
Palvelukeskeisen arkkitehtuurin mallit: neljän mallin tarkastelu

Diplomityö
Turun yliopisto
Informaatioteknologian laitos
Ohjelmistotekniikka
2012
Tapani Lammervo

Tarkastajat:
Antero Järvi
Ville Leppänen

Palvelukeskeinen arkkitehtuuri eli SOA (service-oriented architecture) on arkkitehtuurityyli, joka on vaikuttanut merkittävästi suuntauksena viimeisen vuosikymmenen aikana. Palvelukeskeinen ajattelutapa voidaan omaksua yrityksessä sekä liiketoiminnan että IT:n alueella, ja sillä on potentiaalisesti huomattava vaikutus yritykseen ja tämän toimintaan.

SOAn ymmärtämisen ja toteuttamisen tukemiseksi on esitetty useita SOAa kuvaavia malleja. SOAn aihealueen laajuuden vuoksi yksittäisen mallin avulla voidaan kuitenkin mallintaa SOAa vain osittaisesti. Mallit poikkeavat usein näkökulmiltaan ja kuvausalueeltaan huomattavasti toisistaan, minkä lisäksi useat SOAa kuvaavat mallit ovat monimutkaisia ja abstrakteja. Tällaiset seikat vaikeuttavat mallien ymmärtämistä ja vertailua.

Tässä tutkielmassa tarkastellaan neljää standardointijärjestön esittämää SOA-mallia: OASISin referenssimallia ja referenssiarkkitehtuuria sekä The Open Groupin ontologiaa ja referenssiarkkitehtuuria. Tutkielman tavoitteena on selvittää, millaisia nämä mallit ovat ja millainen suhde niillä on todellisiin tuotteisiin perustuvaan SOA-ympäristöön. Tähän vertailuun käytetään Oraclen SOA-tuotevalikoimaa.

Malleja tutkitaan aluksi yksittäin tarkastelemalla niiden yleisiä ominaisuuksia, sisältöä ja soveltamista. Tämän jälkeen malleja vertaillaan keskenään neljän merkittävän näkökulman avulla. Lopuksi malleja verrataan Oraclen tuotteisiin perustuvaan SOA-ympäristöön.

Tutkimuksen tuloksina esitetään useita havaintoja tutkittavista malleista. Malleissa havaittiin huomattavia eroavaisuuksia erityisesti niiden kuvausalueeseen ja yleiseen näkökulmaan nähden. Malleista tunnistettiin kaksi keskeistä yleistä näkökulmaa: rakennekeskeinen näkökulma ja käsitteellinen, selittävä näkökulma. Mallien vertailun perusteella esitetään viisi osa-alueita, joiden avulla mallien yhdessä kuvaama aihealue voidaan kuvata ja jäsentää karkeasti. Vertailun perusteella tunnistettiin myös kolme merkittävintä tapaa, joilla mallit tukevat SOAn ymmärtämistä ja toteuttamista. Mallien ja Oraclen SOA-ympäristön vertailun perusteella havaittiin, että mallien käyttömahdollisuudet Oraclen SOA-ympäristön ymmärtämisen ja toteuttamisen tukemisessa ovat melko vähäiset. Tutkielman tulosten ensisijainen merkitys on tarkasteltujen SOA-mallien ominaisuuksien ja merkityksen ymmärtämisen tukeminen.

Asiasanat: palvelukeskeinen arkkitehtuuri, SOA, ohjelmistoarkkitehtuuri, malli, referenssimalli, ontologia, referenssiarkkitehtuuri

UNIVERSITY OF TURKU
Department of Information Technology

TAPANI LAMMERVO: Service-Oriented Architecture Models: A Study of Four Models

Master of Science in Technology Thesis, 102 p., 3 app. p.
Software Engineering
April 2012

Service-oriented architecture (SOA) is an architectural style, which has had a remarkable influence as a trend over the last decade. Service-oriented paradigm can be adopted in an enterprise both in business and IT domains, and it has potentially a significant impact on the enterprise and its operations.

Several models for SOA have been proposed to support the understanding and realizing of SOA. Due to the wide scope of SOA, individual models can model SOA only partially. The models often differ in terms of their viewpoints and coverage, and, in addition, many SOA models are complex and abstract. These, among other things, impede the understanding and comparing of the models.

Four SOA models proposed by standards organizations are examined in this study: the OASIS reference model and reference architecture and The Open Group ontology and reference architecture. The goal of the thesis is to find out the characteristics of these models and how the models compare to an SOA environment that is based on actual products. Oracle's SOA products are used for the comparison.

The models are first studied individually by examining their general characteristics, contents, and applicability. Next, the models are compared with each other using four relevant viewpoints. Finally, the models are compared with an SOA environment based on Oracle's products.

Several findings on the examined models are presented as the results of this survey. Significant differences were observed in the models, especially with respect to their coverage and general viewpoint. Two essential general viewpoints were identified from the models: a structure-centered viewpoint and a conceptual, explaining viewpoint. Based on the comparison of the models, five divisions are proposed for outlining and analyzing the collective subject matter of the models. On the basis of the comparison, three essential means for using the models to support the understanding and realizing of SOA are also identified. Based on comparing the models with Oracle's SOA environment, it was concluded that the models have fairly little potential for supporting the understanding and realizing of Oracle's SOA environment. The results of this study have relevance primarily with respect to supporting the understanding of the characteristics and value of the examined SOA models.

Keywords: service-oriented architecture, SOA, software architecture, model, reference model, ontology, reference architecture

Sisältö

Kuvat	iv
Taulukot	v
Käsitteet ja lyhenteet	vi
1 Johdanto	1
1.1 Tausta ja motivointi.....	1
1.2 Tavoitteet	2
1.3 Menetelmät	2
1.4 Rajaus	3
1.5 Työn rakenne.....	4
1.6 Suomennokset	5
2 SOAn mallintaminen	6
2.1 Arkkitehtuurikuvaukset	6
2.2 SOAn yleiskatsaus.....	8
2.2.1 SOAn historia	8
2.2.2 SOAn peruselementit	9
2.2.3 Palvelukeskeisyys	12
2.2.4 SOAn määritelmät	13
2.2.5 SOAn hyödyt	14
2.3 SOA-mallityypit	15
2.3.1 Mallityyppien nimitykset	16
2.3.2 Mallien abstraktiotaso ja kattavuus	17
2.3.3 Mallityyppien merkitys järjestelmän toteuttamisessa.....	18

3	SOAn perusominaisuuksia kuvaavat mallit	21
3.1	OASISin referenssimalli	21
3.1.1	Yleiskatsaus.....	22
3.1.2	Mallin sisältö	22
3.1.3	Mallin soveltaminen	25
3.1.4	Havainnot	26
3.2	The Open Groupin ontologia	27
3.2.1	Yleiskatsaus.....	28
3.2.2	Mallin sisältö	29
3.2.3	Mallin soveltaminen	32
3.2.4	Havainnot	34
4	Referenssiarkkitehtuurit	36
4.1	OASISin referenssiarkkitehtuurin perusta	36
4.1.1	Yleiskatsaus.....	37
4.1.2	Mallin sisältö	38
4.1.3	Mallin soveltaminen	44
4.1.4	Havainnot	45
4.2	The Open Groupin referenssiarkkitehtuuri	48
4.2.1	Yleiskatsaus.....	48
4.2.2	Mallin sisältö	49
4.2.3	Mallin soveltaminen	60
4.2.4	Havainnot	61
5	SOA-tuotteet	64
5.1	Oracle SOA Suite	65
5.2	Muut Oraclen SOA-ratkaisut	71
5.2.1	Oracle SOA Governance.....	71
5.2.2	Oracle Data Integration.....	72
5.2.3	Oracle Enterprise Gateway.....	72
5.2.4	Oracle Business Process Analysis Suite	73
5.2.5	Oracle Application Integration Architecture.....	73
6	SOA-mallien ja -tuotteiden vertailu	74

6.1	SOA-mallien vertailu.....	74
6.1.1	Tarkastelualue.....	75
6.1.2	Rakennekeskeisyys	76
6.1.3	Abstraktiotaso.....	77
6.1.4	Dynaamiset piirteet.....	77
6.1.5	Vertailun yhteenveto.....	78
6.1.6	Johtopäätökset	80
6.2	SOA-mallien vertaaminen Oraclen SOA-tuotteisiin.....	81
6.2.1	OASISin referenssimalli	82
6.2.2	The Open Groupin ontologia.....	83
6.2.3	OASISin referenssiarkkitehtuuri	84
6.2.4	The Open Groupin referenssiarkkitehtuuri	85
6.2.5	Johtopäätökset	87
7	Yhteenveto	89
	Viitteet	95
	Liite A The Open Groupin referenssiarkkitehtuurin metamalli	103
	Liite B The Open Groupin referenssiarkkitehtuurin palvelukategorioiden kuvaukset	104

Kuvat

2.1	Arkkitehtuurikuvauksen käsitteellinen malli.....	7
2.2	Alkukantaisen SOAn elementit.....	10
2.3	SOAn kolmiomalli.....	11
2.4	CBDI-SAE -referenssikehys.....	16
2.5	SOA-mallien abstraktiotason ja kattavuuden vertailu.....	18
2.6	Mallien yhteydet yrityksen arkkitehtuurin abstraktiotasoihin.....	19
2.7	Eri tahojen esittämien mallien merkitys SOAn toteuttamisessa.....	20
3.1	OASISin referenssimallin graafinen tiivistelmä.....	23
3.2	The Open Groupin ontologian luokkahierarkia.....	29
3.3	The Open Groupin ontologian graafinen tiivistelmä.....	30
4.1	The Open Groupin referenssiarkkitehtuurin looginen kerrosrakenne.....	50
4.2	The Open Groupin referenssiarkkitehtuurin palvelukategoriat.....	54
5.1	SCA-komposiitti.....	67

Taulukot

2.1	Palvelukeskeisyyden periaatteet.....	13
2.2	SOA-mallityyppien kuvaukset.....	17
3.1	OASISin referenssimallin palvelu-käsitteen ja palvelun dynamiikkaa kuvaavien pääkäsitteiden kuvaukset.....	24
3.2	OASISin referenssimallin palvelun metatasoa kuvaavien pääkäsitteiden kuvaukset	25
3.3	The Open Groupin ontologian luokkien kuvaukset.....	30
4.1	The Open Groupin referenssiarkkitehtuurin Osallistuminen SOA-ekosysteemissä -näkömän sisältämät mallit.....	40
4.2	The Open Groupin referenssiarkkitehtuurin SOA-ekosysteemin toteuttaminen -näkömän sisältämät mallit.....	41
4.3	The Open Groupin referenssiarkkitehtuurin Omistajuus SOA-ekosysteemissä -näkömän sisältämät mallit.....	43
6.1	SOA-mallien vertailun yhteenveto	79

Käsitteet ja lyhenteet

BAM	Business activity monitoring, suom. liiketoiminta-aktiiviteetin monitorointi.
BPEL	Business Process Execution Language. Lyhennetty nimestä Web Services Business Process Execution Language, WS-BPEL.
CEP	Complex event processing, suom. monimutkaisten tapahtumien käsittely.
EJB	Enterprise JavaBeans.
ESB	Enterprise service bus, suom. palveluväylä.
GRA	Global Reference Architecture Framework.
HTTP	Hypertext Transfer Protocol.
IEEE	Institute of Electrical and Electronics Engineers.
MVC	Model–View–Controller.
OASIS	Organization for the Advancement of Structured Information Standards.
OSB	Oracle Service Bus.
OWL DL	Web Ontology Language, Description Logic.

REST-palvelu	REST-arkkitehtuuriin (Representational State Transfer) perustuva palvelu, jonka rajapinta perustuu HTTP-protokollan vakio- tai muotoisiin metodeihin.
S-RAMP	SOA - Repository Artifact Model and Protocol.
SCA	Service Component Architecture.
SLA	Service level agreement, suom. palvelutasosopimus.
SOA	Palvelukeskeinen arkkitehtuuri (engl. service-oriented architecture). Palvelukeskeistä paradigmaa noudattava arkkitehtuurityyli.
SOAP	Web-palveluissa käytettävä viestintäprotokolla.
TOGAF	The Open Group Architecture Framework. Yritysarkkitehtuurikehys.
UDDI	Universal Description, Discovery and Integration. Palvelurekistereissä käytettävä palvelujen julkaisu- ja hakuprotokolla.
UML	Unified Modeling Language.
W3C	World Wide Web Consortium.
Web service	Web-palvelu. Ohjelmistojärjestelmä, joka on suunniteltu tukemaan koneiden välistä yhteensopivaa vuorovaikutusta, joka tapahtuu verkon välityksellä. [10] Web-palvelut käyttävät tyypillisesti HTTP-protokollaa ja standardin mukaista rajapintaa (kuten WSDL-rajapintaa tai REST-palveluissa HTTP:n metodeja).
WSDL	Web Services Description Language. Web-palvelujen (rajapinnan) kuvaamiseen käytettävä kieli.

Luku 1

Johdanto

1.1 Tausta ja motivointi

Palvelukeskeinen arkkitehtuuri eli SOA (service-oriented architecture) on useiden yritysten ja standardointijärjestöjen vaikutuksesta kehittynyt arkkitehtuurityyli, joka on vaikuttanut suuntauksena erityisesti 2000-luvun alun aikana. [1] Palvelukeskeisyys on paradigma eli ajattelutapa, jossa toiminnallisuutta pyritään jäsentämään palveluiksi ja joka ohjaa ratkaisujen kehittämistä ja käyttämistä. [2] Palvelukeskeinen ajattelutapa voidaan omaksua yrityksessä¹ sekä liiketoiminnan että IT:n alueella, ja sillä on potentiaalisesti huomattava vaikutus yritykseen ja tämän toimintaan. Palvelukeskeisyyden soveltamistavoista ja niiden vaikutuksista on käyty paljon keskustelua. Keskustelussa korostetaan usein SOAn tiettyjä merkittäviä piirteitä ja hyötyjä, kuten liiketoimintaketteryyttä ja resurssien uudelleenkäytettävyyttä [3]. SOA vaikuttaa kuitenkin moneen osapuoleen ja alueeseen yrityksessä, joten sitä voidaan tarkastella varsin monesta näkökulmasta.

SOAn ymmärtämisen ja toteuttamisen tukemiseksi on esitetty useita SOAa kuvaavia malleja. SOAn aihealueen laajuuden vuoksi yksittäisen mallin avulla voidaan kuitenkin mallintaa SOAa vain osittaisesti. Mallit poikkeavat usein näkökulmiltaan ja kuvausalueeltaan merkittävästi toisistaan, ja niiden käyttämät termistöt ovat myös paikoin epäyhteneviä keskenään. Lisäksi useat SOAsta esitetyt mallit ovat

¹ Palvelukeskeisyyttä voidaan soveltaa lähtökohtaisesti missä tahansa organisaatiossa. Tässä tutkielmassa tarkastellaan kuitenkin yksinkertaisuuden vuoksi kohdeympäristönä vain yritystä.

monimutkaisia ja abstrakteja. Tällaiset seikat vaikeuttavat mallien ymmärtämistä ja vertailua.

1.2 Tavoitteet

Tässä tutkielmassa tarkastellaan SOAsta esitettyjä arkkitehtuurillisia malleja. Malleilla tarkoitetaan tässä yleisesti abstrakteja tai yksinkertaistettuja SOAn kuvauksia, jotka on tarkoitettu käytettäväksi SOAn ymmärtämisen tai toteuttamisen tukemiseen. Tutkielmaan on valittu tarkasteltavaksi neljä standardointijärjestön esittämää, keskenään varsin erilaista SOA-mallia. Tutkielman tavoitteena on selvittää, millaisia nämä mallit ovat ja millainen suhde niillä on todellisiin tuotteisiin perustuvaan SOA-ympäristöön. Tutkimuskysymykset määritellään tarkemmin seuraavasti:

1. Millaisia tarkasteltavat SOA-mallit ovat?
 - a) Mitä aiheita mallit kuvaavat?
 - b) Miten mallit tukevat SOAn ymmärtämistä ja toteuttamista?
 - c) Millaisia eroavaisuuksia ja yhtäläisyyksiä malleilla on?
 - d) Millaista kokonaisaihealuetta mallit kuvaavat yhdessä?
2. Millainen suhde tarkasteltavilla malleilla on todellisiin tuotteisiin perustuvaan SOA-ympäristöön?
 - a) Kuinka hyvin mallit vastaavat tällaista SOA-ympäristöä?
 - b) Tukevatko mallit tällaisen SOA-ympäristön ymmärtämistä ja toteuttamista?

Lisäksi tutkielmassa pohditaan mallien merkitystä ja sitä, ovatko ne onnistuneet tavoitteissaan.

1.3 Menetelmät

Malleja tutkitaan aluksi yksittäin tarkastelemalla niiden yleisiä ominaisuuksia, sisältöä ja soveltamista. Tarkastelussa kiinnitetään huomiota erityisesti mallien seuraaviin ominaisuuksiin:

- kuvausalue
- abstraktiotaso
- käyttötarkoitus ja kohdeyleisö
- näkökulma ja lähestymistapa

- kuvaustapa
- huomattavat erot ja yhtäläisyydet muihin malleihin

Tämän jälkeen malleja vertaillaan keskenään neljän merkittävän näkökulman avulla, jotka valittiin malleista tehtyjen havaintojen perusteella. Nämä näkökulmat ovat *tarkastelualue*, *rakennekeskeisyys*, *abstraktiotaso* ja *dynaamiset piirteet*. Vertailulla pyritään määrittämään mallien olennaisia erityspiirteitä, tunnistamaan mallien yhteisiä ominaisuuksia sekä selittämään mallien välisiä eroavaisuuksia. Tämä vertailu toimii myös perustana mallien ja todellisiin tuotteisiin perustuvan SOA-ympäristön väliselle vertailulle.

Lopuksi malleja vertaillaan todellisiin tuotteisiin perustuvaan SOA-ympäristöön. Vertailussa kiinnitetään huomiota siihen, kuinka kattavia mallit ovat; millaisia puutteita niissä on; kuinka helposti mallien ja yksittäisten tuotteiden ja tekniikoiden välinen yhteys on tunnistettavissa; sekä tukevatko mallit valitun SOA-ympäristön ymmärtämistä ja toteuttamista.

1.4 Rajaus

SOAn kuvaamiseksi on esitetty sekä tieteellisiä, kaupallisia että standardointijärjestöjen kehittämiä malleja, joilla on keskenään varsin erilaisia käyttötarkoituksia. Tutkielmaan pyrittiin valitsemaan yleisiä ja relevantteja malleja, jotka ovat riittävän erilaisia keskenään ja täydentävät toisiaan. Tarkasteltavaksi valittiin standardointijärjestöjen esittämiä malleja, sillä nämä edustavat useiden tahojen yksimielisyyttä, niillä on mahdollisuus saada laaja hyväksyntä ja niillä on pääasiassa hyvin määritellyt käyttötarkoitukset. Valitut mallit ovat OASISin (Organization for the Advancement of Structured Information Standards) SOA-referenssimalli ja SOA-referenssiarkkitehtuuri sekä The Open Groupin SOA-ontologia ja SOA-referenssiarkkitehtuuri. Konkreettiseksi vertailukohdaksi valittiin Oraclen SOA-tuotevalikoima. Oraclen tuotteet valittiin ensiksi siksi, että Oraclen SOA-tuotevalikoma on hyvin laaja, mikä mahdollistaa monipuolisen vertailun, ja toiseksi siksi, että Oracle ei ole osallistunut valittujen spesifikaatioiden kehittämiseen, minkä vuoksi sen tuotteita voidaan pitää neutraalina vertailukohtana. Mallien vertailu myös muihin SOA-ympäristöihin olisi tehnyt tutkimuksesta

monipuolisemman, mutta tutkielman rajallisen laajuuden vuoksi muut SOA-ympäristöt jouduttiin jättämään tarkastelun ulkopuolelle.

1.5 Työn rakenne

Aluksi, luvussa 2, luodaan katsaus eräisiin SOAn ja SOA-mallien ymmärtämisen kannalta tärkeisiin aiheisiin. Arkkitehtuurikuvaus-käsite esitellään lyhyesti, minkä jälkeen tarkastellaan SOAa yleisesti ja luodaan alustava katsaus erilaisiin mallityyppeihin ja näiden merkitykseen.

Malleja tutkitaan yksittäin luvuissa 3 ja 4. SOAn perusominaisuuksia kuvaaviin malleihin keskitytään luvussa 3 ja referenssiarkkitehtuureihin luvussa 4. Tarkastelussa edetään järjestyksessä abstrakteimmasta ja teknisesti suppeimmasta mallista konkreettisimpaan ja teknisesti kattavimpaan malliin. Kaikkien mallien tutkimisessa noudatetaan samaa nelivaiheista lähestymistapaa: luodaan yleiskatsaus malliin, esitetään tiivistelmä mallin sisällöstä, tarkastellaan mallin soveltamistapoja ja esitetään mallista tehdyt havainnot. Tarkasteltaviin soveltamistapoihin on sisällytetty joitain tapausesimerkkejä. Nämä koskevat kuitenkin vain mallien hyödyntämistä muiden spesifikaatioiden perustana.

Luvussa 5 esitellään Oraclen SOA-tuotteisiin perustuva SOA-ympäristö ja siinä käytettävä SCA-sovelluskehitysmalli (Service Component Architecture). Oraclen SOA-tuotteista esitellään sen keskeinen SOA-ratkaisu, useista tuotteista koostuva *SOA Suite*, sekä eräitä muita SOAan läheisesti liittyviä ratkaisuja.

Luvussa 6 vertaillaan malleja ensiksi keskenään ja toiseksi Oraclen SOA-ympäristöön. Tässä luvussa muodostetaan tutkielman lopulliset tulokset.

Lopuksi, luvussa 7, esitetään yhteenveto tutkielmasta. Luvussa pohditaan myös mallien merkitystä ja sitä, ovatko ne onnistuneet tavoitteissaan.

1.6 Suomentokset

Muihin lähteisiin perustuva sisältö, kuten mallit, kuvat ja lainaukset, on esitetty tässä tutkielmassa pääosin suomentuna. Eräät kuvat on esitetty ymmärrettävyyden vuoksi alkuperäisellä kielellä.

Luku 2

SOAn mallintaminen

Mallien avulla voidaan kuvata monimutkaista ongelma-aluetta tai järjestelmää yksinkertaistetusti ja abstraktisti ja tällä tavoin tukea ongelma-alueen ymmärtämistä ja järjestelmän toteuttamista. Tässä tutkielmassa tarkasteltavat mallit ovat SOAn arkkitehtuurillisia kuvauksia. Ne keskittyvät pääasiassa SOAn teknisiin piirteisiin mutta ilmentävät myös SOAn muita аспекteja, kuten liiketoimintamuutoksiin sopeutuvaa ja erilaisiin ekosysteemeihin soveltuvaa luonnetta. Tarkasteltavat mallit poikkeavat huomattavasti toisistaan mm. kuvausalueeseen, abstraktiotasoon ja käyttötarkoitukseen nähden. Tässä luvussa käsitellään aiheita, jotka ovat tärkeitä SOAn ja siitä esitettyjen mallien ymmärtämisen kannalta. Luvussa esitellään ensin lyhyesti arkkitehtuurikuvauskäsite, minkä jälkeen tarkastellaan SOAa yleisesti ja luodaan alustava katsaus erilaisiin mallityyppeihin ja niiden merkitykseen.

2.1 Arkkitehtuurikuvaukset

Ohjelmistoarkkitehtuuri-termille on esitetty lukuisia määritelmiä. Eräs usein käytetty määritelmä on IEEE:n (The Institute of Electrical and Electronics Engineers) esittämä ohjelmistointensiivisen järjestelmän arkkitehtuurin määritelmä:

Arkkitehtuuri on järjestelmän perustavanlaatuinen rakenne, joka ilmenee sen komponenteissa sekä komponenttien suhteissa toisiinsa ja ympäristöön; sekä periaatteet, jotka ohjaavat sen suunnittelua ja kehittämistä. [4]

Tämä lienee useiden muiden määritelmien ohella yleisesti melko pätevä arkkitehtuurin määritelmä. On kuitenkin todettu, että arkkitehtuurilla voi olla erilainen merkitys eri

Tässä tutkielmassa tarkastellaan erilaisissa lähteissä esitettyjä SOAn arkkitehtuurillisia kuvauksia – joita tässä tutkielmassa kutsutaan yleisesti malleiksi. Osa lähteistä noudattaa tarkasti edellä kuvattua arkkitehtuurillisen kuvauksen semantiikkaa ja rakennetta, mutta toiset ovat esitykseltään vapaamuotoisia. Kuvauksessa käytetyt näkökulmat on tällöin pääteltävä itse kuvauksesta tai kontekstista.

2.2 SOAn yleiskatsaus

Näkemykset SOAsta ja sen hyödyistä ovat muuttuneet SOAn kehityksen myötä. Alussa keskityttiin SOAn peruselementteihin, kuten palvelurajapintaan, viesteihin ja palvelurekisteriin [1] [8], sekä teknisiin hyötyihin, kuten integroitavuuteen ja uudelleenkäytettävyyteen [1] [9]. Sitten SOAa on alettu tarkastella laajemmassa kontekstissa, ja sen merkittävimpänä hyötynä pidetään nykyisin useimmiten liiketoimintaketteryyttä [3]. SOAn toteuttamista ohjaavana ajattelutapana on alusta asti ollut palvelukeskeinen paradigma. Seuraavien kohtien keskeisenä lähteenä on käytetty Thomas Erlin teosta *Service-Oriented Architecture: Concepts, Technology, and Design* [1].

2.2.1 SOAn historia

SOA on kehittynyt useiden ohjelmistovalmistajien pyrkimyksistä edistää hajautettujen järjestelmien yhteensopivuutta ja sähköistä liiketoimintaa (engl. electronic business, eBusiness). [1] Keskeisessä osassa näissä pyrkimyksissä olivat *web service* -käsite (suom. web-palvelu) ja siihen liittyvät spesifikaatiot. W3C:n (World Wide Web Consortium) määritelmän mukaan web-palvelu on ohjelmistojärjestelmä, joka on suunniteltu tukemaan koneiden välistä yhteensopivaa vuorovaikutusta, joka tapahtuu verkon välityksellä. Web-palvelulla on rajapinta, joka on kuvattu koneella prosessoitavassa muodossa. [10] Web service -spesifikaatiot (WS-*) ovat tällaisten palvelujen ja palvelupohjaisten järjestelmien kehittämiseen suunniteltuja, avoimia ja epäkaupallisia spesifikaatioita. Ensimmäisiä näistä spesifikaatioista olivat W3C:lle vuonna 2000 ehdotettu viestintäprotokolla SOAP ja vuonna 2001 ehdotettu palvelukuvauskieli WSDL (Web Services Description Language) sekä UDDI.orgin vuonna 2000 ehdottama palvelujen julkaisu- ja hakuprotokolla UDDI (Universal Description, Discovery and Integration). [1] UDDI ei ole kuitenkaan saavuttanut

SOAPin ja WSDL:än kaltaista asemaa laajasti hyväksyttynä SOAn elementtinä. [11] [1] Sitten on kehitetty lukuisia muita web service -spesifikaatioita useiden yritysten ja standardointijärjestöjen toimesta.

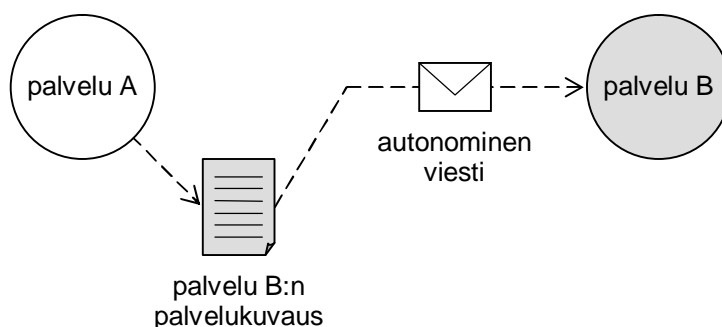
Web service -spesifikaatioita käytettiin aluksi järjestelmien välisten yhteensopivuusongelmien ratkaisemiseen, tekniikka- ja alustaneutraalien kaupallisten palvelujen kehittämiseen sekä julkisten palvelukauppapaikkojen perustamiseen. [1] SOA nähtiin tuolloin pääasiassa arkkitehtuurina, joka koostui ensimmäisten web service -spesifikaatioiden käsitteistä: palveluista, palvelukuvauksista, viesteistä sekä palvelujen julkaisua ja hakua tukevista palvelurekisteristä (engl. service registry) ja palveluvarastosta (engl. service repository). [8] Pian alettiin kuitenkin ymmärtää, että palvelukeskeisyydellä on yritysten välisten käyttötarkoitusten lisäksi merkittäviä käyttömahdollisuuksia yrityksen sisällä. Tämän seurauksena SOA alettiin nähdä ensisijaisesti yritysympäristön arkkitehtuurina, minkä vuoksi SOAa alettiin tarkastella laajemmassa kontekstissa [12], joskin aluksi vielä melko teknisestä näkökulmasta. Tällaisessa kontekstissa SOAsta alettiin käyttää myös termiä *enterprise SOA* (suom. yritys-SOA) [13]. Myöhemmin on alettu korostaa kokonaisvaltaisen palvelukeskeisen ajattelutavan ja liiketoimintalähtöisen lähestymistavan merkitystä SOAn hyötyjen saavuttamisessa. [14] [15] [16] [17] Lisäksi SOAa on alettu tarkastella yhä useammin yhdessä muiden paradigmojen, menetelmien ja tekniikoiden kanssa.

2.2.2 SOAn peruselementit

SOAn perusyksikkö on palvelu, joka on tietyn liiketoiminta-aktiviteetin – kuten ”tarkasta asiakkaan luottotiedot” tai ”tarjoa säädä” – looginen esitys. [2] Palvelun toteutus koostuu implementaatiosta ja kaikesta metadatatista, jota käytetään palvelun kuvaamiseen. Palvelu voidaan implementoida millä tahansa tekniikalla, joka tukee implementaation paljastamista ja kuluttamista palvelustandardien mukaisesti. Palvelun tarjoajan² metadata, kuten palvelurajapinta, viestinnässä käytettävien tietorakenteiden kaavat (engl. schema) ja palvelun käyttöä rajoittavat politiikat, voidaan tarjota

² Palvelun tarjoajalla ja kuluttajalla tarkoitetaan tässä loogisia palveluja ja niiden toteutuksia. Joissain lähteissä näillä termeillä tarkoitetaan palveluja tarjoavia ja kuluttavia henkilöitä ja organisaatioita.

keskitetyksi palvelukuvauksessa³, jonka tarkoitus on tarjota palveluja arvioiville henkilöille ja potentiaalisille palvelun kuluttajille kaikki palvelun arvioinnin ja kuluttamisen kannalta merkittävä informaatio. Palvelut kommunikoivat tyypillisesti viestien avulla. [18] Kommunikaatiossa pyritään usein käyttämään ns. autonomisia viestejä, jotka sisältävät mahdollisimman paljon informaatiota viestin kontekstista ja vähentävät näin palvelujen tarvetta ylläpitää aktiviteettikohtaista tilainformaatiota. [1] Edellä kuvatut, ns. *alkukantaisen SOAn* elementit on esitetty kuvassa 2.2. Näiden perusominaisuuksien – standardien mukaisen viestintäprotokollan ja palvelukuvauksen – avulla eri tekniikoilla toteutetut palvelut voivat olla vuorovaikutuksessa keskenään löyhää kytkentää tukevalla tavalla.

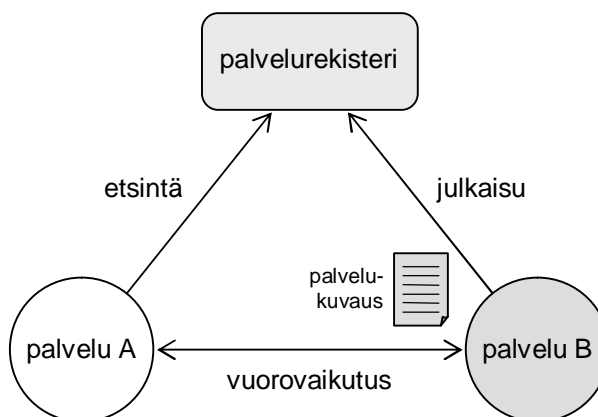


Kuva 2.2. Alkukantaisen SOAn elementit. Palvelu A:lla on pääsy palvelu B:n palvelukuvaukseen, minkä vuoksi sillä on kaikki palvelu B:n kanssa kommunikointiin tarvittava informaatio. [1] Palvelu A toimii kuvassa palvelun kuluttajana ja palvelu B palvelun tarjoajana.

Kuvattujen peruselementtien lisäksi SOAan varhain suunniteltuihin elementteihin kuului myös palvelurekisteri. Tämä toimii palvelun tarjoajia ja kuluttajia yhteen saattavana mekanismina, jonne palvelun tarjoaja voi julkaista palvelukuvauksensa ja josta palvelun kuluttaja voi etsiä ajoaikaisesti tarpeitaan vastaavia palveluja, minkä jälkeen palvelun kuluttaja voi olla suoraan vuorovaikutuksessa palvelun tarjoajan kanssa. [11] [8] Tämän on määrä mahdollistaa palvelupohjaisten sovellusten muodostamisen dynaamisesti löyhää kytkentää edelleen tukevalla tavalla. Michlmayrin et al. [11] mukaan SOA-toteutukset eivät kuitenkaan toteuta tätä dynaamista mallia juuri koskaan, vaan palvelupohjaiset sovellukset muodostetaan staattisesti

³ Palvelun metadatatista muodostuvaa kokonaisuutta kutsutaan joissain lähteissä palvelusopimukseksi (engl. service contract).

suunnitteluaihana. Palvelurekisterin sisältävä SOAn ns. kolmiomalli on esitetty kuvassa 2.3.



Kuva 2.3. SOAn kolmiomalli. Palvelurekisteri toimii palvelun tarjoajan ja palvelun kuluttajan yhteen saattavana mekanismina. Kuvälähteet: [11] [8].

Edellä kuvatut varhaiset SOA-mallit vastaavat läheisesti SOAn ensimmäisten spesifikaatioiden mukaisten elementtien – erityisesti WSDL-rajapintojen, SOAP-viestien ja UDDI-rekisterin – muodostamaa arkkitehtuuria. WSDL- ja SOAP-tekniikoihin perustuvan palvelutyypin rinnalle on sittemmin yleistynyt REST-arkkitehtuuriin (Representational State Transfer) perustuva yksinkertaisempi palvelutyyppi, ns. *RESTful web service*, joka poikkeaa WSDL/SOAP-palvelutyypistä teknisesti monin tavoin. WSDL/SOAP-palvelut voivat säilyttää tilainformaatiota – joskin tätä pyritään usein välttämään – mutta REST-arkkitehtuurissa kommunikaatio on aina tilatonta. WSDL-rajapinnassa voidaan määrittää haluttu määrä metodeja, mutta REST-palvelun rajapinta perustuu HTTP-protokollan (Hypertext Transfer Protocol) vakiomuotoisiin metodeihin (POST, GET, PUT ja DELETE). REST-palvelut voivat käyttää viestinnässä mitä tahansa muotoa, kuten XML- (Extensible Markup Language) tai JSON-muotoa (JavaScript Object Notation). Pautasson, Zimmermannin ja Leymannin [19] mukaan REST-palvelutyyppi soveltuu webin välityksellä tapahtuviin, taktisiin ad hoc -integraatioihin, kuten mashup-sovelluksiin, ja WSDL/SOAP-palvelutyyppi soveltuu puolestaan ammattimaisiin yrityssovellusintegraatioihin, joilla on pitkä elinkaari ja edistykselliset palvelutasovaatimukset.

Tässä kohdassa kuvattujen SOAn peruselementtien lisäksi palvelupohjaisten sovellusten muodostamisessa käytetään lukuisia muita elementtejä, minkä lisäksi SOA toteutetaan usein varsin monimutkaisen infrastruktuurin avulla. SOAn hallintaan liittyvät aiheet ovat myös merkittävä osa SOAn ongelma-alueita. Näitä elementtejä ja käsitteitä kuvataan SOA-malleissa, jotka ovat tämän tutkimuksen varsinaisina kohteina.

2.2.3 Palvelukeskeisyys

Palvelukeskeisyys on SOAn myötä yleistynyt termi, mutta sitä voidaan tarkastella myös yleisenä ilmiönä, joka ilmenee erilaisissa ympäristöissä. Esimerkiksi paikallisesti toimivat yritykset voidaan nähdä palvelukeskeisinä yksikköinä, sillä ne ovat itsenäisiä toiminnallisia järjestelmiä, jotka tarjoavat uudelleenkäytettäviä palveluja muille yrityksille ja käyttävät vuorovaikutuksessa tiettyjä yhteisiä, standardoituja välineitä, kuten kieltä, valuuttaa ja kommunikaatiomuotoja. [1] [20] SOAn kontekstissa palvelukeskeisyydellä on kuitenkin täsmällisempi merkitys, ja sen tukemiseksi on kehitetty muun muassa palvelukeskeisiä mallinnusmenetelmiä. Palvelukeskeisyyttä voidaan soveltaa sekä liiketoimintalogiikkaan että sovelluslogiikkaan siten, että liiketoimintoja mallinnetaan liiketoimintapalveluina, jotka implementoidaan näitä vastaavina sovelluspalveluina. Palvelujen yhdistelemiseen tai ohjaamiseen käytettävä liiketoimintalogiikka voidaan kuvata lisäksi erillisinä kuvauksina, kuten prosessikuvauksina, liiketoimintasääntöinä (engl. business rule) tai tapahtumasääntöinä (engl. event rule). Erottamalla nämä aspektit – liiketoimintapalvelujen määritykset, implementaatiot ja koordinointi – loogisesti erillisiksi kerroksiksi voidaan edistää aspektien välistä riippumattomuutta ja kokonaisratkaisun joustavuutta ja ketteryyttä. [1] [21]

Palvelukeskeisyys määritellään usein periaatteina, jotka ohjaavat palvelujen ja palvelupohjaisten järjestelmien suunnittelua. Taulukossa 2.1 esitetään Thomas Erlin määrittämät, laajasti omaksutut palvelukeskeisyyden periaatteet, jotka koskevat erityisesti ohjelmistojärjestelmien suunnittelua.

Taulukko 2.1. Palvelukeskeisyyden periaatteet. [1]

Periaate	Määritelmä
Löyhä kytkentä (engl. loose coupling)	Palveluilla on suhde, jossa on mahdollisimman vähän riippuvuuksia. Tämä suhde edellyttää vain, että palvelut pysyvät tietoisina toisistaan.
Palvelusopimus (engl. service contract)	Palvelut noudattavat kommunikaatiosopimusta, jonka määrittävät yksi tai useampi palvelukuvaus ja näihin liittyvät dokumentit.
Autonomia (itsehallinto, engl. autonomy)	Palveluiden sisältämä (engl. encapsulate) logiikka on niiden omassa hallinnassa.
Abstrahointi (engl. abstraction)	Palvelut piilottavat ulkopuolisilta kaiken logiikan, jota ei ole kuvattu palvelukuvauksessa.
Uudelleenkäytettävyys (engl. reusability)	Logiikka on jaettu palveluihin tavalla, joka edistää uudelleenkäyttöä.
Yhdisteltävyys (engl. composability)	Palveluja voidaan koordinoida ja yhdistellä komposiittipalveluiksi (engl. composite service).
Tilattomuus (engl. statelessness)	Palvelut säilyttävät mahdollisimman vähän aktiviteettikohtaista informaatiota.
Löydettävyys (engl. discoverability)	Palvelut suunnitellaan ulospäin kuvaaviksi, jotta ne voidaan löytää ja niitä voidaan arvioida saatavilla olevien hakumeکانismien avulla.

2.2.4 SOAn määritelmät

SOA on ohjelmistoarkkitehtuurin tavoin termi, jonka määrittäminen ei ole yksiselitteistä, koska käsitettä voidaan tarkastella varsin monesta näkökulmasta. Seuraavassa on kolmen tahon esittämät SOAn määritelmät:

OASIS

Palvelukeskeinen arkkitehtuuri on paradigma, jota käytetään hajautettujen, mahdollisesti eri omistusalueiden hallinnassa olevien kyvykkyyksien (engl. capability) järjestämiseen ja käyttämiseen. Se tarjoaa yhtenäisen tavan kyvykkyyksien tarjoamiseen, etsimiseen, vuorovaikutukseen ja käyttämiseen sellaisten vaikutusten tuottamiseksi, jotka ovat mitattavien alkuehtojen ja odotusten kanssa yhdenmukaisia. [18]

SOA Manifesto

Palvelukeskeisyys on paradigma ja kehys, joka ohjaa tekemistä.

Palvelukeskeinen arkkitehtuuri (SOA) on arkkitehtuurityyppi, joka seuraa palvelukeskeisyyden soveltamisesta. [22]

The Open Group

Palvelukeskeinen arkkitehtuuri (SOA) on arkkitehtuurityyli, joka tukee palvelukeskeisyyttä.

Palvelukeskeisyys on tapa ajatella palvelujen, palvelupohjaisen kehityksen ja palvelujen ulostulojen (engl. outcome) muodossa.

(...) [2]

The Open Groupin määritelmä jatkuu palvelun ja SOA-arkkitehtuurityylin tarkemmilla määrittelyillä, joissa määritellään muun muassa, että palvelut heijastavat tosimaailman liiketoiminta-aktiviteetteja; SOA-infrastruktuurissa on suositeltavaa käyttää avoimia standardeja; SOA-implementaatiot ovat ympäristökohtaisia; ja SOA edellyttää palveluesitysten ja -implementaatioiden voimakasta hallintoa (engl. governance). [2]

Edellä esitetyt määritelmät ovat luonteeltaan abstrakteja ja periaatteellisia, ja niistä ilmenee palvelukeskeisen paradigman oleellinen osa SOAa ohjaavana tekijänä. Sen sijaan ne eivät juuri pyri kuvaamaan SOAn rakenteellisia seikkoja.

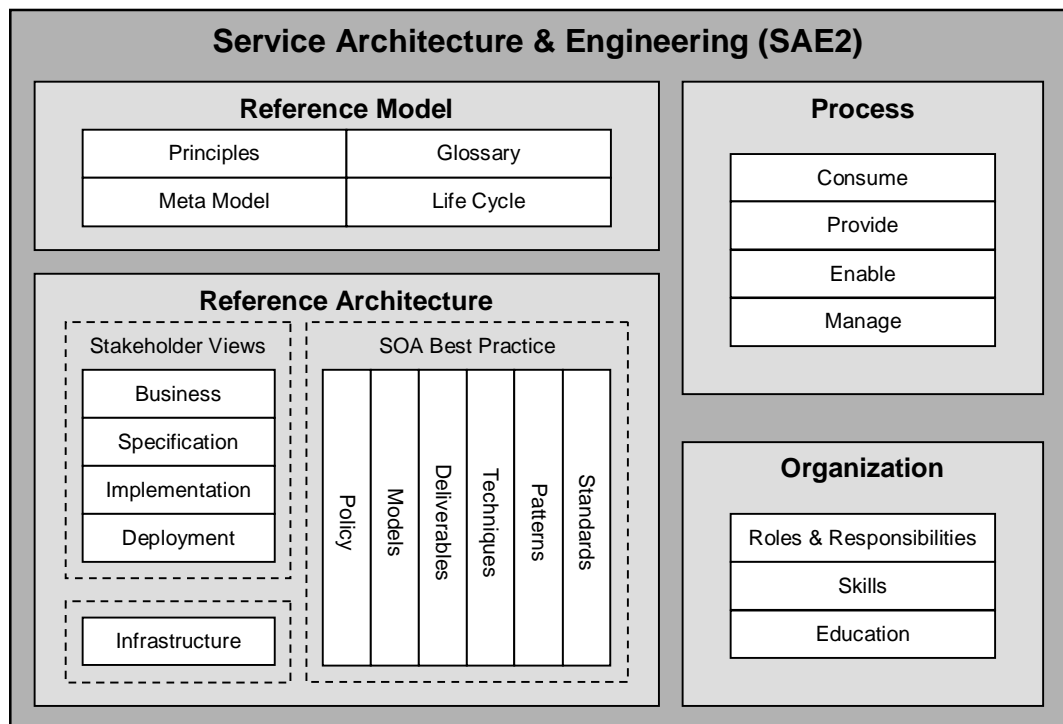
2.2.5 SOAn hyödyt

Näkemykset SOAn hyödyistä ovat muuttuneet SOAn kehityksen ja käyttötapojen muuttumisen myötä. Palvelukeskeisyydellä tavoiteltiin aluksi teknisiä hyötyjä, kuten järjestelmien integroitavuutta ja joustavuutta sekä resurssien uudelleenkäytettävyyttä [9]. Nykyään SOAn merkittävimpänä hyötynä SOA-asiantuntijat mainitsevat Beckerin, Buxmannin ja Widjajan [3] mukaan useimmiten liiketoimintaketteryuden. SOAn nähdään mukautuvan hyvin strategian ja liiketoiminnan muutoksiin, sillä SOAssa korkean tason liiketoimintalogiikan kuvaukset on erotettu palveluista ja näiden implementaatioista, ja palvelupohjaiset prosessit ja järjestelmät ovat muunneltavissa nopeasti vastaamaan uusia vaatimuksia. [21] [23] Muina SOAn hyötyinä SOA-asiantuntijat mainitsevat mm. prosessioptimoinnin eli prosessien automatisoinnin ja hallinnan; ohjelmistokehitystä helpottavan standardoinnin; yrityksen sovellusten

konsolidoinnin; informaation paremman laadun ja saatavuuden; sekä IT:n asettamisen linjaan liiketoiminnan kanssa. Palvelujen uudelleenkäytettävyyttä pidetään myös tärkeänä SOAn hyötynä, mutta useiden yritysten kokemusten mukaan merkittävän uudelleenkäytön saavuttaminen on vaikeaa. [3] SOAn hyötyjen saavuttamista ja SOAn käyttöönottoa vaikeuttavia haasteita ovat yritysten kokemusten mukaan mm. rahoituksen järjestäminen, taloudellisen hyödyn osoittaminen, hallinnon järjestäminen, yhteisen ymmärryksen saavuttaminen SOA-paradigmasta [3], yrityksen sisäinen muutosvastarinta, olemassa olevia liiketoimintaprosesseja kuvaavien prosessimallien puuttuminen sekä SOA-työkalujen kehittymättömyys [24].

2.3 SOA-mallityypit

SOAn omaksumiseen sisältyy monia osa-alueita, joiden tukemiseksi on esitetty malleja ja muita apuvälineitä. Näitä ovat arkkitehtuurillisten mallien lisäksi mm. hallintomallit, kypsyysmallit (engl. maturity model) ja palvelukeskeiset mallinnusmenetelmät. Esimerkkinä moniosaisesta SOAn omaksumista tukevasta kehyksestä voidaan mainita Everware-CBDI:n kaupallinen referenssikehys CBDI-SAE (CBDI Service Architecture & Engineering), jonka osa-alueet ovat referenssimalli (Reference Model), referenssiarkkitehtuuri (Reference Architecture), prosessi (Process) ja organisaatio (Organization) (kuva 2.4). Tässä tutkielmassa keskitytään SOAa kuvaaviin arkkitehtuurillisiin malleihin, joiksi voidaan kuitenkin lukea useita mallityyppejä. Nämä poikkeavat toisistaan muun muassa abstraktiotasoon ja kattavuuteen nähden, minkä lisäksi erilaisilla mallityypeillä on erilainen käyttötarkoitus SOAn toteuttamisessa.



Kuva 2.4. CBDI-SAE -referenssikehys. [25]

2.3.1 Mallityyppien nimitykset

Mallityypeistä käytettävät nimitykset ovat osittain vakiintumattomia – esimerkiksi referenssimalliksi ja -arkkitehtuuriksi kutsutaan varsin erityyppisiä malleja, mikä voi aiheuttaa sekaannusta mallien ymmärtämisessä ja käyttämisessä. [26] [27] Tässä tutkielmassa tarkasteltavien mallien tyypit ovat referenssimalli, ontologia ja referenssiarkkitehtuuri, minkä lisäksi joitain malleja kutsutaan myös kehykseksi. Näiden mallityyppien kuvaukset – jotka ovat yhdenmukaisia tässä tutkielmassa tarkasteltujen mallien kanssa – on esitelty lyhyesti taulukossa 2.2.

Taulukko 2.2. SOA-mallityyppien kuvaukset.

Mallityyppi	Kuvaus
Referenssi-malli	Referenssimalli on abstrakti malli, joka koostuu tietyn ongelma-alueen pienimmästä mahdollisesta yhteisestä joukosta käsitteitä, periaatteita ja suhteita. Sitä käytetään tietyn ympäristön yksikköjen merkittävien suhteiden ymmärtämiseen. [18]
Ontologia	Ontologia on tietyn aihealueen käsitteellinen malli, joka on määritelty tarkasti ja formaalisesti, ja se on jaettu osallistujien kesken. Ontologiaa voidaan käyttää sekä ihmisten että ohjelmistojen välisessä kommunikoinnissa. [28] [29]
Referenssi-arkkitehtuuri	Referenssiarkkitehtuuri on tietyn tietojärjestelmätyypin abstrakti ja geneerinen arkkitehtuuri, jota voidaan käyttää konkreettisempien arkkitehtuurien suunnittelun perustana. [30]
Kehys	Kehys on kokoelma malleja tai muita apuvälineitä, jotka tukevat yhdessä tiettyä tarkoitusta.

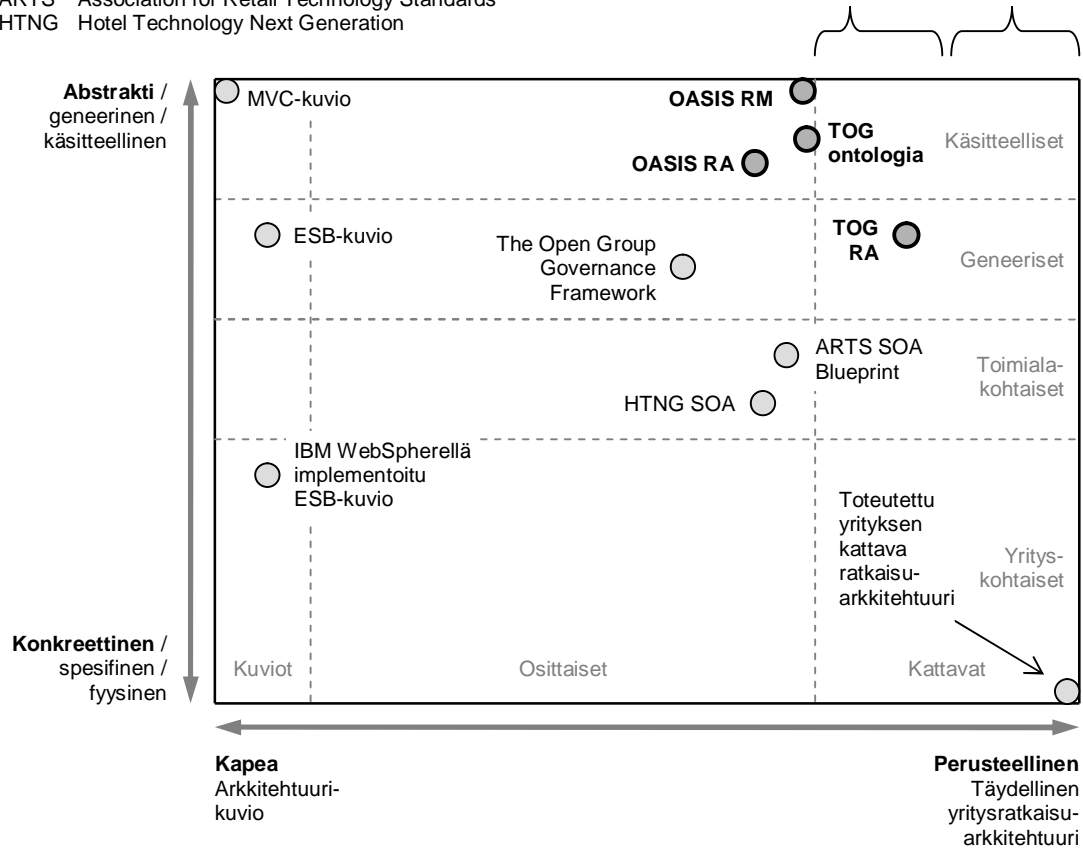
2.3.2 Mallien abstraktiotaso ja kattavuus

SOA-malleja voidaan tarkastella monesta näkökulmasta. Yleisen ymmärtämisen kannalta mallien merkittäviä ominaisuuksia ovat abstraktiotaso ja kattavuus. SOAa voidaan kuvata eri abstraktiotasoilla, ja yritys voi myös noudattaa useita eritasoisia malleja. Ahmed Fattahin [27] esittämän lähestymistavan mukaan SOA-mallit voidaan jakaa abstraktiotasoon nähden käsitteellisiin, geneerisiin, toimialakohtaisiin ja yritysکوhtaisiin malleihin – joita kaikkia Fattah tosin kutsuu yleisesti referenssiarkkitehtuureiksi. Kattavuuteen nähden mallit voidaan Fattahin mukaan jakaa kuvioihin (engl. pattern), osittaisiin malleihin ja kokonaisvaltaisiin (engl. end-to-end) malleihin. Standardointijärjestöt OASIS ja The Open Group käyttävät tätä lähestymistapaa esittämiensä mallien vertailuun (kuva 2.5).

Lyhenteet

- MVC Model–View–Controller
- RM referenssimalli
- TOG The Open Group
- RA referenssiarkkitehtuuri
- ESB enterprise service bus, suom. palveluväylä
- ARTS Association for Retail Technology Standards
- HTNG Hotel Technology Next Generation

- Kattava tekninen referenssiarkkitehtuuri (vain IT-aspekti)
- Kattava referenssiarkkitehtuuri (IT- ja liiketoiminta-aspektit)

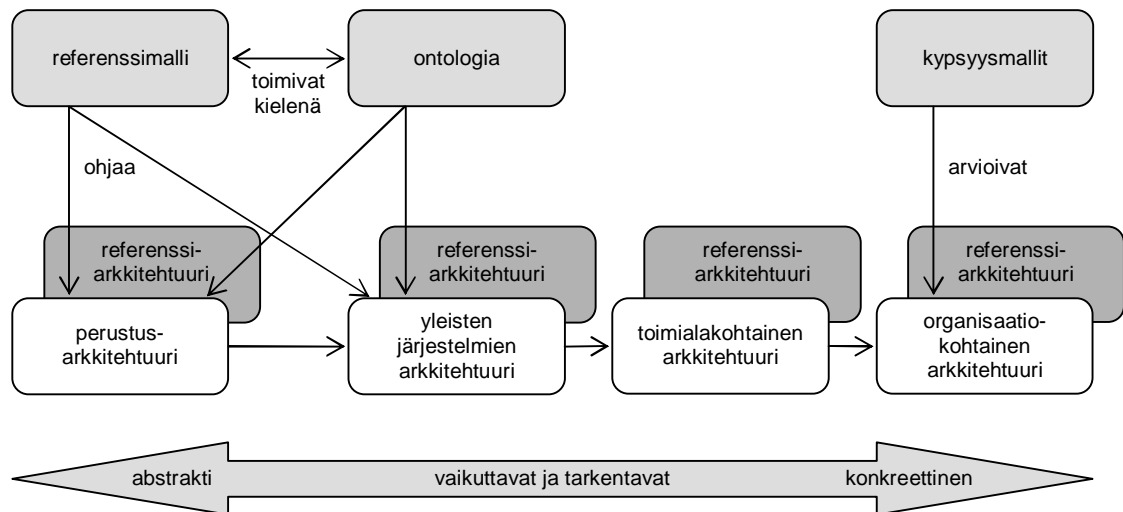


Kuva 2.5. SOA-mallien abstraktiotason ja kattavuuden vertailu. Kuvälähde: [31]. Tässä tutkielmassa käsiteltävät mallit on merkitty kuvaan vahvistettuina. Tämän tutkielman vertailun mukaan OASISin referenssiarkkitehtuuri edustaa yleiseltä näkökulmaltaan käsiteltävistä malleista laajinta näkökulmaa mutta sijoittuu tekniseen kattavuuteen nähden The Open Groupin ontologian ja referenssiarkkitehtuurin väliin.

2.3.3 Mallityyppien merkitys järjestelmän toteuttamisessa

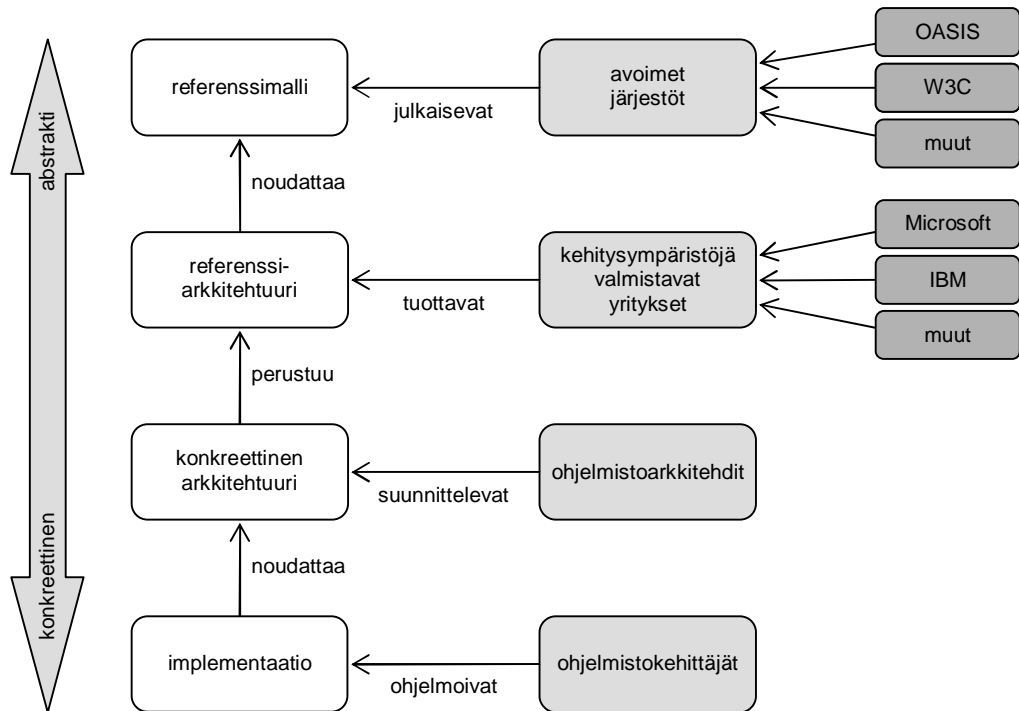
Mallityypeillä on erilaiset käyttötarkoitukset järjestelmän, kuten SOAn tai yritysarkkitehtuurin, toteuttamisessa. Tämä käyttötarkoitus on yhteydessä mallin abstraktiotasoon. TOGAF-yritysarkkitehtuurikehyksen (The Open Group Architecture Framework) [7] lähestymistavan mukaan yrityksen arkkitehtuuria voidaan tarkastella neljällä abstraktiotasolla, jotka muistuttavat läheisesti Fattahin lähestymistavassa käytettyjä abstraktiotasoja. TOGAFin abstraktiotasot ovat perustusarkkitehtuuri, yleisten järjestelmien arkkitehtuuri, toimialakohtainen arkkitehtuuri sekä

organisaatiokohtainen arkkitehtuuri. Yritys voi kehittää arkkitehtuuriaan jokaisella tasolla ja hyödyntää eri tasojen referenssiarkkitehtuureja sekä muita malleja (kuva 2.6).



Kuva 2.6. Mallien yhteydet yrityksen arkkitehtuurin abstraktiotasoihin. Kuvälähde: [31].

Myös julkaisutyypeillä on erilaiset merkitykset SOAn toteuttamisen kannalta. Julkaisutyypeillä tarkoitetaan tässä esim. standardeja, kaupallisia malleja, yrityksen sisäisiä malleja sekä tieteellisiä malleja. Junin et al. [32] mukaan SOAn referenssimallit ovat yleensä avointen standardointijärjestöjen julkaisemia, kun taas SOAn referenssiarkkitehtuureja julkaisevat pääasiassa ohjelmistokehitysympäristöjen valmistajat (kuva 2.7). Tämä jako on kuitenkin karkea, sillä erityyppisiä malleja voivat julkaista useat tahot. Esimerkiksi referenssiarkkitehtuureja voivat julkaista myös standardointijärjestöt, minkä lisäksi yritykset voivat kehittää omia referenssiarkkitehtuureja yrityksen sisäiseen käyttöön. Kuva 2.7 kuvaa kuitenkin suuntaa antavasti eri tahojen esittämien mallien merkityksen SOAn toteuttamisessa.



Kuva 2.7. Eri tahojen esittämien mallien merkitys SOAn toteuttamisessa. [32]

Luku 3

SOAn perusominaisuuksia kuvaavat mallit

Tässä luvussa tarkastellaan kahta mallia, jotka kuvaavat SOAn ymmärtämisen ja toteuttamisen kannalta keskeisimpiä seikkoja. Nämä mallit, OASISin SOA-referenssimalli ja The Open Groupin SOA-ontologia, eivät pyri kuvaamaan SOAa kattavasti vaan keskittyvät valittujen käsitteiden ja näiden suhteiden kuvaamiseen. Ne ovat hyvin käsitteellisiä malleja. Käsitteellisiä malleja voidaan yleisesti tarkastellen käyttää tietoteknisten järjestelmien kehittämisessä esim. ongelma-alueen ymmärtämisen apuvälineinä, kehittäjien ja käyttäjien yhteisinä kommunikaatiovälineinä, järjestelmän vaatimusten määrittämiskeinona sekä järjestelmän suunnittelemisen ja kehittämisen perustana [33]. Tässä luvussa tutkittavat mallit kuvaavat useita samoja, palveluun liittyviä aiheita mutta edustavat varsin erilaisia näkökulmia ja poikkeavat myös muun muassa kuvaustavoiltaan huomattavasti toisistaan.

3.1 OASISin referenssimalli

OASISin referenssimalli on tässä tutkielmassa tarkasteltavista SOA-malleista abstraktein. Se kuvaa palvelu-käsitettä ja siihen läheisesti liittyviä, palvelujen dynamiikkaa ja metatasoa koskevia käsitteitä. Referenssimallin soveltamista tarkastellaan tapausesimerkin avulla: referenssimallia käytetään perustana Yhdysvaltain oikeusministeriön GRA-kehyksessä (Global Reference Architecture Framework), jolla pyritään tukemaan SOAn omaksumista Yhdysvaltain oikeusyksiköissä.

3.1.1 Yleiskatsaus

Vuonna 2006 julkaistu OASISin referenssimalli on hyvin abstrakti malli, joka pyrkii kuvaamaan SOAn olemusta ja palvelukeskeistä ympäristöä. Sen tavoitteena on myös toimia yhteisenä ymmärryksenä SOAsta ja semantiikkana, jota voidaan käyttää eri implementaatioissa ja niiden kesken. [18] OASIS määrittelee referenssimallin seuraavasti:

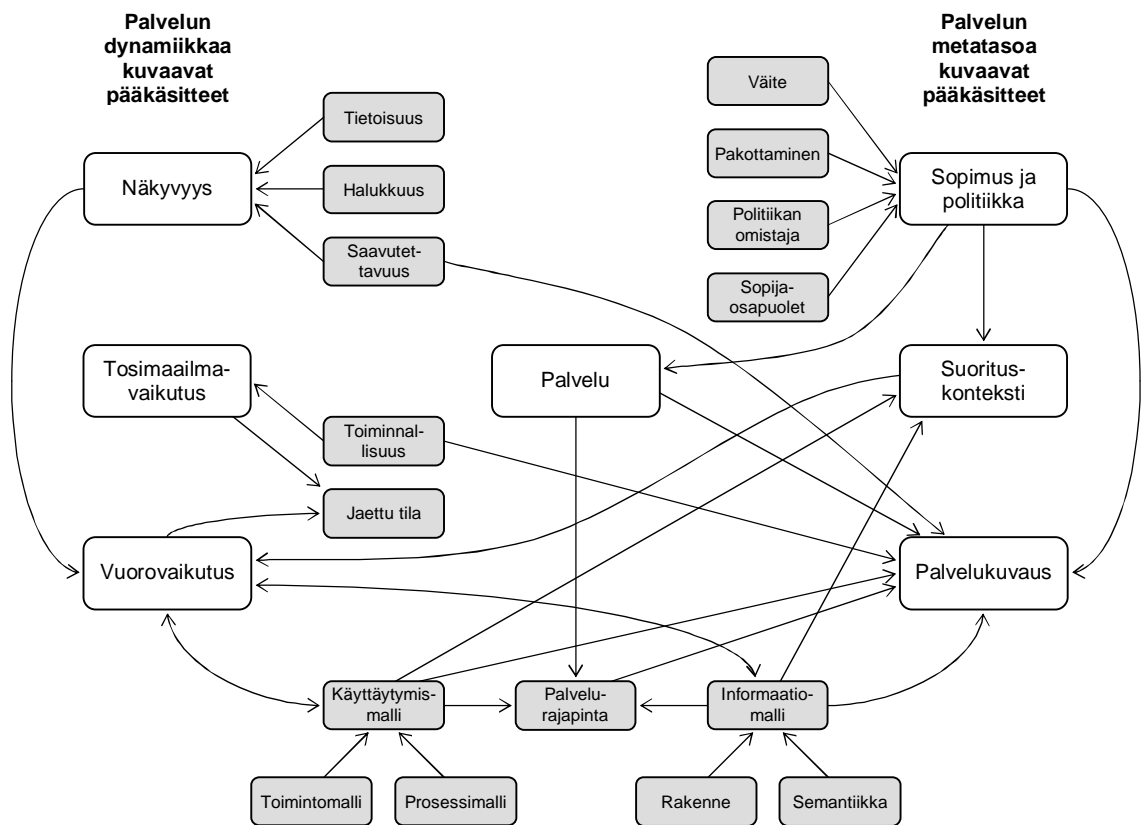
Referenssimalli on abstrakti kehys, jota käytetään tietyn ympäristön yksikköjen merkittävien suhteiden ymmärtämiseen. (...) Referenssimalli koostuu tietyn ongelma-alueen pienimmästä mahdollisesta yhteisestä joukosta käsitteitä, periaatteita ja suhteita. Referenssimalli on riippumaton standardeista, tekniikoista, implementaatioista tai muista konkreettisista yksityiskohdista. [18]

Malli keskittyy ohjelmistoarkkitehtuurin alueeseen mutta on siinä määrin yleinen, että se voi soveltua jossain määrin myös muihin palveluympäristöihin. Referenssimallin käyttömahdollisuuksina mainitaan standardien ja spesifikaatioiden kehittäminen, palvelukeskeisten arkkitehtuurien kehittäminen, kouluttaminen sekä SOAn selittäminen. Malli on suunnattu mm. arkkitehteille, kehittäjille, standardiarkkitehteille ja analyytikoille, päättäjille ja käyttäjille. Referenssimalli esitetään spesifikaatiossa – kuvien 2.5, 2.6 ja 2.7 kanssa yhdenmukaisesti – kaikkein abstrakteimpana ja yleiskäyttöisimpänä mallina, joka toimii perustana muille SOA-malleille ja arkkitehtuureille. Spesifikaatiossa esitetään yhdenmukaisuusohjeistus (engl. conformance guidelines), jonka mukaan arkkitehtuuri on yhdenmukainen referenssimallin kanssa, mikäli siitä voidaan tunnistaa ja osoittaa, kuinka referenssimallissa esitetyt pääkäsitteet ilmenevät kyseisessä arkkitehtuurissa. [18]

3.1.2 Mallin sisältö

SOAa tarkastellaan mallissa OASISin SOA-määritelmän mukaisesti paradigmana, jota käytetään hajautettujen, mahdollisesti eri omistusalueiden hallinnassa olevien kyvykkyyksien järjestämiseen ja käyttämiseen. Kyvykkyydet ovat asioita, joita ihmiset ja organisaatiot luovat ratkaistakseen liiketoiminnassa kohtaamiaan ongelmia [18]. Spesifikaation mukaan SOAn keskeinen arvo on se, että SOA tarjoaa voimakkaan kehityksen tarpeiden (engl. need) ja kyvykkyyksien yhteensovittamiseen sekä

kyvykkyyksien yhdistelemiseen tarpeiden tyydyttämiseksi. Mallissa keskitytään vahvasti palvelu-käsitteeseen ja tähän läheisesti liittyviin käsitteisiin. Mallin pääkäsitteet on palvelua lukuun ottamatta jaettu palvelun dynamiikkaa ja metatasoa kuvaaviin käsitteisiin. Pääkäsitteiden lisäksi malli sisältää useita muita käsitteitä, joita käytetään pääkäsitteiden kuvaamisen apuna. Käsitteet, niiden väliset suhteet ja periaatteet on esitetty mallissa käsittekarttoina ja niitä selittävinä sanallisina kuvauksina. Mallin graafinen tiivistelmä, joka on muodostettu yhdistämällä kaikki mallin käsittekartat, on esitetty kuvassa 3.1. Referenssimallin pääkäsitteet on kuvattu taulukoissa 3.1 ja 3.2.



Kuva 3.1. Referenssimallin graafinen tiivistelmä. Pääkäsitteet on esitetty valkoisella taustavärillä ja apukäsitteet harmaalla. Nuolten osoittamat käsitteet ovat jollain tavalla riippuvaisia niistä käsitteistä, joista nuolet lähtevät. Kaavio on johdettu referenssimallista [18].

Taulukko 3.1. Palvelu-käsitteen ja palvelun dynamiikkaa kuvaavien pääkäsitteiden kuvaukset. Pääkäsitteisiin liittyvät apukäsitteet on merkitty lihavoidulla fontilla. Kuvaukset perustuvat referenssimalliin [18].

Pääkäsite	Kuvaus
Palvelu	Palvelu on mekanismi, jolla sallitaan pääsy yhteen tai useampaan kyvykkyyteen. Pääsy kyvykkyyteen tarjotaan määrätyn rajapinnan avulla ja se noudattaa palvelukuvauksessa esitettyjä rajoituksia ja politiikkoja (engl. policy).
Näkyvyys (engl. visibility)	Näkyvyys on palvelun kuluttajan ⁴ ja tarjoajan välinen suhde, joka toteutuu kun ne kykenevät keskinäiseen vuorovaikutukseen. Näkyvyyden edellytyksiä ovat tietoisuus (engl. awareness), halukkuus (engl. willingness) ja saavutettavuus (engl. reachability).
Vuorovaikutus	Vuorovaikutus on kyvykkyyden käyttämistä suorittamalla palvelun toimintoja (engl. action), mikä saavutetaan tavallisesti viestien vaihtamisen avulla. Palvelun kanssa vaihdettavan informaation rakenne ja semantiikka on määrätty palvelukuvauksen informaatiomallissa . Palvelukuvaus sisältää myös palvelun käyttäytymismallin (engl. behavior model), johon sisältyy toimintomalli (engl. action model), jossa on kuvattu palvelun toiminnot, sekä prosessimalli , jossa on kuvattu vuorovaikutukseen liittyvien toimintojen ja tapahtumien ajalliset suhteet ja ominaisuudet.
Tosimaailma-vaikutus (engl. real world effect)	Kyvyn käyttämisen tarkoitus on toteuttaa yksi tai useampi tosimaailmavaikutus. Tämä vaikutus on joko informaation palauttaminen palvelun kuluttajalle tai tilan muutos yhdessä tai useammassa yksikössä, jonka vuorovaikutuksessa olevat osapuolet jakavat. Tällaisen yksikön tilaa kutsutaan jaetuksi tilaksi (engl. shared state).

⁴ Palvelun kuluttajalla ja tarjoajalla tarkoitetaan OASISin referenssimallissa ihmisiä ja organisaatioita.

Taulukko 3.2. Palvelun metatasoa kuvaavien pääkäsitteiden kuvaukset. Pääkäsitteisiin liittyvät apukäsitteet on merkitty lihavoidulla fontilla. Kuvaukset perustuvat referenssimalliin [18].

Pääkäsite	Kuvaus
Palvelukuvaus	Palvelukuvaus sisältää palvelun käyttämiseen tarvittavan informaation. Tarvittava informaatio riippuu kontekstista ja osapuolien tarpeista. Tärkeää informaatiota on usein erityisesti palvelun saavutettavuutta , toiminnallisuutta (engl. functionality), rajoituksia ja politiikkoja , informaatiomallia ja käyttäytymismallia koskeva tieto. Palvelun rajapinta on osa palvelukuvausta ja väline, joka mahdollistaa vuorovaikutuksen palvelun kanssa. Se sisältää palvelun informaatiomallia ja toiminnallista mallia koskevaa tietoa.
Sopimus ja politiikka (engl. contract and policy)	Politiikka on väite (engl. assertion), jonka politiikan omistaja asettaa rajoitukseksi tai ehdoksi omistamansa yksikön käyttämiselle. Politiikka voidaan pakottaa (engl. enforcement). Sopimus edustaa sen sijaan kahden tai useamman sopijaosapuolen (engl. parties in agreement) välistä yhteispäätöstä (engl. agreement). Sopimukset ovat politiikan tavoin väitteitä , jotka koskevat palvelun käyttämistä ja jotka voidaan pakottaa .
Suorituskonteksti (engl. execution context)	Palveluvuorovaikutuksen suorituskonteksti on niiden teknisten elementtien ja liiketoimintaelementtien muodostama joukko, jotka muodostavat polun kyvykkyyksien omistajien ja tarpeiden omistajien välillä ja mahdollistavat näin vuorovaikutuksen osapuolten välillä. Suorituskontekstiin sisältyviä elementtejä ovat mm. infrastruktuuri-elementit, prosessiyksiköt (engl. process entity), politiikat ja sopimukset . Siihen sisältyvät myös palvelun käyttäytymismallin ja informaatiomallin kuvaukset. Suorituskonteksti sallii palvelun tarjoajien ja kuluttajien olla vuorovaikutuksessa keskenään ja tarjoaa päätöskohdan mahdollisesti voimassa oleville politiikoille ja sopimuksille.

3.1.3 Mallin soveltaminen

OASISin referenssimallin eräs huomattava sovelluskohde on Yhdysvaltain oikeusministeriön GRA-kehys, joka perustuu vahvasti OASISin referenssimalliin ja laajentaa sitä. GRA:n on kehittänyt U.S. Department of Justice's Global Justice Information Sharing Initiative (Global) yhteistyökumppaneineen. GRA on Globalin määritelmän mukaan SOAn merkittävien komponenttien ja niiden välisten suhteiden ymmärtämistä tukeva abstrakti kehys, jonka esittämät yhteiset käsitteet ja määritelmät

toimivat perustana oikeus- ja turvallisuusyhteisöjen yhtenäisten SOA-implemентаatioiden kehittämiseksi. Global pyrkii edistämään GRA:n avulla Yhdysvaltain oikeusyksikköjen välistä informaation jakamista ja tietojärjestelmien yhteensopivuutta. [34] GRA sisältää käsitteellisen kehyksen lisäksi myös muita osioita ja resursseja, kuten palvelujen tunnistamista ja suunnittelemista tukevan metodologian, palvelujen kehittämistä tukevia ohjeita, referenssipalvelumäärittäviä ja muita teknisiä resursseja. [35] [34] GRA ei kuitenkaan nykyisessä muodossaan sisällä referenssiarkkitehtuuria, jota voitaisiin käyttää tietojärjestelmien suunnittelun arkkitehtuurillisena perustana.

GRA:ssa voidaan havaita useita käyttötapoja sen sisältämälle käsitteelliselle mallille ja siten myös tämän perustana toimivalle OASISin referenssimallille. Ensiksi, mallia käytetään kehyksen sisällä yhteisenä käsitteellisenä perustana eri osioiden kesken. Dokumenteissa SOAa tarkastellaan samojen käsitteiden avulla ja käytetään niistä yhteisiä termejä. Toiseksi, mallia käytetään yksittäisten oikeusorganisaatioiden tukemiseen. Mallilla pyritään edistämään SOAn ymmärtämistä ja omaksumista organisaatioissa, minkä lisäksi sitä voidaan käyttää tietojärjestelmien suunnittelun käsitteellisenä perustana. Kolmanneksi, mallia käytetään oikeusorganisaatioiden keskinäisten toimintojen tukemiseen. Yhteisellä käsitteistöllä pyritään tukemaan oikeusorganisaatioiden välistä, tietojärjestelmiä ja informaation vaihtamista koskevaa kommunikointia. [34] Mallilla pyritään myös edistämään tietojärjestelmien yhteensopivuutta ja informaation vaihtamista, joskin OASISin abstraktin referenssimallin merkitys tässä pyrkimyksessä lienee hyvin vähäinen.

3.1.4 Havainnot

OASISin referenssimalli keskittyy suppeaan aihealueeseen – palveluun ja siihen läheisesti liittyviin käsitteisiin – eikä se sisällä teknisiä yksityiskohtia. Mallin voisi tämän vuoksi olettaa olevan helposti ymmärrettävä, mutta mallin abstraktius asettaa sitä vastoin huomattavia haasteita mallin ymmärtämiseksi, sillä käsitteellinen etäisyys mallin ja konkreettisten elementtien ja ilmiöiden välillä muodostuu paikoin hyvin suureksi. Abstraktiudesta johtuvista haasteista mainitaan myös malliin vahvasti perustuvan GRA:n dokumentissa [34].

Referenssimalli kuvaa palvelujen dynamiikkaa ja metatasoa, jotka ovat itsessään abstrakteja aiheita. Osa mallin kuvaamista käsitteistä on erittäin voimakkaita abstraktioita, kuten näkyvyys, tosimaailmavaikutus ja suorituskonteksti. Näiden käsitteiden yhteys konkreettisiin arkkitehtuuri-elementteihin ja ilmiöihin on hyvin epäsuora. Malli kuvaa myös joitain melko konkreettisiä teknisiä elementtejä, kuten palvelua, palvelun rajapintaa, sopimusta ja politiikkaa. Malli ei kuitenkaan erota rakennetta kuvaavia käsitteitä selvästi muista käsitteistä, mikä vaikeuttaa mallin hahmottamista. Tarkastelun abstraktiotaso on kuitenkin mallissa yleisesti varsin korkea, sillä se on erityisesti suunniteltu mahdollisimman yleiseksi malliksi.

Eräs seikka, joka saattaa myös asettaa haasteita mallin ymmärtämiselle, on mallin epätarkka esitysmuoto. Käsitteiden välisiä suhteita ei ole nimetty, vaan ne on selitetty kaavioita seuraavissa sanallisissa kuvauksissa. Suhteet ovat kuitenkin paikoin monimutkaisia ja sanalliset kuvaukset epätarkkoja. Tämän vuoksi esimerkiksi palvelun informaatiomallin ja käyttäytymismallin suhteet palvelun rajapintaan ja palvelukuvaukseen jäävät osittain epäselviksi.

Yhteisen SOA-käsitteistön ja kielen tarjoaminen yhteisöille on referenssimallin ilmeinen hyöty. Toinen keskeinen hyöty on SOAn ymmärtämisen tukeminen, sillä malli tarjoaa teknisistä toteutuksista riippumattoman kuvauksen SOAn peruspiirteistä. Mallin korkea abstraktiotaso ja epätarkka esitysmuoto asettavat kuitenkin haasteita näille käyttötarkoituksille. Referenssimalli on suunniteltu tukemaan myös SOAn toteuttamista siten, että se kuvaa SOAn keskeiset piirteet, joita SOA-toteutusten tulee tukea. Mallin hyöty tässä tarkoituksessa lienee kuitenkin vähäinen, sillä mallin perusluonteisuuden vuoksi useat mallin käsitteistä lienevät itsestään selviä ominaisuuksia nykyaikaisissa SOA-toteutuksissa.

3.2 The Open Groupin ontologia

The Open Groupin ontologia kuvaa palvelupohjaisten sovellusten keskeisiä elementtejä ja niiden suhteita formaalisesti OWL DL -kielellä (Web Ontology Language, Description Logic). Se pyrkii tukemaan SOAn ymmärtämistä, mutta sen koneella

prosessoitavan muodon vuoksi sitä on mahdollista käyttää myös metamallina ja metadatana sovellusten kehittämisessä. Ontologian soveltamistapojen ohessa esitetään lyhyesti tapausesimerkki, jossa ontologia on omaksuttu osaksi S-RAMP (SOA - Repository Artifact Model and Protocol) -spesifikaatiota, joka määrittää yhteisen tietomallin SOA-artefaktivarastoille.

3.2.1 Yleiskatsaus

The Open Groupin ontologia on vuonna 2010 julkaistu standardi, joka OASISin referenssimallin tavoin kuvaa SOAn peruskäsitteitä ja pyrkii tukemaan yhteistä semantiikkaa ja ymmärrystä SOAsta. Ontologia pyrkii erityisesti tukemaan liiketoiminta- ja IT-yhteisöjen yhdenmukaisuutta (engl. alignment) ja keskinäistä yhteisymmärrystä. Ontologian tavoitteena mainitaan lisäksi SOAn omaksumisen helpottaminen. [36] Standardi on osa The Open Groupin pyrkimyksiä edistää ns. Rajatonta Informaatiovirtaa (Boundaryless Information Flow), joka on konsortion mukaan yrityksen infrastruktuurin tavoittele, jossa yrityksellä on pääsy integroituun informaatioon, jota se voi käyttää liiketoimintaprosessien parantamisen tukemiseksi [37]. Spesifikaatiossa ei määritellä ontologia-termiä, vaan siinä viitataan muihin ontologiaa käsitteleviin lähteisiin. Tässä tutkielmassa käytetään kohdassa 2.3.1 esitettyä ontologian määritelmää:

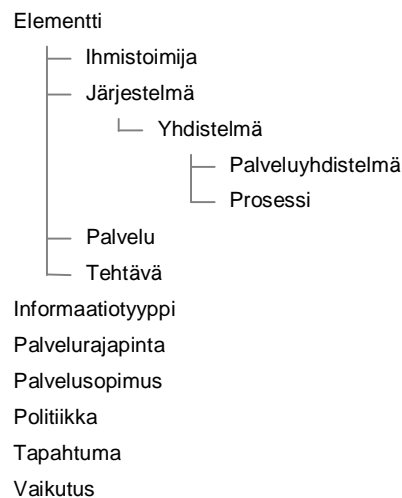
Ontologia on tietyn aihealueen tarkasti ja formaalisesti määritelty käsitteellinen malli, joka on jaettu osallistujien kesken. Ontologiaa voidaan käyttää sekä ihmisten että ohjelmistojen välisessä kommunikoinnissa. [28] [29]

The Open Groupin ontologia poikkeaa OASISin referenssimallista abstraktiotasoon ja esitysmuotoon nähden. Ontologia on konkreettisempi, ja se on esitetty formaalisesti OWL DL -kielellä. Formaalisen esitysmuodon johdosta ontologiaa voidaan potentiaalisesti käyttää mallilähtöisessä SOA-implemtoinnissa SOA-metamallien osana ja arkkitehtuurillisten artefaktien metadatana. Ontologia on tarkoitettu käytettäväksi myös toimialakohtaisten ontologioiden perustana. Spesifikaatiossa esitetään yhdenmukaisuusvaatimukset erikseen OWL-pohjaisille ontologioille ja ei-OWL-pohjaisille sovelluksille, kuten metamalleille ja ohjelmistoille. Vaatimusten

mukaan yhdenmukaisten OWL-pohjaisten ontologioiden tulee sisältää spesifikaatiossa esitetty ontologia kokonaisuudessaan, minkä lisäksi ne voivat sisältää muita rakenteita ja ontologioita. Ei-OWL-pohjaisten yhdenmukaisten sovellusten tulee sisältää määritelty ja yhtenäinen muunnos (engl. transform) ontologian ei-triviaaliin osajoukkoon, minkä lisäksi ne voivat sisältää muita rakenteita ja hyödyntää muita ontologioita. The Open Groupin ontologia on suunnattu mm. liiketoimintahenkilöstölle, arkkitehdeille sekä järjestelmä- ja ohjelmistokehittäjille. [36]

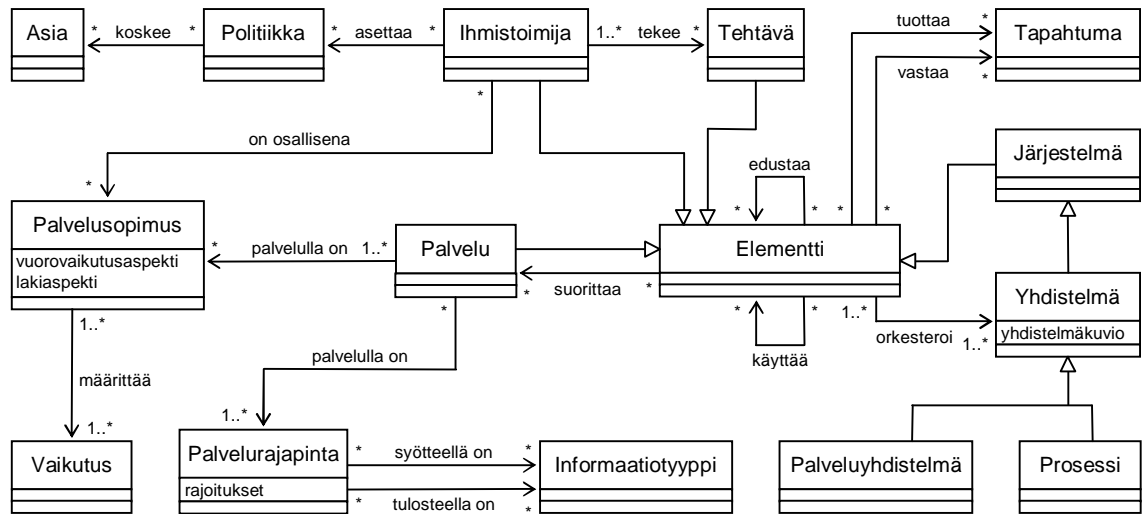
3.2.2 Mallin sisältö

Ontologia pyrkii kuvaamaan SOAn ydinkäsitteitä sekä liiketoiminnallisiin että teknisiin termein. Käsitteet on kuvattu OWL DL -kielellä luokkina ja niiden ominaisuuksina sekä niitä selittävinä sanallisina kuvauksina. Lisäksi kuvausten ohessa on käsitteiden esitykset UML-kaavioina sekä käyttöesimerkkejä käsitteistä. Ontologian luokkahierarkia on esitetty kuvassa 3.2.



Kuva 3.2. Ontologian luokkahierarkia. [36]

Ontologian keskeinen ja monipuolinen luokka on Elementti. Se on luonteeltaan abstrakti luokka, joka sisältää useiden muiden luokkien tarvitsemat perusominaisuudet, minkä vuoksi sitä voidaan käyttää useiden erilaisten luokkien yläluokkana. Graafinen tiivistelmä ontologiasta on esitetty kuvassa 3.3. Ontologian luokat on kuvattu taulukossa 3.3.



Kuva 3.3. Ontologian graafinen tiivistelmä. Kuva perustuu ontologiaan [36].

Taulukko 3.3. Ontologian luokkien kuvaukset. Luokkien ominaisuudet (suhteet ja dataominaisuudet) on merkitty lihavoidulla fontilla. Kuvaukset perustuvat ontologiaan [36].

Luokka	Kuvaus
Elementti	Elementti on ontologian monikäyttöinen perusluokka. Elementti on läpinäkymätön ja tietyllä abstraktiotasolla jakamaton yksikkö. Elementti voi mm. käyttää ja edustaa muita elementtejä. Käyttämällä tarkoitetaan yleisesti jonkinlaista vuorovaikutusta tai esimerkiksi sisältämissuhdetta. Lisäksi elementti voi suorittaa palveluja (olla palvelun suorittavana mekanismina), tuottaa tapahtumia, vastata tapahtumiin sekä orkesteroida yhdistelmiä.
Järjestelmä	Järjestelmä on Elementin alaluokka. Se on järjestetty kokoelma muita elementtejä, joita se käyttää .
Ihmistoimija (engl. HumanActor)	Ihmistoimija on Elementin alaluokka. Ihmistoimija on ihminen tai organisaatio. Ihmistoimija voi Elementiltä perittyjen suhteiden lisäksi tehdä tehtäviä, asettaa politiikkoja sekä olla osallisena palvelusopimuksessa.
Tehtävä (engl. Task)	Tehtävä on Elementin alaluokka. Tehtävä on atominen toiminta, joka tuottaa määritellyn tuloksen. Tehtäviä voivat tehdä ihmistoimijat eli ihmiset tai organisaatiot. Lisäksi Tehtävä perii Elementille määritetyt suhteet.

(jatkuu)

Taulukko 3.3. (jatkuu)

Palvelu	<p>Palvelu on Elementin alaluokka. The Open Group määrittelee palvelun seuraavasti:</p> <p><i>Palvelu on toistettavan toiminnan, jolla on määritelty seuraus (engl. outcome), looginen esitys (...) Se on itsenäinen (engl. self-contained) ja 'musta laatikko' (engl. black box) kuluttajilleen. [2]</i></p> <p>Palvelun toiminnallisuuden suorittavat siihen määrätty elementit. Palvelulla voi olla sopimus ja rajapinta. Palvelu perii lisäksi Elementille määritetyt suhteet, mutta palvelun loogisen luonteen vuoksi se ei voi käyttää, edustaa eikä suorittaa muita elementtejä eikä orkesteroida yhdistelmiä. Näitä ominaisuuksia voivat sen sijaan toteuttaa palvelua suorittavat elementit.</p>
Palvelusopimus	<p>Palvelusopimus määrittelee palvelun käyttämisen ehdot ja vuorovaikutussäännöt, jotka vuorovaikutuksessa olevien osallistujien täytyy hyväksyä. Palvelusopimuksessa määritellään sopimuksen vuorovaikutusaspektit ja lakiaspektit, joihin sisältyy mm. sopimuksen SLA (service level agreement, suom. palvelutasosopimus). Sopimus voi sisältää laillisia velvoitteita ihmistoimijoille, jotka ovat osapuolina sopimuksessa. Palvelusopimuksessa määritellään myös palvelun vaikutus.</p>
Vaikutus	<p>Vaikutus on palvelusopimuksessa määritelty, palvelun kuluttajalle näkyvä seuraus, jonka palvelun suorittavat elementit tuottavat.</p>
Palvelurajapinta	<p>Palvelurajapinta määrittelee, miten toiset elementit voivat olla vuorovaikutuksessa ja vaihtaa informaatiota palvelun kanssa. Palvelurajapinnat ovat tyypillisesti viestipohjaisia. Palvelurajapinnat määritellään aina palvelujen implementaatioista riippumattomasti. Palvelurajapinnan rajoitukset-ominaisuudessa voidaan määritellä formaalisesti tai epäformaalisesti palvelun vuorovaikutusta koskevia rajoituksia, kuten syötteen sallittu arvoalue. Palvelurajapinnassa voidaan määrittää palvelun syötteen ja tulosteen informaatiotyyppi.</p>
Informaatiotyyppi	<p>Informaatiotyyppi määrittelee palvelurajapinnan kautta vaihdettavan informaation – palvelun syötteen tai tulosteen – tyyppin.</p>

(jatkuu)

Taulukko 3.3. (jatkuu)

Yhdistelmä	Yhdistelmä on Järjestelmän alaluokka. Yhdistelmä on joukko asioita, jotka on yhdistetty joltain tarkoitusta varten. Yhdistelmän tyyppi määrittellään yhdistelmän yhdistelmäkuvio -ominaisuudessa. Yhdistelmäkuvioita ovat esim. orkesterointi (engl. orchestration), koreografia (engl. coreography) ja yhteistyö (engl. collaboration). Orkesterointi on kuvio, jossa yksi elementti valvoo ja ohjaa muita elementtejä eli orkesteroi yhdistelmää. Koreografia on kuvio, jossa elementit toimivat yhdessä ennalta määrätyn käyttäytymismallin mukaisesti ilman ohjaajaelementtiä. Yhteistyö on kuvio, jossa elementit toimivat yhdessä mutta toisistaan riippumattomasti – ilman ohjaajaelementtiä tai ennalta määrättyä käyttäytymismallia.
Palveluyhdistelmä	Palveluyhdistelmä on Yhdistelmän alaluokka. Palveluyhdistelmä yhdistelee eli käyttää palveluja. Se voi myös itse toimia palveluna (palvelun suorittavana elementtinä). Palveluyhdistelmä lisää tyypillisesti logiikkaa yhdistelmään yhdistelmäkuvio -ominaisuuden avulla.
Prosessi	Prosessi yhdistelee elementtejä aktiviteettien ja vuorovaikutuksen sarjaksi tai virraksi jonkin työn suorittamiseksi. Prosessin elementit voivat olla esim. ihmistoimijoita, tehtäviä, palveluja ja muita prosesseja. Prosessi lisää aina logiikkaa yhdistelmään yhdistelmäkuvio-ominaisuuden avulla.
Politiikka	Politiikka on mitä tahansa kohdetta koskeva , ihmistoimijan asettama ohjausmäärittelmä, jota kyseinen ihmistoimija aikoo itse noudattaa tai jonka tämä on asettanut toisen ihmistoimijan noudatettavaksi. Poliitiikan tunteminen helpottaa sen kohteen kanssa vuorovaikutuksessa olemista ja parantaa vuorovaikutuksen läpinäkyvyyttä.
Tapahtuma	Tapahtuma on asia, joka tapahtuu ja johon elementti voi haluta vastata . Mikä tahansa elementti voi vastata tapahtumiin ja tuottaa tapahtumia. Elementin tuottamien ja vastaamien tapahtumien tunteminen helpottaa kyseisen elementin kanssa vuorovaikutuksessa olemista ja parantaa vuorovaikutuksen läpinäkyvyyttä.

3.2.3 Mallin soveltaminen

Ontologioita voidaan yleisesti tarkastellen käyttää monin tavoin semanttisten ongelmien ratkaisemiseen tietojärjestelmien kehittämisessä. The Open Groupin ontologialle esitetään spesifikaatiossa kaksi yleistä soveltamistapaa. Ensiksi, ontologiaa voidaan käyttää ymmärtämisen ja kommunikoinnin tukemiseen. Spesifikaatiossa korostetaan

ontologian pyrkimystä tukea SOAn ymmärtämistä siten, että se soveltuu sekä liiketoiminta- että IT-yhteisöille. Toiseksi, ontologiaa voidaan mahdollisesti käyttää mallilähtöisen implementoinnin perustana. Yksittäisinä käyttötapoina mainitaan erityisesti ontologian käyttäminen SOA-metamallien osana ja arkkitehtuurillisten artefaktien metadatanä. [36] Gerberin, Kotzén ja van der Merwen [38] mukaan ontologioiden avulla voidaan potentiaalisesti parantaa epäformaalien arkkitehtuurillisten metamallien laatua ja niihin perustuvien mallien yhtenäisyyttä. The Open Groupin ontologiaa sovelletaan tiettyyn sovellusalueeseen luomalla ontologian luokista kyseisen sovellusalueen asioita kuvaavia instansseja. [36] Spesifikaatio sisältää esimerkkejä ontologian soveltamisesta kahdelta sovellusalueelta. The Open Groupin [39] mukaan ontologian tavoitetta toimia mallilähtöisen implementoinnin perustana ei ole kuitenkaan vielä saavutettu.

The Open Groupin ontologia on omaksuttu keskeiseksi osaksi S-RAMP -spesifikaatiota, joka määrittää yhteisen tietomallin SOA-artefaktivarastoille. Spesifikaation ovat kehittäneet IBM, HP, Software AG ja TIBCO, mutta se on sittemmin luovutettu OASISille jatkokehittäväksi. Spesifikaatio pyrkii helpottamaan SOA-ympäristössä käytettävien teknisten artefaktien, kuten palvelukuvausten, liiketoimintaprosessimallien ja politiikkadokumenttien, käsittelyä ja jakamista sekä edistämään SOA-varastojen ja niihin liittyvien tuotteiden yhteensopivuutta. Ontologiaa voidaan näin ollen potentiaalisesti käyttää epäsuorasti SOA-tuotteiden yhteensopivuuden ja SOA-datan siirrettävyyden tukemiseen. [40] [39]

Ontologioiden eräs merkittävä käyttötapana on semanttisen informaation lisääminen tietojärjestelmiin ja artefakteihin ja tämän informaation käyttäminen semanttisten toimintojen, kuten hakujen ja loogisen päättelyn, tukemiseen ja automatisointiin. Semanttisia tekniikoita hyödyntävää palvelukeskeistä arkkitehtuuria kutsutaan semanttiseksi SOAksi. Semanttisella informaatiolla voidaan kuvata sekä tarjottavia palveluja, palvelujen kuluttajien tarpeita että vuorovaikutuksessa käytettävää informaatiota. Semanttisten ominaisuuksien avulla voidaan potentiaalisesti automatisoida kokonaan tai osittain palvelujen etsiminen, välittäminen, yhdistely ja kutsuminen, mikä tukee löyhää ja skaalautuvaa integraatiota. [41] The Open Groupin

ontologiaa ei ole suunniteltu käytettäväksi sellaisenaan semanttisen SOAn kehittämiseen, mutta tämän tukemiseksi on kehitetty muita ontologioita, kuten OWL-S (Web Ontology Language for Services) [42] ja WSMO (Web Service Modeling Ontology) [43]. Näitä ontologioita ei kuitenkaan käsitellä tarkemmin tässä tutkielmassa.

3.2.4 Havainnot

The Open Groupin ontologian tarkastelutaso on huomattavasti konkreettisempi kuin OASISin referenssimallin, sillä sen käsitteet kuvaavat palvelupohjaisen järjestelmän arkkitehtuurillisia elementtejä ja niiden ominaisuuksia. Tämä on luonnollinen ominaisuus sen kannalta, että ontologia pyrkii toiminaan arkkitehtuurillisten mallien metamallina ja metadatana SOA-implemtoinnissa. Noyn ja McGuinnessin [44] mukaan ontologian käsitteiden tulee yleisesti olla lähellä tarkastelualueen fyysisiä tai loogisia objekteja. The Open Groupin ontologian kuvaamat käsitteet ovat kuitenkin abstrakteja siihen nähden, että ne sisältävät vähän yksityiskohtia ja ovat toteutustekniikoihin nähden neutraaleja.

Ontologia keskittyy referenssimallin tavoin SOAn peruskäsitteisiin. Ontologia kuvaa joitain elementtejä, jotka ovat selvästi referenssimallin tarkastelualueen ulkopuolella, kuten erilaisia yhdistelmiä ja tapahtumaa. OASISin referenssimalli kuvaa toisaalta sellaisia merkittäviä SOAn käsitteitä, joita on vaikeaa tarkastella ontologian näkökulmasta, kuten palvelujen näkyvyyttä ja suorituskontekstia. Näkökulmien erilaisuuden vuoksi mallien tarkastelualueiden vertaaminen on vaikeaa. Ontologiassa kuvataan palvelupohjaisten sovellusten mallintamisen ja implementoinnin näkökulmasta useita samoja asioita, joita referenssimallissa kuvataan palvelujen dynamiikan ja metatason näkökulmasta. Ontologia kuvaa hieman referenssimallia laajemmin SOAn rakenteellisia seikkoja.

The Open Groupin ontologialla on OASISin referenssimallin tavoin potentiaalista arvoa SOAn peruskäsitteitä kuvaavana mallina, jota voidaan käyttää SOAn ymmärtämisen tukemiseen sekä yhteisen käsitteistön ja kielen muodostamiseen. Ontologian verrattain konkreettinen, järjestelmäläheinen muoto helpottanee ontologian ymmärtämistä etenkin teknisesti suuntautuneiden henkilöiden näkökulmasta. Spesifikaatiossa kuvataan

kuitenkin hyvin yksityiskohtaisesti ja laajasti ontologian ominaisuuksia, käyttöä ja OWL-kielistä esitystä, mikä rajoittaa spesifikaation käyttämistä sellaisenaan SOAn ymmärtämisen tukemiseen. Ontologian tietyt abstraktit luokat ja luokkien perintähierarkiat vaikeuttavat myös hieman sen ymmärtämistä. Ontologiaa ei ole omaksuttu laajasti mallilähtöisen implementoinnin perustaksi [39], vaikka sen tekninen muoto tukee ontologian käyttämistä järjestelmien kehittämisessä. The Open Groupin ontologiaa ja muita SOAa tukevia ontologioita on kuitenkin mahdollista käyttää tietojärjestelmien kehittämisessä myös muiden semanttisten ominaisuuksien tukemiseen. Tähän nähden ontologia poikkeaa merkittävästi OASISin referenssimallista, joka tukee järjestelmien kehittämistä vain käsitteellisellä tasolla.

Luku 4

Referenssiarkkitehtuurit

Edellisessä luvussa tutkittiin käsitteellisiä malleja, jotka kuvaavat SOAn keskeisiä rakenteellisia elementtejä, käsitteitä ja näiden suhteita. Tässä luvussa perehdytään kahteen SOAn referenssiarkkitehtuuriin: OASISin referenssiarkkitehtuurin perustukseen (engl. reference architecture foundation) palvelukeskeiselle arkkitehtuurille sekä The Open Groupin SOA-referenssiarkkitehtuuriin. Nämä ovat käsitteellisten mallien tavoin abstrakteja ja tekniikoista riippumattomia malleja, mutta referenssimallista ja ontologiasta poiketen ne pyrkivät kuvaamaan SOAa kattavasti ja ensisijaisesti toteutusta tukevasta näkökulmasta. Tarkasteltavat referenssiarkkitehtuurit ovat kuitenkin – referenssimallin ja ontologian tavoin – keskenään varsin erilaisia malleja, jotka poikkeavat toisistaan niin lähestymistapaan, kuvausalueeseen kuin käyttötapoihinkin nähden.

4.1 OASISin referenssiarkkitehtuurin perusta

OASISin referenssiarkkitehtuurin perusta on monimuotoinen referenssiarkkitehtuuri ja kehys, joka kuvaa SOAn *käsitteellistä perustaa*. Se edustaa kohdassa 2.3.3 mainittua, TOGAF-yritysarkkitehtuurikehyksessä kuvattua yrityksen perustusarkkitehtuuria [6]. Tämä on yrityksen abstraktein arkkitehtuuri, jonka varaan voidaan kehittää konkreettisempia arkkitehtuureja. [7] OASISin referenssiarkkitehtuuri koostuu kolmesta näkymästä, jotka on muodostettu vastaavasti kolmen eri näkökulman mukaan. Nämä näkymät kuvaavat SOA-ekosysteemiä, SOAn toteuttamista sekä SOAn omistamiseen liittyviä prosesseja. Näkymät puolestaan koostuvat malleista. OASISin

referenssiarkkitehtuurin kuvaus noudattaa näin ollen rakenteeltaan kohdassa 2.1 kuvattua, IEEE:n esittämää arkkitehtuurikuvauksen käsitteellistä mallia [6].

4.1.1 Yleiskatsaus

OASISin referenssiarkkitehtuuri on abstrakti kehys, joka kuvaa SOAa monipuolisesti ja pyrkii tukemaan erityisesti SOAn toteuttamista. Referenssiarkkitehtuurissa tarkastellaan palvelukeskeisiä järjestelmiä, niiden hallitsemiseen tarvittavia prosesseja sekä ekosysteemiä, jossa organisaatiot tarjoavat ja kuluttavat palveluja toistensa kanssa. Näitä aiheita kuvataan usein keinoin, kuten mallintamalla niiden arkkitehtuurillisia elementtejä, käsitteitä ja näiden suhteita sekä selittämällä aiheita ja niihin liittyviä seikkoja sanallisesti. Spesifikaatio on vielä keskeneräinen – sen nykyinen versio on vuonna 2011 julkaistu toinen julkinen luonnos. [6] OASISin [45] mukaan malli on kuitenkin suurelta osin valmis. Referenssiarkkitehtuuri tukeutuu vahvasti OASISin referenssimallin käsitteistöön ja malleihin. Referenssiarkkitehtuuri ja sen suhde referenssimalliin määritellään referenssiarkkitehtuurin spesifikaatiossa seuraavasti:

Referenssiarkkitehtuuri mallintaa tarkastelualueen abstrakteja arkkitehtuurillisia elementtejä riippumatta tekniikoista, protokollista ja tuotteista, joita käytetään tietyn ratkaisun implementointiin aihealueella. Se eroaa referenssimallista siten, että referenssimalli kuvaa aihealueen tärkeitä käsitteitä ja suhteita keskittyen siihen, mikä on aihealueen elementeille tunnusomaista; referenssiarkkitehtuuri tarkentaa mallia näyttääkseen täydemmän kuvan ja näyttäen myös sen, mitä mallinnettujen yksikköjen toteuttamiseen sisältyy. Se pysyy kuitenkin riippumattomana yksittäisistä ratkaisuista ja pätee sen sijaan ratkaisujoukkoon. [6]

OASISin referenssiarkkitehtuuri pyrkii toimimaan käsitteellisenä perustana, jonka varaan voidaan kehittää muita referenssiarkkitehtuureja ja lopullisia, konkreettisia arkkitehtuureja. Se ei kuitenkaan sovellu SOAn implementoinnin tekniseksi suunnitelmaksi, sillä lopullisten arkkitehtuurien kehittäminen edellyttää monien konkreettisten suunnittelupäätösten ja tekniikkavalintojen tekemistä. [6] Referenssiarkkitehtuuria voidaan käyttää myös SOAn elementtien, SOA-arkkitehtuurien kokonaisuuden ja omistusaluerajojen ylittämistä koskevien seikkojen ymmärtämisen tukemiseen. [31] Omistusaluerajojen ylittämällä tarkoitetaan palvelujen tarjoamista ja kuluttamista eri omistajien kesken. Referenssiarkkitehtuuri on suunnattu mm. yritysarkkitehdeille, liiketoiminta- ja IT-arkkitehdeille, liiketoiminta-analyytikoille,

päättäjille, käyttäjille ja kehittäjille. Spesifikaation nykyisten, vielä keskeneräisten yhdenmukaisuusvaatimusten mukaan referenssiarkkitehtuurin kanssa yhdenmukaisesta konkreettisesta arkkitehtuurista tulee voida tunnistaa referenssiarkkitehtuurin käsitteet ja niiden suhteet. Referenssiarkkitehtuurissa ei kuitenkaan määritellä tarkasti sitä käsitteiden tai komponenttien joukkoa, jotka luetaan mallin varsinaisiksi elementeiksi – toisin kuin OASISin referenssimallissa ja The Open Groupin ontologiassa. [6]

4.1.2 Mallin sisältö

Referenssiarkkitehtuurin keskeisenä arkkitehtuurillisena pyrkimyksenä on liiketoiminnan ja sitä tukevan IT:n integroiminen. Tämä ilmenee liiketoimintakeskeisenä lähestymistapana, jossa palvelukeskeisyys nähdään paradigmana, jota käytetään arvon vaihtamiseen osallistujien välillä. Referenssiarkkitehtuurissa tarkastellaan SOAa kolmesta näkökulmasta, joiden mukaisesti arkkitehtuurin kuvaus on jaettu kolmeen näkymään. *Osallistuminen SOA-ekosysteemissä* -näkyä kuvaa SOA-ekosysteemiä, jossa osallistujat tarjoavat ja kuluttavat palveluja päämääriensä saavuttamiseksi. Tämä ekosysteemi sisältää ihmisiä, organisaatioita, palveluja, prosesseja, laitteita ja muita yksiköitä, jotka vaikuttavat järjestelmään tai joihin se vaikuttaa. Ekosysteemin oletetaan jakaantuvan useisiin omistusalueisiin ja sisältävän useita hallintorakenteita. *SOA-ekosysteemin toteuttaminen* -näkyä kuvaa palvelupohjaisten järjestelmien toteuttamista SOA-ekosysteemissä. *Omistajuus SOA-ekosysteemissä* -näkyä kuvaa palvelupohjaisten järjestelmien ja SOA-ekosysteemin hallitsemiseen tarvittavia prosesseja. Näkymät koostuvat malleista, jotka on esitetty UML 2 -kielisten kaavioiden sekä sanallisten kuvausten avulla. Referenssiarkkitehtuurissa kuvataan myös mallien arkkitehtuurillisia seurauksia (engl. architectural implications). Nämä ovat malleja konkreettisempia ja yksityiskohtaisempia elementtejä tai mekanismeja, joiden avulla mallit voidaan toteuttaa. Referenssiarkkitehtuurin ensisijainen arkkitehtuurillinen päämäärä on se, että osallistujat ja näiden tarpeisiin sopivat palvelut voivat tehokkaasti tavoittaa toisensa ja olla vuorovaikutuksessa. Referenssiarkkitehtuurin toinen päämäärä on selvästi ymmärrettävän luotettavuustason (engl. level of confidence) mahdollistaminen osallistujien vuorovaikutukselle. Kolmantena päämääränä on palvelupohjaisten järjestelmien skaalautuvuuden tukeminen. [6]

Seuraavissa kohdissa kuvataan referenssiarkkitehtuurin näkymät ja näiden sisältämät mallit sekä annetaan esimerkkejä näkymien arkkitehtuurillisista seurauksista.

4.1.2.1 Osallistuminen SOA-ekosysteemissä -näkyvä

Osallistuminen SOA-ekosysteemissä -näkyvä kuvaa SOA-ekosysteemiä ja siinä toimimista. Palvelupohjaiset järjestelmät ovat luonteeltaan dynaamisia ja tukevat organisaatioiden välistä ohjelmistokomponenttien yhteensopivuutta ja jakamista, toisin kuin perinteiset hajautetut järjestelmät, jotka ovat luonteeltaan staattisempia ja keskitetyn omistuksen ja hallinnan alaisia. SOA-ekosysteemin, jossa palveluja tarjotaan ja kulutetaan mahdollisesti usean organisaation kesken – omistusrajojen yli – ymmärtäminen on tärkeää palvelupohjaisten järjestelmien ja SOAn hallitsemisen ymmärtämisen kannalta. Referenssiarkkitehtuurin kaksi muuta näkymää, *SOA-ekosysteemin toteuttaminen* ja *Omistajuus SOA-ekosysteemissä*, perustuvat tämän näkymän sisältämille malleille. Nämä mallit on kuvattu taulukossa 4.1.

Taulukko 4.1. Osallistuminen SOA-ekosysteemissä -näkyvän sisältämät mallit. Kuvaukset perustuvat referenssiarkkitehtuuriin [6].

Malli	Mallin kuvaus	Keskeiset mallissa tarkasteltavat käsitteet ja aiheet
Sosiaalinen rakenne SOA-ekosysteemissä	Malli kuvaa SOA-ekosysteemiä ympäristönä, jossa on voimassa erilaisia sosiaalisia rakenteita ja joka jakaantuu useisiin omistusalueisiin. Malli kuvaa myös osallistujien keskinäisiä suhteita ja resurssien jakamista tällaisessa ekosysteemissä.	<ul style="list-style-type: none"> • sosiaalinen rakenne <ul style="list-style-type: none"> ○ hierarkkinen rakenne (esim. yritys, valtion oikeudellinen hallintorakenne) ○ vertaisryhmä (esim. markkinat) • osapuoli, toimija ja osallistuja • rooli <ul style="list-style-type: none"> ○ oikeus ja velvollisuus ○ palvelun tarjoaja, kuluttaja, välittäjä ja omistaja • resurssi <ul style="list-style-type: none"> ○ tunniste, omistajuus ja omistajuusraja • halukkuus, luottamus ja riski • politiikka ja sopimus • kommunikaatio <ul style="list-style-type: none"> ○ semanttinen sitoutuminen
Toiminta SOA-ekosysteemissä	Malli kuvaa, miten osallistajat voivat toimia SOA-ekosysteemissä yhdessä päämääriensä saavuttamiseksi: palvelun kuluttajat voivat käyttää tarpeidensa tyydyttämiseksi toisten osallistujien tarjoamien palvelujen kautta kyvykkyyksiä, jotka ovat mahdollisesti kolmansien osapuolten omistuksessa.	<ul style="list-style-type: none"> • tarve, vaatimus ja kyvykkyys • liiketoimintaratkaisu ja yhdisteltävyys • toiminta, tosimaailmavaikutus ja yhteinen toiminta • tila, yksityinen tila ja jaettu tila

Osallistuminen SOA-ekosysteemissä -näkyvän arkkitehtuurillisena seurauksena esitetään esimerkiksi, että on oltava mekanismeja, joiden avulla voidaan tunnistaa ja hallita sosiaalisten rakenteiden jäseniä ja näiden attribuutteja sekä kuvata rooleja. [6]

4.1.2.2 SOA-ekosysteemin toteuttaminen -näköm

SOA-ekosysteemin toteuttaminen -näköm kuvaa palvelupohjaisten järjestelmien toteuttamista SOA-ekosysteemissä. Näkömässä keskitytään erityisesti palvelupohjaisten järjestelmien niihin ominaisuuksiin, jotka ovat tärkeitä palvelujen löytämisen ja vuorovaikutuksen kannalta. Näköm sisältää referenssiarkkitehtuurin keskeisimmät SOAn teknisten rakenteiden kuvaukset ja toimii *Osallistuminen SOA-ekosysteemissä* -näkömän sisältämien sosiaalisen kontekstin kuvauksien ohella perustana *Omistajuus SOA-ekosysteemissä* -näkömän ymmärtämiseksi. *SOA-ekosysteemin toteuttaminen* -näkömän sisältämät mallit on kuvattu taulukossa 4.2.

Taulukko 4.2. SOA-ekosysteemin toteuttaminen -näkömän sisältämät mallit. Kuvaukset perustuvat referenssiarkkitehtuuriin [6].

Malli	Mallin kuvaus	Keskeiset mallissa tarkasteltavat käsitteet ja aiheet
Palvelu-kuvaus	Mallissa tarkastellaan resurssien kuvaamista mutta erityisesti käsitellään palvelukuvauksia palvelujen käyttämisen ja vuorovaikutuksen kannalta.	<ul style="list-style-type: none"> • resurssien kuvaaminen <ul style="list-style-type: none"> ○ alkuperä, kategorisointi ja liitedokumentaatio • palvelukuvauksen elementit <ul style="list-style-type: none"> ○ rajapinta, saavutettavuus, toiminnallisuus, politiikka, metriikka ja noudattaminen • palvelukuvaus vuorovaikutuksen tukena <ul style="list-style-type: none"> ○ suorituskonteksti ja vuorovaikutuskuvaus
Palvelun näkyvyys	Malli kuvaa tietoisuuden muodostamista eri keinoin ja eri ryhmille, palvelukuvauksen merkitystä vuorovaikutushalukkuuden muodostamisessa ja saavutettavuuden vaatimuksia.	<ul style="list-style-type: none"> • näkyvyys liiketoimintakontekstissa • tietoisuus <ul style="list-style-type: none"> ○ tietoisuuden välittäminen ja tietoisuus monimutkaisissa sosiaalisissa rakenteissa • halukkuus • saavutettavuus <ul style="list-style-type: none"> ○ päätepiste, protokolla ja läsnäolo

(jatkuu)

Taulukko 4.2. (jatkuu)

Vuoro-vaikutus palvelujen kanssa	Malli kuvaa, miten palvelujen vuorovaikutus voidaan toteuttaa viestien vaihtamisen keinoin ja miten palvelujen yhteistoiminta voidaan toteuttaa erilaisilla palvelujen yhdistelytavoilla.	<ul style="list-style-type: none"> • viesti, toiminto ja tapahtuma • viestin vaihtaminen <ul style="list-style-type: none"> ○ viestinvaihtokuviot • palvelujen yhdistely <ul style="list-style-type: none"> ○ liiketoimintaprosessi, orkesterointi, yhteistyö ja koreografia
Politiikat ja sopimukset	Malli kuvaa politiikan ja sopimuksen esittämiseen tarvittavia elementtejä sekä politiikan ja sopimuksen pakottamista, mihin sisältyy mm. politiikan tyydyttymisen määrittäminen ja politiikkakonfliktien ratkaiseminen.	<ul style="list-style-type: none"> • politiikan ja sopimuksen esittäminen <ul style="list-style-type: none"> ○ politiikkakehys, looginen kehys ja politiikkaontologia ○ politiikan omistaja, subjekti, objekti ja politiikkarajoitus • politiikan pakottaminen <ul style="list-style-type: none"> ○ politiikkapäätös, politiikkakonflikti ja konfliktin ratkaiseminen

SOA-ekosysteemin toteuttaminen -näkömän arkkitehtuurillisena seurauksena esitetään esimerkiksi, että on oltava versiointimekanismi, jonka avulla voidaan tunnistaa palvelua kuvaavien dokumenttien eri yhdistelmät. [6]

4.1.2.3 Omistajuus SOA-ekosysteemissä -näkömä

Omistajuus SOA-ekosysteemissä -näkömä kuvaa palvelupohjaisten järjestelmien ja SOA-ekosysteemin hallitsemiseen tarvittavia prosesseja. Näkömä täydentää referenssiarkkitehtuurin sosiaalista kontekstia ja teknistä rakennetta kuvaavia näkymiä kuvaamalla, miten SOA-ekosysteemiä ja palvelupohjaisia järjestelmiä voidaan kontrolloida, jotta ne täyttäisivät niille asetetut odotukset. Näkömän sisältämät mallit on kuvattu taulukossa 4.3.

Taulukko 4.3. Omistajuus SOA-ekosysteemissä -näkökulman sisältämät mallit. Kuvaukset perustuvat referenssiarkkitehtuuriin [6].

Malli	Mallin kuvaus	Keskeiset mallissa tarkasteltavat käsitteet ja aiheet
Hallinto (engl. governance)	Malli kuvaa hallintoa ja sen elementtejä yleisesti sekä SOA-hallinnon erityispiirteitä. Mallissa tarkastellaan hallinnon tarkoitusta, organisointia ja toimeenpanoa sekä hallinnon suhdetta hallintaan (engl. management). Lisäksi tarkastellaan SOAn hallinnon osa-alueita ja sen suhdetta muihin hallinnon haaroihin, erityisesti IT-hallintoon.	<ul style="list-style-type: none"> • hallinnon merkitys • hallinnon yleinen malli <ul style="list-style-type: none"> ○ päämäärä, politiikka, johtajuus, hallintokehys, hallintoprosessit, sääntö, säädös, noudattaminen ja metriikka • SOA-hallinto <ul style="list-style-type: none"> ○ omistajuusrajojen vaikutus hallintoon ○ infrastruktuuria koskeva hallinto ○ palveluja koskeva hallinto ○ osallistujien vuorovaikutusta koskeva hallinto
Turvallisuus	Mallissa tarkastellaan tietoturvan ja erityisesti turvallisen vuorovaikutuksen keskeisiä osa-alueita, yleisiä uhkia sekä keinoja, joilla SOA-ekosysteemissä voidaan saavuttaa riittävä tietoturvan luotettavuustaso.	<ul style="list-style-type: none"> • turvallinen vuorovaikutus <ul style="list-style-type: none"> ○ luottamuksellisuus, eheys, todentaminen (engl. authentication), valtuutus (engl. authorization), kiistämättömyys (engl. non-repudiation) ja saatavuus • turvallisuusuhat <ul style="list-style-type: none"> ○ viestin muuttaminen, viestin sieppaaminen, välimiehenä toimiminen, esiintyminen (engl. spoofing), palvelunestohyökkäys, viestintoistohyökkäys ja valheellinen kiistäminen • turvallisuusvastaukset <ul style="list-style-type: none"> ○ salaus, digitaalinen allekirjoitus, viestitunniste, auditointi, loki ja resurssien asteittainen käyttäminen

(jatkuu)

Taulukko 4.3. (jatkuu)

Hallinta	Mallissa kuvataan palvelupohjaisten järjestelmien hallinnan osa-alueita ja hallintakeinoja. Mallissa myös tarkastellaan, missä konteksteissa hallintaa tarvitaan, miten hallintaa voidaan tukea monitoroinnilla ja raportoinnilla sekä mitä vaatimuksia hallinta asettaa SOA-infrastruktuurille.	<ul style="list-style-type: none"> • hallinnan merkitys ja suhde hallintoon • hallintakyvykkyudet SOA-ekosysteemissä <ul style="list-style-type: none"> ○ palvelujen elinkaaren, yhdistelmien, konfiguraation, suorituskyvyn ja palvelutason hallinta ○ tapahtumamonitoinnin ja politiikan hallinta • hallintakeinot <ul style="list-style-type: none"> ○ hallintapolitiikka, verkon hallinta, turvallisuuden hallinta ja käytön hallinta • hallinnan soveltaminen palvelusopimukseen, politiikkaan ja palvelukuvaukseen • monitorointi ja raportointi • hallintaa tukeva infrastruktuuri
SOA-testaus	Mallissa tarkastellaan, mitä elementtejä palveluissa ja SOA-ympäristössä tulee testata, millä tavoin testaaminen voidaan toteuttaa sekä SOA-ekosysteemin asettamia haasteita testaamiselle.	<ul style="list-style-type: none"> • perinteinen ohjelmistotestaus SOA-testaamisen perustana • SOA-ekosysteemin vaikutus testaamiseen <ul style="list-style-type: none"> ○ odottamattomat käyttötavat ja dynaaminen ja vaikeasti simuloitava ympäristö • SOA-testaamisen elementit <ul style="list-style-type: none"> ○ testaamisen kohde, tapa, suorittaja ja raportointi • palvelujen testaaminen <ul style="list-style-type: none"> ○ testaamisen vaiheet ○ palvelutyngän käyttäminen ○ palvelun riippuvuudet

Omistajuus SOA-ekosysteemissä -näkökulman arkkitehtuurillisina seurauksina esitetään esimerkiksi, että suorituskontekstin tulee turvata viestinnän luottamuksellisuus ja eheys, varmistaa palvelujen saatavuus kuluttajille sekä kyetä skaalautumaan kasvavan ekosysteemin mukana. [6]

4.1.3 Mallin soveltaminen

OASISin referenssiarkkitehtuurin ensisijainen tarkoitus on tukea SOAn toteuttamista. Se kuvaa abstraktilla tasolla elementtejä, käsitteitä ja niiden suhteita, jotka tulee

toteuttaa kohdearkkitehtuurissa. [6] Referenssiarkkitehtuurin lähestymistapa on kuitenkin hyvin käsitteellinen, ja siinä kuvataan ja selitetään sen aihealuetta monin paikoin – erityisesti *Omistajuus SOA-ekosysteemissä* -näkyvässä – yleisemmin kuin vain arkkitehtuurillisten rakenteiden kannalta. Mallien arkkitehtuurilliset seuraukset sen sijaan tarjoavat konkreettisempaa tukea SOAn toteuttamiseen.

Referenssiarkkitehtuuri pyrkii kokonaisvaltaiseen kuvaukseen – se on tarkoitettu kattavaksi abstraktiksi SOA-malliksi, jolla on arvoa mm. yritysarkkitehdeille, liiketoiminta- ja IT-arkkitehdeille sekä ylimmän johdon jäsenille, jotka osallistuvat strategiseen liiketoiminta- ja IT-suunnitteluun. Referenssiarkkitehtuuri on tarkoitettu erikokoisten ja -tyyppisten SOA-ekosysteemien tukemiseen yrityksen sisäisestä ekosysteemistä Internetin laajuiseen ekosysteemiin. [6]

Referenssiarkkitehtuuri pyrkii tukemaan erityisesti sellaisen arkkitehtuurin toteuttamista, jonka ominaisuuksia ovat aiemmin mainitut arkkitehtuurilliset päämäärät: tehokkuus, jolla osallistujat ja niiden tarpeisiin sopivat palvelut voivat tavoittaa toisensa ja olla vuorovaikutuksessa, vuorovaikutuksen selvästi ymmärrettävä luotettavuustaso ja palvelupohjaisten järjestelmien skaalautuvuus. [6]

Referenssiarkkitehtuuri on tarkoitettu myös SOAn ymmärtämisen tukemiseen. Se pyrkii tukemaan mm. SOAn elementtien, SOA-arkkitehtuurien kokonaisuuden ja omistusrajojen ylittämistä koskevien seikkojen ymmärtämistä. [31] Referenssiarkkitehtuuri on suunnattu tässä tarkoituksessa mm. päättäjille sekä palvelupohjaisten järjestelmien käyttäjille ja kehittäjille. Spesifikaation mukaan referenssiarkkitehtuuri tukee myös muiden SOA-spesifikaatioiden kehittämistä helpottamalla niiden keskinäisten suhteiden määrittämistä. [6]

4.1.4 Havainnot

OASISin referenssiarkkitehtuuri on referenssimallia yksityiskohtaisempi ja konkreettisempi. Referenssiarkkitehtuuri muistuttaa abstraktiotasoltaan The Open Groupin ontologiaa, joskin nämä edustavat varsin erilaisia näkökulmia, sillä referenssiarkkitehtuuri keskittyy SOAn käsitteellisen perustan kuvaamiseen, ja

ontologia puolestaan keskittyy palvelupohjaisten sovellusten rakenteellisiin ominaisuuksiin. Referenssiarkkitehtuurin arkkitehtuurillisissa seurauksissa kuvataan arkkitehtuuria kuitenkin yleistasoon nähden konkreettisemmin esimerkiksi kuvaamalla tarvittavia arkkitehtuurillisia mekanismeja ja työkaluja. Referenssiarkkitehtuurin voidaan näin ollen katsoa kuvaavan SOAa kahdella abstraktiotasolla.

Referenssiarkkitehtuuri poikkeaa referenssimallista ja ontologiasta huomattavimmin kuvausalueeseen nähden. Niissä kuvataan osittain samoja aiheita – erityisesti *SOA-ekosysteemin toteuttaminen* -näkyvässä kuvataan läheisesti samoja aiheita kuin referenssimallissa ja ontologiassa, joskin siinä tarkastellaan useita käsitteitä laajemmin ja monipuolisemmin. Sen sijaan *Omistajuus SOA-ekosysteemissä* -näkyvän aiheet ovat lähes kokonaan referenssimallin ja ontologian tarkastelualueen ulkopuolella. *Osallistuminen SOA-ekosysteemissä* -näkyvässä kuvattu ekosysteemi ja siinä toimiminen eivät ole varsinaisina tarkastelun kohteina referenssimallissa eivätkä ontologiassa, vaikka nämä – erityisesti referenssimalli – sisältävät myös tämän aihealueen joitain käsitteitä.

Referenssiarkkitehtuurin muoto on vaihteleva. Ensimmäinen näkymä, *Osallistuminen SOA-ekosysteemissä*, mallintaa SOA-ekosysteemiä ja siinä toimimista pääasiassa ei-teknisten käsitteiden avulla. Se pyrkii selittämään ekosysteemin olemusta ja toimii referenssimallin kaltaisena käsitteellisenä perustana kahdelle muulle näkymälle. Näkyvän arkkitehtuurilliset seuraukset ovat perusluonteisempia muiden näkymien arkkitehtuurillisiin seurauksiin verrattuna. Toinen näkymä, *SOA-ekosysteemin toteuttaminen*, kuvaa palvelupohjaisia järjestelmiä: niiden teknisiä elementtejä ja useita niihin liittyviä käsitteitä ja seikkoja. Tämä näkymä pyrkii tukemaan palvelupohjaisten järjestelmien toteuttamista. Kolmas näkymä, *Omistajuus SOA-ekosysteemissä*, kuvaa SOAn omistamiseen liittyviä prosesseja ja selittää miten niitä toteutetaan. Näkyvässä selitetään useita näihin prosesseihin sisältyviä seikkoja ja kuvataan niihin liittyviä teknisiä ja ei-teknisiä käsitteitä. Näkymä toimii suppean oppaan kaltaisena materiaalina ja pyrkii tukemaan SOAn hallitsemista. Tällaisten, näkymien sisältöä ja käyttötarkoitusta koskevien, eroavaisuuksien vuoksi spesifikaatio on jossain määrin epäyhtenäinen. Lisäksi mallien käsitteelliset kuvaukset ovat osittain heikosti

jäsenneltyjä. Näiden seikkojen vuoksi referenssiarkkitehtuurin merkityksen ja käyttötarkoituksen hahmottaminen on vaikeaa eikä mallia miellä helposti arkkitehtuuriksi. Spesifikaatiota kehittävän teknisen komitean [46] mukaan referenssiarkkitehtuuria voidaan käyttää ikään kuin tarkistuslistana asioista, jotka tulee ottaa huomioon SOA-ratkaisujen arkkitehtuurillisessa suunnittelussa ja konkreettisten arkkitehtuurien validoinnissa.

Referenssiarkkitehtuuri on tarkoitettu toteutusorientoituneeksi SOA-malliksi. Se pyrkii tukemaan SOAn toteuttamista kuvaamalla elementtejä, käsitteitä ja suhteita, jotka tulee toteuttaa kohdearkkitehtuurissa. Referenssiarkkitehtuurin käyttömahdollisuudet tässä tarkoituksessa ovat kuitenkin rajalliset, sillä vain sen joissain osissa keskitytään arkkitehtuurillisten elementtien ja käsitteiden mallintamiseen. Lisäksi referenssiarkkitehtuurissa ei määritellä tarkasti sitä arkkitehtuurillisten elementtien ja käsitteiden joukkoa, joka tulee toteuttaa kohdearkkitehtuurissa. Sen sijaan mallien arkkitehtuurilliset seuraukset määritellään melko selvästi, mikä tarjoaa konkreettisempaa ja käytännönläheisempää tukea SOAn toteuttamiseen. Referenssiarkkitehtuuri pyrkii tukemaan SOAn toteuttamista lisäksi selittämällä sanallisesti monia arkkitehtuuriin liittyviä aiheita, kuten SOAn hallitsemiseen tarvittaviin prosesseihin liittyviä seikkoja. Referenssiarkkitehtuuri pyrkii siis tarjoamaan monipuolista tukea SOAn toteuttamiseen, mutta sen monimuotoisuuden, epäyhtenäisyyden ja osittain heikosti jäsennellyn muodon vuoksi sen merkityksen ja ensisijaisen käyttötarkoituksen hahmottaminen on vaikeaa.

Referenssiarkkitehtuurin on tarkoitus tukea myös SOAn ymmärtämistä. Tämä on keskeinen tavoite erityisesti *Osallistuminen SOA-ekosysteemissä* -näkyvässä, joka kuvaa SOA-ekosysteemiä kaikkien osallistujien näkökulmasta ja korostaa omistusrajojen ylittämiseen liittyviä seikkoja. Näkymä kuvaa ekosysteemiä melko yksityiskohtaisesti ja soveltuu näin ollen pääasiassa niille, joiden tulee ymmärtää SOAa syvällisesti. Referenssiarkkitehtuuri pyrkii tukemaan myös SOAn kokonaisuuden ymmärtämistä abstraktilla tasolla [31].

4.2 The Open Groupin referenssiarkkitehtuuri

The Open Groupin referenssiarkkitehtuuri on tässä tutkielmassa tarkasteltavista SOA-malleista konkreettisin. Se kuvaa SOAn loogista teknistä rakennetta laaja-alaisesti ensiksi korkealla abstraktiotasolla kuvaamalla SOAn kerrosrakennetta ja toiseksi matalalla abstraktiotasolla kuvaamalla kerrosten sisältämiä rakennuspaloja (engl. architectural building block, ABB) ja näiden toteuttamia kyvykkyyskäsitteitä. Referenssiarkkitehtuuri edustaa kohdassa 2.3.3 mainittua, TOGAF-yritysarkkitehtuurikehyksessä kuvattua yleisten järjestelmien arkkitehtuuria [23], joka sijoittuu abstraktiotasoltaan perustusarkkitehtuurin ja toimialakohtaisen arkkitehtuurin väliin. Yleisten järjestelmien arkkitehtuurit ovat epätäydellisiä yrityksen kokonaisarkkitehtuuriin nähden mutta pyrkivät olemaan täydellisiä tiettyyn ongelma-alueeseen – tässä tapauksessa SOAn tekniseen arkkitehtuuriin – nähden. [7] Referenssiarkkitehtuurin soveltamistapojen ohessa esitetään lyhyesti tapausesimerkki, jossa referenssiarkkitehtuuri on omaksuttu perustaksi IBM:n kehittämälle pilvipalvelureferenssiarkkitehtuurille.

4.2.1 Yleiskatsaus

The Open Groupin vuonna 2011 julkaisema referenssiarkkitehtuuri kuvaa laaja-alaisesti SOAn teknistä rakennetta loogisesta näkökulmasta. Referenssiarkkitehtuuri koostuu yhdeksästä kerroksesta, jotka keskittyvät erilaisiin vastuualueisiin. Kerrokset sisältävät tiettyjä teknisiä kyvykkyyskäsitteitä toteuttavia arkkitehtuurillisia rakennuspaloja. Nämä ovat arkkitehtuurimallin rakenneosia, jotka kuvaavat yhden aspektin koko mallista [7]. Spesifikaatio perustuu Arsanjanin et al. [47] esittämään, laajasti viitattuun S3-referenssiarkkitehtuuriin (Service-Oriented Solution Stack), jossa kerrokset kuvataan korkealla tasolla. S3 on IBM:n työryhmän kehittämä, ja IBM osallistui merkittävästi myös The Open Groupin referenssiarkkitehtuurin kehittämiseen. The Open Groupin referenssiarkkitehtuurin päämääränä on SOA-ratkaisuarkkitehtuurien ja yritysarkkitehtuurien luomisen ja arvioinnin tukeminen. SOA-ratkaisuarkkitehtuurilla tarkoitetaan toteutettavaa SOA-arkkitehtuuria. Referenssiarkkitehtuuria voidaan käyttää sekä toimialakohtaisten referenssiarkkitehtuurien että konkreettisten arkkitehtuurien kehittämiseen. Referenssiarkkitehtuurissa kuvatuista SOAn arkkitehtuurillisista

rakennuspaloista osa on perustavanlaatuisia tai SOAn keskeisten ominaisuuksien kannalta välttämättömiä, mutta jotkut rakennuspalat ja jopa kerrokset ovat valinnaisia. Välttämättömien ja valinnaisten rakennuspalojen joukkoja ja muita referenssiarkkitehtuurin yhdenmukaisuusvaatimuksia ei ole kuitenkaan määritelty. Referenssiarkkitehtuuri on suunnattu ensisijaisesti SOAa omaksuville organisaatioille, minkä lisäksi se on suunnattu SOA-tuotteita kehittäville ohjelmistotoimittajille, integraattoreille sekä standardointijärjestöille. [23]

4.2.2 Mallin sisältö

Seuraavissa kohdissa tarkastellaan ensin yleisesti referenssiarkkitehtuurin sisältämiä elementtejä ja sen keskeisiä käsitteitä, minkä jälkeen luodaan yleiskatsaus referenssiarkkitehtuurin kerroksiin ja esitellään jokainen kerros yksittäin. Kuvaukset perustuvat referenssiarkkitehtuuriin. [23]

4.2.2.1 Elementit ja keskeiset käsitteet

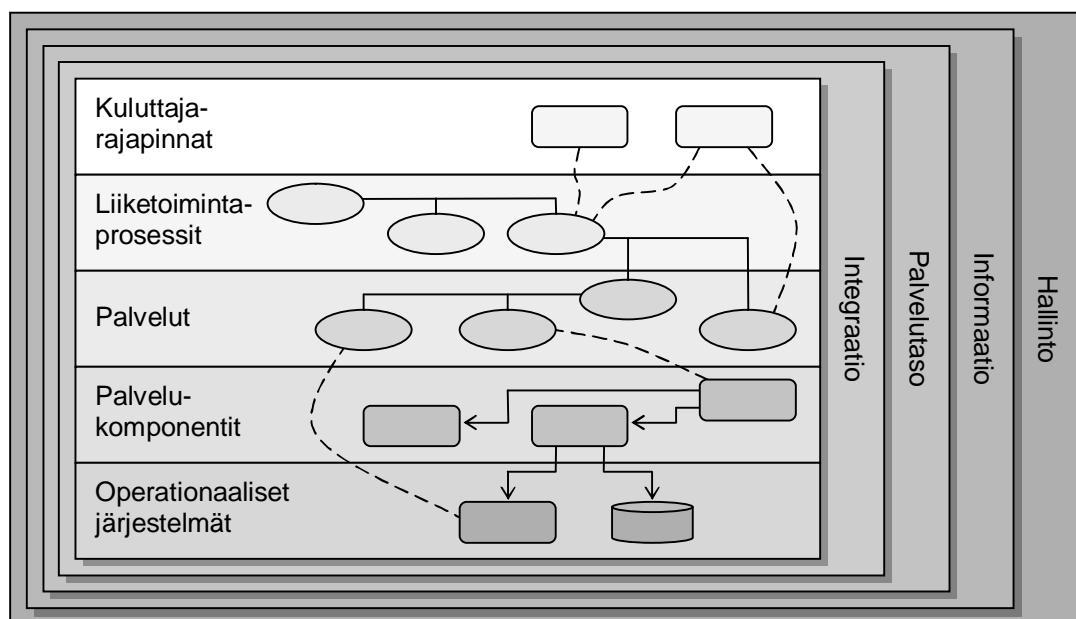
Referenssiarkkitehtuuri kuvaa ensisijaisesti kahta aspektia SOAn toteuttamisesta. Ensiksi, malli kuvaa teknisiä kyvykkyyksiä, jotka SOAn tulee toteuttaa. Nämä edustavat SOAn teknisiä, toiminnallisia vaatimuksia. Toiseksi, malli kuvaa loogisia teknisiä elementtejä, jotka toteuttavat edellä mainittuja SOAlta vaadittavia kyvykkyyksiä. Näitä elementtejä ovat mallin arkkitehtuurilliset rakennuspalat. Näiden ajoaikaisia instansseja kutsutaan referenssiarkkitehtuurissa ratkaisurakennuspaloiksi (engl. solution building block). Kyvykkyydet ja niitä toteuttavat rakennuspalat on ryhmitelty referenssiarkkitehtuurin kerroksiin, jotka edustavat SOAn tiettyjä merkittäviä vastuualueita tai piirteitä. Lisäksi referenssiarkkitehtuurissa tarkastellaan hieman sen implementointia koskevia seikkoja.

Kyvykkyyksien, rakennuspalojen ja kerrosten ohella muita mallissa tarkasteltavia asioita ovat muun muassa rakennuspalojen välisen vuorovaikutuksen vaiheet eli ns. metodiaktiviteetit ja vuorovaikutuskuviot. Lisäksi siinä tarkastellaan hieman arkkitehtuurillisia vaihtoehtoja ja päätöksiä, joihin vaikuttavat avainsuorituskykymittarit (engl. key performance indicator, KPI) ja ei-toiminnalliset vaatimukset (engl. non-functional requirement, NFR). Näitä ja eräitä muita käsitteitä sekä näiden suhteita

kuvataan referenssiarkkitehtuurin metamallissa, jonka tarkoitus on tukea referenssiarkkitehtuurin instanssin luomista. Metamalli on esitetty liitteessä A. Referenssiarkkitehtuuri on kuvattu sanallisesti ja lukuisten kuviodien avulla, joita käytetään esimerkiksi rakennuspalojen riippuvuussuhteiden ja vuorovaikutusten havainnollistamiseen.

4.2.2.2 Kerrokset

Referenssiarkkitehtuuri koostuu yhdeksästä loogisesta kerroksesta, joista jokainen on keskittynyt tiettyjen, toisiinsa läheisesti liittyvien kyvykkyyksien tarjoamiseen. Kerrokset edustavat myös eri aspekteja SOAn arvolupauksista. Kerroksista viisi on horisontaalisia, ja ne toteuttavat SOAn toiminnallisia kyvykkyyksiä. Muut neljä kerrosta ovat vertikaalisia eli läpileikkaavia, ja ne toteuttavat horisontaalisia kerroksia tukevia kyvykkyyksiä. Vertikaaliset kerrokset toteutetaan tyypillisesti ohjelmistotoimittajien tuotteiden avulla. Referenssiarkkitehtuurin kerrosrakenne on esitetty kuvassa 4.1.



Kuva 4.1. Referenssiarkkitehtuurin looginen kerrosrakenne. [23]

Kerroksittaisessa arkkitehtuurissa jokainen kerros voi käyttää välittömästi sen alapuolella olevan kerroksen palveluja. Referenssiarkkitehtuurissa kerrokset voivat käyttää jossain määrin myös muita kuin välittömästi niiden alapuolella olevia kerroksia, minkä vuoksi mallia kutsutaan spesifikaatiossa osittain kerroksittaiseksi

arkkitehtuuriksi. Lisäksi jotkin kerrokset – kuten Liiketoimintaprosessit-kerros (engl. Business Process Layer) tai Integraatiokerros (engl. Integration Layer) – voidaan jättää toteuttamatta matalan kypsyystason SOA-toteutuksissa.

Referenssiarkkitehtuurin kerroksia voidaan ryhmitellä loogisesti horisontaalisten ja vertikaalisten kerrosryhmien lisäksi myös muilla tavoin. Ensiksi, palvelujen implementointia ja rajapintaa tukevia kerroksia ovat Palvelut-kerros (engl. Services Layer), Palvelukomponenttikerros (engl. Service Component Layer) ja Operationaaliset järjestelmät -kerros (engl. Operational Systems Layer), ja toisaalta palvelujen kuluttamista tukevia kerroksia ovat Kuluttajakeros (engl. Consumer Layer), Liiketoimintaprosessit-kerros ja Integraatiokerros. Toiseksi, Operationaaliset järjestelmät -kerros kuvaa fyysisen SOA-arkkitehtuurin ajoaikaisia elementtejä, ja toisaalta kaikki muut kerrokset kuvaavat SOA-arkkitehtuurin loogisia, vastuualueiden mukaan kerroksiksi ryhmiteltyjä elementtejä (joilla on ajoikainen vastine Operationaaliset järjestelmät -kerroksessa).

Operationaaliset järjestelmät -kerros

Operationaaliset järjestelmät -kerrosta voidaan tarkastella kahdesta näkökulmasta. Ensiksi, kerros edustaa kaikkia SOAn elementtejä, joita tarvitaan muiden kerrosten rakennuspalojen tukemiseen. Toiseksi, kerros edustaa koko SOAn (myös muiden kerrosten) ajoaikaista näkymää ja sisältää näin ollen myös muiden kerrosten rakennuspalojen ajoikaiset vastineet (Ratkaisukomponentteina, engl. Solution Component, tai Ratkaisurakennuspaloina, engl. Solution Building Block).

Kerroksen tukemat kyvykkyyskategoriat ovat 1) Palvelun toimittaminen, 2) Ajoikainen ympäristö sekä 3) Virtualisointi- ja infrastruktuuripalvelut.

Esimerkkejä kerroksen rakennuspalloista on esitetty alla. Kyvykkyyskategorian numero on suluissa rakennuspalan kuvauksen jälkeen.

- **Ratkaisukomponentti** on Palvelukomponentin ajoikainen vastine. (1)
- **Sovellukset**, joita ovat paketoituidut sovellukset, kuten ERP (enterprise resource planning, suom. toiminnanohjausjärjestelmä) ja CRM (customer relationship

management, suom. asiakkuudenhallintajärjestelmä), ja erikoisvalmisteiset sovellukset. (1)

- **Tietokanta** (1)
- **Ratkaisualusta**, (engl. Solution Platform) joka edustaa esimerkiksi JVM-virtuaalikonetta (Java Virtual Machine). (2)
- **Ratkaisurakennuspala** on SOAa tukevien rakennuspalojen ajoaikainen vastine. (2)

Palvelukomponenttikerros

Palvelukomponenttikerros edustaa palvelujen toteutuskomponentteja. Nämä ovat komponentteja, jotka voidaan suorittaa palvelusäiliössä ja joiden toimintoja voidaan paljastaa palvelurajapintojen (esimerkiksi WSDL) avulla. Palvelukomponentit voivat käyttää Operationaaliset järjestelmät -kerroksessa kuvattuja järjestelmiä, joita ei voi sellaisinaan paljastaa palveluina.

Kerros tukee sekä suunnitteluajaisia että ajoajaisia kyvykkyksiä. Kerroksen tukemat suunnitteluajaiset kyvykkyyskategoriat ovat 1) Palvelun toteuttaminen ja implementointi, 2) Palvelun julkaiseminen ja paljastaminen sekä 3) Palvelun sijoittaminen (engl. deployment). Kerroksen tukemat ajoajaiset kyvykkyyskategoriat ovat 4) Palvelun kutsuminen ja 5) Palvelun sidonta (engl. binding).

Esimerkkejä kerroksen rakennuspaloista on esitetty alla. Kyvykkyyskategorian numero on suluisissa rakennuspalan kuvauksen jälkeen.

- **Palvelukomponentti** on palvelun implementoiva komponentti, kuten BPEL- (Business Process Execution Language) tai EJB-komponentti (Enterprise JavaBeans). (1)
- **Palvelun julkaisija** (engl. Service Publisher) julkaisee palvelun metadatan ja kuvauksen Hallintokerroksen (engl. Governance Layer) Palveluvarastoon (engl. Service Repository). (2)
- **Integroitu kehitysympäristö palvelun kehittämiseen** (2)
- **Palvelunsijoitusmanageri** (engl. Service Deployment Manager) sijoittaa ajoajaisen palvelukomponentin Palvelut-kerroksen Palvelusäiliöön (engl.

Service Container) ja rekisteröi palvelun kuvauksen Hallintokerroksen Palvelurekisteriin. (3)

- **Palvelun kutsuja** (engl. Service Invoker) ottaa vastaan kutsuja muilta kerroksilta. Se tukee mm. Palvelukomponentin lataamista Palvelut-kerroksen Palvelusäiliöön. (4)

Palvelut-kerros

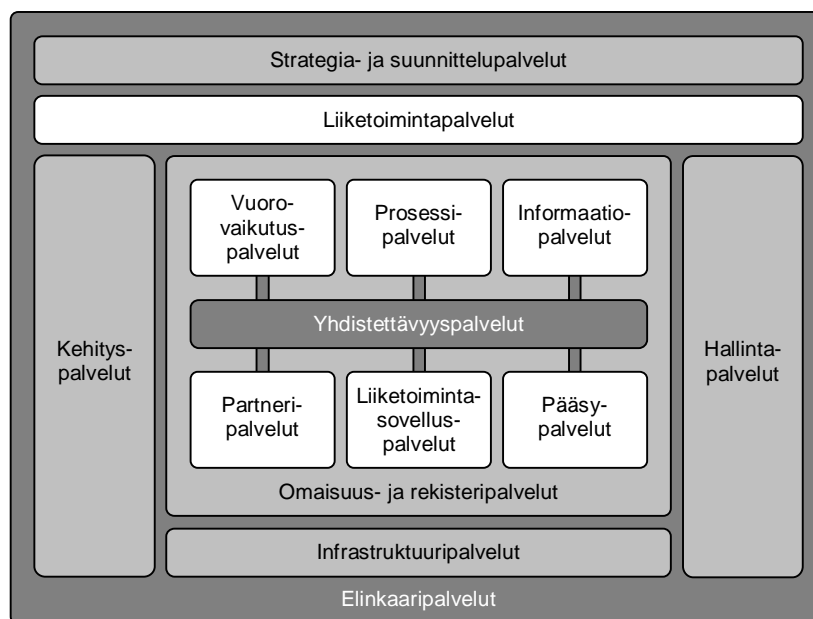
Palvelut-kerros edustaa palvelumäärittäjiä ja implementaatiosta riippumatonta aspektia palvelun käyttämisestä. Kerros toimii abstraktiona palvelun kuluttajien ja implementaation välillä.

Kerroksen tukemat kyvykkyyskategoriat ovat 1) Palvelun määrittäminen, 2) Palvelun ajoaikainen mahdollistaminen, 3) Poliitiikan hallinta, 4) Pääsyn valvonta ja 5) Palvelun klusterointi. Poliitiikan hallinta ja Pääsyn valvonta -kategoriat toteutetaan kokonaan muiden kerrosten rakennuspaloilla.

Esimerkkejä kerroksen rakennuspaloista on esitetty alla. Kyvykkyyskategorian numero on suluissa rakennuspalan kuvauksen jälkeen.

- **Palvelu** (1)
- **Palvelusäiliö/Palvelukäytävä** kapseloi palvelukomponentit ja mahdollistaa palvelujen kutsumisen ja suorittamisen. (2)
- **Klusterimanageri** (engl. Cluster Manager) tukee skaalautuvuutta tarjoamalla klusterointi- ja välimuistitukea. (5)

SOAn palvelut – joita kaikkia Palvelut-kerros edustaa – voidaan jakaa ryhmiin niiden toiminnallisuuden mukaan. Tämä vaikuttaa liiketoiminnan näkemykseen ja ymmärrykseen SOAsta sekä palveluiden ryhmittelyyn palveluportfoliossa. Referenssiarkkitehtuurissa esitetään eräs palveluiden kategorisointitapa, jota voidaan käyttää tarkistuslistana palveluportfolion kehittämisessä. Tämä kategorisointi on esitetty kuvassa 4.2. Lyhyet kuvaukset kategorioista on esitetty liitteessä B.



Kuva 4.2. Referenssiarkkitehtuurin palvelukategoriat. Valkoisella taustavärillä esitettyjen kategorioiden palvelut ovat sovellusalue- tai SOA-toteutuskohtaisia. Muut palvelut ovat generisiä, ja niitä käytetään SOA-toteutuskohtaisten palvelujen suunnitteluun, kehittämiseen, tukemiseen ja hallintaan. Informaatio-, vuorovaikutus- ja prosessipalvelut tukevat MVC-kuviota. [23]

Liiketoimintaprosessit-kerros

Liiketoimintaprosessit-kerros toimii kuluttajarajapinnan ja liiketoimintapalvelujen välisenä kerroksena, jota voidaan käyttää palvelujen yhdistelemiseen ja näin muodostetun toiminnallisuuden tai sovelluksen tarjoamiseen kuluttajalle. Kerros tukee prosessilogiikan määrittelyä (orkesteroinniksi tai koreografiaksi) erillään liiketoiminta-aktiiviteetteja suorittavista palveluista, mikä tukee ratkaisujen muutettavuutta, SOAn ketteryyttä ja IT:n asettamista linjaan liiketoiminnan kanssa.

Kerroksen tukemat kyvykkyyskategoriat ovat 1) Prosessin määrittäminen, 2) Tapahtumankäsittely, 3) Prosessin ajoaikainen mahdollistaminen, 4) Prosessin informaation hallinta, 5) Päätöksen hallinta, 6) Prosessin integrointi, 7) Turvallisuuden ja politiikan noudattaminen sekä 8) Prosessin monitorointi ja hallinta. Kategoriat 2, 5, 7 ja 8 toteutetaan kokonaan muiden kerrosten rakennuspaloilla.

Esimerkkejä kerroksen rakennuspaloista on esitetty alla. Kyvykkyyskategorian numero on suluisissa rakennuspalan kuvauksen jälkeen.

- **Liiketoimintaprosessi** (1)
- **Prosessisäiliö/Prosessimoottori** (engl. Process Container/Process Engine) tarjoaa ympäristön prosessin suorittamiselle sekä ihmisen ja prosessin välisen vuorovaikutuksen hallinnalle. Se vastaa myös prosessi-instanssin hallinnasta. (3)
- **Työnkulkumanageri/Ihmistehtävämanageri** (engl. Workflow Manager/Human Task Manager) muun muassa koordinoi ihmisten ja prosessien välistä vuorovaikutusta. (3)
- **Prosessivarasto** (engl. Process Repository) (4)
- **Prosessipalvelusovittaja** (engl. Process Service Adapter) integroi kerroksen muihin kerroksiin. Se toimii mekanismina, joka sallii prosessien paljastamisen palveluina. (6)

Kuluttajakerros

Kuluttajakerros edustaa rajapintaa SOAn ulkopuolisten kuluttajien ja palvelupohjaisten sovellusten välillä. Se tukee muun muassa asiakassovelluksia ja niiden integroimista palveluihin ja prosesseihin. Kerros mahdollistaa palvelupohjaisten sovellusten kehittämisen kulutuskanavista riippumattomasti ja tukee niiden tarjoamista useiden kanavien, kuten web-sivun, mobiilisovelluksen tai työpöytäsovelluksen, kautta.

Kerroksen tukemat kyvykkyyskategoriat ovat 1) Kuluttajapalvelut, 2) Esityspalvelut, 3) Backend-integrointi, 4) Välimuisti ja suoratoistosisältö (engl. streaming content), 5) Turvallisuus ja yksityisyys sekä 6) Pääsy informaatioon. Kategoriat 3, 5 ja 6 toteutetaan kokonaan muiden kerrosten rakennuspaloilla.

Esimerkkejä kerroksen rakennuspaloista on esitetty alla. Kyvykkyyskategorian numero on suluisissa rakennuspalan kuvauksen jälkeen.

- **Kuluttaja** (engl. Consumer) on ihminen tai järjestelmä. (1)
- **Asiakas/Kanava** (engl. Client/Channel) on elementti, jonka kanssa kuluttaja on vuorovaikutuksessa, kuten web- tai mobiilisovellus. (1)

- **Esitysohjain** (engl. Presentation Controller) muodostaa asiakasnäkymän muiden rakennuspalojen avulla. (2)
- **Kuluttajaprofilimanageri** (engl. Consumer Profile Manager) tukee rajapinnan ja esityksen personointia yksittäiselle kuluttajalle. (2)

Integraatiokerros

Integraatiokerros tukee SOAn elementtien vuorovaikutusta ja löyhää kytkentää heterogeenisessa SOA-ympäristössä. Kerros tukee mm. epäsuoraa sidontaa elementtien välillä sekä erilaisia kommunikointia tukevia kyvykkyyskäsitteitä, kuten viestien välittämistä, reitittämistä ja muuntamista sekä protokollan konvertoimista. Kerroksen kyvykkyyskäsitteet toteutetaan usein palveluväylän (engl. enterprise service bus, ESB) avulla.

Kerroksen tukemat kyvykkyyskäsitteet ovat 1) Kommunikaatio, palveluvuorovaikutus ja integraatio, 2) Viestin prosessointi, 3) Palvelutaso, 4) Turvallisuus sekä 5) Hallinta. Turvallisuus-kyvykkyyskäsitteet toteutetaan kokonaan Palvelutasokerroksen (engl. Quality of Service Layer) avulla.

Esimerkkejä kerroksen rakennuspalloista on esitetty alla. Kyvykkyyskäsitteiden numero on suluisissa rakennuspalon kuvauksen jälkeen.

- **Adaptteri** (engl. Adapter) tukee SOAn ulkopuolisiin järjestelmiin ja komponentteihin kohdistuvaa vuorovaikutusta. Se tukee reitittämistä, viestin muuntamista ja protokollan konvertoimista kaikkien kerrosten rakennuspalojen ja SOAn ulkopuolisten kohteiden välillä. (1)
- **Eriaikaisen viestinnän manageri** (engl. Asynchronous Messaging Manager) tukee luotettavaa eriaikaista viestintää esimerkiksi viestijonon avulla. (1)
- **Välittäjä** (engl. Mediator) käsittelee palvelujen pyynnöt ja vastaukset. Se käyttää Reitittäjää, Viestin muuntajaa (engl. Message Transformer), Protokollan konvertoijaa sekä Datan yhdistelijää (engl. Data Aggregator). Se tukee muun muassa staattista palvelujen yhdistelyä. (1)
- **Tapahtuman välittäjä** (engl. Event Broker) (2)
- **Transaktiomanageri** (3)

- **Lokin kirjoittaja** (engl. Logger) tukee järjestelmän poikkeuksien ja vakauden monitorointia pitämällä kirjaa viestin reitittämistä ja palvelukutsua koskevista aktiviteeteista. (5)

Palvelutasokerros

Palvelutasokerros tukee erilaisten ei-toiminnallisten vaatimusten toteutumisen tarkkailemista ja hallintaa SOAssa sekä liiketoiminnan että IT:n tasolla. Nämä vaatimukset koskevat mm. turvallisuutta, IT-järjestelmien tilaa, liiketoiminta-aktiviteetteja ja politiikan noudattamista. Tarkkailun keinoja ovat mm. monitorointi, poikkeamien raportointi ja lokin tuottaminen.

Kerroksen tukemat kyvykkyyskategoriat ovat 1) Komentojen ja kontrollin hallinta, 2) Turvallisuuden hallinta, 3) IT-järjestelmien monitorointi ja hallinta, 4) Sovellusten ja SOAn monitorointi ja hallinta, 5) Liiketoiminta-aktiviteetin monitorointi ja hallinta, 6) Tapahtumien hallinta, 7) Poliitiikan monitorointi ja pakottaminen, 8) Konfiguraation ja muutoksen hallinta sekä 9) Datan varastointi. Datan varastointi -kyvykkyyskategoria toteutetaan kokonaan Hallintokerroksen avulla.

Esimerkkejä kerroksen rakennuspaloista on esitetty alla. Kyvykkyyskategorian numero on suluissa rakennuspalan kuvauksen jälkeen.

- **Pääsyn valvoja** tukee muun muassa palvelukutsun ja viestin reitittämisen todentamista ja valtuutusta sekä datan pääsyoikeuksia. Sitä käytetään referenssiarkkitehtuurin kaikissa muissa kerroksissa. (2)
- **Ratkaisumanageri** (engl. Solution Manager) koordinoi ratkaisun hallintaa – esim. elinkaarta, turvallisuutta, saatavuutta, konfiguraatiota ja muutosta. (4)
- **Monitorointimetriikkatyökalut** (engl. Monitoring Metric Tools) mittaa, kerää ja arvioi metriikoita säännöllisesti palveluista ja keskeisistä prosesseista. Se testaa metriikoita politiikkoja vasten. (4)
- **Liiketoiminta-aktiviteettimonitori** (engl. Business Activity Monitor) monitoroi tapahtumia, liiketoimintaprosessien liiketoiminta-aktiviteetteja sekä palveluja. (5)

- **Politiikan pakottajaa** (engl. Policy Enforcer) käytetään Hallintokerroksessa määriteltyjen politiikkojen pakottamiseen kaikissa muissa kerroksissa, näiden kerrosten politiikanpakottamiskohdissa. (7)

Informaatiokerros

Informaatiokerros (engl. Information Layer) tukee erilaisia informaation hallintaa koskevia kyvykkyyksiä, kuten informaatiopalvelujen toteuttamista, informaation yhdistelemistä erilaisista lähteistä, metadatan hallintaa, liiketoiminta-analytiikkaa, datan varastointia erilaisissa muodoissa sekä tapahtumien ja yrityksen sanastojen ja tietomallien määrittämistä.

Kerroksen tukemat kyvykkyysskategoriat ovat 1) Informaatiopalvelut, 2) Informaation integrointi, 3) Perusinformaationhallinta, 4) Informaation turvallisuus ja suojaaminen, 5) Liiketoiminta-analytiikka, 6) Informaation määrittäminen ja mallintaminen sekä 7) Informaatiovarasto.

Esimerkkejä kerroksen rakennuspaloista on esitetty alla. Kyvykkyysskategorian numero on suluisissa rakennuspalan kuvauksen jälkeen.

- **Informaatiopalvelukäytävä** (engl. Information Services Gateway) toimii pääsykohtana kerrokseen. Se tukee informaation paljastamista palveluina ja valvoo pääsyä dataan. (1)
- **Datan yhdistelijä** (engl. Data Aggregator) yhdistelee dataa erilaisista datalähteistä. Se tukee yhtenäisen datanäkymän tai -mallin luomista. Se käyttää useita muita rakennuspaloja, kuten Datan virtualisointimanageria, Datan validoijaa sekä Datan laatumanageria. (1)
- **Sisältömanageri** (engl. Content Manager) tukee strukturoimattoman sisällön, kuten kuvien, tekstidokumenttien ja web-sivujen, hallintaa. Se tukee mm. sisällön tallentamista, hakemista ja turvaamista. (3)
- **Master datan hallintaympäristö** (engl. Master Data Authoring Environment) (3)
- **Liiketoimintatapahtuma** (6)

- **Datavarasto** edustaa mm. analyttisen datan varastoa, operationaalisen datan varastoa, master data -varastoa, strukturoimattoman datan varastoa ja metadatavarastoa. (7)

Hallintokerros

Hallintokerros edustaa kyvykkyyskäsitteitä, jotka tukevat SOAn mukauttamista yrityksen strategioiden ja tavoitteiden toteuttamiseksi asetettuihin politiikkoihin, ohjeistuksiin ja standardeihin sekä SOA-ratkaisujen liiketoiminta-arvon toteuttamista. Kerros tukee hallintoprosesseja, joita ovat noudattamisprosessi, joka pyrkii varmistamaan että SOAn politiikkoja, ohjeistuksia ja standardeja noudatetaan; vapautusprosessi, joka käsittelee poikkeamia noudattamisessa; sekä kommunikaatioprosessi, joka edistää SOA-hallinnon ymmärtämistä yrityksessä. Hallintoa sovelletaan sekä palvelujen että SOA-ratkaisujen elinkaaren ja portfolion hallintaprosesseihin, mikä koskettaa referenssiarkkitehtuurin kaikkia muita kerroksia.

Kerroksen tukemat kyvykkyyskategoriat ovat 1) Hallinnon suunnitteleminen, 2) Hallinnon määrittäminen, 3) Hallinnon mahdollistaminen ja implementointi, 4) SOA-metadatan varastointi ja hallinta, 5) Liiketoimintasäännön määrittäminen ja hallinta, 6) Poliitiikan määrittäminen ja hallinta, 7) Monitorointi, 8) Hallinta sekä 9) Työnkulku.

Esimerkkejä kerroksen rakennuspalloista on esitetty alla. Kyvykkyyskategorian numero on suluissa rakennuspalan kuvauksen jälkeen.

- **Palveluvarasto** on erityinen omaisuusvarasto, joka tukee palvelu- ja hallintoartefaktien, kuten palvelurajapintojen, politiikkojen, versioinformaation ja hallintoprosessien, suunnitteluajasta varastointia ja jakamista. (1–4)
- **Palvelurekisteri** tukee palvelujen ajoajasta mainostamista (engl. advertise), löytämistä, sidontaa ja virtualisointia. Se voi käyttää Palveluvaraston dataa. (3, 4)
- **Liiketoimintasääntö** rajoittaa jotain liiketoiminnan aspektia, kuten prosessia, palvelua tai informaatiota. (5)

- **Liiketoimintasääntömanageri** (engl. Business Rule Manager) tukee muun muassa liiketoimintasääntöjen määrittämistä ja jakelua sekä niiden ja politiikkojen yhdenmukaistamista. (2, 5)
- **Politiikka** (6)
- **Politiikkamanageri** tukee mm. politiikan määrittämistä, jakelua ja ylläpitoa. (2, 3, 6)
- **Kojelauta** (engl. Dashboard) tarjoaa tosiaikaisen näkymän sekä hallintoprosessien että hallinnoitavien prosessien tilasta. (7)

4.2.3 Mallin soveltaminen

Referenssiarkkitehtuurin ensisijainen tarkoitus on SOAn toteuttamisen tukeminen ja se on näin ollen suunnattu ensisijaisesti SOAa omaksuville organisaatioille. Referenssiarkkitehtuuri on keskittynyt vahvasti SOAn kyvykkyyksien, rakennuspalojen ja näiden välisten yhteyksien mallintamiseen. Tällä abstraktilla lähestymistavalla referenssiarkkitehtuuri pyrkii tukemaan SOAn toteuttamista monin tavoin. Ensiksi, referenssiarkkitehtuuri pyrkii tukemaan SOAn suunnittelua vaatimuslähtöisesti sekä SOAn mukauttamista liiketoiminnallisiin ja muuttuviin teknisiin vaatimuksiin. Toiseksi, se pyrkii tukemaan SOA-toteutuksen loogista, tuotteista riippumatonta suunnittelua sen sijaan, että SOAn toteuttaminen perustuisi tietyn tuotteen tarjoamiin kyvykkyyksiin. Kolmanneksi, se pyrkii tukemaan implementaatiopäätösten siirtämistä myöhäiseen vaiheeseen SOAn toteuttamisessa. Kyvykkyyksien ja rakennuspalojen lisäksi referenssiarkkitehtuurissa tarkastellaan joitain sen suunnitteluvaihtoehtoja ja implementoimista koskevia seikkoja, joskin vähäisessä määrin. Referenssiarkkitehtuurin mukaan sillä on merkittävä osa SOAn arvolupausten, kuten kustannusten vähentämisen, ketteryyden ja kilpailukyvyn lisäämisen, toteuttamisessa. Se ei kuitenkaan ole tarkoitettu normatiiviseksi, vaan suuntaa-antavaksi malliksi. SOAn toteuttamisen tukemisen ohella referenssiarkkitehtuuri on tarkoitettu tukemaan SOA-arkkitehtuurin arviointia sekä kommunikointia. [23]

Referenssiarkkitehtuuri on suunnattu SOAa omaksuvien organisaatioiden lisäksi SOA-tuotteiden toimittajille, jotka voivat käyttää sitä referenssiarkkitehtuurin kanssa

yhdenmukaisten tuotteiden suunnitteluun. Spesifikaation kohdeyleisönä mainitaan myös integraattorit ja standardointijärjestöt. [23]

Referenssiarkkitehtuurin eräs sovelluskohde on IBM:n kehittämä pilvipalvelureferenssiarkkitehtuuri (Cloud Computing Reference Architecture, CCRA), jonka IBM on luovuttanut The Open Groupille standardointityöhön. CCRA:n mukaan pilvipalveluarkkitehtuurit ovat palvelukeskeisiä arkkitehtuureja, joilla on lähtökohtaisesti samat ominaisuudet kuin SOA:lla, mutta pilvipalveluarkkitehtuureilla on lisäksi tiettyjä erityisominaisuuksia. SOA-referenssiarkkitehtuuri toimii arkkitehtuurillisena perustana CCRA:lle, jossa kuvatut elementit sijoittuvat SOA-referenssiarkkitehtuurissa kuvattuihin loogisiin kerroksiin. [48]

4.2.4 Havainnot

The Open Groupin referenssiarkkitehtuuri kuvaa laaja-alaisesti SOAn rakennetta teknisestä ja loogisesta näkökulmasta. Se kuvaa, millaisista elementeistä SOA koostuu, mikä niiden tarkoitus on, miten ne toimivat yhdessä ja miten ne ryhmittyvät loogisesti keskenään. Referenssiarkkitehtuuri on näin ollen varsin rakennekeskeinen verrattuna OASISin referenssimalliin ja referenssiarkkitehtuuriin, jotka keskittyvät kuvaamaan SOAn erinäisiä ominaisuuksia hyvin käsitteellisestä näkökulmasta. Tosin OASISin referenssiarkkitehtuurin arkkitehtuurilliset seuraukset keskittyvät hyvin samankaltaisiin toteutuksellisiin seikkoihin kuin The Open Groupin referenssiarkkitehtuuri mutta huomattavasti heikommin jäsennellyllä tavalla. Myös ontologia on The Open Groupin referenssiarkkitehtuuriin nähden hieman käsitteellisempi. The Open Groupin referenssiarkkitehtuuri on näin ollen tutkittavista malleista kaikkein konkreettisin, vaikka myös se on tekniikoista riippumattomana abstrakti malli.

The Open Groupin referenssiarkkitehtuuri noudattaa kauttaaltaan melko yhtenäistä näkökulmaa ja muotoa, ja se on myös hyvin jäsennelty. Näiden seikkojen ja referenssiarkkitehtuurin verrattain konkreettisen luonteen vuoksi sen tarkastelualue on huomattavasti helpommin hahmotettavissa kuin OASISin referenssiarkkitehtuurin. The Open Groupin referenssiarkkitehtuurin tarkastelualue muistuttaa jossain määrin OASISin referenssiarkkitehtuurin *SOA-ekosysteemin toteuttaminen* ja *Omistajuus SOA-*

ekosysteemissä -näkyvien tarkastelualueita, mutta erilaisten näkökulmien vuoksi mallien sisällöt poikkeavat huomattavasti toisistaan. Jälkimmäisen näkymän SOA-testaus-malli ja sosiaalisia rakenteita kuvaava *Osallistuminen SOA-ekosysteemissä* -näkyvä ovat kuitenkin lähes kokonaan The Open Groupin referenssiarkkitehtuurin tarkastelualueen ulkopuolella. Referenssimallilla ja ontologialla on The Open Groupin referenssiarkkitehtuurin kanssa vain vähän yhteisiä käsitteitä, sillä referenssimalli ja ontologia kuvaavat pääasiassa SOAn muita ominaisuuksia ja yksityiskohtia.

Referenssiarkkitehtuuri pyrkii ottamaan huomioon SOAn toteuttamisen eri vaiheet. Siinä tarkastellaan erikseen SOAn vaatimuksia, loogisia rakennuspaloja sekä hieman implementointia koskevia seikkoja. Referenssiarkkitehtuuri pyrkii näin ollen tukemaan monipuolisesti SOAn elinkaarta vaatimusten määrittämisestä implementointiin ja vaatimusmuutoksiin mukauttamiseen.

Referenssiarkkitehtuurin kerrosjako on SOAn monimutkaiseen rakenteeseen nähden yksinkertaistettu ja selväpiirteinen, mikä tukee SOAn ominaisuuksien ja merkittävien kyvykkyyskokonaisuuksien ymmärtämistä ja SOAn loogisen rakenteen hahmottamista. Mallin läheisempi tarkastelu osoittaa kuitenkin siinä joitain heikkouksia. Ensiksi, kerrosrakenne ei kuvaa hyvin palvelun kuluttamisen rekursiivista luonnetta: useat erilaiset elementit voivat kuluttaa palveluja, ja palvelut voivat puolestaan kuluttaa toisia palveluja. Lisäksi SOAssa prosessit ovat tyypillisesti myös palveluja. Toiseksi, kerrosten keskinäiset suhteet noudattavat melko löyhästi kuvattua kerrosjakoa. Kolmanneksi, kerrosten kyvykkyudet ovat monin paikoin päällekkäisiä. Useat kerrosten kyvykkyudet ja jopa kyvykkyyskategoriat toteutetaan toisten kerrosten rakennuspaloilla. Neljänneksi, joidenkin kyvykkyyskokonaisuuksien sijoittaminen kyseiseen kerrosjakoon on ongelmallista. Esim. politiikkoihin ja tapahtumiin liittyvät kyvykkyudet ovat hajautuneet useiden kerrosten kesken. Kerroksista voidaan havaita, että horisontaalisten kerrosten kyvykkyudet ovat pääsääntöisesti yhtenäisempiä kokonaisuuksia kuin vertikaalisten. Viidenneksi, referenssiarkkitehtuuri kuvaa sekä suunnittelu- että ajoaikaisia elementtejä, mikä vaikeuttaa referenssiarkkitehtuurin ymmärtämistä. Tämä ilmenee erityisesti Operationaaliset järjestelmät -kerroksessa, joka edustaa muiden kerrosten ajoaikaista näkymää mutta toisaalta kuvaa myös muiden

kerrosten rakennuspalojen tukemiseen tarvittavia elementtejä ja lisäksi käyttää eräitä muiden kerrosten rakennuspaloja. Kerrosjako kuvaa näin ollen melko karkeasti SOAn monimutkaista rakennetta.

Referenssiarkkitehtuuri on kokonaisuudessaan varsin toteutusorientoitunut SOA-malli. Muihin tässä tutkielmassa tarkasteltuihin SOA-malleihin nähden se on melko käytännönläheinen ja sen ensisijainen käyttötarkoitus – SOA-toteutuksen loogisen, vaatimuslähtöisen suunnittelun tukeminen – on ilmeinen. Referenssiarkkitehtuurilla on myös potentiaalia SOAn ymmärtämistä tukevana mallina. Korkean tason kuvaus SOAn loogisista kerroksista kuvaa SOAn keskeisiä piirteitä ja kyvykkyyskokonaisuuksia verrattain helposti omaksuttavalla tavalla, joskin yksinkertaisuudessaan väistämättä karkeasti. Muista tutkittavista SOA-malleista poiketen referenssiarkkitehtuuri pyrkii myös kuvaamaan kattavasti SOAn teknistä rakennetta. Kuvaus on kuitenkin hyvin monimutkainen – referenssiarkkitehtuurissa määritetään yhteensä 251 kyvykkyyttä ja 140 rakennuspalaa – minkä vuoksi malli sopii tässä tarkoituksessa vain SOAn syvällisen ymmärtämisen tukemiseen.

Luku 5

SOA-tuotteet

Tässä tutkielmassa tarkasteltavat SOA-mallit ovat abstrakteja, yleisiä ja tekniikoista riippumattomia malleja. SOAn tekninen toteuttaminen edellyttää useiden arkkitehtuurillisten ja teknisten valintojen tekemistä. Tangin et al. [13] esittämän luokittelun mukaan SOA-arkkitehtuurityyli voidaan jakaa kuuteen alatyyliin, joiden tekniset toteutukset, hyödyt ja heikkoudet poikkeavat toisistaan. Nämä tyylit ovat 1) WSDL/SOAP-palvelutyypin ja web service -spesifikaatioihin perustuva SOA, 2) REST-palvelutyypin ja Web 2.0:aan perustuva SOA, 3) tapahtumaohjattu SOA, 4) komponenttitekniikoihin, kuten SCA, perustuva SOA, 5) grid-laskentaa hyödyntävä SOA sekä 6) hybridityyli, joka yhdistelee edellä mainittuja tyylejä. Kukin näistä tyyleistä asettaa erityisiä vaatimuksia SOAn infrastruktuurille.

Monet ohjelmistovalmistajat ovat kehittäneet SOAa tukevia alustoja. Nämä tukevat tyypillisesti useiden edellä mainittujen SOA-tyylien ominaisuuksia. Laajimpiin SOA-alustoihin kuuluvat Oraclen Fusion Middleware -tuoteperheeseen perustuva SOA Suite, IBM:n WebSphere-tuoteperheeseen perustuva SOA Foundation sekä Microsoftin BizTalk-palvelimeen perustuva SOA-alusta. Muita huomattavia SOA-alustatoimittajia ovat mm. TIBCO, avoimen lähdekoodin JBoss-alustaa kehittävä Red Hat sekä SAP. [49] [50] SOA-mallien konkreettisena vertailukohtana käytetään tässä tutkielmassa Oraclen laajaan SOA-tuotevalikoimaan perustuvaa SOA-ympäristöä. Oraclen välikerrosohjelmistotuoteperhe, Oracle Fusion Middleware, sisältää useita SOAn toteuttamista tukevia ratkaisuja, joista keskeisin on SOA Suite. Muut Oraclen SOAan liittyvät ratkaisut tarjoavat edistyneisempää lisätoiminnallisuutta.

Tämän luvun tuotekuvaukset perustuvat lukuisiin Oraclen web-sivuihin ja näiden kautta saatavilla oleviin white paper ja data sheet -aineistoihin. Keskeisimmät näistä ovat Oracle SOA Suite 11g white paper [51], Oracle Technology Networkin SOA Suite -sivusto [52] sekä SOA Suiten dokumentaationsivusto [53]. Yksittäisiä lähteitä ei ole mainittu tuotekuvauksissa.

5.1 Oracle SOA Suite

Oracle SOA Suite koostuu monista SOAn toteuttamista ja hallitsemista tukevista tuotteista. Se kattaa myös Oraclen kahden muun tuotesarjan, BPM Suiten (Business Process Management) sekä EDA Suiten (Event-Driven Architecture), tuotteet. SOA Suite pyrkii erottumaan kilpailevista SOA-alustoista kattavalla, integroidulla ja muiden toimittajien tuotteiden kanssa yhteensopivalla tuotekokonaisuudella. Tuotteiden integroituneisuudella tarkoitetaan useaa seikkaa. Ensiksi, SOA Suite tukeutuu yhtenäiseen kehitysympäristöön: SOA Suitea tukeva JDeveloper-kehitysympäristö on tarkoitettu IT- ja liiketoimintahenkilöstön yhteiseksi työkaluksi, jota voidaan käyttää SOAn toteuttamisen kaikissa vaiheissa. Toiseksi, SOA Suite tukeutuu yhtenäiseen suoritusympäristöön: SOA Suite voidaan suorittaa kokonaan Oraclen WebLogic-sovelluspalvelimella. Kolmanneksi, SOA Suite sisältää keskitetyn hallinta- ja monitorointityökalun, Oracle Enterprise Managerin, jonka avulla palvelimia ja niillä suoritettavia sovelluksia voidaan hallita ja monitoroida. SOA Suite on tarkoitettu toimivaksi useiden muiden toimittajien ja avoimen lähdekoodin välikerrosohjelmistojen – kuten sovelluspalvelimien ja sääntömoottoreiden – kanssa. SOA Suite pyrkii myös tukemaan avoimia standardeja.

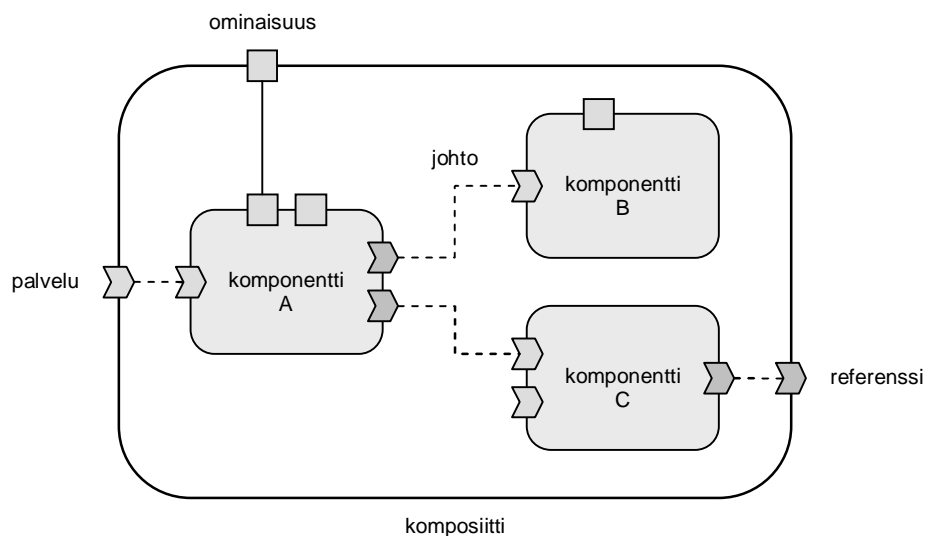
Seuraavissa kohdissa esitetään SOA Suiten käyttämä sovelluskehitysmalli SCA, SOA Suiten sisältämät komponentit sekä eräät SOA Suitteen läheisesti liittyvät tuotteet.

SCA-sovelluskehitysmalli

Palvelupohjaisten sovellusten kehittäminen SOA Suitessa perustuu SCA-malliin. SCA on yhteiseen suoritusympäristöön perustuva malli, joka pyrkii helpottamaan palvelukomponenttien implementoimista eri tekniikoilla ja komponenttien yhdistelyä

komposiittisovelluksiksi. SCA-sovellusten kehittäminen ja suorittaminen edellyttää yhteistä SCA-suoritusympäristöä – tässä tapauksessa Oracle WebLogic SCA Runtime - suoritusympäristöä – mikä vahvistaa komponenttien välistä kytkentää mutta salli sovelluskehittäjien hallita komponentteja ja niiden yhdistelmiä verrattain tehokkaasti. SCA:n suoritusympäristöstä riippuva palvelumalli poikkeaa siis oleellisesti tekniikkaneutraaleista ja heterogeenisiin ympäristöihin soveltuvista palvelumalleista. SCA-sovellukset voidaan kuitenkin integroida myös SCA-suoritusympäristön ulkopuolisiin palveluihin, sovelluksiin ja resursseihin, minkä vuoksi SCA:ta voidaan käyttää yhdessä muiden palvelutekniikoiden, kuten WSDL/SOAP- ja REST-tekniikoiden kanssa.

SCA:n palvelumalli keskittyy vahvasti palvelukomponenttiin ja sen ominaisuuksiin, toisin kuin tekniikkaneutraalit palvelumallit, jotka keskittyvät standardin mukaiseen palvelurajapintaan. SCA:n palvelumallissa komponenteilla voi olla muille komponenteille paljastettujen toimintojen eli *palvelujen* lisäksi *referenssejä* muihin komponentteihin sekä *ominaisuuksia* (engl. property), joiden arvot SCA-sovelluksen kehittäjä voi asettaa. Komponentteja voidaan yhdistellä komposiiteiksi yhdistämällä komponenttien referenssejä ja palveluja toisiinsa ns. *johdoilla* (engl. wire). Palveluja, referenssejä ja ominaisuuksia voidaan paljastaa myös komposiitin ulkopuolelle, jolloin komposiittia voidaan käyttää komponenttina toisessa komposiitissa. Lisäksi SCA tukee sidontamekanismien ja politiikkojen määrittämistä. SCA:n komponentteihin ja komposiitteihin perustuva palvelumalli on esitetty kuvassa 5.1.



Kuva 5.1. SCA-komposiitti.

SCA on alun perin usean yrityksen – mukaan lukien Oracle, IBM ja SAP – muodostaman Open SOA -yhteisön kehittämä spesifikaatiojoukko, jonka jatkokehittäminen luovutettiin OASISille vuonna 2007. SCA-spesifikaatioissa on määritelty tuki tietyille implementaatiotekniikoille, kuten Javalle, C++:lle ja BPEL:lle, sekä sidontatekniikoille, kuten web service -sidonnalle, EJB Session Bean -sidonnalle sekä JMS-sidonnalle (Java Message Service) [54]. SCA-sovelluksia voidaan kuitenkin käytännössä kehittää vain käytettävän SCA-suoritusympäristön tukemilla tekniikoilla. Oraclen SCA-suoritusympäristön tukemat komponenttitekniikat ovat BPEL, Spring sekä Oraclen omat komponenttityypit, jotka ovat Business Rule, Human Task ja Mediator.

Oracle Service Bus

Oracle Service Bus (OSB) on Oraclen palveluväylätoteutus. Se toimii yhdistäjänä, välittäjänä sekä vuorovaikutuksen hallitsijana yrityksen elementtien, kuten palvelujen, perinnejärjestelmien (engl. legacy system), ERP-tuotteiden ja muiden ESB-instanssien, välillä. OSB tukee mm. monipuolisia viestintäominaisuuksia, viestinnän monitorointia, SLA:n noudattamisen valvontaa, erilaisia turvallisuusominaisuuksia, palvelun virtualisointia sekä useita palveluversioita. OSB voidaan integroida UDDI-standardin mukaisen palvelurekisterin kanssa.

Oracle BPEL Process Manager

BPEL Process Manager tukee BPEL-prosessien suunnittelua, suorittamista ja hallintaa. BPEL Process Manager sisältää prosessimoottorin, jolla voidaan suorittaa BPEL-standardin mukaisia prosessikuvauksia. Oracle BPEL Process Designer on JDeveloperin ohjelmalisäke, joka tukee BPEL-prosessin suunnittelua ja toteuttamista SCA-komponenttina. BPEL Process Manager tukee myös ihmistyönkulun (engl. human workflow) sisällyttämistä prosesseihin.

Oracle Business Activity Monitoring

Business Activity Monitoring (BAM) on liiketoimintaprosessien ja -palvelujen monitorointityökalu ylimmän johdon ja keskijohdon henkilöille. Sen avulla voidaan kehittää mm. tosiaikaisia datavirtoja kuvaavia kojelautoja; sääntöjä, joiden perusteella voidaan lähettää varoituksia tietyissä olosuhteissa esimerkiksi sähköpostilla; sekä muuttuvaan dataan tai tapahtumiin (esimerkiksi SLA:ta tai KPI:tä koskeviin) perustuvia raportteja.

Oracle Business Rules

Oracle Business Rules (OBR) tukee liiketoimintasääntöjen luomista, hallintaa erillään sovelluksista ja käyttämistä. Liiketoimintasääntöjä voidaan käyttää päätöskomponentteina SCA-sovelluksessa, BPEL-prosessissa tai kirjastona Java-sovelluksessa. OBR:in Rules Engine -moottori käsittelee liiketoimintasääntöjä ja tuottaa dynaamisesti päätöksiä, jotka ohjaavat prosessien ja sovellusten suorittamista päätöskohdissa. Muita OBR:in komponentteja ovat muun muassa liiketoimintasääntöjen luomiseen, tarkasteluun ja muokkaamiseen tarkoitettu JDeveloperin Rules Designer -lisäke sekä liiketoimintasääntöjä käyttävän Java-sovellusten kehittämiseen tarkoitettu Rules SDK (software development kit) -kirjasto.

Oracle Complex Event Processing

Complex Event Processing (CEP) on tapahtumaohjattujen sovellusten toteuttamiseen tarkoitettu ympäristö. Se tukee monimutkaisten tapahtumakyselyiden tekemistä useista tapahtumavirroista ja uusien, ns. huomattavien tapahtumien tuottamista näiden

kyselyiden perusteella. CEP:in avulla voidaan toteuttaa myös modulaarisia tapahtumankäsittelyverkkoja (engl. event processing network), jotka voivat sisältää hyvin monimutkaista hierarkkista tapahtumankäsittelylogiikkaa. CEP sisältää SQL-kieleen (Structured Query Language) perustuvan CQL-kyselykielen (Oracle Continuous Query Language), jonka avulla voidaan tehdä jatkuvia kyselyjä useista tapahtumavirroista. CEP tukee myös tapahtumainformaation visualisointia, kuorman simulointia sekä komponenttitason vasteajan monitorointia.

Oracle Adapters

Adapters sisältää useita JCA:han (Java Connectivity Architecture) perustuvia adaptoreita, joiden avulla sovelluksia ja muita resursseja voidaan integroida SOA-sovelluksiin. Adaptoreita voidaan käyttää komponenttina SCA-sovelluksissa. Adaptoreihin sisältyy mm. tiedostoja ja tietokantoja tukevia tekniikka-adaptoreita; perinne- ja suurkaneyjärjestelmiä (engl. mainframe system) tukevia legacy-adaptoreita; pakattujen sovellusten, kuten SAP-toiminnanohjausjärjestelmän, adaptoreita; sekä Oracle-sovellusten adapteri. Adapterit tukevat resurssien käyttämistä erilaisilla rajapinnoilla, kuten JCA- tai WSDL-rajapinnalla.

Oracle Mediator

Mediator on välittäjänä toimiva komponentti, jota käytetään osana SCA-sovellusta. Se tukee mm. reitittämistä, muuntamista, validointia ja virheen käsittelyä sekä erilaisia viestinvaihtokuvioita, kuten samanaikaista (engl. synchronous) ja eriaikaista viestintää sekä tapahtuman julkaisua ja tilaamista. Mediator on tarkoitettu käytettäväksi komposiittisovelluksessa toisin kuin OSB, jota voidaan käyttää yrityksen laajuisena integrointi- ja palveluvirtualisointityökaluna.

Oracle Human Workflow

Human Workflow tukee ihmistyön sisällyttämistä SCA-sovellukseen ja BPEL-prosessiin. Human Workflow tukee mm. tehtävien luomista, antamista suoritettavaksi, aikarajoja, muutoksista ilmoittamista, organisointia sekä priorisointia. Tehtävät voidaan esittää käyttäjille esimerkiksi Oracle BPM Worklist -sovelluksen tai sähköpostin avulla. Human Task -SCA-komponentit suoritetaan Human Task Service Engine -moottorilla.

Oracle B2B

B2B on komponentti, jota käytetään kommunikointiin partnereiden tietojärjestelmien kanssa. Se tukee viestintää ja dokumenttien vaihtoa, ja sen avulla voidaan toteuttaa prosesseja, jotka sisältävät yhteistyötä partnereiden kanssa. Sitä voidaan käyttää adapterin avulla SCA-sovelluksissa. B2B tukee lisäksi mm. dokumenttien hallintaa ja muunnoksia, turvallista ja luotettavaa viestintää eri protokollilla, partnereiden profiilien ja sopimusten hallintaa, raportointia sekä monitorointia.

Oracle User Messaging Service

User Messaging Service (UMS) tukee kaksisuuntaista viestintää SOA Suiten ja käyttäjien välillä useiden kanavien, kuten sähköpostin, SMS:n ja pikaviestinnän, välityksellä. Se integroituu useiden SOA Suiten komponenttien, kuten BPEL-prosessien, Human Workflow'n ja BAM:in, kanssa.

Oracle Metadata Repository

Metadata Repository on Fusion Middleware -tuoteperheen komponentti, jota käytetään useiden SOA Suiten komponenttien metadatan ja konfiguraatiodatan varastointiin. Siihen voidaan varastoida mm. tapahtumia, liiketoimintasääntöjä sekä XSLT- (Extensible Stylesheet Language Transformations), XSD- (XML Schema Definition) ja WSDL-tiedostoja. Metadata Services (MDS) on erityinen varasto tietyyntyyppisten sovellusten, kuten SCA-sovellusten, kustomointimetadatan varastointiin.

Oracle Web Services Manager

Oracle Web Services Manager (OWSM) tukee erityisesti turvallisuutta koskevien palvelupolitiikkojen hallintaa ja pakottamista. Se tukee mm. todentamis- ja valtuutuspolitiikkojen määrittämistä, identiteetin propagointia ja viestin salaamista. Politiikkoja hallitaan ja niiden noudattamista ja politiikkatapahtumia monitoroidaan Enterprise Managerin avulla. Politiikkoja voidaan asettaa suunnitteluajankana JDeveloperin avulla ja ajoaikana Enterprise Managerin avulla. Politiikat säilytetään keskitetysti WSM Repository -varastossa ja jaellaan pakottamispisteisiin agenteille

(WSM Agent). OWSM tukee myös vaikutusanalyysin suorittamista ennen politiikkamuutosten tekemistä sekä politiikan liittämistä WSDL-tiedostoon.

Oracle Enterprise Manager

Oracle Enterprise Manager (OEM) on Fusion Middleware -tuoteperheen keskitetty, selainkäyttöinen hallinta- ja monitorointityökalu, jolla voidaan hallita ja monitoroida SOA-sovelluksia ja SOA-infrastruktuuria. Sen avulla voidaan monitoroida ja tarkastella mm. palvelimia, moottoreita, SCA-sovelluksia, SCA-komponentteja, palveluja, palveluriippuvuuksia, prosesseja, transaktioita, viestintää, poikkeuksia, tapahtumia, suorituskykyä sekä saatavuutta. Se integroituu muihin SOA Suiten tuotteisiin, kuten BAM:iin, BPEL Manageriin, Web Services Manageriin ja OSB-konsoliin. Se sisältää konfiguraatioinformaatiovaraston, johon tallennetaan SOA Suiten, palveluväylän ja BPEL-prosessien konfiguraatioinformaatio. OEM tukee myös sijoittamisen hallintaa.

Oracle JDeveloper

JDeveloper on Fusion Middleware -tuoteperheen pääasiallinen kehitysalusta. Se on ilmainen integroitu Java-kehitysympäristö, joka on tarkoitettu suunnittelu- ja kehitystyökaluksi SOAn toteuttamisen kaikissa vaiheissa. Se on suunnattu mm. ohjelmistokehittäjille, arkkitehdeille ja liiketoiminta-analyytikoille. JDeveloperin avulla voidaan kehittää, sijoittaa ja testata mm. SCA-komponentteja, SCA-sovelluksia, BPEL-prosesseja, liiketoimintasääntöjä, web-palveluja, EJB-komponentteja sekä web-sovelluksia.

5.2 Muut Oraclen SOA-ratkaisut

Oracle tarjoaa SOA Suiten lisäksi useita muita ratkaisuja SOAn tukemiseen. Merkittävimmät näistä ovat Oracle SOA Governance, Data Integration, Enterprise Gateway, BPA Suite sekä Application Integration Architecture.

5.2.1 Oracle SOA Governance

Oracle SOA Governance on Oraclen ratkaisu SOAn hallintoon. Se on keskeinen osa Oraclen SOA-tuotevalikoimaa, vaikka sitä tarjotaan SOA Suitesta erillisenä ratkaisuna. Oracle SOA Governance keskittyy SOAan liittyvän omaisuuden (engl. asset) hallintaan

ja näkyvyyteen sen koko elinkaaren aikana, hallintoprosessien automatisointiin sekä SOA-toteutuksen edistymisen ja sen tuottaman arvon näkyvyyteen. Oracle SOA Governance sisältää seuraavat komponentit:

- **Enterprise Repository** tukee SCA-sovellusten, palvelujen, prosessien ja muun IT-omaisuuden metadatan suunnitteluajasta tallennusta ja hallintaa niiden koko elinkaaren aikana. Se on tarkoitettu keskitetyksi, useiden käyttäjäryhmien käyttämäksi varastoksi kaikelle SOA-informaatiolle. Se sisältää mm. hallintoprosesseja suorittavan prosessimoottorin, valmiita muokattavia hallintoprosesseja sekä varaston ja rekisterin välisen synkronointiapuvälineen.
- **Service Registry** on Enterprise Repositoryn kanssa tiiviisti integroitu, UDDI-standardin mukainen rekisteri, joka toimii Enterprise Repositoryn ajoaikaisena rajapintana. Se tukee mm. palvelujen sidontaa, ajoajasta sijaintiläpinäkyvyyttä sekä pääsyä kontekstista riippuvaan palveluversioon.
- Oracle SOA Governance sisältää myös Web Service Manager ja Enterprise Manager -työkalut, jotka sisältyvät myös SOA Suiteen.

5.2.2 Oracle Data Integration

Oracle tarjoaa useita tuotteita datan integrointiin heterogeenisistä lähteistä liittyviin tarpeisiin, kuten datan siirtoon, muuntamiseen, profilointiin, replikointiin, datan laadun hallintaan sekä datapalvelujen toteuttamiseen. Oraclen keskeinen datan integrointia tukeva tuote on Data Integrator, joka tukee useita datan integrointiin liittyviä kyvykkyyksiä, kuten suurten datamäärien muuntamista ja validointia sekä datapalvelujen toteuttamista.

5.2.3 Oracle Enterprise Gateway

Enterprise Gateway on turvaratkaisu, joka toimii käytävänä ulkopuolisten palvelun kuluttajien ja yrityksen web-palvelujen välillä. Se toimii palomuurina ja välittää turvalliset palvelukutsut palveluväylälle. Se tukee mm. todentamista, valtuutusta ja identiteetin propagointia. Sen toiminta perustuu vahvasti politiikkoihin ja WS-Policy-spesifikaatioon ja se sisältää erinäisiä politiikan hallintaa tukevia apuvälineitä. Enterprise Gateway integroituu muun muassa SOA Suiten ja Oracle SOA Governancen tuotteiden kanssa.

5.2.4 Oracle Business Process Analysis Suite

Business Process Analysis (BPA) Suite on liiketoiminta-analyttikoille suunnattu ratkaisu, joka tukee liiketoimintaprosessien suunnittelua, mallintamista, simulointia, analysointia, optimointia, julkaisua, varastointia, jakamista sekä prosessimallien implementoinnin automatisointia. Se perustuu Oraclen partnerin, IDS Scheerin, ARIS-tuotteisiin. BPA Suitessa luodut prosessimallit voidaan jakaa SOA Suiten kanssa.

5.2.5 Oracle Application Integration Architecture

Application Integration Architecture (AIA) on kehys, joka on suunniteltu tukemaan palvelupohjaisten sovellusten ja prosessien kehittämistä ja integrointia sekä SOAn nopeaa omaksumista yrityksessä. Se perustuu Fusion Middleware -tuotteisiin, erityisesti SOA Suiteen ja BPM Suiteen. AIA sisältää seuraavat komponentit:

- **AIA Foundation Pack** on kokoelma erilaisia apuvälineitä. Näitä ovat mm. valmiit dataobjektimääritykset (kuten Tili, Myyntitilaus ja Asiakas), palvelumääritykset (kuten tilin luominen) sekä neljällä abstraktiotasolla kuvatut prosessimääritykset (kuten myyntiä tukevat prosessit). AIA Foundation Pack sisältää myös parhaita käytäntöjä, suunnittelukuvioita, ohjelmointimalleja sekä erinäisiä työkaluja SOAn toteuttamisen eri vaiheisiin.
- **Pre-Built Integrations** eli valmiit integraatiot tukevat Oraclen tuotteiden, muiden valmistajien tuotteiden ja räätälöityjen ohjelmistojen välisiä integraatioita. Valmiisiin integraatioihin sisältyy sovellusten välisiä *suoria integraatioita*, jotka ohjaavat sovellusten välisiä datavirtoja ja datan synkronointia, sekä *prosessi-integraatiopaketteja*, jotka käyttävät yhtä tai useampaa integraatiotyyliä, kuten datakeskeistä integraatiota, web-palvelu-integraatiota tai prosessikeskeistä integraatiota.

Luku 6

SOA-mallien ja -tuotteiden vertailu

Edellisissä luvuissa on tehty useita havaintoja tutkittavista SOA-malleista ja tarkasteltu Oraclen tuotteisiin perustuvaa SOA-ympäristöä. Tässä luvussa keskitytään ensin vertailemaan malleja keskenään neljän näkökulman avulla, jotka on valittu edellisten lukujen havaintojen perusteella. Tämän jälkeen malleja verrataan Oraclen SOA-ympäristöön. Vertailuilla pyritään tukemaan edelleen mallien ymmärtämistä ja arvioimaan mallien suhdetta Oraclen SOA-ympäristöön. Luvussa muodostetaan työn lopulliset tulokset.

6.1 SOA-mallien vertailu

Tässä tutkielmassa tarkasteltavat SOA-mallit kuvaavat useita samoja aiheita ilman ilmeisiä ristiriitoja. Mallien ja niiden keskinäisten suhteiden ymmärtämistä vaikeuttavat kuitenkin muun muassa mallien monimutkaisuus ja niiden näkökulmia ja muotoa koskevat eroavaisuudet. Tämän vuoksi malleja vertaillaan tässä muutaman keskeisen näkökulman avulla. Tämä vertailu toimii myös perustana mallien ja SOA-tuotteiden väliselle vertailulle. Malleista pyrittiin tunnistamaan edellisten lukujen havaintojen perusteella mallien ymmärtämisen kannalta tärkeitä seikkoja, kuten merkittäviä ominaispiirteitä, yhtäläisyyksiä ja eroavaisuuksia. Vertailuun valittiin näkökulmia, joiden avulla voidaan tarkastella ja selittää kyseisiä seikkoja sekä jäsenellä ja luokitella malleja. Näkökulmien valinnassa otettiin myös vaikutteita kohdassa 2.3.2 kuvatusta Fattahin vertailutavasta. Näkökulmien määrä pyrittiin pitämään melko vähäisenä keskeisten seikkojen korostamiseksi. Käytettävät näkökulmat ovat tarkastelualue, rakennekeskeisyys, abstraktiotaso ja dynaamiset piirteet. Mallien vertailun

yhteenvedoon (kohdassa 6.1.5) on otettu mukaan myös mallien keskeiset käyttötavat ja heikkoudet. Tarkastelualue ja abstraktiotaso vastaavat läheisesti Fattahin lähestymistavassa käytettyjä näkökulmia.

6.1.1 Tarkastelualue

Mallin tarkastelualue – sen kuvaamat aiheet – on keskeisin mallin ominaisuuksista. Fattahin esittämässä vertailussa SOA-mallit on järjestetty kattavuuden mukaiseen järjestykseen ja sijoitettu yksinkertaisiin luokkiin. Tässä tutkielmassa käsiteltävät mallit sijoittuvat luokkiin ”osittainen referenssiarkkitehtuuri” ja ”kattava tekninen referenssiarkkitehtuuri”. Tämä esitys on havainnollinen ja helposti ymmärrettävä, mutta se on myös epätarkka ja ilmeisesti virheellinen tai vanhentunut, sillä OASISin referenssiarkkitehtuuri on siinä esitetty tässä tutkielmassa käsiteltävistä malleista suppeimpana. Kyseinen malli edustaa kuitenkin yleisesti tarkastellen laajinta näkökulmaa, sillä siinä tarkastellaan palvelukeskeisyyden soveltamista yrityksessä yleisemmin keskittyen SOAn teknisten piirteiden lisäksi myös SOA-ekosysteemiin ja SOAn omistamiseen liittyviin prosesseihin. Tekniseltä kattavuudeltaan malli on kuitenkin The Open Groupin referenssiarkkitehtuuria suppeampi. Tällaisten painotuserojen vuoksi mallien kattavuusjärjestystä ei voida määrittää yksiselitteisesti. Referenssimallissa, ontologiassa ja The Open Groupin referenssiarkkitehtuurissa tarkastellaan SOAa pääosin teknisenä järjestelmänä, joka kuitenkin peilaa liiketoimintoja ja liiketoiminnan logiikkaa. Näistä teknisistä malleista The Open Groupin referenssiarkkitehtuurin tarkastelualue on selvästi laajin, sillä siinä tarkastellaan varsin kattavasti SOAn teknistä arkkitehtuuria – sekä sovelluselementtejä että infrastruktuuria. Sovelluselementeillä tarkoitetaan tässä elementtejä, jotka ovat loogisesti palvelupohjaisen sovelluksen osia. Tällaisia elementtejä ovat mm. palvelurajapinnat, palvelukomponentit, politiikat, prosessikuvaukset, tietorakenteiden kaavat, sovellusten data, käyttöliittymät, tapahtumat, liiketoimintasäännöt, adapterit, välittäjäkomponentit ja sovellukseen integroidut manuaaliset tehtävät. Infrastruktuuri-elementeillä tarkoitetaan elementtejä, jotka tukevat palvelukeskeisiä sovelluksia suunnittelu- tai ajoaikana. Näihin elementteihin luetaan tässä mm. sovelluspalvelimet, moottorit, säiliöt, varastot, rekisterit ja työkalut, kuten kehitysympäristöt ja hallinta- ja monitorointityökalut. The Open Groupin

referenssiarkkitehtuuri keskittyy vahvasti infrastruktuuri-elementteihin, mutta se kuvaa korkealla tasolla myös lukuisia sovelluselementtejä ja niiden suhteita. OASISin referenssimalli kuvaa sen sijaan pääasiassa vain keskeisimpiä sovelluselementtejä ja niihin liittyviä käsitteitä, sillä siinä viitataan infrastruktuuriin vain pinnallisesti. The Open Groupin ontologia keskittyy sovelluselementteihin eikä kuvaa lainkaan infrastruktuuria.

6.1.2 Rakennekeskeisyys

Kohdassa 2.1 esitellyn IEEE:n määritelmän mukaan ohjelmistointensiivisen järjestelmän arkkitehtuuri on sen ”perustavanlaatuinen *rakenne*, joka ilmenee sen komponenteissa sekä komponenttien suhteissa toisiinsa ja ympäristöön; sekä periaatteet, jotka ohjaavat sen suunnittelua ja kehittämistä”. Kaikki tässä tutkielmassa tarkasteltavat SOA-mallit kuvaavat SOAn loogisia rakenteellisia elementtejä, mutta mallien välillä ilmenee perustavanlaatuinen ero siinä, kuinka voimakkaasti ne keskittyvät rakenteellisiin seikkoihin. Malleissa kuvataan myös muita, käsitteellisempiä seikkoja, mikä vaikuttaa muun muassa mallien soveltamistapoihin ja siihen, kuinka hyvin malleja voidaan vertailla keskenään ja SOA-tuotteisiin.

OASISin referenssimalli ei keskity erityisesti rakenteellisiin seikkoihin, vaan se pyrkii kuvaamaan yleisesti SOAn olemuksen ja palvelukeskeisen ympäristön keskeisimpiä käsitteitä. Pääosa mallista kuvaa kuitenkin rakenteellisia elementtejä, mutta lähes puolet mallista kuvaa abstraktimpia, erityisesti palvelujen dynamiikkaa koskevia käsitteitä. Sen sijaan The Open Groupin ontologia – joka on tarkoitettu käytettäväksi muun muassa SOA-metamallien osana ja artefaktien metadatanä – keskittyy kokonaan rakenteellisten elementtien mallintamiseen. OASISin referenssiarkkitehtuuri pyrkii myös kuvaamaan SOAn arkkitehtuurillisia elementtejä, mutta sen lähestymistapa on siitä huolimatta enemmän käsitteellinen kuin rakennekeskeinen. Se keskittyy – jokseenkin heikosti jäsennellyllä tavalla – kuvaamaan SOAn toteuttamisen, SOA-ekosysteemin ja SOAn hallitsemiseen liittyvien prosessien käsitteellistä perustaa. Rakenteellisia seikkoja käsitellään pääosin mallin arkkitehtuurillisissa seurauksissa ja *SOA-ekosysteemin toteuttaminen* -näkyvässä. The Open Groupin referenssiarkkitehtuuri puolestaan keskittyy ontologian tavoin hyvin voimakkaasti rakenteellisten elementtien

kuvaamiseen. Se kuvaa myös SOAn teknisiä kyvykkyyksiä, jotka ovat kuitenkin läheisesti yhteydessä rakenteellisiin elementteihin.

6.1.3 Abstraktiotaso

Fattahin esittämässä vertailussa SOA-mallit on järjestetty myös abstraktiotason mukaiseen järjestykseen ja sijoitettu yksinkertaisiin luokkiin. Tässä tutkielmassa tarkasteltavat mallit sijoittuvat luokkiin ”käsitteellinen” ja ”geneerinen”. Mallien abstraktiotason määrittäminen ei kuitenkaan ole yksiselitteistä. Kaikki tässä tutkielmassa tarkasteltavat mallit ovat abstrakteja, tekniikoista riippumattomia malleja, jotka kuvaavat SOAn loogisia elementtejä. Kaikki mallit eivät kuitenkaan ole rakennekeskeisiä, ja toisaalta mallit keskittyvät eri yksityiskohtiin. Mallien abstraktiusjärjestys voidaan näin ollen määritellä vain karkeasti niiden yleisen lähestymistavan ja näkökulman perusteella. Mallin abstraktius vaikuttaa kuitenkin merkittävästi muun muassa sen käyttötapoihin.

OASISin referenssimalli pyrkii kuvaamaan SOAn olemusta ja palvelukeskeistä ympäristöä keskeisimpien käsitteiden avulla. Se on malleista lähimpänä palvelukeskeisen paradigman kuvausta ja voidaan näin ollen katsoa abstrakteimmaksi malliksi. The Open Groupin referenssiarkkitehtuuri sen sijaan kuvaa kattavasti ja yksityiskohtaisesti SOAn teknistä arkkitehtuuria. Se on malleista lähimpänä SOAn konkreettista teknistä toteutusta ja voidaan siten katsoa malleista konkreettisimmaksi. The Open Groupin ontologia kuvaa SOAn keskeisiä sovelluselementtejä. OASISin referenssiarkkitehtuurin muoto on monin paikoin ontologiaa käsitteellisempi, mutta se pyrkii kuvaamaan kattavammin SOAn toteuttamista ja arkkitehtuurillisia seurauksia ja voidaan siihen nähden katsoa ontologiaa konkreettisemmaksi. Tämä järjestys on yhdenmukainen Fattahin esittämän vertailun kanssa.

6.1.4 Dynaamiset piirteet

Tässä tutkielmassa tarkasteltavat mallit kuvaavat useita SOAn dynaamisia, ajoaikaisia piirteitä. Näitä seikkoja voidaan kuvata muun muassa abstrakteina käsitteinä tai rakenteellisten elementtien ominaisuuksina. Dynaamisilla piirteillä ei ole tärkeää osaa kaikissa malleissa, mutta joissain malleissa niillä on varsin suuri vaikutus mallin

muotoon ja ymmärrettävyyteen. Näissä tapauksissa dynaamisella näkökulmalla voidaan jäsentää mallia ja selittää tämän ominaisuuksia. OASISin referenssimallissa dynaamiset seikat ovat varsin keskeisessä osassa, sillä dynaamiset käsitteet muodostavat suuren osan mallin käsitteistä ja pääosan mallin abstrakteista käsitteistä. Dynaaminen näkökulma auttaa erottelemaan mallin abstrakteja käsitteitä rakennekeskeisistä käsitteistä ja jäsentelemään näin mallia. The Open Groupin ontologia sitä vastoin keskittyy täysin SOAn loogisten objektien ja niiden suhteiden mallintamiseen eikä kuvaa lainkaan dynaamisia seikkoja. Tämä ilmentää ontologian rakennekeskeistä ja suunnitteluaikeista luonnetta. OASISin referenssiarkkitehtuurissa kuvataan erinäisiä dynaamisia seikkoja – kuten palvelujen löytämistä ja vuorovaikutusta koskevia aiheita sekä turvallisuushkia – mutta niillä ei ole merkittävää vaikutusta mallin vaihtelevaan ja heikosti jäsenneltyyn muotoon. The Open Groupin referenssiarkkitehtuurissa dynaamisia seikkoja puolestaan kuvataan johdonmukaisesti ja muista seikoista jokseenkin erillään. Rakennuspalojen välisiä vuorovaikutussuhteita ja -kulkua kuvataan erikseen eri kerroksen kuvauksissa, minkä lisäksi SOAn ajoaikaisia elementtejä ja niiden suhteita kuvataan erityisesti Operationaaliset järjestelmät -kerroksessa. The Open Groupin referenssiarkkitehtuuria voidaan siten jäsentää ja selittää dynaamisen näkökulman avulla.

6.1.5 Vertailun yhteenveto

SOA-mallien vertailun yhteenveto on esitetty taulukossa 6.1. Yhteenvedossa on esitetty myös mallien keskeiset käyttötavat ja heikkoudet.

Taulukko 6.1. SOA-mallien vertailun yhteenveto.

	OASISin referenssimalli	The Open Groupin ontologia	OASISin referenssi-arkkitehtuuri	The Open Groupin referenssi-arkkitehtuuri
Tarkastelualue	SOAn olemus ja palvelukeskeinen ympäristö: keskeisimmät sovelluselementit ja niihin liittyvät käsitteet, infrastruktuuri pinnallisesti	SOAn keskeiset sovelluselementit	SOAn ja sen toteuttamisen käsitteellinen perusta: keskeiset sovelluselementit, infrastruktuuri, SOA-ekosysteemi, SOAn omistamiseen liittyvät prosessit	SOAn tekninen arkkitehtuuri kattavasti: pääasiassa infrastruktuuri, myös sovelluselementit
Rakenneskeisyys	käsitteellinen; ei pyri erityisesti rakenneskeiseen kuvaukseen	erittäin rakenneskeinen	pyrkii kuvaamaan rakenteellisia elementtejä, lähestymistapa kuitenkin enemmän käsitteellinen	erittäin rakenneskeinen; kuvaa myös SOAn teknisiä kyvykkyyskäsitteitä
Abstraktiotaso	abstraktein; lähimpänä palvelukeskeisen paradigman kuvausta	2. abstraktein; kuvaa valikoituja keskeisiä sovelluselementtejä	2. konkreettisin; kuvaa SOAn toteuttamista käsitteellisestä näkökulmasta	konkreettisin; lähimpänä SOAn konkreettista teknistä toteutusta
Dynaamiset piirteet	merkittävä osa käsitteistä kuvaa palvelujen dynamiikkaa	ei kuvaa dynaamisia piirteitä	kuvaa erinäisiä dynaamisia seikkoja, mutta nämä eivät vaikuta merkittävästi mallin muotoon	kuvaa johdonmukaisesti dynaamisia seikkoja, kuten rakennuspalojen välisiä vuorovaikutussuhteita ja -kulkuja; SOAn ajoaikaisia elementtejä kuvataan Operationaaliset järjestelmät -kerroksessa
Keskeiset käytötavat	SOAn yhteisen ymmärryksen ja semantiikan tukeminen, SOAn toteuttamisen tukeminen käsitteellisellä tasolla	SOAn yhteisen ymmärryksen ja semantiikan tukeminen, metamallina ja metadatanä toimiminen, mallilähtöisen implementoinnin tukeminen	SOAn toteuttamisen tukeminen; yritysarkkitehtuurin tukeminen; SOAn toteuttamisen käsitteellisen perustan, SOAn kokonaisuuden ja SOA-ekosysteemin ymmärtämisen tukeminen	SOAn ja yritysarkkitehtuurin toteuttamisen ja arvioinnin tukeminen vaatimuslähtöisesti, SOAn merkittävien kyvykkyyskokonaisuuksien ja teknisen arkkitehtuurin ymmärtämisen tukeminen, SOA-tuotteiden suunnittelun tukeminen

(jatkuu)

Taulukko 6.1. (jatkuu)

Heikkoudet	abstraktius ja epätarkka muoto vaikeuttavat ymmärtämistä	yksityiskohtainen ja formaalinen muoto ja eräät abstraktit luokat vaikeuttavat ymmärtämistä	käsitteellinen ja valikoiva lähestymistapa sekä osittain heikosti jäsennelty muoto	monimutkaisuus, kerroskuvauksen karkeus
-------------------	--	---	--	---

6.1.6 Johtopäätökset

Edellisissä kohdissa esitetty vertailu osoitti useita huomattavia eroavaisuuksia SOA-mallien kesken. Kullakin mallilla on merkittäviä ominaispiirteitä. Tämän vuoksi malleista voidaan tehdä vain melko karkeita yleisluonteisia johtopäätöksiä.

Tarkasteltavien SOA-mallien aihealue on laaja ja monimuotoinen, minkä vuoksi se on jokseenkin vaikeasti jäsennettävissä. Rakennekeskeisten ja käsitteellisten mallien aiheet ovat myös hyvin eriluonteisia. Vertailun perusteella esitetään seuraavat viisi osa-aluetta, joiden avulla mallien kokonaisaihealuetta voidaan kuvata ja jäsentää karkeasti:

1. SOAn sovelluselementit
2. SOAn infrastruktuuri
3. Organisaatio ja SOA-ekosysteemi
4. SOAa tukevat prosessit
5. Edellisiin liittyvät käsitteelliset aiheet

Ensimmäiset kaksi osa-aluetta edustavat kohdassa 6.1.1 kuvattuja rakenteellisia sovellus- ja infrastruktuuri-elementtejä. Myös kolmas ja neljäs osa-alue edustavat tässä konkreettisia aiheita – joita on tosin tarkasteltu melko käsitteellisestä näkökulmasta OASISin referenssiarkkitehtuurissa. Viides osa-alue edustaa käsitteellistä, selittävää näkökulmaa ja käsitteellisiä aiheita, jotka liittyvät muihin, konkreettisempiin aiheisiin. Tällaisia käsitteellisiä aiheita ovat mm. palvelujen dynamiikka, suorituskontekstin skaalautuvuus, toimijoiden välinen luottamus sekä SOA-ekosysteemin vaikutus testaamiseen. Tämä jako SOAn konkreettisten tai rakenteellisten ja käsitteellisten seikkojen välillä muistuttaa hieman jakoa järjestelmän rakenteen ja periaatteiden välillä, joka ilmenee kohdassa 2.1 esitetystä IEEE:n arkkitehtuurimäärittelyssä.

Tutkittavilla SOA-malleilla on myös niiden soveltamista koskevia ominaispiirteitä. Kuitenkin myös soveltamistapojen joukosta voidaan tunnistaa joitain huomattavimpia tapoja, joilla mallit tukevat SOAn ymmärtämistä ja toteuttamista. Mallien tarkastelun ja vertailun perusteella esitetään kolme merkittävintä tapaa, joilla mallit tukevat SOAn ymmärtämistä ja toteuttamista:

1. Mallit tarjoavat yhteisen termistön ja merkityksen SOAn keskeisille käsitteille.
2. Mallit kuvaavat SOAn arkkitehtuurillisia elementtejä ja niiden suhteita. Tämä tukee sekä SOAn rakenteen ymmärtämistä että SOAn toteuttamista.
3. Mallit selittävät SOAn ja sen toteuttamisen käsitteellistä perustaa selittämällä SOAn ymmärtämisen ja toteuttamisen kannalta tärkeitä aiheita ja ilmiöitä.

Mallien yleisenä heikkoutena voidaan pitää abstraktiutta, joka asettaa haasteita mallien ymmärtämiselle. Myös mallien perusteellisuus rajoittaa niiden käyttöä SOAn ymmärtämisen tukemisessa. SOAn perusominaisuuksien ymmärtämistä voitaisiin luultavasti tukea paremmin yksinkertaisella rakennekeskeisellä mallilla, erillisellä kuvauksella SOAn käsitteellisestä perustasta ja toteutustekniikkaesimerkeillä.

6.2 SOA-mallien vertaaminen Oraclen SOA-tuotteisiin

Tässä tutkielmassa tarkasteltavat SOA-mallit pyrkivät tukemaan SOAn ymmärtämistä ja toteuttamista monin tavoin. Mallit ovat kuitenkin abstrakteja, yleisiä ja tekniikoista riippumattomia, minkä vuoksi ei ole helposti arvioitavissa, kuinka hyvin mallit vastaavat todellisiin tuotteisiin perustuvaan SOA-ympäristöön ja millainen merkitys malleilla on tällaisen ympäristön ymmärtämisessä ja toteuttamisessa. Seuraavissa kohdissa malleja verrataan yksittäin Oraclen tuotteisiin perustuvaan SOA-ympäristöön. Vertailussa kiinnitetään huomiota siihen, kuinka kattavia mallit ovat, millaisia puutteita niissä on, kuinka helposti mallien ja yksittäisten tuotteiden ja tekniikoiden välinen yhteys on tunnistettavissa sekä tukevatko mallit Oraclen SOA-ympäristön ymmärtämistä ja toteuttamista.

6.2.1 OASISin referenssimalli

OASISin referenssimalli kuvaa suppeasti ja hyvin abstraktilla tasolla palvelukeskeisyyden keskeisimpiä käsitteitä, jotka liittyvät erityisesti palvelujen dynamiikkaan ja metatasoon. Referenssimallin korkean abstraktiotason vuoksi ei ole helposti havaittavissa, mitkä Oraclen tuotteet tai näiden tukemat tekniikat toteuttavat referenssimallin käsitteet, kuten saavutettavuuden tai suorituskontekstin. Referenssimallin käsitteet kuvaavat kuitenkin selvimmin sellaisia SOAn perustekniikoihin perustuvia elementtejä kuin WSDL-kielisiä palvelurajapintoja, SOAP-viestejä ja WS-Policy-spesifikaation mukaisia politiikkoja. Myös useat infrastruktuuri-elementit toteuttavat referenssimallin käsitteitä, mutta niiden yhteys referenssimalliin on epäsuora. Tällaisia infrastruktuuri-elementtejä ovat muun muassa näkyvyyttä tukevat varastot ja rekisterit, kuten Oracle SOA Governancen Enterprise Repository ja Service Registry, sekä suorituskontekstiin osallistuvat elementit, kuten WebLogic-sovelluspalvelin, SCA-suoritusympäristö ja OSB-palveluväylä.

Referenssimallin kuvaama palvelutyyppi perustuu standardin mukaiseen palvelurajapintaan, jonka avulla palvelua voidaan käyttää yhteensopivasti heterogeenisessä ympäristössä. Tämän mallin mukaisilla WSDL/SOAP-palveluilla on olennainen osa Oraclen SOA-ympäristössä esim. integrointimekanismina. Oraclen SOA-ympäristön ensisijaisen sovelluskehitysmallin, SCA:n, palvelutyyppi poikkeaa kuitenkin referenssimallista merkittävästi, sillä SCA:n komponenttikeskeinen palvelutyyppi perustuu tiettyyn komponenttimalliin ja yhteiseen suoritussympäristöön. Referenssimalli ei näin ollen kuvaa hyvin kaikkia Oraclen SOA-ympäristössä käytettäviä palvelutyyppisiä. Muilta osin referenssimallin käsitteet sopivat kuitenkin myös SCA-ympäristöön.

Referenssimallin erittäin abstraktin ja suppean luonteen ja toisaalta myös perusteellisen kuvauksen vuoksi sen käyttömahdollisuudet Oraclen SOA-ympäristön ymmärtämisen ja toteuttamisen tukemisessa ovat hyvin vähäiset. Malli keskittyy palvelukeskeisyyden perusominaisuuksien kuvaamiseen, minkä vuoksi se ei tarjoa käytännössä juuri lainkaan tukea tai ohjausta SOA-ympäristön toteuttamiseen tai palvelupohjaisten sovellusten kehittämiseen. Referenssimallin abstraktiuden, suppeuden ja perusteellisen kuvauksen

vuoksi se soveltuu sellaisenaan huonosti myös Oraclen SOA-ympäristön ja SOAn perustekniikoiden ymmärtämisen tukemiseen. Konkreettisemmän aineiston yhteydessä mallin käsitteitä kuitenkin voitaneen hyödyntää SOAn joidenkin perusominaisuuksien ja näitä toteuttavien tekniikoiden ja Oraclen SOA-tuotteiden merkityksen selittämisessä.

6.2.2 The Open Groupin ontologia

The Open Groupin ontologia kuvaa SOAn sovelluselementtejä hieman referenssimallia monipuolisemmin ja hyvin rakennekeskeisesti. Ontologia ei kuvaa lainkaan SOAn infrastruktuuria. Ontologian käsitteiden ja Oraclen SOA-ympäristön elementtien välisen yhteyden hahmottamista vaikeuttavat hieman ontologian tietyt abstraktit luokat, luokkien perintähierarkiat sekä ontologian formaalinen, yksityiskohtainen muoto. Oraclen SOA-ympäristöstä voidaan kuitenkin tunnistaa useita ontologian kuvaamia elementtejä, kuten WSDL-kieliset palvelurajapinnat, WS-Policy-spesifikaation mukaiset politiikat, Human Task -SCA-komponentit, BPEL-prosessi-SCA-komponentit sekä Complex Event Processing -ympäristön tapahtumat. Ontologiasta puuttuu kuitenkin joitain tärkeitä Oraclen SOA-ympäristön sovelluselementtejä, kuten liiketoimintasääntökomponentti, välittäjäkomponentti (Mediator) ja adapteri.

Referenssimallin tavoin ontologia kuvaa paremmin rajapintakeskeistä palvelutyyppejä kuin SCA:n komponenttikeskeisiä palveluja. Ontologia ei sisällä erikseen palvelukomponenttiluokkaa, ja toisaalta yleisluonteinen Palveluyhdistelmä-luokka ei vastaa hyvin SCA-komposiittia, sillä tämä noudattaa SCA:n komponenttimallia eikä lisää yhdistelmään logiikkaa, vaan tämä voidaan tehdä esim. BPEL-komponentin avulla.

Ontologiaa ei käytetä metamallina eikä metadatanä Oraclen SOA-ympäristössä, eikä Enterprise Repository tue ontologiaa hyödyntävää S-RAMP-spesifikaatiota. Ontologia kuvaa useita SCA-ympäristössä käytettäviä sovelluselementtejä muttei ohjaa palvelupohjaisten sovellusten muodostamista muulla tavoin. Ontologiaa voidaan hyödyntää – kenties yksinkertaistetussa muodossa – muun aineiston ohessa SOAn perusasioiden selittämisessä. Oraclen SOA-ympäristön ymmärtämistä tukisi kuitenkin

paremmin malli, joka kuvaa erityisesti Oraclen SOA-ympäristössä käytettäviä sovelluselementtejä.

6.2.3 OASISin referenssiarkkitehtuuri

OASISin referenssiarkkitehtuuri kuvaa SOAn ja sen toteuttamisen käsitteellistä perustaa. Mallissa tarkastellaan SOAa tiettyjen aihealueiden ja käsitteiden kautta, ja mallin arkkitehtuurilliset seuraukset on esitetty ja ryhmitelty näiden aiheiden ja käsitteiden mukaan. Mallissa ei näin ollen tarkastella SOAn teknistä kokonaisarkkitehtuuria tai tämän osakokonaisuuksia.

Referenssiarkkitehtuurin käsitteellisen luonteen vuoksi sen vertaaminen konkreettiseen arkkitehtuuriin on vaikeaa. Tämä koskee erityisesti *Osallistuminen SOA-ekosysteemissä* -näkömää, joka kuvaa SOA-ekosysteemiä ja siinä toimimista. Referenssiarkkitehtuuria voidaan kuitenkin verrata Oraclen tuotteisiin perustuvaan SOA-ympäristöön sen arkkitehtuurillisten seurausten avulla. Yksityiskohtainen vertailu on tämän työn laajuuden ulkopuolella, mutta yleisesti voidaan havaita, että mallin arkkitehtuurisille seurauksille voidaan pääosin tunnistaa niitä toteuttavia tai muistuttavia ominaisuuksia Oraclen SOA-tuotteista ja näiden tukemista tekniikoista. Keskeisiä tällaisia tekniikoita ja tuotteita ovat WSDL, SOAP, WS-Policy, BPEL, Enterprise Repository, Web Services Manager, Enterprise Manager, OSB-palveluväylä, BPEL Process Manager, Mediator sekä Enterprise Gateway. Referenssiarkkitehtuurin näennäisestä laaja-alaisuudesta huolimatta useat Oraclen SOA-tuotteet jäävät kuitenkin kokonaan tai lähes kokonaan referenssiarkkitehtuurin tarkastelualueen ulkopuolelle. Mallissa ei käsitellä lainkaan tai juuri lainkaan esimerkiksi sellaisia SOAn toteuttamiseen liittyviä aiheita kuin liiketoimintasääntöjä, ihmistyön sisällyttämistä sovelluksiin tai datan integrointia. Mallissa ei myös käsitellä SOAn omistamiseen liittyvää liiketoiminta-aktiviteetin monitorointia (BAM). Referenssiarkkitehtuuri kuvaa näin ollen monialaisesti mutta melko valikoivasti SOAn ja sen toteuttamisen käsitteellistä perustaa sekä tämän toteuttamiseen tarvittavia arkkitehtuurillisia mekanismeja.

Referenssiarkkitehtuuri kuvaa ja selittää useita aiheita, joita ei voida tarkastella hyvin vain tuotteiden ja tekniikoiden näkökulmasta, kuten SOA-ekosysteemiä ja SOAn

omistamiseen liittyviä aiheita, kuten hallintoa ja testausta. Referenssiarkkitehtuurin arkkitehtuurilliset seuraukset sen sijaan ovat monin paikoin perusluonteisia ja pääosin sisältyvät Oraclen SOA-ympäristöön, minkä vuoksi referenssiarkkitehtuuri ei tarjoa merkittävää ohjausta Oraclen SOA-ympäristön toteuttamiseen tai palvelupohjaisten sovellusten kehittämiseen Oraclen SOA-ympäristössä. Kohdassa 4.1.4 todettiin, että spesifikaatiota kehittävän teknisen komitean mukaan referenssiarkkitehtuuria voidaan käyttää ikään kuin tarkistuslistana asioista, jotka tulee ottaa huomioon SOA-ratkaisujen arkkitehtuurillisessa suunnittelussa ja konkreettisten arkkitehtuurien validoinnissa. Referenssiarkkitehtuuri soveltuu näin ollen parhaiten tällaiseksi monimuotoiseksi vertailukohdaksi, jota voidaan käyttää tarvittaessa SOAn toteuttamisen ja organisoinnin tukemiseen.

6.2.4 The Open Groupin referenssiarkkitehtuuri

The Open Groupin referenssiarkkitehtuuri kuvaa laaja-alaisesti SOAn teknistä arkkitehtuuria – erityisesti sen rakennuspaloja ja näiden toteuttamia kyvykkyyskäsitteitä. Rakennuspalat on kuvattu Oraclen SOA-tuotteisiin nähden hienojakoisesti, minkä vuoksi niiden vertaaminen tuotteisiin ei ole suoraviivaista. Yksityiskohtainen vertailu on tämän työn laajuuden ulkopuolella, mutta yleisesti voidaan havaita, että rakennuspalojen ja Oraclen SOA-tuotteiden ominaisuuksien välillä on laajasti samankaltaisuutta. Referenssiarkkitehtuurin jako kerroksiin ja rakennuspaloihin ei kuitenkaan pääosin sellaisenaan vastaa Oraclen SOA-tuoteryhmittelyä.

Useimmat referenssiarkkitehtuurin kerrokset kuvaavat monen SOA-tuotteen ominaisuuksia. *Operationaaliset järjestelmät -kerros* kuvaa erilaisia SOAa tukevia elementtejä, joihin sisältyvät Oraclen SOA-tuotteista pääasiassa vain WebLogic-sovelluspalvelin ja Oracle Database -tietokanta. *Palvelukomponenttikerroksen* kuvaamiin elementteihin lukeutuvat mm. SCA-komponentit, EJB-komponentit, suunnitteluajankaisia kyvykkyyskäsitteitä tukeva JDeveloper sekä ajoajankaisia kyvykkyyskäsitteitä tukeva WebLogic-sovelluspalvelin ja siihen sisältyvä SCA-suoritusympäristö. *Palvelukerroksen* kuvaamia elementtejä ovat WSDL/SOAP-palvelut, SCA-komponenttien palvelut, REST-palvelut, WebLogic ja SCA-suoritusympäristö. Referenssiarkkitehtuurin kerrosjako kuvaa kuitenkin luontevammin WSDL/SOAP-

tyyppisiä palveluja, joissa rajapinta on selvästi erotettu palvelukomponentista, kuin SCA-palveluja, joissa palvelun ja komponentin suhde on kiinteämpi. *Liiketoimintaprosessit-kerroksen* kuvaamiin elementteihin sisältyvät BPEL-prosessit, BPEL Process Manager, Human Workflow ja Enterprise Repository. *Kuluttajakerroksen* kyvykkyyksiä toteuttaviin Oraclen tuotteisiin lukeutuu pääasiassa Fusion Middleware -tuoteperheen ADF-sovelluskehityskehys (Application Development Framework), joka soveltuu eri kanavien kautta kulutettavien yrityssovellusten kehittämiseen. *Integraatiokerroksen* kuvaamia elementtejä ovat OSB-palveluväylä, Mediator, Adapters, CEP ja BPEL Process Manager. *Palvelutasokerroksen* kuvaamiin elementteihin lukeutuu samoin useita tuotteita: Enterprise Manager, BAM, CEP, Web Services Manager sekä OSB-palveluväylä. *Informaatiokerroksen* kuvaamia elementtejä puolestaan ovat Data Integrator, Metadata Repository, Oracle Database sekä SOA Suiten EDL-kieliset (Event Definition Language) tapahtumat. Lisäksi kerros tukee master datan hallintaa (engl. master data management, MDM) sekä sisällön hallintaa (engl. content management), joihin Oracle tarjoaa erillisiä tuotteita. *Hallintokerroksen* kuvaamiin elementteihin lukeutuu WS-Policy-politiikkojen lisäksi myös useita tuotteita, kuten Enterprise Repository, Service Registry, Business Rules, Web Services Manager sekä BAM. Referenssiarkkitehtuuri kuvaa näin ollen jossain määrin suurinta osaa Oraclen SOA-tuotteista. Tässä tutkielmassa tarkastelluista tuotteista referenssiarkkitehtuurin ulkopuolelle jäävät B2B, User Messaging Service, Business Process Analysis Suite, Enterprise Gateway sekä Application Integration Architecture. Referenssiarkkitehtuuri kuvaa myös muutamaa sellaista tuotetta, jotka eivät sisälly Oraclen varsinaiseen SOA-tuotevalikoimaan. Tällaisia tuotteita ovat ADF-sovelluskehityskehys, master datan hallintatuotteet sekä sisällönhallintatuotteet.

The Open Groupin referenssiarkkitehtuuri kuvaa näin ollen melko kattavasti Oraclen SOA-ympäristön keskeisiä piirteitä. Kuluttajarajapintoja tukevia elementtejä lukuunottamatta referenssiarkkitehtuuri ei kuvaa merkittävästi Oraclen SOA-tuotteiden ulkopuolisia tuotteita. Referenssiarkkitehtuurin kerros- ja rakennuspalajako ei kuitenkaan pääosin vastaa Oraclen SOA-tuoteryhmittelyä, minkä vuoksi rakennuspalaja toteuttavien tuotteiden tunnistaminen on osittain vaikeaa.

Referenssiarkkitehtuurin korkean tason kerroskuvausta voidaan käyttää SOAn ja palvelupohjaisten sovellusten loogisen rakenteen hahmottamiseen, sillä tämä rakenne ei ole helposti havaittavissa Oraclen laajasta SOA-tuotevalikoimasta. Kerroskuvaus ei kuitenkaan sellaisenaan juuri ilmennä Oraclen SOA-tuotteiden merkitystä kokonaisarkkitehtuurissa, minkä vuoksi tuotteiden merkitys ja niiden ja referenssiarkkitehtuurin välinen suhde on tarvittaessa selitettävä erikseen. Referenssiarkkitehtuuri pyrkii tukemaan SOAn toteuttamisen loogista, tuotteista riippumatonta suunnittelua vaatimuslähtöisesti. Oraclen SOA-tuotteet toteuttavat kuitenkin melko kattavasti referenssiarkkitehtuurin rakennuspalat, joten referenssiarkkitehtuurilla ei ole merkittävää arvoa Oraclen SOA-ympäristön toteuttamista tai palvelupohjaisten sovellusten kehittämistä Oraclen SOA-ympäristössä ohjaavana tai tukevana mallina.

6.2.5 Johtopäätökset

Edellisissä kohdissa esitetyn vertailun perusteella voidaan tehdä useita havaintoja. Ensiksi, rajapintakeskeisellä palvelutyypillä on merkittävä osa tarkasteltavissa SOA-malleissa. Tämän vuoksi ne sopivat hyvin yhteen WSDL/SOAP-palvelutyypin kanssa, mutta komponenttikeskeisen SCA-palvelutyypin kanssa ne sopivat melko huonosti yhteen. WSDL:än ja SOAPin lisäksi muita mallien kannalta merkittäviä SOAn perustekniikoita ovat WS-Policy ja BPEL.

Toiseksi, mallien teknisessä kattavuudessa ja abstraktiotasossa Oraclen SOA-ympäristöön nähden on varsin suuria eroja mallien kesken. Referenssimalli, joka voidaan katsoa malleista teknisesti suppeimmaksi ja abstrakteimmaksi, kuvaa käsitteitä, jotka liittyvät pääasiassa edellä mainittuihin SOAn perustekniikoihin: WSDL:ään, SOAPiin ja WS-Policyyn. The Open Groupin referenssiarkkitehtuuri, joka on malleista teknisesti kattavin ja konkreettisin, kuvaa sen sijaan melko kattavasti Oraclen SOA-tuotteisiin perustuvan arkkitehtuurin keskeisiä ominaisuuksia.

Kolmanneksi, mallien ja Oraclen SOA-tuotteiden välinen yhteys on osittain vaikeasti tunnistettavissa. Jotkut Oraclen SOA-tuotteet toteuttavat tai tukevat malleissa kuvattuja

elementtejä selvästi, mutta Oraclen SOA-tuotteita vastaavia kokonaisuuksia ei pääosin ole tunnistettavissa sellaisinaan malleista. Tämä johtuu muun muassa mallien loogisesta, ei-tuotekeskeisestä näkökulmasta ja toisaalta mallien osittaisuudesta.

Neljänneksi, mallien käyttömahdollisuudet Oraclen SOA-ympäristön ymmärtämisen ja toteuttamisen tukemisessa ovat melko vähäiset. Tarkasteltavat mallit on tarkoitettu yleisiksi, abstrakteiksi, tekniikoista riippumattomiksi malleiksi, minkä vuoksi ne soveltuvat pääasiassa SOAn yleisluonteiseen selittämiseen ja SOAn toteuttamisen toimittajariippumattomaan suunnitteluun. Ne eivät kuvaa hyvin Oraclen SOA-ympäristön sovelluselementtejä eivätkä tue sellaisinaan hyvin Oraclen SOA-tuotteiden ymmärtämistä. Mallien vähäisenä käyttömahdollisuutena voidaan pitää sitä, että jotkut mallien käsitteelliset tai loogiset kuvaukset voivat soveltua konkreettisempien kuvausten täydentämiseen, jolloin ne voivat tukea joidenkin tuotteiden merkityksen tai SOAn loogisen rakenteen ymmärtämistä. Mallien käyttömahdollisuudet Oraclen SOA-ympäristön toteuttamisen tukemisessa ovat myös vähäiset, sillä Oraclen SOA-tuotteet tarjoavat varsin kattavan kehyksen SOAn toteuttamiseen ja palvelupohjaisten sovellusten kehittämiseen. Referenssiarkkitehtuurien vähäisenä käyttömahdollisuutena tässä tarkoituksessa voidaan pitää sitä, että niitä voidaan käyttää tarvittaessa ikään kuin tarkistuslistana asioista, jotka tulee ottaa huomioon SOAn toteuttamisessa.

Luku 7

Yhteenveto

Tässä tutkielmassa tarkasteltiin neljää SOA-mallia. Tutkielman ensimmäisenä tavoitteena oli selvittää, millaisia tarkasteltavat SOA-mallit ovat. Erityisesti pyrittiin selvittämään, mitä aiheita mallit kuvaavat, miten ne tukevat SOAn ymmärtämistä ja toteuttamista, millaisia eroavaisuuksia ja yhtäläisyyksiä malleilla on sekä millaista kokonaisaihealuetta mallit kuvaavat yhdessä. Toisena tavoitteena oli selvittää, millainen suhde malleilla on todellisiin tuotteisiin perustuvaan SOA-ympäristöön. Tässä pyrittiin erityisesti selvittämään, kuinka hyvin mallit vastaavat tällaista SOA-ympäristöä sekä tukevatko mallit tällaisen SOA-ympäristön ymmärtämistä ja toteuttamista. Tutkittaviksi malleiksi valittiin neljä standardointijärjestön esittämää abstraktia, tekniikoista riippumatonta SOA-mallia: OASISin referenssimalli ja referenssiarkkitehtuuri sekä The Open Groupin ontologia ja referenssiarkkitehtuuri. Vertailukohtana käytettäväksi SOA-tuotteiksi valittiin Oraclen SOA-tuotevalikoima.

Ennen mallien tutkimista luotiin katsaus eräisiin SOAn ja SOA-mallien ymmärtämisen kannalta tärkeisiin aiheisiin. Arkkitehtuurikuvaus-käsite esiteltiin lyhyesti, minkä jälkeen tarkasteltiin SOAa yleisesti ja luotiin alustava katsaus erilaisiin mallityyppeihin ja niiden merkitykseen.

Malleja tutkittiin aluksi yksittäin. Tarkastelussa kiinnitettiin huomiota erityisesti mallien kuvausalueeseen, abstraktiotasoon, näkökulmaan, kuvaustapaan ja soveltamistapoihin. Mallien välisiä yhtäläisyyksiä ja eroavaisuuksia tarkasteltiin myös alustavasti. Yksittäisten mallien tutkimisen jälkeen esiteltiin Oraclen laaja SOA-tuotevalikoima ja

Oraclen SOA-ympäristössä käytettävä SCA-sovelluskehitysmalli. Tämän jälkeen malleja vertailtiin keskenään neljän keskeisen näkökulman avulla, jotka valittiin edellisten lukujen havaintojen perusteella. Nämä näkökulmat olivat tarkastelualue, rakennekeskeisyys, abstraktiotaso ja dynaamiset piirteet. Lopuksi malleja verrattiin Oraclen SOA-ympäristöön. Vertailussa kiinnitettiin huomiota siihen, kuinka kattavia mallit ovat, millaisia puutteita niissä on, kuinka helposti mallien ja yksittäisten tuotteiden ja tekniikoiden välinen yhteys on tunnistettavissa sekä tukevatko mallit Oraclen SOA-ympäristön ymmärtämistä ja toteuttamista.

Tarkasteltavissa malleissa havaittiin merkittäviä eroavaisuuksia erityisesti niiden kuvausalueeseen ja yleiseen näkökulmaan nähden. Malleista tunnistettiin kaksi keskeistä yleistä näkökulmaa: rakennekeskeinen näkökulma ja käsitteellinen, selittävä näkökulma. OASISin referenssimalli havaittiin hyvin suppeaksi malliksi, jonka näkökulma on erittäin käsitteellinen. Sen keskeinen käyttötapa on SOAn peruskäsitteiden yhteisen ymmärryksen ja semantiikan tukeminen, mutta referenssimallin abstraktius ja epätarkka muoto vaikeuttavat tätä tavoitetta. The Open Groupin ontologia havaittiin hyvin rakennekeskeiseksi malliksi, joka kuvaa SOAn merkittäviä sovelluselementtejä. Sen huomattavimmat käyttötavat ovat SOAn peruskäsitteiden yhteisen ymmärryksen ja semantiikan tukeminen sekä metamallina ja metadatanä toimiminen. Mallin yksityiskohtainen, formaalinen, luokkiin perustuva muoto vaikeuttaa kuitenkin mallin käyttämistä SOAn ymmärtämisen tukemiseen. OASISin referenssiarkkitehtuuri havaittiin laaja-alaiseksi kehykseksi, joka kuvaa SOA-ekosysteemiä ja useita SOAn toteuttamiseen ja omistamiseen liittyviä aiheita käsitteellisestä näkökulmasta. Sitä voidaan käyttää SOAn toteuttamista tukevana, tarkistuslistan kaltaisena viitekehyksenä ja SOAn käsitteellistä perustaa selittävänä mallina. Näitä tavoitteita vaikeuttavat kuitenkin mallin käsitteellinen ja valikoiva lähestymistapa sekä osittain heikosti jäsenneilty muoto. The Open Groupin referenssiarkkitehtuuri havaittiin hyvin rakennekeskeiseksi ja laaja-alaiseksi tekniseksi malliksi, joka kuvaa korkealla tasolla SOAn loogista kerrosrakennetta ja matalalla tasolla SOAn arkkitehtuurillisia rakennuspaloja ja erityisesti infrastruktuuri-elementtejä. Sen keskeiset käyttötavat ovat SOAn ja yritysarkkitehtuurin toteuttamisen tukeminen

vaatimuslähtöisesti sekä SOAn loogisen kerrosrakenteen ja teknisen arkkitehtuurin ymmärtämisen tukeminen.

Tutkittavien SOA-mallien yhdessä kuvaama aihealue havaittiin laajaksi, monimuotoiseksi ja jokseenkin vaikeasti jäseneltäväksi. Mallien vertailun perusteella esitettiin seuraavat viisi osa-aluetta, joiden avulla aihealuetta voidaan kuvata ja jäsentää karkeasti:

1. SOAn sovelluselementit
2. SOAn infrastruktuuri
3. Organisaatio ja SOA-ekosysteemi
4. SOAa tukevat prosessit
5. Edellisiin liittyvät käsitteelliset aiheet, jotka edustavat käsitteellistä, selittävää näkökulmaa

Mallien keskinäisen vertailun perusteella tunnistettiin myös seuraavat kolme merkittävintä tapaa, joilla tarkasteltavat mallit tukevat SOAn ymmärtämistä ja toteuttamista:

1. Mallit tarjoavat yhteisen termistön ja merkityksen SOAn keskeisille käsitteille.
2. Mallit tukevat SOAn loogisen rakenteen ymmärtämistä ja SOAn toteuttamista kuvaamalla SOAn arkkitehtuurillisia elementtejä ja niiden suhteita.
3. Mallit selittävät SOAn ja sen toteuttamisen käsitteellistä perustaa selittämällä SOAn ymmärtämisen ja toteuttamisen kannalta tärkeitä aiheita ja ilmiöitä.

Mallien yhteiseksi heikkoudeksi havaittiin se, että niiden abstraktius ja perusteellisuus vaikeuttavat mallien ymmärtämistä. SOAn perusominaisuuksien ymmärtämistä voitaisiin luultavasti tukea paremmin yksinkertaisella rakennekeskeisellä mallilla, erillisellä kuvauksella SOAn käsitteellisestä perustasta ja toteutustekniikkaesimerkeillä.

Mallien ja Oraclen SOA-ympäristön vertailun perusteella tehtiin myös useita havaintoja. Ensiksi, mallit sopivat hyvin yhteen rajapintakeskeisen WSDL/SOAP-palvelutyypin kanssa, mutta komponenttikeskeisen SCA-palvelutyypin kanssa ne sopivat melko huonosti yhteen. Toiseksi, mallien teknisessä kattavuudessa ja abstraktiotasossa on hyvin suuria eroja. Teknisesti varsin suppea ja abstrakti OASISin referenssimalli kuvaa käsitteitä, jotka liittyvät pääasiassa SOAn perustekniikoihin:

WSDL:ään, SOAPiin ja WS-Policyyn. Sen sijaan teknisesti laaja ja verrattain hyvin konkreettinen The Open Groupin referenssiarkkitehtuuri kuvaa melko kattavasti Oraclen SOA-ympäristön keskeisiä ominaisuuksia. Kolmanneksi, mallien ja Oraclen SOA-tuotteiden välinen yhteys on osittain vaikeasti tunnistettavissa. Jotkut Oraclen SOA-tuotteet toteuttavat tai tukevat malleissa kuvattuja elementtejä selvästi, mutta Oraclen SOA-tuotteita vastaavia kokonaisuuksia ei pääosin ole tunnistettavissa sellaisinaan malleista. Neljänneksi, mallien käyttömahdollisuudet Oraclen SOA-ympäristön ymmärtämisen ja toteuttamisen tukemisessa ovat melko vähäiset, sillä mallit eivät kuvaa hyvin Oraclen SOA-ympäristön sovelluselementtejä eivätkä tue sellaisinaan hyvin Oraclen SOA-tuotteiden ymmärtämistä, ja toisaalta Oraclen SOA-tuotteet tarjoavat varsin kattavan kehyksen SOAn toteuttamiseen ja palvelupohjaisten sovellusten kehittämiseen.

Tässä tutkielmassa tarkastellut mallit ovat lukuisten henkilöiden ja jopa useiden vuosien kehitystyön tuloksia. Malleissa havaittiin kuitenkin heikkouksia, ja niiden käyttömahdollisuudet Oraclen SOA-ympäristön ymmärtämisen ja toteuttamisen tukemisessa havaittiin melko vähäisiksi. Tämän vuoksi voidaan pohtia mallien käytännöllistä arvoa ja sitä, ovatko mallit jollain tavoin epäonnistuneet tavoitteissaan. Mallien merkityksen ymmärtämiseksi pohditaan ensin hieman niiden kehittämisen tarkoitusta. SOAn yleistymisen ja omaksumisen merkittävänä haasteena – joka on vaikuttanut erityisesti SOAn yleistymisen alkuaikana – on ollut epäselvyys siitä, mitä SOA merkitsee ja miten se tulee toteuttaa. Tarkastellut mallit pyrkivät tarjoamaan yleisiä ja standardoituja, useiden tahojen yksimielisyyteen perustuvia ratkaisuja näihin kysymyksiin ja edistämään siten SOAn omaksumista SOA-yhteisössä, erityisesti SOAa omaksuvissa yrityksissä. Jos mallien onnistumista arvioidaan siihen nähden, että ne ovat laajoille kohderyhmille suunnattuja ja ne pyrkivät tarjoamaan käytännössä hyödyllistä tukea SOAn omaksumiseen, voitaneen katsoa, etteivät mallit ole kaikilta osin onnistuneet hyvin tavoitteissaan, sillä mallit soveltuvat – jokseenkin vaikeasti lähestyttävän muotonsa vuoksi – sellaisinaan pääasiassa henkilöille, joiden tulee ymmärtää SOAa syvällisesti, kuten SOA-arkkitehdeille, -konsulteille, -kouluttajille ja muille SOA-asiantuntijoille. Mallit ovat kuitenkin merkittävien standardointijärjestöjen spesifikaatioina saaneet huomiota, joten voidaan olettaa, että ne ovat osaltaan hieman

edistäneet SOAn merkityksen ja toteuttamisen ymmärtämistä sekä yhteistä termistöä SOA-yhteisössä. Myös S3-malli, johon The Open Groupin referenssiarkkitehtuuri perustuu, vaikuttaa saaneen alalla melko runsaasti huomiota. IBM:n työryhmän kehittämä S3 perustuu useisiin varhaisiin SOA-projekteihin [47], ja IBM osallistui merkittävästi myös The Open Groupin referenssiarkkitehtuurin kehittämiseen, joten referenssiarkkitehtuurilla on oletettavasti melko vahva yhteys IBM:n tuotteiden ominaisuuksiin. Muilla tarkastelluilla malleilla sen sijaan ei ole havaittavaa yhteyttä yksittäisen yrityksen tuotteisiin. Arvioitaessa referenssiarkkitehtuurien arvoa SOAn toteuttamisessa tulee ottaa huomioon erilaiset kohdeympäristöt. Oraclen SOA-tuotevalikoima, joka edustaa yhtä laajimmista SOA-alustoista, ohjaa SOAn teknistä toteuttamista varsin vahvasti. Mikäli SOAn toteuttamista suunnitellaan vähäisemmällä resursseilla tai matalan kypsyydystason SOAa pyritään kehittämään korkeammalle tasolle, on referenssiarkkitehtuureilla ja erityisesti The Open Groupin referenssiarkkitehtuurilla oletettavasti jonkin verran enemmän tai jopa huomattavasti enemmän arvoa SOAn suunnittelemisessa. Tarkastellut mallit eivät kuitenkaan korkean abstraktiotasonsa vuoksi tue merkittävästi esimerkiksi palvelupohjaisten sovellusten integroitavuutta tai SOA-työkalujen yhteensopivuutta, joskin The Open Groupin ontologialla olisi metadatan ja metamallina yleistyessään tähän potentiaalia. Tässä tutkielmassa arvioitiin referenssiarkkitehtuurien merkitystä pääasiassa teknisestä näkökulmasta vertaamalla niitä Oraclen SOA-ympäristöön. OASISin referenssiarkkitehtuuri pyrkii tukemaan myös SOAn omistamiseen liittyviä prosesseja, mutta mallin hyötyä ei arvioitu vertailussa tästä näkökulmasta. Mallia voidaan kuitenkin käyttää tarkistuslistana myös tältä osin. Malleilla havaittiin olevan selvää arvoa muiden spesifikaatioiden kehittämisessä, sillä OASISin referenssiarkkitehtuuria lukuun ottamatta tarkasteltuja malleja on hyödynnetty muissa spesifikaatioissa. Malleilla voidaan näin ollen arvioida olevan jonkin verran arvoa SOA-yhteisölle, erityisesti tietyille ryhmille ja tietyissä tapauksissa, mutta ne vaikuttavat onnistuneen jossain määrin puutteellisesti käytännössä hyödyllisen tuen tarjoamisessa laajoille kohderyhmille.

Tämän tutkielman tulosten ensisijainen merkitys on tarkasteltujen SOA-mallien ominaisuuksien ja merkityksen ymmärtämisen tukeminen. Tuloksia ja vertailussa käytettyjä näkökulmia on kuitenkin mahdollista käyttää myös muiden SOA-mallien

tarkasteluun ja arviointiin. SOAn teknisen toteuttamisen kannalta merkittävänä tuloksena voidaan pitää sitä, että mallien käyttömahdollisuudet Oraclen SOA-ympäristön ymmärtämisen ja toteuttamisen tukemisessa havaittiin melko vähäisiksi. Tämän voidaan olettaa pätevän jossain määrin myös muihin SOA-alustoihin, jotka ohjaavat vahvasti SOAn toteuttamista ja palvelupohjaisten sovellusten kehittämistä.

Tutkielman rajoituksena voidaan pitää sitä, että siinä tutkittujen mallien määrä on melko vähäinen – joskin tutkitut mallit edustavat standardointijärjestöjen spesifikaatioina useiden osapuolten yksimielisyyttä. Vähäisen mallimäärän vuoksi tulosten perusteella ei voida tehdä kovin yleispäteviä johtopäätöksiä SOA-malleista ja niiden käyttötavoista. Muut SOA-mallit voivat kuvata SOAa tavoilla, jotka poikkeavat merkittävästi tämän tutkielman havainnoista. Malleja on lisäksi verrattu vain Oraclen tuotevalikoimaan perustuvaan SOA-ympäristöön. Vertailu myös esimerkiksi yksinkertaisempaan tai ns. matalamman kypsyystason SOA-ympäristöön olisi tuottanut oletettavasti monipuolisempia tuloksia.

SOAn ongelma-alue on laaja, minkä vuoksi SOAn omaksumisen tukemiseksi on esitetty arkkitehtuurillisten mallien – joita tässä tutkielmassa tarkastellut mallit edustavat – lisäksi useita muita malleja, kuten hallintomalleja ja kypsyysmalleja. Tässä tutkielmassa tarkasteltiin arkkitehtuurillisten mallien merkitystä melko teknisestä näkökulmasta ja havaittiin, että tarkasteltavien mallien käyttömahdollisuudet Oraclen SOA-ympäristön ymmärtämisen ja toteuttamisen tukemisessa ovat melko vähäiset. SOA-malleja koskevana jatkotutkimuksena voitaisiin selvittää SOA-mallien merkitystä muissa käyttötarkoituksissa tai muiden SOAan liittyvien mallien merkitystä SOAn omaksumisen tukemisessa. SOAn hyötyjen saavuttamista ja SOAn käyttöönottoa vaikeuttavia merkittäviä haasteita ovat muun muassa hallinnon järjestäminen ja yhteisen ymmärryksen saavuttaminen SOA-paradigmasta [3], minkä vuoksi jatkotutkimuksena voitaisiin selvittää mallien merkitystä esimerkiksi näiden haasteiden ratkaisemisessa.

Viitteet

- [1] Erl, T. (2005). *Service-Oriented Architecture: Concepts, Technology, and Design*. Prentice Hall, Upper Saddle River, New Jersey, USA.
- [2] The Open Group. (2007). *Service-Oriented Architecture (SOA)*. Online white paper,
<https://www2.opengroup.org/ogsys/jsp/publications/PublicationDetails.jsp?catalogno=w074>. Viitattu 2.4.2012.
- [3] Becker, A., Buxmann, P. & Widjaja, T. (2009). Value Potential and Challenges of Service-Oriented Architectures – A User and Vendor Perspective. *Proceedings of the 17th European Conference on Information Systems (ECIS 2009)*, Verona, Italia, kesäkuu 2009, 2085–2096.
- [4] IEEE. (2007). ISO/IEC 42010:2007(E) IEEE Std 1471-2000. *Systems and software engineering – Recommended practice for architectural description of software-intensive systems*.
- [5] Smolander, K. (2002). Four Metaphors of Architecture in Software Organizations: Finding out The Meaning of Architecture in Practice. *Proceedings of the 2002 International Symposium on Empirical Software Engineering (ISESE'02)*, lokakuu 2002, 211–221.

- [6] OASIS. (2011). Reference Architecture Foundation for Service Oriented Architecture Version 1.0 Committee Specification Draft 03 / Public Review Draft 02. <http://docs.oasis-open.org/soa-rm/soa-ra/v1.0/csd03/soa-ra-v1.0-csd03.pdf>. Viitattu 2.4.2012.
- [7] The Open Group. (2011). *TOGAF Version 9.1*. Online-versio: <http://pubs.opengroup.org/architecture/togaf9-doc/arch/>. Viitattu 2.4.2012.
- [8] W3C. (2002). *Web Services Architecture*. <http://www.w3.org/TR/2002/WD-ws-arch-20021114/>. Viitattu 2.4.2012.
- [9] Bieber, G. & Carpenter, J. (2001). *Introduction to Service-Oriented Programming (Rev 2.1)*. Online white paper, www.openwings.org/download/specs/serviceorientedintroduction.pdf.
- [10] W3C. (2004). *Web Services Glossary*. <http://www.w3.org/TR/ws-gloss/>. Viitattu 2.4.2012.
- [11] Michlmayr A., Rosenberg F., Platzer C., Treiber M. & Dustdar S. (2007). Towards Recovering the Broken SOA Triangle – A Software Engineering Perspective. *Proceedings of the 2nd International Workshop on Service-oriented Software Engineering (IW-SOSWE '07)*, Dubrovnik, Kroatia, syyskuu 2007, 22–28.
- [12] Papazoglou M. P. (2003). Service-Oriented Computing – Concepts, Characteristics and Directions. *Proceedings of the Fourth International Conference on Web Information Systems Engineering (WISE'03)*, joulukuu 2003, 3–12.
- [13] Tang, L., Dong, J., Zhao, Y. & Tsai W.-T. (2010). A Classification of Enterprise Service-Oriented Architecture. *Proceedings of the fifth IEEE International Symposium on Service Oriented System Engineering (SOSE 2010)*, Nanjing, Kiina, kesäkuu 2010, 74–81.

- [14] Berkem, B. (2008). From The Business Motivation Model (BMM) To Service Oriented Architecture (SOA). *Journal of Object Technology*, vol. 7 no 8, marraskuu–joulukuu 2008, 57–70.
- [15] Lewis G. A. & Smith D. B. (2008). Service-Oriented Architecture and its Implications for Software Maintenance and Evolution. *Frontiers of Software Maintenance (FoSM 2008)*, Peking, Kiina, syyskuu–lokakuu 2008, 1–10.
- [16] Nayak N., Nigam A., Sanz J., Marston D. & Flaxer D. (2006). Concepts for Service-Oriented Business Thinking. *IEEE International Conference on Services Computing (SCC'06)*, Chicago, Illinois, USA, syyskuu 2006, 357–364.
- [17] Lo, A. & Yu, E. (2007). From Business Models to Service-Oriented Design: A Reference Catalog Approach. *Proceedings of the 26th international conference on Conceptual modeling (ER'07)*, 87-101.
- [18] OASIS. (2006). *Reference Model for Service Oriented Architecture 1.0*. <http://docs.oasis-open.org/soa-rm/v1.0/>. Viitattu 2.4.2012.
- [19] Pautasso, C., Zimmermann, O. & Leymann, F. (2008). RESTful Web Services vs. “Big” Web Services: Making the Right Architectural Decision. *Proceedings of the 17th international conference on World Wide Web (WWW 2008)*, Peking, Kiina, huhtikuu 2008, 805–814.
- [20] Abrams, C. & Schulte, R. W. (2008). *Service-Oriented Architecture Overview and Guide to SOA Research*. Gartner.
- [21] Offermann, P., Hoffmann, M. & Bub, U. (2009). Benefits of SOA – Evaluation of an Implemented Scenario against Alternative Architectures. *Proceedings of the 13th Enterprise Distributed Object Computing Conference Workshops (EDOCW 2009)*, Auckland, Uusi-Seelanti, syyskuu 2009, 352–359.

- [22] Arsanjani A., Booch G., Boubez T., Brown P. C., Chappell D., deVadoss J., Erl T., Josuttis N., Krafzig D., Little M., Loesgen B., Manes A. T., McKendrick J., Ross-Talbot S., Tilkov S., Utschig-Utschig C. & Wilhelmsen H. (2009). *SOA Manifesto*. <http://www.soa-manifesto.org/>. Viitattu 2.4.2012.
- [23] The Open Group. (2011). *SOA Reference Architecture*. <https://www2.opengroup.org/ogsys/jsp/publications/PublicationDetails.jsp?catalogno=c119>. Viitattu 2.4.2012.
- [24] Kokko, T., Antikainen, J. & Systä, T. (2009). Adopting SOA – Experiences from nine Finnish organizations. (2009). *Proceedings of the 13th European Conference on Software Maintenance and Reengineering (CSMR '09)*, Kaiserslautern, Saksa, maaliskuu 2009, 129–138.
- [25] Sprott, D. (2010). The CBDI-SAE Reference Framework in 2010. *CBDI Journal*, syyskuu 2010, <http://everware-cbdi.com/index.php?cID=65&cType=document>. Viitattu 2.4.2012. Everware-CBDI.
- [26] Fettke, P. & Loos, P. (2006). Using Reference Models for Business Engineering – State-of-the-Art and Future Developments. *Proceedings of the Third International Conference on Innovations in Information Technology (IIT '06)*, Dubai, Yhdistyneet arabiemiirikunnat, marraskuu 2006, 1–5.
- [27] Fattah, A. (2009). Enterprise Reference Architecture. *Via Nova Architectura*, kesäkuu 2009, <http://www.via-nova-architectura.org/magazine/magazine/enterprise-reference-architecture.html>. Viitattu 2.4.2012.
- [28] Gruber, T. R. (1993). A Translation Approach to Portable Ontology Specifications. *Knowledge Acquisition – Special issue: Current issues in knowledge modeling*, vol. 5 no 2, kesäkuu 2009, 199–220.

- [29] Broekstra, J., Klein, M., Decker, S., Fensel, D., van Harmelen, F. & Horrocks, I. (2001). Enabling Knowledge Representation on the Web by Extending RDF Schema. *Proceedings of the tenth World Wide Web conference (WWW'01)*, Hongkong, Kiina, 467–478.
- [30] Angelov, S., Grefen, P. & Greefhorst, D. (2009). A Classification of Software Reference Architectures: Analyzing Their Success and Effectiveness. *Joint Working IEEE/IFIP Conference on Software Architecture & European Conference on Software Architecture (WICSA/ECSA 2009)*, Cambridge, UK, syyskuu 2009, 141–150.
- [31] The Open Group, OMG & OASIS. (2009). *Navigating the SOA Open Standards Landscape Around Architecture*. Online white paper, <https://www2.opengroup.org/ogsys/jsp/publications/PublicationDetails.jsp?publicationid=12196>. Viitattu 2.4.2012.
- [32] Jun, L., Zhiqiang, Y., Jianzhong, Q. & Shukuan, L. (2009). A Role-Based Reference Model for the Service Properties of Service Oriented Architecture. *International Forum on Information Technology and Applications (IFITA '09)*, Chengdu, Kiina, toukokuu 2009, 341–344.
- [33] Kung, C. H. (1989). Conceptual Modeling in the Context of Software Development. *IEEE Transactions on Software Engineering*, vol. 15 no 10, lokakuu 1989, 1176–1187.
- [34] U.S. Department of Justice's Global Justice Information Sharing Initiative. (2011). *Global Reference Architecture (GRA) Framework Version 1.9*. <http://it.ojp.gov/docdownloader.aspx?ddid=1223>. Viitattu 2.4.2012.
- [35] U.S. Department of Justice's Global Justice Information Sharing Initiative. *Global Reference Architecture (GRA)*. <http://www.it.ojp.gov/default.aspx?area=nationalInitiatives&page=1015>. Viitattu 2.4.2012.

- [36] The Open Group. (2010). *Service-Oriented Architecture Ontology*. <https://www2.opengroup.org/ogsys/jsp/publications/PublicationDetails.jsp?catalogno=c104>. Viitattu 2.4.2012.
- [37] The Open Group. *Boundaryless Information Flow*. <http://www3.opengroup.org/aboutus/vision/bif>. Viitattu 2.4.2012.
- [38] Gerber, A., Kotzé, P. & van der Merwe, A. (2010). Towards the Formalisation of the TOGAF Content Metamodel Using Ontologies. *Proceedings of the 12th International Conference on Enterprise Information Systems (ICEIS), Volume 2: Artificial Intelligence and Decision Support Systems*, Funchal, Madeira, Portugali, kesäkuu 2010, 54–64.
- [39] Harding, C. J. (2011). *Deciphering The Open Group's SOA Ontology*. Konferenssiesitys, SOA + Cloud Symposium, Brasilia, huhtikuu 2011. http://www.soasympoium.com/home2011/pdf_brazil/Chris_Harding_Deciphering_the.pdf. Viitattu 2.4.2012.
- [40] IBM, HP, Software AG & TIBCO. (2010). *SOA - Repository Artifact Model and Protocol Specification (S-RAMP) Foundation Revision 1.0*. <http://www.oasis-open.org/committees/download.php/40620/FS-RAMP-Specification-Foundation-FINAL.pdf>. Viitattu 2.4.2012.
- [41] Vitvar, T., Mocan, A., Kerrigan, M., Zaremba, M., Zaremba, M., Moran, M., Cimpian, E., Haselwanter, T. & Fensel, D. (2007). Semantically-enabled service oriented architecture: concepts, technology and application. *Service Oriented Computing and Applications (SOCA)*, vol. 1 no 2, 129–154.
- [42] OWL-S Coalition. (2008). *OWL-S 1.2 Release*. <http://www.ai.sri.com/dam/services/owl-s/1.2/>. Viitattu 2.4.2012.

- [43] Conceptual Models for Services Working Group (CMS WG). (2005). *D1v1.0 Web Service Modeling Ontology (WSMO) CMS WG Final Draft*. <http://cms-wg.sti2.org/doc/D1v1.0%20Web%20Service%20Modeling%20Ontology%20%28WSMO%29.html>. Viitattu 2.4.2012.
- [44] Noy, N. F. & McGuinness, D. L. (2001). *Ontology Development 101: A Guide to Creating Your First Ontology*. Stanford Knowledge Systems Laboratory Technical Report KSL-01-05 and Stanford Medical Informatics Technical Report SMI-2001-0880, maaliskuu 2001, <http://www.ksl.stanford.edu/people/dlm/papers/ontology-tutorial-noy-mcguinness-abstract.html>. Viitattu 2.4.2012.
- [45] Lublinsky, B. (2009). *A Reference Architecture Foundation for SOA Draft Was Submitted to Public Review*. <http://www.infoq.com/news/2009/11/RAFSOA>. Viitattu 2.4.2012.
- [46] Nickull, D. & Estefan, J. (2009). *Reference Model and Reference Architecture for SOA*. Konferenssiesitys, 21st Enterprise Architecture Practitioners Conference, San Diego, Kalifornia, USA, helmikuu 2009, www.opengroup.org/conference-live/uploads/40/18505/nickull.pdf. Viitattu 2.4.2012.
- [47] Arsanjani, A., Zhang, L.-J., Ellis, M., Allam, A. & Channabasavaiah, K. (2007). S3: A Service-Oriented Reference Architecture. *IT Professional*, vol. 9 no 3, toukokuu–kesäkuu 2007, 10–17.
- [48] IBM. (2011). *Cloud Computing Reference Architecture v2.0*. <http://www.opengroup.org/cloudcomputing/uploads/40/23840/CCRA.IBMSubmission.02282011.doc>. Viitattu 2.4.2012.

- [49] Krishnan, D. (2010). *Gartner Vendor Report: Application Infrastructure For Systematic SOA-Style Application Projects*. <http://www.infoq.com/news/2010/12/gartner-systematic-soa-report>. Viitattu 2.4.2012. Toissijainen lähde.
- [50] Current Analysis. *Service Oriented Architecture (SOA) Suites*. <http://www.currentanalysis.com/P/bts-SOA.asp>. Viitattu 2.4.2012.
- [51] Davies, J., Shaffer, D. & L'her, D. (2009). *Oracle SOA Suite 11g*. Online white paper, <http://www.oracle.com/technetwork/middleware/soasuite/overview/wp-soa-suite-11gr1-2-129551.pdf>. Viitattu 2.4.2012.
- [52] Oracle. *Oracle Technology Network: SOA Suite*. <http://www.oracle.com/technetwork/middleware/soasuite/index.html>. Viitattu 2.4.2012.
- [53] Oracle. *Oracle Fusion Middleware Release 1 (11.1.1.5) Documentation Library: Oracle SOA Suite, Business Process Management Suite, and Web Services Documentation*. http://docs.oracle.com/cd/E21764_01/soa.htm. Viitattu 2.4.2012.
- [54] OASIS Open CSA. *Service Component Architecture*. <http://www.oasis-open.org/sca>. Viitattu 2.4.2012.

Liite B

The Open Groupin referenssiarkkitehtuurin palvelukategorioiden kuvaukset

The Open Groupin referenssiarkkitehtuurissa esitetyt palvelukategoriat on kuvattu lyhyesti alla. Kuvaukset perustuvat referenssiarkkitehtuuriin. [23]

1. **Vuorovaikutuspalvelut** tukevat vuorovaikutusta sovelluksen ja käyttäjien välillä.
2. **Prosessipalvelut** tukevat palvelujen yhdistelyä tukemalla esimerkiksi liiketoimintaprosessien tai liiketoimintasäännöillä ohjattujen sovellusten toteuttamista.
3. **Informaatiopalvelut** tukevat pääsyä dataan, datan yhdistelyä ja datavirtoja.
4. **Liiketoimintasovelluspalvelut** toteuttavat ydinliiketoimintalogiikkaa.
5. **Pääsypalvelut** (engl. Access Services) tukevat perinnesovellusten integrointia SOAan.
6. **Partneripalvelut** tukevat yrityksen ja sen partnereiden välistä vuorovaikutusta.
7. **Yhdistettävyysspalvelut** (engl. Service Connectivity Services), kuten palveluväylä (engl. enterprise service bus, ESB), tukevat yhteyttä palvelun tarjoajan ja kuluttajan välillä.
8. **Omaisuus- ja rekisteripalvelut** (engl. Asset and Registry Services) tarjoavat pääsyn omaisuuteen, kuten palvelukuvauksiin, palveluihin ja politiikkoihin.

9. **Infrastruktuuripalvelut** tarjoavat infrastruktuurin toiminnallisuutta palveluina ja virtualisoivat resurssi riippuvuuksia.
10. **Hallintapalvelut** tukevat SOAn monitorointia.
11. **Liiketoimintapalvelut** tarkastelevat liiketoimintasovelluksia ja parantelevat niitä tosiaikaisen metriikan perusteella.
12. **Strategia- ja suunnittelupalvelut** tukevat yrityksen kehittämistä esimerkiksi tuottamalla yrityksen tavoitetilaa tukevan siirtymäsuunnitelman.
13. **Kehityspalvelut** tarjoavat erilaisten sovelluskehitystyökalujen toiminnallisuutta palveluina.
14. **Elinkaaripalvelut** tukevat palvelujen ja muiden SOA-elementtien elinkaaren hallintaa.