1	E-R-malli .....	24
3.3.2	UML-luokkakaavio .....	24
3.3.3	Ontologia .....	25
3.3.4	XML Schema .....	26
3.3.5	Tietomallien vertailu .....	27
4	Tietomallinnus Tikesos-hankkeessa .....	28
4.1	Word-dokumenteista Excel-taulukoihin .....	29
4.2	CCTS tietomallinnuksessa .....	30
4.3	Tietomallinnuksen ja prosessimallinnuksen yhteys .....	31
5	Sosiaalihuollon asiakastietojen tietomallinnus .....	38
5.1	Tietosisältöjen yhtenäistäminen .....	39
5.2	Luokitukset .....	42
5.3	Tietokomponenttitaulukot .....	45
5.4	CCTS ja tietokomponenttitaulukot .....	48
5.5	Tietokomponenttien tarkentaminen .....	50
5.5.1	CCTS-mukainen tarkentaminen .....	50
5.5.2	Periytyminen .....	52
5.5.3	Substituutioryhmät ja lisäykset .....	53

5.5.4	Yhteenveto tarkentamisen keinoista.....	56
5.6	Roolien esittäminen tietomallissa .....	57
6	Sosiaalihuollon asiakasasiakirjojen tietomallinnus.....	59
6.1	Asiakirjarakenteet .....	60
6.2	Sisällölliset asiakirjarakennetaulukot.....	63
6.3	Sisällöllisten asiakirjarakenteiden määrittelyprosessi .....	64
6.4	CCTS ja asiakirjataulukot.....	67
6.5	Tekniset asiakirjarakenteet eli skeemat.....	69
7	Yhteenveto.....	72
	LÄHTEET .....	75
A	Tietomallien vertailu.....	79
B	Toimeentulotukipäätöksen asiakirjarakennetaulukko.....	87

# 1 Johdanto

Sosiaalihuolto on lainsäädäntöön pohjautuvaa toimintaa, jota harjoittavat pääasiassa kunnat ja yksityinen sektori kuntien toimeksiannosta. Sosiaalihuollossa on yli 20 palvelutehtävää, joista tunnetumpia ovat toimeentulotuen myöntäminen, lastensuojelu, päihdehuolto, vammaispuolvelut ja kehitysvammaisten erityishuolto.

Sosiaalihuollossa käsitellään suuria määriä asiakkaita koskevia tietoja, joiden käsitteellinen sisältö vaihtelee palvelutehtävittäin. Asiakasta koskevat tiedot tallennetaan asiakastietojärjestelmien tietokantoihin, joista tuotetaan sosiaalihuollon asiakasasiakirjoja kuten toimeentulotukipäätös. Sosiaalihuollossa on tunnistettu yli 150 asiakasasiakirjaa.

Sosiaalihuollossa asiakastiedoilla tarkoitetaan asiakkaista tallennettavia asiakirjallisia tietoja, joissa on asiakkaiden henkilötietojen lisäksi tietoa myös asiakkaiden läheisistä (kuten perhe) sekä sosiaalihuollon ammatillisista henkilöistä. Sen lisäksi asiakastiedoissa on tietoa asiakasta koskevista hallintoasioista, asiakkaille tuotettavista palveluista ja asiakkaiden saamista sosiaalietuuksista. Tikesos-hankkeessa on tarkoitus mallintaa niitä tietoja, joita tarvitaan sosiaalihuollon tehtävien hoitamisessa. Tällä tarkoitetaan sosiaalihuollon asiakasasiakirjoja, ei hallinnon lomakkeita.

Tällä hetkellä sosiaalihuollon asiakastietomallit ja asiakasasiakirjat ovat pääsääntöisesti järjestelmä- tai kuntakohtaisia. Ainakin asiakastietojärjestelmien tietokantalomakkeiden valintalistat ovat kunnittain erilaiset, sillä kunnat ovat parametrisoineet esimerkiksi meno-, tulo- ja varallisuuslajit kuntakohtaisesti. Tietosisällöt ovat epäyhtenäisiä sekä rakenne- että semantiikkatasolla, mikä hankaloittaa tietojen siirtämistä järjestelmien välillä.

Tikesos-hankkeen tavoitteena on mallintaa sosiaalihuollon palvelutehtäviin liittyvien asiakastietojen yhtenäinen käsitteellinen sisältö ja tietojen väliset keskinäiset suhteet. Sosiaalihuoltoon liittyy useita käsitteellisiä erityispiirteitä, joita on avattu luvussa 2. Tuotoksena syntyy sosiaalihuollon asiakastietomalli (luku 5), joka esittää tietojen yhtenäisen käsitteellisen sisällön kuvaavien tietokomponenttien määrittelyt (luku 5.1) ja tietojen väliset keskinäiset suhteet. Tietomallin mallinnusprosessi kuvataan luvussa 3.1. Sen lisäksi Tikesos-hankkeessa mallinnetaan asiakasasiakirjat (asiakasasiakirjojen mallinnusprosessi kuvataan luvussa 6). Tuotoksena syntyvät sisällölliset asiakirjarakenteet (luku 6.2 ja liite B) ja tekniset asiakirjarakenteet eli XML skeemat (luku 6.5). Hankkeen tietomallinnuksen käytäntöjä ja taustatekniikoita on kuvattu luvussa 4.

Tietomallit ovat abstrakteja malleja, joiden tarkoitus on kuvata, kuinka tietoa voidaan esittää ja käyttää tehokkaasti. Tietomallinnus puolestaan viittaa prosessiin, jolla tietoa rakenteistetaan ja järjestetään. Tietomallin avulla esitetään tietyn sovellusalan ymmärrystä sekä kuvataan organisaatiossa käsiteltävää tietoa. Sosiaalihuollossa tekniset asiakirjarakenteet ja niissä käytettävät tietokomponentit ovat määritelty XML Schema -kielellä. Luvussa 3.3 on tutkittu tietokomponenttien keskinäisten suhteiden mallintamiseksi seuraavia kuvaustekniikoita: E-R-kaaviot,

UML-luokkakaaviot, ontologiat ja XML Schema. Teoriatasolla tietomalleja ja tietomallinnusta käsitellään luvussa 3.

Tikesos-hankkeessa sosiaalihuollon asiakastietojen mallintaminen on aloitettu vuonna 2008 kaikille asiakirjoille yhteisistä perustiedoista, jotka koskevat henkilöitä ja organisaatioita eri toimijaroleissa. Vuonna 2009 kehittämiskokonaisuuteen kuului sosiaalihuoltoon soveltuvan tietomallin määrittely, sosiaalihuollon eri palvelutehtävissä käsiteltävien asiakastietojen mallintaminen ja edelleen palvelukohtaisten tietojen harmonisoiminen siltä osin kuin eri tehtävissä tarvitaan samansisältöisiä tietoja. Samoja asiakastietoja hyödynnetään eri palvelutehtävissä ja eri asiakirjatyypeissä.

## 2 Sosiaalihuollon käsitteelliset erityispiirteet

---

Sosiaalihuollon asiakastietomalli laaditaan kuvaamaan kaikkia sosiaalihuollon asiakastyöstä talteen otettavia asiakastietoja. Mallin tulee siten kattaa kaikki sosiaalihuoltoon sisältyvät palvelut ja niissä myönnettävät etuudet, kaikki sosiaalihuollon asiakastyöhön osallistuvat toimijat, kaikki erilaiset roolit, joissa nämä toimijat esiintyvät asiakirjoissa sekä toimintaan oleellisesti liittyvät artefaktit, kuten asiakirjat. Tavoitteena on, että käsitteellinen malli kuvaa mahdollisimman hyvin nimenomaan sosiaalihuollon toimintaympäristöä ja siinä tapahtuvaa toimintaa. Siksi asiakastietomallia varten on tehtävä käsitteanalyysia siitä, mitä erityispiirteitä käytettävillä käsitteillä on sosiaalihuollon toimintaympäristössä.

Sosiaalihuollossa kansalaisille tarjottavia palveluja leimaa kiinteä yhteys lainsäädäntöön, jolla palvelujen järjestämistä ja toteutusta säädellään. Erityisesti sosiaalihuoltoon säädetyistä laeista sosiaalihuoltolaki (SHL 1982) ja sosiaalihuollon asiakaslaki (AsL 2000) ovat niin sanottuja sosiaalihuollon yleislakeja. Näiden lisäksi toimintaa ohjataan lukuisilla erityislaeilla. Kukin erityislaki tarkentaa sosiaalihuoltolaissa kunnan sosiaalitoimelle asetettuja velvoitteita sosiaalipalvelujen järjestämisestä yhden palvelutehtävän osalta. Siksi sosiaalihuoltoa kuvataan tietomallissa palvelutehtävien ja niissä annettavien sosiaalipalvelujen kautta.

Tässä luvussa luodaan katsaus Sosiaalihuollon asiakastietomalliin kuuluvista keskeisistä käsitteistä. Käsittemallin ylin taso muodostuu toimijoista, niiden rooleista ja toiminnasta, jota sosiaalihuollossa toteutetaan. Luvussa 2.1 luodaan katsaus siihen, miten sosiaalihuoltoa jäsennetään tehtävien, palvelujen ja palveluprosessien avulla sekä identifioimalla käsiteltäviä asioita. Luvussa 2.2 kuvataan sosiaalihuollon kontekstissa esiintyviä toimijoita sekä niiden rooleja eri yhteyksissä. Seuraavissa luvuissa 2.3 ja 2.4 käsitellään esimerkkeinä sosiaalihuollon toimijoista kahta keskeistä käsitettä, jotka esiintyvät sekä laeissa että käytännön asiakastyössä. Sekä asiakkaan että perheen käsite on rajoiltaan usein epämääräinen, koska niitä määritellään eri tavoin eri yhteyksissä.

### 2.1 Sosiaalipalvelut ja asiat

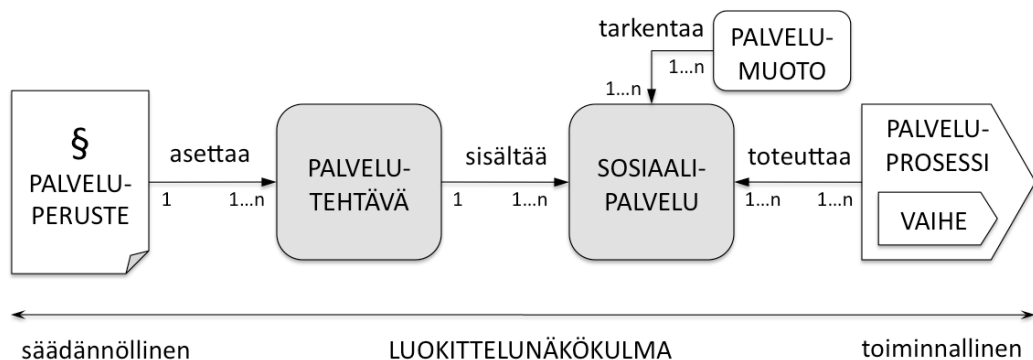
---

Sosiaalihuollon asiakasasiakirjojen konteksti muodostuu palvelunjärjestäjän ja palveluntuottajan toiminnasta sosiaalipalvelujen ja sosiaalietuuksien antamiseksi asiakkaille. Kontekstia kuvaavilla tiedoilla on keskeinen merkitys asiakirjallisen tiedon hallinnassa ja asiakirjoihin kohdistuvissa hauissa. Sosiaalihuollon asiakastietomallissa palvelujen toteuttamisen ympäristöä kuvataan rakenteisen tiedon avulla sekä asiakirjojen sisällöissä että niihin liitettävissä kuvailutiedoissa. Asiakirjojen syntykontekstia kuvaavat tiedot varustetaan sekä yksilöivillä että luokittelevilla tunnuksilla.

Asiakaslain 6. pykälässä säädetään, että sosiaalihuollon järjestämisen tulee perustua viranomaisen tekemään päätökseen. Pykälän tarkoituksena on parantaa asiakkaan oikeusturvaa selkeyttämällä palvelujen aloittamisen menettelyä ja tekemällä niitä koskevaa päätöksentekoa näkyvämmäksi asiakkaalle (HE 137/1999). Sosiaalihuollon asiakkaille siis myönnetään etuuk- sia ja järjestetään palveluja päätöksillä, jotka perustuvat tiettyihin lainkohtiin. Tämä asiakasla- kiin kirjattu periaate voi osaltaan selkeyttää asiakkuuden hämäryyttä sosiaalihuollossa, ensisi- jaisesti tietysti asiakkaalle, mutta myös tietomallinnuksen kannalta. Jokainen sosiaalihuollon asiakasasiakirja liitetään asiaan, jonka käsittelyyn tai käsittelyn seurauksena toteutettuun pal- veluun se liittyy.

Jotta yksittäisen asiakkaan jokainen vireille tullut asia pystytään erottamaan kaikista muista sosiaalihuollossa käsiteltävistä asioista, kukin asia on varustettava yksilöivällä tunnisteella, joka liittää yhteen kaikki siihen asiaan palveluprosessien eri vaiheissa kuuluvat asiakirjat. Sosi- aalihuollon kontekstia luokittelevan tiedon osalta hankkeessa on muun muassa laadittu Sosiaa- lipalvelujen luokitus, joka sisältää useita erillisiä, mutta toisiaan täydentäviä toimintaa ja sen organisoimista kuvaavia luokituksia. Sosiaalipalvelujen luokitusta täydennetään vielä pienem- millä, palveluja tarkentavilla luokituksilla. Tietomalli rakentuu siis palvelutoiminnan osalta sitä kuvaavista luokittelevista ja yksilöivistä tiedoista.

Sosiaalipalvelujen luokitus sisältää palvelutehtävien luokituksen, sosiaalipalvelujen luokituksen ja sosiaalihuollon palveluprosessien luokituksen. Kullakin luokituksella on erilaiset luokittelu- perusteet, koska ne jäsentävät sosiaalipalveluja eri näkökulmista. Kuvassa 1 on esitetty sosiaa- lipalvelujen luokituksen sisältämät erilliset luokitukset ja niiden suhteet toisiinsa. Kuvassa esitetään myös eri luokitusten luokitteluperusteet säädännöllisen ja toiminnallisen näkökul- man jatkumona.



Kuva 1 Sosiaalipalvelujen luokituksen rakenne ja luokittelunäkökulmat

Sosiaalihuoltoa ohjaava lainsäädäntö toimii luokitteluperusteena sosiaalihuollon palvelutehtä- ville. Palvelutehtävät muodostuvat pääasiassa sosiaalihuollon erityislakien perusteella, mutta joillekin tehtäville ei ole olemassa erillislakeja, vaan niitä hoidetaan sosiaalihuollon yleislakien perusteella. Kukin palvelutehtävä perustuu yhteen palveluperusteeseen eli lakiin tai muuhun toimintaa ohjaavaan normiin. Toisaalta yhdessä palveluperusteessa voidaan asettaa kunnalle myös useampia kuin yksi palvelutehtävä.

Sosiaalihuoltoa toteutetaan käytännössä palveluprosesseina ja niiden vaiheina. Palveluproses- seja luokitellaan siten toiminnallisesta näkökulmasta tarkastelemalla sosiaalihuollon ammatti-

henkilöiden työhön kuuluvia työprosesseja asiakastyössä. Tästä työstä tuotoksina syntyvät toimintaprosessien kuvakset eri palvelutehtävissä. Sosiaalihuollon palveluprosesseja mallinetaan osana Tikesos-hankkeen toiminta-arkkitehtuurin kuvausta, ja prosessien luokittelua on tehty yhteistyössä hankkeen prosessimallinustyöryhmän kanssa. Palveluprosessien ja niiden vaiheiden luokittelun luokitusperusteena on siis käytännön työn toteutus ja luokittelunäkökulma on toiminnallinen. Sosiaalipalvelut on luokiteltu yhdistämällä säädännöllinen ja toiminnallinen näkökulma. Kukin palvelutehtävä sisältää vähintään yhden sosiaalipalvelun. Kunnalle asetettu tehtävä edellyttää aina yhden tai useamman sosiaalipalvelun järjestämistä kunnassa esiintyvää tarvetta vastaavasti. Jokainen sosiaalipalvelu kuuluu vähintään yhden palvelutehtävän hoitamiseen mutta yhteen palvelutehtävään voi sisältyä useita sosiaalipalveluja. Kunkin sosiaalipalvelun toteuttaminen edellyttää vähintään yhtä palveluprosessia. Palveluprosesseja voi kuitenkin sisältyä useita kuhunkin sosiaalipalveluun jo siitä syystä, että kunkin sosiaalipalvelun toteuttamiseen voi osallistua useita eri toimijatahoja. Kunnan tehtävänä on järjestää sosiaalipalveluja mutta sen ei ole välttämättä tuotettava niitä itse. Myös samassa kuntaorganisaatiossa sosiaalipalvelun järjestäjä ja tuottaja voivat olla eri palveluyksiköt, jolloin niillä on erilliset palveluprosessinsa.

Sosiaalipalveluja on joissakin tapauksissa syytä luokitella tarkemmin kuin yleisellä tasolla. Palvelumuodoilla tarkoitetaan sosiaalipalveluja tarkentavia luokituksia, jotka on määritelty palvelutehtäväkohtaisesti, koska sosiaalipalveluja kuvataan eri palvelutehtävissä hyvinkin erilaisilla tavoilla. Esimerkiksi lastensuojelussa tarvittavaa tarkennetta siitä, kuuluuko sosiaalipalvelu avo-, sijais- vai jälkihuoltoon, ei tarvita muissa palvelutehtävissä. Palvelumuodot siis muodostavat useita luokituksia, joita hyödynnetään ainakin palvelukohtaisissa asiakasasiakirjoissa kuvaamaan mitä tahansa sellaista sosiaalipalvelun toteutusta kuvaavaa lisämäärettä, joka on syytä kirjata näkyviin. Tähän luokitusperheeseen on pyritty sisällyttämään keskeisimmät palvelumuotoja kuvaava luokitukset. Tosiasiassa palvelumuotoja kuvaavia luokituksia tarvitaan asiakirjoissa todennäköisesti enemmänkin. Luokituksia voi lisätä luokitusperheeseen sitä mukaa kuin luokitustarpeita tunnistetaan ja luokituksia valmistuu. Pienempiä erillislukituksia laaditaan osana Tikesos-hankkeen asiakirjojen sisällön yhtenäistämistyötä. Luokitusperhe on tarkoitettu mahdollisimman joustavaksi ja siihen voi ehdottaa lisättäväksi esimerkiksi tilastoinnissa, tutkimuksessa tai johtamisessa tarvittavia luokituksia.

## 2.2 Toimijat ja niiden roolit

---

Sosiaalihuollon asiakastyössä toimijoina on organisaatioita ja henkilöitä. Palveluja järjestetään ja useimmiten tuotetaan osana eri organisaatioiden toimintaa. Organisaatiolla tarkoitetaan tiettyä tarkoitusta varten luotua sosiaalista järjestelmää, jolla on määritellyt tavoitteet ja tarkoituksellisesti luotu rakenne (Lehmuskoski ja Kuusisto-Niemi 2008, 65). Sosiaalihuoltoa toteutetaan sekä yksityisissä että julkisissa organisaatioissa. Kunnallisille organisaatioille on säädetty sosiaalihuoltoon kuuluvia tehtäviä. Henkilöistä tallennettavien tietojen sisältö riippuu suuresti siitä, osallistuvatko he toimintaan yksityishenkilöinä vai ammattilaisina työtehtäviinsä liittyen. Siksi henkilötoimijoina esitetään yksityishenkilö ja ammatillinen henkilö, jotka esiintyvät erilaisissa rooleissa eri tilanteissa, kuten organisaatiotkin. Rooleilla ilmaistaan toimijan tehtävät siinä kontekstissa, johon asiakirja liittyy. Kontekstilla tarkoitetaan tässä siis sosiaalihuollon palvelutehtävää, sosiaalipalvelua, palveluprosessia sekä yksittäisen asian käsittelyä. Roolit ovat yleensä ajasta riippuvaisia, ja näin ollen väliaikaisia. Toimijalla voi olla useitakin rooleja samas-

sa sosiaalipalvelussa, myös yhtäaikaaisesti. Esimerkiksi henkilö voi olla toisen henkilön huoltajana (eli laillisena edustajana) ja samalla äitinä, hakemuksen laatijana ja sosiaalihuollon asiakkaana. Organisaatio voi toimia esimerkiksi palvelunjärjestäjänä tai palveluntuottajana. Palveluntuottaja on myös esimerkki roolista, jossa voi toimia joko yksityishenkilö (esimerkiksi perhehoitaja), ammatillinen henkilö (esimerkiksi ammatinharjoittaja) tai organisaatio. (Huttunen ym. 2009)

Rooli voi sisältää ominaisuuksia, joilla kuvaillaan toimijoita ja olosuhteita, joissa se voi esiintyä. Rooleja käytetään siksi, että samaa tietoelementtiä voidaan käyttää eri toimijan tietona. Tämä on varsinkin useamman toimijan verkostossa tärkeää, sillä toimijan rooli saattaa myös näyttäytyä erilaisena eri yhteistyötahoille riippuen siitä, millaisia yhteyksiä toimijoilla on keskenään. Kun päästään yhteisymmärrykseen kunkin henkilön roolista eri yhteyksissä, eri roolit voidaan määritellä omina tietotyypeinään ja viitata niistä samaan henkilöön. Erilaisia toimijarooleja tunnustetaan hankkeen dokumentaatiotyöryhmien laatimista sisältömäärityksistä. Sosiaalihuollon eri palvelutehtävissä ja eri asiakirjoissa toimijoilla on erilaisia rooleja, joiden tunnistaminen ei aina ole yksinkertaista. Niiden esiintymistä ja keskinäisiä suhteita havainnollistetaan siksi ristiintaulukointien avulla UML-luokkakaavioiden lisäksi. Taulukossa 1 esitetään sosiaalihuollon palveluissa esiintyvät toimijat sekä roolit, joissa ne voivat esiintyä sosiaalihuollon asiakirjoissa. (Huttunen ym. 2009)

Taulukko 1. Toimijat ja niiden roolit

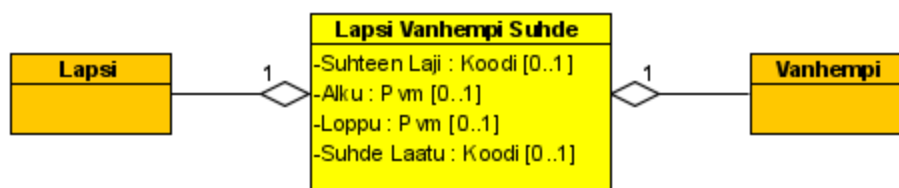
	Ammatillinen henkilö	Yksityishenkilö	Organisaatio	Abstrakti rooli
Asiakas		X		X
Asiantuntija	X	X		X
Edunvalvoja		X		
Edunvalvonnan alainen		X		X
Edunvalvontavaltuutettu		X		
Edunvalvontavaltuuttaja		X		X
Hakija		X		X
Hakijan puoliso		X		X
Huollettava		X		X
Huoltaja		X		
Laatija	X	X	X	X
Lapsi		X		X
Lisätietojen antaja	X	X		X
Palvelunjärjestäjä			X*	X
Palveluntuottaja	X	X	X*	
Puoliso		X		
Päätöksentekijä	X			X
Samassa taloudessa asuva		X		X
Sitoumuksen saaja	X	X		X
Suostumuksen antaja	X	X		X

Suunnitelmaan osallistuva	x	x		x
Vastuhenkilö	x			x
Takaaja		x		x
Tiedoksiantaja	x	x		x
Tiedoksisaja		x		x
Tutkiva taho	x		x	x
Työnantaja		x	x	x
Työryhmä	x			x
Valmistelija	x			x
Vanhempi		x		x
Vastaanottaja	x	x	x	x
Yrittäjä		x		

Roolien ja toimijoiden välisten kytkösten lisäksi sosiaalihuollossa esiintyy roolien välisiä suhteita (taulukko 2), kuten lapsen ja vanhemman välinen suhde. Roolisuhteissa on aina kaksi osapuolta, jotka osallistuvat suhteeseen. Kuvassa 2 esitetystä lapsen ja vanhemman välisessä suhteesta esiintyy roolit lapsi ja vanhempi. Taulukosta 1 voidaan todeta, että kumpikin näistä rooleista voi kuulua ainoastaan yksityishenkilölle. Yleisesti tietomallinnuksessa ei ole käytetty roolien viittauksia toisiin rooleihin, vaan roolista viitataan aina varsinaiseen toimijaan. Roolisuhteet ovat kuitenkin tästä säännöstä poikkeus. Käsitelmässä roolisuhdet esiintyvät omana yläluokkana, joka on yhteydessä toimijaroleihin.

Taulukko 2. Roolien välisiä suhteita

	Osapuoli 1	Osapuoli 2
Edunvalvontasuhte	Edunvalvoja	Edunvalvonnan alainen
Edunvalvontavaltuutussuhde	Edunvalvontavaltuutettu	Edunvalvontavaltuuttaja
Huoltajuussuhde	Huoltaja	Huollettava
Lapsi-vanhempi suhde	Lapsi	Vanhempi
Parisuhde	Puoliso	Puoliso



Kuva 2. Esimerkki roolisuhdeista UML-luokkakaaviona

## 2.3 Asiakas

Sosiaalihuoltolaissa ei määritellä sosiaalihuollon asiakasta tai asiakkuutta. Sosiaalihuollon asiakaslaisa asiakkaaksi määritellään sosiaalihuoltoa hakeva tai käyttävä henkilö (AsL 2000, 3 §).

Yleensä sosiaalihuoltoa käyttävä henkilö on sama kuin se, joka on sitä hakenut. Joissakin tapauksissa hakijana ja käyttäjänä voi kuitenkin olla eri henkilö. Sosiaalihuoltoa hakevaksi ei tässä yhteydessä ole mielekäästä lukea hakemuksen laatijaa, jos se laaditaan toisen henkilön puolesta, vaan henkilö, jolle tai jonka puolesta palvelua tai etuutta tosiasiallisesti haetaan. Hakemuksen voi esimerkiksi vajaavaltaisen puolesta täyttää läheinen tai vaikka sosiaalihuollon ammattihenkilö, joka ei siis ole sosiaalihuollon asiakas. Toisaalta sosiaalihuoltoa hakevan ja käyttävän henkilön erottaminen toisistaan ei ole oleellista asiakkaan määritelmän kannalta. Oleellista siinä on se, että asiakkuus alkaa jo hakemisvaiheessa ja jatkuu niin kauan kuin sosiaalihuollon palveluja tai etuuksia saadaan. Asiakkuus voi kuitenkin alkaa hakemisen lisäksi myös muulla tavoin, esimerkiksi ilmoituksesta. Hallinnollisen asian käsittelyn kannalta asiakkuus alkaa asian vireillepanosta, joka voi tapahtua eri tavoin. Siksi asiakaslaissa esitetty määritelmä ei sinänsä ole kattava sosiaalihuollon asiakkaan käsitelmää.

Asiakaslaissa asiakas ja hänen laillinen edustajansa määritellään asianosaisiksi viranomaisen asiakirjojen sisältämän asiakasta koskevan henkilötiedon osalta (AsL 2000, 11 §). Kaikki asianosaiset eivät kuitenkaan ole välttämättä asiakkaita. Henkilötietolaki lähestyy samaa asiaa rekisteröidyn näkökulmasta. Jokaisella rekisteröidyllä on oikeus hänestä tallennettuun henkilötietoon. Henkilötiedolla tarkoitetaan tässä "kaikenlaisia luonnollista henkilöä taikka hänen ominaisuuksiaan tai elinolosuhteitaan kuvaavia merkintöjä, jotka voidaan tunnistaa häntä tai hänen perhettään tai hänen kanssaan yhteisessä taloudessa eläviä koskeviksi" (HetiL 1999, 3 §). Sosiaalihuollon asiakirjoihin saatetaan tallentaa henkilötietoja sellaisistakin henkilöistä, jotka eivät ole asiakkaita tai asianosaisia kyseisessä asiassa. Tietosuojavaltuutetun kannanoton mukaan asiakasasiakirjoihin voidaan tallentaa tietoja sivullisista henkilöistä, jos se on tarpeellista annettavan palvelun kannalta. Tämä ei kuitenkaan tee henkilöstä rekisteröityä eikä hänellä ole oikeutta asiakirjaan, vaan kaikki asiakasrekisteriin tallennettu tieto koskee sitä henkilöä, joka on asiakkaana kyseisessä tapauksessa. Tietosuojavaltuutetun kannanoton mukaan vain asiakkaat ovat henkilötietolain tarkoittamia rekisteröityjä.

Asiakaslain perusteluissa todetaan, että vaikka sosiaalihuoltolaissa ei määritellä asiakasta, sen määritelmä voidaan johtaa sosiaalihuollon soveltamisalaa koskevasta 1. pykälästä, jonka mukaan "sosiaalihuollolla tarkoitetaan sosiaalipalveluja, toimeentulotukea, sosiaaliavustuksia ja niihin liittyviä toimintoja, joiden tarkoituksena on edistää ja ylläpitää yksityisen henkilön, perheen ja yhteisön sosiaalista turvallisuutta ja toimintakykyä" (AsL 2000, 1 §).

Toiseksi asiakaslain perusteluissa todetaan, että asiakkuutta määrittelee lain 13. pykälä, jonka mukaan "kunnan on huolehdittava sosiaalihuoltoon kuuluvina velvollisuuksina sosiaalipalvelujen järjestämisestä asukkailleen ja toimeentulotuen antamisesta kunnassa oleskelevalle henkilölle." Lisäksi lain 15 §:ssä järjestämisvelvollisuutta laajennetaan kiireellisissä tapauksissa myös muihin kuin kunnan asukkaisiin.

Asiakaslain yksityiskohtaisissa perusteluissa asiakkaalla tarkoitetaan sosiaalihuoltoa hakevaa, käyttävää tai sen kohteena olevaa henkilöä. Perusteluissa todetaan: "Sosiaalihuollossa asiakkuus on monitahoinen ja asiakkuus-käsite on tarpeen nähdä sisällöltään laajana. Asiakkuus kattaa kaikki ne henkilöt, jotka hakevat ja käyttävät sosiaalihuollon palveluja tai saavat sosiaalihuollon etuuksia. Asiakkaita ovat myös ne, jotka saavat esimerkiksi palveluja ja etuuksia koskevaa ohjausta tai neuvontaa. Asiakkuus aktualisoituu tietyn toimenpiteen, hakemuksen, päätöksen, ilmoituksen tai palvelun kautta. Määritelmä tarkoittaa myös sellaisia asiakkuussuhteita, joissa asiakas omasta tahdostaan riippumatta tulee sosiaalihuollon piiriin." Lain perusteluissa asiakkuus nähdään hyvin laajasti: se kattaa myös ne henkilöt, jotka saavat palveluja ja etuuksia

koskevaa ohjausta tai neuvontaa. Samoin esimerkiksi lapsi, josta on tehty lastensuojeluilmoitus ja jota koskeva asia otetaan käsittelyyn, tulee lastensuojelun asiakkaaksi, vaikka hän itse ei hae eikä varsinaisesti käytäkään sosiaalihuoltoa.

Asiakaslaisissa mainittu suunnitelmantekovelvollisuus ei edellä mainitusta syystä rajaa asiakkuuden käsitettä. Laissa todetaan: "sosiaalihuoltoa toteutettaessa on laadittava palvelu-, hoito-, kuntoutus- tai muu vastaava suunnitelma, jollei kyseessä ole tilapäinen neuvonta ja ohjaus tai jollei suunnitelman laatiminen muutoin ole ilmeisen tarpeetonta" (AsL 2000, 7 §). Suunnitelmaa ei siis laadita kaikille asiakkaille, mikäli hänen saamansa ohjaus tai neuvonta voidaan katsoa tilapäiseksi, eikä asiakkuutta määritetä sen perusteella, onko asiakkaalle laadittava suunnitelma.

Asetuksessa sosiaalihuollon asiakasmaksuista määritellään palvelun käyttäjäksi yksi tai useampi henkilö, joka käyttää sosiaali- ja terveydenhuollon palvelua tai johon palvelu kohdistuu (AmL 1992, 2 §). Erityisesti mainitaan kotipalvelu ja lasten päivähoito, joissa palvelun käyttäjäksi katsotaan lapsen ohella hänen kanssaan yhteistaloudessa elävät vanhemmat tai muut huoltajat. Asiakasmaksuasetus on iältään vanhempaa säädöskantaa kuin asiakaslaki.

Asiakaslain perusteluissa todetaan lisäksi, että sosiaalihuollon erityislainsäädännössä asiakas määritellään säätämällä siitä, kenelle laissa tarkoitettuja palveluja tai etuuksia annetaan. Asiakkuutta ei välttämättä määritellä *lain tarkoituksen* määrittelyn yhteydessä, vaikka siinä useinkin mainitaan, kenen etua lailla pyritään turvaamaan. Asiakkuuden määrittelyssä oleellista on se, kuka laissa määritellään palvelun tai etuuden saajaksi. Saman logiikan mukaan sosiaalihuoltolain tarkoitusta koskevaa pykälääkään ei siis pitäisi tulkita niin, että sekä yksityinen henkilö, perhe ja yhteisö ovat sosiaalihuollon asiakkaita sen perusteella, että lain tarkoituksena on heidän sosiaalisen turvallisuutensa ja toimintakykynsä edistäminen ja ylläpitäminen. Samalla tavoin myöskään pelkkä kunnan alueella asuminen ei tee henkilöstä sosiaalihuollon asiakasta, vaan pikemminkin määrittelee kunnalle asetetun järjestämisvelvollisuuden rajat. Sosiaalihuollon asiakkaan tarkempaa määrittelyä ei siksi pysty tekemään sosiaalihuoltolain perusteella, vaan se joudutaan tekemään erityislakien perusteella palvelutehtäväkohtaisesti.

Sosiaalihuollon erityislaeissa palvelujen ja etuuksien saajia – ja siten myös asiakkaita – määritellään tarkemmin palvelutehtäväkohtaisesti. Asiakkaan mallintamisessa lähdetään liikkeelle siitä, että asiakas on aina yksilö, kuten sosiaali- ja terveysministeriön julkaisemassa oppaassa todetaan (STM 2001). Joissakin palvelutehtävissä saman palvelun tai etuuden saajaksi on kuitenkin katsottava useampi kuin yksi henkilö. Nämä tapaukset mallinnetaan palvelutehtäväkohtaisesti.

## 2.4 Perhe ja vanhemmuus

---

Perhe määritellään sosiaalihuollon laeissa eri tavoin. Lähtökohtana perheen määrittelylle on lapsen vanhemmuuden ja huoltajuuden käsitteet. Lapsen huollosta ja tapaamisoikeudesta annetun lain mukaan "lapsen huoltajia ovat hänen vanhempansa tai henkilöt joille lapsen huolto on uskottu" (LhtL 1983). Lapsen vanhemmista puhutaan monissa sosiaalihuollon säädöksissä ja muissa yksityisoikeuden alaan kuuluvissa säädöksissä. Laeissa ei määritellä sitä, tarkoittaanko vanhemmuudella pelkästään biologista vanhemmuutta vai sisältääkö se myös ottovanhemmuuden. Ainakin laissa (AdoL 1985) ja asetuksessa (AdoA 1997) lapseksiottamisesta van-

hemmuudella tarkoitetaan nimenomaan biologista vanhemmuutta ja ottovanhemmuus pidetään erillisenä käsitteenä, mikä on lain tarkoituksen kannalta tärkeää. Muissa yhteyksissä ei selkeästi tehdä eroa vanhemman ja ottovanhemman roolien välille. Lapsille järjestettävässä sosiaalihuollossa on kuitenkin useimmiten tärkeää erottaa nämä roolit toisistaan. Siksi ottovanhemmuus on syytä mallintaa vanhemmuuden alakäsitteeksi. Myös väestötietojärjestelmässä ottovanhemmuus on omana tietonaan.

Vanhemmuus liittyy säädöksissä poikkeuksetta alaikäiseen henkilöön. Se ei siis sisällä vanhemmuutta aikuisten ja täysivaltaisten ihmisten keskinäisenä henkilösuhteena. Tuleekin selvittää, onko vanhemmuus täysi-ikäisten vajaavaltainen kohdalla tarpeellinen tieto esimerkiksi kehitysvammaisten erityishuollossa. Laissa kehitysvammaisten erityishuollosta säädetään kehitysvammahuollon palveluksista myös kehitysvammaisen henkilön vanhemmille, perheenjäsenille, huoltajille ja muille läheisille.

Yleisessä kielenkäytössä perheellä on perinteisesti tarkoitettu henkilöitä, joita sitoo yhteen biologinen sukulaisuus. Aiemmin perheeseen saattoi kuulua useita sukupolvia, jotka myös usein asuivat samassa taloudessa. Biologinen sukulaisuussuhde on nykyisinkin perheen määrittelyn lähtökohta, sillä se koostuu lähtökohtaisesti vanhemmista ja heidän lapsistaan. Perheellä tarkoitetaan useimmiten kuitenkin ns. ydinperhettä, johon kuuluu vain kaksi sukupolvea, ja isovanhemmat jätetään perhemäärittelyn ulkopuolelle. Lisäksi perheen käsitteellinen sisältö on laajentunut tarkoittamaan enenevässä määrin myös muita kuin biologisesti keskenään sukua olevia henkilöitä, jotka asuvat samassa taloudessa. Yleisessä kielenkäytössä puhutaan ns. uusperheistä, jolloin biologisella sukulaisuudella ei aina ole ratkaisevaa merkitystä perheenjäsenten välisten suhteiden määrittämisessä.

Sosiaalihuollon säädöksissä perhe määritellään asiakasmaksulaissa, asiakasmaksuasetuksessa ja laissa toimeentulotuesta. Näiden lisäksi perheestä puhutaan useissa säädöksissä, kuten lastensuojelulaissa sekä laissa ja asetuksessa lasten päivähoitosta. Niissä ei kuitenkaan määritellä tarkemmin, mitä perheellä kussakin yhteydessä tarkoitetaan.

Asetuksessa sosiaali- ja terveydenhuollon asiakasmaksuista perhe on määritelty tarkoittamaan ”yhteistaloudessa avioliitossa tai avioliitonomaisissa olosuhteissa eläviä henkilöitä sekä molempien samassa taloudessa eläviä alaikäisiä lapsia” (AmA 1992). Myös asiakasmaksulain 7a § on muutettu yhteensopivaksi asetuksen kanssa perheen määrittelyn osalta (AmL 1992). Lakimuutoksen perusteluissa todetaan muutoksen yhdenmukaistavan perheen käsitteen lasten päivähoiton maksun määräytymisen osalta. Säädöksen voi tulkita tarkoittamaan puolisoita ja kummankin biologista lasta tai ottolasta. Toimeentulotuesta annetussa laissa perhe määritellään samalla tavalla (TtL 1997, 3 §). Laissa lasten kotihoidon ja yksityisen hoidon tuesta tarkennetaan perheen määritelmä kattamaan ”yhteistaloudessa eläviä vanhempia tai muita huoltajia sekä vanhemman tai muun huoltajan kanssa yhteistaloudessa avioliitossa tai avioliitonomaisissa olosuhteissa elävää henkilöä alaikäisine lapsineen” (LkhL 1996, 2 §). Perhe laajennetaan tässä siis koskemaan myös muita lapsen huoltajia kuin vanhempia.

Perhe siis määritellään sosiaalihuollon säädöksissä yhteistalouden, vanhempi-lapsi-suhteen ja parisuhteen avulla. Lasten kotihoidon tuen ja yksityisen hoidon tuen osalta määritelmään on liitetty myös lapsen huoltajuus, mutta muissa säädöksissä määritelmä kattaa vain lapsen vanhemmat, joiden oletetaan olevan myös lapsensa huoltajia, kuten asia useimmiten onkin. Kaikkia näitä käsitteitä on siksi pidettävä perhettä määrittävinä ominaisuuksina sosiaalihuollon toimin-

takontekstissa. Perheen mallintamista vaikeuttaa kuitenkin palvelutehtäväkohtaisten määrittelyjen yhteensopimattomuus.

### 3 Tietomallit

Tietomallin päätarkoituksena on tukea informaatiojärjestelmän kehittämistä määrittelemällä tiedon muotoa ja merkitystä. Tietomalli on silta reaali maailman informaatiosta tietokantoihin (tai ylipäätään sähköiseen muotoon) talletettavaan tietoon (kuva 3).



Kuva 3. Tietomallin muodostuminen

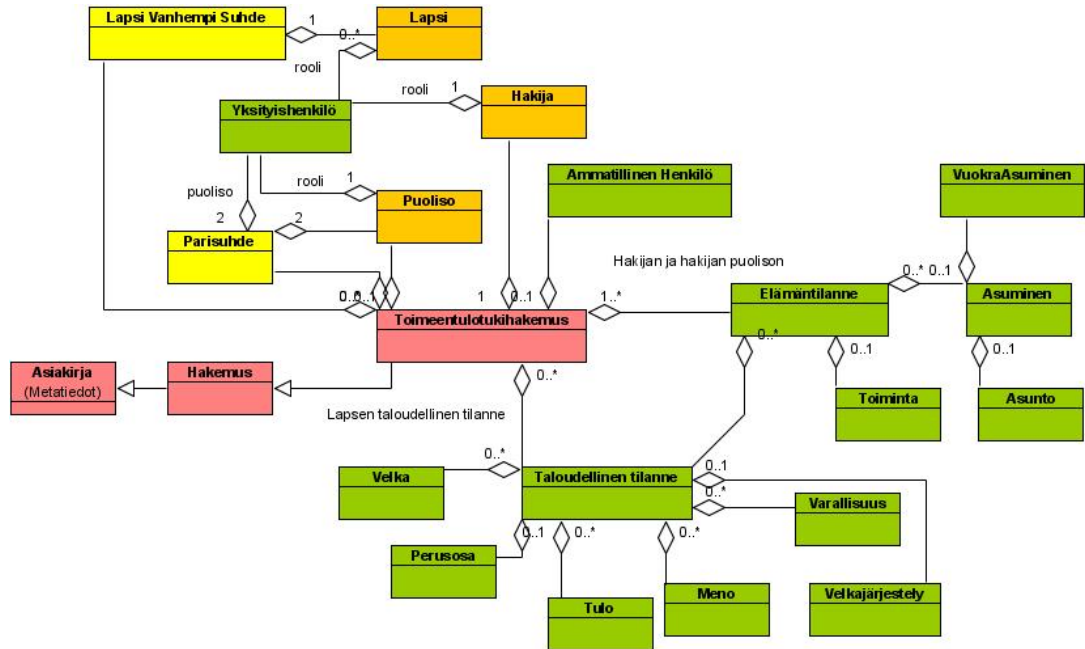
Tieto voi olla joko asiaa ihmisen vastaanottamana ja ymmärtämänä (*informaatiota*) tai konkreettisesti esitysmuodossa ilmaistuna (*dataa*) (Tietotekniikan liitto 2008, 326). *Tietoelementti* (data element, ts tietoalkio) on yksittäinen tieto, tai useiden tietojen muodostama yksikkö (Tietotekniikan liitto 2008, 52). Useista tietoelementeistä muodostuu *tietorakenne* (data structure), joka on joukko tietoja, jotka on järjestetty yhdeksi kokonaisuudeksi (Tietotekniikan liitto 2008, 335).

*Tietomalli* (data model) määrittelee formaalisti tietoelementtejä ja elementtien välisiä suhteita tietyssä sovellusalueessa. Formaalisuus tarkoittaa sitä, että tietomalli noudattaa tiettyä yhteisesti sovittua muotoa, kuten jotain tietomallinnustekniikkaa. Tietomalli auttaa käyttäjiä tai avainhenkilöitä ymmärtämään selkeästi organisaation tietovaatimuksia sekä havainnollistamaan, mitkä tietorakenteet ovat tarpeen organisaation toiminnan tukemiseen. Tietomallin tulee ilmaista liiketoimintaprosessien läpi käytettävien tietoelementtien sisältöä, muotoa, koosta ja sääntöjä. Mallissa on ilmaistuna mitä tietoa vaaditaan, ja miten tietoa pitäisi järjestellä. Se ei välttämättä heijastu siihen, mitä operaatioita tiedolla tulisi olla tehtävissä. (Ponniiah 2007) Usein tietomallien ohella voidaan puhua myös *domain modelista* (ts *domain object model*), joka on malli sovellusalueesta, jolla organisaatio toimii. Korkean tason tietomalli (käsitelmä) vastaa domain modelia.

Tietomalli toimii kriittisenä työkaluna käyttäjien ja avainhenkilöiden kanssa kommunikoinnissa sekä tietokantasuunnittelijoille luonnoksena tietokantajärjestelmää varten. *Tietokanta* (database) sisältää tiettyä kohdetta kuvaavan tietojen kokoelman, joita tietojärjestelmä käyttää ja päivittää (Tietotekniikan liitto 2008, 330). Tietomalli vastaa samanaikaisesti käyttäjien ja kehittäjien tarpeita (Ponniiah 2007).

Kuvassa 4 on esitetty esimerkki tietomallista. Kuvasta käy ilmi, millaista tietoa (tietoelementtejä, luokkia) Toimeentulotukihakemuksessa kysytään, ja mitkä tiedot liittyvät toisiinsa. Esimerkiksi Taloudelliseen tilanteeseen liittyviä elementtejä ovat muun muassa

Tulo, Meno ja Varallisuus. Elementtien väliset suhteet voivat olla erilaisia; tässä kuvassa on käytetty vain UML-notaation mukaista periytymissuhdetta ja koostumussuhdetta.



Kuva 4. Esimerkki tietomallista: Toimeentulotukihakemuksen tietoelementit UML-luokkakaaviona.

Tietomallinnuksen luonteeseen kuuluu, että samasta aihealueesta voidaan tehdä useita erilaisia tietomalleja. Sen takia tulisikin tiedostaa ne seikat, jotka muodostavat hyvän tietomallin, sekä kriteerit, joiden avulla rinnakkaisia tietomalleja voidaan verrata. Tietomallin perimmäisenä tarkoituksena on kuitenkin tukea mahdollisimman hyvin koko tietojärjestelmän suunnittelua ja sen sisältämiä tietovaatimuksia. (Simsion ja Witt 2005)

Tietomallin tärkeimmät ominaisuudet ovat seuraavat (Simsion ja Witt 2005):

- *Täydellisyys.* Tukeeko malli kaikkea sitä tietoa, mitä tietovaatimukset edellyttävät?
- *Toistumattomuus.* Onko kullekin tiedolle vain yksi paikka tietomallissa?
- *Liiketoimintaprosessien toteuttaminen.* Kuinka hyvin liiketoimintaprosesseissa esiintyvät säännöt toteutuvat tietomallissa?
- *Tiedon uudelleenkäyttö.* Ovatko tietomalli tai sen pienemmät osat uudelleenkäytettävissä eri tarkoituksissa kuin mihin ne oli alun perin suunniteltu?
- *Vakaus ja joustavuus.* Kuinka hyvin tietomalli vastaa muuttuviin tietovaatimuksiin?
- *Eleganttius.* Tarjoaako tietomalli käytännöllisen ja yksinkertaisen tiedon jaottelun?
- *Kommunikaatio.* Kuinka tehokkaasti tietomalli tukee kommunikointia järjestelmän suunnittelussa mukana olevien avainhenkilöiden kesken?
- *Integraatiot.* Miten tietomalli toimii yhdessä organisaation muun arkkitehtuurin kanssa?

Tietomallin suunnittelussa täytyy pyrkiä tasapainoon näiden ominaisuuksien kanssa, koska kaikkia edellä mainittuja ominaisuuksia ei voida saavuttaa samanaikaisesti ilman ristiriitoja.

Luvussa 3.1 esitellään lyhyesti prosessi, jota käytetään yleisesti tiedon mallinnuksessa. Luvussa 3.2 kerrotaan tietomallinnuksen tasoista ja eritasoisista tietomalleista. Luvussa 3.3 esitellään suosituimmista tiedon mallinnustekniikoista E-R-kaaviot, UML-luokkakaaviot, ontologiat sekä XML Schema, Tiedon mallinnustekniikoista on tehty myös vertailu (liite A), jonka tuloksia luvussa 3.3 esitellään.

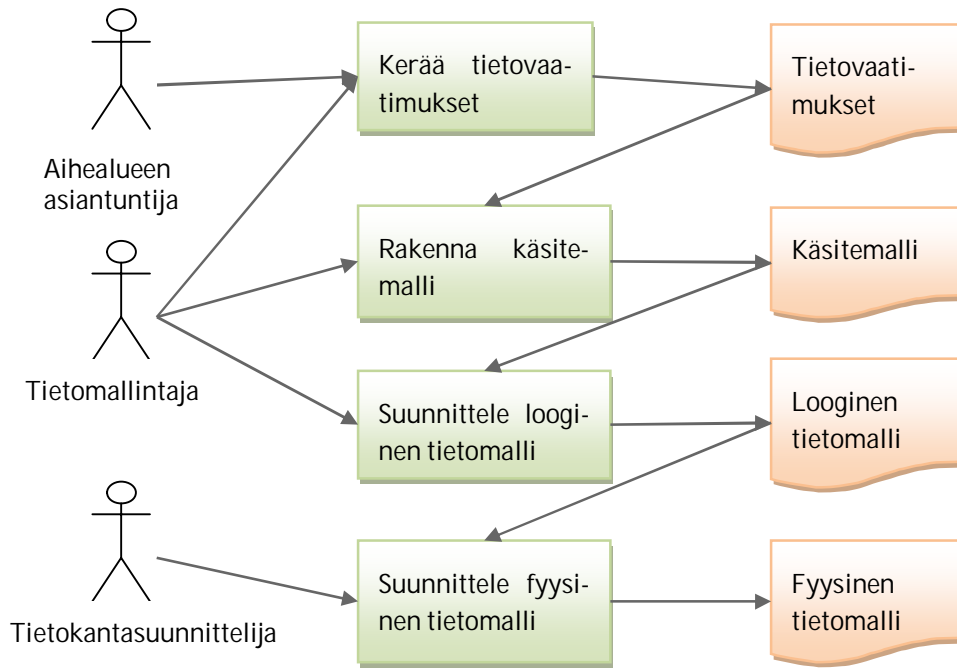
### 3.1 Tiedon mallinnuksen prosessi

---

Tietomallin luominen on haastava prosessi, johon tarvitaan sekä sisällön asiantuntijoiden että teknisten asiantuntijoiden panostusta. *Tiedon mallinnuksella* (data modeling) kuvataan reaali maailman tiedon vaatimuksia organisaatioiden avainhenkilöiden ymmärtämässä muodossa. Tietomallit koostuvat yleensä seuraavista komponenteista:

- *objektit* tai *entiteetit* (objects or entities); mitä tietoelementtejä mallinnettavassa kohteessa on?
- *attribuutit* tai *ominaisuudet* (attributes or characteristics); mitä ominaisuuksia tietoelementeillä on, millä ne erotetaan toisistaan?
- *tunnisteet* (identifiers); miten kukin tietoelementti yksilöidään?
- *suhteet* (relationships); mitkä ovat tietoelementtien väliset yhteydet?
- *symbolit* (simple symbols); miten kaikki edellä mainitut komponentit kuvataan havainnollisesti?

Kuvassa 5 havainnollistetaan eritasoisten tietomallien tuottamisessa mukana olevat toimijat sekä mallinnustehtävien tuotokset.



Kuva 5. Tietomallinnuksen toimijat, tehtävät ja tuotokset (mukailten Simsion ja Witt 2005)

Tietomallin tuottaminen alkaa organisaation tietotarpeiden selvittämisestä, jossa liiketoiminnan vaatimukset ja aihealueen asiantuntijoiden tietämys ovat avainasemassa. Näistä koostetut Tietovaatimukset toimivat lähdemateriaalina käsittemallin rakentamiselle. Käsittemallia tarkentamalla päästään Loogiseen tietomalliin, josta taas siirtyminen Fyysiseen tietomalliin on kohtuullisen suoraviivainen tehtävä. Fyysisen tietomallin suunnittelee yleensä Tietokantasuunnittelija, jolla on tietämystä käytetystä tietokannasta ja sen hallintajärjestelmästä. Tietomallintaja työstä ja ohjaa kaikkia muita vaiheita paitsi fyysisen tietomallin suunnittelua.

Yleisen tietomallinnusprosessin vaiheet ovat seuraavat (Ponniiah 2007):

1. *Tunnista liiketoimintaobjektit* (identify business objects). Tässä vaiheessa tunnistetaan ne liiketoimintaan liittyvät objektit, joista tarvitaan tietomalli. Yleensä tunnistaminen tapahtuu tutkimalla liiketoiminnan kannalta olennaisia prosesseja. Prosesseja tutkimalla saadaan selville kussakin prosessissa tarvittavat tietoelementit (eli prosessin tietotarpeet). Kootusta tietoelementtistauksesta täytyy tämän jälkeen vielä ryhmitellä samaan asiaan liittyvät tietoelementit. Näin tunnistetaan tietoelementeistä liiketoimintaobjekteja (käsitteitä).
2. *Tunnista suorat suhteet* (identify direct relationships). Edellä tunnistetut liiketoimintaobjektit ovat yhteyksissä toisiinsa. Tutkimalla objekteja suhteessa toisiin objekteihin löydetään suoria suhteita, jotka voidaan nimetä sekä määritellä tarkemmin (kuten kardinaliteetti, pakollisuus ym).
3. *Lisää attribuutit* (add attributes). Attribuutit kuvaavat liiketoimintaobjekteja, joten ne ovat olennaista tietoa tietomallissa. Vaiheessa 1 tehdystä tietoelementtien listauksesta voidaan muodostaa attribuuttilistoja kullekin niin ikään vaiheessa 1 löydetyille liiketoimintaobjektille.

4. *Määäää tunnisteet* (assign identifiers). Tunnisteet ovat yleensä attribuutteja tai attribuuttijoukkoja, joiden arvot yksilöivät objektin instansseja. Tässä vaiheessa kullekin liiketoimintaobjektille valitaan tunnisteeksi jokin sen (tunnisteeksi sopivista) attribuuteista, tai mikäli sellaista attribuuttia ei ole, sellainen luodaan.
5. *Sisällytä liiketoimintasäännöt* (incorporate business rules). Liiketoimintasäännöt hallitsevat objektien, suhteiden ja attribuuttien käyttötapoja. Kardinaliteettien esittäminen kuvaa yhdenlaisia liiketoimintasääntöjä.
6. *Validoi tietomalli* (validate the data model). Tässä vaiheessa halutaan varmistaa, että tietomalli oikeasti kuvaa organisaation tietovaatimuksia. Sen lisäksi samalla tarkastetaan tietomallin täydellisyys ja yksinkertaisuus, jotta sitä voidaan käyttää käyttäjien kommunikointityökaluna.

Hyvin samankaltaista prosessia käytetään muun muassa sanastotyössä, käsiteanalyysissä ja ontologian rakentamisessa. Lähtökohtana ovat sovellusalalla käytetyt käsitteet, joita tutkimalla (käsiteanalyysillä) selvitetään käsitteiden sisältöä (kuten määritelmät, attribuutit, rajoitteet ja käyttötarkoitus) ja käsitteiden välisiä suhteita.

## 3.2 Tietomallinnuksen tasot

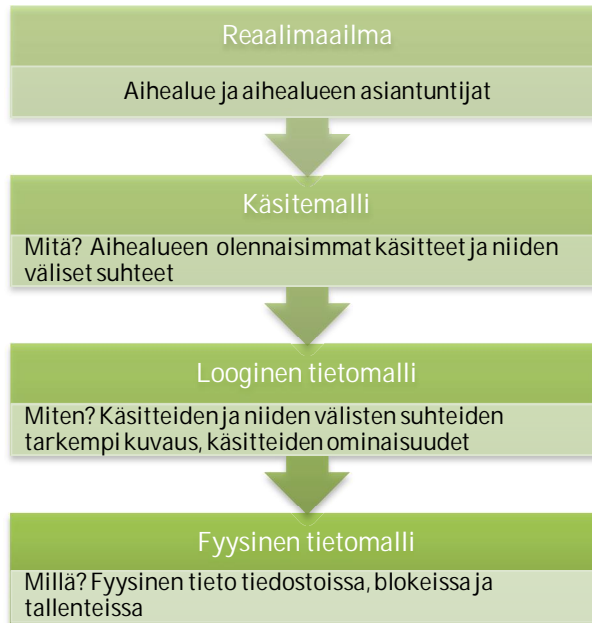
---

Tietomallit kuvaavat organisaation tietovaatimuksia eri tarkkuustasoilla. Tietomallinnuksen tarkoituksena on sekä toimia apuna kommunikoinnissa käyttäjien kanssa että toimia ohjeistuksena tietokannan rakentamiselle. Tietomalli vastaa samanaikaisesti käyttäjien ja kehittäjien tarpeita. Tietomallin tulee olla korkealla ja yleisellä tasolla, jotta käyttäjät ymmärtävät mallia ja malli toimii kommunikoinnin apuna – toisaalta taas tietomallin tulee olla tarpeeksi yksityiskohdainen, jotta sen perusteella voidaan tehdä tietokanta. Yksi tietomalli ei voi olla samaan aikaan yleisellä ja yksityiskohtaisella tasolla, joten erillisiä tietomalleja tuotetaan tiedon eri tasoille. (Ponniiah 2007)

Tietomallinnuksen tasot voidaan johtaa loogisesti *tiedon tasoista* (information levels), joita voidaan tunnistaa tietovaatimuksista. Tiedon tasot ovat seuraavat (mukaiillen Ponniiah 2007):

- *Käsitetaso* (ts sovellustaso, conceptual level): tietosisällöt, ihmisen käsitys sovellusalueesta, tietojärjestelmä käyttäjien ymmärtämänä.
- *Rakennetaso* (ts looginen taso, logical level): yksityiskohdat tietosisällöistä, ei yksityiskohtia tiedon tallennusympäristöstä.
- *Talletustaso* (ts fyysinen taso, physical level): yksityiskohdat tiedon tallennusympäristöön liittyen, yksityiskohdat tietokannan toteuttamiseen.

Näiltä eri tiedon tasoilta tuotetaan eritasoisia tietomalleja, vastaten tiedon tasoja: käsitetasolta käsittemalli, rakennetasolta looginen tietomalli ja talletustasolta fyysinen tietomalli. Tietomallien eri kuvaustasot on esitetty kuvassa 6:



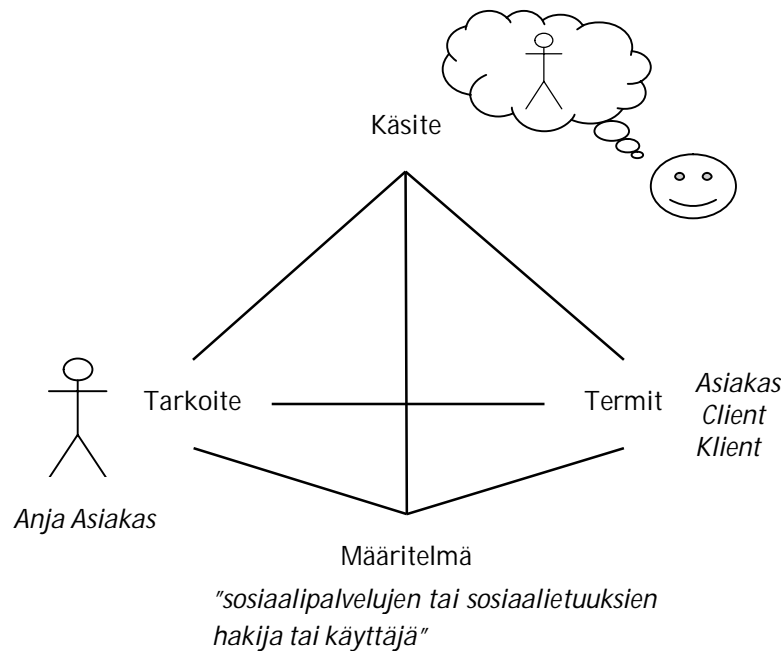
Kuva 6. Tietomallinnuksen eri tasot

Reaalimaailman tasolta saadaan tietovaatimuksia käsitelmän rakentamiseksi. Käsitelmän tasolla tietomalli ei ole sitoutunut mihinkään tekniikkaan tai ohjelmistoon, eikä rajoita tietokannan toteuttamistapaa. Loogisen tietomallin tasolla huomioidaan tietokannan toteutustapa. Fyysisen tietomallin tasolla tietomallissa on mukana tietokannan toteuttamiseen käytettävä tietokannanhallintajärjestelmä. Seuraavassa kukin tasoista on esitelty tarkemmin.

- *Käsitteellinen tietomalli* tai *käsitelmä* (conceptual data model, concept model) on korkean tason kuvaus sovellusalan tietovaatimuksista. *Käsitteet* (concepts) ovat mielikuvia, jotka liittyvät todellisuuden ilmiöihin, *tarkoitteisiin* (referents). Koska käsitteet ovat abstrakteja mielikuvia, tarvitsemme kommunikointiin määritelmiä ja termejä. *Määritelmät* (definitions) ovat käsitteen sanallisia kuvauksia, ja *termit* (terms) ovat käsitteiden kielellisiä nimityksiä, joiden avulla käsitteen koko sisältöön voidaan viitata. Kuvassa 7 on havainnollistettu tarkoitteen, käsitteen, määritelmän ja termin välisiä suhteita. (Suonuuti 2006) Käsitelmällinnuksen tason tarkoituksena on tehdä tiedon esitystavasta selkeä ja ymmärrettävä aihealueen asiantuntijoille. Mallin selkeys ja yksinkertaisuus ovat olennaisinta, ja tasolle ei oteta mukaan tietorakenteiden yksityiskohtia, ohjelmistojen ominaisuuksia ja laitteiston rajoituksia. Tietomallin kuvauksen täytyy koostua yksinkertaisista, yleisen tason merkinnöistä, jotka ovat käyttäjien intuitiivisesti ymmärrettävissä. Käsitelmät voidaan ajatella terminologisen työn apuvälineenä, kun käsitteelliset tietomallit taas toimivat IT-alalla olennaisena osana tietomallinnusta. Tässä dokumentissa käsitelmän ja käsitteellisen tietomallin eroavaisuuksiin ei oteta kantaa, vaan käytetään yleisesti käsitteellisestä tietomallista termiä käsitelmä.
- *Looginen tietomalli* (logical data model) tuo tietomallinnuksen lähemmäksi tietokannan toteutusta. Tässä vaiheessa otetaan huomioon se, minkä tyyppistä tietokantaa ollaan suunnittelemassa. Tietokannan tyyppi (kuten relaatiotietokanta) vaikuttaa loogisen tietomallin suunnitteluun ja koostumukseen. Tasolle ei oteta mukaan ohjelmien tai laitteiston rajoituksia eikä mitään tiettyä tietokannanhallintajärjestelmää. Tietomallin

kuvaustapa vaihtelee tietokannan tyyppin mukaan; esimerkiksi relaatiotietokantaa kuvataan kaksiulotteisina taulukoina eli relaatioina.

- *Fyysinen tietomalli* (physical data model) on hyvin kaukana aihealueen asiantuntijoista ja loppukäyttäjistä, jolloin sitä ei myöskään voida käyttää kommunikoinnin apuna. Tämän tason mallin päätarkoitus on toimia suunnitelmana ja ohjeistuksena tietokannan rakentamiseen. Malli sisältää paljon monimutkaisia yksityiskohtia tietorakenteesta, suhteista ja rajoitteista. Mallin rakentamiseen liittyy hyvin tiiviisti valittu tietokannan hallintajärjestelmä ja järjestelmän ominaisuudet sekä rajoitukset.



Kuva 7. Tarkoitteen, käsitteen, määritelmän ja termin väliset suhteet (mukaillen Suonuuti 2006)

Kuten kuvassa 6 havainnollistetaan, tietomallien tasot eivät ole irrallisia suhteessa toisiinsa, vaan yleisemmistä malleista voidaan siirtyä tarkempiin ja teknisempiin malleihin. Tietomallin sisältämä tieto pysyy, mutta sen rakenne ja yksityiskohtaisuus muuntuu käyttötarkoituksen mukaisesti eri tasojen välillä. Taulukossa 3 on esitetty karkea rajanveto, mitä kullakin tietomallin kuvaustasolla yleensä esitetään. Monet tietomallien kuvaustavat, kuten UML, eivät mene täydellisesti tämän taulukon mukaisesti. Usein jo käsitetasolle otetaan mukaan entiteettien attribuutteja ja mahdollisesti identifiointitunnuksia.

Taulukko 3. Eri kuvaustasojen erot

	Käsitemalli	Looginen tietomalli	Fyysinen tietomalli
Entiteettien nimet	x	x	x
Attribuutit	(x)	x	x
Attribuuttien tietotyypit		x	x
Attribuuttien kardinaliteetit		x	x
Tunnisteet		(x)	x
Entiteettien väliset suhteet	x	x	x
Suhteiden kardinaliteetit	(x)	x	x

### 3.3 Tietomallinnusmenetelmien vertailu

Tietojen mallinnukseen on olemassa lukemattomia eri menetelmiä. Eri menetelmistä tehtiin vertailu, jotta löydettäisiin hankkeelle sopiva tietomallinnusmenetelmä. Tietomallien vertailun tavoitteena oli selvittää yksityiskohtaisemmin kunkin tietomallinnustekniikan ominaisuuksia ja ilmaisuvoimaa, sekä mikä tärkeintä, käytettävyyttä hankkeen tietomallinnuksessa. Tiedon mallinnustavoista on käyty paljon keskustelua, joihin tässä luvussa esitetty tietomallien vertailu on antanut osaltaan tukea tietomallinnusta koskevien päätösten tekemiselle.

Kullakin vertailuun valitulla tiedon mallinnustavalla on omat hyvät ja huonot puolensa. Vertailun mallinnustapoja käytetään yleensä eri tarkoituksiin, sillä ne kuvaavat termejä, käsitteitä ja käsitteiden välisiä suhteita eri tavoin sekä työkalutuki vaihtelee. Tietomallien tekninen jatko-työdyntäminen XML-skeemojen tuottamisessa täytyy myös ottaa huomioon, koska XML-asiakirjojen rakenne kuvataan XML-skeemoilla. Vertailussa on otettu kukin näistä aihealueista huomioon.

Tässä luvussa esitellään suosituimmista tiedon mallinnustavoista E-R-kaaviot ja UML-luokkakaaviot, modernimpana ja monikäyttöisempänä mallinnustapana ontologiat sekä varsinkin asiakirjojen tietomallinnuksessa toimiva XML Schema.

Luvussa 3.3.1 esitellään E-R-malli, joka on perinteinen tietokantojen mallinnusmenetelmä. Luvussa 3.3.2 kerrotaan lyhyesti UML-luokkakaaviosta, joka on yleisesti käytetty mallinnusmenetelmä oliopohjaisten ohjelmistojen kehityksessä. Luvussa 3.3.3 esitellään ontologiat, jotka ovat moderneja ja monikäyttöisiä käsitelmalleja, joita käytetään varsinkin semanttisessa webissä. Luvussa 3.3.4 kuvataan XML Schema, jonka avulla esitetään rakenteisia dokumentteja. Näiden tiedon mallinnustapojen valinta vertailuun oli perusteltua hankkeessa tehtyjen kokeilujen ja etukäteistietämyksen valossa. Liitteessä A on kuvattu vertailun sisällön lisäksi tietomallien vertailussa käytetyt kriteerit, joiden valossa kutakin tiedon mallinnustapaa on tarkasteltu. Luvussa 3.3.5 on pohdittu vertailutaulukon avulla tehtyjä huomioita kustakin tiedon mallinnustavasta.

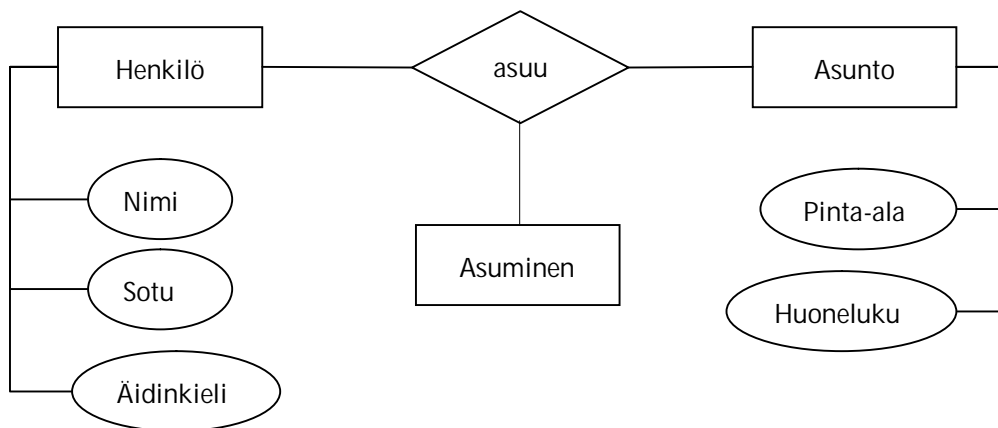
### 3.3.1 E-R-malli

---

Moderni käsiteanalyysi perustuu Peter Chenin vuonna 1976 esittelemään käsite-yhteysmallinnusmenetelmään eli E-R-malliin (Hovi ym. 2003). *E-R-malli* (Entity-Relationship model) on abstrakti käsitteellinen esitys rakenteellisesta tiedosta. E-R-malli rakentuu kolmesta osasta: entiteetti (entity), suhde (relationship) ja ominaisuus (attribute) (Sugumaran ja Storey 2006). Entiteetit voivat olla toisten entiteettien ominaisuuksia, jolloin entiteetit linkittyvät toisiinsa.

Vielä nykyäänkin E-R-malli on yksi suosituimmista ja laajalti käytetyimmistä tiedonmallinnustekniikoista. Malli kuvaa organisaation tietovaatimuksia joukkona attribuutteja sisältäviä entiteettejä, jotka ottavat osaa suhteisiin. E-R-mallinnuksessa käytetyt symbolit eivät ole täysin standardoidut, mutta eri merkintätapojen yhteydet on helppo huomata, mikäli edes joku merkintätapa on ennestään tuttu.

E-R-mallinnuksella esitetään organisaation tietyn aihealueen tietoa niin, että se ei ole yhteydessä mihinkään tietokantaohjelmistoon tai laitteistoon. Tämän takia menetelmä sopii hyvin käsitteelliseen tietomallinnukseen. Malli ei sisällä liikaa yksityiskohtia, joka on myös rajoite - esimerkiksi suhteiden rajoitteita ei ilmaista kovin selkeästi. Suhteiden esittäminen voidaan nähdä epätäydellisenä ja epätarkkana.



Kuva 8: Esimerkki E-R-mallista

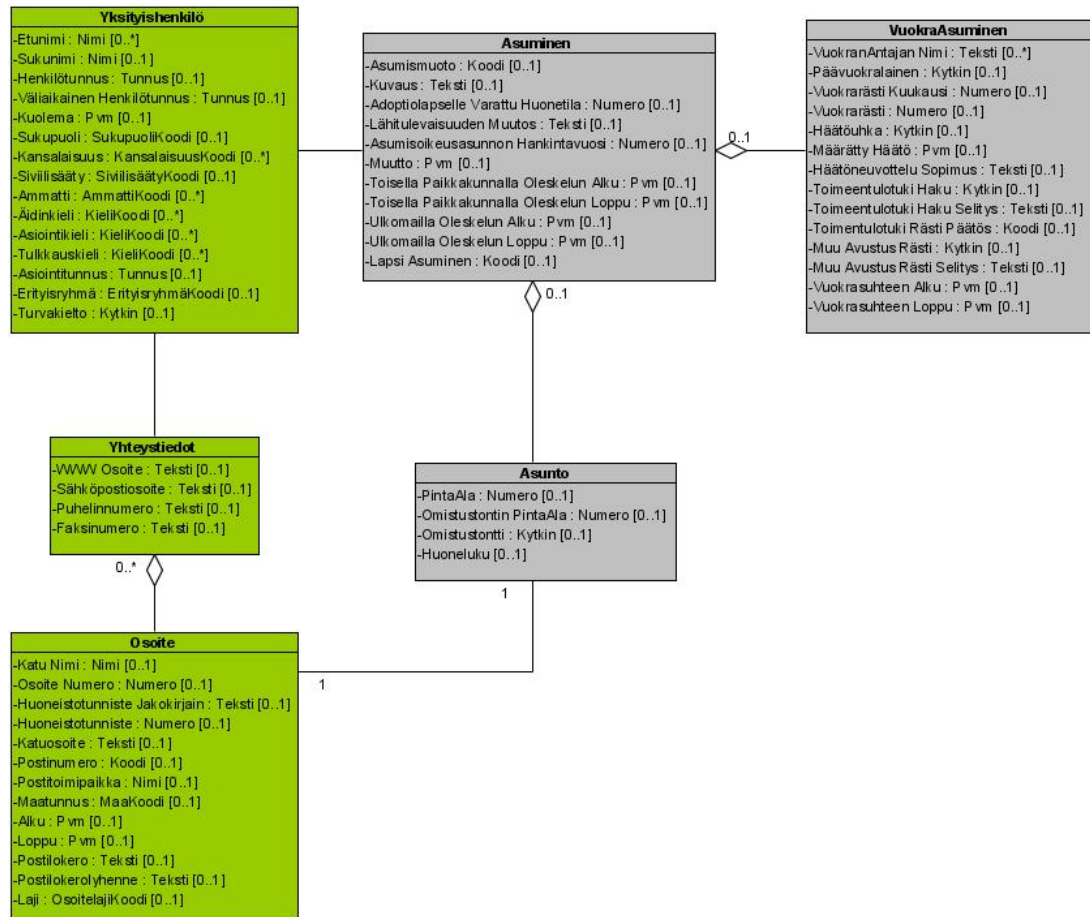
### 3.3.2 UML-luokkakaavio

---

*Unified Modeling Language* (UML) on oliopohjainen mallinnusmenetelmä, jossa sekä tieto että sen käyttäytyminen kiteytetään objekteihin. Oliopohjainen mallinnusmenetelmä oli alun perin kehitetty oliopohjaisten ohjelmointikielten koodin suunnitteluun. Siitä huolimatta tätä mallinnustapaa voidaan soveltaa käsitteelliseen tietomallinnukseen ja edelleen tietokantasuunnitteluun. UML onkin nykyään suosittu ja laajalti käytetty oliopohjainen lähestymistapa tiedon esittämiseen.

UML sisältää laajan kirjon eri kaaviotyyppejä, joista UML-luokkakaavio on yksi tärkeimmistä. Vaikka UML ei ole varsinaisesti tiedonmallinnustekniikka, niin UML:n staattinen luokkakaavio (UML Class Diagram) on toiminnaltaan minkä tahansa perinteisen käsite-yhteysmallinnusmenetelmän (kuten E-R-mallin) vastine. Sen luokat ovat käytännössä entiteettejä, ja

assosiaatiot ovat luokkien välisiä suhteita. Sen lisäksi luokkakaavioissa on erityisiä symboleita joillekin asioille, jotka esitetään perussymboleilla muissa notaatioissa, ja kaavioista puuttuu joitain E-R-malleissa käytettyjä symboleja. UML-luokkakaaviot kykenevät kuitenkin esittämään E-R-mallia laajemmin suhteiden rajoitteita. (Hay 1999)



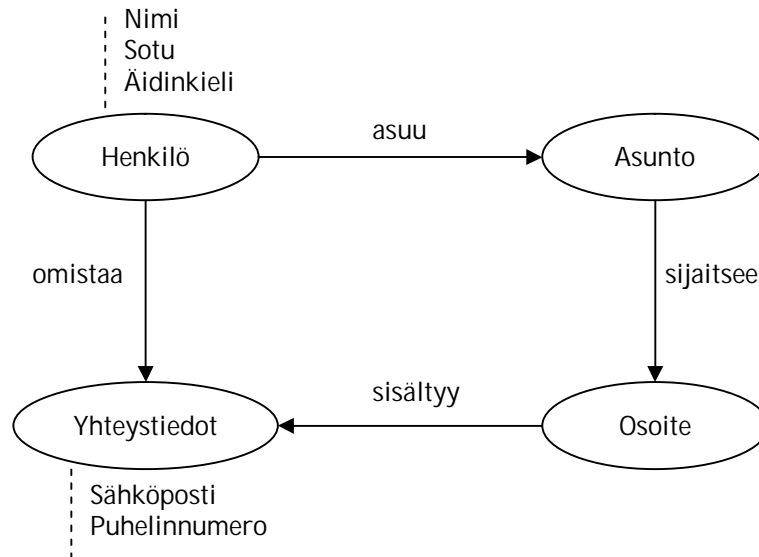
Kuva 9. Esimerkki UML-luokkakaaviosta

### 3.3.3 Ontologia

Ontologia (ontology) on käsitelmä, joka esittää joukon tiettyyn aihealueeseen liittyviä käsitteitä ja niiden välisiä suhteita. Ontologia määrittelee ryhmän rakenteita, joita käytetään esittämään reaali maailman ilmiöitä (Antoniou ja van Harmelen 2004). Tiivistettynä ontologia on selkeä ja yksiselitteinen määrittely yhteisestä käsitteistöstä (Studer ym. 1998). Kuvassa 10 on esitetty esimerkki pienestä ontologiasta.

Ontologia koostuu rajoitetusta määrästä termejä, termien määrittelyä sekä perusolettamuksista (aksioomista), jotka liittyvät termeihin. Näistä olettamuksista muodostuvat termien väliset suhteet. (Sugumaran ja Storey 2006) Termit ilmaisevat aihealueen tärkeitä käsitteitä, ja niiden väliset suhteet sisältävät usein luokkahierarkioita. Luokkahierarkioiden avulla voidaan organisoida tietämystä puumaiseen muotoon. (Antoniou ja van Harmelen 2004)

Luokkasuhteiden lisäksi ontologia voi sisältää tietoa kuten ominaisuuksia, arvorajoituksia, luokkia koskevia väitteitä ja tarkkoja määritelmiä luokkien välisistä loogisista suhteista. Hyvät ontologiat ovat todellisuuden esitystapoja, joiden tarkkuus-, yksityiskohtaisuus- ja täydellisyystasot ovat määriteltyjä. (Antoniou ja van Harmelen 2004)



Kuva 10: Esimerkki ontologiasta

### 3.3.4 XML Schema

XML on tarkoitettu ensisijaisesti tiedon merkityksen ilmaisuun, jonka ansiosta sitä käytetään usein tiedon siirtämisen muotona eri sovellusten välillä. Tämän takia XML-dokumentteja käytetään myös vaihtoehtoisena tekniikkana relaatiotietokannalle. XML-dokumentit säilövät tietoa, jonka rakenne on määritelty XML Schema -kielellä, joten näistä muodostuu eräänlainen tietomalli. (Simsion ja Witt 2005)

XML Schema määrittelee XML-muotoisten dokumenttien rakennetta (W3C 2004a, W3C 2004b, W3C 2008). Se kuvaa asiakirjoissa käytettäviä komponentteja (XML-tietotyyppejä ja -elementtejä) hierarkkisesti, sekä esittää rajoittuneesti komponenttien välisiä suhteita (lähinnä koostumussuhde ja periytyminen). Skeemoissa on myös yksinkertaisempia rakenteita kuten perustyyppit tekstejä, numeroita ja koodistoja varten.

XML skeemoilla pystytään kuvaamaan samoja objektiluokkia ominaisuuksineen kuin muillakin tietomallinnusvälineillä. XML-skeemoja ei yleensä pidetä perinteisessä mielessä tietomalleina, koska niitä yleensä käytetään pelkästään XML-dokumenttien yhteydessä rakenteen kuvaamiseen. XML Schema on kuitenkin tehokas kieli, jossa on samoja piirteitä kuin muissa tietomallien kuvaamistekniikoissa. XML Schemassa on myös ominaisuuksia, joita muissa tekniikoissa ei ole, kuten objektiluokkien rajoittaminen ja substituutioryhmät.

### 3.3.5 Tietomallien vertailu

---

Tietomallien vertailua varten laadittiin vertailukehys (kuvattu liitteessä A). Kehykseen listattiin kriteerejä, joita vertailussa haluttiin tuoda esille sosiaalihuollon asiakasasiakirjojen tietomallinnusta ajatellen. Tällaisia tietoja olivat konkreettinen tieto tietomallista, tietomallin ilmaisuvoimasta, tietomallin työkalutuksesta sekä tietomallin teknisestä jatkohyödyntämisestä.

Vertailussa oli tarkoituksena selvittää valittujen tietomallien ominaisuuksia ja soveltuvuutta sosiaalihuollon asiakastietojen tietomallinnukseen. Myös tietomallin tekninen hyödynnettävyys oli tärkeässä asemassa, sillä hankkeessa on päätetty käyttää XML-muotoisia asiakirjoja, joiden rakenne esitetään XML Schema -kaaviokielellä.

Vertailun tulokset on esitetty liitteen A taulukossa. Seuraavassa yleisiä huomioita ja pohdintaa kunkin tietomallin soveltuvuudesta sosiaalihuollon asiakasasiakirjojen tietomallinnukseen:

- *E-R-kaaviot* soveltuvat tietokantojen suunnitteluun. Malli soveltuu huonosti laajojen sanastojen tai käsittemallien esittämiseen, koska kuvaustapa vaatii paljon tilaa, jolloin yhteen kuvaukseen mahtuva ja ymmärrettävissä oleva tietomäärä on hyvin rajoitettu. Linkitettävyyden XML-skeemoihin on huono.
- *UML luokkakaavio* sopii oliopohjaisten ohjelmistojen ja tietojärjestelmien suunnitteluun. UML:llä voi kuvata myös asiakirjarakenteita ja niissä käytettäviä komponentteja (kuten CCTS). UML soveltuu auttavasti sanastojen esittämiseen. Käsittemallin esittämisessä UML luokkakaaviot ovat hyviä. Asiakirjojen ja komponenttien kuvaamisessa pitää käyttää UML:n laajennoksia, jos halutaan esittää komponentteihin liittyviä rajoitteita tai muita XML Schema -kieleen liittyviä asioita. XML-skeemojen ja UML-kaavioiden linkittäminen ei ole helppoa: siihen on olemassa joitakin työkaluja, mutta niiden toimivuus ja käytettävyys eivät ole riittävän hyviä.
- *Ontologiat* sopivat erittäin hyvin sanastojen ja käsittemallien luomiseen. Ontologioilla voidaan esittää hyvin tarkasti kaikki käsitteiden väliset suhteet, vaikka koostumussuhteen esitystapa onkin hieman monimutkainen. Malli ei sovellu tietojärjestelmien ja tietokantojen suunnittelun päätyökaluksi. Ontologioista voi periaatteessa johtaa XML-skeemoja, mutta muunnoksen säännöt eivät ole selkeitä, ja työkalutuki on huono.
- *XML Schema* sopii asiakirjarakenteiden ja niissä käytettävien komponenttien esittämiseen. Malli ei suoranaisesti sovellu tietojärjestelmien ja tietokantojen suunnitteluun, vaikka komponenttisisällöt ohjaavatkin sitä jollain tavalla. XML Schema soveltuu auttavasti sanastojen ja käsittemallien esittämiseen. Käsitemalleissa pystytään esittämään vain perussuhteita. Skeemojen graafisissa esityksissä pystytään näitä suhteita näyttämään.

Jokainen työkalu sopii siis erittäin hyvin omaan tarkoitukseensa, mutta vain tyydyttävästi tai huonosti muihin tarkoituksiin. Mallien väliset muunnokset toimivat huonosti, ja niissä häviää aina jotain tietoa. Yhdellä tietomallilla ei pystytä esittämään aivan kaikkia asioita.

## 4 Tietomallinnus Tikesos-hankkeessa

Tietomallinnuksen ohjenuorana on käytetty hyväksi kansainvälistä *Core Component Technical Specification (CCTS)* -mallia (UN/CEFACT 2009a). Tikesos-hankkeessa tietomallinnuksen yhtenä päämääränä on saada luotua asiakasasiakirjakohtaisia XML-skeemoja dokumenttirakenteiden esittämiseksi. Tästä tulee vaatimus, että valitun tiedon mallinnustavan tulisi olla mahdollisimman yhteensopiva skeemaesitystavan kanssa. Skeemat itsessään asettavat kohtuullisen vähän vaatimuksia tietomallille, sillä skeemoissa tarvitsee esittää vain seuraavanlaisia asioita:

- elementin nimi
- elementin tietotyyppi
- elementin laajennus/rajoitus
- elementin pakollisuus ja esiintymien mahdollinen määrä
- elementtien koostumussuhteet

Nämä asiat voidaan poikkeuksetta esittää millä tahansa vertailun tietomallilla, mutta tietomallien muunnettavuus käyttökelpoisiksi skeemoiksi on vaikeasti toteutettavissa. Tämän vuoksi skeemat ja muu tietomalli kannattaneen pitää edelleenkin erillään. Skeemoista kylläkin saadaan luotua muita tietomalleja, kuten UML-luokkakaavioita, joka on tietomallinnusmenetelmänä varteenotettava vaihtoehto. Jos tietomallilta kaivataan graafista ja helposti lähestyttävää esitysmuotoa, niin XML-skeemoista luodut UML-esitykset ovat hyviä ilman erillisen tietomallin luomista alusta asti. Eli XML-skeemojen ja CCTS-taulukoiden lisäksi käytetään UML-luokkakaavioita tietomallin graafisena kuvaustapana. Mallinnuskielenä on toiminut UML 2.1 Class Diagram (OMG 2009) ja mallinnustyökaluna Visual Paradigm (Visual Paradigm 2010).

UML:n symboleista on otettu käyttöön seuraavat:

- Class
- Aggregation
- Association
- Attribute (type, multiplicity)
- Package
- Cardinality

UML-kaavioita on lähdetty piirtämään CCTS-taulukoiden pohjalta. Muunnos CCTS-taulukoiden ja UML-luokkakaavioiden välillä on tehty käsin noudattamalla seuraavanlaisia muunnossääntöjä:

- CCTS ABIE<sup>1</sup> → UML Class (luokka)
- CCTS ASBIE → UML Class, johon Aggregation-yhteys
- CCTS BBIE → UML Class attribuutti

---

<sup>1</sup> Lyhyt johdanto CCTS-komponentteihin on luvussa 5.4, tarkemmat tiedot lähteissä (Huttunen 2009) ja (UN/CEFACT 2009a).

CCTS:n ja UML:n näkökulma hankkeessa on hieman eri: CCTS tähtää rakenteisten asiakirjojen komponenttipohjaiseen esittämiseen, kun UML tähtää asiakastietojen ja asiakasasiakirjojen (käsitteellisten) tietomallien graafiseen esittämiseen. Näin ollen mallien ei välttämättä tarvitse olla täsmälleen toisiaan vastaavat.

Graafista tietomallinnusta (eli UML-kaavioita) on tehty tarpeen mukaan, sillä kaikkia tietoja ei ole nähty tarpeelliseksi esittää ainakaan yhdessä yksittäisessä UML-kaaviossa tiedon suuresta määrästä johtuen. Kaavioita on käytetty silloin, kun ne selvästi ovat tarpeen monimutkaisten yhteyksien hahmottamiseen tietyssä aihepiirissä (kuten terveyteen liittyvät komponentit) sekä korkean tason tietomallin hahmottamiseen (asiakas ja asiakkaasta kerättävät tiedot aihealueittain).

Luvussa 4.1 kerrotaan hankkeen lähdemateriaaleista (sisältömäärityksistä) ja siitä, kuinka materiaalia on jalostettu eteenpäin. Luvussa 4.2 esitellään CCTS-malli, jota on käytetty tietomallinnuksen apuna. Luvussa 4.3 luodaan yhteys prosessimallinnuksen ja tietomallinnuksen kesken.

## 4.1 Word-dokumenteista Excel-taulukoihin

---

Sosiaalihuollon sisällöllisten asiakirjamääritysten lähtökohtana on, että kaikki palvelutehtäväkohtaiset asiakastietojärjestelmissä tarvittavat vähimmäistietotarpeet, asiakkaan perustietomääritykset ja palvelunantajan perustietomääritykset pyritään mahdollisimman pitkälle yhtenäistämään ja sovittamaan hankkeessa määriteltyihin asiakasasiakirjarakenteisiin. Tätä määrittelytyötä tehtiin hankkeessa vuoden 2008 aikana jo valmistuneiden sisältömääritysten osalta, muun muassa toimeentulotuen, lasten päivähoidon ja sosiaalisen luototuksen tietotarpeista. Tällöin asiakasasiakirjarakenteet kuvattiin Word-dokumenttimuodossa.

Asiakasasiakirjarakenteet laadittiin erillisinä dokumentteina kunkin palvelutehtävän kaikista suppean asiakirjatyyppin mukaisista asiakirjoista kuten toimeentulotukihakemuksesta ja lasten päivähoidon tuloksetuksesta. Määrittelytyön tässä vaiheessa tietosisältöjen yhtenäistämistä voitiin toteuttaa ja kuvata ainoastaan jo määriteltyjen asiakkaan ja palvelunantajan perustietojen osalta ja jossain määrin tiedon esitysmuotojen yhtenäistämisen osalta. Tiedon esitysmuodoissa otettiin kantaa siihen, tulisiko kukin asiakirjan yksittäinen tieto tai tietoryhmä esittää luokituksena, valmiina koodina vai vapaana tekstinä. Tässä yhteydessä määrityksiin on jäänyt myös useita vaihtoehtoja esitysmuodolle. Asiakasasiakirjarakenteiden kokoajilla ei ole ollut tuntemusta tietoarkkitehtuurin teknisistä mahdollisuuksista tai vaatimuksista eikä riittävästi teknistä asiantuntemusta tukena. Vuoden 2008 aikana hankkeessa määritellyt asiakirjallisten tietosisältöjen tekniset ratkaisut olivat myös kehitystyön alla. Näistä seikoista johtuen sisällöllisten tietotarpeiden ja teknisten tietomääritysten keskinäinen yhteensovittaminen ja vuorovaikutus olivat vielä kehittymässä.

Sisällöllisten asiakirjamääritysten osalta käännekohtana voidaan pitää siirtymistä palvelutehtäville yhteisten tietokomponenttien kokoamiseen ja kuvaamiseen, jota on selvitetty luvussa 5.1. Tietokomponenttien määrittelyn avulla pystyttiin ratkaisemaan asiakirjamäärittelytyötä vaikeuttaneita ongelmia, kun hajanaisista yksittäisistä tiedoista päästiin semanttisiin tietokokonaisuuksiin. Teknisten komponenttien esitysmuotojen tarkentuminen on myös siirtänyt

tietojen tarkentamisvaatimuksen sisällöllisiin asiakirjamäärytyksiin, jolloin vaihtoehtoisista esitysmuodoista on voitu siirtyä eksaktiin kuvaustapaan.

Vuoden 2009 aikana sisällölliset asiakirjamäärytykset päätettiin jatkossa esittää Word-dokumenttien sijasta Excel-taulukkomuodossa ja samalla esitystavaksi valittiin jo tietokomponenttien määrittelytyössä kehitetty ja osin jo vakiintunut taulukkorakenne. Tässä vaiheessa taulukon tarkoituksena oli esittää ne sisällölliset asiakirjamäärytykset, joita ei ole kuvattu asiakkaan tai palvelunantajan perustiedoissa, asiakasasiakirjojen yleisessä rakenteessa tai meta-tiedoissa.

## 4.2 CCTS tietomallinnuksessa

---

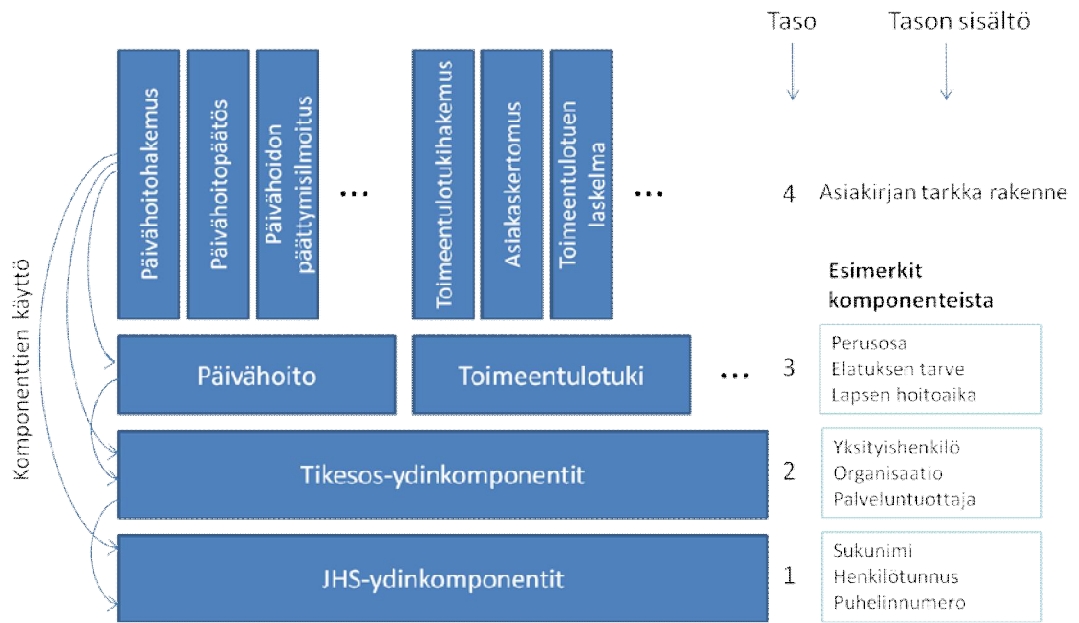
CCTS määrittelee keinot yleiskäyttöisten semanttisten rakennusyksiköiden, *ydinkomponenttien* (Core Component), tunnistamiseen, tallentamiseen ja uudelleenkäytön maksimoimiseen. Ydinkomponentit on suunniteltu esittämään yleisiä käytössä olevia liiketoimintaan liittyviä tietoja, auttamaan uusien liiketoimintaan liittyvien sanastojen ja tietomallien luomisessa sekä jo olemassa olevien sanastojen rakenteistamisessa. Ydinkomponentteja voi hyödyntää myös tiedonvaihtosanomien (kuten asiakirjojen) muodostamiseen.

Ydinkomponenteilla voidaan kuvata reaali maailman ilmiöitä kuten henkilö, organisaatio, velka, osoite ja yritys. Niistä muodostetaan *objektiluokkia*. Objektiluokille määritellään *ominaisuuksia*, jotka voivat perustua yksinkertaisiin tietotyyppeihin (*ilmentymiin*) tai muihin objektiluokkiin. Komponenttikirjasto muodostaa näin ollen myös tietomallin, koska objektiluokkiin perustuvat ominaisuudet luovat assosiaatiosuhteita eri objektiluokkien välille.

Ydinkomponentin sisältämä tieto on semanttisesti ymmärrettävissä käyttäjien kesken. Komponenttien kuvaukset eivät ole sidottuja mihinkään tiettyyn kuvauskieleen. Näin ollen ydinkomponenttien avulla voidaan välittää tietoa eri kuvauskielten, teollisuudenalojen ja aluerajojen kesken. (Huttunen ym. 2009)

CCTS-mallissa käsitteet jaetaan kahteen kategoriaan: ydinkomponentteihin ja niihin pohjautuviin toimialakohtaisiin komponentteihin. Toimialakohtaiset komponentit on sidottu johonkin kontekstiin. Sosiaalihuoltokin on toimiala, joka muodostaa samalla komponenttien ja niistä koostuvan tietomallin kontekstin. Esimerkiksi komponentti *Varallisuus* määritellään sosiaalihuollossa niin, että sen kuvausta voi käyttää toimeentulotuen määrän laskemiseen. Muilla toimialoilla komponentti *Varallisuus* voi saada muita ominaisuuksia. Esimerkiksi verotuksessa varallisuuden kuvauksen perustehtävänä on oikean veromäärän laskeminen.

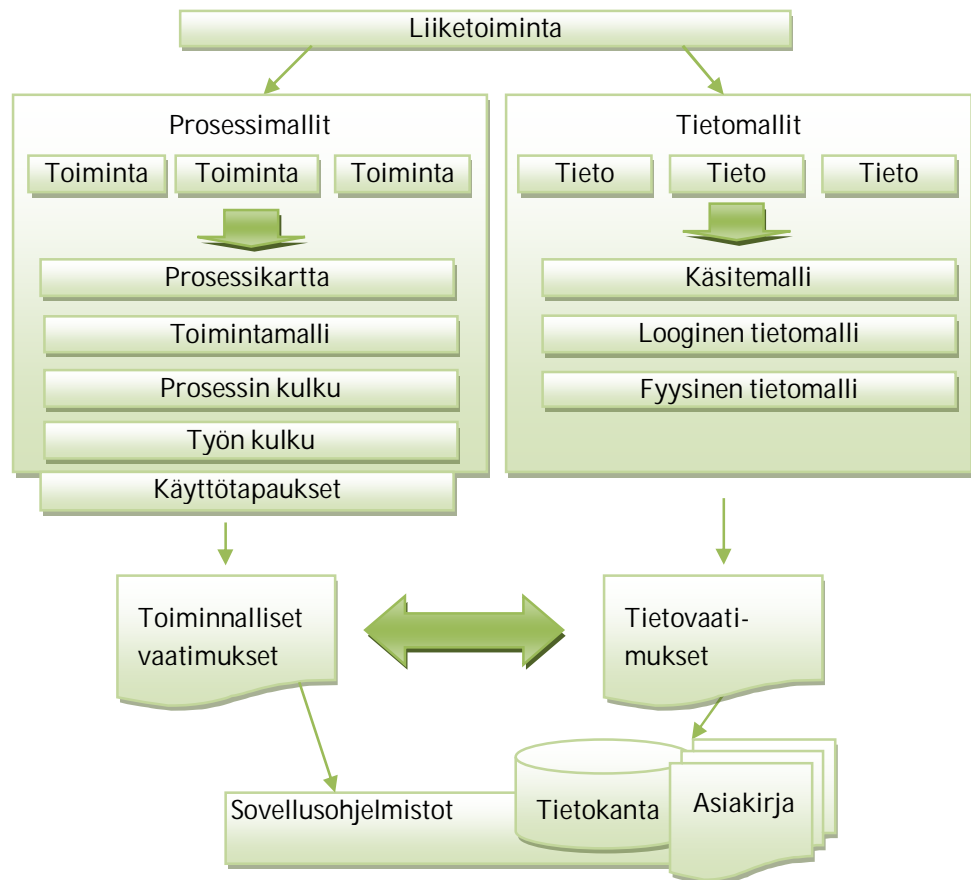
CCTS-mallia sovelletaan hankkeessa toimialakohtaisten tietokomponenttien muodostamiseen. Komponenteilla kuvataan sosiaalihuollon kontekstissa olevia reaali maailman ilmiöitä. Komponentit sijaitsevat komponenttikirjastossa kahdella tasolla: Tikesos-ydinkomponentit ja palvelutehtäväkohtaiset tietokomponentit (ks. kuvan 11 tasot 2 ja 3). Tasojen esittely löytyy lähteestä (Huttunen ym. 2009).



Kuva 11. Komponenttikirjasto (tasot 2 ja 3), sen linkitys JHS-komponentteihin (taso 1) ja komponenttikirjastoon perustuvat asiakirjarakenteet (taso 4).

### 4.3 Tietomallinnuksen ja prosessimallinnuksen yhteys

Prosessilähtöisen tai tietovirtalähtöisen sovelluskehityksen lähtökohta on usein liiketoimintaan liittyvän toiminnan kuvaamisessa (kuva 12). Yleensä liiketoimintaan kuuluva toiminta mallinnetaan prosessimalleiksi. Samalla tunnistetaan toimintaan liittyvää tietoa, joka mallinnetaan tietomalleiksi. Yhdessä nämä molemmat mallit tuovat vaatimuksia sovelluksille ja tietokannalle, sekä täydentävät samalla toisiaan. (Simsion ja Witt 2005) Prosessimallit ja tietomallit kuvaavat samaa asiaa eri näkökulmista – prosessimallin lähtökohtana on *toiminta*, kun tietomallin lähtökohtana on toiminnassa tarvittava *tieto*.



Kuva 12. Prosessimallin ja tietomallin näkökulmat

Yleensä tietojärjestelmä koostuu tietokannasta (kuvassa Tietokanta) sekä siihen tietoa tallentavista, muokkaavista ja noutavista ohjelmista (Sovellusohjelmistot). Nämä ohjelmat on suunniteltu toteuttamaan Prosessimallin (ts. Toiminnalliset vaatimukset), joka määrittelee liiketoiminnassa tarvittavat prosessit, jotka järjestelmän tulee toteuttaa. Vastaavasti Tietokannan määrittelee Tietomalli, joka kuvaa millaista tietoa kantaan tallennetaan ja miten tieto järjestetään (ts. Tietovaatimukset). (Simsion ja Witt 2005) Tietomallit jakautuvat tasoihin (luku 3.2) tarkkuuden ja käyttötarkoituksen mukaan. Samoin prosessimallinnuksessa on julkishallinnon suosituksen "Prosessien kuvaaminen" (JUHTA 2008) mukaisesti neljä tasoa: Prosessikartta, Toimintamalli, Prosessin kulku ja Työn kulku:

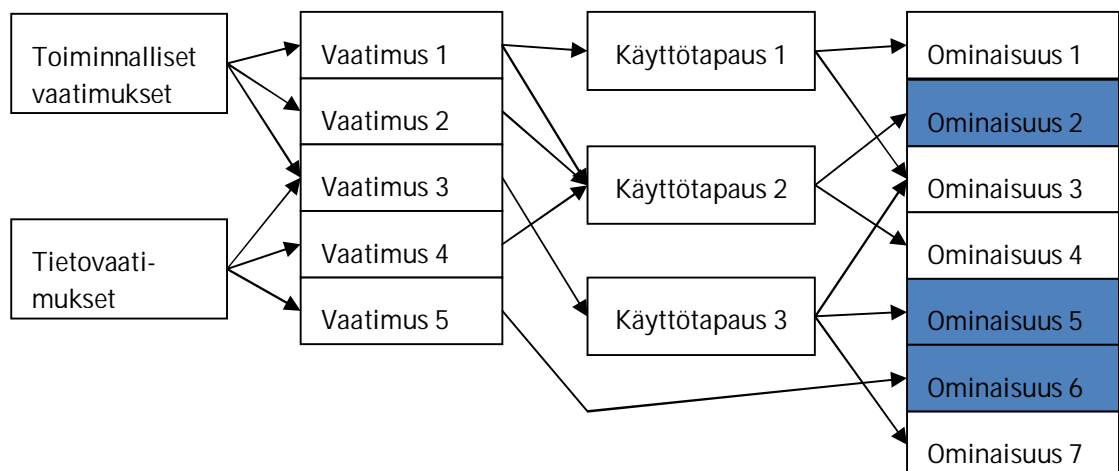
- *Prosessikartta* on prosessikuvausten ylin taso, jonka tarkoituksena on esittää yleiskuva organisaatiosta ja sen ympäristöstä. Organisaation toiminnot kuvataan siinä kokonaisuuksittain mm. ydinprosessien ja tukiprosessien avulla.
- *Toimintamallitaso* tarkoittaa prosessikartan kuvauksia jakamalla siinä esiintyvät prosessit osaprosesseiksi. Tällä tasolla osaprosessien väliset yhteydet kuvataan ja määritellään prosessien omistajat, tavoitearvot ja mittarit.

- *Prosessin kulkutason* kuvauksissa toimintamallitasolla esiintyviä osaprosesseja tarkennetaan edelleen. Prosessien toimijat, tuotokset sekä toimijoiden väliset viestit kuvataan. Myös järjestelmät kuvataan usein erillisinä toimijoina.
- *Työn kulkutasolla* kuvausta tarkennetaan edellisestä tasosta. Tällä tasolla nähdään missä muodossa tieto eri toimijoiden välillä liikkuu. Prosesseihin tulevien ja lähtevien tietojen tyypit ja tietokenttien pituudet kuvataan niin tarkalla tasolla, että kuvausten pohjalta voidaan rakentaa sähköinen palvelu.

Tarkempaa tietoa toiminnassa tarvittavasta tiedosta tarvitaan viimeistään prosessin kulkutason kuvattaessa ja erityisesti työn kulkutason kuvauksia mallinnettaessa.

Prosessikaavioista ja tietomalleista saadaan vaatimuksia järjestelmille ja järjestelmissä käytettäville tiedoille. Kun nämä vaatimukset esitetään käyttötapauksien avulla, niistä saadaan johdettua järjestelmien toimintoja ja muita ominaisuuksia ohjelmistovaatimusten muodossa. Käyttötapauksen päätarkoitus onkin usein toimia kommunikointivälineenä asiakasvaatimusten kartoittamisessa ohjelmistovaatimuksiksi. (Haikala ja Märijärvi 2006)

*Käyttötapaukset* kuvaavat järjestelmän toiminnallisuutta alkaen käyttäjien näkökulmasta. Käyttäjä käyttää järjestelmää ja aloittaa siinä tapahtumaketjuja, joiden seurauksena käyttäjä saa jonkin tehtäväkokonaisuuden suoritetuksi. Yhteen järjestelmän käyttötapaukseen liittyy yleensä yksi tai useampi käyttäjärooli. Henkilöroolien lisäksi käyttäjä (rooli) voi olla myös toinen järjestelmä. (Haikala ja Märijärvi 2006)



Kuva 13. Vaatimuksista käyttötapauksiin ja ohjelmistojen ominaisuuksiin (mukaillen Haikala ja Märijärvi 2006)

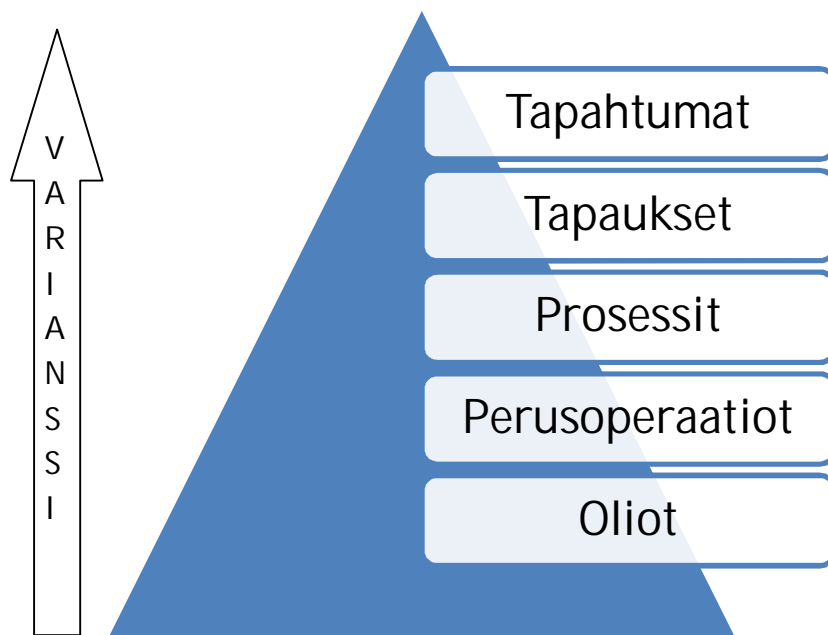
tapauksittain, jolloin ne jalostuvat sovelluksen tai siihen kuuluvan tietokannan ominaisuuksiksi. Yksi käyttötapaus voi koostua yhdestä tai useammasta toiminnallisesta vaatimuksesta tai toiminnallisesta vaatimuksesta yhdessä yhden tai useamman tietovaatimuksen kanssa.

Käyttötapausta ei voida toteuttaa pelkästä tietovaatimuksesta. Tietovaatimus (kuvassa vaatimus 5) voi kuitenkin johtaa tietokannan tai sovelluksen ominaisuudeksi ilman, että se näkyy käyttötapauksena. Samoin voi käydä toiminnallisen vaatimuksen kohdalla.

Määrittelyvaiheen lisäksi käyttötapauksia voidaan käyttää myös ohjelmistoprojektin muissa vaiheissa: käyttötapauksen kautta voidaan testata sovellusta ja tietokantaa ja sovelluksen käyttöohjeet voidaan kirjoittaa niiden avulla. (Haikala ja Märijärvi 2006)

Tietomallit ja prosessimallit toimivat itsenäisesti. Kuitenkin väliajoin tulisi tarkistaa, että tietomallia on kehitetty prosessimallia vasten siten, että tietomallissa on mukana kaikki prosessimallia tukevat tiedot, ja prosessimalli käyttää samoja käsitteitä ja termejä, jotka tietomallissa on määritelty.

Prosessimallien ja tietomallin vertaaminen voidaan suorittaa esimerkiksi CRUD-matriisin avulla, joka peilaa prosessien vaiheet tietomallin luokkiin, näyttäen mikäli prosessin vaihe luo, lukee, päivittää tai poistaa (Create, Read, Update or Delete) luokkien instansseja (Simsion ja Witt 2005). "Oleellista on pyrkiä tunnistamaan ne käsitteet [ja edelleen käsitteitä vastaavat tietomallin luokat], jotka ovat suhteellisen invariatteja – näiden varaan voidaan rakentaa nopeammin muuttuvia elementtejä" (Ritvanen 2009). Kuvassa 14 havainnollistetaan perusoperaatioiden (transformations eli luo, lue, päivitä, poista) sijoittuminen suhteessa tapahtumiin (events), tapauksiin (cases), luokan instansseihin eli olioihin (objects) ja prosesseihin (Ritvasen 2009 mukaan Kasszian 2006).



*Kuva 14. Olioihin ja perusoperaatioihin liittyvä varianssi on pienempi kuin niitä hyödyntävien prosessien, tapauksien ja tapahtumien (mukaillen Kasszian 2006).*

Perusoperaatioiden määrittelyn yhteydessä tarkastellaan olioiden ominaisuuksia (attribuutteja) ja ominaisuuksiin liittyviä rajoitteita (constraints) kuten ominaisuuksien saamia arvoja ja ominaisuuden arvon mahdollisia vaikutuksia muiden ominaisuuksien arvoihin. Lisäksi perusoperaatioiden määrittelyyn vaikuttavat olioiden väliset suhteet, erityisesti, kun halutaan säilyttää tietokannan eheys esimerkiksi poista-operaatioissa, joka usein edellyttää poistettavaan olioon tehtyjen viittausten tarkastamisen ennen poisto-operaation suorittamista.

Hankkeen määrityksissä (kuten Kärki 2007 tai Väinälä 2007) sosiaalipalveluja on kuvattu toimintälähtöisesti. Sisältömääritykset kuvaavat ensin yleisesti työn etenemisen ja sitten työssä tarvittavat minimitiedot. Työn eteneminen on sisältömäärityksissä usein kuvattu yleisellä tasolla tekstimuotoisella prosessikuvauksella ja mahdollisella korkean tason yleiskuvalla. Nämä ovat tärkeä lähtökohta prosessimallinnukselle. Vastaavasti sisältömääritysten tarvittavat minimitiedot ovat tärkeä lähtökohta tietomallintamista varten. Sisältömääritysten sisältämiä yleisen tason tekstimuotoisia prosessikuvauksia on lähdetty tarkentamaan prosessimalleiksi käyttämällä apuna mm. sisältömääritysten tietokuvauksia, sekä mahdollisuuksien mukaan valmiita palvelutehtäväkohtaisia asiakasasiakirjarakenteita.

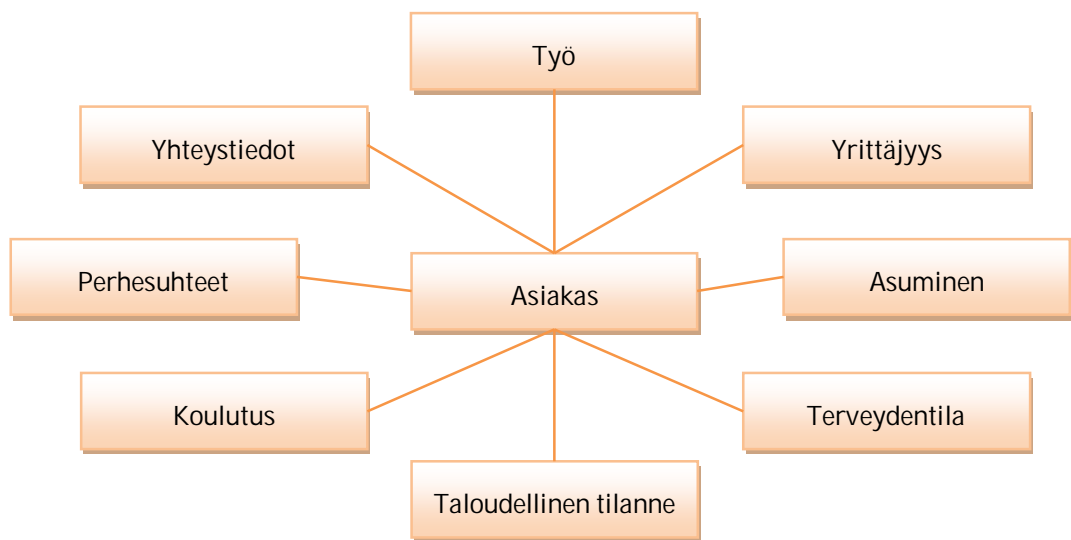
Yleisiä prosessin kulkutasoja on kuvattu palvelutehtävittäin, kuten toimeentulon turvaamiseen kuuluvan asian vireille tuleminen, ja prosessikaavioissa tietovirtoja voidaan havainnollistaa asiakirjakohtaisella tietomallilla, kuten toimeentulotukihakemus (kuvat 15 ja 16).





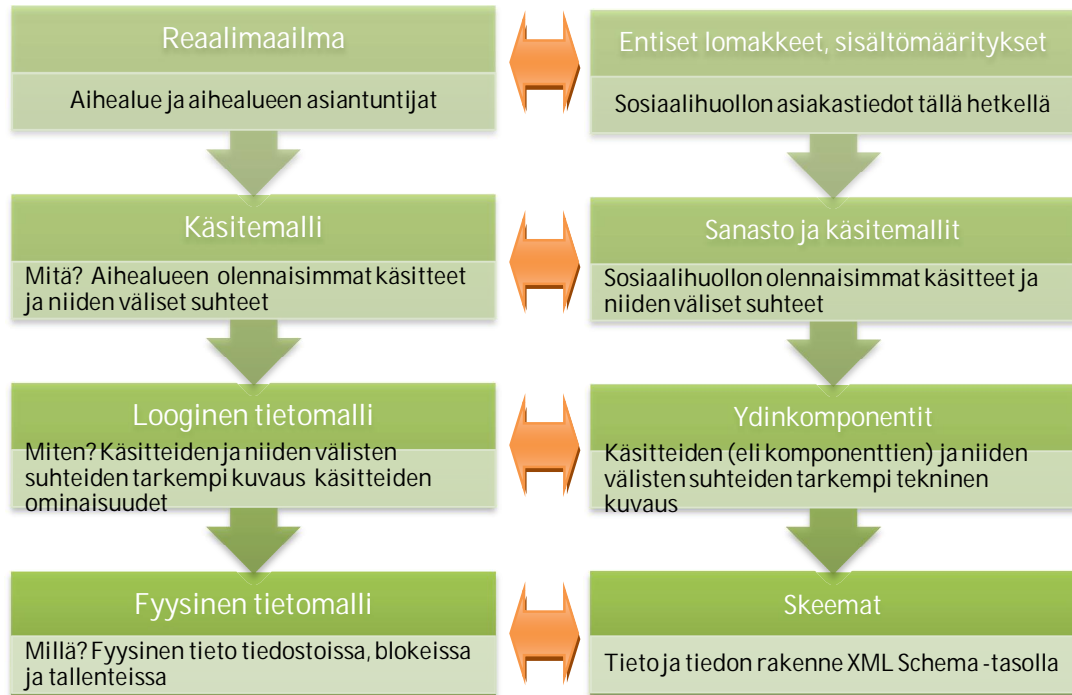
## 5 Sosiaalihuollon asiakastietojen tietomallinnus

Sosiaalihuollossa kerätään asiakastietoja, jotka liittyvät asiakkaan sen hetkiseen elämäntilanteeseen. Kerättävän tiedon määrä riippuu kyseessä olevasta palvelutehtävästä; esimerkiksi päivähoitossa ei selvitetä asiakkaiden koulutustilannetta tai asuntotilannetta, jotka muun muassa toimeentulotuessa ovat merkityksellisiä.



Kuva 17. Esimerkkejä sosiaalihuollon asiakkaasta kerättävistä tiedoista aihealueittain

Kuvassa 17 on esitetty asiakkaan elämäntilanteen osa-alueita, joista sosiaalihuollossa kerätään tietoa. Tiedon ytimessä on *Asiakas*, ja hyvin usein myös tarpeen tietää *Yhteystiedot* (miten asiakkaaseen ollaan yhteydessä). Esimerkiksi toimeentulotukea hakiessa asiakkaan *Taloudellinen tilanne* on olennaisessa osassa, jolloin asiakkaan täytyy selvittää kaikki tulot, menot, varallisuus sekä velat, joita hänellä on, ja jotka vaikuttavat mahdollisesti saatavan toimeentulotuen määrään.



Kuva 18. Tietomallinnuksen tasojen yhteys hankkeessa tehtyyn tietomallinnukseen

Kuva 18 havainnollistaa luvussa 3.2 esitellyn tietomallinnuksen tasojen yhteyden hankkeessa tehtyyn ja edelleen tekeillä olevaan tietomallinnustyöhön. Tämän hetken Reaalimaailmaa edustavat olemassa olevat lomakkeet ja tietojärjestelmät, joiden pohjalta sisältömääritykset on tehty. Käsitelmä tasoa edustaa sosiaalihuollon Sanasto asiakastietojärjestelmää varten, sekä sosiaalihuollon asiakastietojen tietomallista tuotetut Käsitelmät. Loogista tietomallia edustavat Ydinkomponentit, joissa on määritelty teknisemmällä tasolla käsitteitä ja niiden välisiä suhteita. Fyysisenä tietomallina voidaan nähdä Skeemat, koska asiakastiedot tallennetaan XML-muotoisiin dokumentteihin (ikään kuin tietokantaan), joiden rakenne on määritelty XML Schema -kielellä.

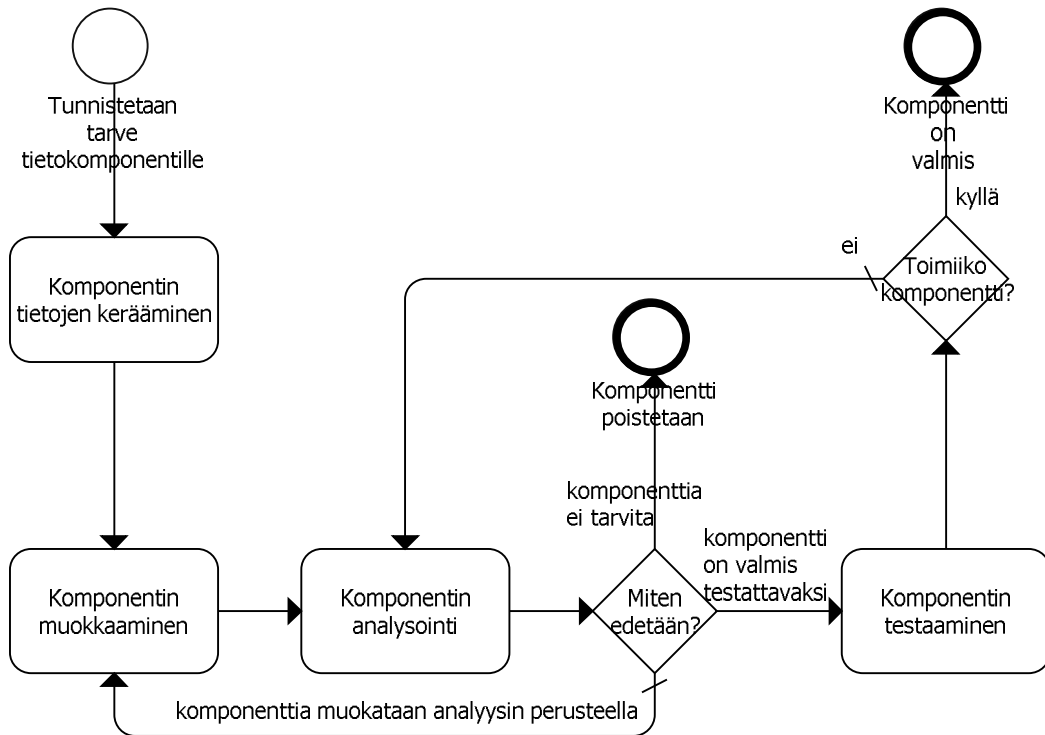
Luvussa 5.1 esitellään tietosisältöjen yhtenäistämistä ja tietokomponenttien luomista. Luvussa 5.2 esitellään tietokomponenttitaulukon rakenne ja annetaan lopuksi esimerkki tietokomponentin luomisesta. Luvussa 5.3 kuvataan CCTS-mallin soveltaminen tietokomponenttitaulukoihin.

## 5.1 Tietosisältöjen yhtenäistäminen

Tikesos-hankkeessa laadittujen sosiaalihuollon palvelutehtäväkohtaisten sisältömääritysten systemaattisen yhtenäistämisen tarve todettiin loppuvuodesta 2008. Eri palvelutehtävissä ja eri asiakastietojärjestelmissä käytettävä käsitteistö on kirjavaa. Tietojen yhtenäistämisen avulla pyritään kehittämään tietojärjestelmien yhteensopivuutta. Kun asiakasasiakirjoissa määritellyt tiedot ovat yhtenäisiä, tiedonvaihto eri asiakastietojärjestelmien ja sähköisen arkiston välillä helpottuu ja jo kerättyjä tietoja voidaan hyödyntää. Sisällön yhtenäistämistä on kuvattu tarkemmin dokumentissa *Sisällön yhtenäistäminen sosiaalihuollon teknisessä asiakirjastandardoinnissa* (Nevalainen ym. 2009).

Työskentelyn tavoitteena on sosiaalihuollon asiakastietomallin kehittäminen tietokomponenttitasolla. Tietokomponentilla (TK) tarkoitetaan asiayhteyden vuoksi yhteenkuuluvien yksittäisten tietojen joukkoa. Tietokomponentteihin on sisällytetty palvelutehtäväkohtaisten sisältö-määritysten lisäksi myös aiemmin hankkeessa määritellyt asiakkaan ja palvelunantajan perustiedot. Kirjoittamishetkellä tunnistettuja tietokomponentteja on jo yli 150 kappaletta.

Tietosisältöjen yhtenäistäminen tietokomponenteiksi voidaan nähdä toistuvana syklisenä prosessina (kuva 19). Sen vaiheita ovat tietokomponentin tarpeen tunnistaminen, tietojen kerääminen, analysointi ja muokkaaminen, testaaminen ja valmistuminen.



Kuva 19. Tietokomponentin luominen

Yhtenäistämisen prosessin käynnistyessä alettiin sisältömäärittämiä käydä systemaattisesti läpi ja etsiä semanttisesti yhteneväisiä tietoja. Tietokomponenttien työstöön osallistuu Tikesos-hankkeesta sekä teknistä että sosiaalihuollon asiantuntemusta omaavia projektitutkijoita. Tietokomponentteja työstetään Excel-taulukoiden ja GoogleDocs-taulukoiden avulla. GoogleDocs-ympäristö mahdollistaa kaikille tietokomponenttien työstöön osallistuvilla reaaliaikaisen tiedon komponenttien sisällöstä.

Tietokomponentti luodaan silloin, kun sisältömäärittämissä havaitaan semanttisesti yhteenkuuluvaa joukkoa tietoja, joka kuvaa jotakin reaali maailman ilmiötä. Kun tarve tietokomponentille on tunnistettu, siihen liittyvät tiedot kerätään taulukkoon. Teknisellä tasolla tietokomponentti muodostaa objektiluokan, jolla on ominaisuuksia (kenttiä). Tietokomponentin analyysin perustehtävänä on tunnistaa ja määritellä objektiluokan kaikki tarvittavat ominaisuudet niin, että ne muodostavat loogisen kokonaisuuden. Kenttien analysointiin kuuluu tiedon esitysmuodon valitseminen, tekstimuotoisten määritelmien luonnosteleminen sekä esimerkkien tai selitteen lisääminen.

Komponentti voidaan analysoinnin perusteella pilkkoa kahdeksi tai useammaksi uudeksi komponentiksi. Tällöin prosessi palaa alkuun. Analysoinnissa huomioidaan se, ettei kaikkien komponenttien kenttien ole välttämätöntä näkyä asiakirjoissa. Valitusta komponentista voidaan poimia vain ne kentät, joita tarvitaan kuhunkin asiakasasiakirjaan. Analysointi johtaa toisinaan koko komponentin poistamiseen. Tällöin komponentti siirretään erilliselle välilehdelle toisten poistettujen komponenttien kanssa. Tämä mahdollistaa komponenttien kenttien ja niiden sisältämän tiedon palauttamisen tarvittaessa.

Kenttien esitysmuodoissa pyritään hyödyntämään valmiita luokituksia. Sisältömäärityksissä on olemassa joitakin luokitus ehdotuksia tai viittauksia kansallisesti käytettäviin luokituksiin. Analysointivaiheessa näitä luokitus ehdotuksia tarkastellaan luokitusvaihtoehtojen määrän ja toimivuuden näkökulmasta. Aina samannimisistäkin luokituksia ei voida yhtenäistää, koska niiden sisältö poikkeaa liiaksi toisistaan. Luokitusten muodostamisessa sovelletaan Koodistopalvelun ohjeita, joista erityisesti huomioidaan tunnistettuja tärkeitä ehtoja luokituksen liittämiseksi koodistopalveluun (Hämäläinen 2008, s. 4). Luokituksista on kerrottu tarkemmin luvussa 5.2.

Komponenttia muokataan analysoinnin yhteydessä. Siihen voidaan esimerkiksi lisätä halutun kentän esitysmuodoksi valmis luokitus. Esimerkiksi *Kielitaito*-komponenttiin on otettu käyttöön Euroopan neuvoston kehittämä kielten oppimisen, opetuksen ja arvioinnin viitekehys. Tietokomponentin analysoinnissa voidaan käyttää myös projektin ulkopuolisten asiantuntijoiden apua. Asiantuntijoiden apu on välttämätöntä erityistä asiantuntemusta vaativissa tietokomponenteissa kuten *Yrittäjyys* tai *Terveystila*, joista kysyttävien asioiden merkityksiä ja tarkoituksia voi olla vaikea hahmottaa ilman aihealueen tarkempaa asiantuntemusta. Komponenttiin voidaan lisätä tai siitä voidaan poistaa kenttiä asiantuntijoiden kommenttien pohjalta. Kenttiä on mahdollista siirtää myös toiseen semanttisesti paremmin soveltuvaan komponenttiin.

Tietokomponentteja muokataan ja analysoidaan projektitutkijoiden yhteistyönä kokouksissa. Tietokomponentille nimetty vastuuhenkilö tekee kokouksia edeltävää selvitystyötä ja vastaa kokouksissa sovittujen muutosten loppuun viemisestä. Kokousten merkitys korostuu silloin kun eri komponentit ovat hyvin lähellä toisiaan (esimerkiksi *tukitoimi* ja *erityisen tuen tarve*) ja varsinkin jos toisiinsa liittyville komponenteille on nimetty eri vastuuhenkilö. Monimutkaisia ja laajoja asioita kuvaavien komponenttien kohdalla graafisten mallien piirtäminen auttaa hahmottamaan komponenttien ja niiden kenttien välisiä yhteyksiä.

Komponenttien muokkaamiseen sisältyy myös pyrkimys johdonmukaiseen esitysmuotoon sekä hyvään suomen kielen käyttöön. Muokkaamisen jälkeenkin kaikki sisältömäärityksissä määritellyt tiedot on edelleen voitava tunnistaa komponentista. Komponentin muokkaamisen jälkeen se voidaan myös palauttaa jonkin tai joidenkin kenttien osalta takaisin analysoitavaksi.

Komponentteja testataan käyttämällä niitä asiakasasiakirjoissa, jolloin saadaan tietoa komponentin toimivuudesta. Mikäli toimivuudessa on puutteita, palautetaan komponentti uudelleen analysoitavaksi. Samalla saattaa ilmetä myös tarve uusien tietokomponenttien määrittelyyn. Esimerkiksi päivähoidon asiakasasiakirjoja muodostettaessa huomattiin tarve uudelle tietokomponentille. Tietokomponentti *toistumistiheys* määriteltiin, koska tarvittiin komponenttia, jonka avulla pystytään kuvaamaan tapahtuman toistumista tietyssä ajanjaksossa. Uudella komponentilla voidaan kuvata muun muassa sitä kuinka monta kertaa kuukaudessa

lapsi tarvitsee hoitoa iltaisin. Tietokomponentti *lapsen hoitoaika* käyttää komponenttia toistumistiheys.

Tietokomponentti-työryhmä vie komponentteja säännöllisesti projektipäälliköiden arvioitavaksi. Tietokomponenttitaulukko on julkaistu keskeneräisenä myös Tikesos-hankkeen sivustolla. Tämän avulla pyritään saamaan asiantuntijoilta kommentteja ja ehdotuksia tietokomponenttien sisällöstä mahdollisimman varhaisessa vaiheessa. Projektipäälliköiden hyväksymät tietokomponentit lähetetään sisällön asiantuntijoille ja järjestelmätoimittajille kommentoitaviksi. Tietokomponentti on valmis, kun kaikki sisältömääritykset on huomioitu, asiakasasiakirjat ovat valmiit ja komponentti on virallisesti hyväksytty hankkeen johtoryhmässä.

## 5.2 Luokitukset

---

Tikesos-hankkeessa tuotetuissa palvelutehtäväkohtaisissa sisältömäärityksissä on rakenteistettu kerättäviä tietoja muun muassa käyttämällä hyödyksi luokituksia. Luokitus on tietyn aihealueen ryhmittely luokkiin, joihin toisistaan eroavat asiat sijoitetaan. Luokituksilla voidaan rajata asiakirjan kenttien arvoja. Sisältömäärityksissä on käytetty joko valmiita luokituksia, kuten maa ja kieliluokitus, tai kehitetty uusia luokituksia.

Luokitusten tarkoituksena on muun muassa parantaa sekä asiointipalvelun että asiakastietojärjestelmän käytettävyyttä ja helpottaa tilastointia ja tiedonhakuja. Luokitukset viedään THL:n ylläpitämälle kansalliselle koodistopalvelimelle. Tällöin luokitukset ovat yhdenmukaisia ja niitä voidaan käyttää kansallisesti. "Kansallinen koodistopalvelu on menetelmä varmistaa laajasti käytettävien rakenteiden, luokitusten ja sanastojen laatu ja huolehtia niiden ylläpidosta." Koodistopalvelimelle tuotettavien luokituksille on asetettu kuusi kriteeriä (Hämäläinen ym. 2008):

1. Käyttö on riittävän laajaa.
2. Luokitus on sisällöltään ja rakenteeltaan laadukas.
3. Luokituksen käyttöönotosta on odotettavissa merkittävää lisäarvoa.
4. Käyttöönotto ei aiheuta haittaa terveydenhuollon ammattilaille tai potilaille, tai mahdollinen haitta arvioidaan merkittävästi hyötyjä pienemmäksi.
5. Luokitusta on mahdollista hyödyntää potilastietojärjestelmissä ja KANTA-arkistossa.
6. Luokituksen ylläpidosta on uskottavasti sovittu ja se on mahdollista kustantaa joko THL:n koodistopalvelun tai sovittun ylläpitäjätahon budjetista.

Sisältömäärityksiä tehtäessä luokkien teossa ei ole välttämättä huomioitu koodistopalvelun asettamia kriteereitä, joten luokituksia ei voida hyväksyä käytettäväksi ennen kuin niitä on peilattu kriteereihin. Koodistopalvelimen kriteereistä varsinkin kolme ensimmäistä pitää ottaa huomioon jo heti luokitustyön alussa. Kun tarkastellaan luokituksen käytön laajuutta, pitää ottaa muut (kuten Kelan ja VRK:n) järjestelmät huomioon; tällöin luokituksella on lisäarvoa tulevaisuudessa, kun eri järjestelmissä on käytössä samoja luokituksia. Luokituksen sisällön eli yksittäisten luokkien laadun ja loogisuuden varmistaminen on ensiarvoisen tärkeää jo työnlaadunkin varmistamiseksi. Laadukkaita luokituksia voidaan hyödyntää myös muissa palvelutehtävissä.

Luokitusten tarkastelu konkretisoituu tietokomponentteja luotaessa. Jos sisältömäärityksissä esitysmuoto on luokka, tarkastellaan luokituksen käyttötarkoitusta, loogisuutta ja laajuutta. Esimerkiksi toimeentulotuessa velkajärjestely on esitetty kaksiportaisena luokituksena:

- 01 Päätös velkajärjestelystä
- 02 Vireillä oleva hakemus

Kun luokitusta tarkastellaan koodistopalvelun kriteereitä ajatellen, luokituksen käyttö ei ole riittävän laajaa, eikä luokitus ole sisällöltään ja rakenteeltaan tarpeeksi laadukas ehdotettavaksi koodistopalvelimelle. Tämän johtopäätöksen myötä esitysmuodoksi tulee pelkkä teksti. Jos kuitenkin on olemassa valmis luokitus, jota voidaan hyödyntää ja se täyttää koodistopalvelun kriteerit, voidaan luokitus ottaa käyttöön. Esimerkiksi kielitaidon arviointiin otettiin käyttöön eurooppalaisen viitekehyksen luokitus.

Useammassa palvelutehtävässä voi olla luokiteltu samaa asiaa, kuten tulolajeja, menolajeja ja varallisuuslajeja. Tämä helpottaa luokitusten yhtenäistämistä. Joissakin tapauksissa sisältömäärityksen luokitus ei ole tarpeeksi kattava, mutta laajentamalla siitä voidaan saada laadukas luokitus. Esimerkiksi oleskeluoikeusluokitus sisältömäärityksissä esitetään kolmiportaisena

- A jatkuva
- B tilapäinen
- P pysyvä

Ulkomaalaislakia (30.4.2004/301) lähteenä käyttäen luokitus laajeni kuusiportaiseksi. Samalla luokituksesta tuli myös kansallisesta näkökulmasta katsottuna yhtenäisempi ja paremmin hyödynnettävä. Lopullinen luokitus on:

- 1 = jatkuva (A)
- 2 = tilapäinen (B)
- 3 = pysyvä (P)
- 4 = EY oleskelulupa (P-EY)
- 5 = määräaikainen oleskelukortti
- 6 = pysyvä oleskelukortti

Seuraava esimerkki on velka-luokituksen hahmottamisesta ja yhtenäistämisestä. Sisältömäärityksissä tieto asiakkaan veloista on kuvattu sosiaalisessa luototuksessa (kuva 20) ja toimeentulotuessa (kuva 21). Sosiaalisessa luototuksessa on velan lajeista luokitus (velkatyyppi) ja toimeentulotuessa esitysmuotona on vapaa teksti (velkalaji), jolla voi kuvata joko velan nimeä tai tyyppiä. Tässä vaiheessa päätetään, luodaanko luokitus vai ei. Velkalaji on luokituksena hyvin yleinen. Sosiaalisen luototuksen sisältömäärityksissä oleva luokitus ei kuitenkaan ole riittävän kattava, eli sisällöltään ja rakenteeltaan se ei ole laadukas. Luokituksen laatua voi kuitenkin parantaa lisäämällä siihen puuttuvia velkalajeja. Vaikka velkalajiluokituksen käyttötarkoitusta ei sisältömäärityksissä ole tuotu selkeästi esille, voi olettaa, että tulevaisuudessa luokitus helpottaisi velkatietojen siirtymistä verottajalta sosiaalihuollon tietojärjestelmiin. Näiden perusteiden jälkeen voidaan todeta, että kannattaa luoda velka-luokitus.

Tieto	Tieto (taso 2)	Tiedon esitysmuoto	Muuta
Todellinen velkamäärä		€	Tieto voidaan merkitä myöhemmin.
Velkatyyppi		Valikkoalaryhmittelyn otsikko	
	Pankkilainaa	Kyllä/ei	
	Kulutusluottoja	Kyllä/ei	
	Luottokortteja	Kyllä/ei	
	Takausvelkaa	Kyllä/ei	
	Asuntolainaa	Kyllä/ei	
	Laskuja	Kyllä/ei	
	Yksityisvelkaa	Kyllä/ei	
	Muuta, mitä	Kyllä/ei	
Velkaantumisen		vuosiluku	

Kuva 20. Velkalajit sosiaalisessa luototuksessa

**5.3.9 VELAT**

Tieto hakijan velasta ja velkajärjestelystä. Hakijan lisäksi tiedot tarvitaan myös puolisoista/kumppanista ja kaikista alle 18-vuotiaista lapsista.

**5.3.9.1 Velkalaji**

Tieto velan nimestä tai tyypistä, esimerkiksi opintolaina, asuntolaina, ulosottovelka.  
Esitysmuoto: vapaa teksti

**5.3.9.2 Velan määrä**

Euromääräinen tieto velan kokonaismäärästä.  
Esitysmuoto: €

Kuva 21: Velkalajit toimeentulotuessa

Luokituksen luomisen ensimmäinen vaihe on ottaa sisältömäärityksissä annetut luokan velkatyyppi ja velkalaji lajit luokitukseen. Sen jälkeen tarkastellaan, mitä velkalajeja yleisesti on olemassa ja lisätään ne tarpeen mukaan luokitukseen. Kun luokitus on tarpeeksi kattava, annetaan luokitus kommentoitavaksi sisällön asiantuntijoille ja sen jälkeen hyväksytään luokitus. Alla lopullinen velka-luokitus velkalajeineen, jota voi ehdottaa vietäväksi koodistopalvelimelle:

- 1 Asuntolaina
- 2 Kulutusluotto
- 3 Lasku
- 4 Luottokorttivelka
- 5 Opintolaina
- 6 Pankkilaina
- 7 Takausvelka
- 8 Ulosottovelka
- 9 Vuokrat
- 10 Yksityisvelka

11 Yrittösvelka

12 Muu velka

Luokka	Velka		
kenttä	Laji	<b>Luokitus: Velka</b>	SLAT/henkilön tilanteen alkukartoitus, TAT / Hakemus
kenttä	Kuvaus	teksti	SLAT/henkilön tilanteen alkukartoitus, TAT/Hakemus
kenttä	Määrä	€	SLAT/henkilön tilanteen alkukartoitus, VelkaluetteloTAT /Hakemus, ADOPA/ hakijoiden perustiedot ja Perheen sisäinen adoptio
kenttä	Lyhennyserä	€	TAT / Hakemus,SLAT / Velkaluettelo
kenttä	Korko	€	TAT/hakemus, ADOPA/ hakijoiden perustiedo
kenttä	Lyhennyksen aikayksikkö	TK Aikayksikkö	TAT/Hakemus,SLAT/Velkaluettelo
kenttä	Koron aikayksikkö	TK Aikayksikkö	lisätty 21.4.2009, koska koron käsittelyajanjakso voi olla eri kuin lyhennyksen ajanjakso.
kenttä	Asiakkaan suhde	teksti	SLAT / Velkaluettelo
kenttä	Kanssavelallinen	TK Yksityishenkilö	SLAT / Velkaluettelo
kenttä	Takaaja	TK Yksityishenkilö	SLAT / Velkaluettelo
kenttä	Velkojan nimi	teksti	SLAT / Velkaluettelo
kenttä	Käyttötarkoitus	teksti	SLAT / Velkaluettelo
kenttä	Hoitamatta alkaen	päivämäärä	SLAT / Velkaluettelo
kenttä	Irtisanomispäivä	päivämäärä	SLAT / Velkaluettelo
kenttä	Muu vakuus	teksti	SLAT / Velkaluettelo
kenttä	Syntymisajankohdan päivä	pp	SLAT/henkilön tilanteen alkukartoitus, Velkaluettelo
kenttä	Syntymisajankohdan kuukausi	kk	SLAT/henkilön tilanteen alkukartoitus(vuosi), Velkaluettelo
kenttä	Syntymisajankohdan vuosi	vvv	SLAT/henkilön tilanteen alkukartoitus(vuosi), Velkaluettelo
kenttä	Ulosotossa perittävänä olevat velat	€	SLAT/Velkaluettelo
kenttä	Alkuperäinen määrä	€	SLAT/Velkaluettelo
kenttä	Erääntyneen saatavan määrä	€	SLAT/Velkaluettelo
kenttä	Vuokrarästi	Vuokrarästi TK	
kenttä	Vähennyskelppoisuus	€	LASA/Lapsen elatusavun suuruuden arvioiminen

1	Asuntolaina
2	Kulutusluotto
3	Lasku
4	Luottokorttivelka
5	Opintolaina
6	Pankkilaina
7	Takausvelka
8	Ulosottovelka
9	Vuokrarästi
10	Yksityisvelka
11	Yrittösvelka
12	Muu velka

Kuva 22. Velkaluokituksen käyttö tietokomponentissa

### 5.3 Tietokomponenttitaulukot

Tietokomponenttien kokoamisessa ja muokkaamisessa käytetään hyväksi Excel-taulukoita. Tietokomponenttitaulukko muodostuu seuraavista sarakkeista:

*Komponentin tyyppi* on joko luokka tai kenttä. Luokka muodostuu yhdestä tai useammasta kentästä.

*Komponentin nimi* on mahdollisimman lyhyt ja kuvaava. Luokan nimeä ei tarvitse toistaa kenttien nimissä. Esimerkiksi tietokomponentin `Asunto` kenttää `pinta-ala` ei tarvitse enää nimetä asunnon `pinta-ala`ksi vaan pelkkä `pinta-ala` riittää. Viittaus luokan nimeen ei tuo uutta informaatiota ja pidentää kentän nimeä. Nimissä käytetään mieluummin yksikkö- kuin monikkomuotoa. Jos kentän esitysmuotona on teksti ja kentällä kuvataan useita eri samantyyppisiä ilmiöitä, niin nimessä voi käyttää monikkoa. Esimerkiksi tietokomponentissa `Asunto` kentän `Muutostyöt` nimi on monikkomuodossa, koska se on tekstimuotoinen ja kentällä kuvataan kaikkia asunnossa tarvittavia muutostöitä. Toistuvien kenttien nimet ovat yksikkömuodossa, koska jokaisella kentän ilmentymällä kuvataan vain yhtä asiaa.

*Tiedon esitysmuoto* voi viitata toiseen tietokomponenttiin, jolloin esitysmuotona on esimerkiksi TK Ajanjakso. Esitysmuotona voi olla myös teksti, jolloin kenttään voidaan kirjoittaa vapaata tekstiä. Luokitusta käytettäessä esitysmuotona voi olla luokitus kokonaisuudessaan, esimerkiksi komponentissa Tu10 kentän verotettavuus esitysmuoto on luokitus: 1 = brutto, 2 = netto. Laajempaan luokitukseen on vain viitattu tiedon esitysmuodon kohdalla, esimerkiksi komponentin Tu10 kentässä Laji viitataan tuloluokitukseen. Laajemmat luokitukset, jotka eivät ole koodistopalvelimella, on koottu erilliseen taulukkoon. Koodistopalvelimen luokitusta käytettäessä tiedon esitysmuotona annetaan luokituksen nimi ja lähteenä mainitaan koodistopalvelin. Näitä luokituksia ei ole poimittu erilliseen taulukkoon. Teknisessä työssä tietokomponenttitaulukossa annetut esitysmuodot muunnetaan sopiviksi teknisiksi ilmentymätermeiksi. Ilmentymätermien luettelo löytyy julkishallinnon suosituksesta 170 (JUHTA 2009).

*Asiakirjalla* viitataan luokan tasolla niihin asiakasasiakirjoihin, joissa luokkaa on käytetty. Kentän tasolla viitataan sisältömäärityksissä oleviin asiakirjoihin. Kirjaamisen helpottamiseksi palvelutehtäville on annettu omat lyhenteet kuten päivähoitolle PHAT ja toimeentulotuolle TAT.

*Lähde* viittaa sisältömäärityksiin tai asiakkaan tai palvelunantajan perustietoihin. Lähdemerkintä voi olla esimerkiksi TAT 5.3.10.2, joka tarkoittaa toimeentulotuen sähköisesti julkaistun sisältömäärityksen numeroitua osaa. Kirjoina julkaistujen sisältömääritysten numerointi voi näistä poiketa. Lähdemerkinnän tarkoitus on helpottaa sisältömääritysten oikeaan kohtaan palaamista komponentteja analysoitaessa.

*Määritelmä* on käsitteen kuvaus, jonka tulee erottaa käsite sen lähikäsitteistä (Terminologian sanasto, TSK 36.) Sanastokeskus osallistuu määritelmien laatimiseen.

*Selitys/esimerkki* sisältää luokan tai kentän käytön ymmärtämistä helpottavia tai määritelmää täydentäviä lisätietoja ja esimerkkejä.

*Kentän toistuvuus* viittaa siihen voiko kenttä esiintyä useammin kuin kerran. Tietomallin joustavuuden vuoksi kentät on hyvä määritellä toistuviksi. Tekstimuotoiset kentät eivät ole toistuvia.

*Kentän pakollisuus* viittaa siihen tuleeko kentän esiintyä ainakin kerran. Tietomallin joustavuuden vuoksi kentät on hyvä määritellä valinnaisiksi.

*Päivämäärä* kertoo ajankohdan, jolloin komponenttia on viimeksi muutettu.

*Muutokset*-kohtaan kirjataan tehdyt muutokset ja niiden perustelut.

*Versio* kertoo komponentin tilan:

Kesken	Komponentin sisältö on vielä muokkauksessa.
Käytössä	Komponentti on kesken, mutta sitä on alustavasti käytetty asiakasasiakirjoja mallinnettaessa
Versio 0.001...	Komponentin runko on valmis. Versionumero kasvaa isompien päivitysten myötä.

Versio 0.01	Komponentti on varustettu määritelmillä
Versio 0.02	Sisällön asiantuntijoille kommenttikierrokselle valmiit komponentit.
Versio 0.6	Kommenttikierroksella käyneet ja johtoryhmässä hyväksytyt.
Versio 1.0	Kaikki sisältömääritykset on huomioitu, asiakasasiakirjat ovat valmiit ja komponentti on virallisesti hyväksytty.

*Vastuuhenkilö* vastaa komponentin muokkaamisesta.

*Kommentti*-kohtaan vastuuhenkilö voi laittaa muistiinpanoja selvitettävistä asioista. Tähän kenttään myös muut kuin vastuuhenkilö voivat lisätä omia huomioitaan tai kysymyksiään komponentista.

Seuraavassa esimerkissä kuvataan yhtä asiakirjamallinnuksen yhteydessä luotua tietokomponenttia. Päivähoidon sisältömäärityksissä todetaan erityistä hoitoa ja kasvatusta tarvitsevan lapsen oppimis- ja kuntoutussuunnitelmaan liittyvistä tavoitteista seuraavasti:

7.4.6.5 Lyhyen aikavälin (~3-6kk) tavoitteet tärkeysjärjestyksessä Tavoitteiden tulisi olla konkreettisia asioita, jotka kuvataan mahdollisimman arkikielisesti ja yksityiskohtaisesti.

Esitysmuoto: vapaa teksti

7.4.6.6 Tavoitteiden seuranta-/arviointipäivä

Esitysmuoto: pppkkvvvv

7.4.6.7 Menetelmät, välineet

Esitysmuoto: vapaa teksti

7.4.6.8 Vastuuhenkilö

Esitysmuoto: vapaa teksti

tai PA Sosiaalihuollon ammatillisen henkilön nimi

(Väinälä 2007, 39)

Koska myös muissa sosiaalihuollon palvelutehtävissä kuin päivähoitossa asetetaan tavoitteita, päätettiin luoda *Tavoite*-tietokomponentti, jota voidaan käyttää asiakasasiakirjoissa palvelutehtävästä riippumatta. Ensimmäiseksi komponentti sai kentän kuvaus, jonka esitysmuoto on teksti, ja joka mahdollistaa tavoitteen kuvaamisen sisältömäärityksen mukaisesti arkikielisesti ja yksityiskohtaisesti. Seuraavaksi komponenttiin lisättiin kenttä aikataulu, jonka esitysmuoto on teksti ja jossa voidaan kuvata tavoitteen suunniteltua toteutumisen aikataulua. Esimerkiksi jos kyseessä on lyhyen aikavälin tavoite, tähän kohtaan voidaan kirjata ~3–6 kk. Kenttien aset-

taminen ja seuranta esitysmuoto on päivämäärä. Näihin kenttiin voidaan kirjata sekä tavoitteen asettamis- että arviointipäivä. Menetelmä-kentän esitysmuoto on teksti ja siihen voidaan kirjata menetelmät ja välineet, joiden avulla tavoite pyritään saavuttamaan. Koska tavoitteet sisältömäärityksen mukaan halutaan esittää tärkeysjärjestyksessä, komponenttiin liitettiin kenttä prioriteetti. Prioriteetti ilmaistaan numerona siten, että tärkein tavoite saa numeron 1. Tavoitteille on sisältömäärityksissä haluttu nimetä myös vastuuhenkilö, joten lopuksi tämä kenttä liitettiin osaksi komponenttia. Tavoite-komponentti käyttää jo aiemmin määriteltyä komponenttia *Vastuuhenkilö*. Sisältömäärityksistä taulukkomuotoon muutetusta komponentista osa on nähtävissä taulukossa 4.

*Taulukko 4. Tietokomponentti Tavoite*

	Komponentin nimi	esitysmuoto	Määritelmä
<b>Luokka</b>	<b>Tavoite</b>		
kenttä	Kuvaus	Teksti	Konkreettinen asia, joka kuvataan mahdollisimman arkikielisesti ja yksityiskohtaisesti
kenttä	Asettaminen	Pvm	Päivä, jolloin tavoite on asetettu
kenttä	Aikataulu	Teksti	Tavoitteen suunniteltu toteutumisen aikaväli
kenttä	Seuranta	Pvm	Päivä, jolloin tavoitteen toteutumista arvioidaan
kenttä	Menetelmä	Teksti	Menetelmät/välineet, joilla pyritään saavuttamaan tavoitetta
kenttä	Prioriteetti	Número	Tavoitteen prioriteetti, 1=korkein prioriteetti (tärkein tavoite). Jos asiakirjassa tai komponentissa käytetään useita Tavoite-komponentteja, prioriteeteilla voi ilmaista tärkeysjärjestyksen
kenttä	Vastuuhenkilö	TK Vastuuhenkilö	Henkilö, joka on vastuussa työskentelystä tavoitteen saavuttamiseksi

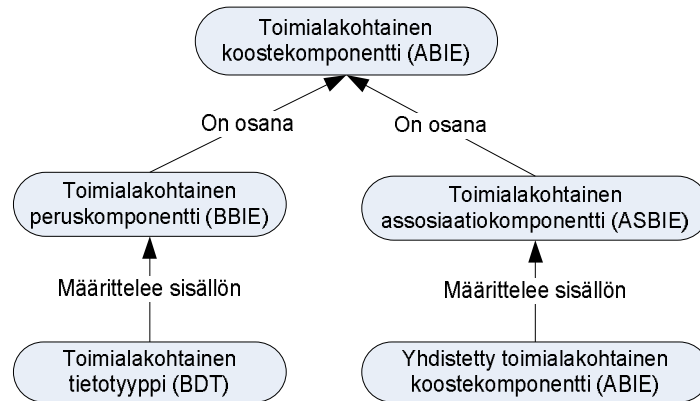
## 5.4 CCTS ja tietokomponenttitaulukot

Tekniseen tietomallinnustyöhön sovellettu CCTS-menetelmä (ks luku 4.2) määrittelee joukon metatietoja, joilla tietokomponentit ja niiden kentät tulisi varustaa. Metatietovaatimukset riippuvat komponentin tyypistä. Hankkeessa määritellään vain toimialakohtaisia komponentteja (Business Information Entity), jotka jakaantuvat kolmeen kategoriaan (ks. kuva 23):

*Toimialakohtaiset koostekomponentit* (Aggregate Business Information Entity, ABIE) vastaavat tietokomponenttitaulukoissa määriteltyjä luokkia.

*Toimialakohtaiset peruskomponentit* (Basic Business Information Entity, BBIE) vastaavat tietokomponenttitaulukoissa määriteltyjä yksinkertaisiin tietotyyppeihin perustuvia kenttiä.

*Toimialakohtaiset assosiaatiokomponentit* (Association Business Information Entity, ASBIE) vastaavat tietokomponenttitaulukoissa määriteltyjä muihin objektiluokkiin perustuvia kenttiä.



Kuva 23. Toimialakohtaisten komponenttien suhteet CCTS-mallissa.

Kuvassa 23 esiintyvä toimialakohtainen tietotyyppi (Business Data Type, BDT) määrittelee peruskomponenttien tekniset esitysmuodot, jotka eivät perustu muihin koostekomponentteihin, vaan suoraan yksinkertaisiin tietotyypeihin kuten Teksti tai Koodi. Tietotyypit vastaavat julkishallinnon suosituksessa 170 (JUHTA 2009) määriteltyjä ilmentymätermejä.

Taulukossa 4 esitetystä esimerkistä luokka `Tavoite` on ABIE-komponentti. Kaikki kentät paitsi `Vastuuhenkilö` ovat BBIE-komponentteja. Kenttä `Vastuuhenkilö` on ASBIE-komponentti, koska se perustuu muualla määriteltyyn luokkaan `Vastuuhenkilö`.

Tietokomponenttitaulukoissa olevat sarakkeet vastaavat osittain CCTS-mallin metatietoja. Tarkemmin vastaavuus määritellään seuraavasti (vrt. luku 5.3).

*Komponentin tyyppi ja tiedon esitysmuoto* kertovat yhdessä CCTS-komponentin tyypistä. Jos komponentin tyyppi on `luokka`, rivistä muodostetaan ABIE-komponentti. Jos komponentin tyyppi on `kenttä` ja tiedon esitysmuoto -sarakkeessa on viittaus toiseen tietokomponenttiin, niin rivistä muodostetaan ASBIE-komponentti. Muista kentistä muodostetaan BBIE-komponentteja.

*Komponentin nimestä* muodostetaan CCTS-komponentin nimen perusosa korvaamalla skandinaaviset merkit kirjaimilla `a` ja `o` ja muuttamalla jokaisen sanan alkukirjain suuraakkoseksi. Luokkien kohdalla nimen perusosa toimii objektiluokkaterminä (Object Class Term) ja kenttien kohdalla ominaisuusterminä (Property Term). Lisäksi alkuperäinen nimi muodostaa CCTS-metatiedon *ammattisana* (Business Information Business Term).

*Tiedon esitysmuotoa* käytetään vain jos kyseessä on kenttä. Jos kentästä muodostuu BBIE-komponentti, esitysmuoto määrittelee komponentin ilmentymätermin (Representation Term). Ilmentymätermiä voi tarkentaa määretermillä (Datatype Qualifier). Esimerkiksi jos kyseessä on koodi, määretermiksi voi laittaa käytettävän koodiston nimen. ASBIE-komponenttien esitysmuoto on linkki toiseen ABIE-komponenttiin. Linkin vastaavuus CCTS-mallissa on Associated Object Class.

Mikäli tietokomponenteista muodostetaan tarkennettuja (Qualified) versioita, myös objekti-termeihin ja ominaisuustermeihin voi lisätä määretermejä tarkennetuissa komponenteissa.

*Määritelmä* on yksi tärkeimmistä sarakkeista. Määritelmät ovat CCTS-mallissa pakollisia, ja Definition-niminen metatieto sisältyy jokaisen komponentin kuvaukseen. Määritelmät siirtyvät tietokomponenttitaulukoista XML-skeemoihin asti muuttumattomina.

Sarakkeen *selitys/esimerkki* sisältö vastaa CCTS-mallissa komponentin käyttöohjeita (Usage Rules). Mallin mukaan käyttöohjeet voivat olla tekstimuotoisia tai rakenteisia. Hankkeessa tuotetaan tekstimuotoisia käyttöohjeita.

*Kentän toistuvuus* ja *kentän pakollisuus* määrittelevät BBIE- tai ASBIE-komponentin kardinaliteetin. CCTS-mallissa kardinaliteetti ilmaistaan komponentin esiintymien minimi- ja maksimimääränä (minimum occurrence ja maximum occurrence). Määrät ovat ei-negatiivisia kokonaislukuja, maksimimäärän voi määritellä myös rajattomaksi (unbounded). Mikäli kentän pakollisuus tietokomponenttitaulukossa on "ei", niin minimimääräksi tulee 0, ja muuten 1. Mikäli kenttä on määritelty toistuvaksi, maksimimääräksi tulee "unbounded", ja muissa tapauksissa 1.

Sarakkeilla *asiakirja*, *lähde*, *päivämäärä*, *muutokset*, *versio*, *vastuuhenkilö* ja *kommentti* ei ole tarkkaa vastinetta CCTS-mallissa. Sarakkeet kuvaavatkin lähinnä tietokomponenttityön kulkua, ei työn lopputulosta.

## 5.5 Tietokomponenttien tarkentaminen

---

Tietokomponenttien suunnittelussa kannattaa pitää mielessä ainakin seuraavat tietomallinnuksen ominaisuudet: täydellisyys, toistumattomuus, tiedon uudelleenkäyttö, vakaus ja joustavuus, eleganttius (ks. Luku 3). Nämä ominaisuudet puoltavat useimmiten mahdollisimman yleiskäyttöisten tietokomponenttien muodostamista.

Hyvä esimerkki yleiskäyttöisestä tietokomponentista on `Osoite`. Tietokomponentti on muodostettu asiakkaan perustiedoissa (Kortelainen 2006) määritellyistä tietoryhmistä "Vakinainen kotimainen osoite", "Tilapäinen kotimainen osoite", "Postiosoite", "Vakinainen ulkomainen osoite", "Tilapäinen ulkomainen osoite" ja "Paras osoite". Kaikki nämä tietoryhmät sisältävät samankaltaista tietoa, ja niiden erot jäävät liian pieniksi erillisten tietokomponenttien muodostamiseksi. Myös tietoryhmien semanttiset merkitykset ovat päällekkäisiä: mikä tahansa kotimainen tai ulkomainen osoite voi olla postiosoitteena tai parhaana osoitteena.

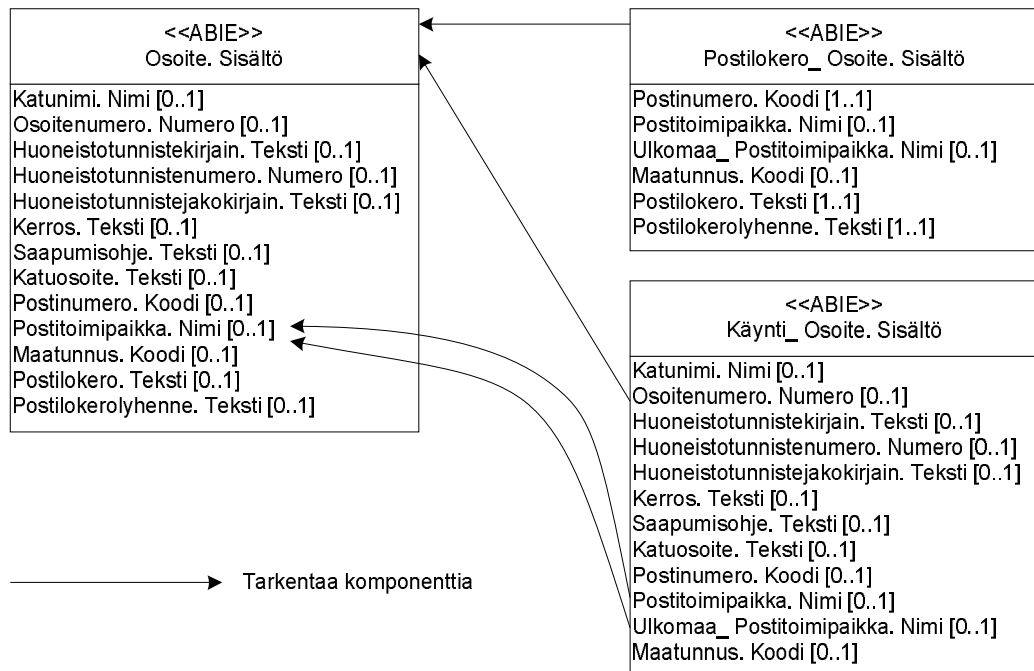
Tietotarpeiden kartoittamisessa, käsiteanalyysissä ja domain model -tason tietomallinnuksessa objekteja analysoidaan usein *ylä-* ja *alakäsitteiden* (ja vastaavasti *ylä-* ja *alaluokkien*) avulla. Alaluokat ovat yläluokan tarkennuksia. Yllä olevassa esimerkissä `Osoite` on yläluokka, ja kaikki tarkemmat osoitteet ovat sen alaluokkia. Ylä- ja alaluokkien mallintamiseen ja luokkien suhteiden ylläpitämiseen on olemassa useita eri tekniikoita.

### 5.5.1 CCTS-mukainen tarkentaminen

---

CCTS-mallissa objektiluokista voi muodostaa useita eri tarkennettuja versioita. Mekanismin englanninkielinen nimi on *Qualification*. Yläluokka tarkennetaan lisäämällä sen nimeen objektiluokan määretermi (object class qualifier) ja tarkentamalla sen sisältöä. Alaluokat voivat käyttää osajoukkoja yläluokan sisältämistä komponenteista. Luokan sisältämien komponenttien

nimiä voi lisäksi muokata lisäämällä niihin määretermejä. Myös kardinaliteettimäärytyksiä voi tiukentaa. Kuvassa 24 on esimerkki *Osoite*-komponentin tarkentamisesta postilokero-osoitteeksi ja käyntiosoitteeksi. *Osoite* on yläluokka, joka esittää mitä tahansa osoitetta. Alaluokat *Postilokero\_ Osoite* ja *Käynti\_ Osoite* käyttävät vain tarpeellisia osajoukkoja yläluokan komponenteista. Lisäksi postitoimipaikan nimi ilmestyy tarkennetuissa komponenteissa sekä alkuperäisessä että tarkennetussa muodossa (*Ulkomaa\_ Postitoimipaikka*). Myös kardinaaliteettimäärytyksiä on esimerkissä tarkennettu: postilokero-osoitteessa postinumero ja postilokero ovat nyt pakollisia komponentteja.



Kuva 24: Esimerkki CCTS-tarkentamisen käytöstä.

Tietokomponenttien tarkentaminen voi tapahtua muun muassa palvelutehtäväkohtaisesti: esimerkiksi komponenttia *Tulo* voi tarkentaa toimeentulotuen kontekstissa muodostamalla siitä *Toimeentulotuki\_ Tulo* -komponentti. Tarkennettuun komponenttiin voi poimia vain tarvittavia emokomponentin kenttiä ja tarkentaa niitä tarvittaessa. Esimerkiksi BBIE-kenttää *Määrä* voi tarkentaa kentäksi *Huomioitu\_ Määrä* CCTS-sääntöjen mukaisesti. Ominaisuustermiä *Määrä* tarkentavan määretermin *Huomioitu* vastaavuus CCTS-mallissa on Property Term Qualifier.

XML-skeemoissa, jotka pohjautuvat CCTS-komponentteihin, alaluokat voidaan liittää yläluokkiin ainoastaan dokumentaatiotasolla (*xs:documentation*- ja *xs:annotation*-elementtien avulla). Alaluokista muodostettuja XML-tyyppejä ei linkitetä yläluokan XML-tyyppiin teknisellä tasolla. Voidaan sanoa, että CCTS-mukainen tarkentaminen ei näy fyysisen tason tietomallissa.

Tarkentamisen käyttöä on kokeiltu hankkeessa toimeentulotuen asiakasasiakirjojen mallintamisen yhteydessä. Tarkentaminen kasvattaa tietokomponenttien lukumäärää merkittävästi muodostamalla useita eri versioita samasta tietokomponentista jopa saman palvelutehtävän sisällä. Tämä tekee jo valmiiksi monimutkaisesta tietomallista vielä monimutkaisemman ja

hankaloittaa sen ylläpitoa. Lisäksi tietojen siirrettävyys asiakirjasta toiseen tulee vaikeammaksi. Tarkentamismekanismien käyttöä voi harkita tulevaisuudessa, mikäli tunnistetaan selkeitä tietokomponenttikohtaisia kenttien osajoukkoja.

### 5.5.2 Periytyminen

---

Periytyminen (extension) perusajatuksena on se, että yläluokka sisältää kaikille alaluokille yhteisiä ominaisuuksia. Alaluokat perivät kaikki yläluokan ominaisuudet ja lisäävät niihin omia ominaisuuksiaan. Yläluokkaa voidaan näin ollen kutsua alaluokkien yhteiseksi nimittäjäksi. Periytymistä käytetään laajasti ohjelmistojen suunnittelussa, jossa hyödynnetään usein UML-luokkakaavioita. Monissa olio-ohjelmointikielissä periytyminen on erittäin tärkeä perustyökalu.

Tietomallinnuksessa periytymistä hyödynnetään usein käsitelmien ja domain model -tason tietomallien rakentamisessa. Periytyminen helpottaa muun muassa päällekkäisten objektiluokkien löytämistä ja niiden sisällön analysointia. Yläluokkien muodostaminen ei ole aina helppoa. Yllä olevassa osoite-esimerkissä alaluokkien postilokero-osoite ja käyntiosoite yhteiseksi ominaisuusjoukoksi jäisi vain postinumero, postitoimipaikan nimi, ja maatunnus. Lopputulos on hämäävä, koska nämä ominaisuudet eivät kuvaa enää käsitettä Osoite, vaan käsitettä Postitoimipaikka.

Tietokantakeskeisessä tietomallinnuksessa alaluokat muodostetaan niin, että (Simsion ja Witt 2005):

- *Alaluokat eivät ole päällekkäisiä.* Yllä olevassa esimerkissä osoite voi olla joko postilokero-osoitteena tai käyntiosoitteena, ei yhdistelmänä
- *Alaluokkien joukko on tyhjentävä.* Osoite on aina joko postilokero-osoite tai käyntiosoite, mikään muu vaihtoehto ei ole mahdollinen.

Osoite-esimerkissä alaluokkien joukko ei välttämättä ole tyhjentävä, jos halutaan mallintaa sellaisiakin postiosoitteita, jotka eivät ole postilokeroisten osoitteita eivätkä käyntiosoitteita. Postiosoite-nimisen alaluokan lisääminen voi kuitenkin rikkoa ensimmäisen säännön, koska sama osoite toimii usein sekä postiosoitteena että käyntiosoitteena.

Tietokantakeskeisissä loogisen ja fyysisen tason tietomalleissa periytymistä ei hyödynnetä sellaisenaan, koska sitä ei voi toteuttaa nykyisissä relaatiotietokannoissa aivan suoraan. XML-skeemarakenteissa periytymiselle on olemassa vastaavuus, `xs:extension`-rakenne. Periytyminen lisäksi XML-skeemoissa voi myös rajoittaa komponentteja `xs:restriction`-rakenteen avulla poimimalla rajoitettuun objektiluokkaan vain tarvittavia emokomponentin ominaisuuksia. Periytymistä ja rajoittamista hyödynnetäänkin useassa eri XML-dialektissa. Esimerkiksi yhdysvaltalainen NIEM perustuu vahvasti periytyminen ja rajoittamisen käyttöön.

XML-keskeisessä tietomallinnuksessa alaluokkien ei tarvitse olla päällekkäisiä ja tyhjentäviä. Periytymistä hyödynnetäänkin usein niin, että yläluokkaan otetaan mukaan laajahko ominaisuusjoukko, jota voi laajentaa ja/tai rajoittaa alaluokkien avulla.

Tikesos-hankkeessa periytymistä voitaisiin hyödyntää palvelutehtäväkohtaisten tietokomponenttien muodostamiseen ydintietokomponenteista. Periytyminen avulla ydinkomponentteihin voi lisätä palvelutehtäväkohtaisia kenttiä. Esimerkiksi tietokomponenttia `Taloudellinen`

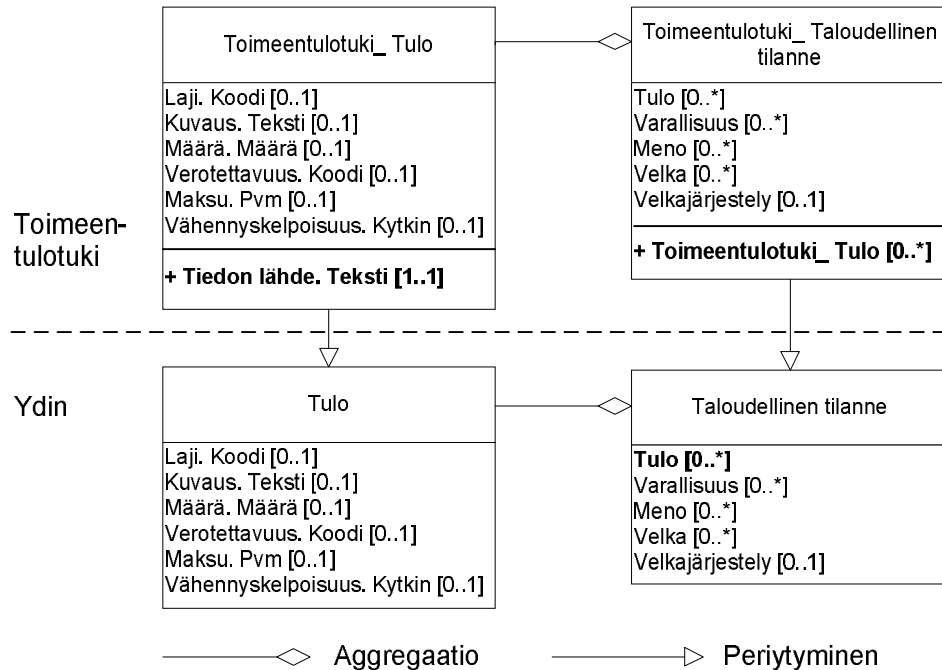
tilanne voi laajentaa toimeentulotukikontekstissa lisäämällä siihen ASBIE-ominaisuus Perusosa. Perusosa on palvelutehtäväkohtainen tietokomponentti, jolla on käyttöä vain toimeentulotuen palvelutehtävässä.

Palvelutehtäväkohtaisten laajennusten liian ahkera käyttö saattaa johtaa siihen, että samantyyppiset laajennukset tehdään useissa eri palvelutehtävissä samaan ydinkomponenttiin. Lisätyt kentät voivat jopa saada eri nimet eri palvelutehtävissä, vaikka niiden semanttinen merkitys olisi sama. Tämä kannattaa ottaa huomioon komponenttikirjaston ylläpitoa suunniteltaessa. Palvelutehtäväkohtaisia laajennuksia on hyvä käydä välillä läpi yhtenäistämistarpeiden tunnistamiseksi. Jos sama laajennus on käytössä useissa eri palvelutehtävissä, se on syytä liittää suoraan ydinkomponenttiin komponenttikirjaston päivityksen yhteydessä. Samaa ydinkomponenttia ei kannata laajentaa yhdessä palvelutehtävässä kuin kerran, koska eri laajennusten semanttiset erot voivat jäädä liiankin pieniksi, ja komponenttien määrä voi kasvaa turhan nopeasti.

### *5.5.3 Substituutioryhmät ja lisäykset*

---

Vaikka periytyminen näyttääkin toimivalta käsitteellisen tai domain model -tason tietomallinnuksessa, ja vaikka sen toteutus on XML-rakenteissa mahdollista myös loogisella ja fyysisellä tasolla, ei periytyminen käyttö ole aina ongelmaton. Kuvassa 25 on esimerkki tietokomponentin laajentamisesta periytyksen avulla. Esimerkissä lisätään uusi ominaisuus "Tiedon lähde" toimeentulotuen kontekstissa ydinkomponentitasolla olevaan Tulo-tietokomponenttiin. Tulo-tietokomponenttia hyödynnetään myös tietokomponentin Taloudellinen tilanne ominaisuutena. Muutos ei kuitenkaan välity automaattisesti tähän tietokomponenttiin, vaan siitä joudutaan ensin tekemään laajennettu versio nimeltään Toimeentulotuki\_Taloudellinen tilanne. Laajennetussa versiossa ominaisuus Toimeentulotuki\_Tulo ei kuitenkaan korvaa vanhaa Tulo-ominaisuutta.



Kuva 25: Periytyksen ongelmat loogisen tietomallin tasolla. Laajennettu tietokomponentti *Toimeentulotuki\_Tulo* ei korvaa vanhaa *Tulo*-ominaisuutta komponentissa *Taloudellinen tilanne*.

Samanlainen ongelma on huomattu myös yhdysvaltalaisen NIEMin tietomallin suunnittelussa. Ratkaisuksi on ehdotettu *substituutioryhmien* (Substitution group) ja *lisäysten* (Augmentation) käyttöä. Nämä tekniikat voivat toimia periytyksen rinnalla ja täydentää sitä.

Substituutioryhmä on XML Schema -kielen keino ilmaista XML-elementin korvattavuus toisella elementillä. Substituutioryhmän avulla voi määritellä sellaisia sääntöjä kuin

- "Siinä missä voi käyttää elementtiä *Nimi*, voi käyttää myös elementtiä *Name*"
- "Osoite voi ilmaista elementillä *Rakenteinen\_ Osoite* tai *Tekstimuotoinen\_ Osoite*"
- "Kaikissa paikoissa missä voi käyttää elementtiä *Organisaatio*, voi käyttää myös sen tarkennettua versiota *Julkishallinnon\_ Organisaatio*".

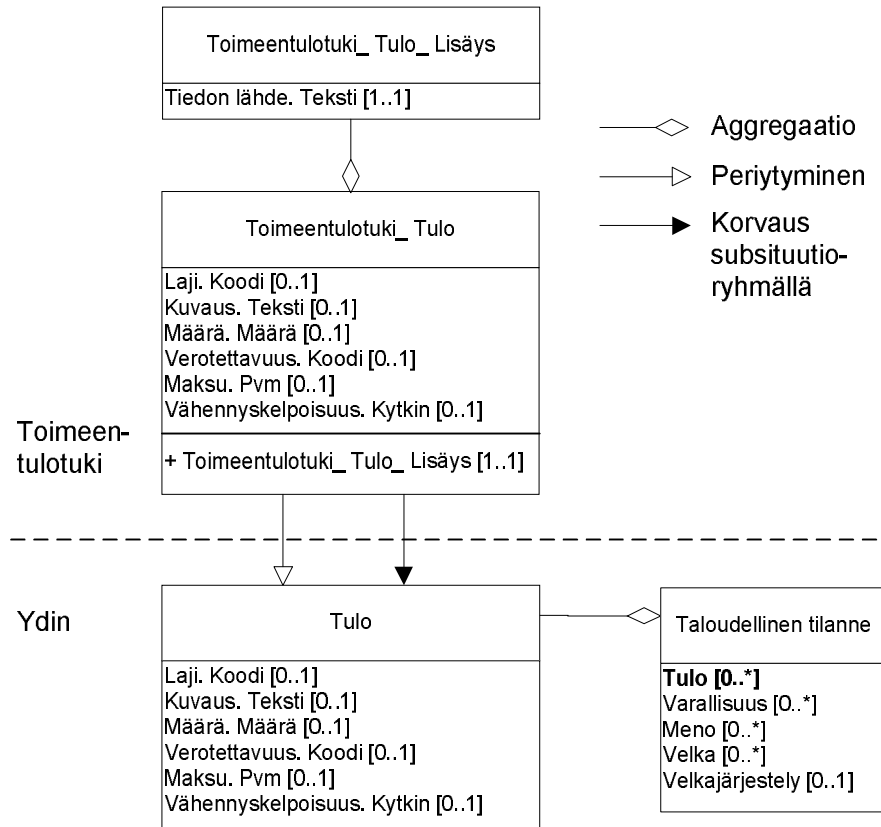
Uusi elementti voi joko periä vanhan elementin sisällön (*xs:extension*) tai rajoittaa sen sisältöä (*xs:restriction*). Aivan mielivaltaisia korvauksia mekanismin avulla ei siis voi tehdä, vaan linkki vanhan elementin määrittelyyn on aina säilytettävä.

Substituutioryhmällä määritelty sääntö ei tarkoita vanhan elementin ehdotonta korvausta uudella elementillä. Ne komponentit, joiden sisällössä on käytetty vanhaa elementtiä, voivat edelleen käyttää sitä. Sääntö antaa vain mahdollisuuden käyttää näissä paikoissa myös uutta elementtiä. Kuvassa 26 esitetty esimerkki havainnollistaa substituutioryhmien käytön. Komponentti *Toimeentulotuki\_Tulo* on muodostettu laajentamalla *Tulo*-komponenttia (uutena ominaisuutena on *Toimeentulotuki\_Tulo\_Lisäys*). Uusi laajennettu komponentti on myös määritelty vanhan *Tulo*-komponentin korvaajaksi substituutioryhmän avulla. Komponentissa *Taloudellinen tilanne* voi nyt käyttää jompaakumpaa näistä *Tulo*-komponenteista.

Uusi kenttä Tiedon lähde on voitu lisätä myös suoraan laajennettuun tietokomponenttiin Toimeentulotuki\_Tulo. Esimerkissä lisäys on tapahtunut komponentin Toimeentulotuki\_Tulo\_Lisäys avulla. Kyseessä on NIEM-spesifi sääntö, jonka tarkoituksena on mahdollistaa eri toimialoilla tehtyjen lisäysten samanaikainen käyttö uusissa komponenteissa. Säännön perusteluna on komponenttien uudelleenkäytettävyys. Esimerkiksi sosiaalisen luotuksen palvelutehtävissä voidaan ainakin teoriassa hyödyntää toimeentulotuen palvelutehtävässä tehtyjä laajennuksia. Tällaisia uudelleenkäyttötarpeita ei tietomallissa ole toistaiseksi havaittu, joten lisäyskomponenttien käyttö vaikuttaa tässä vaiheessa sekavalta. Toisaalta, lisäyskomponenttien selkeä erottaminen voi olla hyödyllistä jos tietomallista kehitetään tietokantakeskeinen versio.

Substituutioryhmille ei ole vastaavuutta tietokantakeskeisessä tietomallinnuksessa loogisella tai fyysisellä tasolla. Lisäyskomponentit voidaan toteuttaa relaatiotietokannoissa erillisinä tauluina. CCTS-mallissa XML-suunnittelusäännöt kieltävät substituutioryhmien käytön. Kielto on kuitenkin luonteeltaan lievä: substituutioryhmien käyttö rikkoo yhteensopivuuden CCTS-tietomallin kanssa, mutta organisaatiot voivat hyödyntää substituutioryhmiä omissa projekteissaan. Tikesos-hankkeessa ei varsinaisesti hyödynnetä CCTS-tietomallia, vaan rakennetaan omaa tietomallia CCTS-periaatteilla.

Substituutioryhmien käyttöä on pohdittu hankkeessa aiemminkin. Silloin todettiin, että "Tyypien periytymisessä substituutioryhmien käyttö on NIEMissä selvästi perusteltua. UN/CEFACT-malli kuitenkin kieltää substituutioryhmien käytön. Tikesos-asiakirjoissa substituutioryhmien käyttö ei ole nähty välttämättömäksi. Lisäksi se on todettu hieman hankalaksi näyttömuotomäärittysten kehityksessä." (Huttunen ym. 2009). Tämän pohjalta luotiin sääntö "Komponenttien määrittelyissä EI SAA käyttää `xs:substitutionGroup`-rakennetta." Sääntö perustui UN/CEFACT-mallin silloin voimassa olleeseen versioon (UN/CEFACT 2006). Uudessa standardin versiossa (UN/CEFACT 2009b) standardiin on tullut lievennyksiä substituutioryhmien käytön suhteen, joten tilannetta voidaan tarkastella uudestaan uuden tiedon valossa hankkeen todellisten tarpeiden pohjalta.



Kuva 26: Lisäysten ja substituutioryhmien käyttö tietomallinnuksessa. *Toimeentulotuki\_Tulo* on nyt käytettävissä kaikissa niissä paikoissa joissa voi käyttää alkuperäistä *Tulo*-komponenttia.

#### 5.5.4 Yhteenveto tarkentamisen keinoista

Tietokomponenttien eri tarkennustekniikoilla on omia hyviä ja huonoja puoliaan. CCTS-tarkentamisen suurin ongelmakohta on se, ettei tarkennetuissa komponenteissa voi olla aivan uusia ominaisuuksia, vaan jokaiselle uudelle kentälle on löydettävä vastaavuus emokomponentista. Periytyminen on oiva tekniikka käsitemallinnuksessa, mutta loogisen tason tietokantakeskeisen tietomallin rakentamisessa sitä ei voida käyttää suoraan. Periytymistä voi sen sijaan hyödyntää XML-rakenteissa, vaikka se joudutaan usein yhdistämään substituutioryhmien käyttöön. Substituutioryhmien käyttöä ei kuitenkaan suositella CCTS-mallissa.

Hankkeessa ei ole toistaiseksi hyödynnetty mitään tietokomponenttien tarkentamiskeinoa, koska perustarkoituksena on ollut eri palvelutehtävissä käytettävien tietojen vahva yhtenäistäminen. Periytyminen tai vastaavan tekniikan käyttö olisi tehnyt tämän työn hankalammaksi. Tietokomponenttien tarkentamista voidaan tarvita kuitenkin tulevaisuudessa.

Tietojen yhtenäistäminen on hankkeessa edelleen menossa, ja tarkentamisen käyttöönotto voi tässä vaiheessa olla ennenaikaista. Mikäli tarkentaminen koetaan tarpeelliseksi, ehdotamme seuraavaa:

- XML-keskeisessä tietomallinnuksessa voidaan ottaa käyttöön periytymisen (`xs:extension`) ja substituutioryhmien (`xs:substitutionGroup`) yhdistelmä. Päätös lisäyskomponenttien käyttöön otosta riippuu lisäyskomponenttien uudelleen käytettävyydestä.
- Tietokantakeskeisen loogisen tason tietomallin rakentamisessa tulee välttää komponenttien tarkentamisen käyttöä.

## 5.6 Roolien esittäminen tietomallissa

---

Luvussa 2.2 henkilöiden ja organisaatioiden lukuisia rooleja ja niiden päällekkäisyyttä kuvattiin sosiaalihuollon erityispiirteinä. Rooleja pitää pystyä esittämään tietomallissa riittävällä tarkkuudella.

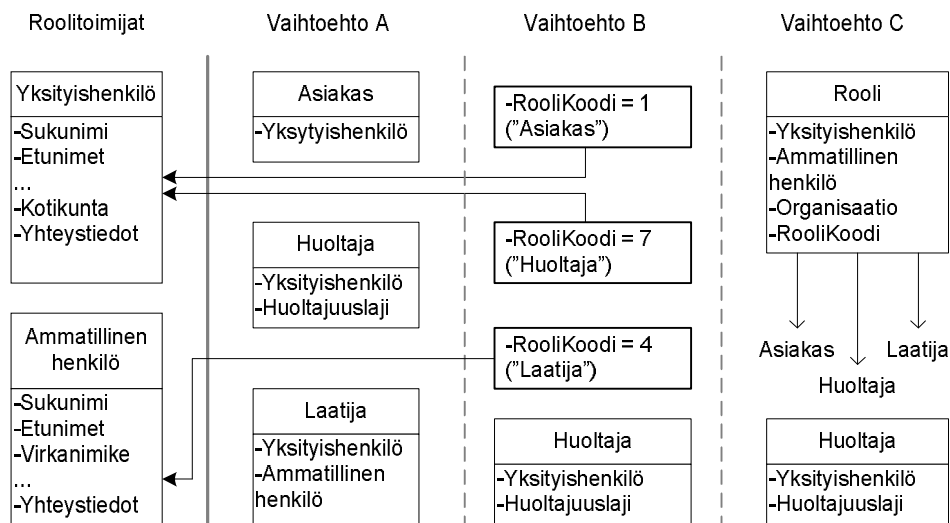
Roolien esittämistä XML-tasolla on kuvattu sosiaalihuollon asiakasasiakirjojen teknisissä määrittelyissä (Huttunen ym. 2009). Esitystapa perustuu siihen, että rooli muodostaa tietokomponentin, josta on linkki toimijaan eli yksityishenkilöön, ammatilliseen henkilöön tai organisaatioon. Ehdotus pohjautuu siihen oletukseen, että roolikomponenteissa olevilla toimijoilla voi olla myös roolikohtaisia ominaisuuksia. Esimerkiksi toimija `Yksityishenkilö` voi saada muitakin ominaisuuksia, jos hän on roolissa `Huoltaja`, koska halutaan tallentaa tieto siitä, onko huoltaja määräysenvarainen. Oletus on pitänyt paikkansa, mutta tällaisia roolikohtaisia tietoja on tullut tietomalliin hyvin vähän. Useimmiten rooli on ilmestynyt sisältömäärittelyissä eräänlaisena otsikkona, joka liitettiin toimijaan. Jokaisen roolin muuttaminen tietokomponentiksi on kasvattanut tietokomponenttien määrää ehkä tarpeettomastikin.

Muitakin mahdollisia esitystapoja on olemassa, ja niitä käydään tässä lyhyesti läpi. Kuvassa 27 on esitetty kolme vaihtoehtoa roolien kuvaukseen XML-keskeisen tietomallin tasolla. Vaihtoehto A vastaa aiemminkin käytettyä esitystapaa: tässä esimerkissä `Asiakas`, `Laatija` ja `Huoltaja` muodostavat tietokomponentteja. Tietokomponentit `Asiakas` ja `Laatija` sisältävät vain viittauksia toimijoihin. Tietokomponentissa `Huoltaja` on toimijan lisäksi myös roolikohtainen ominaisuus `HuoltajuuSLaji`. Vaihtoehdon A hyvänä puolena on se, että roolikohtaisia ominaisuuksia on helppo lisätä tietomallin kehittyessä, jos niitä ilmenee. Lisäksi tietokomponenteilla on helppo esittää toimijoiden valintaa koskevia sääntöjä: kuvan esimerkissä asiakkaana voi olla vain yksityishenkilö, kun taas laatijana sekä yksityishenkilö että ammatillinen henkilö. Vaihtoehdolla on kuitenkin huonojakin puolia. Tietokomponenttien määrä kasvaa, koska jokaisen uuden roolin lisäys aiheuttaa myös uuden komponentin lisäyksen. Rooleja ei tämän takia voi myöskään lisätä jälkikäteen ilman tietomallin muuttamista.

Vaihtoehdossa B asiakkaasta ja laatijasta ei muodosteta tietokomponentteja, koska näillä rooleilla ei ole roolikohtaisia ominaisuuksia. Roolitoimijoita kuvaaviin tietokomponentteihin lisätään uusi toistuva ominaisuus `RooliKoodi`. Kaikki mahdolliset roolit esitetään koodistona. Roolit ilmaistaan asiakirjainstansseissa lisäämällä sopiva roolikoodi roolitoimijan kuvaukseen. Esimerkissä asiakkaan rooli mallinnetaan lisäämällä roolikoodi 1 (asiakas) yksityishenkilöä kuvaavaan elementti-instanssiin. Laatijana esimerkissä on ammatillinen henkilö, joten roolikoodi 4 (laatija) on lisätty ammatillisen henkilön kuvaukseen. Huoltajakin kuvataan samalla tavalla, mutta tietokomponentti `Huoltaja` on silti tarpeen, koska huoltajuuSLajia ei voida ilmaista pelkän roolikoodin avulla. Vaihtoehto mahdollistaa roolilistauksen helpon laajentamisen jälki-

käteenkin, jos lisättävissä rooleissa ei ole roolikohtaisia ominaisuuksia. Vaihtoehdon huonona puolena on se, että roolikoodin lisääminen suoraan roolitoimijan kuvaukseen irrottaa roolin kontekstista. Jos kuvataan tällä tavalla asiakirjaa, konteksti on useimmiten tiedossa. Mutta jos tietomallin pohjalta rakennetaan relaatiotietokantaa, roolien liittäminen esimerkiksi Yksityishenkilö-taulukkoon on huono ajatus, koska tätä taulukkoa käytetään todennäköisesti muihin-kin tarkoituksiin hyvin monipuolisesti.

Viimeisessä vaihtoehdossa C rooli ilmaistaan `Rooli`-nimisellä tietokomponentilla. Tietokomponentti sisältää ainakin viittaukset kaikkiin mahdollisiin roolitoimijoihin ja roolikoodin. Roolikomponenttiin ei voida sisällyttää roolikohtaisia ominaisuuksia. Samassa esimerkissä `Huoltaja` joudutaan siis edelleen kuvaamaan erillisellä tietokomponentilla. Tietokomponentti `Huoltaja` voi perustua tietokomponenttiin `Rooli` (periytymissuhde) tai käyttää sitä yhtenä ominaisuutena (aggregaattiosuhde). Huoltajan kaltaiset roolikohtaisia ominaisuuksia sisältävät roolit voi mallintaa myös samalla tavalla kuin vaihtoehdossa A, koska `Rooli`-komponentin käyttö ei varsinaisesti lisää mitään uutta tietoa. Vaihtoehdon C hyvänä puolena on se, että samaa periaatetta käytetään usein myös tietokantakeskeisessä tietomallinnuksessa (Simsion ja Witt 2005, s. 125). Samoilla linjoilla on myös CCTS-tietomalli (UN/CEFACT 2010). Englanninkielisissä tietomalleissa roolikomponentista käytetään useimmiten sanaa `Party`. Ratkaisun huonona puolena voi olla kontekstin puute (samalla tavalla kuin vaihtoehdossa B) sekä se, että `Rooli` on jopa liiankin yleinen tietokomponentti. Tietomalleissa hyödynnetäänkin usein myös sen alaluokkia.

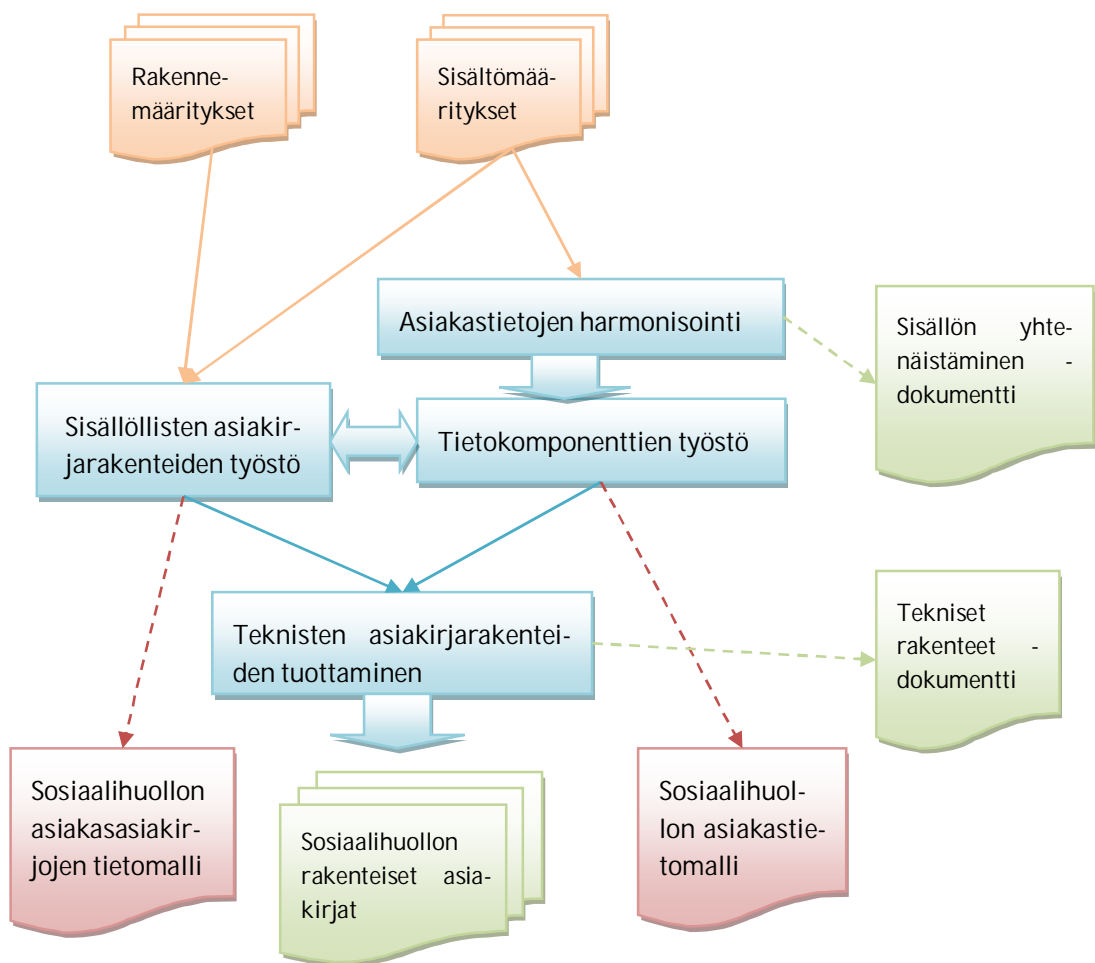


Kuva 27: Roolien esittäminen tietomallissa.

Tämän pohdinnan pohjalta ehdotetaan, että tietomallissa voi käyttää kaikkia yllä listattuja keinoja roolien esittämiseen. Mallinnuksessa tulee kuitenkin suosia vaihtoehdon C käyttöä.

## 6 Sosiaalihuollon asiakasasiakirjojen tietomallin- nus

Yleisesti sosiaalihuollon asiakirjat muodostuvat asiakkaan sen hetkisen elämäntilanteen ja palvelutarpeiden selvittämisestä. Asiakirjarakenteiden ja sisältömäärittysten kuvaukset voi karkeasti jakaa neljään osa-alueeseen: asiakirjojen varsinainen tietosisältö (*tietotarpeet*), *lomakkeet*, joilla tämä tietosisältö kerätään, *tekniset rakenteet*, joita käytetään tietosisällön tallentamiseen ja *näyttö/tulostusmuodot*, joita sovelletaan tallennetun tietosisällön esittämiseen (mukailleen Glushko ja McGrath 2002). Sisällölliset asiakirjarakenteet keskittyvät lähinnä teknisiin rakenteisiin ja näyttö/tulostusmuotoihin. Tietotarpeita ja lomakkeita ne kuvaavat vain pintapuolisesti. Seuraava kuva havainnollistaa asiakasasiakirjojen mallinnusprosessia:



Kuva 28. Sosiaalihuollon rakenteisten asiakirjojen ja tietomallien muodostuminen

Kuvassa on esitetty syötteet ja tehtävät, joiden lopputuloksena tuotetaan Sosiaalihuollon rakenteiset asiakirjat. Syötteinä toimivat Rakennemääritykset ja Sisältömääritykset. Rakennemääritykset (kuten Kortelainen 2008 tai Kortelainen ja Luukkonen 2009) sisältävät kuvaukset asiakirjojen yleisestä ja asiakirjatyypikohtaisista sisällöistä. Sisältömääritykset (kuten Kärki 2007 tai Väinälä 2007) sisältävät asiakkaista ja palvelunantajista kerättävät perustiedot sekä palvelukohtaiset minimitietovaatimukset asiakirjoittain. Sekä rakennemääritykset että sisältömääritykset kuvaavat lähinnä tietotarpeita, mutta ottavat usein kantaa muihinkin osa-alueisiin eli määrityksissä viitataan *esitysmuotoihin* (tekniset rakenteet) ja kuvataan mitä *tietoryhmiä ja -kokonaisuuksia* (tietotarpeet) tulee *kerätä* (lomakkeet) ja *näyttää* (näyttö/tulostusmuodot). Nämä osa-alueet ovat aikaisemmin tuotetuissa määrityksissä toisistaan usein vaikeasti erotettavissa ja toisaalta hankalasti yhteen sovitettavissa kokonaisuuden ymmärtämiseksi.

Syötteistä syntyvät tuotoksina asiakastiedon harmonisoinnista kertova Sisällön yhtenäistäminen -dokumentti, teknisten asiakirjarakenteiden tuottamista ohjeistava Tekniset rakenteet -dokumentti sekä Sosiaalihuollon rakenteiset asiakirjat eli XML-skeemat. Näiden ohella syntyvät myös tämän dokumentin kannalta olennaiset Sosiaalihuollon asiakastietomalli (luku 5) ja Sosiaalihuollon asiakasasiakirjojen tietomalli (tämä luku).

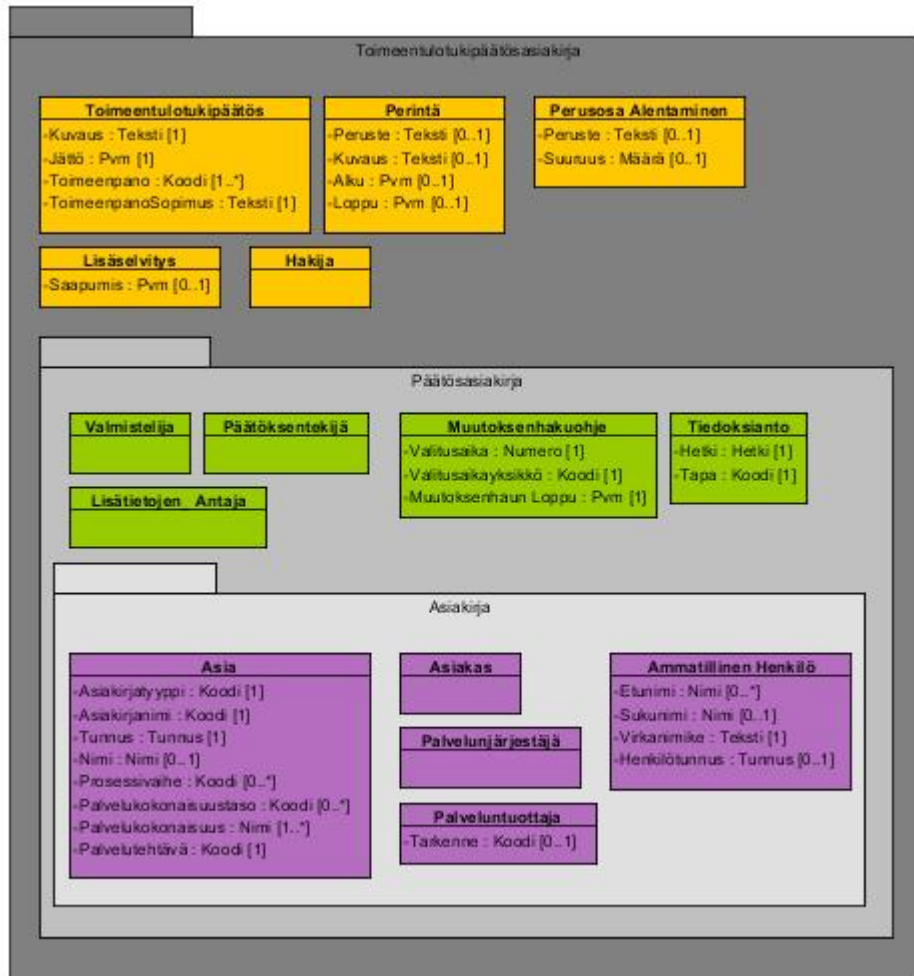
Luvussa 6.1 esitellään sisällöllisten asiakirjarakenteiden kokoamisen perusteita ja taustaa. Luvussa 6.2 esitellään asiakirjataulukoiden rakenne, johon liittyen liitteessä B on esitetty esimerkkitaulukko. Luvussa 6.3 kuvataan asiakirjarakennetaulukoiden kokoamisen työnkulku. Luvussa 6.4 selvitetään asiakirjataulukoiden ja CCTS-taulukoiden yhteys, ja luvussa 6.4 kuvataan lyhyesti CCTS-taulukoista tuotettavat tekniset rakenteet eli XML-skeemat.

## 6.1 Asiakirjarakenteet

---

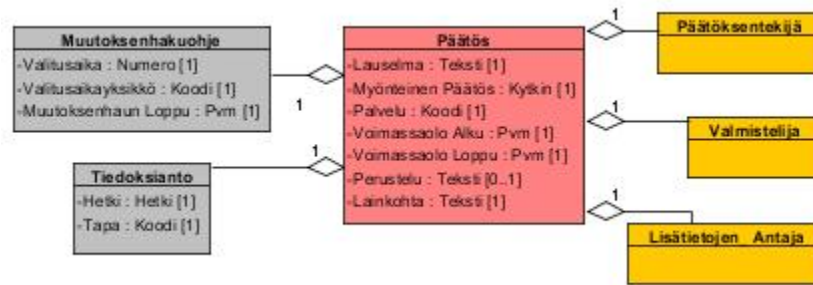
Asiakirjarakenteet on jaettu sisällöllisissä asiakirjarakenteissa kahteen osaan: *asiakasasiakirjojen yleisen rakenteen (AYR) -mukaisiin rakenteisiin* ja *asiakirjakohtaisiin rakenteisiin*.

AYR-mukaiset rakenteet on muodostettu kuvauksen (Kortelainen 2008), yleisten asiakirjatyypien rakennemääritysten (kuten Kortelainen ja Luukkonen 2009) ja metatietomääritysten (Paakkanen ym. 2009) pohjalta. Kaikki nämä rakenteet perustuvat erikseen määriteltyihin tietokomponentteihin.

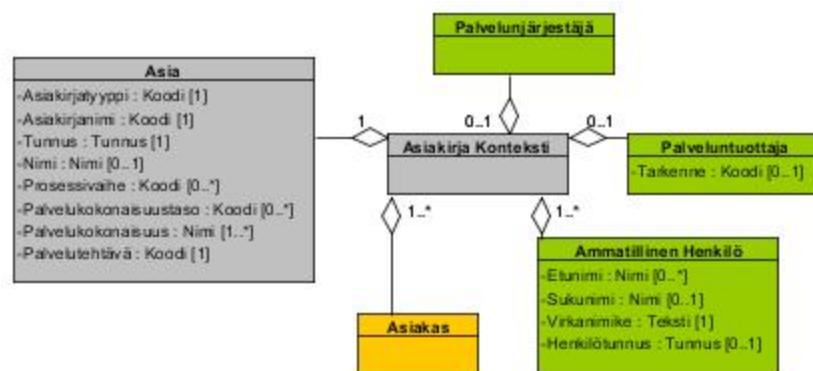


Kuva 29. Asiakirjan rakenteen kolme eri tasoa

Asiakasasiakirjoja mallinnetaan kuvan mukaisesti kolmella eri tasolla, jotka ovat seuraavat: asiakirjan yleinen rakenne (Asiakirja), asiakirjatyypikohtainen rakenne (Päätösasiakirja) sekä varsinaisen asiakasasiakirjan rakenne (Toimeentulotukipäätösasiakirja). Nämä tasot ovat koosteisia, eli esimerkkiasiakirjassa toimeentulotukipäätös koostuu asiakirjakohtaisista tiedoista (kuva 29), yleisistä päätöksen tiedoista (kuva 30) sekä yleisen asiakirjan tiedoista (kuva 31).



Kuva 30. Esimerkki asiakirjatyypikohtaisesta rakenteesta



Kuva 31. Esimerkki asiakirjan yleisestä rakenteesta

Asiakasasiakirjojen rakenteen määrittely on alkanut palvelukohtaisista tietomäärittelyistä (muun muassa Kärki 2007). Nämä tietomäärittelyt kertovat, mitä tietoja asiakirjojen tulee sisältää. Määrittelyjä on jatkoyöstetty Excel-taulukkomuotoon, jossa on otettu huomioon sisällön yhtenäistämistyön tulokset kuten tietokokonaisuuksien määrittelyt. Asiakirjat sisältävät usein asiakkaaseen ja asiakkaan elämäntilanteeseen liittyviä tietoja, joiden perusteella asiakas voi saada sosiaalipalveluja ja -etuuksia.

Asiakirjakohtaiset rakenteet sisältävät kaikki ne tiedot, joita ei ole huomioitu rooleissa, roolisuhteissa eikä AYR-mukaisissa rakenteissa. Esimerkiksi päivähoitohakemuksessa palvelutarpeen ajanjakso on määriteltävä asiakirjakohtaisena rakenteena, koska sitä ei ole kuvattu rooleissa, roolisuhteissa eikä hakemuksen yleisessä rakenteessa. Rakenteet perustuvat pääsääntöisesti tietokomponentteihin. Ainoan poikkeuksen tästä säännöstä muodostavat asiakirjakohtaiset yksinkertaisiin tietotyyppihin perustuvat kentät. Nämä kentät määritellään suoraan sisällöllisissä asiakirjarakenteissa. Kenttiä ei saa ryhmitellä asiakirjarakenteissa yhteisen otsikon alle. Mikäli tämä tarve ilmenee, kentistä muodostetaan uusi tietokomponentti ja käytetään sitä asiakirjarakenteen kuvauksessa. Näin pystytään tunnistamaan yhteisiä kenttärühmiä helpommin ja yhtenäistämään niiden sisältöjä. Asiakirjojen näyttö/tulostusmuodoissa voi olla lisäotsikoita, jotka eivät ole rakenteiden tai tietokomponenttien nimiä. Sisällöllisissä asiakirjarakenteissa näitä otsikoita ei ole vielä määritelty.

## 6.2 Sisällölliset asiakirjarakennetaulukot

---

Sisällölliset asiakirjarakenteet perustuvat erikseen määriteltyihin tietokomponentteihin. Sisällöllisillä asiakirjarakenteilla pyritään kuvaamaan taulukkomuodossa sisältö- ja rakennemäärittelyksissä esitetyt tietotarpeet käyttämällä tietokomponentteja. Tietokomponentit toimivat asiakirjarakenteissa tietojen tallennusmuotona, josta on suora linkki teknisiin rakenteisiin. Sisällöllisissä asiakirjarakennetaulukoissa otetaan kantaa myös näyttö/tulostusmuotoihin, eli kuvataan mitä tietoja ja missä järjestyksessä tulee poimia asiakirjaan tallennetuista tietokomponenteista käyttäjälle esittämistä varten. Tietokomponentit ja sisällölliset asiakirjarakenteet muodostavat sen kokonaisuuden, jolla sosiaalihuollon asiakirjarakenteet on kuvattu.

Sisällöllisiä asiakirjarakenteita on kuvattu Excel-taulukon muodossa (ks. esimerkki liitteessä B). Yhden asiakirjatyypin rakenteet on kuvattu aina yhdellä Excel-kirjan välilehdellä. Taulukossa ohjeistetaan myös sisällöllisten asiakirjarakenteiden tulostusjärjestys näyttömuotoa varten. Käytännön työn toteuttamisen, asiakirjahallinnon ja myös asiakkaan palvelun kannalta olisi hyödyllistä, että samaan tyyppiin kuuluvien asiakirjojen sisällön rakenne olisi samankaltainen. Esimerkiksi eri palvelutehtävien suunnitelmat tai hallintopäätökset sisältävät hyvin paljon samankaltaisia rakenneosia. Asiakirjojen tyyppien ohella asiakirjojen sisältöosan rakenteen muodostumiseen vaikuttaa asian käsittelyn luonnollinen järjestys. Monet toimintaprosessit etenevät tietyssä järjestyksessä ja tiedot kirjataan myös asiakirjoihin vastaavassa järjestyksessä. Asiakirjojen sisältämä asian käsittelyn luonnollinen järjestys on johdettavissa sosiaalihuollon toiminnallisista lähtökohdista. Tästä syystä asiakirjojen määrittelyn lähtökohtana tulee olla toiminnasta johdetut tietotarpeet ja vaatimukset tiedon esittämiselle. Taulukko muodostuu seuraavaksi selitetyistä sarakkeista.

*Tikesos-määrittely*, sarakkeet A–H. Sarake A sisältää roolien ja roolisuhteiden nimet ja asiakirjakohtaisten rakenteiden nimet. Se sisältää myös uusien asiakirjarakenteissa (eikä tietokomponenteissa) määriteltyjen kenttien nimet. Esimerkiksi päivähoidon palvelutarpeen selvityksessä lapsen kotona puhuttava kieli on määritelty asiakirjakohtaiseksi kentäksi, koska sitä ei ole määritelty lapsen perustiedoissa eikä missään muussa tietokomponentissa, jota tässä asiakirjassa käytetään. Sarakkeet B–H sisältävät tietokomponenttien osat, jotka tulevat asiakirjan *suppeaan* näyttö/tulostusmuotoon. Syynä tähän on se, että tietokomponenteissa on usein tulostuksen kannalta tarpeettomia tietoja. Esimerkiksi asiakkaan perustiedoissa on tietoa asiakkaan aikaisemmista kotikunnista, hänen aiemmin käyttämistään nimistä ja ammatista. Kaikki nämä kentät sisältävät tärkeitä taustatietoja ja niille on varattu paikka tietokomponentissa ja asiakirjarakenteissa. Asiakirjan suppeassa näyttö/tulostusmuodossa niitä ei kuitenkaan aina ole järkevä esittää.

Joissakin tapauksissa sarakkeet B–H eivät kuvaa näyttö/tulostusmuotoa vaan kertovat tietotarpeista ja ohjaavat lomakkeiden muodostamista. Tässä tapauksessa sarakkeen L (näyttömuoto-ohje) teksti on "Kaikki kentät tulostetaan".

*Tiedon esitysmuoto*, sarake I. Sarake sisältää viittauksen tietokomponenttiin, jota käytetään sarakkeessa A nimetyn rakenteen esittämiseen. Mikäli sarakkeessa A on asiakirjakohtainen kenttä, tiedon esitysmuotona voi olla joku yksinkertainen perustyyppi, kuten teksti tai päivämäärä.

*Dokumentaatio*, sarake J. Käytetään vain jos kyseessä on asiakirjakohtainen rakenne. Sarake J sisältää viittauksen sisältömäärittelyyn, jossa tämän rakenteen sisältö on mainittu. Koska suoria linkkejä dokumentaatioon ei aina löydy, solun sisältö voi olla tyhjä.

*Määritelmä*, sarake K. Käytetään vain jos kyseessä on asiakirjakohtainen rakenne. Sarake K sisältää tämän rakenteen (esimerkiksi yksinkertaisen kentän) määritelmän.

*Näyttömuoto-ohje*, sarake L. Ohje koskee vain suppeaa näyttö/tulostusmuotoa. Sarake täydentää sarakkeita B–H ja sisältää lisää ohjeita näyttö/tulostusmuodon laatimiseen. Sarakkeen joi-takin yleisiä sisältöjä on kuvattu alla:

- *Ei näyttömuodossa*. Rakennetta ei tulosteta, vaikka se löytyisikin asiakirjasta.
- *Käytetään yhtenäistä ohjetta*. Osa tiedoista tulostetaan asiakirjan ylä- tai alatunnisteeseen tai jollakin yhtenäisellä tavalla asiakirjan alku- tai loppuosaan. Yhtenäisten ohjeiden muodostaminen sisältyy hankkeen 2010 työsuunnitelmaan.
- *Ks. roolimääritelmä*. Rakenne on rooli, ja sen kuvaus löytyy taulukon alkuosasta.
- *Kaikki kentät tulostetaan*. Kyseisen rakenteen kuvauksessa sarakkeet B–H kertovat tietotarpeista ja ohjaavat lomakkeiden muodostamista. Mikäli tietokomponentti löytyy asiakirjasta, kaikki sen kentät tulostetaan suppeaankin näyttömuotoon.

*Selitys/esimerkki*, sarake M. Sisältää lisäohjeita, vakiotekstejä tai esimerkkejä rakenteen sisäl-löstä.

*Toistuva*, sarake N. "Kyllä" tarkoittaa, että rakenne tai kenttä voi asiakirjassa toistua. Esimerkik-si toimeentulotukihakemuksessa lapsen taloudellinen tilanne on toistuva rakenne, koska per-heessä voi olla useita lapsia. "Ei" tarkoittaa, että rakenne tai kenttä voi esiintyä vain kerran.

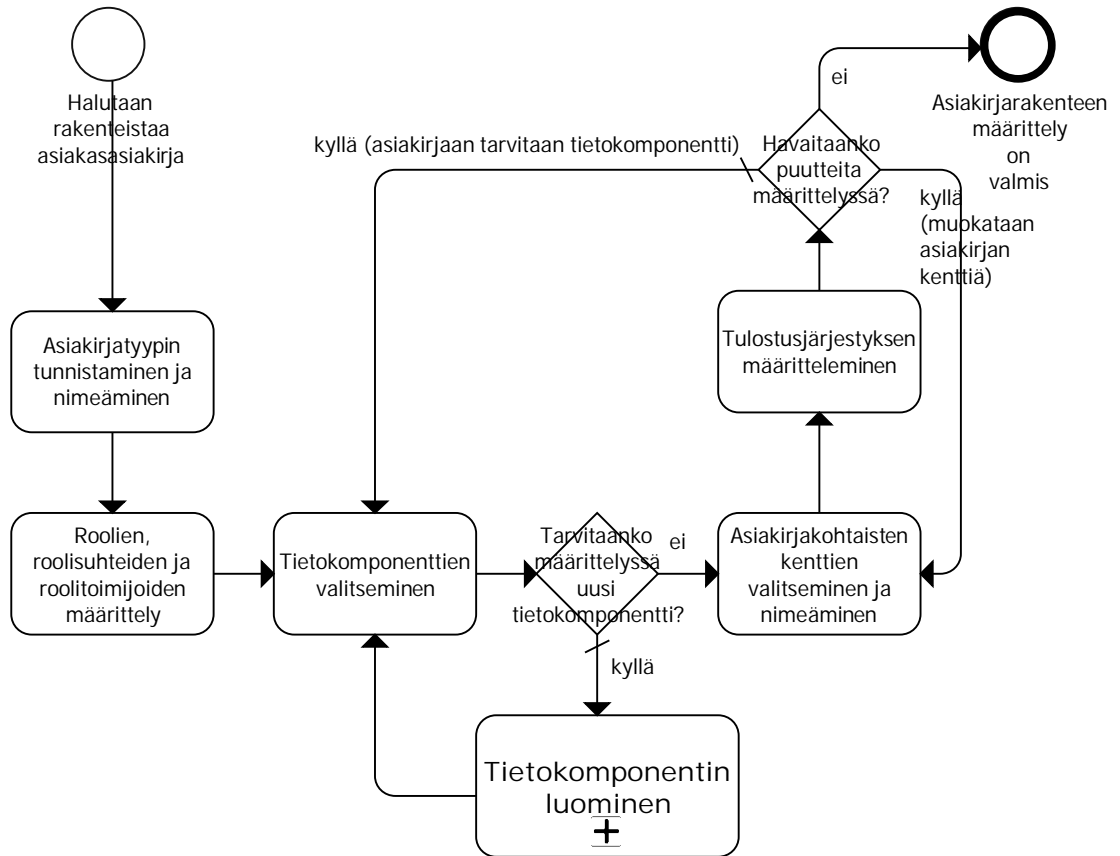
*Pakollinen*, sarake O. "Kyllä" tarkoittaa, että rakenteen tai kentän tulee esiintyä asiakirjassa ainakin kerran. "Ei" tarkoittaa, että rakenne tai kenttä on valinnainen. Rakenteet on pyritty tekemään mahdollisimman joustaviksi. Niissä tapauksissa, joissa rakenteen pakollisuudesta ei ole päätetty, rakenne on määritelty valinnaiseksi.

*Järjestys*, sarake P. Sarake on osa näyttö/tulostusmuoto-ohjeita. Sarake sisältää numerot 1...n, jotka kertovat missä järjestyksessä rakenteet ja kentät ilmestyvät näyttö/tulostusmuodossa. Jos rakenteen sisältö on esitetty tarkemmalla tasolla sarakkeilla B–H, mutta järjestysnumeroa ei löydy, tämä tarkoittaa, että rakenteen sisällön esittämisjärjestykseen ei oteta kantaa, vaan sen voi tulostaa missä tahansa järjestyksessä. AYR-mukaisten asiakirjarakenteiden listauksessa järjestysnumerot "Asiakirjan" tai yleisen asiakirjatyypin (kuten "Hakemus") kohdalla järjestys-numero viittaa lähinnä asiakirjan otsikkoon ja kehottaa käyttämään yhtenäistä ohjetta. Yhte-näistämiseen on pyritty myös muiden rakenteiden järjestyksen osalta.

### 6.3 Sisällöllisten asiakirjarakenteiden määrittelyprosessi

Sisällöllisten asiakirjamäärittelyjen kuvaustapaa (kuva 32) voidaan hyvin kuvata iteratiiviseksi kehittämiseksi; useamman kierroksen jälkeen (esimerkiksi erilaisten otsikointien ja erillisten

tietokomponentti-välilehtien mallintamisen avulla) määrittelytyössä päädyttiin koko ajan lähemmäksi nyt mallinnettua teknisessä määrittelytyössä helpommin hyödynnettävää esitystapaa, jonka luontia ja rakennetta kuvataan tarkemmin luvussa 6.2.



Kuva 32. Sisällöllisten asiakirjarakenteiden määrittelyprosessi

Sisällöllisten asiakirjarakenteiden määrittely alkaa, kun hankkeen sisällön asiantuntijat jaottelvat ja kokoavat käsiteltävänä olevan palvelutehtävän sisällölliset tietomääritykset asiakirjarakenteisiin eli tarkentavat, mikä tietosisältö kuuluu mihinkin asiakirjatyyppiin ja minkä nimisen suppean asiakirjatyypin nämä tiedot muodostavat. Joidenkin palvelutehtävien osalta tietotarpeet on jo määritelty asiakirjallisina tietosisältöinä, kun taas toisissa tehtävissä sisällöllisesti yhteenkuuluvina tietokokonaisuuksina, joita tarvitaan palveluprosessin eri vaiheissa. Kun tarvittavat suppeat asiakirjatyypit on tunnistettu ja nimetty, alkaa tiivis yhteistyö sisällön asiantuntijoiden ja teknisten asiantuntijoiden kesken. Aivan kuten tietokomponenttien kokoamisessa myös sisällöllisten asiakirjarakenteiden määrittelytyössä täytyy huomioida tietoelementtien semanttinen yhteys ja tekninen esitysmuoto. Sisällöllisten asiakirjarakenteiden kokoaminen aloitetaan roolien, roolisuhteiden ja roolitoimijoiden määrittelyllä eli hahmotetaan, ketä kaikkia tässä suppeassa asiakirjatyypissä on mukana.

Sosiaalihuollon asiakasasiakirjoissa esiintyy paljon *rooleja*. Saman asiakirjan kontekstissa yksi henkilö tai organisaatio voi hoitaa useita eri tehtäviä, joihin viitataan termillä rooli. Esimerkiksi päivähoitohakemuksessa lapsen huoltaja (henkilön ensimmäinen rooli) saa automaattisesti roolin asiakas (toinen rooli). Lisäksi hän voi saada roolin hakemuksen laatija (kolmas rooli), jos hän on täyttänyt hakemuksen itse. Vaikka henkilö esiintyy asiakirjassa kolmessa eri roolissa,

hänen henkilötietojaan ei ole syytä tulostaa kolmeen kertaan. Näyttömuoto-ohjeissa kuvataan se tapa, millä henkilötietoja tulostetaan.

Roolien lisäksi asiakirjoissa esiintyy *roolisuhteita*. Roolisuhde on tietokomponentti, jolla kuvataan usean roolin vuorovaikutusta. Esimerkiksi huoltajan ja huollettavan suhteena on huoltajuus. Useat roolit ja suhteet sisältyvät automaattisesti jokaiseen asiakirjaan, koska ne liittyvät asiakkaan perustietoihin (Kortelainen 2006).

Roolit ja roolisuhteet on jaettu eri kategorioihin:

- *Kaikissa asiakirjatyypeissä olevat roolit*. Roolit, jotka sisältyvät asiakkaan perustietoihin (Kortelainen 2006) tai palvelunantajan perustietoihin (Kortelainen 2007) esimerkiksi asiakas, edunvalvoja ja huoltaja.
- *Kaikissa asiakirjatyypeissä olevat roolisuhteet*. Roolisuhteet, jotka sisältyvät asiakkaan perustietoihin (Kortelainen 2006) tai palvelunantajan perustietoihin (Kortelainen 2007) esimerkiksi huoltajuus ja edunvalvonta.
- *Yleisen asiakirjatyypin mukaiset roolit*. Roolit, joita on kuvattu yleisten asiakirjatyypin (kuten hakemus, päätös ja sopimus) rakennemäärityksissä. Roolit määräytyvät asiakasasiakirjan yleisen tyypin mukaisesti (ks. Paakkanen 2008). Esimerkiksi, jos asiakirja kuuluu yleiseen tyyppiin Päätös, niin tähän roolikategoriaan sisältyy valmistelijä, päätöksentekijä, lisätietojen antaja, tiedoksiantaja ja tiedoksisaja (vrt. Kortelainen ja Luukkonen 2009).
- *Suppean asiakirjatyypin mukaiset roolit*. Roolit, jotka tulevat suppean asiakirjatyypin (Paakkanen ym. 2008) määrityksistä. Nämä roolit ovat asiakirjatyypikohtaisia, ja niitä tunnustetaan palvelutehtäväkohtaisista sisältömäärityksistä. Esimerkiksi toimeentulotukihakemuksessa suppean asiakirjatyypin mukaisina rooleina ovat hakija ja lapsi. Huomioitavaa on, että hakemuksen laatija (yleisen asiakirjatyypin Hakemus mukainen rooli) ei ole sama rooli kuin hakija, koska hakemuksen laatijana voi olla esimerkiksi sosiaalihuollon ammatillinen henkilö, joka laatii hakemuksen hakijan puolesta.
- *Suppean asiakirjatyypin mukaiset suhteet*. Roolisuhteet, jotka tulevat suppean asiakirjatyypin määrityksistä. Nämä roolisuhteet tunnustetaan samalla tavalla kuin suppean asiakirjatyypin mukaiset roolit. Esimerkiksi toimeentulotukihakemuksessa asiakirjatyypikohtaisina suhteina ovat lapsivanhempi-suhde ja parisuhde. Näihin roolisuhteisiin sisältyvät tiedot on määritelty palvelutehtäväkohtaisissa tietomäärityksissä (kuten Kärki 2007). Roolisuhteisiin osallistuvat roolit on mainittava taulukon alkuosassa olevassa rooliluettelossa.

Roolien kuvauksen pääosana on *roolitoimija*. Toimijana voi olla yksityishenkilö, ammatillinen henkilö tai organisaatio. Roolitoimijana ei voi olla muu rooli. Sisällöllisissä asiakirjarakenteissa ei oteta kantaa roolien välisiin riippuvuuksiin muutoin kuin roolisuhteilla, eikä muodosteta roolihierarkioita. Rakenteissa ei määritellä sääntöjä kuten "asiakkaina on hakija, hakijan puoliso ja heidän alaikäiset lapsensa". Perusteluna on roolihierarkioiden, rooliriippuvuuksien ja asiakkuuden määritelmän monimutkaisuus sosiaalihuollossa. Asiakirjarakenteet luodaan niin, että ne ovat mahdollisimman joustavia.

Sisällölliset asiakirjarakenteet perustuvat erikseen määriteltyihin tietokomponentteihin; kaikki roolit ja roolisuhteet perustuvat tietokomponentteihin ja asiakirjojen yleinen rakenne kuvataan

tietokomponentteina. Sisällöllisten asiakirjarakenteiden kokoamisessa tunnistetaan jo määrittelyistä tietokomponenteista kussakin asiakirjassa tarvittavat tietokomponentit ja tarvittavista tietokomponenteista tunnistetaan ja nimitetään tarvittavat kentät.

Määrittelytyössä voidaan myös joutua tarkentamaan jo olemassa olevia tietokomponentteja. Tätä on kuvattu luvussa 5.1. Sisällöllisiä asiakirjarakenteita määriteltäessä syntyy myös uusia tietokomponentteja. Nämä voivat olla yhden palvelutehtävän tarvitsemia komponentteja tai jopa sellaisia, joita tarvitaan vain yhdessä suppean asiakirjatyypin mukaisessa asiakirjassa. Näin tietokomponentteja validoidaan samalla kun sisällöllisiä asiakirjarakenteita määritellään ja toisinpäin; kun tietokomponenttien sisällöt muuttuvat, asiakirjat päivitetään uusia komponentteja vastaaviksi.

Tietokomponentit ovat oleellisessa roolissa, kun määrittelytyössä pyritään tietosisältöjen yhdenmukaistamiseen, mutta tietokomponenttien mahdollisimman laaja käyttö ei ole itsetarkoituksellista. Kaikki sisällöllisten asiakirjarakenteiden tietosisällöt eivät ole esitettävissä tietokomponentteina. Nämä tiedot saattavat olla semanttisesti lähellä jotakin määriteltyä tietokomponenttia, mutta kyseisessä asiakirjassa vain yksittäisenä kenttänä esiintyvänä, jolloin kokonaisen tietokomponentin käyttö ei ole tarpeen.

Sisällölliset asiakirjarakenteet on kuvattu taulukkomuodossa siten, että ne kuvaavat suppeaa näyttömuotoa eli asiakirjan tulostusmuotoa. Kaikki tietyn suppean asiakirjatyypin käyttämät tietokomponentit ovat periaatteessa kokonaisina komponentteina kaikkine kenttineen tämän asiakirjan käytössä. Kaikki kentät eivät kuitenkaan ole aina välttämättömiä, joten niiden tulostaminenkaan ei ole silloin tarpeen. Esimerkiksi tietokomponentissa *maahanmuutto* on yhteensä kahdeksan kenttää, joista toimeentulotukihakemus-asiakirjassa tarvitaan vain kolmea kenttää (oleskeluoikeus, oleskeluluvan voimassaolo ja työluvan voimassaolo). Sisällöllisten asiakirjarakenteiden esityksessä otetaan kantaa ja määritellään roolien, roolisuhteiden ja kaikkien muiden tarvittavien tietokomponenttien tai yksittäisten kenttien osalta niiden näkyvyys eli tulostuminen, ja tätä kuvausta tarkennetaan tarvittaessa näyttömuotoa koskevalla ohjeituksella.

## 6.4 CCTS ja asiakirjataulukot

---

Sisällölliset asiakirjarakenteet kertovat asiakirjan sisältämistä tiedoista tietokomponenttien avulla. Kokonaiset asiakasasiakirjat koostuvat lukuisista viittauksista, koska kukin tieto on määritelty vain yhdessä kaikille asiakasasiakirjoille yhteisessä paikassa. Asiakirjassa voi olla myös asiakirjakohtaisia kenttiä. Vaikka sisällölliset asiakirjarakenteet ovat jo paljon lähempänä teknistä maailmaa sisältömäärityksiin ja rakennemäärityksiin verrattuna, varsinkin tekninen rakenne voidaan muodostaa vain silloin, kun asiakirja on mallinnettu CCTS-mallin mukaisesti. Mallinnus suoritetaan *skeemataulukoiden* avulla. Skeemataulukot noudattavat samaa rakennetta kuin UN/CEFACT-järjestön tekemät Core Components Library (CCL) -kirjastossa (UN/CEFACT 2010) olevat taulukot. Rakennetta ei käydä tässä dokumentissa tarkemmin läpi, koska se on hyvin samantyyppinen kuin tietokomponenttitaulukoiden rakenne. Taulukossa 5 on esimerkki asiakasasiakirjan skeemataulukosta.

Taulukko 5. Esimerkki CCTS-muotoisesta toimeentulotukipäätöksen määrittämisestä. Osa alkupe-  
räisen taulukon sarakkeista on piilotettu.

ABIE/ BBIE/ ASBIE/ ACC/B CC/	Dictionary Entry Name (auto generated)	Definition Mandatory	Object Class Term	Property Term	Repre- sentat- ion Term	Qualified Data Type UID	Associated Object Class	Occur- rence Min	Occur- rence Max	XML name
ABIE	Toimeentulotukipaatos. Sisalto		Toimeentulo tukipaatos							Toimeentul- otukipaatos
ASBIE	Toimeentulotukipaatos. Header		Toimeentulo tukipaatos	Header		sos:Header	Header	1	1	Header
ASBIE	Toimeentulotukipaatos. Huoltajuussuhde		Toimeentulo tukipaatos	Huoltajuuss- uhde		sos:Huoltajuus- suhde	Huoltajuuss- uhde	0	unbou- nded	Huoltajuuss- uhde
ASBIE	Toimeentulotukipaatos. Edunvalvontavaltuutussu- hde		Toimeentulo tukipaatos	Edunvalvont- avaltuutussu- hde		sos:Edunvalvo- ntavaltuutussu- hde	Edunvalvont- avaltuutussu- hde	0	unbou- nded	Edunvalvont- avaltuutussu- uhde
ASBIE	Toimeentulotukipaatos. Edunvalvontasuhde		Toimeentulo tukipaatos	Edunvalvont- asuhde		sos:Edunvalvo- ntasuhde	Edunvalvont- asuhde	0	unbou- nded	Edunvalvont- asuhde
ASBIE	Toimeentulotukipaatos. Asiakirja		Toimeentulo tukipaatos	Asiakirja		sos:Asiakirja	Asiakirja	1	1	Asiakirja
ASBIE	Toimeentulotukipaatos. Yksityishenkilö		Toimeentulo tukipaatos	Yksityishen- kilo		sos:Yksityishe- nkilo	Yksityishen- kilo	1	unbou- nded	Yksityishen- kilo
ASBIE	Toimeentulotukipaatos. Organisaatio		Toimeentulo tukipaatos	Organisaatio		sos:Organisaa- tio	Organisaatio	1	unbou- nded	Organisaati- o
ASBIE	Toimeentulotukipaatos. Osoite		Toimeentulo tukipaatos	Osoite		sos:Osoite	Osoite	1	unbou- nded	Osoite
ASBIE	Toimeentulotukipaatos. Paatos		Toimeentulo tukipaatos	Paatos		sos:Paatos	Paatos	1	1	Paatos
BBIE	Toimeentulotukipaatos. HakemusKuvaus. Teksti	Yhteenveto hakemuksen sisällöstä	Toimeentulo tukipaatos	HakemusKu- vaus	Teksti			0	1	HakemusK- uvaus Teksti
ASBIE	Toimeentulotukipaatos. Perusosa		Toimeentulo tukipaatos	Perusosa		tat:Perusosa	Perusosa	0	1	Perusosa
BBIE	Toimeentulotukipaatos. Toimeenpano. Koodi	etuuden suoritustapa	Toimeentulo tukipaatos	Toimeenpan- oKoodi	Koodi			0	unbou- nded	Toimeenpa- noKoodi
BBIE	Toimeentulotukipaatos. ToimeenpanoSopimus. Teksti	kuvaus päätöksentekij- än ja hakijan välisestä, etuuden toimeenpanoo- n liittyvästä sopimuksesta	Toimeentulo tukipaatos	Toimeenpan- oSopimus	Teksti			0	1	Toimeenpa- noSopimus Teksti
ASBIE	Toimeentulotukipaatos. Perinta		Toimeentulo tukipaatos	Perinta		tat:Perinta	Perinta	0	unbou- nded	Perinta
ASBIE	Toimeentulotukipaatos. AmmatillinenHenkilo		Toimeentulo tukipaatos	Ammatilline- nHenkilo		sos:Ammatillin- enHenkilo	Ammatilline- nHenkilo	1	unbou- nded	Ammatilline- nHenkilo

Skeemataulukko määrittelee objektiluokan, jonka nimi on sama kuin määriteltävän asiakirjan nimi. Taulukko pohjautuu sisällöllisten asiakirjarakennetaulukoiden ensimmäiseen sarakkeeseen, joka sisältää viittaukset tietokomponenttikirjastossa oleviin luokkiin. Nämä viittaukset muodostavat ASBIE-komponentteja eli objektiluokan ominaisuuksia.

Kaikista sisällöllisissä asiakirjarakenteissa määrittelyissä mainituista rooleista ei muodosteta ASBIE-komponentteja. Jos roolin kuvaukseen ei sisälly mitään roolikohtaista tietoa, vaan siinä on ainoastaan viittaus roolitoimijaan, rooli saadaan mallinnettua muilla menetelmillä. Roolikoodi voidaan esittää esimerkiksi suoraan roolitoimijaobjektin ominaisuutena. Jos rooli sisältyy johonkin muuhun tietokomponenttiin, jota käytetään asiakirjan rakenteessa, ei roolia tarvitse lisätä objektiluokkaan erikseen. Esimerkkinä on tilanne, jossa rooli osallistuu johonkin asiakirjassa ilmestyvään roolisuhteeseen. Tässä tapauksessa riittää, että määritellään roolisuhte asiakirjaobjektiluokan ominaisuudeksi. Esimerkiksi roolit *Huoltaja* ja *Huollettava* ovat roolisuhteen *Huoltajuussuhde* ominaisuuksia ja tulevat asiakirjaa kuvaavaan objektiluokkaan sen kautta. Joihinkin asiakirjaobjektiluokkiin ei tule yhtään roolia kuvaavaa ominaisuutta, koska yleisimmät roolit osallistuvat roolisuhteisiin tai ovat muiden tietokomponenttien ominaisuuksia.

Asiakirjaobjektiluokkaan lisätään aina myös kaikki roolitoimijat (tällä hetkellä Yksityishenkilö, Ammatillinen henkilö ja Organisaatio) ja tietokomponentti Osoite. Lisäys tapahtuu tällä tasolla, koska näin pystytään muodostamaan yksinkertaisempia viittauksia roolitoimijoihin ja osoitteisiin asiakirjainstansseissa. Lisätietoa viittausten toteutuksesta löytyy lähteestä (Huttunen ym. 2009, luku 6.3).

Asiakasasiakirjakohtaiset kentät muodostavat BBIE-komponentteja. Ne ovat asiakirjaobjektiluokan ominaisuuksia, jotka perustuvat yksinkertaisiin tietotyyppihin kuten Teksti, Koodi ja Numero. Tyyppit on määritelty suosituksessa JHS 170 (JUHTA 2009).

## 6.5 Tekniset asiakirjarakenteet eli skeemat

---

Varsinaiset asiakirjamääritykset ovat usein hyvin lyhyitä ja tiiviitä kokonaisuuksia johtuen siitä, että tiedot koostetaan valmiista tietokomponenteista. XML Schema -kielellä määritellyt asiakirjarakenteet sisältävät pääosin viittauksia tietokomponenttikirjastoon. Myös tietokomponenttikirjasto on esitetty XML-skeemojen avulla.

Jokainen tietokomponentti muodostaa XML-tyypin ja siihen perustuvan XML-elementin. Elementtien ja tyyppien nimeämiseen sovelletaan dokumentissa (Huttunen ym. 2009) määriteltyjä sääntöjä. Esimerkiksi taulukossa 4 esitetty tietokomponentti Tavoite määritellään XML Schema -kielellä näin:

```
<xs:element name="Tavoite" type="sos:TavoiteTyyppi"/>
<xs:complexType name="TavoiteTyyppi">
  <xs:sequence>
    <xs:element ref="jhs:KuvausTeksti" minOccurs="0">
      <xs:annotation>
        <xs:documentation>Konkreettinen asia, joka kuvataan mahdollisimman arkikielisesti ja yksityishohtaisesti</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element ref="sos:AsettaminenPvm" minOccurs="0">
      <xs:annotation>
        <xs:documentation>Päivä, jolloin tavoite on asetettu</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element ref="sos:AikatauluTeksti" minOccurs="0">
      <xs:annotation>
        <xs:documentation>Tavoitteen suunniteltu toteutumisen aikaväli. Esimerkki: ~3-6kk tai ~1v</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element ref="sos:SeurantaPvm" minOccurs="0">
      <xs:annotation>
        <xs:documentation>Päivä, jolloin tavoitteen toteutumista arvioidaan</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element ref="sos:MenetelmaTeksti" minOccurs="0">
      <xs:annotation>
```

```

        <xs:documentation>Menetelmät/välineet, joilla pyritään saavuttamaan tavoitetta</xs:documentation>
    </xs:annotation>
</xs:element>
<xs:element ref="sos:PrioriteettiNumero" minOccurs="0">
    <xs:annotation>
        <xs:documentation>Tavoitteen prioriteetti, 1=korkein prioriteetti (tärkein tavoite). Jos asiakirjassa tai komponentissa käytetään useita Tavoite-komponentteja, prioriteeteilla voi ilmaista tärkeysjärjestyksen</xs:documentation>
    </xs:annotation>
</xs:element>
<xs:element ref="sos:Vastuuhenkilö" minOccurs="0" maxOccurs="unbounded">
    <xs:annotation>
        <xs:documentation>Henkilö, joka on vastuussa työskentelystä tavoitteen saavuttamiseksi</xs:documentation>
    </xs:annotation>
</xs:element>
</xs:sequence>
</xs:complexType>

```

Asiakirjarakennetta kuvaavasta skeemataulukosta on hyvin lyhyt matka varsinaiseen XML-skeemaan. Koska asiakirjarakenteet perustuvat tietokomponentteihin, asiakirjaskeemat ottavat tietokomponenteista muodostetut skeemat käyttöön `import`-lauseella. Skeemataulukossa määritelty ABIE-objektiluokka muodostaa XML-tietotyyppin ja siihen perustuvan XML-elementin. Asiakirjarakenteita kuvaavat skeemat noudattavat siis samoja periaatteita kuin tietokomponentteja kuvaavat skeemat. Roolit kuvataan avainten ja avainviitteiden avulla (Huttunen ym. 2009, luku 3.4). Esimerkiksi taulukon 5 pohjalta muodostettu toimeentulotukipäätöksen XML-skeeman pääsisältö on tällainen:

```

<xs:element name="Toimeentulotukipaatos" type="tathakemus:ToimeentulotukipaatosTyyppi">
    <xs:unique name="EdunvalvojaAvain">
        <xs:selector xpath="/sos:Yksityishenkilö"/>
        <xs:field xpath="@r:id"/>
    </xs:unique>
    ...
    <xs:keyref name="EdunvalvojaAvainViite" refer="tathakemus:EdunvalvojaAvain">
        <xs:selector xpath="//sos:Edunvalvoja/sos:YksityishenkilöRooliViite"/>
        <xs:field xpath="@r:ref"/>
    </xs:keyref>
</xs:element>
<xs:complexType name="ToimeentulotukipaatosTyyppi">
    <xs:sequence>
        <xs:element ref="sos:Header"/>
        <xs:element ref="sos:Huoltajuussuhde" minOccurs="0" maxOccurs="unbounded"/>
        <xs:element ref="sos:Edunvalvontavaltuutusuhde" minOccurs="0" maxOccurs="unbounded"/>
        <xs:element ref="sos:Edunvalvontasuhte" minOccurs="0" maxOccurs="unbounded"/>
        <xs:element ref="sos:Asiakirja"/>
        <xs:element ref="sos:Yksityishenkilö" maxOccurs="unbounded"/>
        <xs:element ref="sos:Organisaatio" maxOccurs="unbounded"/>
        <xs:element ref="sos:Osoite" maxOccurs="unbounded"/>
        <xs:element ref="sos:Paatos"/>
        <xs:element ref="tathakemus:HakemusKuvausTeksti" minOccurs="0"/>
        <xs:element ref="tat:Perusosa" minOccurs="0"/>
    </xs:sequence>
</xs:complexType>

```

```

<xs:element ref="tathakemus:ToimeenpanoKoodi" minOccurs="0"
maxOccurs="unbounded"/>
<xs:element ref="tathakemus:ToimeenpanoSopimusTeksti" minOccurs="0"/>
<xs:element ref="tat:Perinta" minOccurs="0" maxOccurs="unbounded"/>
<xs:element ref="sos:AmmatillinenHenkilo" maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>
<!-- BBIE-komponentit -->
<xs:element name="HakemusKuvausTeksti" type="r:TekstiTyyppi"/>
<xs:element name="ToimeenpanoKoodi" type="r:KoodiTyyppi"/>
<xs:element name="ToimeenpanoSopimusTeksti" type="r:TekstiTyyppi"/>

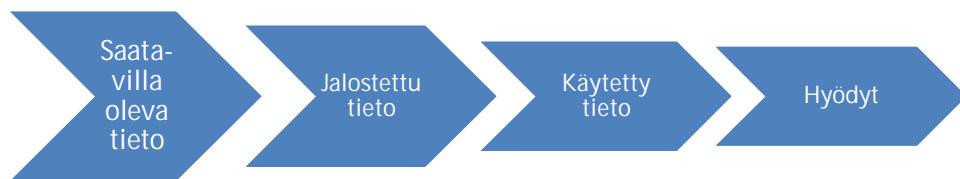
```

XML-komponenttikirjastot ja asiakirjarakenteet muodostavat tietomallin eräänlaisen esitystavan. Vaikka XML-skeemat sisältävätkin elementtien määritelmiä ja esimerkkejä, tekniset asiantuntijat tarvitsevat tarkempia soveltamisoppaita rakenteiden hyödyntämiseen. Soveltamisoppaina toimivat sisällölliset asiakirjarakenteet ja tietokomponenttitaulukot.

## 7 Yhteenveto

Attribuutti, entiteetti, instanssi, kaavio, käsite, luokka, malli, objekti, olio, ontologia, ominaisuus, termi, jne. ovat tietomallinnukseen liittyviä asioita, joita tarkasteltiin luvuissa 2 ja 3. Luvussa 4 tarkasteltiin tietomallinnusta Tikesos-hankkeessa. Rakenteilla oleva tietomalli soveltuu asiakirjarakenteiden määrittelyyn. Tietomallia ei kuitenkaan voida käyttää sellaisenaan tietokantojen rakentamiseen, vaan se vaatii ensin normalisointia. Myös perusoperaatiot (lue, luo, päivitä, poista) tulee määritellä ja varmistaa perusoperaatioiden avulla tavoitetilan prosessimallien ja tietomallin yhteensopivuus. Normalisointi ja perusoperaatioiden määrittely on toistaiseksi jätetty tekemättä, mutta se on viimeistään siinä vaiheessa välttämätöntä, kun käyttötapaukset saadaan kuvattua tavoitetilaan liittyen.

Tällä hetkellä Tikesos-hankkeessa jalostettu sosiaalihuollon palvelutehtävistä saatavilla olevaa tietoa ja siitä on muodostumassa sosiaalihuollon asiakastietomalli. Kuvassa 33 havainnollistetaan sitä, että hyötyjä (gains) saavutetaan tiedoilla, joita käytetään (mukaihen Ritvanen 2009).



*Kuva 33. Tiedon jalostaminen ja käyttö*

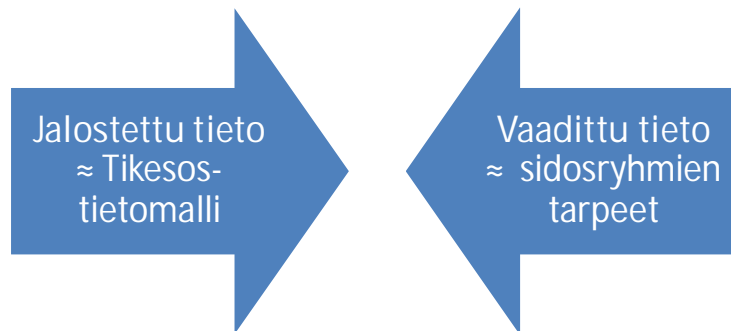
Tulevaisuudessa pitää tarkastella sosiaalihuollon asiakastietomallia siitä näkökulmasta, että mitä päätöksiä pitää tehdä, jotta tietyt tavoitteet (goals) saavutetaan. Päätöksien tekemiseksi tarvitaan tehdä tietoihin perustuvia analyysejä ja raportteja, joten saatavilla olevan, jalostetun ja käytetyn tiedon sijaan puhutaan vaaditusta tiedosta. (mukaihen Ritvanen 2009)



*Kuva 34. Sosiaalihuollossa tehdään päätöksiä, jotka vaativat tietoa jopa useista eri tietolähteistä.*

Vuoden 2010 aikana on tarkoitus aloittaa sosiaalihuollon päätöksenteossa tarvittavien tietojen kerääminen ja analysointi. Erityisesti toimeentulotuen ja sosiaalisen luototuksen päätökset ovat tarkastelun kohteena. Kun päätöksentekoon tarvittavia tietoja määritellään, niin samalla määritellään myös (liike)toimintasääntöjä, joiden pohjalta voitaneen osittain automatisoida esimerkiksi toimeentulotukipäätösten tekemistä, varsinkin, jos Kelan tietojärjestelmistä saadaan etuustiedot sähköisesti kuntien asiakastietojärjestelmien käyttöön.

Tikesos-hankkeen määrittelyjen arviointi tulee tehdä eri sidosryhmien edustajien toimesta, jotta saavutetaan halutut tavoitteet ja hyödyt.



*Kuva 35. Asiakasasiakirjojen tietojen perusteella muodostettu tietomalli tulee peilata eri sidosryhmien tietotarpeisiin ja varmistaa, että tavoitetilan mukaiset tietotarpeet on mukana sosiaalihuollon asiakastietomallissa*

Luvussa 5 esitettiin sosiaalihuollon asiakastietomallinnus ja -malli. Sosiaalihuollon asiakastietomallia voidaan käyttää esimerkiksi seuraavasti:

- verrataan sosiaalihuollon asiakastietomallia esimerkiksi kuntien asiakastietojärjestelmien tietomalleihin ja kehitetään sosiaalihuollon palvelutehtäviä tukevia sovelluksia sosiaalihuollon asiakastietomallin pohjalta
- määritellään ja suunnitellaan uusia sovelluksia, kuten kansallisia sähköisiä palveluja kansalaisille, sosiaalihuollon palvelutehtäviin hyödyntäen sosiaalihuollon asiakastietomallia
- määritellään ja suunnitellaan rajapintoja, kuten Kelan SOKY-rajapinta, hyödyntäen sosiaalihuollon asiakastietomallia

Sosiaalihuollon asiakasasiakirjojen mallinnus ja mallit esitettiin luvussa 6. Sosiaalihuollon asiakasasiakirjamalleja voidaan käyttää esimerkiksi seuraavasti:

- verrataan malleja kuntien asiakastietojärjestelmien tuotoksiin ja kehitetään asiakastietojärjestelmiä siten, että niiden tuotokset vastaavat sosiaalihuollon asiakasasiakirjamalleja
- muodostetaan mallin mukaisia asiakirjoja ja hyödynnetään niitä sosiaalihuollon prosesseissa.
- käytetään asiakirjamalleja lomakkeiden määrittelyyn ja kehittämiseen.

Sisällöllisten asiakirjarakenteiden kehittämistyö on jatkuvaa toimintaa. Sisällön asiantuntijoiden eri vaiheissa antama palaute ja kommentit asiakirjarakenteista ovat kehittämistyön etenemisessä välttämättömiä. Asiakirja-työryhmä vie määritellyjä asiakirjarakenteita projektipäälliköiden käsittelyyn. Asiakirjarakennetaulukot on julkaistu keskeneräisenä myös Tikesos-hankkeen sivustolla. Sisällöllinen asiakirjarakenne on valmis, kun siihen on saatu kommentteja sidosryhmältä, rakenne on muokattu kommenttikierroksen jälkeen ja hyväksytty hankkeen johtoryhmässä.

## LÄHTEET

---

- AdoL 1985. Laki lapseksiottamisesta 8.2.1985/153.
- AdoA 1987. Asetus lapseksiottamisesta 30.5.1987/508.
- AsL 2000. Laki sosiaalihuollon asiakkaan asemasta ja oikeuksista 22.9.2000/812.
- AmA 1992. Asetus sosiaali- ja terveydenhuollon asiakasmaksuista 9.10.1992/912.
- AmL 1992. Lakisosiaali- ja terveydenhuollon asiakasmaksuista 3.8.1992/734.
- Antoniou G., van Harmelen F. 2004. *A Semantic Web Primer*, Massachusetts Institute of Technology, The MIT Press 2004. ISBN 0-262-01210-3.
- Glushko R. J., McGrath T. 2002. *Document Engineering for e-Business*, Doc Eng '02, November 2002.
- Haikala I., Märijärvi J. 2006. *Ohjelmistotuotanto*, Talentum, Helsinki 2006.
- Hay D. C. 1999. *A Comparison of Data Modeling Techniques*, Essential Strategies Inc.
- HE 137/1999. Hallituksen esitys Eduskunnalle laeiksi sosiaalihuollon asiakkaan asemasta ja oikeuksista sekä sosiaalihuoltolain muuttamisesta ja eräksi niihin liittyviksi laeiksi.
- HetiL 1999. Henkilötietolaki 22.4.1999/523.
- Hovi A., Huotari J., Lahdenmäki T. 2003. *Tietokantojen suunnittelu & indeksointi*, Docendo Finland Oy, Porvoo, 2003.
- Huttunen, R., Hyppönen, K., Martikainen, I., Hotti, V., Nevalainen, J. 2009. *Sosiaalihuollon asiakasasiakirjojen tekniset määrittelyt*. Sosiaalialan tietoteknologiahanke, 9.6.2009. <http://www.sosiaaliportti.fi/File/c9457d06-651d-4776-ae70-e1107c6f0bd2/Sosiaalihuollon+asiakasasiakirjojen+tekniset+m%C3%A4%C3%A4ritykset.pdf> (28.8.2009)
- Hämäläinen P., Koskue S., Lehtonen J., Ojala M., Palojoki S. 2008. *Koodistopalvelun käsikirja*, Versio 0.2, 14.3.2008, Stakes. <http://sty.stakes.fi/NR/rdonlyres/C0432B2A-7AB6-4D4E-AFE2-3927962AE423/12271/Koodistopalvelun%C3%A4sikirja.pdf>
- JUHTA - Julkisen hallinnon tietohallinnon neuvottelukunta. 2008. *JHS 152 Prosessien kuvaaminen*. <http://www.jhs-suositukset.fi/suomi/jhs152>

- JUHTA - Julkisen hallinnon tietohallinnon neuvottelukunta. 2009. *JHS 170 Julkishallinnon XML-skeemat*. <http://www.jhs-suositukset.fi/web/guest/jhs/recommendations/170>
- Kassian Ladislav. 2006. From Now to Than – Model-based Change. Manuskript (unveröffentlicht): *Unfolding Intrinsic Complexity*, 2006.
- Kortelainen P. 2006. *Asiakkaan perustiedot sosiaalihuollon asiakastietojärjestelmissä: Tietokuvaukset, standardit, luokitukset ja suositukset*, Versio 1.0, 14.12.2006, Sosiaalialan tietoteknologiahanke. <http://www.sosiaaliportti.fi/File/66621f83-bc02-49ec-80f3-814a5d0ae610/Perustiedot+1.0.pdf> (23.4.2009)
- Kortelainen P. 2007. *Palvelun antajan perustiedot sosiaalihuollon asiakastietojärjestelmissä: Tietokuvaukset, standardit, luokitukset ja suositukset*, Versio 1.0, 31.7.2007, Sosiaalialan tietoteknologiahanke. <http://www.sosiaaliportti.fi/File/b651ff27-eb61-485f-b7a2-ecd1c276caf9/Palvelun+antajan+perustiedot+1.0.pdf> (23.4.2009)
- Kortelainen P. 2008. *Sosiaalihuollon asiakasasiakirjojen yleinen kansallinen rakenne*. Sosiaalialan tietoteknologiahanke, 2008. <http://www.sosiaaliportti.fi/File/1491ca00-6ff8-49f3-aedf-57c1c23d1c09/Asiakasasiakirjojen+yleinen+rakenne.pdf> (23.4.2009)
- Kortelainen P., Luukkonen T. 2009. *Yleiset asiakirjatyyppit*. Sosiaalialan tietoteknologiahanke, 2009. <http://www.sosiaaliportti.fi/Page/768a8cf6-1f62-44de-a9cf-98613d5673ad.aspx> (23.4.2009)
- Kärki J. 2007. *Toimeentulotuen asiakastiedot sosiaalihuollon asiakastietojärjestelmässä: Tietokuvaukset ja suositukset*, Versio 1.0, 27.6.2007, Sosiaalialan tietoteknologiahanke.
- Lehmuskoski A., Kuusisto-Niemi S. 2007. *Sosiaalialan sanaston työstäminen lausuntojen pohjalta*. Hankeraportti, Sosiaalialan tietoteknologiahanke, Terveystieteiden ja -talouden laitos, Kuopion yliopisto, 2007.
- Lehmuskoski A., Kuusisto-Niemi S. 2008. *Sosiaalialan sanasto asiakastietojärjestelmää varten*. Stakes [nykyisin THL], raportteja 30/2008. <http://www.sosiaaliportti.fi/File/9150c01f-cce2-4a56-b4ee-d529e77d3660/Sanasto.pdf> (17.4.2009)
- LhtL 1983. Laki lapsen huollosta ja tapaamisoikeudesta 8.4.1983/361.
- LkhL 1996. Laki lasten kotihoidon ja yksityisen hoidon tuesta 20.12.1996/1128.
- Nevalainen J., Martikainen I., Miettinen A., Hotti V., Huttunen R., Savolainen J. 2009. *Sisällön yhtenäistäminen sosiaalihuollon teknisessä asiakirjastandardoinnissa*. Sosiaalialan tietoteknologiahanke, 2009. <http://www.sosiaaliportti.fi/File/20d5041c-e8e4-4b3b-9ffd-0286505384a1/Sisällön+yhtenäistäminen+sosiaalihuollon+tekisessä+asiakirjastandardoinnissa.pdf>
- OMG – Object Management Group. 2009. Documents Associated with UML version 2.3 Beta 2. <http://www.omg.org/spec/UML/2.3/>

- Paakkanen E., Laaksonen M., Kortelainen P., Mykkänen J., Suhonen M. 2009. *Sosiaalihuollon asiakasasiakirjojen metatiedot*, Versio 1.0, 9.6.2009, Sosiaalialan tietoteknologiahanke.
- Paakkanen E., Silvennoinen R., Suhonen M., Luukkonen T., Nuotio T., Kaija M. 2008. *Sosiaalihuollon asiakirjatyypiluokitus*. Sosiaalialan tietoteknologiahanke, 2008.  
<http://www.sosiaaliportti.fi/File/c08453a4-8b61-4f75-8b13-cb360b459b37/Asiakirjatyypiluokitus.pdf> (17.4.2009)
- Paakkanen E., Suhonen M., Silvennoinen R., Luukkonen T., Savolainen J., Martikainen I., Nevalainen J., Mykkänen J., Kortelainen P. 2008. *Sosiaalihuollon asiakirjatyypiluokittelu ja asiakirjatyypimäärittelyt*. Sosiaalialan tietoteknologiahanke, 2008.  
<http://www.sosiaaliportti.fi/File/0f753d1f-dfe4-4b4e-a641-b703c4a11796/Sosiaalihuollon+asiakirjatyypiluokittelu+ja+asiakirjatyypim%C3%A4%C3%A4rittelyt.pdf> (2.4.2009)
- Ponniiah P. 2007. *Data Modeling Fundamentals: A practical guide for IT professionals*, John Wiley & Sons Inc 2007. ISBN 0-471-79049-4.
- Ritvanen H. 2009. *Data Warehousing and Business Intelligence in Enterprises*. Esitys TKK 14.10.2009. [https://noppa.tkk.fi/noppa/kurssi/t-86.5141/luennot/T-86\\_5141\\_ritvanen\\_bi.pdf](https://noppa.tkk.fi/noppa/kurssi/t-86.5141/luennot/T-86_5141_ritvanen_bi.pdf) (30.4.2010)
- SHL 1982. Sosiaalihuoltolaki 17.9.1982/710.
- Simsion G. C., Witt G. C. 2005. *Data modeling essentials*. Third Edition, Elsevier 2005.
- STM 2001. Sosiaalihuollon asiakkaan asema ja oikeudet. STM:n oppaita 2001:11.
- Studer R., Benjamins V. R., Fensel D. 1998. *Knowledge Engineering: Principles and Methods*, IEEE Transactions on Data and Knowledge Engineering, 25, 1998, pp. 161–197.
- Sugumaran V., Storey V. C. 2006. *The Role of Domain Ontologies in Database Design: An Ontology Management and Conceptual Modeling Environment*. ACM Transactions on Database Systems, Vol. 31, No 3, September 2006, Pages 1064–1094.
- Suonuuti H. 1997. *Guide to terminology*. Nordterm 8. Tekniikan Sanastokeskus, Helsinki, 1997.
- Suonuuti H. 2006. *Sanastotyön opas*. Sanastokeskus TSK ry, Laser-Paino Oy, Helsinki 2006.
- Tietotekniikan liitto. 2008. *ATK-sanakirja*. Talentum 2008.
- TtL 1997. Laki toimeentulotuesta 30.12.1997/1412.
- UN/CEFACT – United Nations, Centre for Trade Facilitation and Electronic Business. 2006. *XML Naming and Design Rules. Version 2.0*. 17. helmikuuta 2006.  
<http://unece.org/cefact/xml/XML-Naming-and-Design-Rules-V2.0.pdf> (30.04.2010)
- UN/CEFACT – United Nations, Centre for Trade Facilitation and Electronic Business. 2009a. *Core Components Technical Specification. Version 3.0*. 29. syyskuuta 2009.  
<http://www.unece.org/cefact/codesfortrade/CCTS/CCTS-Version3.pdf> (30.04.2010)

- UN/CEFACT – United Nations, Centre for Trade Facilitation and Electronic Business. 2009b. *XML Naming and Design Rules Technical Specification. Version 3.0*. 17. joulukuuta 2009. <http://unece.org/cefact/xml/UNCEFACT+XML+NDR+V3p0.pdf> (30.04.2010)
- UN/CEFACT (United Nations, Centre for Trade Facilitation and Electronic Business). 2010. *Core Component Library (UN/CCL), version 09B*. [http://unece.org/cefact/codesfortrade/unccl/CCL\\_index.htm](http://unece.org/cefact/codesfortrade/unccl/CCL_index.htm) (30.04.2010)
- Visual Paradigm. 2010. *Visual Paradigm for UML*. <http://www.visual-paradigm.com/product/vpuml/>
- Väinälä A. 2007. *Lasten päivähoiton asiakastiedot sosiaalihuollon asiakastietojärjestelmässä: Tietokuvaukset ja suositukset*, Versio 1.0, 20.9.2007, Sosiaalialan tietoteknologiahanke. <http://www.sosiaaliportti.fi/File/129cce49-a5fc-4854-aefd-327c9a265d07/P%C3%A4iv%C3%A4hoito+1.0.pdf>
- W3C. 2004a. *XML Schema Part 1: Structures Second Edition*. W3C Recommendation, 28.10.2004. <http://www.w3.org/TR/2004/REC-xmlschema-1-20041028/> (21.4.2009)
- W3C. 2004b *XML Schema Part 2: Datatypes Second Edition*. W3C Recommendation, 28.10.2004. <http://www.w3.org/TR/2004/REC-xmlschema-2-20041028/> (21.4.2009)
- W3C. 2008. *Extensible Markup Language (XML) 1.0 (Fifth Edition)*. W3C Recommendation, 26.11.2008. <http://www.w3.org/TR/2008/REC-xml-20081126/> (21.4.2009)

## A Tietomallien vertailu

---

Seuraavassa on listattu vertailussa käytetyt vertailukriteerit aihepiireittäin ja kuvauksineen.

### Tietomalli

- *Käyttötarkoitus.* Mikä on menetelmän alkuperäinen käyttötarkoitus?
- *Sovellettavuus.* Mihin menetelmää on alkuperäisen käyttötarkoituksen lisäksi käytetty?
- *Kuvauskieli.* Mitä kieltä tietomallin esittämiseen käytetään kussakin tietomallissa?

### Termit

- *Nimet.* Miten käsitteitä nimetään?
- *Tunnisteet ja yksilöinti.* Kuinka käsitteiden yksilöinti tapahtuu?
- *Synonymia.* Miten käsitellään synonymiaa? Miten liitetään toisiinsa sanat, jotka viittaavat samaan tarkoitteeseen? Miten käsitellään synonyymien sisältämiä vivahde-eroja ja niitä vastaavia lähikäsitteitä?
- *Polysemia.* Miten menetelmässä käsitellään sanojen monimerkityksisyyttä? Miten esitetään saman sanan eri merkitykset saman kohdealueen sisällä ja kohdealueiden välillä?
- *Termisuositukset.* Ohjaako menetelmä valitsemaan suositeltavan termin?

### Käsitteet

- *Käsitteen ala.* Määritelläänkö käsitteelle sen kattama ala? Miten esitetään se, mitkä tarkoitteet käsite kattaa?
- *Objektiluokat.* Kuinka käsitteitä luokitellaan, mihin objektiin ("yliluokkaan") käsite liittyy/kuuluu?
- *Ominaisuudet.* Käsitteiden ominaispiirteet, eli mikä erottaa käsitteet toisistaan, ja miten ominaisuudet ilmaistaan?
- *Ilmentymät/sisältö.* Millaisia tietosisältöjä käsitteillä voi olla, ja miten sisällön rajoitteita voidaan esittää?
- *Yksilö- ja yleiskäsitteet.* Mahdollistaako menetelmä yksilö- ja yleiskäsitteiden erottamisen toisistaan?
- *Määritelmät.* Mihin ja miten käsitteiden yksiselitteiset määritelmät sijoitetaan? Pystyykö käsitteiden määritelmiä tarkentamaan eri konteksteissa?

- *Huomautukset.* Mahdollistaako menetelmä lisätietojen antamisen määritelmän lisäksi?

#### Käsitteiden väliset suhteet

- *Koostumus (vahva koostumus).* Kuinka esitetään sellainen koostumussuhde, jossa osakokonaisuudet eivät voi olla olemassa itsenäisesti?
- *Aggregaatio (heikko koostumus).* Kuinka esitetään sellainen koostumussuhde, jossa osakokonaisuudet ovat olemassa myös itsenäisesti?
- *Assosiaatio.* Kuinka käsitteiden liittyminen "jollain tapaa" toisiinsa ilmaistaan?
- *Periytyminen.* Kuinka käsitteiden hierarkiassa erikoistetaan käsitteitä perimällä yläkäsitteeltä ominaisuuksia ja lisäämällä näihin käsitteen omia ominaisuuksia?
- *Roolit.* Kuinka erilaiset sosiaalihuollossa esiintyvät toimijaroolit (kuten asiakas, huoltaja, lapsi, ym.) ilmaistaan?
- *Ryhmittely.* Kuinka käsitteitä jaotellaan esimerkiksi palvelutehtävittäin tai aihepiireittäin eri tasoille?
- *Muut suhteet.* Kuinka edellä mainittuihin suhteisiin kuulumattomat muunlaiset suhteet voidaan esittää?
- *Rajoitteet.* Sisältääkö menetelmä työvälineitä käsitteiden välisten suhteiden sisältämien rajoitteiden ilmaisemiseen? Miten rajoitteet ilmaistaan?

#### Dokumentointivälineet ja työkalut

- *Graafinen esitys.* Kuinka havainnollisia kuvia tietomallista saadaan, ja kuinka helppoa kuvien generointi on? Graafiseen esitykseen käytettäviä työkaluja?
- *Muokattavuus.* Kuinka tietomallin sisällön päivittäminen, muokkaaminen, lisääminen ja poistaminen tapahtuu? Muokkaukseen käytettäviä työkaluja? Voiko mallin graafista esitystä muokata, ja saadaanko muutokset varsinaiseen tietomalliin?
- *Dokumentointi ja ylläpidettävyys.* Kuinka tietomallin muutoksia hallitaan suhteessa dokumentointiin, kuinka dokumentaatio pysyy ajan tasalla? Ylläpitoon käytettäviä työkaluja?

#### Tekninen hyödynnettävyys

- *XML-skeemojen tuottaminen.* Miten tietomallista voidaan tuottaa XML-skeemoja, onko automatisointi mahdollista, kuinka käyttökelpoisia tuotetut skeemat ovat?
- *Tietomallin muutosten välittyminen XML-skeemoihin.* Kuinka tietomalliin tehdyt muutokset siirtyvät XML-skeemoihin?
- *Asiakirjataso.* Kuinka tietomallilla esitetään asiakasasiakirjatason tietoja (esimerkiksi päivähoitohakemus)?
- *Versiointi.* Kuinka tietomallin muutoksia hallitaan suhteessa XML-skeemoihin, kuinka skeemat pysyvät ajan tasalla?
- *Linkitettävyys prosessimallien tietovirtoihin.* Kuinka tietomallia voidaan hyödyntää tietovirtojen lisäämisessä prosessimalleihin?
- *Suhde CCTS:ään.* Miten CCTS-standardi sovituu tietomalliin, rajoittaako se tietomallia ja kuinka sulavaa CCTS:n käyttö on tietomallin kanssa?

TIETOMALLIEN VERTAILUKEHYS

	UML-luokkakaavio	E-R -malli	XML Schema	Ontologia
Tietomalli				
Käyttötarkoitus	Tietojärjestelmien suunnittelussa ja kehittämisessä tiedon kuvaaminen käyttäjille ja tietokannan suunnittelijoille ymmärrettävällä tavalla. Kehitetty olio-ohjelmoinnin tarpeisiin.	Ohjelmistotuotannossa tiedon kuvaaminen abstraktilla ja käsitteellisellä tasolla. Keino suunnitella ja mallintaa tietokantoja.	XML-dokumenttien rakenteen eli komponenttien hierarkian, niiden välisten suhteiden ja rajoitteiden kuvaaminen.	Tietämyksen mallintaminen tietokoneen ymmärtämässä muodossa. Semanttinen web.
Sovellettavuus	Käsittemallinnus, tietomallinnus.	(Relaatio)tietokantojen suunnittelu, käsittemallinnus.	Tietomallinnus, asiakirjarakenteiden mallinnus, tietojen validointi, lomakkeiden ja käyttöliittymien generointi.	Asiantuntijajärjestelmät, luonnollisen kielen ymmärtäminen, semanttinen päättely ja tekoälysovellukset.
Kuvauskieli	UML 2.0 (Class Diagram)	Chenin notaatio	XML Schema 1.0	RDF, RDFS ja OWL
Termit				
Nimet	Käsitteet (luokat) yksilöidään nimen avulla.	Käsitteet (entiteetit) yksilöidään nimen avulla.	Jokaisella käsitteellä (elementillä) on yksilöllinen nimi (attribuutti name).	Jokaisella käsitteellä on yksi nimi (määriteltynä rdf:about-attribuutilla).
Tunnisteet ja yksilöinti	Luokkien nimien tulee olla yksilöllisiä. Yleensä määrätään luokalle attribuutti ("id-tunnus" kuten henkilötunnus), joka yksilöi luokan.	Käsitteeseen liittyvä(t) attribuutti/attribuutit voivat toimia käsitteen yksilöinnissä. Yksilöivät attribuutit merkitään alleviivauksena nimen yhteydessä.	Käsitteet yksilöidään nimen ja nimiavaruuden avulla (esimerkiksi nimiavaruus:nimi).	Jokaisella käsitteellä on yksilöllinen URI-tunniste. Tunniste voidaan määritellä attribuuteilla rdf:ID tai URI. Sen lisäksi nimiavaruus yksilöi käsitteitä.
Synonymia	Ei voida esittää tietomallissa, mutta voidaan liittää ohjeistukseen tai muihin oheisdokumentaatioon.	Ei voida esittää tietomallissa, mutta voidaan liittää ohjeistukseen tai muihin oheisdokumentaatioon.	On mahdollista esittää substitutioryhmien (attribuutti substitutionGroup) avulla.	Voidaan esittää usealla eri tavalla, kuten elementillä owl:equivalentClass
Polysemia	Ei voida esittää tietomallissa, käsitteet pyritään nimeämään siten, ettei polysemiaa synny.	Ei voida esittää tietomallissa, käsitteet pyritään nimeämään siten, ettei polysemiaa synny.	Ei voida esittää tietomallissa, käsitteet pyritään nimeämään siten, ettei polysemiaa synny.	Käsitteille voidaan määritellä useita määritelmiä elementillä rdfs:comment, tai käsitteet voidaan lisätä erillisinä (yksilöinti tapahtuu URI:n perusteella).
Termisuositukset	Ei voida esittää tietomallissa, valitaan tietomallissa käytettävät termit etukäteen.	Ei voida esittää tietomallissa, valitaan tietomallissa käytettävät termit etukäteen.	Ei voida esittää tietomallissa, valitaan tietomallissa käytettävät termit etukäteen.	Termisuosituksiin voidaan ottaa kantaa kommentteissa tai synonyymien yhteydessä.

	UML-luokkakaavio	E-R -malli	XML Schema	Ontologia
Käsitteet				
Käsitteen ala	Käsitteen alaa voidaan rajata luokkaan liitettävien alaluokkien avulla. Luokka kattaa kaikki sen alaluokat sekä ilmentymät.	Ei voida esittää. Käsitteen alaa voidaan enintään päätellä siihen liittyvien entiteettien avulla.	Käsitteen ala voidaan määritellä nimiavaruuden avulla; mistä nimiavaruudesta käsite tulee, sitä alaa se edustaa.	Kommentteihin ja/tai määritelmiin voidaan lisätä ohjeistusta alasta, jossa käsitettä käytetään. Käsite kattaa kaikki sen ilmentymät. Nimiavaruuksilla voidaan myös määritellä käsitteen alaa, mikäli käytössä on useiden ontologioiden nimiavaruuksia.
Objektiluokat	Kukin käsite on oma luokkansa. Luokalla voi olla ylikuokkia ja aliluokkia, jolloin luokka sijoittuu hierarkiaan.	Kukin käsite on oma entiteettinsä, jota voidaan pitää objektiluokan vastineena.	Tyyppi määrää objektiluokan. Elementin on objektiluokka.	Kukin käsite on oma luokkansa (owl:Class). Luokilla voi olla ylikuokkia, jotka ilmaistaan elementillä rdfs:subClassOf. Kullekin käsitteelle voidaan määritellä, mikäli käsite on erillinen suhteessa toiseen/toisiin käsitteisiin elementillä owl:disjointWith, eli käsitteen ilmentymät eivät voi kuulua samalla minkään toisen käsitteen ilmentymiin.
Ominaisuudet	Käsitteelle voidaan määritellä ominaisuuksia, joille voidaan sen lisäksi määrittää tietotyyppi.	Ominaisuudet esitetään erillisinä ympyröinä, jotka liittyvät käsitteeseen. Ominaisuus voi saada monia arvoja (multi-valued), mikäli ympyrän reunaviivat ovat kaksinkertaiset. Ominaisuudet voivat myös olla sellaisia, joiden arvo tulee muualta (derived) kuten vaikka lasketaan muiden attribuuttien arvoista.	Käsitteelle voidaan määritellä ominaisuuksia (xs:attribute). Ominaisuudet ovat esitettävissä muitten käsitteiden kautta hierarkkisesti.	Luokille voidaan määritellä ominaisuuksia elementillä owl:ObjectProperty

	UML-luokkakaavio	E-R -malli	XML Schema	Ontologia
Ilmentymät/sisältö	ISO-standardin mukainen ilmentymän määritelmä, eli ilmentymä on tietotyyppi. Tietotyyppi määrittelee millaisia arvoja varsinainen tieto voi saada. Varsinaista tietoa, missä luokkien ominaisuuksille annetaan varsinaisia arvoja, ei tietomallissa esitetä.	Ei määritellä. Tietokantasuunnittelussa ilmentymät tulevat vasta itse tietokannan sisältöön, johon E-R - kaavio ei ota kantaa.	Käsitteelle voidaan määritellä tietotyyppi (attribuutti type), joka rajaa ilmentymälle mahdollisen arvojoukon. Käsitteelle voidaan myös määritellä oletusarvo (attribuutti default) tai kiinteä arvo (attribuutti fixed). Elementille voidaan määritellä onko elementti pakollinen (use="required"). Elementille voidaan myös määritellä arvojen, arvojoukon, pituuden, tyhjän tilan ja tietotyypin rajoituksia elementin xs:restriction avulla.	Ilmentymät voidaan määritellä antamalla luokkien ominaisuuksille varsinaisia arvoja. Ominaisuuksien rajoitteet rajoittavat ilmentymien arvoja.
Yksilö- ja yleiskäsitteet	Ei voida esittää eroa.	Ei voida esittää eroa.	Yksilökäsitteet voi esittää elementteinä kun yleiskäsitteet ovat tyyppejä.	Yleiskäsite esitetään luokkana, yksilökäsite luokan ilmentymänä.
Määritelmät	Ei voida esittää tietomallissa, mutta voidaan liittää ohjeistukseen tai muuhun oheisdokumentaatioon.	Ei voida esittää tietomallissa, mutta voidaan liittää ohjeistukseen tai muuhun oheisdokumentaatioon.	Voidaan lisätä kuhunkin elementtiin ja tyyppiin elementillä xs:annotation ja sen alielementillä xs:documentation.	Määritelmiä voidaan lisätä kuhunkin luokkaan elementillä rdfs:comment
Huomautukset	Voidaan liittää tietomalliin muistilapunomaisesti (muistilappu-symboli joka liitetään katkoviivalla siihen liittyvään luokkaan).	Ei voida esittää tietomallissa, mutta voidaan liittää ohjeistukseen tai muuhun oheisdokumentaatioon.	Voidaan lisätä kuhunkin elementtiin ja tyyppiin elementillä xs:annotation ja sen alielementillä xs:documentation.	Huomautuksia voidaan lisätä kuhunkin luokkaan elementillä rdfs:comment
<b>Käsitteiden väliset suhteet</b>				
Koostumus (vahva koostumus)	Oma symboli vahvan koostumuksen esittämiseksi (musta salmiakki).	Osa–kokonaisuus-suhteita voidaan esittää rajaamalla suorakaiteen sisään kaikki koosteeseen liittyvät koostenteetit.	Käsite voidaan määritellä joko simpleType-tyyppiseksi, jolloin sillä ei ole lapsielementtejä, tai complexType-tyyppiseksi, jolloin elementti koostuu useista alielementeistä.	Käsitteelle voidaan luoda ominaisuus "partOf" tai "hasPart", joilla saadaan esitettyä osa–kokonaisuus-suhteita. Näillä ominaisuuksilla voi olla alaominaisuuksina "vahva koostumus" (composition) ja "heikko koostumus" (aggregation), joilla erotetaan minkä tyyppinen koostumus on kyseessä. Koostumuksen esittämiseksi voidaan myös käyttää owl:DatatypeProperty -rakennetta.
Aggregaatio (heikko koostumus)	Oma symboli heikon koostumuksen esittämiseksi (valkea salmiakki).			

	UML-luokkakaavio	E-R -malli	XML Schema	Ontologia
Assosiaatio	Assosiaatiota voidaan esittää luokkien välisillä viivoilla. Mikäli assosiaatio on vain tiettyyn suuntaan, voidaan käyttää viivan päässä suunnan osoittavaa nuolta.	Käsitteiden välille voidaan luoda assosiaatiosuhde.	On mahdollista esittää avainten ja avainviitteiden kautta (luettavuus ei ole paras mahdollinen).	Käsitteelle voidaan luoda assosiaatio-ominaisuus, joka saa arvoksi sen luokan mihin käsite assosioituu.
Periytyminen	Oma symboli periytymisen esittämiselle (valkeapäinen nuoli).	Oma isa-suhdemerkintä periytymisen esittämiselle. Periytyminen käsitetään ylä- ja alatyypin avulla.	Periytyminen tapahtuu elementtejä laajentamalla (xs:extension).	Kullakin käsitteellä (luokalla) voi olla yläluokkia, jotka ilmaistaan elementillä rdfs:subClassOf. Tällöin tapahtuu periytymistä, kun käsitteelle tulee automaattisesti myös kaikkien sen yli luokkien ominaisuudet.
Roolit	Roolit voidaan esittää luokkien välisinä periytymissuhteina (esimerkiksi asiakas perii henkilön, ja asiakkaalla voi olla omia ominaisuuksiaan henkilön ominaisuuksien lisäksi). Luokkien väliseen viivaan voidaan lisätä kommentiksi "Rooli", mikäli tarpeen. Luokkien välille voidaan myös luoda assosiaatio, joka nimetään rooliksi tai roolin nimeksi. Rooli voi myös olla oma yläluokkansa, joka on yhteydessä luokkiin.	Käsitteiden välille voidaan luoda suhde "rooli" tai nimetä rooli kuten "asiakas". Entiteettien välisiin suhdeviivoihin voidaan myös merkitä roolin nimi.	Rooliviittaukset voidaan esittää xs:keyref, xs:key ja xs:unique -elementtien avulla.	Käsitteelle voidaan luoda ominaisuus "rooli" (roleOf, hasRole), joka saa arvoksi sen roolin, missä käsite toimii.
Ryhmittely	Luokkarakenteen avulla käsitteet ryhmittyvät hierarkkisesti.	Ei voida esittää.	Ryhmittelyä voidaan tehdä nimiavaruuksien avulla: laitetaan samaan ryhmään kuuluvat elementit samaan nimiavaruuteen.	Luokkarakenteen avulla käsitteet ryhmittyvät hierarkkisesti. Ryhmittelyä voidaan myös tehdä nimiavaruuksien avulla.
Muut suhteet	Muita suhteita voidaan esittää viivoilla tarpeen mukaan. Viivoja voidaan myös nimetä vapaamuotoisesti.	Muita suhteita voidaan esittää viivoilla tarpeen mukaan. Viivoja voidaan myös nimetä vapaamuotoisesti.	Viittaukset voidaan esittää xs:keyref, xs:key ja xs:unique -elementtien avulla. Elementtien korvaavuudet voidaan esittää substitutioryhmien kautta.	Muiden suhteiden määrittely on hyvin helppoa määrittelemällä käsitteelle uusia ominaisuuksia (eli suhteita), jotka linkitetään toiseen käsitteeseen. Ominaisuuksia voidaan nimetä aivan vapaasti, eli periaatteessa millainen tahansa suhde on mahdollista esittää ontologiassa.

	UML-luokkakaavio	E-R -malli	XML Schema	Ontologia
Rajoitteet	Voidaan esittää UML-notaation sallimia rajoitteita suhteisiin liitettävillä symboleilla. Tällaisia rajoitteita ovat kardinaliteettisuhteet (merkintöinä suhdeviivan päiden ohessa), luokkien välinen riippuvaisuus (katkoviivanuoli), rajoitteet kuten XOR (pakollisuus), ym. loogiset operaattorit (merkintöinä suhdeviivan ohessa {}-sulkujen sisällä). Aaltosulkeiden sisään voidaan myös lisätä erilaisia sanallisia rajoitteita kuten disjoint (luokkien erillisuus), complete (luokkien täydellisyys) ym.	Notaation sallimat rajoitteet suhteisiin liitettävillä symboleilla. Kardinaliteettisuhteet (merkintöinä suhdeviivan päiden ohessa), pakollisuussuhteet, XOR. Suhteisiin voidaan lisätä attribuutteja.	Koostumussuhteita voi rajoittaa hyvin tarkasti (minOccurs, maxOccurs, sequence, choice).	Suhteita voidaan rajoittaa kardinaliteetilla (min, max, tarkka) ja joukko-opin operaatioiden avulla (unioni, leikkaus, on olemassa (some), kaikilla arvoilla (all)) sekä suhteen tyyppiä voidaan rajoittaa (functional, inverse functional, transitive, symmetric, asymmetric, reflexive, irreflexive). Suhteen aihealuetta (domain) ja kattavuutta (range) voi myös rajoittaa. Suhteille voidaan määritellä käänteis-suhteita.
Dokumentointivälineet ja työkalut				
Graafinen esitys	Malli on itsessään graafinen, joten erillistä graafista esitystä ei tarvitse.	Malli on itsessään graafinen, joten erillistä graafista esitystä ei tarvitse.	Työkalujen (kuten XMLSpy, hyperModel) avulla voidaan esittää kaaviokuvia.	Mahdollista työkalujen (kuten Protege) avulla voidaan esittää muun muassa luokkahierarkiana, käsitekaaviona ja luokkakaaviona.
Muokattavuus	Tietomallin muokattavuus helppoa työkalujen avulla. Muokkaukset tulevat suoraan graafiseen malliin.	Tietomallin muokattavuus helppoa työkalujen avulla. Muokkaukset tulevat suoraan graafiseen malliin.	Tietomallin muokattavuus helppoa työkalujen avulla. Riippuen työkalusta kaaviokuvien muokkaukset siirtyvät skeemoihin (XMLSpy).	Tietomallin muokattavuus helppoa työkalujen avulla. Tietomallista tuotetuista kuvista ei yleensä voida muokata.
Dokumentointi ja ylläpidettävyys	Tietomalli toimii samalla myös dokumentaationa. Sen ohelle voidaan tehdä vapaamuotoista oheisdokumentaatiota, mikäli tarpeen.	Tietomalli toimii samalla myös dokumentaationa. Sen ohelle voidaan tehdä vapaamuotoista oheisdokumentaatiota, mikäli tarpeen.	Skeemoista voidaan luoda javadoc-tyyppinen dokumentaatio html-sivuna ja word-dokumenttina. Dokumentaatio joudutaan luomaan aina uudelleen mikäli skeemat muuttuvat.	Ontologiasta voidaan tuottaa html-sivuja, joissa voi olla myös kaaviokuvia. Dokumentaatio joudutaan luomaan aina uudelleen, mikäli ontologiat muuttuvat.
Tekninen hyödynnettävyys				
XML-skeemojen tuottaminen	Onnistuu automaattisesti ohjelmistojen (kuten hyperModel) kautta.	On mahdollista ohjelmistojen kautta, mutta voi olla vaikeaa.	Ei tarvitse tuottaa, yksi tietomalli.	Onnistuu automaattisesti ohjelmistojen (kuten TopBraid) kautta.
Tietomallin muutosten välittyminen XML-skeemoihin	Muutokset eivät välity suoraan, skeemat täytyy tuottaa uudelleen.	Muutokset eivät välity suoraan, skeemat täytyy tuottaa uudelleen.	Muutokset tehdään suoraan skeemaan.	Muutokset eivät välity suoraan, skeemat täytyy tuottaa uudelleen.

	UML-luokkakaavio	E-R -malli	XML Schema	Ontologia
Asiakirjataso	Kustakin asiakasasiakirjasta voidaan tehdä oma luokkakaavionsa, joko supistetussa muodossa tai kaikki luokkien tiedot (kuten attribuutit) näyttäen.	Kustakin asiakasasiakirjasta voidaan tehdä oma E-R-kaavionsa.	Kullakin asiakirjalla on oma skeemansa, jossa on määritelty mitä elementtejä asiakirja sisältää. Elementit voidaan määritellä muissa nimiavaruuksissa kuin itse asiakirjassa.	Voidaan toteuttaa samaan tapaan kuin XML-skeemoissa. Kunkin asiakirjan rakenne voidaan myös halutessa esittää omassa "sosiaalihuollon asiakas-asiakirjat" -ontologiassa.
Versiointi	Kaavioista tuotettava skeemahierarkiaan uudet skeemat, mikäli kaavioita muutetaan.	Kaavioista tuotettava skeemahierarkiaan uudet skeemat, mikäli kaavioita muutetaan.	Skeemahierarkian avulla ylläpidetään skeemojen eri versioita. Vanhat skeemat ovat aina saatavilla.	Mahdollista soveltaa samanlaista tapaa kuin skeemojen kanssa.
Linkitettävyyssprosessimallien tietovirtoihin	Ei voida tehdä, tai on hyvin hankalaa mikäli käytössä on QPR:n Process-Guide, muilla työkaluilla (kuten Eclipse) saattaa olla mahdollista.	Lienee mahdollista, mutta hyvin hankalaa.	Sopivilla XSLT-muunnoksilla voi olla mahdollista päästä informaatiohierarkiaan, mutta se on hankalaa.	Lienee mahdollista, mutta hyvin hankalaa.
Suhde CCTS:ään	On yhdistettävissä CCTS:n kanssa, mutta voi tulla konflikteja ja yhteensovittamista (kuten mitä tehdä perimisen kanssa).	On mahdollista yhdistää CCTS:n kanssa, mutta rajoittaa silloin tietomallia.	CCTS-taulukoista on mahdollista tuottaa skeemat suoraan.	On yhdistettävissä CCTS:n kanssa, mutta CCTS asettaa tiettyjä rajoitteita ontologiaan

## B Toimeentulotukipäätöksen asiakirjarakennetaulukko

TOIMEENTULOTUKIPÄÄTÖS					19.4.2010	versio	0.46	
Tikesos-määrittäminen	Tiedon esitysmuoto	Dokumentaatio	Määritelmä	Näyttömuoto-ohje (koskee suppeaa näyttömuotoa)	Selitys/esimerkki	Toistuva	Pakollinen	Järjestys
Toimeentulotukipäätös	Asiakirja on hallintopäätös perustoimeentuluesta/täydentävästä toimeentuluesta/ennaltaehkäisevästä toimeentuluesta sekä mahdollisesta toimeentulotuen perinnästä.							
Kaikissa asiakirjatyypeissä olevat roolit								
Asiakas	Asiakas			Käytetään yhtenäistä ohjetta		kyllä	kyllä	
Yksityishenkilö								
Edunvalvoja	Edunvalvoja			Ei näyttömuodossa		kyllä	ei	
Yksityishenkilö								
Organisaatio								
Edunvalvontavaltuutettu	Edunvalvontavaltuutettu			Ei näyttömuodossa		kyllä	ei	
Yksityishenkilö								
Edunvalvottava	Edunvalvottava			Ei näyttömuodossa		kyllä	ei	
Yksityishenkilö								
Edunvalvontavaltuuttaja	Edunvalvontavaltuuttaja			Ei näyttömuodossa		kyllä	ei	
Yksityishenkilö								
Huoltaja	Huoltaja			Ei näyttömuodossa		kyllä	ei	
Yksityishenkilö								
Huollettava	Huollettava			Ei näyttömuodossa		kyllä	ei	
Yksityishenkilö								
Palvelunjärjestäjä	Palvelunjärjestäjä			Käytetään yhtenäistä ohjetta		ei	ei	
Organisaatio								
Palveluntuottaja	Palveluntuottaja			Käytetään yhtenäistä ohjetta		kyllä	ei	

Yksityishenkilö				
Organisaatio				
Ammatillinen henkilö				
Palveluyksikkö				
Laatija	Laatija	Käytetään yhtenäistä ohjetta	ei	kyllä
Ammatillinen henkilö				
Sukunimi				
Etunimi				
Virkanimike				
Organisaatio				
Nimi				
Yhteystiedot				
Osoite				
Paras osoite		Näyttömuodossa on vain käynti- ja postiosoite		
Puhelinnumero				
Faksinumero				
Kaikissa asiakirjatyypeissä olevat suhteet				
Edunvalvontavaltuutusuhde	Edunvalvontavaltuutusuhde	Ei näyttömuodossa	kyllä	ei
Edunvalvontasuhte	Edunvalvontasuhte	Ei näyttömuodossa	kyllä	ei
Huoltajuussuhde	Huoltajuussuhde	Ei näyttömuodossa	kyllä	ei
Yleisen asiakirjatyyppin mukaiset roolit				
Valmistelija			kyllä	kyllä
Ammatillinen henkilö				
Sukunimi				
Etunimet				
Virkanimike				
Organisaatio				
Nimi				
Päätöksentekijä			ei	kyllä
Ammatillinen henkilö				
Sukunimi				
Etunimet				

Virkanimike		
Organisaatio		
Nimi		
Lisätietojen antaja		kyllä ei
Ammatillinen henkilö		
Sukunimi		
Etunimet		
Virkanimike		
Organisaatio		
Nimi		
Yhteystiedot		
Osoite	Näyttömuodossa käytösioite ja postiosoite	
Puhelinnumero		
Faksinumero		
Tiedoksisaja		kyllä kyllä
Yksityishenkilö	Jos henkilö on edunvalvoja tai edunvalvontavaltuutettu, tulostetaan lisäksi otsikko "Edunvalvoja" tai "Edunval- vontavaltuutettu". Ei tuloste- ta jos sama henkilö on Hakija.	
Sukunimi		
Etunimet		
Henkilötunnus		
Väliaikainen henkilötunnus		
Yhteystiedot		
Osoite		
Paras osoite		
Puhelinnumero		
Sähköpostiosoite		
Viranomainen	Muutoksenha- kuviranomainen	

Organisaatio	
Nimi	
Yhteystiedot	
Osoite	
Paras osoite	
Kirjelmän vastaanottaja	Muutoksenha- kukirjelmän vastaanottaja
Organisaatio	
Nimi	
Yhteystiedot	
Osoite	
Paras osoite	

Suppean asiakirjatyyppin mukaiset roolit

Hakija	Tulostetaan lisäotsikko "Tiedoksisaja", mikäli tämä on sama henkilö kuin Tiedoksisaja.	kyllä	kyllä	3
Yksityishenkilö				
Sukunimi				
Etunimet				
Henkilötunnus				
Väliaikainen henkilötunnus				
Yhteystiedot				
Osoite	Näyttömuodossa on vain paras osoite			
Paras osoite				
Puhelinnumero				
Sähköpostiosoite				
Pankkiyhteys				
BIC-koodi				
IBAN tunnus				
Lapsi		kyllä	ei	4

Yksityishenkilö								
Sukunimi								
Etunimet								
Henkilötunnus								
Väliaikainen henkilötunnus								
Yhteystiedot			Näyttömuodossa on vain paras osoite					
Osoite								
Paras osoite								
Puoliso			Ei näyttömuodossa					
Yksityishenkilö								
Vanhempi			Ei näyttömuodossa					
Yksityishenkilö								
Suppean asiakirjatyyppin mukaiset suhteet								
Lapsi-vanhempisuhte		Lapsi-vanhempisuhte		Ei näyttömuodossa		kyllä	ei	
Parisuhde		Parisuhde		Ei näyttömuodossa		kyllä	ei	
AYR-mukaiset asiakirjarakenteet								
Header			Käytetään yhtenäistä ohjetta			ei	kyllä	0
Laatimisaika								
Hyväksymisaika								
Voimassaolo								
Asiakirja						1		
Palvelunjärjestäjä		rooli		Ks. Roolimääritelmä				
Palveluntuottaja		rooli		Ks. Roolimääritelmä				
Asiakas		rooli		Ks. Roolimääritelmä				
Laatija		rooli		Ks. Roolimääritelmä				
Päätös						ei	kyllä	2
Lauselma								5
Myönteinen päätös								6
Palvelu								7
Voimassaolo								
Perustelu								8
Lainkohta								9

Valmistelija	rooli								10
Paatoksentekijä	rooli								11
Lisätietojen antaja	rooli								12
Muutoksenhaku									17
Keino									
Ohjeteksti									
Viranomainen	rooli								
Kirjelmän vastaanottaja	rooli								
Aika									
Aikayksikkö									
Päätymispäivä									
Tiedoksisaja	rooli								
Tiedoksiantomenettely									18
Asiakirjakohtaiset rakenteet									
Hakemuksen kuvaus	teksti	5.6.4					ei	ei	
Perusosa									13
Täysimääräinen									
Alentaminen									
Peruste									
Suuruus									
Kuvaus									
Lainkohdat									
Alennettu									
Päätöksen toimeenpano	teksti	5.6.7	etuuden suoritus- tustapa		tilisiirto, lasku, maksusitoumus, käteismaksu		kyllä	kyllä	14
Sopimus toimeenpanosta	teksti	5.6.7	kuvaus päätök- sentekijän ja hakijan välises- tä, etuuden toimeenpanoon liittyvästä so- pimuksesta				ei	ei	15

Perintä	Perintä	kyllä	ei	16
Peruste				
Kohde				
Kuvaus				
Ajanjakso				
Päätöksentekijän allekirjoitus	sähköinen	ei	ei	19