

Tampereen ammattikorkeakoulu
Sähkötekniikan koulutusohjelma
Ohjelmistotekniikka
Janne Tenhovirta

Tutkintotyö

TEKNOLOGIASELVITYS

MICROSOFT BIZTALK SERVER JA ORACLE-TIETOKANTA

Työn ohjaaja
Työn tilaaja
Tampere 11/2008

Ohjelmistotekniikan lehtori Erkki Hietalahti
Cybersoft Oy

Tampereen ammattikorkeakoulu
Sähkötekniikan koulutusohjelma, ohjelmistotekniikka

Janne Tenhovirta

Teknologiaselvitys: Microsoft Biztalk Server ja Oracle-tietokanta

22 sivua

Marraskuu 2008

Työn ohjaaja ohjelmistotekniikan lehtori Erkki Hietalahti

Työn tilaaja Cybersoft Oy

TIIVISTELMÄ

Tämän työn tarkoituksena oli selvittää Microsoft Biztalk -integroitipalvelimen toimintaa ja yhteensopivuutta Oracle-tietokannan kanssa. Tavoitteena oli Biztalk-sovellusten suunnittelun peruseriaatteiden selventäminen ja pienen integrointisovelluksen toteuttaminen Oracle-tietokannan ja kolmannen osapuolen tarjoaman ulkoisen liittymän välille. Tämä integrointisovellus on osana suurempaa tietokantasovelluskokonaisuutta.

Biztalk-integrointiohjelmisto on tarkoitettu organisaatioiden ja yritysten työkaluksi integroimaan järjestelmiä ja sovelluksia toisiinsa. Biztalk-sovellusten kehitysympäristönä toimii Microsoft Visual Studio.NET. Biztalk-palvelin kehitysympäristöineen oli asennettu erilliselle palvelinkoneelle, jota käytettiin etätyöpöytäyhteyden kautta. Sovelluksen kehitysprosessina käytettiin vesiputousmallia.

Työn tuloksena syntyi tärkeä osa asiakkaan tietokantasovellukseen. Integrointisovellus toimii erittäin luotettavasti, eivätkä käyttökatkot tai virheellinen toiminta ole sitä vaivanneet.

Tampere Polytechnic
Electrical Engineering, Software Engineering

Janne Tenhoviirta
Microsoft Biztalk Server & Oracle Database
22 pages
December 2008
Thesis Supervisor: Lecturer Erkki Hietalahti
Co-operating Company: Cybersoft Oy

ABSTRACT

The purpose of this thesis was integrate a third party data source to an Oracle database. This was done by designing and creating a Microsoft Biztalk Server integration application. Also important was to gain some knowledge of general integration theory and understanding Biztalk's approach to this field of software development. Learning how to use Biztalk came as a byproduct of the whole process. The end result application became a part of a larger database application.

The Biztalk integration server is meant to live up to its name i.e. serving integration needs. The server runs applications developed in Microsoft's own Visual Studio.NET. The Biztalk add-on is an easy to use graphic design tool. The Biztalk Server and development add on were installed on a server machine running. Development was done via a remote desktop connection. The development process was very iterative, bearing resemblance to the waterfall development model.

The resulting application of this thesis is a somewhat important part of the client's database application. The integration as a whole is very reliable with minimal erroneous function.

Termien ja lyhenteiden luettelo

| | |
|---------------------|--|
| BAM | Liiketoimintamonitorointi |
| Biztalk | Biztalk Server R2 -integrointipalvelin |
| BPEL | Liiketoimintaprosessin suorituskieli, WS:n osa |
| BPM | Liiketoimintaprosessin hallinta |
| BRE | Liiketoimintasääntömoottori |
| B2B | Yritysten välinen toiminta |
| EAI | Yrityksen sisäinen sovellusintegraatio |
| Integrointipalvelin | Integrointisovelluksia pyörittävä palvelin |
| ODBC | Microsoftin määrittelemä rajapinta tietokantayhteyksille |
| Orkestraatio | Biztalk-integrointisovelluksen logiikkaydin |
| Skeema | XML-rakenteen kuvaus |
| SOA | Palvelukeskeinen arkkitehtuuri |
| SOAP | Protokolla proseduurien etäkutsuille, WS:n osa |
| Web Services(Ws) | Webin yli toimivien palveluiden tarjoamista ja käyttämistä helpottava kokoelma protokollia |
| WSDL | XML-standardi WS-palveluiden kuvaamiseksi |
| W3C | Webin standardeja hallitseva organisaatio |
| XML | Täsmällisen tiedon esittämiseen tarkoitettu standardi |
| XSD | Standardi XML-rakenteen kuvauksesta |
| XSLT | XML-rakenteen muutoskieli |

SISÄLLYSLUETTELO

| | |
|--|----|
| 1 Johdanto | 1 |
| 2 Biztalk-integrintipalvelinohjelmisto | 2 |
| 2.1 Käyttötarkoitus | 2 |
| 2.2 Arkkitehtuuri | 2 |
| 2.3 XML-kieli Biztalk-palvelimen yhteydessä | 4 |
| 2.4 Biztalk-sovelluksen sisäinen rakenne | 5 |
| 2.4.1 Vastaanottoportti, lähetysportti ja viestilaatikko | 6 |
| 2.4.2 Sovitin | 6 |
| 2.4.3 Orkestraatiot | 6 |
| 2.4.4 Viestimuunnokset | 7 |
| 2.4.5 Web Services-palvelu | 7 |
| 2.4.6 Viestien ohjaus | 8 |
| 2.5 Biztalk Server ja muut integrintiohjelmit | 8 |
| 2.6 Kehitysympäristö | 9 |
| 2.6.1 Oracle-tietokantasovittimen asennus | 9 |
| 2.6.2 Oracle tietokantasovittimen ominaisuudet | 10 |
| 3 Biztalk-sovelluskehitys | 10 |
| 3.1 Visual Studio.NET-kehitysympäristö | 11 |
| 3.2 Biztalk Administration Console | 12 |
| 3.3 Vastaanottoportti | 13 |
| 3.4 Putki | 13 |
| 3.5 Skeema | 14 |
| 3.6 Kartta | 14 |
| 4 Integrintisovelluksen suunnittelu ja toteutus | 15 |
| 4.1 Kohdejärjestelmä | 15 |
| 4.2 Ulkoinen liittymä | 16 |
| 4.3 Sovelluksen suunnittelu | 16 |
| 4.4 Sovelluksen toteutus | 17 |
| 4.4.1 Orkestraatio | 17 |
| 4.4.2 Skeemat | 18 |
| 4.4.3 Kartta | 19 |
| 5 Tulosten arviointi | 20 |
| Lähteet | 22 |

1 JOHDANTO

Nykyaikaisissa yrityksissä kaiken toiminnan tehokkuuden parantaminen on entistä tärkeämmässä asemassa. Tietojärjestelmillä kyllästetyn yritysmaailman tehokkuutta parantamaan on kehittynyt useita toiminnanohjausjärjestelmiä. Yksinkertaistettuna toiminnanohjausjärjestelmillä tavoitellaan yrityksen koko toimintaprosessin hallitsemista yhdellä käyttöliittymällä. Toimintaprosessi käsittää kaiken mitä yritys tekee ja mitä sen sisällä tapahtuu, siivousaikatauluista kirjanpitoon.

Usein yrityksen sisällä on ollut useita tietojärjestelmiä eri asioiden tai osastojen hallintaan. Sovellus- ja järjestelmäintegraatio on tehokas keino saattaa nämä tietojärjestelmät saman käyttöliittymän alle. Integroinnin jälkeen järjestelmän käyttäjä näkee vain käyttöliittymän rajapintana vanhoihin erillisiin järjestelmiin. Moni toiminnanohjausjärjestelmä onkin rakennettu järjestelmäintegraation avulla. Biztalk Server -ohjelmisto on Microsoftin ratkaisu yritysten ja yhteisöjen järjestelmä- ja sovellusintegraatiotarpeisiin.

Käytännössä Biztalk on palvelintyyppinen ohjelmisto, jonka kehitystyökalu toimii Microsoft Visual Studio .NET -ympäristössä. Eri järjestelmien yhteenliittämisen helpottamiseksi Microsoft tarjoaa Biztalkin mukana useita sovittimia järjestelmiä tai sovelluksia varten. Oraclen tietokannan kanssa toimimiseen löytyy Microsoftilta ilmainen sovitin.

Työn varsinaisena tavoitteena oli luoda integrointisovellus asiakkaan järjestelmän ja kolmannen osapuolen tietolähteen välille. Nollatason lähtötiedot asettivat omat vaatimuksensa Biztalk-integrointisovelluksen luomiseen. Suuri osa työstä menikin Biztalk-palvelimen toiminnan ymmärtämiseen ja selvittämiseen.

2 BIZTALK-INTEGROINTIPALVELINOHJELMISTO

Biztalk on Microsoftin integrointipalvelinohjelmisto. Tämä työ pohjautuu Biztalk Server 2006 R2-versioon, joka julkaistiin syksyllä 2007.

2.1 Käyttötarkoitus

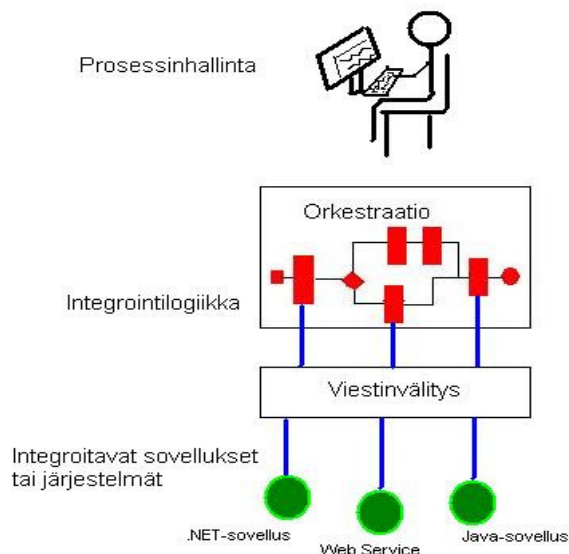
Microsoftin strategiassa Biztalk-integrointipalvelinohjelmisto on tarkoitettu organisaatioiden ja yritysten työkaluksi luomaan automatisoituja integrointiprosesseja kolmella eri alueella. Organisaation sisäisissä prosesseissa (Enterprise Application Integration, EAI) Biztalk yleensä toimii ydinsovelluksena palvelukeskeisessä arkkitehtuurissa (Service-oriented Architecture, SOA). Toisena alueena on organisaatioiden välinen integrointi (Business-to-Business, B2B). Biztalkin kolmantena käyttöalueena Microsoft pitää liiketoimintaprosessin hallintaa (Business Process Management, BPM). (MSDN a 2008)

2.2 Arkkitehtuuri

Biztalk-integrointisovelluksen arkkitehtuurissa kokonaisprosessi jakautuu kolmeen kerrokseen, jotka ovat esitettynä kuviossa 1. Alimmassa kerroksessa ovat ulkoiset järjestelmät ja sovellukset, jotka joko käyttävät integrointisovelluksen kautta toisten sovellusten tietoja tai antavat tietojaan ulkopuolisten käytettäväksi.

Toisessa kerroksessa sijaitsee integrointilogiikan toteutus. Biztalk-integrointisovelluksen ytimenä on prosessilogiikan sisältämä orkestraatio. Sen sisältämää vaiheiden sekvenssiä suoritetaan niin kauan kuin sovellus ja sen osat ovat aktiivisena, eli niin kauan kuin prosessinohjaaja haluaa. Logiikkaytimiä eli orkestraatioita voi olla useita. Viestinvälitys hoitaa sekä sovelluksen sisäisen viestiliikenteen että kommunikoinnin ulkoisten liittymien

kautta toisten sovellusten kanssa. Yleensä logiikan suoritus alkaa joko ulkoisesta palvelupyynnöstä tai ajastettuna. (MSDN a 2008)



Kuvio 1. Integrointisovelluksen arkkitehtuuri.

Ylimmäisessä kerroksessa istuu liiketoimintaprosessin ohjaaja, joka tarkkailee ja hallitsee prosessia. Valvojan ei tarvitse tuntea toteutusta, mutta tällaisessa tilanteessa Business Activity Monitoring -ohjelmankomponentin käyttö on tarpeen. BAM-monitorointia käyttäen esim. myyntijohtaja voi tarkkailla kaupan sujuvuutta tutustumatta tarkemmin koko järjestelmän toteutukseen. (MSDN c 2008)

Prosessinhallinnassa voidaan käyttää Microsoftin liiketoimintasääntömootoria (Business Rules Engine, BRE), josta on hyötyä erityisesti monimutkaisten, monta logiikkaydintä sisältävien integrointisovellusten hallinnassa. Lyhyesti sanottuna sääntömootori on .NET-yhteensopiva luokkakirjasto, joka auttaa luomaan ja ylläpitämään integrointisovelluksia mahdollisimman pienellä vaivalla. Lisäksi järjestelmässä voidaan käyttää myös automaattista raportointia, jolloin jatkuvan valvonnan tarve vähenee. (MSDN b 2008)

2.3 XML-kieli Biztalk-palvelimen yhteydessä

XML eli Extensive Markup Language on standardi tiedon ja sen rakenteen esittämisestä. Se on World Wide Web Consortium-organisaation eli W3C:n hyväksymä. Alunperin XML-kieli kehitettiin Web-dokumenttien tietosisällön kuvaamiseen. XML-kielillä siis kuvataan tietoa lisäämällä siihen metatietoa. Tällöin tieto saadaan kuvattua yksikäsitteisesti. Biztalk-integrointipalvelin käyttää XML-kielisiä viestejä sisäisessä liikenteessään. Tästä on hyötyä, sillä järjestelmäintegroinnissa XML-kieli on paljon käytetty kuvauskieli, ja esimerkiksi Web Services-protokollakentän SOAP-protokollan avulla liikkuvat palveluiden pyyntö- ja vastausviestit ovat XML-kielisiä. (W3C 2008)

Tietokannoissa oleva informaatio on yleensä strukturoitua, taulukkomäistä. Tämän tyyppisen tiedon esittämiseen XML-kieli on hyvin sopiva työkalu. Tiedon täsmällisyys on tietokantojen tärkeä ominaisuus. Seuraavassa pieni esimerkki XML-kielisestä rakenteesta:

```
<Car>
  <Name>Ford Mustang</Name>
  <Color>Metallic green</Color>
  <Engine>5.7 liter V8</Engine>
  <Power>350 hp</Power>
  <Price>$23990</Price>
</Car>
```

Tietokantaympäristöön käännettynä edellä oleva XML-rakenne sisältää yhden rivin viisi saraketta dataa Car-nimisestä taulusta.

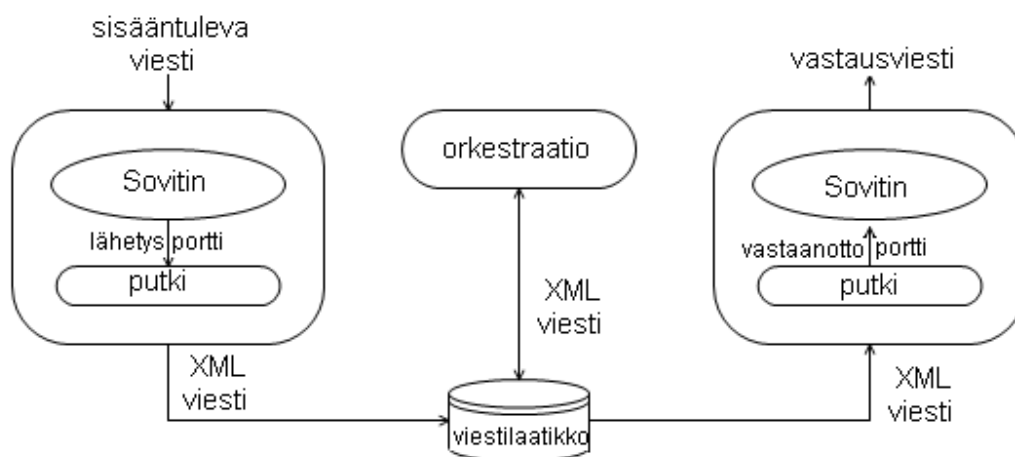
Biztalk-integrointipalvelimen sisäisten viestien sekä integrointisovelluksesta ulos lähtevien viestien rakenne kuvataan käyttämällä W3C-organisaation XML-skeemastandardia, lyhyesti XSD eli XML Schema Definition. Mikäli integrointisovellukseen sisään tuleva data ei ole valmiiksi XML-standardin mukainen, tehdään sille muunnos. XSD-standardi määrittää XML-dokumentin rakenteen tagien, hierarkian ja tietotyyppien avulla. XSD-

standardin mukaisesti määriteltyä skeemaa vasten voidaan varmentaa XML-dokumentteja, jolloin voidaan todeta niiden rakenteellinen virheettömyys.

Biztalk käyttää XML-dokumenttien muunnoksissa Extensible Stylesheet Language Transformations -muunnoskieltä, jonka lyhenne on XSLT. Biztalkissa skeemamuunnokset eli kartoitukset toteutetaan XSLT-kielillä, jota voisi kuvailla tyylitiedostona- tai skriptinä. Muunnokset ovat tarpeen, kun lähdejärjestelmästä saatu XML-skeema poikkeaa kohdejärjestelmän vaatimasta skeemasta. (W3C 2008)(MSDN a 2008)

2.4 Biztalk-sovelluksen sisäinen rakenne

Biztalk-integrointisovelluksen sisäinen toimintamalli pohjautuu XML-kielisiin viesteihin. Toimintaa pyörittää Biztalk-palvelimen viestinvälitysydin, joka lähettää ja vastaanottaa viestit käsittelyvaiheesta toiseen. Kuviossa 2 on Biztalk-sovelluksen sisäinen rakenne pääkomponenttitasolla.



Kuvio 2. Biztalk-palvelimen sisäinen rakenne

Orkestraatio sisältää prosessilogiikan ja kommunikoi lähettävän ja vastaanottavan osapuolen kanssa viestilaatikon välityksellä. Seuraavissa kappaleissa käsitellään kuvan komponentteja perusteellisesti. (MSDN d 2008)

2.4.1 Vastaanottoportti, lähetysportti ja viestilaatikko

Palvelimelle tulevat viestit vastaanotetaan vastaanottoportin avulla. Portit ovat aina joko yksisuuntaisia tai pyyntö - vastaus-portteja. Vastaanottoporttiin täytyy aina liittää vastaanottoaika, eli lähde josta vastaanottoportti vastaanottaa viestit. Vastaanottoaika taas koostuu sovittimesta, jonka kautta palvelin ottaa viestejä vastaan, ja putkesta, jolla viestit muokataan palvelimen viestinvälitysmekanismin sopivaan muotoon.

Vastaanottosovittimen jälkeen viesti menee putkeen, jossa se muunnetaan haluttuun muotoon, esimerkiksi muunnetaan tekstitiedosta XML-viestiksi tai puretaan moniosainen viesti yksinkertaisempaan muotoon. Vastaanottoportin läpi tultuaan viesti päättyy viestilaatikkoon, joka on käytännössä Microsoft SQL Server -tietokanta josta viestinvälitysydin ohjaa viestit oikeisiin kohteisiin. Vastaanottoportteja vastaavat lähetyspäissä lähetysportit. Ne muistuttavat toiminnaltaan vastaanottoportteja, ja koostuvat samoista osista. Sovitin ja putki löytyvät siis myös lähetysporteista. (MSDN d 2008)

2.4.2 Sovitin

Biztalk-integrointipalvelinohjelmistolle on ilmaiseksi ladattavissa on sovittimia monille standardeille ja järjestelmille. Sovitin on rajapinta Biztalk-palvelimen ja ulkoisen järjestelmän tai sovelluksen välillä. Ilmaiseksi saatavien sovittimien lisäksi omien sovittimien ohjelmointi on mahdollista. Myös kolmansien osapuolten maksullisia sovittimia on myytävänä. Tässä työssä tullaan käyttämään Oracle-tietokannan kanssa yhteensopivaa sovitinta ja tiedostosovitinta. (MSDN d 2008)

2.4.3 Orkestraatiot

Orkestraatiot eli liiketoimintalogiikan kuvaukset ovat integrointisovelluksissa keskeisessä roolissa. Niiden avulla määritellään palvelimen toimintalogiikka, mitä milläkin tietovirralla tehdään ja minne niitä jatkossa lähetetään ja missä muodossa. Orkestraatiot ovat siis ikään

kuin integrointisovellusten moottoreita. Ne saavat halutut viestit viestilaatikosta ja voivat palvelua kutsuttaessa toteuttaa sisäisiä funktioita tai käynnistää ulkoisten palvelujen kutsuja. Tällä tavoin orkestraatioilla voidaan muodostaa ns. komposiittisovelluksia, jotka käyttävät usean eri järjestelmän osia. Orkestraation ehtologiikan avulla voidaan toimintaa ohjata viestien sisällön tai tyyppin mukaan. Esimerkiksi viestin virheellinen data voisi käynnistää viestin pyytämisen lähteestä uudestaan.

Biztalkilla toteutetut orkestraatiot ovat yhteensopivia muiden BPEL-kieltä tukevien integrointijärjestelmien kanssa. BPEL-kieltä käytetään liiketoimintaprosessien suorittamiseen. Vieraassa järjestelmässä luotuja monimutkaisia liiketoimintalogiikoita ei tarvitse luoda Biztalkilla uudelleen vaan ne voidaan tuoda XML-muotoisena Biztalkiin, kunhan vieras järjestelmä vaan tukee BPEL-kieltä. Tämä toimii tietysti myös toisinpäin: Biztalkin orkestraatiot voidaan viedä vieraaseen järjestelmään. (MSDN d 2008)

2.4.4 Viestimuunnokset

Biztalkin XML-viestejä voidaan muuntaa skeemasta toiseen karttojen avulla. Skeema on XML-viestin rakenteen kuvaus. Viestejä voidaan siis muuntaa rakenteesta toiseen. Tämä on tärkeä ominaisuus, sillä ulkoisten järjestelmien XML-viestit ovat harvoin rakenteeltaan identtisiä. Vaikka sisällölle ei muunnoksia tai korjauksia tarvitsisikaan tehdä, joudutaan skeemamuunnos useimmiten tekemään. Viestimuunnoksien avulla eri lähteistä saadut tietovirrat saadaan rakenteeltaan homogeenisiksi. (MSDN d 2008)

2.4.5 Web Services -palvelu

Biztalk-integrointipalvelimella voidaan sekä käyttää muiden tarjoamia palveluita että tarjota omia palveluita muille. SOA-protokollan mukaisten rajapintojen käyttäminen on yksinkertaista. WDSL-kuvauksen osoitteen avulla Biztalk muodostaa automaattisesti palvelun kutsumisen ja viestin vastaanoton XML-skeemat. Palvelua kutsuvaa orkestraatiota

varten Biztalk luo myös valmiiksi portin. Kehittäjän harteille jää kutsuviestin luominen ja sen ohjaaminen orkestraatiosta palvelun kutsuporttiin.

Biztalkilla voidaan tarjota orkestraatioita ja skeemoja automatisoidusti palveluina. Orkestraation tarjoaminen tarkoittaa sitä, että SOAP-portillinen orkestraatio julkaistaan Web Service-palveluna. Biztalk luo orkestraatiolle automaattisesti WSDL-kuvauksen, jonka avulla palvelua kutsutaan muista järjestelmistä. Skema voidaan myös julkaista automaattisesti Web Servicenä. Biztalk luo tarvittavat portit ja WSDL-kuvauksen. (MSDN e 2008)

2.4.6 Viestien ohjaus

Viestien ohjauksella tarkoitetaan tässä kohtaa tietovirtojen määränpään määräytymistä Biztalkissa. Palvelukeskeisessä arkkitehtuurissa viestin ohjaus aloitetaan palvelun pyytäjän kutsusta. Pyyntöön rakennetaan vastaus orkestraatioissa määritetyn toimintalogiikan mukaisesti. Valmistuttuaan vastaus toimitetaan pyytäjälle eli kun orkestraation kuvaamat operaatiot on suoritettu. Perinteinen integrointipalvelin toimii siten, että järjestelmään tuleva viesti ohjautuu lähettäjän tai sisällön perusteella. Palvelukutsulle määritettyjä parametreja voidaan käyttää hyväksi logiikan suorituksessa. Esimerkiksi tietoa voidaan hakea saman palvelun avulla useasta eri kohteesta. Kohdevalinta määritellään palvelukutsun parametreilla. (MSDN d 2008)

2.5 Biztalk Server ja muut integrointiohjelmistot

Microsoftin mukaan integrointijärjestelmän toteuttamisesta Biztalk-integrointipalvelimen avulla on yritetty tehdä helppoa, edullista ja yksinkertaista. Microsoftin mielestä heidän valttinsa on integrointisovelluksien kehitysympäristön helppokäyttöinen graafinen käyttöliittymä. Biztalk-integrointipalvelimen kilpailijoita ovat mm. IBM WebSphere ja BEA AquaLogic Service Bus. Kilpailijoihinsa nähden Biztalk ei sovi yhtä hyvin paljon kapasiteettia vaativien sovellusten toteuttamiseen. Pienten ja keskisuurten

integrointiprojektien toteuttamiseen se kuitenkin sopii. Microsoftin muiden tuotteiden suosio on luultavasti yleisimpiä syitä Biztalkin valintaan integrointipalvelimeksi. Moni valitsee mielellään tutun ja helpon käyttöliittymän.

Biztalk on edennyt jo kolmannen sukupolven versioonsa, joten se on luultavasti päässyt jo yli lapsentaudeistaan. Sukupolvien välillä on tapahtunut kehitystä ja ero muutaman vuoden takaiseen versioon on suuri. Luottamusta herättää myös se että Microsoft kehittää aktiivisesti Biztalkia. Microsoftin tarjoamien tilastojen mukaan Biztalkia käyttää myös suurin osa suuryrityksistä. (MS Biztalk 2008).

2.6 Kehitysympäristö

Sain Biztalk-integrointipalvelimen käyttööni valmiiksi asennettuna. Biztalk-palvelin toimii Windows Server 2003 käyttöjärjestelmässä tavallisessa palvelinkoneessa. Keskustelin asennuksesta sen suorittaneen kollegani kanssa. Varsinainen asentaminen sujui kollegani mukaan Microsoftin tarjoaman ohjeen mukaan yksinkertaisesti ja ohjatusti. Ongelmia ilmeni vasta käyttökuntoon saattamisessa. Jotta liikenne palvelimelle ja siltä pois päin saatiin kulkemaan, täytyi palomuurista avata portteja. Kokemattomuus ja järjestelmätuntemuksen puute olivat asentajan rasitteena tässä tapauksessa. Lopulta palvelin saatiin kuitenkin toimintaan. Biztalk-palvelinta käytettiin Windowsin etätyöpöytätoiminnon avulla tavalliselta työasemakoneelta.

2.6.1 Oracle-tietokantasovittimen asennus

Oracle-tietokantasovitin Biztalk-integrointipalvelimelle sisältyy ilmaiseen ”Microsoft BizTalk Adapters for Enterprise Applications”-pakettiin. Sen asennustiedoston lataamisen ja purkamisen jälkeen valitsin asennettavaksi Oracle-tietokantasovittimen. Asennuksen jälkeen aloin selvittää sovitin toiminnan testaamista. Käytin apuna Googlen avulla löytynyttä esimerkkiä, jossa sovitin käyttöä opastetaan esimerkkikuvien avulla. Kaikki ei kuitenkaan sujunut esimerkkien kuvaamalla tavalla, sillä sovitin ei kyennyt luomaan

yhteyttä Oracle-kantaan. Tutkin ensiksi ODBC-tietokanta-ajurista vaadittavaa versiota. Selvisi että Biztalk-integroitipalvelimellamme oli vanhempi versio tarvittavasta ODBC-ajurista. Onneksi oikea ajuri löytyi Oraclen omilta sivuilta. (Code Project 2008).

Biztalk-palvelin ei kuitenkaan edelleenkään saanut yhteyttä Oracle-tietokantaan. Tällä kertaa syylliseksi osoittautui vanha versio Oracle 9i-tietokannan asiakassovelluksesta. Tällä kertaa oikeata uutta versiota ei webin hakukoneiden avulla löytynytäkään. Koko työpäivän kestänyt etsintä Oraclen sivuilta ja Googlen avulla ei tuottanut tulosta. Lopulta sain oikean version asiakassovelluksesta asennettua käyttämällä Oraclen seuraavan sukupolven asennusohjelmaa. Tämän jälkeen palvelimella oli oikeat versiot sekä ODBC-ajurista että Oracle 9i-asiakassovelluksesta ja Biztalk sai viimein yhteyden Oracle-kantaan.

2.6.2 Oracle-tietokantasovittimen ominaisuudet

Microsoftin toimittaman sovittimen ominaisuudet mahdollistavat tietokannan monipuolisen käytön. Sovittimen avulla tietokantaan voi lähettää vapaamuotoisia SQL-kyselyitä ja suodattaa kyselyn tuloksia sekä syöttää ja päivittää taulun sisältöä. Myös tiedon poistaminen taulusta onnistuu. Sovitin mahdollistaa myös moninaisten SQL-lauseiden suorittamisen, kuten esimerkiksi taulun luomisen tai tuhoamisen. Lauseiden suoritusta voidaan ajastaa ja asettaa suoritettavaksi toistuvasti. Myös tietokantafunktioiden käyttö onnistuu Oracle-tietokantasovittimen avulla. (MSDN f 2008)

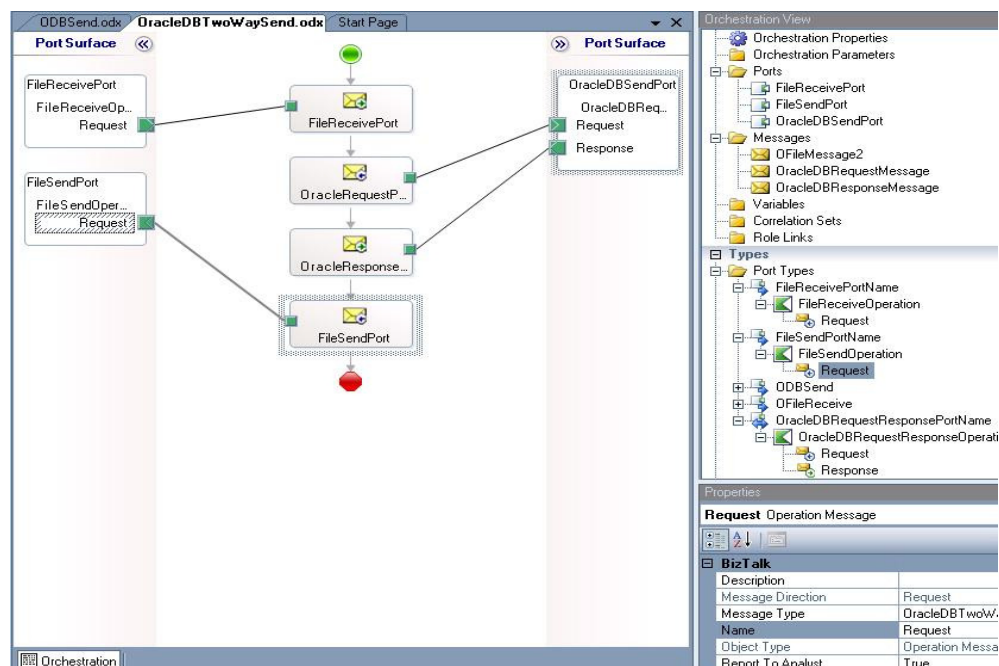
3 BIZTALK-SOVELLUSKEHITYS

Suurin osa Biztalk-integroitsovelluksen kehittämisestä tehdään Visual Studio.Net -ympäristössä. Yleinen ja monille hyvin tuttu ympäristö auttaa tutun käyttöliittymän ansiosta sopeutumaan nopeasti uuteen työkaluun. Toinen integroitsovelluksen toteuttamisessa tarvittava työkalu on Biztalk Administration Console -hallintakonsoli. Sen avulla luodaan

portit, joilla tehdään kytkennät ulkoisiin järjestelmiin. Seuraavaksi kuvataan lyhyesti integrointisovelluksen luomisessa tarvittavia työkaluja.

3.1 Visual Studio.NET -kehitysympäristö

Biztalk-integrointisovelluksen orkestraatiot eli logiikkaytimet ja sen osat luodaan tutussa Visual Studio.Net -ympäristössä. Kuvan 3 mukainen orkestraatio vastaanottaa tiedoston tiedostoadapterilta ja sen jälkeen lähettää SQL-kyselyn Oracle-tietokantaan. Vastauksen tultua tiedosto lähetetään jälleen eteenpäin. Orkestraatio on yksinkertainen esimerkki eikä tiedoston sisältö muutu tai kyselyn tulos vaikuta prosessiin mitenkään. Kuviossa 3 oikealla oleva Orchestration View näyttää kaikki orkestraation komponentit ja niiden tyypit ja Properties-ikkunassa komponenttien ominaisuudet. Siitä näkee, että orkestraatiossa on kolme erityyppistä porttia. FileReceivePort vastaanottaa tiedostoadapterilta tiedoston, FileSendPort lähettää tiedoston eteenpäin. Nämä kaksi ovat Request-portteja. OracleDBSendPort sen sijaan on Request-Response-portti eli saman portin kautta voidaan sekä lähettää kysely että vastaanottaa vastaus tarvitsematta käyttää kahta erillistä porttia.



Kuvio 3. Orkestraatioesimerkki.

Orkestraatio koostuu graafisista työkaluista eli muodoista. Käytettävissä olevia muotoja ovat esimerkiksi lähetys- ja vastaanottoportit ja toisen orkestraation kutsumiseen tai käynnistämiseen tarvittavat välineet. Eräs tärkeimmistä työkaluista on Transform, jolla XML-viesti muunnetaan skeemasta toiseen. Transform-muoto käyttää karttaa eli mapia, jossa muutos määritellään. Transform täytyy aina sijoittaa Construct Message-työkalun sisään. Construct Message siis rakentaa uuden viestityypin orkestraatioissa. Uusi tyyppi muodostetaan joko muuntamalla eli käyttämällä Transform-muotoa, joka määrittää käytettävän kartan, tai muodostamalla tyhjästä kokonaan uuden viestityypin. Tyhjästä uutta viestityyppiä muodostettaessa Construct Message -muodon sisään laitetaan Message Assignment -muoto. Viestinrakennuksessa voidaan käyttää useita Transform- tai Message Assignment -muotoja.

Orkestraatioon luodut virtuaaliset lähetys- ja vastaanottoportit liitetään Biztalk Administration Console -hallintakonsolissa ”fyysisiin” portteihin, joita voi olla useita sekä vastaanotto- että lähetyspuolella. Lisäksi yksittäinen portti voi olla yksi- tai kaksisuuntainen. Kaksisuuntaisuus tarkoittaa sitä, että portti voi sekä tuoda viestin että vastaanottaa vastauksen.

Orkestraatio voi käynnistää myös toisia integrointisovelluksia ja esim. odottaa niiden suorituksen päättymistä tai saada prosessin jatkamiseen tarvittavia tietoja niiltä. Toisaalta yhteen sovellukseen voi sisältyä useita orkestraatioita, jotka toimivat yhdessä. Tämä voi helpottaa prosessin hahmottamista pienemmissä osissa, joilla on kaikilla alku ja loppu. Orkestraatioita suunnittelija voi hahmotella mielensä mukaan. Sovelluksessa voi esim. olla yksi ns. isäntäorkestraatio, jonka alku ja loppu ovat myös koko prosessin alku ja loppu, tai sitten orkestraatioita on vain useita peräkkäin suoritettavana. (MSDN e 2008)

3.2 Biztalk Administration Console

Biztalk-integrointiohjelmistossa hallintakonsoli eli Biztalk Administration Console on integrointisovelluksen toteuttajalle tärkeä työkalu. Sillä luodaan orkestraation portteja vastaavat portit ja määritetään niiden asetukset. Hallintakonsolin portit ovat

integrointisovelluksen rajapinta ulkoisiin järjestelmiin. Sillä myös hallitaan integrointisovelluksia, mm. pysäytetään sovellusten osia. Hallintakonsolin Event Viewer - tapahtumamonitori mahdollistaa vianmäärityksen. Hallintakonsoli näyttää kaikki Biztalk-integrointipalvelimelle asennetut integrointisovellukset. (Technet b 2008)

3.3 Vastaanottoportti

Vastaanottoportti luodaan hallintakonsolilla vastaamaan Visual Studio.Net -ympäristössä luodun orkestraation portteja. Jotta vastaanottoporttia voidaan käyttää, täytyy siihen liittää yksi tai useampi, jossa määritetään viestin vastaanotossa käytettävä sovitin ja määritellään sen asetukset. Toisin sanoen vastaanottoportti kertoo, mistä, ja miten vastaanottoportti vastaanottaa tietoa. Siihen voidaan liittää myös kartta, jolla voidaan muuntaa XML-kielisen viestin rakenne toiseksi. (Technet a 2008)

3.4 Putki

Vastaanottoportissa määritellään käytettävä pipeline eli putki, jolla tehdään tarvittaessa esim. salauksenpurku sisääntulevalle datalle. Biztalk-integrointiohjelmiston hallintakonsoli sisältää muutaman valmiin putkityypin, esim. XML assembler -putken, joka luo tekstimuotoisesta tiedostosta XML-muotoisen viestin ja Passthru-putken, joka välittää datan eteenpäin sellaisenaan.

Putkia voi myös itse rakentaa ja muokata Visual Studio .Net -ympäristössä. Putkessa on neljä mahdollista järjestyksessä suoritettavaa vaihetta. Aluksi voidaan tarvittaessa purkaa salattu viesti. Sen jälkeen viesti voidaan purkaa tai muokata Biztalkin käyttöön Disassemble-moduulissa. Kolmanneksi suoritetaan XML-rakenteen varmennus, mikäli tarpeellista. Viimeisenä vaiheena voidaan tehdä autentikointi viestin lähettäjän identiteetille. (Technet a 2008)

3.5 Skeema

Biztalk-integrointiohjelmistossa tietoa käsitellään XML-kielisenä. Sisään tuleva datan täytyy olla XML-kielistä tai se pitää muuntaa XML-kieliseksi. Jokaista XML-viestityyppiä vastaa skeema, joka luodaan Visual Studio .Net -ympäristössä. Skeemojen luonnissa käytetään ns. velho-työkalua joka vaiheittain ohjatusti luo valitun XML-viestin perusteella sitä vastaavan skeeman. Skeema luodaan siirtämällä kursori object navigatorissa sanan ”solution” päälle ja klikkaamalla hiiren oikeanpuoleista nappia ja valitsemalla ”Add Generated Items”. Velhon suorituksen lopputuloksena Biztalk luo XSD-päätteisen skeema-tiedoston. (Technet a 2008)

3.6 Kartta

Vastaanotetun viestin sisältämää dataa tai rakennetta muokataan kartan avulla. Orkestraatioissa eli logiikkaytimessä luodun Transform-muodon sisältönä on kartta. Kartta luodaan graafisen velhon avustuksella. Velho käynnistyy valitsemalla objektiselaimesta sovelluksen nimi hiiren oikeanpuoleisella painikkeella ja klikkaamalla New Item.

Kartalla muunnetaan data kahden erityyppisen XML-skeeman välillä. Käyttäjä valitsee karttaan lähtöskeeman sekä kohdeskeeman. Skeemojen rakenteita voi tarkastella karttanäkymässä mutta ei muokata. Skeemojen välillä oleva ruudukko on ns. neutraalia aluetta, jolla kaikki muutokset dataan tehdään. Dataa voidaan esimerkiksi muuntaa, lisätä, vähentää, katkoa ja yhdistellä.

Tietoja muokataan graafisten funktioiden avulla, joita on runsaasti. Funktioiden parametrit täytyy itse määritellä. Funktioita on matemaattisia, esim. vähennys, lisäys ja neliöjuuri sekä loogisia funktioita, kuten vertailuoperaattoreita, merkkijono-operaattoreita, aikaan ja päivämäärään liittyviä, erilaisia konversiofunktioita ja tietokantaan liittyviä funktioita. Funktioita voi myös koodata itse mutta valmiilla funktioilla pärjää pitkälle. (Technet a 2008)

4 INTEGROINTISOVELLUKSEN SUUNNITTELU JA TOTEUTUS

Tässä kappaleessa kuvataan tutkintotyön varsinainen käytännön osuus, eli Biztalk-integrointisovelluksen luominen Oracle-tietokannan ja ulkoisen tietolähteen välille. Ensin kuvataan kohdejärjestelmä ja sille tehdyt muutokset sekä lyhyesti ulkoinen tietolähde.

Sitten esitellään lyhyesti sovelluksen suunnittelu. Sen jälkeen selostetaan sovelluksen kehitysprosessin aikana tehdyt toimenpiteet eli käydään läpi toteutuksen vaiheet.

4.1 Kohdejärjestelmä

Järjestelmä, johon integrointisovelluksemme syöttää dataa, on Oraclen tietokanta, versio 9i. Integrointisovelluksen tuomaa dataa varten tietokantaan luotiin taulu, johon data siis syötetään. Tauluun saattaa tulla satoja rivejä yhden päivityskerran aikana. Kantatoteutuksessa on lisäksi taulutriggeri eli skripti, joka suoritetaan ennen jokaisen rivin syöttämistä. Tämä skripti tarkistaa, ettei kantaan syötetä saman sisältöisiä rivejä useampaan kertaan sekä suorittaa datalle viimeisen muunnoksen ennen sen päätymistä tietokantaan.

Kohdejärjestelmän toteutuksen määrittely ja suunnittelu tehtiin valmiiksi ennen integrointisovelluksen toteuttamista. Kohdejärjestelmä asetti siis vaatimukset integrointisovelluksen tuomalle datalle. Sekä tulevan datan rakenne että sen informaatio sisältö pitää olla eheä, jotta tiedon syöttäminen kantaan onnistuu.

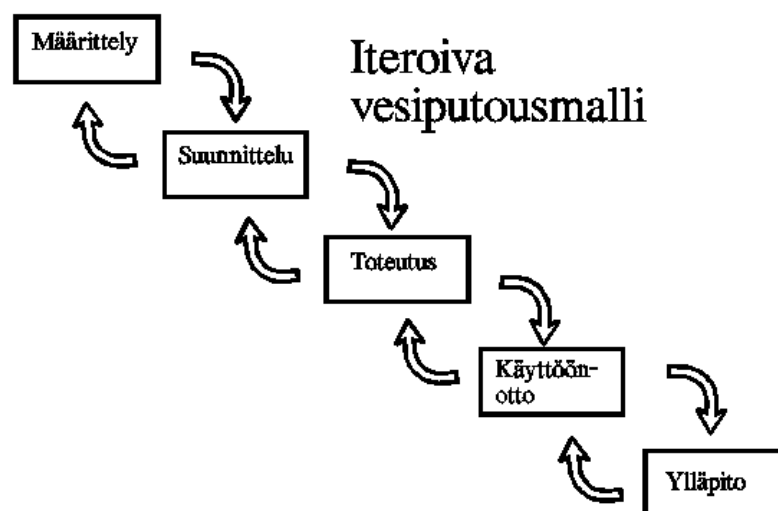
Kohdejärjestelmään toteutettiin myös integrointisovelluksen tuomaa tietoa hyväksi käytävä sovellusosuus Oracle Forms 6i -kehitystyökalulla. Tietokantasovellusta ei käsitellä tässä työssä tämän enempää.

4.2 Ulkoinen liittymä

Kohdejärjestelmän tarvitsema tieto on julkisen instituution tuottamaa. Tieto tallennetaan instituution tietojärjestelmästä XML-tiedostoiksi verkkolevyille, johon päästään käsiksi käyttäjätunnuksen ja salasanan avulla. Tiedon saamisesta käyttöön on tehty sopimus kohdejärjestelmän omistaman yrityksen ja edellä mainitun julkisen instituution välillä.

4.3 Sovelluksen suunnittelu

Lähtökohta suunnittelulle oli selvä. Piti saada aikaan tietovirta ulkoisesta tietolähteestä asiakkaan järjestelmän tietokantaan. Asiakas oli valinnut jo etukäteen sovelluksen toteutuslueksi Biztalk Server -integroitipalvelimen. Projektin aloitusvaiheessa käytännön toteutuksesta ei ollut vielä mitään tietoa, mutta asiakasvaatimusten perusteella toteutettava sovellus oli yksinkertainen. Iteratiivista kokonaisprosessia voisi kuvailla lähinnä kuvion 4 vesiputousmallin mukaiseksi. Välillä jouduttiin palaamaan toteutuksesta takaisin suunnittelupöydälle, ennen kuin taas päästiin jatkamaan yhtä askelta alemmas. Kun opetellaan käyttämään uutta teknologiaa, iteroiva vesiputousmalli tarjoaa riittävästi joustoa uusien asioiden oppimisen ja suunnittelun samanaikaisuuteen. (Kalliala 2008)



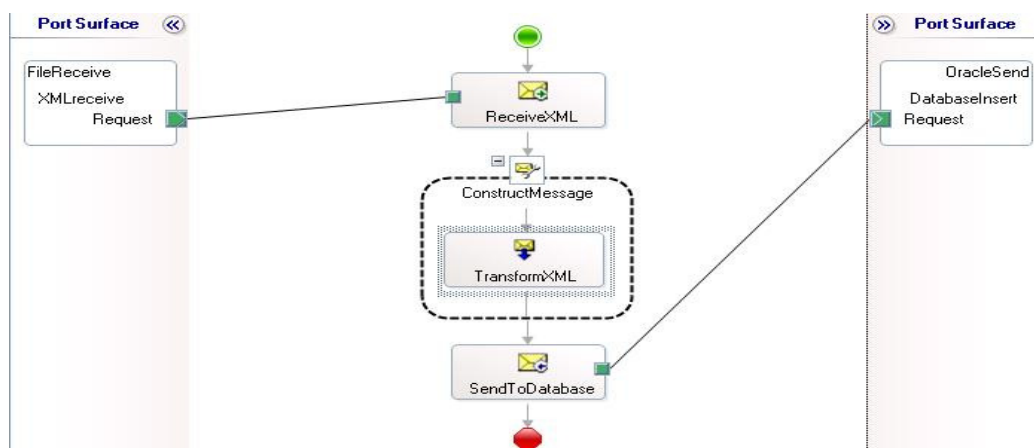
Kuvio 4. Iteroiva vesiputousmalli (Kalliala 2008)

4.4 Sovelluksen toteutus

Toteutettu logiikka hakee tiedostosovittimen avulla vastaanottoporttiin XML-kielisen tiedoston verkkolevyiltä käyttäjätunnuksen ja salasanan takaa, muuntaa sen oikeaan muotoon ja lähettää Oracle-yhteensopivan tietokantasovittimen kautta kohdejärjestelmän tietokantaan. Logiikka toimii ajastetusti ja käynnistyy kymmenen minuutin välein. Toteutettu sovellus on yksinkertainen ja sisältää orkestraation lisäksi kaksi skeemaa ja viestimuunnoksen suorittavan kartan.

4.4.1 Orkestraatio

Melko monimutkaistenkin orkestraatioiden eli logiikkakuvausten luominen on Biztalkin graafisella työkalulla mahdollista. Tässä työssä kuviossa 5 kuvatun orkestraation toteutus oli kuitenkin yksinkertainen. Se sisältää yhden yksisuuntaisen vastaanottoportin, yhden kaksisuuntaisen lähetysportin sekä yksisuuntaisen lähetysportin. Vastaanottoportti käyttää tiedostosovittinta ja molemmat lähetysportit Oracle-tietokantasovittinta. Porttien lisäksi orkestraatioon kuuluu graafinen muoto sekä vastaanotetulle että lähetettävälle viestille. Viimeisenä mukana on muoto viestin muokkaamiselle kohdejärjestelmän tietokantaan sopivaksi. Viestin muokkaava muoto sisältää kartta-komponentin, joka muuntaa ja muokkaa vastaanotetun viestin sisällön kohdejärjestelmän tietokannalle sopivaksi.

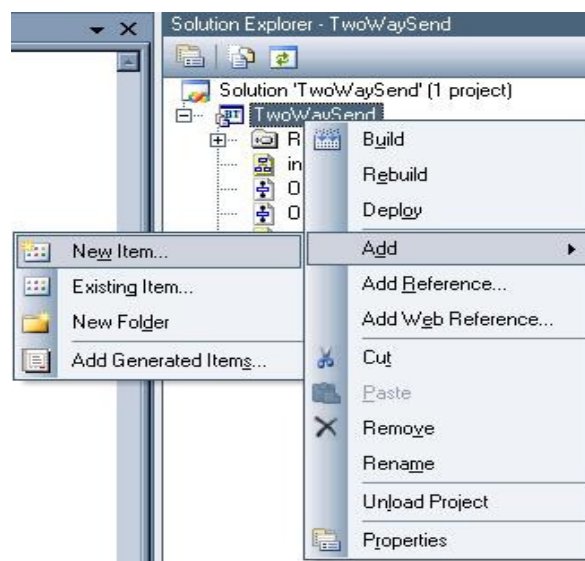


Kuvio 5. Integrointisovelluksen orkestraatio

Vastaanottoportti ottaa tiedoston vastaan ja data talletetaan XML-viestiksi. Seuraavaksi luodaan lähetettävän datan sisältävä viesti muuntamalla se lähtöskeeman mukaiseksi kartan avulla. Viimeiseksi muunnettu data lähetetään kohdejärjestelmän tietokantaan Oracle-tietokantasovittimen insert-lausetta käyttäen. (Technet a 2008)

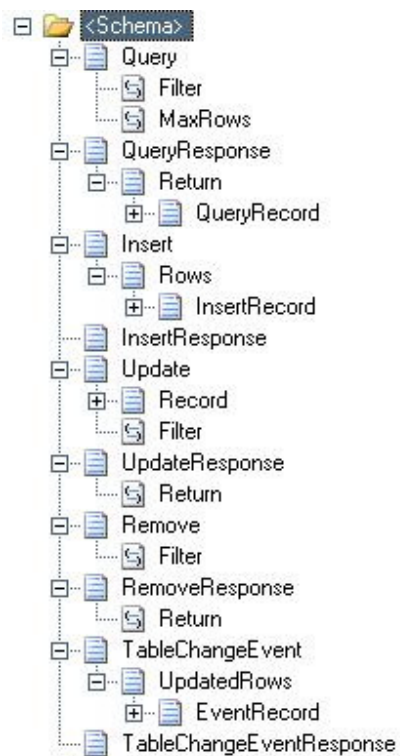
4.4.2 Skeemat

Integrointisovelluksemme tarvitsee kahta skeemaa. Ensimmäinen kuvaa tiedostoadapterilla haettavaa XML-tiedostoa ja toinen kohdejärjestelmän tietokannan taulua, johon data lopulta syötetään. Skeemat luotiin Biztalk-integrointipalvelimen Visual Studio.Net-kehitysympäristössä. Skeeman luonti käynnistyy valitsemalla valikosta kuvion 6 mukaisesti Add Generated Items.



Kuvio 6. Add Generated Items

Käyttäjän apuna toimivan graafisen käyttöliittymän avustuksella molemmat skeemat saatiin mallinnettua. Ulkoisesta liittymästä tulevaa tiedostoa varten luotu skeema mallinnettiin rakenteeltaan täsmälleen samanlaista XML-tiedostoa vasten. Lähtevää dataa vastaava skeema taas mallinnettiin kohdejärjestelmän tietokannan taulua vasten. Kuviossa 7 on kohdejärjestelmän tietokannan taulua vastaavan skeeman rakennekuva.



Kuvio 7. Tietokannan taulua vastaava skeema

Biztalk loi skeemat automaattisesti, kun mallinnettava kohde valittiin. Kun yksi Oracle-tietokantojen yleisistä ongelmista on tietotyyppien yhteensopivuus epäilytti tietokannan taulun yhteensopivuus lähtöskeeman kanssa. Data olisi ongelmatapauksessa jouduttu lähettämään merkkijonopohjaisena ja muuntamaan vasta kohdejärjestelmässä tietokantatriggerin sisällä. Ongelmia ei tämän suhteen kuitenkaan onneksi ilmennyt. (Technet a 2008)

4.4.3 Kartta

Datan rakenteen muuttaminen kohdejärjestelmään sopivaksi vaatii skeeman muunnoksen. Skeema muunnetaan mapin eli kartan avulla. Karttakomponentti luodaan graafisesti Visual Studio.Net -ympäristössä. Integrointisovelluksemme ulkoisesta lähteestä saama data on täsmällistä ja taulukkomuotoista. Vastaanotetun viestin tietosisällöstä muuttuu XML-rakenteen saraketta vastaava tieto kohdejärjestelmän tietokantataulun saraketta vastaavaksi. Myös tietotyyppien täytyy vastata kohdejärjestelmän tietokantataulun tietotyyppettä. Nämä

muunnokset tapahtuvat automaattisesti, kun tieto viedään skeemasta toiseen. Karttatyökalulla siirretään tarvittava data tuloskeemasta lähtöskeemaan piirtämällä osoittimella viiva kahden tietueen väliin. (Technet a 2008)

5 TULOSTEN ARVIOINTI

Työn tarkoituksena oli oppia Biztalk-integrointipalvelimen käyttöä ja dokumentoida opittuja asioita. Oracle-tietokantasovitin oli myös tärkeä saada toimimaan. Työn tuloksena syntynyt pieni integrointisovellus hakee XML-muotoista tietoa ulkoisen liittymän välityksellä ja syöttää saadut tiedot kohdejärjestelmän tietokantaan. Biztalk-integrointipalvelimen käytön oppiminen ja integrointisovelluksen toteuttaminen oli osa projektia, johon sisältyi myös tietokannan yhteydessä toimiva sovelluskerros.

Kohdejärjestelmän sovelluskerroksen käyttäjäkommenttien perusteella voisi arvioida kokonaistoteutusta, jonka osa integrointisovellus siis on. Asiakasyrityksen kohdejärjestelmän sovellusvastaava on ollut omien sanojensa mukaan hyvin tyytyväinen. Tällä perusteella integrointisovellustakin voidaan pitää onnistuneena. Biztalk-sovelluspalvelin on tässä tehtävässä osoittautunut luotettavaksi. Vain muutamia kertoja on haettu data jäänyt kohdejärjestelmän tauluun syöttämättä. Tätä voidaan pitää vähäisenä määränä sovellukselle, jota käytetään jokaisen tunnin sisällä kuusi kertaa.

Henkilökohtaisesti käteen jäi paljon opittua. Iteratiivinen vesiputousmalli auttoi omaksumaan uuden teknologian yksityiskohtia joustavasti. Vaikka Biztalk onkin joiltain osin pitkälle kehittynyt työkalu, vaikuttaa se jotenkin hajanaiselta. Vipuja täytyy väänellä sieltä täältä ja kaikkien toimintojen hallitsemisen oppiminen tuntuu suurelta työltä. Tämä hajanaisuus on tietysti hieman ironista kun kyseessä on toimintojen integroimiseen tähtäävä tuote.

Järjestelmäintegraatioala on kokonaisuudessaan käytännössä niin suuri ettei opittava lopu kesken. Microsoftin lähestymistapa alaan on tullut Biztalk Server -integraatiopalvelimen

kautta nyt jonkun verran tutuksi. Tältä osaamisohjalta on jatkossa entistä helpompi lähteä kehittämään monimutkaisiakin integrointisovelluksia.

LÄHTEET

- Code Project 2008. CodeProject: BizTalk Oracle Adapter Installation.[online]
[viitattu 18.10.2008]
http://www.codeproject.com/KB/biztalk/BizTalk_Oracle_adapter.aspx
- Kalliala 2008. Eija Kalliala: Miksi olioita systeemyöhön. [online] [viitattu 1.11.2008]
http://myy.haaga-helia.fi/~kalei/oliomats/oliot_systyossa.html
- MS Biztalk 2008. Microsoft Biztalk Server. [online] [viitattu 21.10.2008]
<http://www.microsoft.com/biztalk/en/us/default.aspx>
- MSDN a 2008. Biztalk Server Developer Center. [online] [viitattu 26.10.2008]
<http://msdn.microsoft.com/en-us/biztalk/default.aspx>
- MSDN b 2008. Business Rules Engine. [online] [viitattu 26.10.2008]
<http://msdn.microsoft.com/en-us/library/aa561216.aspx>
- MSDN c 2008. Business Activity Monitoring. [online] [viitattu 24.10.2008]
<http://msdn.microsoft.com/en-us/library/aa561137.aspx>
- MSDN d 2008. Biztalk Runtime Architecture. [online] [viitattu 01.11.2008]
<http://msdn.microsoft.com/en-us/library/aa562161.aspx>
- MSDN e 2008. Developing Biztalk Server Applications. [online] [viitattu 29.10.2008]
<http://msdn.microsoft.com/en-us/library/aa577497.aspx>
- MSDN f 2008. What Operations Can be Performed Using the Adapter? [online]
[viitattu 21.10.2008]
<http://msdn.microsoft.com/en-us/library/cc185435.aspx>
- Technet a 2008. Microsoft Biztalk Server R2 Help. [online] [viitattu 05.11.2008]
<http://technet.microsoft.com/en-us/library/aa548004.aspx>
- Technet b 2008. Using the Biztalk Administration Console. [online] [viitattu 05.11.2008]
<http://technet.microsoft.com/en-us/library/aa578089.aspx>
- W3C 2008. The World Wide Web Consortium. [online] [viitattu 18.10.2008]
www.w3c.org