

MDG Teknologia Enterprise Architect työkalussa: Case Wärtsilä

Nico Nilsson



Tekijä(t) Nico Nilsson	
Koulutusohjelma Tietojenkäsittelyn koulutusohjelma	
Raportin/Opinnäytetyön nimi MDG Teknologia Enterprise Architect työkalussa: Case Wärtsilä	Sivu- ja liitesivumäärä 43+12
<p>Opinnäytetyön aihe on peräisin Wärtsilässä toimivasta Enterprise Architecture tiimistä. Työn tarkoituksena oli luoda Enterprise Architect ohjelmistossa olevan MDG Teknologian avulla työkalulaatikoita. Työkalulaatikoiden avulla haluttiin yhtenäistää eri diagrammien ulkoasua rajoittamalla saatavilla olevien elementtien määrää. Tämä helpotti mallintajien työtä ja teki diagrammeista yhteneväisemmän näköisiä.</p> <p>Opinnäytetyö alkaa luvulla kohdeyrityksestä eli Wärtsilästä. Luku keskittyy kertomaan Wärtsilän tämän hetkisestä strategiasta sekä kestävästä kehityksestä. Luvussa esitellään myös yksittellen eri Wärtsilän liiketoimintalinjat; Energy Solutions, Marine Solutions ja Services.</p> <p>Opinnäytetyössä mainitaan seuraavaksi kokonaisarkkitehtuuriista painottaen sen tuomiin hyötyihin yritykselle ja mitä yritykseltä vaaditaan sen onnistuneeseen toteutukseen. Käydään läpi eri viitekehysjä, kuten TOGAF, Zachman Frameworkin sekä Julkisen hallinnon KA-kypsyystasomalli.</p> <p>Seuraavaksi kerrotaan miten ytyössä käytettiin hyödyksi SPARX Systemsin kehittämää Enterprise Architect ohjelmistoa sekä sen tarjoaman MDG Teknologian mahdollisuuksia.</p> <p>Lopuksi kerrotaan itse projektista ja miten se toteutettiin alkaen lähdetiedon etsimisestä päättyen elementtien ja connectorien viimeistelyyn muokkaamalla niiden ulkoasua. Aloitetaan kertomalla projektin alustuksesta sekä esittämällä mallin, jota noudatettiin rakennettaessa MDG Teknologiaa. Kerrotaan myös vaihe vaiheelta miten MDG Teknologia rakennettiin ja miten rakennetun teknologian saa halutessaan käyttöön. Lopuksi kerrotaan mitä konkreettista hyötyä rakennetuista työkalulaatikoista on ja miten se näkyy itse ohjelmistossa.</p> <p>Opinnäytetyö päättyy pohdinnalla, jossa käydään läpi projektin kulkua ja pohditaan asioita, jotka vaikuttivat lopulta projektin lopputulokseen.</p>	
Asiasanat Kokonaisarkkitehtuuri, ohjelmistot, kehitys, diagrammit	

Sisällys

1	Johdanto	2
2	Wärtsilä	3
2.1	Strategia	3
2.2	Kestävä kehitys	4
2.3	Liiketoiminta-alueet	6
2.3.1	Energy Solutions	6
2.3.2	Marine Solutions	6
2.3.3	Services	7
3	Kokonaisarkkitehtuuri	9
3.1	TOGAF 9.1	9
3.1.1	Tuetut arkkitehtuurimallit	10
3.1.2	ADM	12
3.2	Zachman Framework	13
3.3	Julkisen hallinnon KA-kypsyystasomalli	14
3.3.1	Kypsyystasomallin portaat	15
3.3.2	Kypsyystasomallin tavoitteet ja tarkoitus	17
3.3.3	Kypsyystasomallin käyttö	17
4	Enterprise Architect	19
4.1	Enterprise Architect työkaluna	19
4.2	MDG Teknologia	20
5	Projekti	21
5.1	Projektin alustus	21
5.2	Projektin toteutus	22
5.2.1	EATemplate Profile-kansio	23
5.2.2	EATemplate Toolbox-kansio	26
5.2.3	EATemplate Diagram-kansio	29
5.2.4	MDG tiedoston luonti	30
5.2.5	MDG Teknologian käyttöönotto	35
5.2.6	MDG Teknologian testaaminen	36
5.2.7	MDG Teknologian jakaminen	37
5.2.8	Shape Script	38
5.3	Projektin yhteenveto	39
6	Pohdinta	40
	Lähteet	42
	Liitteet	44
	Liite 1. EATemplate Profile 1(3)	44
	Liite 2. EATemplate Profile 2(3)	45

Liite 3. EATemplate Profile 3(3).....	46
Liite 4. EATemplate Toolbox Application Landscape Diagram.....	47
Liite 5. EATemplate Toolbox Business Process Diagram	48
Liite 6. EATemplate Toolbox Business Strategy Map	49
Liite 7. EATemplate Toolbox EATemplate	50
Liite 8. EATemplate Toolbox Enterprise Data Model – Level 1	51
Liite 9. EATemplate Toolbox Product Portfolio Map.....	52
Liite 10. EATemplate Toolbox Value Network Diagram	53
Liite 11. EATemplate Diagram 1(2)	54
Liite 12. EATemplate Diagram 2(2)	55

Termit ja lyhenteet

ADM	TOGAF arkkitehtuurin kehitysmetodi
Archimate	Avoin graafinen mallinnuskieli
BPMN	Graafinen liiketoimintaprosessien mallintamiseen tarkoitettu mallinnuskieli
Connector	Määrittelevät elementtien väliset suhteet, kuten perimisen ja erilaiset tietovirrat
Elementti	Mallinnusyksikkö, joita yhdistelemällä connectoreita hyväksi käyttäen saadaan aikaiseksi diagrammeja
Enterprise Architect	SPARX Systemsin kehittämä UML-pohjainen suunnitteluun ja analyysiin tarkoitettu mallinnustyökalu
KA	Kokonaisarkkitehtuuri/Enterprise Architecture
MDG Teknologia	Enterprise Architectin sisältämä työkalu, jolla voidaan luoda uusia tai muokata jo olemassa olevia teknologioita ja mallinnuskieliä
Shape Script	Enterprise Architectissa oleva kieli, jonka avulla voi muokata objektien ulkoasua
Stereotyyppi	Käytetään hyödyksi elementtien kanssa niiden ominaisuuksien laajentamiseen
UML	Standardoitu graafinen mallinnuskieli
XML	Metakieli, jolla määritellään rakenteellisia merkintäkieliä

1 Johdanto

Aloitin Wärtsilällä Marine Solutions liiketoiminta-alueen Enterprise Architecture tiimissä kesätyöntekijänä ja eräs minulle annetuista tehtävistä oli keksiä keino miten rakentaa työkalulaatikoita, joihin voisi koota juuri tiettyyn diagrammiin sopivat elementit ja connectorit ilman turhia työkaluja.

Tiimillämme oli käytössä SPARX Systemsin kehittämä Enterprise Architect ja ohjelmistossa on käytössä monia eri teknologioita aina Archimatesta BPMN:ään. Jokainen teknologia pitää sisällään eri tarkoituksiin soveltuvia elementtejä ja niitä yhdistäviä connectoreita. Enterprise Architectin käyttäjän haasteena kuitenkin on valita mitä elementtejä ja connectoreita käyttää juuri tiettyntyyppiseen diagrammiin, koska niitä molempia on tarjolla valtavat määrät.

Hetken kuluttua tehtävän aloittamisen jälkeen löysin SPARX Systemsin omista materiaaleista tavan miten rakentaa toimiva työkalulaatikko Enterprise Architectiin. Ainut keino suorittaa tämä oli käyttää apuna ohjelmistoon sisään rakennettua MDG Teknologiaa. MDG Teknologian avulla voi muokata jo olemassa olevia elementtejä ja connectoreita sekä tehdä tarpeen tullen kokonaan uusia. Päädyin tehtävässä muokkaamaan jo olemassa olevia elementtejä ja connectoreita sekä tarvittaessa muokata niiden ulkoasua niin värien kuin myös ulkomuodon osalta. Ulkomuodon muokkaus tehtiin käyttäen hyväksi Shape Script nimistä Enterprise Architectiin sisäänrakennettua ohjelmointikieltä, joka on suunniteltu muokkaamaan eri elementtejä käyttäjän haluamalla tavalla.

Tämän tehtävän ratkaisuun ja toteutukseen perustuu opinnäytetyöni ja valitsinkin juuri tämän aiheen opinnäytetyökseksi sen haastavuuden vuoksi. Tehtävää aloittaessani minulla ei ollut lainkaan kokemusta Enterprise Architect ohjelmistosta, joten aloitin tutustumalla sen käyttöliittymään eri lähteiden kautta. Koska ohjelmisto on suhteellisen pieni, tarvittava lähdemateriaali löytyi lähinnä sen kehittäjän SPARX Systemsin omilta sivuilta. Löysin lähteitä edellisen lisäksi vain yhdestä toisesta lähteestä, tanskalaiselta IT-yritykseltä TigerTeamsilta. Edellä mainittujen kahden lähteen pohjalta lähdin kehittämään työkalupakkeja ja monen mutkan kautta sain työkalupakit toimimaan toivomallani tavalla ja koen, että niistä on konkreettista hyötyä niitä käyttäville työntekijöille.

2 Wärtsilä

Wärtsilä on vuonna 1834 perustettu yritys, jolla on kansainvälisesti johtava asema edistyksellisen teknologian ja kokonaislinkaariratkaisujen toimittajana merenkulku- ja energiamarkkinoilla. (Wärtsilä Lyhyesti 2017)

Wärtsilän visiona ja missiona on olla asiakkaiden arvostetuin kumppani tehostamalla asiakkaiden liiketoimintaa ja muokata merenkulku- ja energiamarkkinoita kehittämällä edistyksellisiä teknologioita keskittymällä elinkaaren aikaiseen suorituskykyyn. Ympäristötehokkuus on myös yksi avainsanoista Wärtsilän liiketoiminnassa. (Tämä on Wärtsilä 2017)

Wärtsilällä on yhteensä ympäri maailmaa noin 18 000 työntekijää. Suurin osa työntekijöistä työskentelee Euroopassa, jossa työntekijöitä on yhteensä noin 58% Wärtsilän henkilöstömäärästä. Toiseksi eniten työntekijöitä sijaitsee Aasiassa, jossa työntekijöitä on noin 28% prosenttia Wärtsilän henkilöstömäärästä. Muut maanosat, joissa Wärtsilällä on toimipisteitä ovat muun muassa Pohjois- ja Etelä-Amerikassa sekä Afrikassa. (Tämä on Wärtsilä 2017)

2.1 Strategia

Strategiana toimii tällä hetkellä tähtääminen kannattavaan kasvuun tarjoamalla asiakkaille teknologiaa ja elinkaariratkaisuja merenkulku- ja energia-alalla. Asiakkaiden tavat toimia ovat muuttuneet viime vuosien saatossa huomattavasti ja asiakkaiden ympäristötietouden lisääntyminen ja muuttuvat energiatarpeet ovat pakottaneet myös Wärtsilän muutamaaan toimintatapaan. Asiakkaiden ympäristöystävällisiin ja energiatehokkaisiin tarpeisiin pystytään vastaamaan nykyaikaisten tuotteiden ja palvelujen avulla. Esimerkkinä monipolttoainetuotteet ja nesteytettyä maakaasua hyödyntävien moottoreiden (LNG) ratkaisut ovat alan johtavia ja vastaavat ympäristöystävällisten tuotteiden jatkuvasti kasvavaan kysyntään. Panostus digitalisaatioon ja sen tutkimukseen tarjoavat asiakkaille lisäarvoa ja samalla panostukset luovat vankan perustan, joka turvaa Wärtsilän aseman johtavana markkinainnovaatioiden kehittäjänä ja samalla vahvistaa sitä entisestään. (Wärtsilä Strategia 2017)

Organisaatiosta tekee erittäin suorituskykyisen sen ammattitaitoiset ja sitoutuneet työntekijät. Tämän mahdollistaa Wärtsilän innovaatiokulttuuri sekä turvallisuuden, monimuotoisuuden ja korkeiden eettisten periaatteiden korostaminen. (Tämä on Wärtsilä 2017)

Wärtsilän tavoitteena on pitää koko yrityksen toiminta tehokkaana, kannattavana ja kilpailukykyisenä eri markkinoilla. Edellä mainittujen tavoitteiden avulla pyritään täyttämään osakkeenomistajien odotukset sekä myös edesauttaa ja edistää yhteiskunnan hyvinvointia. Ympäristöasioihin panostaminen näkyy myös yrityksen päätavoitteessa, joka on toimittaa teknologioita ja palveluita, joiden hyötysuhde on korkea ja ympäristökuormitus vähäinen. Tähän tavoitteeseen voidaan päästä jatkuvasti kehittämällä eri liiketoiminta-alueiden tuotteiden ja palveluiden ympäristösuorituskykyä eteenpäin, ottaen huomioon niiden elinkaarinäkökulman. Myös eräs huomion kohde on työskentely tutkimuksen ja kehittämistoiminnan parissa ja hyödyntää uutta teknologiaa, jotta Wärtsilä voi säilyttää markkinoiden teknologiajohtavuuden. (Wärtsilä Strategia 2017)

2.2 Kestävä kehitys

Kestävällä kehityksellä on erittäin suuri rooli Wärtsilän liiketoiminnassa. Wärtsilä on sitoutunut kestäväan kehitykseen ja vastuulliseen liiketoimintaan. Sitoumus perustuu yrityksen missioon, visioon ja strategiaan, joiden avulla luodaan perusta yhtiön tuotteiden ja toimintojen kehittämiseksi. (Wärtsilä Kestävä Kehitys 2017)

Keskeinen asema kestävässä kehityksessä on teknologinen johtajuus kilpailijoihin verrattuna. Erityisesti liiketoiminta-alueet Wärtsilä Energy Solutions ja Wärtsilä Marine Solutions keskittyvät kestävien ratkaisujen kehittämiseen ja tarjoamiseen omien palveluiden ja tuotteidensa kautta asiakkailleen. Wärtsilä Services puolestaan on keskittynyt tarjoamaan tukea eri Wärtsilän tarjoamille tuotteille ja palveluille sekä alan johtavaa teknologiaa päivitys- ja modernisointikonaisuuksien kautta käytössä oleviin laitteisiin. Korkeat juridiset ja eettiset normit ovat yhtiössä myös erittäin suuressa roolissa, joten myös toimittajien ja liikekumppaneiden edellytetään osana Wärtsilän kokonaisarvoketjua noudattamaan niitä. (Tämä on Wärtsilä 2017)

Kestävän kehityksen edistämiseksi tärkein rooli Wärtsilässä on toimittaa asiakkaille ympäristöystävällisiä ratkaisuja ja palveluita. Wärtsilä investoi jatkuvasti uusien teknologioiden kehittämiseksi ja yhä tehokkaampien ja ympäristöystävällisempien ratkaisujen löytämiseksi. Tehokkaampia ja ympäristöystävällisempiä ratkaisuja voidaan saavuttaa monin eri tavoin, kuten parantamalla hyötysuhdetta, vähentämällä kaasumaisia ja nestemäisiä päästöjä ja jätteiden määrää, torjumalla melua sekä panostamalla jäte- ja painolastivesien puhdistukseen. Näiden ratkaisujen avulla asiakkaat pystyvät myös kehittämään omaa liiketoimintaansa kestävämpään suuntaan niin lyhyellä, kuin pitkällä tähtäimellä. (Wärtsilä Kestävä Kehitys 2017)

Wärtsilän kestävään kehitykseen kuuluu kolme pääpainopistettä: talous, ympäristö, ja sosiaalinen vastuu. Keskittymällä talouteen, yrityksen kannattavuus lisääntyy. Keskittymällä ympäristöön, laajenee tuoteportfolio entistä ympäristöystävällisimmillä tuotteilla ja palveluilla. Keskittymällä sosiaaliseen vastuuseen, vastuullinen liiketoiminta vahvistuu samalla. Wärtsilä valmistaa monia eri moottoreita ja ympäristökeskeisyys on niissäkin erittäin tärkeässä asemassa. Tärkeimpiä ominaisuuksia tämän sukupolven moottoreissa ovatkin mm. luotettavuus, turvallisuus ja pitkä käyttöikä. Ympäristöystävällisistä ominaisuuksista voidaan mainita raskaalle polttoöljylle vaihtoehtoiset polttoaineet, mahdollisuus käyttää moottoreissa joustavasti eri polttoaineita, sekä moottorin maksimaalinen hyötysuhde ja alhaiset elinkaarikustannukset. (Wärtsilä Kestävä Kehitys 2017)

Wärtsilän kehittämä uuden sukupolven Wärtsilä 31 –moottori onkin ympäristöystävällisten ratkaisujen avulla saavuttanut tavoitteet korkeampiin hyötysuhteisiin, dieselpolttoainekulutuksen ollessa jopa niin alhainen, kuin 165g / kWh. Moottoria onkin kehitetty ympäristö edellä, joten siitä on tehty kolme eri versiota: diesel-, monipolttoaine- (DF) ja kaasumoottori (SG). Wärtsilä 31 –moottori (kuva 1) onkin saanut Guinness World Records järjestöltä tunnustuksen maailman tehoikkaimpana nelitahtidieselmoottorina ja tämä on osoitus Wärtsilän johtoasemasta markkinoilla uuden teknologian kehittäjänä. (Wärtsilä Media 2017)



Kuva 1. Wärtsilä 31 (Wärtsilä Products 2017)

2.3 Liiketoiminta-alueet

Wärtsilällä on kolme liiketoiminta-aluetta: Energy Solutions, Marine Solutions ja Services. Energy Solutionsin liikevaihto on noin 20%, Marine Solutionsin 35% ja Servicen 45% koko Wärtsilän nykyisestä liikevaihdosta.

2.3.1 Energy Solutions

Wärtsilä Energy Solutionsilla työskentelee noin 900 henkilöä ja Energy Solutions on tällä hetkellä johtava kansainvälinen järjestelmäintegraattori, joka on tarjoaa asiakkailleen ympäristökeskeisiä ratkaisuja.

Eri tuotteisiin kuuluu polttomoottorikäyttöisiä voimalaitoksia, aurinkovoimaloita sekä nesteytetyn maakaasun (LNG) terminaali- ja jakelujärjestelmiä. Voimalaitosten kysyntä on riippuvainen väestön kasvusta ja talouden kehityksestä. Tämän vuoksi sähkönkulutuksen kasvaessa uusien voimalaitosten rakentaminen sekä vanhaa kapasiteettia korvaavien laitosten tarve lisääntyy. Myös poliittinen paine panostaa vähähiilisiin voimalaitoksiin lisää uusiutuvien energiantuotannon kasvua. (Tämä on Wärtsilä 2017)

Wärtsilä Energy Solutionin vahvuudet muihin alalla oleviin kilpailijoihin verrattuna ovat joustavuus sekä polttoaineissa että operoinnissa, tuotteiden ja palvelujen energiatehokkuus ja päästömääräysten noudattaminen, hybridilaitokset, ja kilpailukykyiset pääomakustannukset ja mahdollisuus toimittaa kokonaisratkaisuja. (Tämä on Wärtsilä 2017)

Wärtsilä Energy Solutions on toimittanut tuotteitaan maailmanlaajuisesti 176 eri maahan ja toimittamiensa voimalaitosten yhteiskapasiteetti on 63 GW. (Tämä on Wärtsilä 2017)

2.3.2 Marine Solutions

Wärtsilä Marine Solutions toimittaa meriteollisuudessa sekä öljy- ja kaasuteollisuudessa toimiville asiakkaille turvallisia, ympäristöystävällisiä, tehokkaita, joustavia ja taloudellisia ratkaisuja. (Tämä on Wärtsilä 2017)

Toimialaan vaikuttavia tekijöitä ovat pääasiassa maailmantalouden kehitys ja sen vaikutus kauppaan ja kuljetuskapasiteetin tarpeeseen. Muita tekijöitä, jotka vaikuttavat muun muassa öljy- ja kaasuteollisuuden kehitykseen ovat polttoaineiden hinnat, saatavuus ja kysyntä. Kasvavat polttoainekustannukset sekä ympäristömääräysten kehitys kasvattavat

uusien yhä taloudellisempien aluksien, kuten kaasukäyttöisten alusten (LNG) kysyntää. Muita vaikuttavia tekijöitä ovat telakoiden kapasiteetti, uusien alusten hinnat, vanhojen alusten käytöstä poistaminen ja romuttaminen, korkotaso, sekä rahtihinnat. (Tämä on Wärtsilä 2017)

Wärtsilä Marine Solutionsin vahvuuksiin kuuluu kilpailijoihin verrattuna merenkulku- sekä offshore-alan laajin valikoima tuotteita ja ratkaisuja sekä kattavin valikoima vaihtoehtoja polttoainejoustavuutta, suorituskykyä ja ympäristövaatimuksia koskeviin tarpeisiin. (Tämä on Wärtsilä 2017)

2.3.3 Services

Wärtsilä Services on selvästi suurin Wärtsilän liiketoiminta-alueista yli kahden miljardin euron liikevaihdolla. Wärtsilä Services puolella työskentelee tällä hetkellä noin 11 000 henkeä 160 eri paikkakunnalla. Palveluverkosto on alan laajin ja se käsittää yhteensä noin 12 000 asiakasta ympäri maailmaa.

Wärtsilä Services liiketoiminta-alueen tehtävänä on tukea Wärtsilän asiakkailleen toimitettuja tuotteita ja järjestelmiä niiden koko elinkaaren ajan optimoimalla laitteiston hyötysuhdetta ja suorituskykyä. Asiakkaita tuetaan laitteiden käytettävyyden parantamisen ohella myös tarjoamalla varaosia, sekä käyttö-, hallinnointi- ja optimointipalveluja. Suurin kysyntätekijä huoltoliiketoiminnan kannalta on Wärtsilän asennetun laitekannan suuruus ja kehitys. Kunnossapito-, telakointi-, ja käyttöönototarpeeseen vaikuttaa suoraan markkinatilanne. Markkinatilanteella on myös suoraa vaikutusta kaluston ja laitteiden käyttöasteeseen. Kaluston elinkaarivaihe taas vaikuttaa kunnossapito- ja mahdollisiin tasokorotus- tai modernisointitarpeisiin. Laitteiston, huollon ja suorituskyvyn optimointia edistävät solmitut kumppanuussopimukset. Kumppanuussopimukset taas parantavat toiminnallista tehokkuutta, alentavat kustannuksia ja lisäävät liiketoiminta-alueen kasvua. (Tämä on Wärtsilä 2017)

Wärtsilä Services liiketoiminta-alueen vahvuuksia muihin alan kilpailijoihin verrattuna ovat pitkäaikaiset asiakassuhteet sekä asiakkaiden tarpeiden ymmärtäminen. Vahvuuksiin kuuluu myös toimitettujen laitteiden koko elinkaaren kattava tarjonta sekä alan laajin huoltotarjonta, jota tukee Wärtsilän maailmanlaajuinen huoltoverkosto. Huoltoverkoston kautta tulee myös kyky optimoida laitteiden käyttöä ja suorituskykyä. Myöskään digitalisaatiota ei olla unohdettu, vaan asiakkaille tarjotaan moderneja digitaalisia ratkaisuja, joiden avulla asiakkaat pystyvät tehostamaan laitteidensa toimintaa käyttäen hyödyksi reaaliaikaista dataa ja analytiikkaa. Services liiketoiminta-alueella, kuten myös

muilla liiketoiminta-alueilla tavoitteena on tuotteiden ja palvelujen korkea laatu, asiantunteva tuki, palvelujen varma saatavuus sekä toiminta, joka ottaa huomioon ympäristön. (Tämä on Wärtsilä 2017)

3 Kokonaisarkkitehtuuri

Kokonaisarkkitehtuuri on strategian käytäntöön panoa. Kokonaisarkkitehtuuri auttaa sekä yrityksen nykytilan analysoinnissa että myös muutoksen läpiviennissä antaen menetelmän, välineet ja johdonmukaisen työskentelytavan, jolla yrityksen strategiset tavoitteet saadaan konkretisoitua aina toimenpiteisiin asti. Kokonaisarkkitehtuurilla on keskeinen rooli esimerkiksi digitalisaatiossa eli yrityksen toiminnan uudistamisessa ja kehittämisessä tietotekniikan keinoin. Kuvassa 2 on esitettyinä Valtiovarainministeriön käyttämä malli, joka perustuu julkisen hallinnon malliin JHS 179. Kokonaisarkkitehtuurilla saadaan myös välineet ja menettelyt tiedon hallintaan, määrittelyyn ja hyödyntämiseen. (Valtiovarainministeriö 2017)



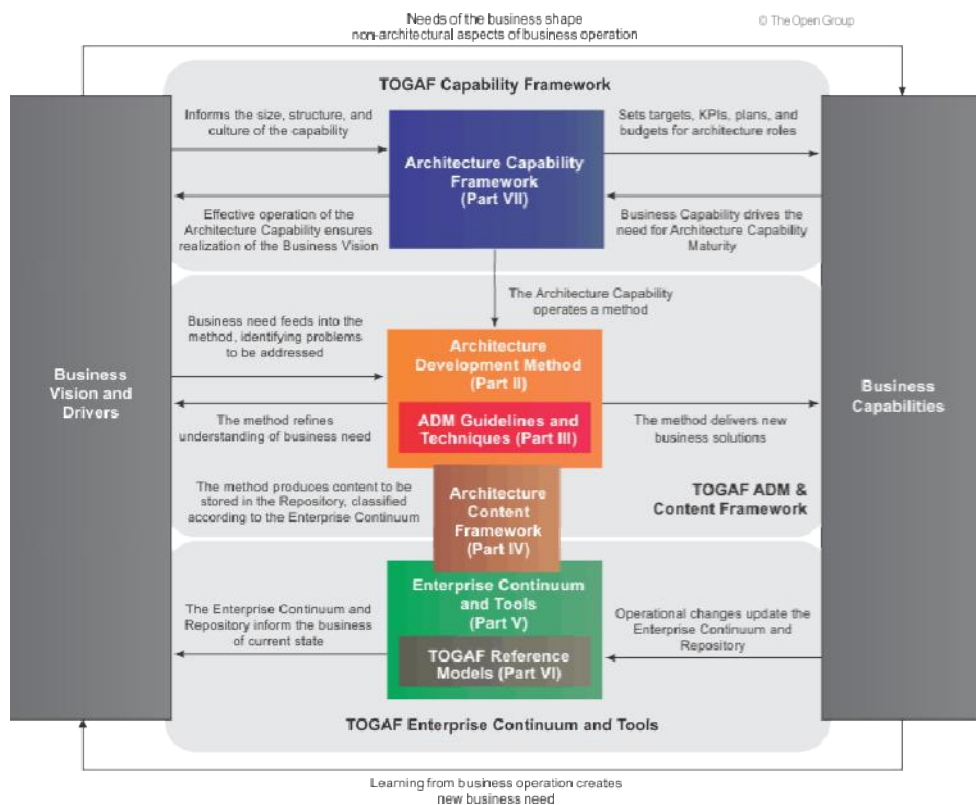
Kuva 2. Valtiovarainministeriön kokonaisarkkitehtuuri digitalisaatiossa
(Valtiovarainministeriö kokonaisarkkitehtuuri 2017)

Kolme suosittua viitekehysmallia ovat TOGAF 9.1, Zachman Framework, sekä TOGAF:iin perustuva Julkisen hallinnon malli. Seuraavissa luvuissa ovat kaikki nämä kolme viitekehysmallia esiteltyinä.

3.1 TOGAF 9.1

TOGAF 9.1 on The Open Groupin kehittämä ja ylläpitämä viitekehysmalli. TOGAF 9.1 on kehitetty kokonaisarkkitehtuurin kehittämistä varten yritysten eri tarpeisiin. Ensimmäinen TOGAF:n malli esiteltiin jo vuonna 1995 ja se perustui silloin Yhdysvaltain Puolustusministeriön kehittämään tekniseen arkkitehtuurin viitekehukseen tietohallintoon. Puolustusministeriö antoi The Open Groupille luvan kehittää uutta viitekehystä heidän kehöksensä päälle, joka oli jo itsessään monen vuoden kehityksen tulos ja monen miljoonan dollarin investointi. (TOGAF 9.1 docs 2017)

TOGAF:ia on auttanut kehittämään ja viemään eteenpäin yli 300 arkkitehtuuriseen foorumiin kuuluvaa yritystä ja organisaatiota, joista jotkin ovan maailmanlaajuisesti johtavassa asemassa alallaan. TOGAF:in käyttäminen yrityksen kokonaisarkkitehtuurissa varmistaa, että arkkitehtuuri on läpi yrityksen tasaista, kuvaa hyvin eri sidosryhmien tarpeita ja ottaa huomioon yrityksen nykytilan sekä tarkkailee tulevaisuudessa tulevia tarpeita. Kokonaisarkkitehtuurin kehittäminen ja ylläpitäminen on teknisesti monimutkaista ja aikaa vievää. TOGAF:illa on iso rooli arkkitehtuurin kehittämisprosessin standardoimisessa ja TOGAF tarjoaa viitekehysten (kuva 3), joka antaa yritykselle avaimet rakentaa juuri tietyille yritykselle toimivan ratkaisun. (TOGAF 9.1 docs 2017)



Kuva 3. TOGAF:n kyvykkyysskartta (TOGAF 9.1 docs 2017)

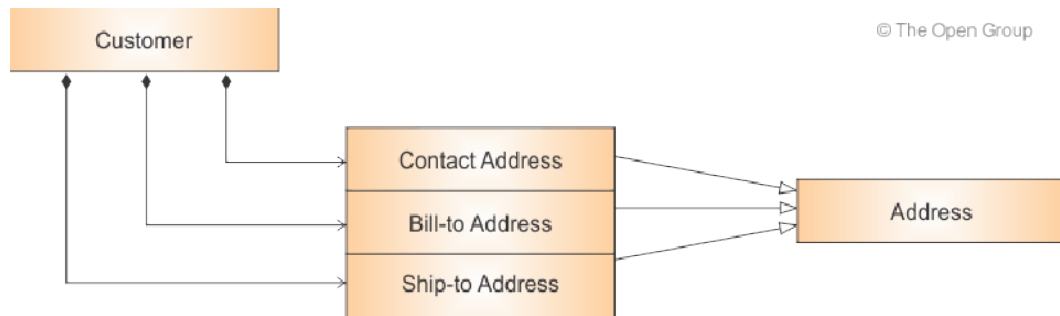
3.1.1 Tuetut arkkitehtuurimallit

TOGAF tukee neljää eri arkkitehtuurimallia: Liiketoiminta-arkkitehtuuria, Data-arkkitehtuuria, Sovellusarkkitehtuuria ja Teknologia-arkkitehtuuria.

Liiketoiminta-arkkitehtuuri määrittelee yrityksen liiketoimintastrategian, hallintotavan sekä tärkeimmät liiketoimintaprosessit. Liiketoiminta-arkkitehtuurin tarkoitus on myöskin kuvata miten yrityksen täytyy toimia saavuttaakseen tavoitteensa. Tietämys liiketoiminta-arkkitehtuurissa on edellytys minkään muun arkkitehtuurimallin työstämiselle ja sen vuoksi liiketoiminta-arkkitehtuuri on ensimmäinen arkkitehtuurimalli, jota aletaan työstää.

Liiketoiminta-arkkitehtuuria tarvitaan myös käytännössä osoittamaan liiketoiminnillista arvoa sidosryhmille. Liiketoiminta-arkkitehtuurin laajuus riippuu yrityksen kokonaisarkkitehtuurin laajuudesta. Liiketoiminnan mallintamiseen voidaan käyttää avuksi UML pohjaisia diagrammeja. (TOGAF 9.1 docs 2017)

Kuvassa 4 on kuvattuna UML-pohjainen esimerkki liiketoimintadiagrammista.



Kuva 4. UML-kuvaus liiketoimintadiagrammista (TOGAF 9.1 docs 2017)

Data-arkkitehtuurin tavoitteena on ymmärtää ja korjata datanhallintaongelmia. Datanhallintaongelmia lähestyttäessä on huomioitava käytössä olevat arkkitehtuuriset standardit ja miten yrityksen eri liiketoimintalinjat käyttävät näitä hyväksi. Data-arkkitehtuurin tuloksena syntyy monia eri arkkitehtuuriin liittyviä dokumentteja, kuten arkkitehtuurikuvausta, arkkitehtuurivaatimusta ja arkkitehtuurikomponenttia kuvaavat dokumentit. (TOGAF 9.1 docs 2017)

Sovellusarkkitehtuurin tavoitteena on auttaa yritystä määrittämään tavoitetilansa sekä olemassa olevat resurssit sovellusarkkitehtuurille. Sovellusarkkitehtuurin tavoitteet määräytyvät yrityksen sille asettaman laajuuden sekä tavoitteiden mukaan. Kuten data-arkkitehtuurissa, sovellusarkkitehtuurin tuloksena syntyy eri arkkitehtuuria kuvaavia dokumentteja. Tuloksena saattaa myös syntyä eri mallinnettuja diagrammeja, kuten sovelluksen käyttötapausdiagrammi (Application Use-Case diagram) sekä sovellusten kommunikaatiodiagrammi (Application Communication diagram). (TOGAF 9.1 docs 2017)

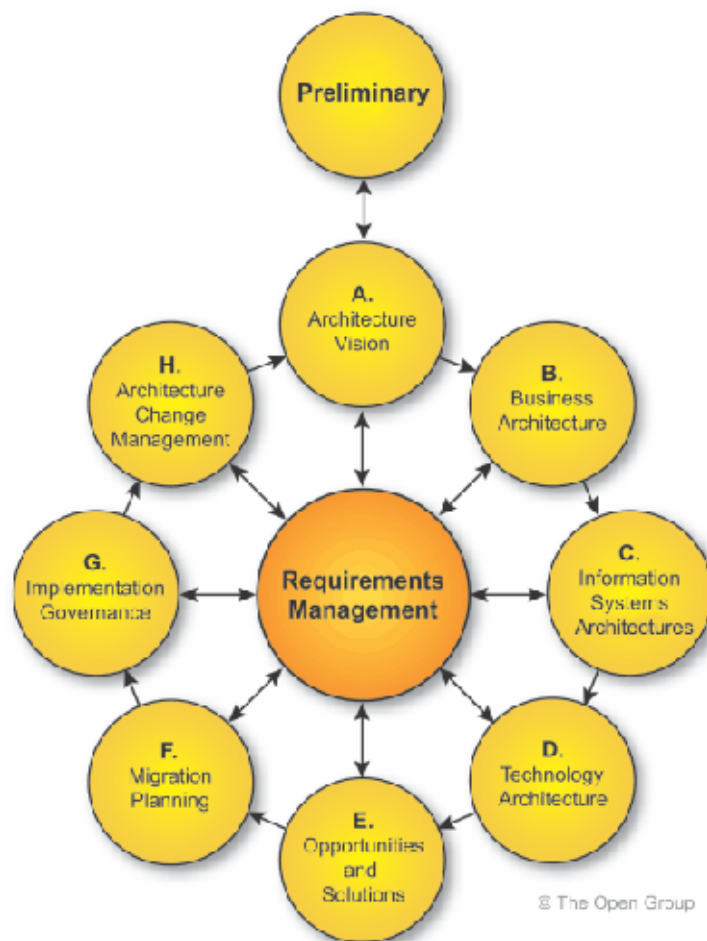
Teknologia-arkkitehtuurin tavoitteena on muiden arkkitehtuurimallien mukaan määrittää yrityksen tavoitetila sekä selvittää mitä teknologioita yritykselle on tarjolla ja mitä niistä kannattaa hyödyntää. Tuloksena teknologia-arkkitehtuurista syntyy eri arkkitehtuurimalleja kuvaavia dokumentteja sekä yrityksen teknologioihin liittyviä katalogeja ja diagrammeja. (TOGAF 9.1 docs 2017)

3.1.2 ADM

TOGAF:n tarjoama ADM (Architecture Development Method) eli metodi arkkitehtuurin kehittämiseksi tarjoaa testatun ja toistettavissa olevan prosessin arkkitehtuurien kehittämiseen. (TOGAF 9.1 docs 2017)

ADM:ään kuuluu arkkitehtuurisen viitemallin käyttöönotto, arkkitehtuurisen sisällön tuottaminen sekä arkkitehtuurien realisoitien tarkkailu. Kaikkia edellä mainittuja eri vaiheita suoritetaan sykleissä, joiden tuloksena yrityksen arkkitehtuuri muuttuu hallittavampaan suuntaan mahdollistaen yrityksen tavoitteisiin pääsyn. (TOGAF 9.1 docs 2017)

ADM:ään (kuva 5) kuuluu yhdeksän eri vaihetta, joista jokaisen on tarkoitus edistää yrityksen tekemää arkkitehtuurityötä. Eri vaiheissa mm. kuvataan eri arkkitehtuurityössä käytettäviä malleja ja tarkastellaan miten arkkitehtuurityö etenee ja onko mahdollisille muutoksille tarvetta. (TOGAF 9.1 docs 2017)



Kuva 5. ADM (Architecture Development Cycle) (TOGAF 9.1 docs 2017)

3.2 Zachman Framework

Zachman Framework on kokonaisarkkitehtuurin kehittämiseen käytetty viitekehysmalli, jonka konseptin John Zachman keksi 1980-luvulla työskennellessään IBM:llä. Zachman Frameworkissa korostuu kysymykset mitä, miten, milloin, kuka, missä ja miksi. Se vastaa näihin kysymyksiin ja auttaa näin hahmottamaan monimutkaisia ideoita. (Zachman 2017)

Itä-Suomen yliopiston ja Aalto-yliopiston teettämässä SOLEA-hankkeessa Zachmanin viitekehysmallia kuvataan muihin malleihin verrattuna eksaktiksi ja käytännölliseksi. Kuten muissa viitekehysmalleissa Zachmanin malli ei tarjoa suoraan ratkaisua kuinka tietyn yrityksen kokonaisarkkitehtuuri tulisi rakentaa. Se ei myöskään kerro millaisia prosesseja kokonaisarkkitehtuurityö sisältää ja kuinka eri osa-alueet tulee kuvata. (SOLEA-hanke 2017)

Muihin tarjolla oleviin viitekehyksiin verrattuna Zachmanin viitekehys on sen loogisuuden vuoksi sitä ensi kertaa käyttävälle yksinkertainen ymmärtää. Muut viitekehykset ovat sisällöltään paljon teknisempiä Zachmanin viitekehukseen verrattuna. Zachmanin viitekehys antaa kuvan koko organisaatiosta ja tarjoaa kuvauksia loogisesti useasta eri näkökulmasta. (SOLEA-hanke 2017)

Zachmanin viitekehysten ohella voidaan myös hyödyntää useita eri metodeja, työkaluja ja notaatioita, jonka mahdollistaa viitekehysten neutraalisuus. Viitekehys parantaa kommunikaatiota eri sidosryhmien välillä ja lisää tietoa eri riskeistä, joita liittyy arkkitehtuurisen työn tekoon. Viitekehysten avulla organisaatio voi myös uudelleen miettiä organisaation tapoja tehdä palveluita ja sovelluksia sekä mahdollisuuksien mukaan kehittää eteenpäin näitä tapoja.

Kuvassa 6 on viimeisin versio V3.0 Zachmanin viitekehysmallista, joka on julkaistu vuonna 2011.

The Zachman Framework for Enterprise Architecture™

The Enterprise Ontology™



Kuva 6. Zachman viitekehysmalli kokonaisarkkitehtuurille (Zachman 2017)

3.3 Julkisen hallinnon KA-kypsyystasomalli

Julkisen hallinnon Kokonaisarkkitehtuuri-kypsyystasomalli on valtiovarainministeriön käyttämä malli, joka pohjautuu kypsyystasomalli CMM (Capability Maturity Model):iin. Kypsyystasomalli kuuluu osana julkisen hallinnon kokonaisarkkitehtuurin kuvaukseen (kuva 7) ja sitä hyödynnetään viitekehystenä koko julkisen hallinnon arkkitehtuurikyvykkyuden nykytilan arvioimiseen sekä kokonaisarkkitehtuuroiminnan kehittämisen suunnitteluun. (Julkinen hallinto 2017)

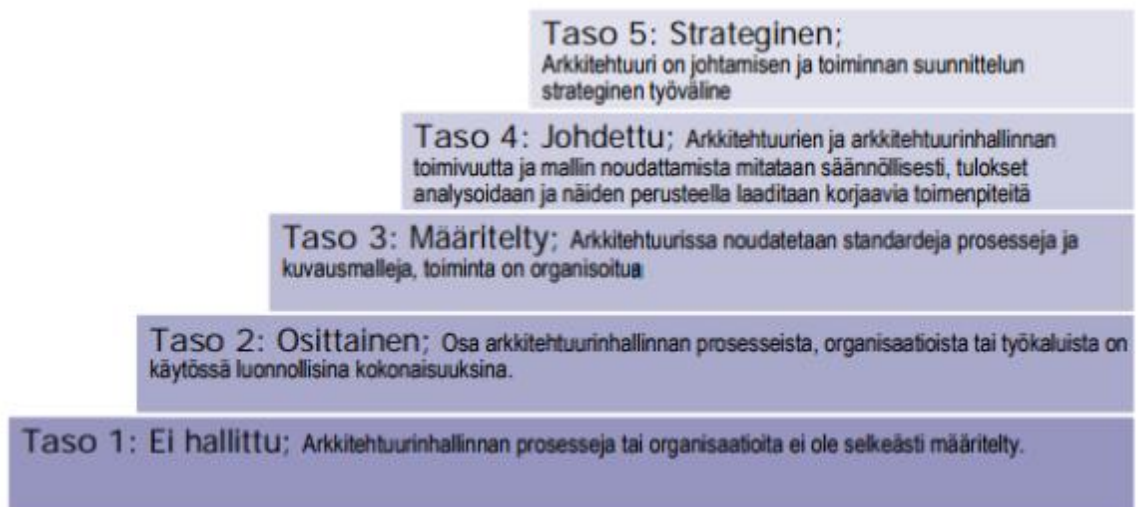
Arkkitehtuurikyvykkyuden kypsyystasomallin perimmäisenä tarkoituksena on ohjata julkiseen hallintoon kuuluvia organisaatioita sekä julkisen hallinnon arkkitehtuurityöhön osallistuvia itsenäiseen arvioon oman organisaationsa arkkitehtuurityön tasosta sekä kyvykkydestä. (Julkinen hallinto 2017)



Kuva 7. Julkisen hallinnon kokonaisarkkitehtuurin osa-alueet (Julkinen hallinto 2017)

3.3.1 Kypsyystasomallin portaat

CMM kypsyystasomalli jäsentää yrityksen toiminnan ja prosessien kypsyttä viiteen eri kypsyystasoportaaseen (kuva 8). Kypsyystasoportaiden tarkoituksena on ollut auttaa yritystä helposti käynnistämään hyvin rajatun arkkitehtuurityön kehittämisprojektin. Tämän mahdollistaa eri askelmien konkreettisuus. Portaita pitkin nouseminen aina ylimmälle eli viidennelle portaalle asti on tarkoitus olla motivoivaa, koska jopa pienet parannukset yrityksen arkkitehtuurisissa edistysaskeleissa näkyvät heti kypsyysarvosanoissa. Toisin sanoen kokonaisarkkitehtuuritoiminnan kypsyys ja kyvykkyys kasvaa tasojen mukaan. (Julkinen hallinto 2017)



Kuva 8. Kypsyystasomallin portaat (Julkinen hallinto 2017)

Organisaation arkkitehtuurin kypsyystason ollessa tasolla yksi, sen eri arkkitehtuuriin liittyviä menetelmiä tai prosesseja ei ole selkeästi määritelty. Organisaatiossa on tasolla yksi harvoin omaa arkkitehtuuriimiä ja orgisaation arkkitehtuurista sekä sen hallinasta saattaa vastata yksi tai muutama henkilö. Julkisen hallinnon ohjeiden mukaan melkein kaikki organisaatiot yltyvät vähintään tasoon yksi, koska tasolla nolla arkkitehtuurin tarvetta ei ole joko tiedostettu tai sen mahdollisuuksia ei tunneta riittävän hyvin. Toisin sanoen tasolla yksi organisaatio tunnistaa kokonaisarkkitehtuurin tarpeen ja tietää sen mukana tuomat hyödyt. (Julkinen hallinto 2017)

Kypsyystasolla kaksi organisaatiolla on osittain käytössä arkkitehtuurihallinnan prosesseja ja arkkitehtuurimenetelmiä on käytössä organisaatiolaajuisesti. Kehittäminen on kattaa monta organisaation osa-aluetta ja on osin tuloksellista. Toisaalta tasolla kaksi organisaatiolla ei vielä ole kattavaa prosessinäkemyistä tai toimintamallia arkkitehtuurille. Arkkitehtuurin kuvaustapa saattaa olla olemassa, mutta se saattaa huomattavasti poiketa standardeista. Arkkitehtuuria varten on nimetty organisaation tietohallinnosta vastuuhenkilö, jonka vastuulla on arkkitehtuurin kehittäminen osana organisaation yleisiä suunnitteluprosesseja. Kehittäminen ei ole kuitenkaan vielä jatkuvaa vain sattumanvaraista. (Julkinen hallinto 2017)

Kypsyystasolla kolme organisaation arkkitehtuurin kehityksessä noudatetaan standardoituja prosesseja ja kuvausmalleja. Organisaatio on myöskin kuvannut kattavasti arkkitehtuurimenetelmän ja sitä koskevat hallintamenettely. Tasolla kolme arkkitehtuurimenetelmä muodostaa jo kattavan kokonaisarkkitehtuurin, jossa huomioidaan toiminnan, tiedon, järjestelmien ja teknologian kokonaisarkkitehtuurin näkökulmat. Arkkitehtuurityö on myöskin suunnitelmallista ja pystyy tuottamaan nyky- ja tavoitetila-arkkitehtuureja, jotka ohjaavat tietojärjestelmien toimintaa organisaatiossa yhdessä sovittuun suuntaan. Arkkitehtuurin kehittämisestä vastaavat henkilöt ovat taas saaneet arkkitehtuurikoulutusta ulkopuolisilta kursseilta sertifikaattien muodossa. (Julkinen hallinto 2017)

Kypsyystasolla neljä organisaatiolla on pitkälle kehitetty arkkitehtuuri ja arkkitehtuurinhallinta, joita mitaillaan säännöllisesti. Arkkitehtuurityön tuloksellisuus voidaan varmistaa kontrolliprosesseilla sekä myös tarkkailla, että yhdessä sovittuja menetelmiä ja linjauksia noudatetaan. Mittaustuloksia analysoidaan ja tarkkaillaan sekä tarpeen tullen reagoidaan tehden arkkitehtuurimalleihin tarvittavia toimenpiteitä. Jatkuva kehittäminen on kypsyystason neljä keskiössä ja toimivuuden sekä yhteisesti sovittujen menetelmien mittaaminen korostuu. Arkkitehtuurikuvausten yhteensopivuutta tarkkaillaan

myös säännöllisesti. Arkkitehtuurimallien jatkokehitys perustuu mittaustuloksista saatuihin tuloksiin ja näitä seuraaviin jatkotoimenpiteisiin.

(Julkinen hallinto 2017)

Kypsyystasolla viisi eli ylimmällä arkkitehtuurikyvykkyyden tasolla organisaatiossa arkkitehtuuri on johtamisen ja toiminnan suunnittelun strateginen työväline. Arkkitehtuurihallintaprosesseja kehitetään johdonmukaisesti parhaiden käytäntöjen mukaan ja arkkitehtuurin kehittämäinen ja jalkauttaminen ovat osa toiminnan kokonaiskehittämistä. Arkkitehtuurin kehittäminen on osa organisaation johtamis- ja strategia prosesseja sekä organisaation arkkitehtuurimittareiden mittaustuloksia hyödynnetään jatkuvasti julkisen hallinnon arkkitehtuurin kehittämisessä. Tasolla viisi oleva organisaatio toimii esimerkkinä koko julkisen hallinnon kokonaisarkkitehtuuritoiminnalle. (Julkinen hallinto 2017)

3.3.2 Kypsyystasomallin tavoitteet ja tarkoitus

CMM:n kypsyystasomallilla on monia eri organisaatioita hyödyttäviä tavoitteita ja tarkoituksia. Sen päätehtävä on keskittyä organisaation kokonaisarkkitehtuurikyvykkyyden arviointiin ja kehitykseen. Kokonaisarkkitehtuurikyvykkyyden kehityksen malli mahdollistaa kyvykkyyden kehittämisen selkeästi luokittelemalla tavoitteet. Näin arkkitehtuurityön edistyminen ja sen seuranta on organisaatiolle helpompaa. Kypsyystasomalli tarjoaa myös arkkitehtuurikyvykkyyden kehittämiseen ja nykytilan arvioimiseen tarkoitettuja mittareita, joiden mittaustuloksista voidaan nähdä mahdollisia kehittämisen kohteita arkkitehtuurissa. Mittauksia voidaan suorittaa sekä yrityksen sisäisesti, mutta ne on myös mahdollista teettää halutessa ulkopuolisella taholla. (Julkinen hallinto 2017)

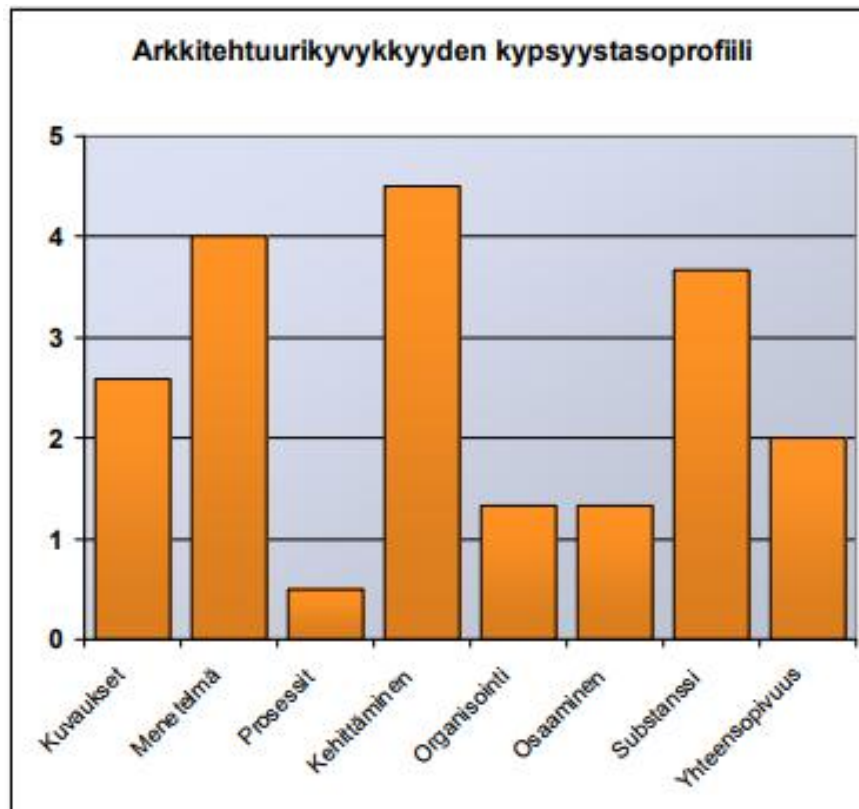
JHS:n arkkitehtuurikyvykkyyden kypsyystasomalli on tarkoitettu yhteensopivaksi julkisessa hallinnossa käytettyihin kokonaisarkkitehtuurin kyvykkyyden arviointimalleihin. Tämän mahdollistaa kypsyystasomallin pohjautuminen aikaisemmin toteutettuihin kuntasektorin sekä valtiohallinnon arkkitehtuurikyvykkyyden kypsyystasomalleihin. Kaikki edellä mainittujen mallit perustuvat yleiseen Yhdysvalloissa Carnegie Mellon yliopistossa kehitettyyn CMM-kypsyystasomalliin. (Julkinen hallinto 2017)

3.3.3 Kypsyystasomallin käyttö

Arkkitehtuurikyvykkyyden kypsyystasomallin yhtenä päätarkoituksena on auttaa organisaatioita arvioimaan itsenäisesti oman arkkitehtuurin ja sen hallinnan tasoa. Kypsyystaso auttaa organisaatioita myös hahmottamaan arkkitehtuurinsa nykytilan sekä

nykytilan pohjalta toteuttaa kehittämissuunnitelmia sekä arvioimaan kehitystä arkkitehtuurin alueella. (Julkinen hallinto 2017)

Yksittäisen organisaation arkkitehtuurikyvykkyyden kypsyystasolle voidaan luoda osa-aluekohtainen profiili (kuva 9), jossa jokaiselle osa-alueelle annetaan arvosana. Kuvassa 9 olevassa kypsyystasoprofiilissa on esimerkki kypsyystasoprofiilista sekä siinä näkyvistä eri arvosanoista. (Julkinen hallinto 2017)



Kuva 9. Kypsyystasoprofiili (Julkinen hallinto 2017)

4 Enterprise Architect

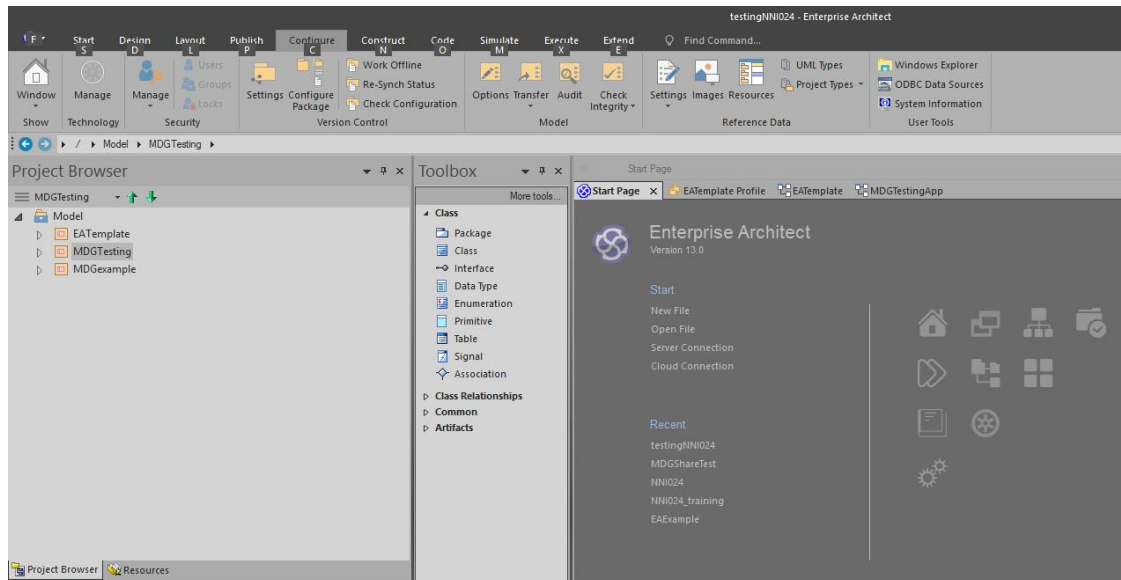
Enterprise Architect on UML-pohjainen suunnitteluun ja analyysiin tarkoitettu työkalu, joka tukee mm. UML, BPMN sekä Archimate mallinnuskieliä. Enterprise Architectin uusimman version on 13.5, joka mm. lisäsi tuen ohjelmiston pilvipohjaisille ratkaisuille. (SPARX Systems 2017)

4.1 Enterprise Architect työkaluna

Enterprise Architectiin on suunniteltu monia eri näkökulmia arkkitehtuurityöhön, kuin myös projektin hallintaan tarkoitettuja ominaisuuksia. Ohjelmistolla voi luoda tuotteiden elinkaarimalleja liiketoiminta ja IT-järjestelmistä ja simuloida eri prosesseja ja käyttäytymismalleja, jotta käyttäjä saa hyvän kuvan miten jokin yrityksen liiketoiminnan kannalta tärkeä järjestelmä toimii. Ohjelmistossa on myös sisäänrakennettu editori lähdekoodin muokkaamista varten, jonka avulla käyttäjä voi muokata mallien takana olevaa lähdekoodia. Tuettuihin ohjelmistokieliin kuuluvat mm. Java, C#, C++, Python, Visual Basic sekä PHP. (SPARX Products 2017)

Ohjelmistolla voi myös mallintaa ja hallita monimutkaisia tietueita sekä generoida HTML-pohjaisia dokumentteja ja raportteja, jotka yritys voi halutessaan julkaista Intranet sivuillaan tai dokumentinhallintajärjestelmässä. Myös eri työvaiheiden automatisointi on erittäin vahvasti esillä ja käyttäjä voi automatisoida tiettyjä työvaiheita, kuten HTML-raportin luonnin aina työpäivän päätteeksi. Automatisointi toimii käytännössä ohjelmoimalla jollakin edellä mainituista kielistä. Enterprise Architectissa löytyy myös tuki relaatiotietokannoille, joiden mallinnus on ohjelmistossa mahdollista. Eri tuettuja tietokantoja ovat mm. MySQL sekä Oracle Database. (SPARX Products 2017)

Kuvassa 10 on ruutukaappaus Enterprise Architect 13 käyttöliittymästä, jossa esillä on lokaalisti tallennetut kansiot.



Kuva 10. Enterprise Architect käyttöliittymä

Enterprise Architectista on saatavilla kuusi eri versiota aina Desktopista, joka on pelkistetympi versio ohjelmistosta, Ultimateen, joka sisältää kaikki tarjolla olevat ominaisuudet. Kaikki muut versiot paitsi Desktop sisältävät mahdollisuuden luoda omia työkalupakkeja MDG Teknologian avulla, jota käsitellään seuraavassa luvussa.

4.2 MDG Teknologia

Enterprise Architect ohjelmistoon sisällytetyn MDG Teknologia välineen avulla käyttäjä voi luoda, muokata ja hallita työkalulaatikossa olevia malleja, jotka pohjautuvat UMLään tai myös vaihtoehtoisesti sisällyttää työkalulaatikkoon Non-UML malleja, kuten BPMN tai Archimate. (SPARX 2017, 5.)

Käyttäjä voi halutessaan MDG Teknologian avulla laajentaa Enterprise Architectin ominaisuuksia, kuten valmiita malleja ja diagrammeja sekä elementejä, jotka sopivat paremmin loppukäyttäjän tarpeisiin. MDG Teknologia suunniteltiin alusta asti uusiksi Enterprise Architectin versiossa 10, missä teknologiaan lisättiin uusia ominaisuuksia ja graafisen käyttöliittymän ulkoasua paranneltiin. Käytin tässä opinnäytetyössä hyödykseni Enterprise Architectin versiota 13. Versioon 10 verrattuna muutokset ovat lähinnä graafisia, eikä uusia toiminnallisuuksia ole tullut merkittävästi lisää. (TigerTeam 2017a)

Kokemusteni perusteella MDG Teknologian käyttö on kuin rakentaisi korttitaloa: yksikin puuttuva osa rakenteissa takaa sen, että mikään ei toimi.

5 Projekti

Wärtsilän minulle antaman tehtävän tarkoituksena oli luoda heidän käyttämäänsä SPARX Systemsin kehittämään Enterprise Architect työkaluun valmiita templateja, jotka valitsemalla käyttäjä saisi käyttöönsä vain tarvitsemansa työkalut tietyn prosessin luomiseen ja hallintaan.

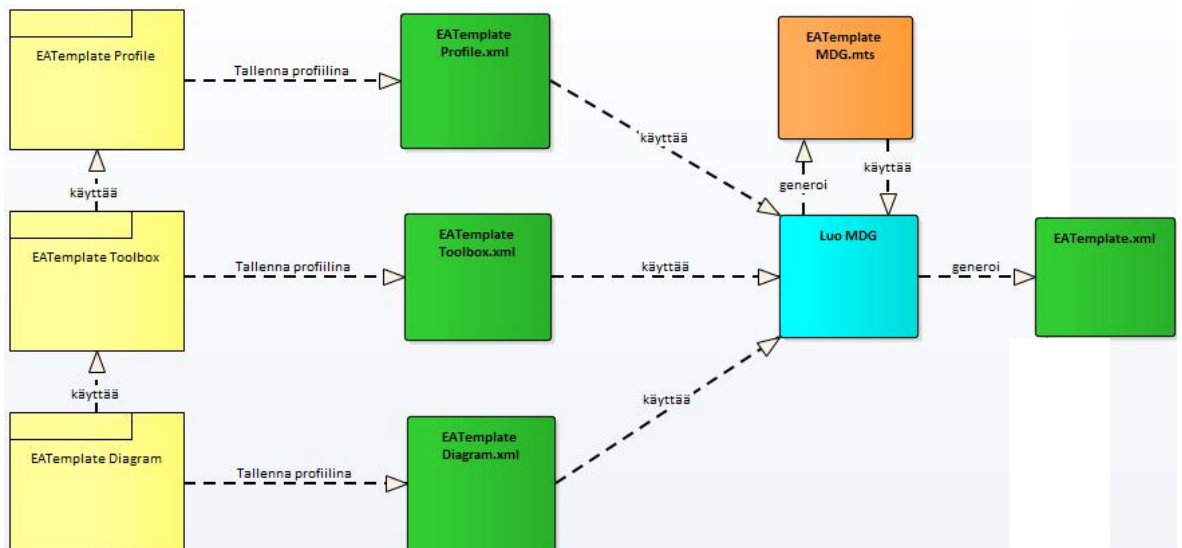
Ongelmana tähän asti oli ollut se, että kun uutta diagrammia alettiin mallintaa, saatavilla olevien elementtien ja niitä yhdistävien connectorien määrä oli erittäin laaja. Tämän vuoksi käyttäjillä oli hankaluuksia valita oikeat elementit oikeantyyppiseen diagrammiin mikä taas johti siihen, että vaikka kaksi diagrammia oli tietotyypiltään samanlaiset niissä käytettiin hyödyksi aivan eri elementtejä. Näin saatiin idea luoda oma työkalulaatikko, missä olisi vain rajalliset työkalut aina tietyn diagrammin mallintamiseen. Näin diagrammeista saadaan yhtenäisen näköiset, jotka sopivat ennalta sovittuihin ratkaisuihin, kuin myös Wärtsilän teemaan väreiltään ja ulkoasultaan. Yhtenäinen teema eri elementtien välillä myös lisää niistä rakentuvien diagrammien ymmärrettävyyttä loppukäyttäjän kannalta.

Hetken mietittyäni tehtävänantoa ja etsittyäni tietoa SPARXin Internet-sivuilta päädyin käyttämään hyödykseni Enterprise Architect työkalun sisään rakennettua MDG Teknologiaa. MDG on ainut tapa Enterprise Architectissa, jonka avulla käyttäjä voi muokata ohjelmiston tarjoamia työkalupakkeja haluamakseen riisuen turhat ja tarpeettomat elementit valikoimasta pois luoden templatien uudesta työkalupakista.

5.1 Projektin alustus

Ennen MDG Teknologian luontia tulee Enterprise Architectin kansiorakenne olla oikean tyylinen, jotta oikeat asiat löytyvät oikeasta paikasta. Jotta MDG Teknologian luominen on mahdollista, se vaatii käyttäjältä kolmea asiaa. Ensimmäiseksi tulee luoda profiili, joka pitää sisällään kaikki halutut elementit oikean tyyppisiin diagrammeihin. Toiseksi tulee luoda työkalulaatikko, johon kaikki diagrammeihin tulevat elementit ovat järjestettynä, jotta ne on helppo paikantaa myöhemmin. Kolmanneksi täytyy luoda diagrammi, johon työkalulaatikko on yhdistettynä, jotta profiilissa mainitut elementit voidaan mallintaa MDG Teknologiaan sopivaksi. Tämän jälkeen teknologia luodaan taas oman työkalunsa kautta yhdistäen nuo kolme äsken mainittua vaihetta keskenään. (TigerTeam 2017a)

Kuvassa 11 on mallinnettuna graafiseen muotoon miltä valmiin MDG Teknologian arkkitehtuuri näyttää ja miten eri tiedostot ovat suhteessa toisiinsa. Arkkitehtuuri perustuu Tanskalaisen IT-yrityksen TigerTeamin arkkitehtuurivisioon valmiista MDG Teknologian mallista, jonka perusteella kehitin myös oman mallini. (TigerTeam 2017a)

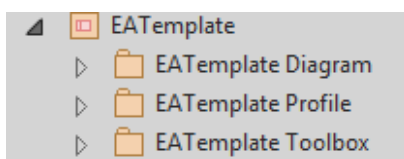


Kuva 11. Eri tiedostojen yhteydet toisiinsa valmiissa mallissa

5.2 Projektin toteutus

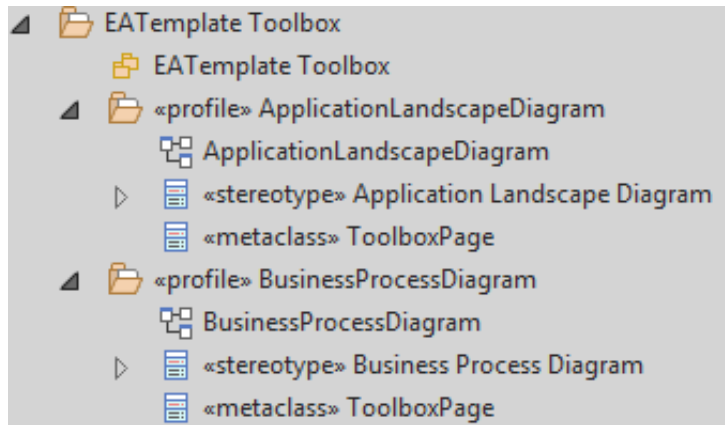
Aloitin projektin luomalla Enterprise Architect työkaluun uuden projektin, joka pitäisi sisällään kaikki tarvittavat tiedostot ja kansiot. Loin uuden projektin lokaalisti oman tietokoneeni kovalevylle enkä yrityksen tuotanto- tai testauskantaan, koska muuten luomani työkalut olisivat olleet välittömästi muiden käyttäjien käytettävissä. Tätä en halunnut vaan säilytin projektissa olevia tiedostoja ja kansioita lokaalisti omalla tietokoneella ennen kuin pistin ne jakoon muille tiimiläisille. Tiedostojen jaosta muille käyttäjille kerrotaan enemmän projektin luvussa 5.2.7.

MDG Teknologia tarvitsee toimiakseen kolme kansiota; Profile, Toolbox ja Diagram joiden avulla luodaan toimiva teknologia. Loin siis ensin nuo kolme kansiota luomaani kansiorakenteeseen (kuva 12).



Kuva 12. Kansiorakenne

Kansiorakenne luotiin pitämään automaattisesti sisällä pohja diagrammille, johon kunkin kansion sisältö tulisi rakentaa. Esimerkkinä toimii Toolbox-kansion alla oleva rakenne (kuva 13), josta on näkyvillä kaksi diagrammiin perustuvaa työkalulaatikkoa, Application Landscape Diagram ja Business Process Diagram. Työkalulaatikko tarkoittaa tässä tapauksessa Toolbox-kansioon tehtyä kansiota, jonka alla on diagrammi.



Kuva 13. Toolbox-kansiorakenne

5.2.1 EATemplate Profile-kansio

Kolmen kansion eli Profile, Toolbox ja Diagram luonnin jälkeen aloitetaan luomaan itse MDG Teknologiaa luomalla oma profiili EATemplate Profile-kansioon (kuva 14).



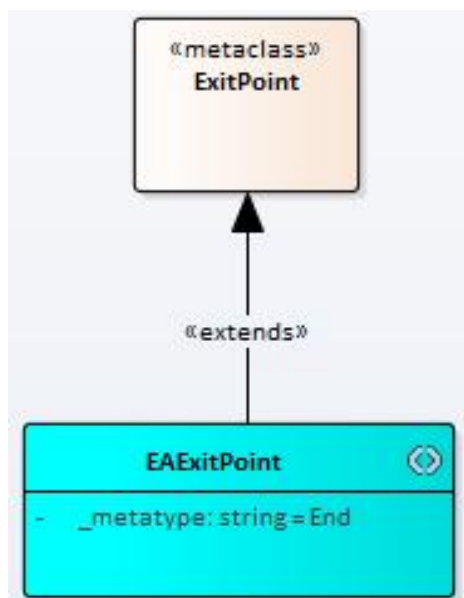
Kuva 14. Profiili

Profiilille on suotavaa antaa selkeä ja helposti muistettava nimi, koska nimeä tarvitaan myös projektin myöhemmissä vaiheissa. Päätin antaa profiilin nimeksi EATemplate, koska nimi on helposti muistettava ja kuvaa hyvin mitä olen projektissa tavoittelemassa.

Profiilin luonnin jälkeen luotiin profiiliin kaikista alimman kansion alla olevaan EATemplate diagrammiin sisältöä. Kyseinen diagrammi on nähtävissä kuvassa 14 olevasta kansiorakenteesta. Diagrammiin luotiin profiilin kannalta kriittisiä elementtejä sekä näitä yhdistäviä connectoreita. Profiilin alla oleva diagrammia voidaan pitää koko kansiorakenteen sydämenä, johon kerätään kaikki eri työkalupakeissa käytettävät elementit ja elementit yhdistävät connectorit.

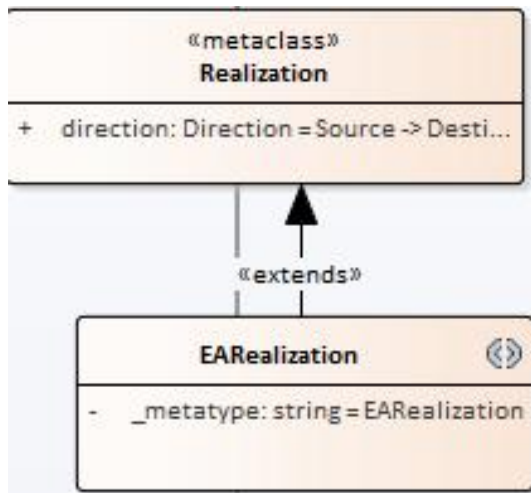
MDG Teknologiassa elementti koostuu kahdesta osasta: metaluokasta ja stereotyypistä. Metaluokaksi valitaan haluttu elementti tai connector. Tämän jälkeen luodaan metaluokalle pari stereotyyppiä, joka laajennetaan metaluokkaan connectorin avulla. Esimerkkinä kuvassa 15 on mallinnettu metaluokka nimeltä ExitPoint, joka on elementtiin perustuva metaluokka.

Metaluokan alapuolelle on luotu stereotyyppi nimeltä EAExitPoint, jolle on annettu yksi attribuutti nimeltä `_metatype`. Syy miksi stereotyyppejä tarvitaan on se, että jos halutaan luoda esimerkiksi useita ExitPoint elementtejä joista jokainen on ulkoasultaan erilainen voidaan luoda vain yksi metaluokka ExitPoint ja sen alle niin paljon eri stereotyyppejä kuin eri variaatioita ExitPoint elementistä tulee olemaan.



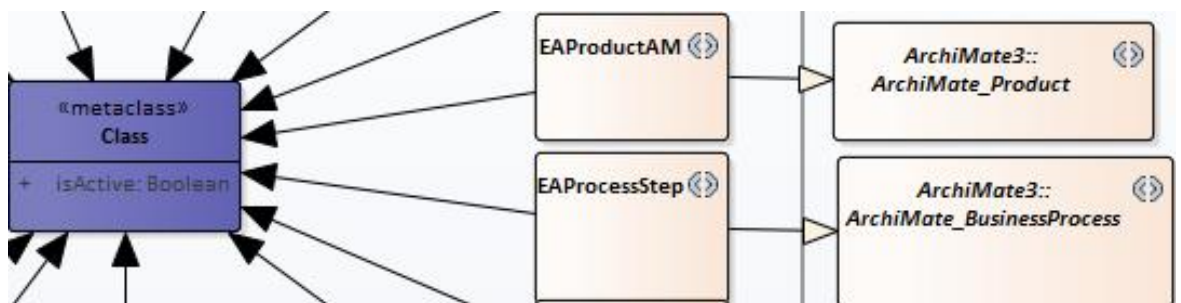
Kuva 15. Elementin metaluokka ja stereotyyppi

Toisena esimerkkinä kuvassa 16 on connector nimeltään Realization eli realisointi. Connectorin mallintaminen on samanlaista elementin mallintamiseen, mutta connectorin attribuutteihin tulee joitakin muutoksia elementteihin verrattuna. Realization metaluokkaan tulee connectoreille attribuutiksi aina connectorin suunta eli kulkeeko connector lähteestä kohteeseen vai kohteesta lähteeseen. Tässä tapauksessa connector on laitettu automaattisesti kulkemaan lähteestä kohteeseen.



Kuva 16. Connectorin metaluokka ja stereotyyppi

Kaikki Enterprise Architect ohjelmistossa ovat metaluokat ovat tyypiltään UML:ää. Joskus kuitenkin saattaa olla tarvetta eri työkaluille, kuten ArchiMateen tai BPMN:ään perustuvilla työkaluille ja connectoreille. Tämän tyyppisiä työkaluja saatetaan kutsua Enterprise Architect ohjelmiston eri käyttöohjeissa non-UML työkaluiksi. Kuvassa 17 on kuvattuna miten non-UML elementti tulee mallintaa.



Kuva 17. non-UML elementin mallinnus

Kuvassa on esimerkkinä kaksi non-UML elementtiä: EAProductAM ja EAProcessStep. Miten non-UML elementin mallinnus käytännössä toteutetaan on luomalla ensin elementistä riippumatta metaluokka Class, joka toimii non-UML elementtien sydämenä. Metaluokkaan lisätään yksi attribuutti nimeltään isActive, jonka tietotyyppiksi valitaan Boolean. Boolean on yksi Enterprise Architectiin sisään rakennetuista tietotyypeistä, joita voi käyttää hyödyksi attribuuteissa. Uusia attribuutteja on mahdollista luoda myös itse, jos sisäänrakennetuista attribuuteista ei löydy elementtiin sopivaa attribuuttia. Yksi lisäämistäni attribuuteista oli String, jonka tyyppistä attribuuttia ei jostain syystä löytynyt ollenkaan Enterprise Architectistä. Attribuutin valitsemisen jälkeen luodaan normaaliin tapaan stereotyyppi, jonka nimi kuvastaa sitä elementtiä jota ollaan luomassa ja laajennetaan stereotyyppi metaluokkaan connectorin avulla. Tämän jälkeen luodaan lisäksi stereotyyppi siitä teknologiasta, jota halutaan käyttää hyväksi. Tässä tapauksessa

käyttämämme teknologia on ArchiMate 3. Stereotyypin nimeksi tulee määritellä teknologia sekä mitä elementtiä siitä teknologista halutaan käyttää.

Ylläolevassa kuvassa on molemmissa elementissä haluttu käyttää Archimate 3 teknologiaa, joista toisessa on käytössä Archimaten Product elementti ja toisessa Archimaten BusinessProcess elementti. Stereotyyppien luomisen jälkeen ensimmäiseksi luodut stereotyypit pitää generalisoida Archimate stereotyyppeihin connectorien avulla. Generalisoidun connectorin voi ylläolevassa kuvassa havaita connectorin valkoisesta nuolesta, kun taas laajennuksessa käytetyssä connectorissa käytetään hyväksi mustaa nuolta.

Profiili tallennetaan kuvassa 14 näkyvästä <<profile>> EATemplate kansion kohdalta valitsemalla vaihtoehto Save Package as UML Profile. Tämän valitsemalla Enterprise Architect luo profiilista XML-tiedoston, jota voidaan käyttää joko itsenäisesti tai osana MDG Teknologiaa. Tulen käyttämään tuota tiedostoa osana MDG Teknologiaa. Kaikki luodut XML tiedostot on hyvä tallentaa samaan kansioon, koska myöhemmin itse MDG Teknologiaa luodessa Enterprise Architect tarkistaa teknologiaa varten luodut XML-tiedostot vain yhdestä kansioista.

Liitteissä 1-3 on mallinnettuna Profile-kansioon kuuluvat metaluokat ja stereotyypit.

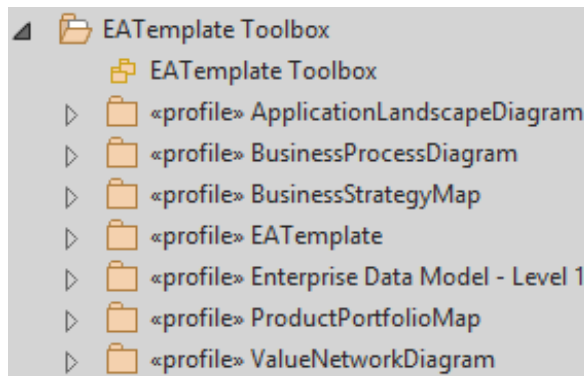
5.2.2 EATemplate Toolbox-kansio

Profile-kansion luonnin jälkeen on hyvä lähteä luomaan toolbox-kansiota. Toolbox-kansion tarkoitus on järjestää profiilissa olevat elementit siten, että ne on helppo löytää ja lisätä ne myöhemmin diagrammiin. (TigerTeam 2017b)

Projektin alussa luotiin toolbox-kansio ja sen alle automaattisesti diagrammi. Seuraavat muokkaukset tehdään diagrammiin hyvin samalla periaatteella mitä profile-kansiossa olevaan diagrammiin tehtiin. Ensin siis luodaan kuvassa 18 näkyvän EATemplate Toolbox-diagrammin alle profiili.

Toolbox-kansiossa olevia profiileja täytyy olla niin monta, kuin MDG Teknologiassa halutaan olevan eri työkalulaatikoita. Profiilien alle taas täytyy rakentaa diagrammi, johon kootaan halutut elementit ja connectorit tiettyyn työkalulaatikkoon. Kuvassa 18 on luotuna seitsemän eri profiilia, joista kaikki paitsi EATemplate perustuu joko diagrammiin, mallinnuskarttaan tai datamalliin. EATemplateen on koottu kaikissa muissa profiileissa

käytetyt elementit ja connectorit yhteen paikkaan, jotta niiden testaaminen yhdessä olisi mahdollisimman yksinkertaista.



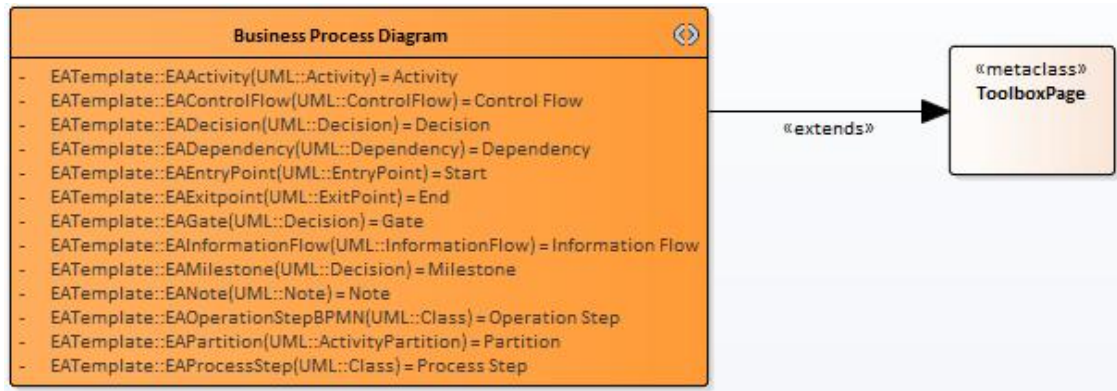
Kuva 18. Toolbox-kansion eri profiilit

Luonti aloitetaan samalla periaatteella mitä profiileja luotaessa tehdään eli viemällä diagrammeihin metaluokkaelementti sekä stereotyyppi. Näiden välille luodaan connectorin avulla laajennus stereotyypistä metaluokkaan. Metaluokkaan ei profiiliin tavoin tule mitään attribuutteja vaan kaikki attribuutit tulevat stereotyyppiin. Stereotyypissä olevat attribuutit ovat työkalulaatikon toimimisen kannalta erittäin tärkeitä ja jos ne on konfiguroitu väärin tai niitä ei ole, työkalulaatikko ei yksinkertaisesti toimi.

Kuvassa 19 on esimerkkinä Toolbox-kansion BusinessProcessDiagram profiilin diagrammiin luotu stereotyyppi ja metaluokka, josta näkyy Business Process Diagram stereotyypin eri attribuutit.

Attribuuteille on erittäin tarkat säännöt miten ne tulee asetella stereotyyppiin. Syntaksi attribuuteille on:

<ItseLuotuProfiili::ItseLuotuProfiilielementti(SisäänrakennettuProfiili::SisäänrakennettuElementti) = TyökalulaatukossaNäkyväNimi>



Kuva 19. BusinessProcessDiagram profiili Toolbox-kansiossa

Kuten kuvasta 19 voi nähdä, on attribuuteille tarkoitettua syntaksia täytynyt noudattaa erittäin huolellisesti. Otetaan esimerkiksi Business Process Diagram stereotyypin ylin attribuutti. Ensinnäkin luodaan itse profiili eli EATemplate. Tämän jälkeen luodaan itse profiilielementti, jonka täytyy olla samanniminen Profile-kansion diagrammissa sijaitsevan tietyn stereotyypin kanssa. Seuraavaksi lisätään jokin Enterprise Architectin sisäänrakennetuista profiileista eli tässä tapauksessa UML ja tämän jälkeen lisätään jokin UML:n sisäänrakennettu elementti mihin halutaan itse luodun elementin perustuvan. Tämän jälkeen vielä nimetään luotu elementti sen nimiseksi millä se halutaan näkyvän työkalulaatikossa.

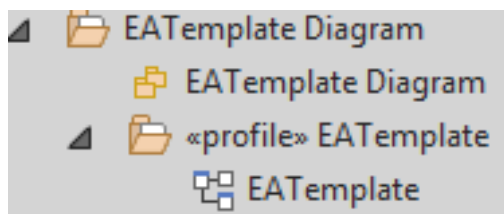
Huomioitavaa on se, että kaikkien sisäänrakennettujen profiilien täytyy olla attribuuteissa UML:ää, koska Enterprise Architect perustuu juuri tähän mallinnuskieleen. Jos halutaan luoda non-UML elementti, joka perustuu esimerkiksi Archimateen tai BPMN:n niin se täytyy merkitä eri tavalla eivän kuten Profile-kansion diagrammissa. Merkintä tehdään laittamalla sisäänrakennetuksi profiiliksi yhä UML, mutta sisäänrakennetuksi elementiksi Class oli haluttu elementti tai connector sitten minkäläinen tahansa. Se minkäläinen elementistä tai connectorista lopulta tulee täytyy määrittellä Profile-kansion diagrammissa. Hyvä esimerkki tästä on kuvassa 17 näkyvä non-UML mallinnus. Kuvassa 19 alin attribuutti on tehty non-UML mallin mukaan.

Toolboxin tallennus tehdään Profilen tallennuksesta poiketen hieman eri tavalla. Profilen tallennus tehtiin <<profile>> EATemplate kansion kohdalta valitsemalla vaihtoehto Save Package as UML Profile. Toolboxin tallennus taas täytyy tehdä kansion sijaan diagrammin kautta oikealla hiiren klikkauksella klikkaamalla sen taustaa ja valitsemalla Save as Profile. Eri tallennustavasta huolimatta Enterprise Architect luo Toolboxista samanlaisen XML tiedoston kuin se loi myös Profilesta.

Liitteissä 4-10 on mallinnettuna Toolbox-kansion diagrammeissa olevat eri diagrammityytit sekä niihin kuuluvat elementit ja connectorit.

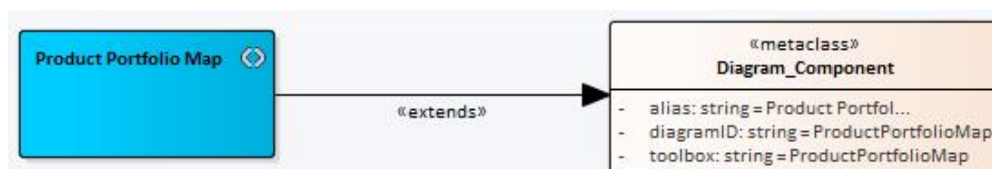
5.2.3 EATemplate Diagram-kansio

Seuraavaksi luodaan profiili EATemplate Diagram (kuva 20) diagrammiin sekä elementit ja connectorit diagram-kansioon. Kuten profile- ja toolbox-kansioissa diagrammiin täytyy luoda metaluokka ja stereotyyppi sekä connectori joka laajentaa stereotyypin metaluokkaan. Selvyyden vuoksi etäluokka ja stereotyyppi luodaan nyt siis EATemplate diagram kansiossa olevan <<profile>> EATemplate kansiossa olevaan EATemplate diagrammiin.



Kuva 20. EATemplate Diagram kansiorakenne

Toolbox-kansiosta poiketen kaikkien elementtien metaluokat ja stereotyypit tulee sijoittaa samaan diagrammiin. Myöskin kaikki attribuutit tulee nyt sijoittaa metaluokkaan eikä stereotyyppiin. Metaluokkaan tulevat attribuutit ovat vähintään yhtä tärkeitä, kuin toolbox-kansion diagrammin stereotyyppiin tulevat attribuutit työkalulaatikon toiminnan kannalta. Jokaisen metaluokan tulee sisältää kolme attribuuttia: alias, diagramID ja toolbox. Attribuutti aliaksen tietomuodoksi tulee String ja sen sisällöksi tulee nimi, joka tulee näkyviin työkalulaatikon diagrammin nimikentässä. Attribuutti diagramID:n tiedostomuodoksi tulee myös String ja sen sisällöksi tulee MDG diagrammin tyyppi, joka tulee näkymään diagrammin nimikehyksessä. Attribuutti toolboxin tiedostomuodoksi tulee myöskin String ja sen sisällöksi tulee työkalulaatikon nimi, jonka MDG automaattisesti aukaisee kun halutaan luoda uusi diagrammi. Tämä attribuutti on kaikista tärkein, koska muuten luodessa uutta diagrammia työkalulaatikko ei aukea.



Kuva 21. Product Portfolio Mapin diagramrakenne

Kuten kuvasta 21 näkyy Profile-kansio ja Toolbox-kansion diagrammit ovat suunniteltu siten, että kaikissa kolmessa attribuutissa voidaan käyttää lähes samaa nimeä.

Diagrammi tallennetaan Toolboxin tavoin diagrammista oikealla hiiren klikkauksella klikkaamalla sen taustaa ja valitsemalla Save as Profile.

Liitteissä 11-12 on mallinnettuna Diagram-kansiossa olevat diagrammit käyttäen hyödyksi metaluokkaa ja stereotyyppiä.

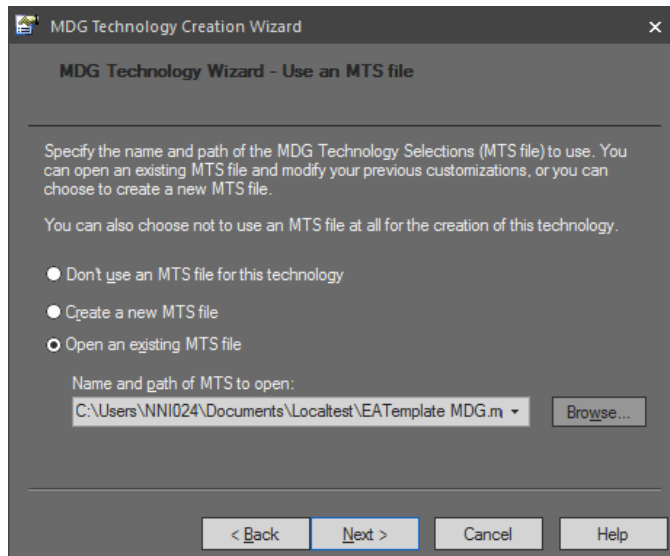
5.2.4 MDG tiedoston luonti

Sen jälkeen kun XML-tiedosto Profilesta, Toolboxista ja Diagrammista on luotu voidaan alkaa luomaan niitä kaikkia yhdistävää MDG tiedostoa. Tässä vaiheessa kannattaa kuitenkin vielä varmistaa, että kaikki luodut XML-tiedostot ovat samassa kansiossa. Tämä sen vuoksi, että MDG tiedostoa luodessa Enterprise Architect etsii XML-tiedostoja vain yhdestä kansioista.

MDG tiedostojen luontiin on Enterprise Architectissa siihen tarkoitettu työkalu MDG Generator. MDG Generator luo kaksi tiedostoa: yhden, joka on tiedostomuodoltaan .mdg ja pitää sisällään tietoa mitä sisällyttää luotuun MDG:seen ja yksi .xml tiedosto missä sijaitsee itse MDG tiedosto.

MDG Generaattorin löytää Enterprise Architectista kirjoittamalla hakuun "MDG" ja menemällä esille tulevaan "Publish" hakemistoon, josta löytyy "Generate MDG Technology" vaihtoehto.

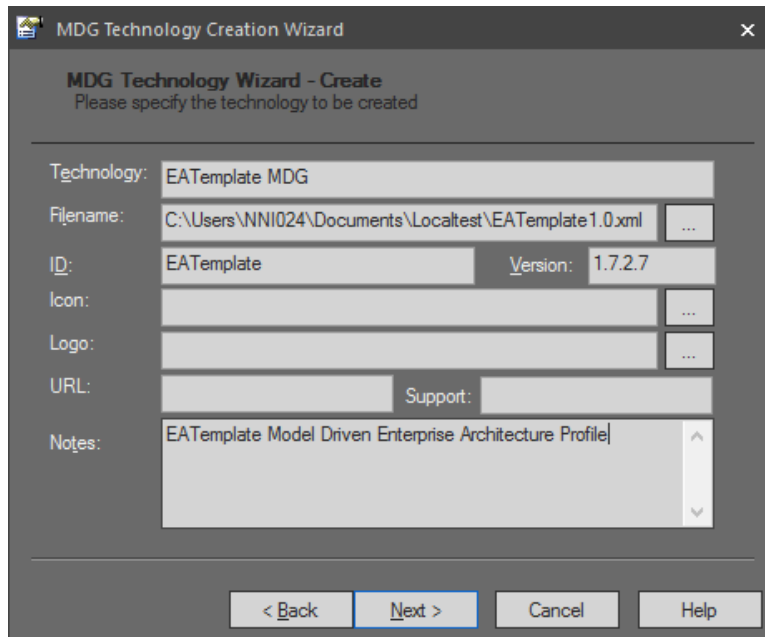
Klikkaamalla tuota vaihtoehtoa avautuu MDG:n asennusohjelma, joka auttaa ja ohjaa MDG:n luomisessa. Asennusohjelman ensimmäisessä ikkunassa kerrotaan vian yleistä tietoa MDG:stä eikä käyttäjää pyydetä syöttämään vielä mitään tietoa. Seuraavassa ikkunassa täytyy valita luodaanko kokonaan uusi MDG tiedosto vai päivitetäänkö jo olemassa olevaa tiedostoa. Olin jo luonut aikaisemmin MDG tiedoston, joten kuvassa 22 on valittuna jälkimmäinen vaihtoehto sekä tiedostopolku mistä tiedosto löytyy. Asennusohjelma käyttää hyödyksi MDG tiedostoa luotaessa MTS-tiedostoa.



Kuva 22. MDG asennusohjelma ikkuna 2

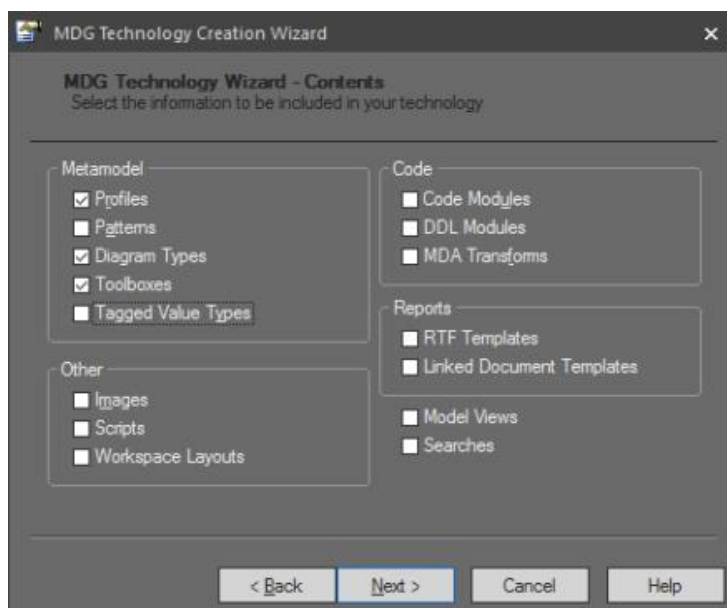
Seuraavassa ikkunassa määritellään hakemisto, johon luotu tiedosto tallennetaan. Koska olen päivittämässä jo olemassa olevaa tiedostoa, luon uuden tiedoston samaan hakemistoon kuin vanhan milloin se korvaa vanhan tiedoston automaattisesti uuteen versioon.

Neljännessä ikkunassa (kuva 23) tulee täyttää MDG tiedoston kannalta tärkeää tietoa. Ikkunassa tulee täyttää MDG Teknologian nimi, jonka voi päättää vapaasti. Filename eli tiedostopolku tulee automaattisesti esille edellisten ikkunoiden perusteella. Tärkein kenttä on ID, jonka täytyy olla täysin sama kuin profiilissa ja diagrammissa käytetty ID tai muuten mikään ei MDG Teknologiasa toimi. Versionumeroa kannattaa päivittää sitä mukaan kun MDG Teknologiaa päivittää ja lisää ominaisuuksia, jotta pysyy mukana mitä missäkin versiossa on. Icon, Logo, URL, Support ja Notes kenttiä ei ole pakollista täyttää, mutta lisäksi Notesin kohdalle lyhyen kuvauksen mitä MDG Teknologia on suunniteltu tekemään.



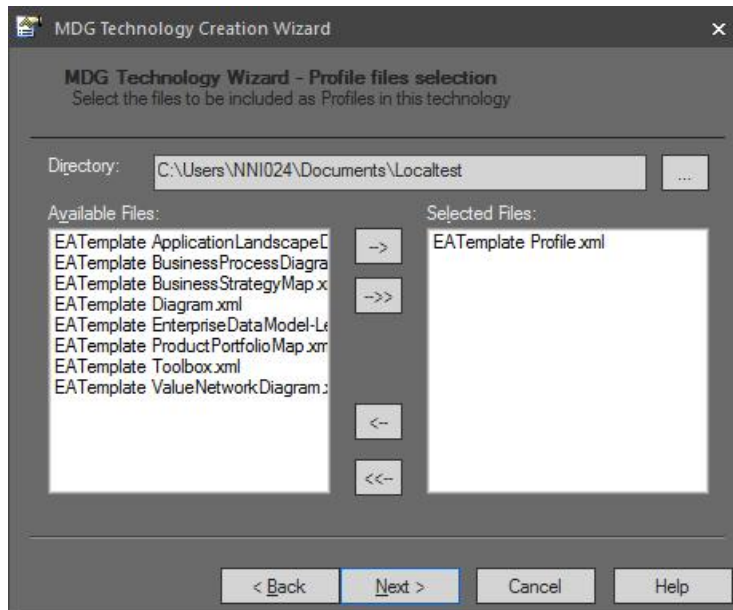
Kuva 23. MDG asennusohjelma ikkuna 4

Kuvassa 24 näkyy seuraava ikkuna, jossa määritellään mitä tietoa halutaan MDG Teknologian pitävän sisällään. Tässä tapauksessa olen valinnut sisällyttää teknologiaan vain metamallista jo luomamme profiilin, toolboxin ja diagrammin.



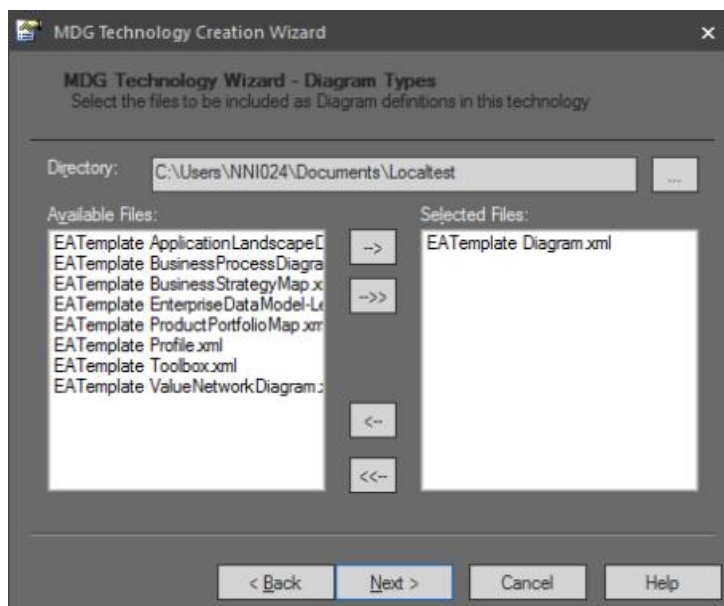
Kuva 24. MDG asennusohjelma ikkuna 5

Seuraavassa ikkunassa (kuva 25) määritellään mitä tallennettua XML-tiedostoa MDG Teknologia käyttää profiilitiedostona. Tiedostoksi valitaan se, joka luotiin ja tallennettiin Profile-kansion diagrammista.



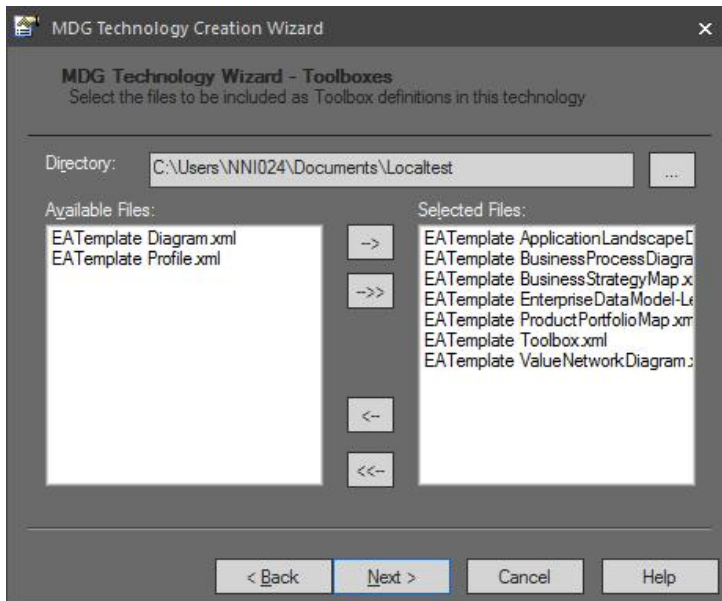
Kuva 25. MDG asennusohjelma ikkuna 6

Ikkunassa seitsemän (kuva 26) määritellään mitä diagrammia käytetään ja tiedostoksi valitaan se XML-tiedosto, joka tallennettiin Diagram-kansion diagrammista.



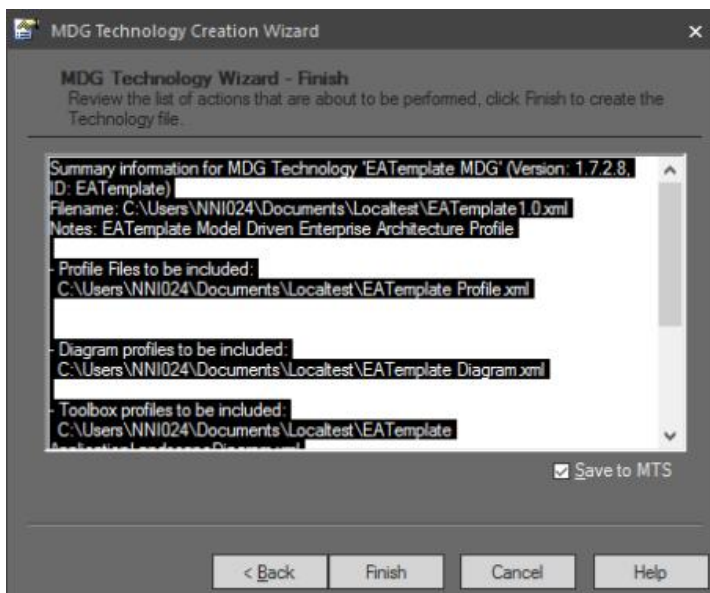
Kuva 26. MDG asennusohjelma ikkuna 7

Ikkunaan kahdeksan (kuva 27) tulee valita kaikki toolboxit, jotka tallennettiin Toolbox-kansion eri diagrammeista. Kuvassa 27 eri toolboxeja on seitsemän, joista yksi eli EA Template Toolbox.xml on kooste kaikista muista toolboxeista.



Kuva 27. MDG asennusohjelma ikkuna 8

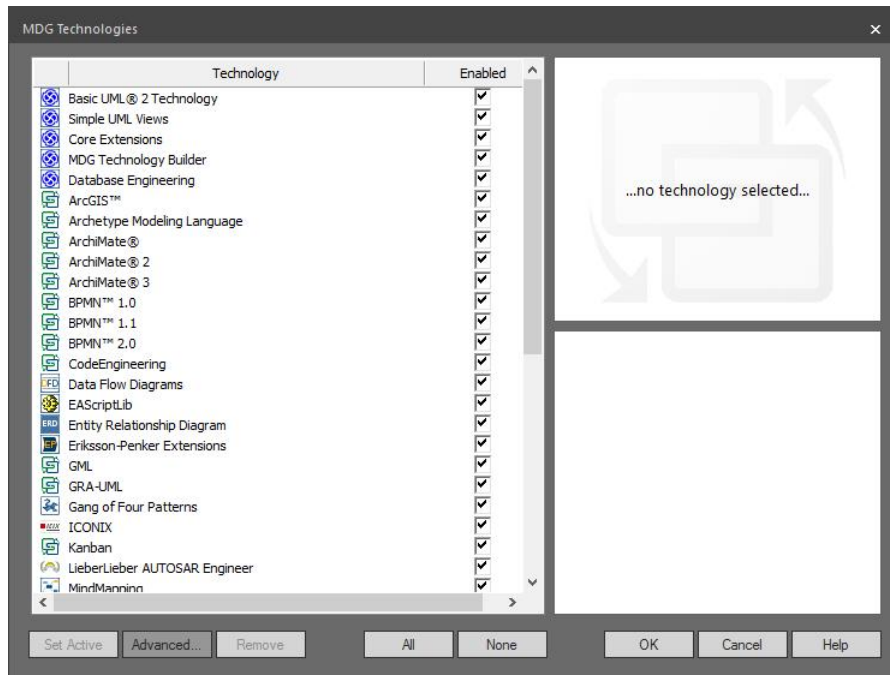
Tämän jälkeen aukeaa MDG asennusohjelman viimeinen ikkuna (kuva 28), jossa on kerrottu koosteen muodossa mitä kaikkia asioita MDG Teknologia tulee sisältämään. Kooste kannattaa lukea huolella läpi, jotta kaikki xml-tiedostot ovat oikeissa paikoissa ja tiedostopolut ovat oikein. Ikkunan oikeassa alareunassa vaihtoehto "Save to MTS" kannattaa jättää päälle tai jos se ei ole vielä päällä niin valita se. Asennus valmistuu klikkaamalla "Finish" nappia.



Kuva 28. MDG asennusohjelma ikkuna 9

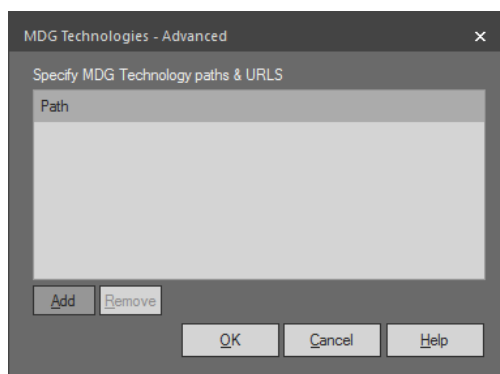
5.2.5 MDG Teknologian käyttöönotto

MDG Teknologia otetaan käyttöön Enterprise Architectin Configure-välilehden Manage osiosta. Manage osiosta avautuu uusi ikkuna, josta näkyy kaikki tällä hetkellä käytössä olevat MDG Teknologiat. Kuten kuvasta 29 näkyy niin EATemplate ei vielä ole listassa, joten se pitää lisätä sinne.



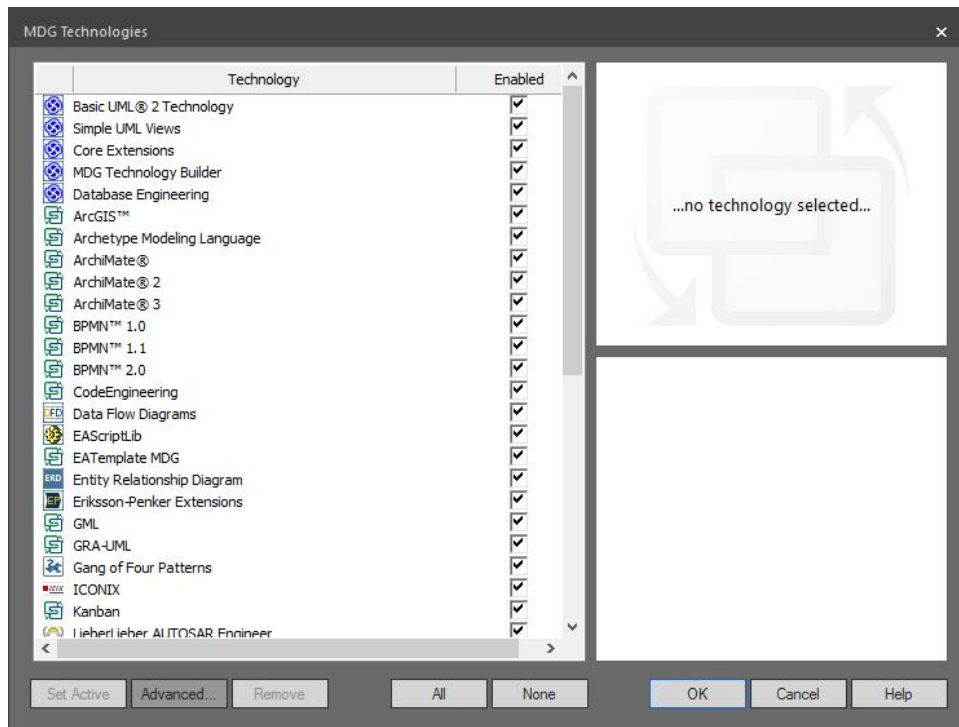
Kuva 29. Käytössä olevat MDG Teknologiat

MDG Teknologian lisäys tapahtuu Advanced valikon kautta (kuva 30), josta voi valita ottaako teknologian käyttöönsä paikallisesti tietokoneen kiintolevyllä vai etänä URL:n kautta. Kaikki luodut MDG tiedostot tallennettiin tietokoneelle paikallisesti, joten valitaan ikkunasta Add Path vaihtoehto. Riittää, että kertoo oikean kansion missä xml-tiedostot sijaitsevat ja Enterprise Architect osaa automaattisesti siirtää sen tiedon pohjalta tiedostot itse ohjelmistoon.



Kuva 30. MDG tiedostojen siirtotapa

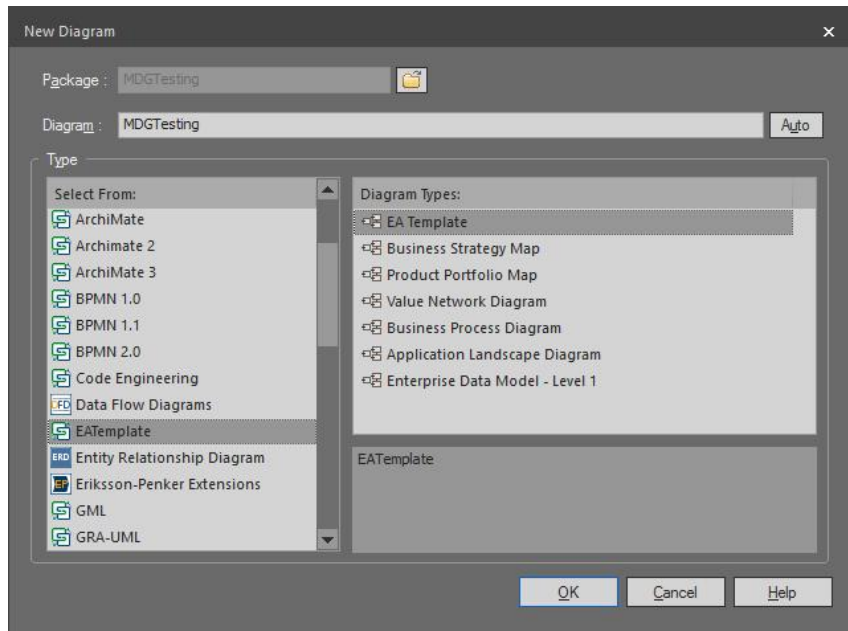
Tiedostojen siirron jälkeen luodun EATemplate MDG Teknologian tulisi olla näkyvässä muiden teknologioitten kanssa kuvassa 31 olevassa ikkunassa.



Kuva 31. Käytössä olevat MDG Teknologiat

5.2.6 MDG Teknologian testaaminen

Testasin MDG Teknologiaa luomalla kuvassa 11 nähdyn diagrammin. Loin ensin uuden kansion ja kansioon kyseisen diagrammin käyttämällä EATemplate MDG Teknologiaa ja valitsemalla diagrammiin sopivan diagrammityyppin. Kuten kuvasta 32 näkyy, EATemplate on yksi valittavista MDG Teknologioista. Valitessa EATemplaten ikkunan oikealle puolelle tulee lista eri diagrammityypeistä, jotka kuuluvat teknologiaan. Diagrammitypit ovat toisin sanoen kaikki eri työkalulaatikot, jotka luotiin Toolbox-kansiossa olevaan diagrammiin. Koska kuvan 11 diagrammi ei vastannut mitään työkalulaatikoista, päätin valita työkalulaatikoksi EA Templaten joka oli siis kooste kaikista muista työkalulaatikoista.



Kuva 32. Uuden diagrammin luonti

Valittuani työkalulaatikokseni EA Templaten aukesi minulle suoraan Enterprise Architectin käyttöliittymään kaikki valittavissa olevat työkalut. Jos olisin valinnut diagrammityyppiksi esimerkiksi arvoverkkomallin tai liiketoimintaprosessidiagrammin niin minulle olisi työkalulaatikkoon avautunut vain ne työkalut mitä tarvitsisin niiden rakentamiseen.

5.2.7 MDG Teknologian jakaminen

Enterprise Architectissa MDG Teknologian jakamiseen muille käyttäjille on käytössä monia eri vaihtoehtoja. Eri vaihtoehtoja punnitessa on hyvä ottaa huomioon kuinka monelle käyttäjälle teknologiaa jaetaan sekä onko teknologiaa tarkoitus päivittää myöhemmässä vaiheessa.

Jos MDG Teknologia tulee käyttöön vain yhdelle henkilölle, voi teknologian luomisesta syntyvän kuvassa 23 näkyvän xml-tiedoston siirtää tietokoneessa sijaitsevaan %APPDATA% kansioon. Enterprise Architectin käynnistyessä se hakee automaattisesti kyseisestä kansioista kaikki MDG Teknologiat ja ottaa ne automaattisesti käyttöön. Tämä tosin edellyttää Enterprise Architectin uudelleen käynnistystä. Xml-tiedoston sijoittaminen %APPDATA% kansioon ei mielestäni ole hyvä tapa jakaa MDG teknologiaa useille käyttäjille, koska xml-tiedosto tulee sijoittaa koneesta käsin kyseiseen kansioon ja se voi tulla työlääksi.

Tapa, jota käytin teknologian jakamiseen muille käyttäjille oli sijoittaa xml-tiedosto Enterprise Architectin Resources kansioon. Resources kansion tulee sijaita SQL-palvelimella tuotanto- tai testiympäristössä, johon otetaan automaattisesti yhteyttä

Enterprise Architectin käynnistyessä. Käynnistyessä kaikki kantaan yhteydessä olevat käyttäjät saavat automaattisesti käyttöönsä MDG Teknologian. Tämän tavan avulla MDG Teknologian päivittäminen useille käyttäjille on helppoa, koska päivitetty tiedosto tulee viedä vain yhteen paikkaan.

5.2.8 Shape Script

Tehtyäni eri työkalulaatikat valmiiksi sain vielä tehtäväksi muokata connectorien nimeltä information flow ulkoasua. Ulkoasua piti muokata niin, että eri elementtien väliset tietovirrat tuli erottaa toisistaan värien avulla. Ainut tapa miten connectorien ulkoasun muokkaaminen oli mahdollista oli shape scriptin avulla.

Eri elementit ja connectorit, jotka ovat saatavilla perustuvat UML:n mukaiseen ulkoasuun. Shape scriptin avulla voi muokata elementtejä niin väriltään kuin myös muodoltaan ja sama pätee connectoreihin. (SPARX Shape Scripts 2017)

Eräät muokattavista connectoreista olivat Value Network Diagrammissa eli arvoverkkomallissa olevat Money Flow, Goods Flow ja Service Flow connectorit. Money Flow connectorin haluttiin olevan väriltään punainen, jotta se erottuisi vahvasti joukosta ja lukija pystyisi heti tunnistamaan diagrammista miten eri tiedot, palvelut ja tuotteet liikkuvat elementtien välillä. Shape scriptejä voi muokata Enterprise Architectin UML Types-valikon kautta, joka löytyy parhaiten Configure välilehdeltä. Valikkoon luodaan ensin uusi stereotyyppi, jonka täytyy olla nimeltään muotoa `<ItseLuotuProfiili::ItseLuotuProfiilielementti>`. Money Flow:n tapauksessa stereotyypin nimeksi tuli siis `EATemplate::EAMoneyFlow`. Base classiksi eli pohjaluokaksi tulee valita se metaluokka, johon kyseinen elementti tai connector on laajennettu Profile-kansiossa. Tässä tapauksessa pohjaluokaksi tuli siis information flow.

Itse shape script luodaan connectorille UML Types-valikon Override Appearance kohdasta valitsemalla vaihtoehto Shape Script ja painamalla Edit nappia. Money Flow connectorin Shape Scriptistä tuli seuraavanlainen:

```
shape main {
    noshadow=true;
    SetLineStyle("Dash");
    SetPen(255, 0, 0, 2);
    LineTo(100, 0);
}

shape target {
    SetFillColor(255, 0, 0);
    StartPath();
}
```

```

MoveTo(0, 0);
LineTo(16, 6);
LineTo(16, -6);
EndPath();
FillPath();
}

```

Shape Scriptin main-luokassa määritellään connectorin linjan tyyli, väri sekä mistä mihin connector kulkee. Halusin sisällyttää connectorin päähän nuolen ja toteutin sen lisäämällä uuden luokan nimeltä target ja piirtämällä nuolen käsin kordinaattien avulla. Lopulta Money Flow connectorista (kuva 33) tuli seuraavan näköinen:



Kuva 33. Money Flow connector

5.3 Projektin yhteenveto

Projektin lopputuloksena syntyi MDG Teknologian avulla tehtyjä toimivia työkalulaatikoita, joiden avulla diagrammien mallintajat saivat helposti käyttöönsä vain tarvittavat elementit ja connectorit tietyn diagrammityyppin luomiseen. Työkalulaatikoiden avulla diagrammien ulkonäköä saadaan yhtenäistettyä ja ehkä jopa jossain vaiheessa standardoitua. Useita diagrammeja on aikaisemmin ollut saatavilla vain PowerPoint muodossa, mutta työkalulaatikoiden avulla niitä voidaan myös alkaa siirtää Enterprise Architect ohjelmistoon.

Projekti oli haastava ennestään tuntemattoman ohjelmiston vuoksi, mutta ohjelmiston valmistajan kattavan dokumentaation vuoksi työ eteni kuitenkin mielestäni hyvin. Olin myös suorassa yhteydessä ohjelmistokehittäjiin SPARX Systemsissä ja pystyin ongelmatilanteissa pyytämään heiltä apua MDG Teknologiaan liittyviin ongelmiin sekä miten parhaiten ratkaista ne.

Työtä voidaan jatkokehittää kehittämällä uusia työkalulaatikoita, joiden sisältämät elementit ja connectorit sopivat eri diagrammien mallinnusten tarpeisiin

Kuvassa 11 on esimerkki työkalulaatikoiden avulla mallinnetusta arvoverkkomallista. Sekä elementit ja connectorit ovat molemmat joko UML tai non-UML pohjaisia ja niiden ulkoasua on tarvittaessa muokattu Shape Scriptien avulla.

6 Pohdinta

MDG Teknologian kehittäminen Enterprise Architect ohjelmistoon annettiin minulle alunperin yhtenä työtehtävistä. Tutkittuani asiaa tarkemmin huomasin kuinka paljon mahdollisuuksia uusien työkalulaatikoiden kehittämiseksi MDG Teknologian avulla oli ja miten sillä pystyttiin auttamaan Enterprise Architectia joka päivä käyttävien työtä.

Ongelmana oli eri elementtien ja connectoreiden suuri määrä ja valinnanvaraa oli aivan liikaa ottaen huomioon kuinka monta elementtiä yhteen diagrammiin tarvittiin. Tämän seurauksena saman tyyppiset diagrammit olivat keskenään eri näköisiä, koska ne oltiin rakennettu eri elementeistä. MDG Teknologian avulla haluttiin korjausta tähän ongelmaan korjaamalla eri diagrammien ulkoasua yhtenäisemmäksi.

Itse toimeksianto oli haastava minulle ennestään tuntemattoman työkalun vuoksi sekä sen sisältämien eri teknologioiden takia. MDG Teknologia osoittautui haastavaksi kehittää sen virheherkkyyden takia. Toisin sanoen pieni virhe missä tahansa teknologiaan kuuluvassa osassa oli riittävä kaatamaan koko teknologian niin ettei mikään toiminut. Toimeksianto oli kuitenkin mielenkiintoinen, joten lopulta monen erehdyksen kautta virheistä päästiin eroon ja teknologia saatiin onnistuneesti käyttöön.

Projektin alussa oli haasteena löytää hyvää materiaalia ja käyttöohjeita miten MDG Teknologiaa kannattaisi lähteä kehittämään eteenpäin ja miten tehdä se mahdollisimman tehokkaasti. Parhaiksi lähteiksi projektin tekoon osoittautuikin Enterprise Architectin kehittäjän SPARX:n omat sivut ja niiden sisältämät dokumentaatiot ja käyttöohjeet eri ohjelmiston mahdollisuuksista. Myös tanskalaisen IT-yrityksen TigerTeamin omat käyttöohjeet MDG Teknologian käyttöönottoon liittyen tarjosivat paljon tukea varsinkin projektin alkuun saamisessa.

Erityistä pään raapimista tarjosi non-UML pohjaiset elementit ja miten saada ne osaksi MDG Teknologiaa. Vastaus löytyi lopulta SPARX:n omilla sivuilla olleesta dokumentista, missä neuvottiin vaihe vaiheelta miten tuo saadaan tehtyä. Ongelmana ennen tuon dokumentin löytämistä oli se, että tuo dokumentti oli nimenomaan suunnattu Enterprise Architectin versiolle 13. Olin selaillut läpi muita SPARX:n tarjoamia dokumentteja, mutta niistä ei siinä tilanteessa ollut apua, koska ne oli tarkoitettu ohjelmiston aiemmille versioille. Joskus Enterprise Architectin eri versioiden välillä ei ollut juuri mitään eroavaisuuksia, mutta tietyissä tilanteissa kuten juuri tässä oli erittäin tärkeää seurata juuri itsellä käytössä olevalle versiolle tarkoitettuja ohjeita.

Työkalulaatikoiden kehittäminen oli hyödyllistä, koska eri diagrammeissa käytettäviä elementtejä ja connectoreita haluttiin yhtenäistää. Näin eri diagrammien ulkoasu vastaa sitä yhtenäistä linjaa, jota halutaan käyttää niin elementtien ja connectorien värien- kuin myös niiden ulkomuodon puolesta. Lopputulos on se, että eri diagrammit ovat väreiltään Wärtsilän brändin mukaisia ja seuraavat niille asetettua yhtenäistä linjaa.

Opinnäyteprosessi meni kokonaisuudessaan mielestäni hyvin. Sain aloitella työn tekemistä rauhassa ilman suurempaa kiirettä, joten itse työn tekemiselle ja lopulta sen hiomiselle jäi runsaasti aikaa. Aiemmin hankittu osaaminen koulun ICT-Architectures- sekä ohjelmointikurssien kautta auttoi minua pääsemään työssä alkuun. Etenkin Shape Scriptien kirjoittaminen helpottui huomattavasti, koska kielessä käytettävä syntaksi oli suurimmilta osin samaa muiden ohjelmointikielien kanssa.

Uskon, että tämän opinnäytetyön tekeminen tuotti teknisessä mielessä suurta ammatillista hyötyä ja avasi myös hieman silmiäni kokonaisarkkitehtuurin tuomiin hyötyihin sekä sen mallintamisen maailmaan.

Lähteet

Julkinen hallinto 2017. Julkisen hallinnon kokonaisarkkitehtuuri, KA-Kypsyystasomalli. Luettavissa: <http://vm.fi/documents/10623/307673/Kypsyystasomalli/e15a9c97-bdcf-4cbf-b1fa-31e9e99a188b>. Luettu: 30.8.2017.

SOLEA-hanke 2017. Zachman Framework. Luettavissa: http://epublications.uef.fi/pub/urn_isbn_978-952-61-0723-3/urn_isbn_978-952-61-0723-3.pdf. Luettu 31.8.2017.

SPARX 2017. MDG Technologies. Luettavissa: <http://www.sparxsystems.fr/resources/user-guides/modeling/mdg-technologies.pdf>. Luettu: 7.7.2017.

SPARX Products 2017. Luettavissa: <http://www.sparxsystems.com/products/ea/>. Luettu: 7.7.2017.

SPARX Shape Scripts 2017. Luettavissa: http://www.sparxsystems.com/enterprise_architect_user_guide/10/extending_uml_models/shape_editor_2.html. Luettu: 11.7.2017.

SPARX Systems 2017. Luettavissa: <https://www.sparxsystems.eu/start/home/>. Luettu: 7.7.2017.

TigerTeam 2017a. Luettavissa: <https://www.tigerteam.dk/2011/how-to-develop-mdgs-for-enterprise-architect-part-1/>. Luettu: 7.7.2017.

TigerTeam 2017b. Luettavissa: <https://www.tigerteam.dk/2011/how-to-develop-mdgs-for-enterprise-architect-part-2/>. Luettu: 11.7.2017.

TOGAF 9.1 docs 2017. Luettavissa: <http://pubs.opengroup.org/architecture/togaf9-doc/arch/>. Luettu: 31.8.2017.

Tämä on Wärtsilä 2017. Luettavissa: <https://cdn.wartsila.com/docs/default-source/local-files/finland/t%C3%A4m%C3%A4-on-w%C3%A4rtsil%C3%A4-20166aa4eb4a7f0f601bb10cff00002d2314.pdf?sfvrsn=0>. Luettu: 5.7.2017.

Valtiovarainministeriö 2017. Luettavissa: <http://vm.fi/kokonaisarkkitehtuuri-digitalisaatiossa>. Luettu: 6.7.2017.

Valtiovarainministeriö kokonaisarkkitehtuuri 2017. Valtiovarainministeriön kokonaisarkkitehtuuri digitalisaatiossa kuvistuskuvat. Luettavissa: <http://vm.fi/kokonaisarkkitehtuuri-digitalisaatiossa>. Luettu: 6.7.2017.

Wärtsilä Kestävä Kehitys 2017. Luettavissa: <https://www.wartsila.com/sustainability>. Luettu: 21.9.2017.

Wärtsilä Lyhyesti 2017. Luettavissa: <https://www.wartsila.com/fi/wartsila>. Luettu: 5.7.2017.

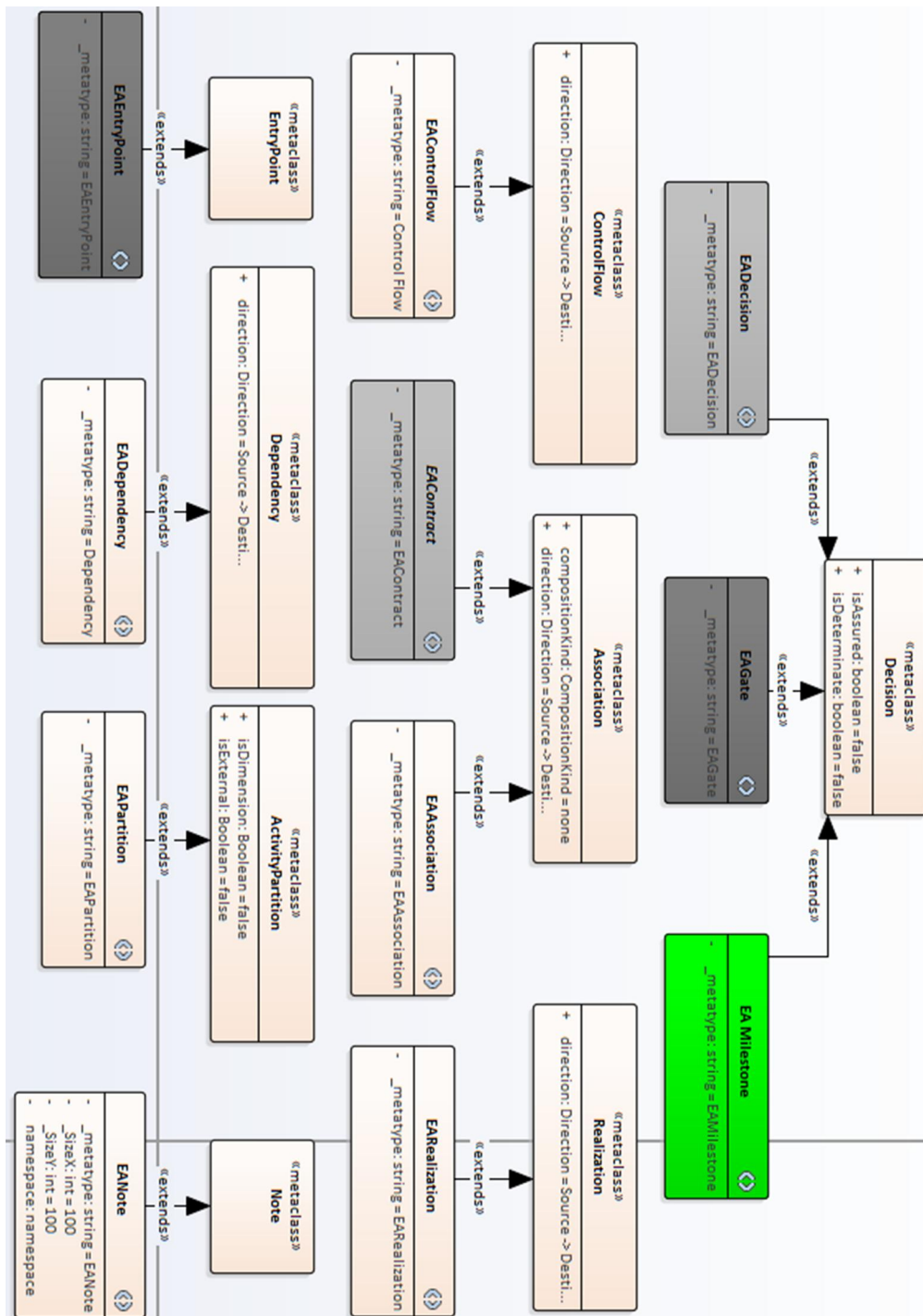
Wärtsilä Media 2017. Luettavissa: <https://www.wartsila.com/fi/media-fi/uutinen/02-06-2015-uusi-wartsila-31--moottori-paasi-guinnessin-ennatystenkirjaan>. Luettu: 6.7.2017.

Wärtsilä Products 2017. Wärtsilä 31 kuvituskuva. Luettavissa: <https://www.wartsila.com/products/marine-oil-gas/engines-generating-sets/diesel-engines/wartsila-31>. Luettu: 6.7.2017.

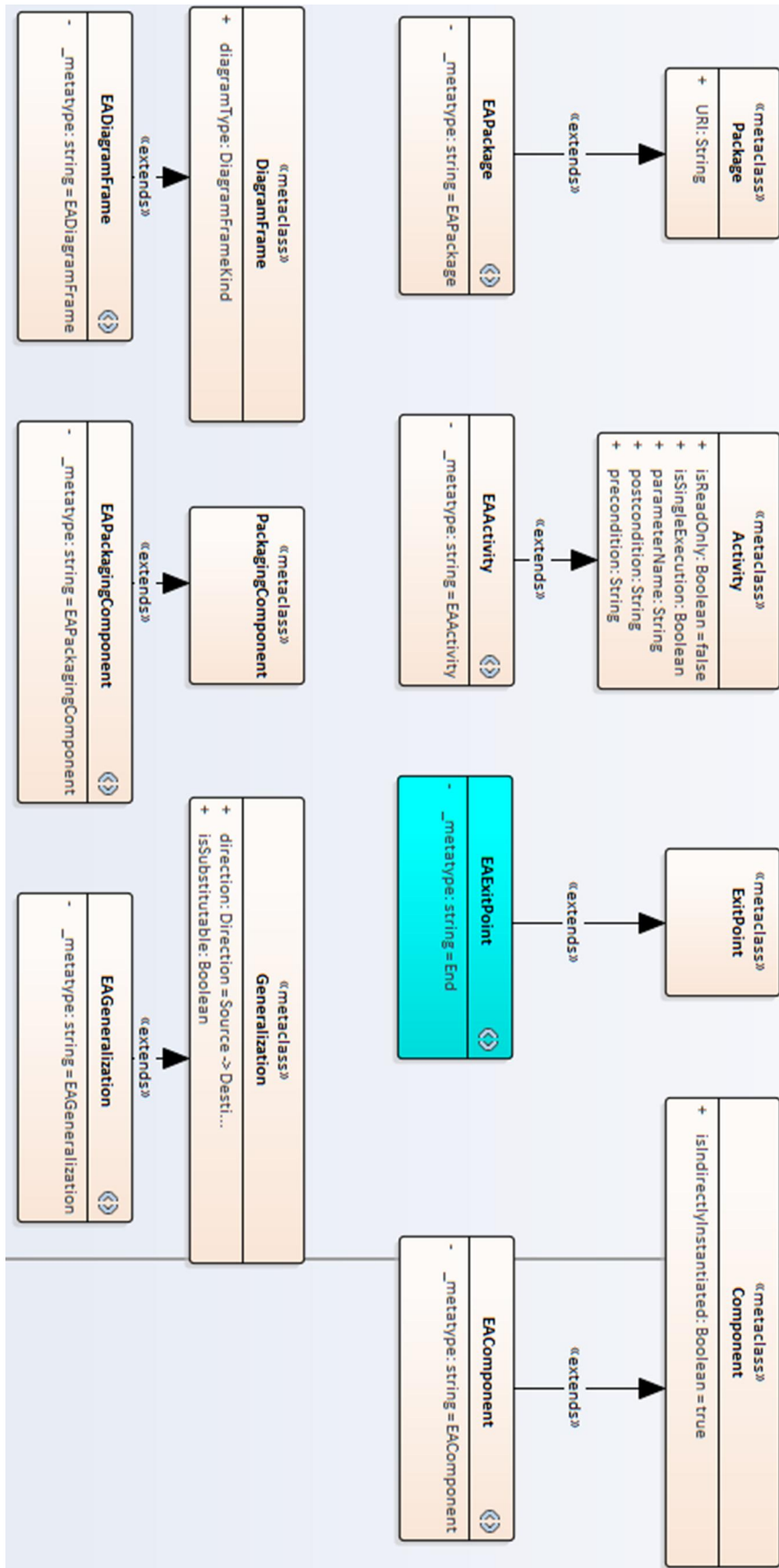
Wärtsilä Strategia 2017. Luettavissa: <https://www.wartsila.com/fi/wartsila/strategia>. Luettu: 21.9.2017.

Zachman 2017. Luettavissa: <https://www.zachman.com/about-the-zachman-framework>. Luettu: 12.7.2017.

Liite 2. EATemplate Profile 2(3)



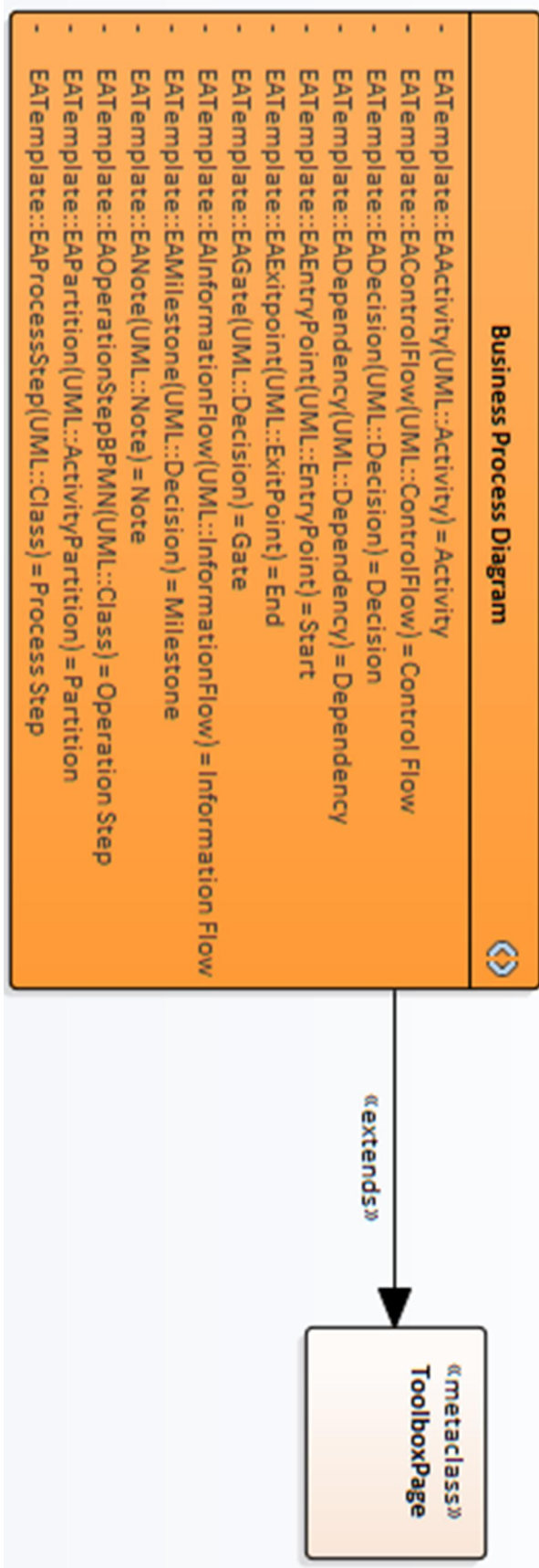
Liite 3. EATemplate Profile 3(3)

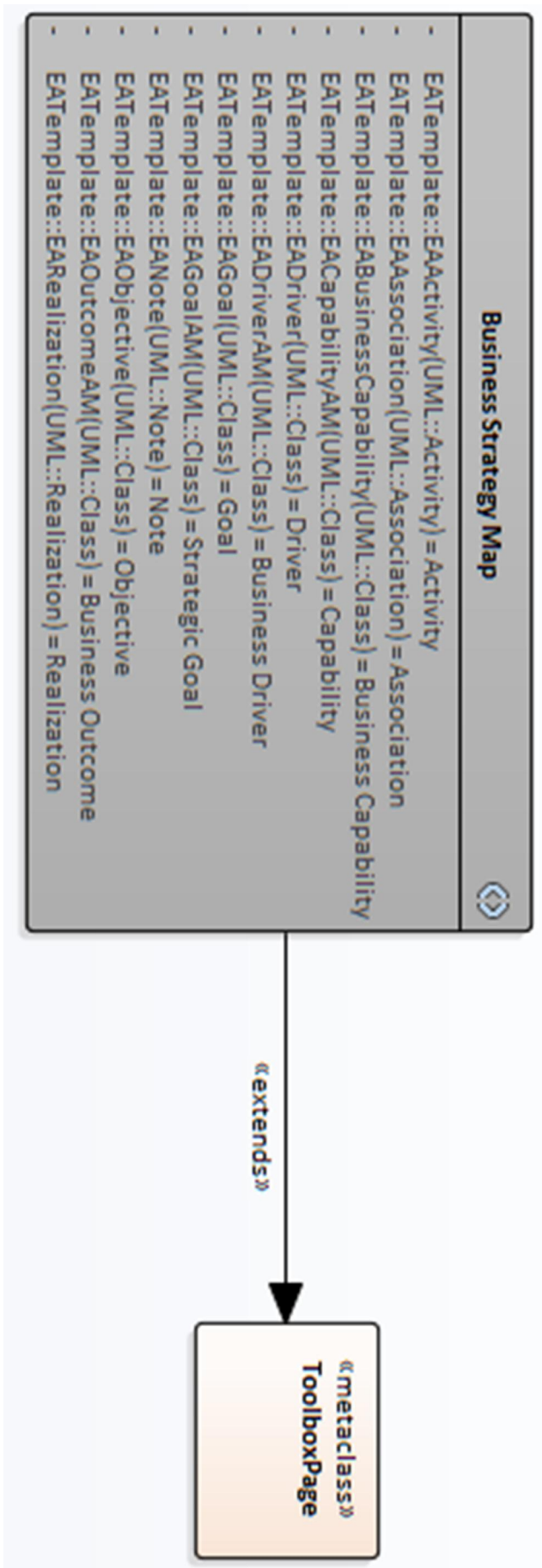


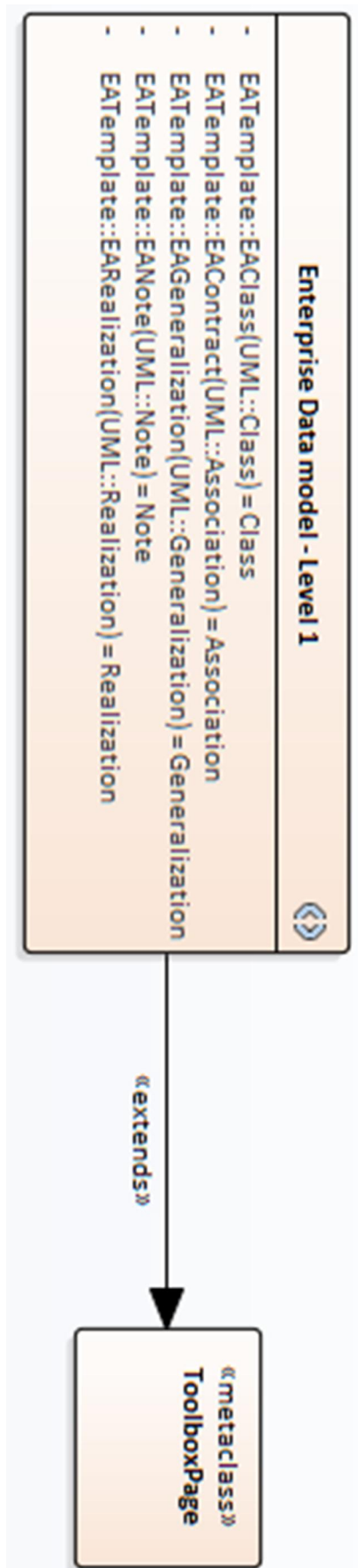
Liite 4. EATemplate Toolbox | Application Landscape Diagram



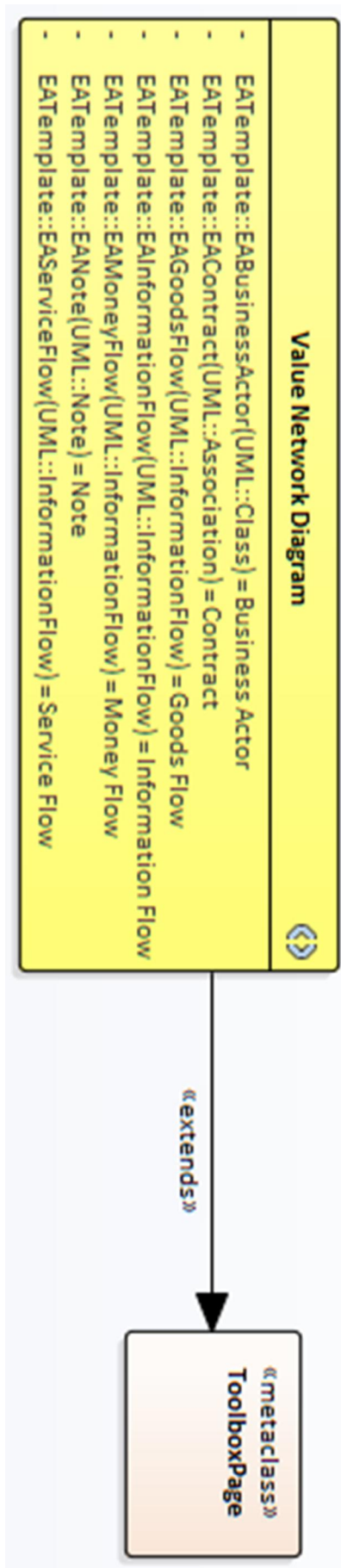
Liite 5. EATemplate Toolbox | Business Process Diagram



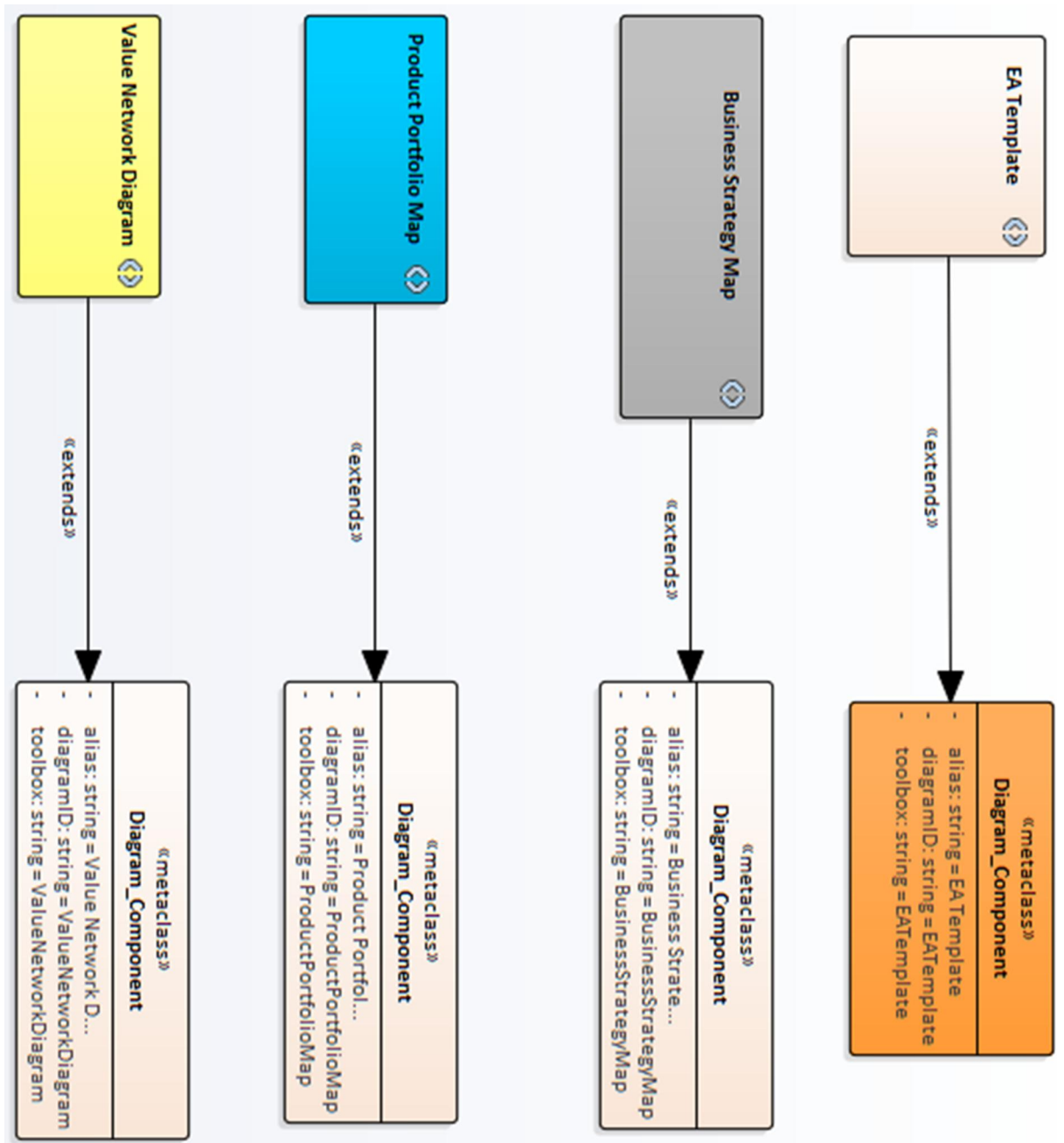








Liite 11. EA Template Diagram 1(2)



Liite 12. EATemplate Diagram 2(2)

