REAL TIME BUSINESS INTELLIGENCE
Införandet av Real Time BI i en SOA-miljö

REAL TIME BUSINESS INTELLIGENCE
Introduction of Real Time BI in a SOA-Environment

Examensarbete i informationsteknologi med inriktning mot informationssystem
Grundnivå 30 Högskolepoäng
Vårterminen 2018

Thomas Malmén

Handledare: Jesper Holgersson
Examinator: Joeri van Laere
**Sammanfattning**

Business Intelligence blir allt viktigare bland företag idag i enighet med att mer och mer data produceras. Den data som produceras är i sig något som kan användas för att analysera företaget i syfte att ta bättre beslut. Behovet är också stort att få ut den informationen i realtid varav även Real Time BI är en trend i sig. Viljan och behovet av att arbeta med Real Time BI är stort varav det kräver att företagen tittar på lösningar kring införandet av det. Många företag har idag en traditionell arkitektur på sin databas och står i startgroparna till att ändra sin arkitektur.

Den här studien tittar närmare på huruvida det är realistiskt att kombinera Real Time BI i en SOA-miljö. SOA är en arkitektur som är tjänsteorienterad och studien har tittat på problematiken kring kombinationen med Real Time BI genom en fallstudie gjord på en organisation. Utifrån den fallstudien går studien ut på följande frågeställning:

"Hur kan Real Time BI realiseras i en SOA-miljö?"

Delfråga: Vilka aspekter kan/kan inte realiseras?

För att få ut så bra svar som möjligt så används kvalitativ metod med tillvägagångssättet intervjuer och dokumentgranskning genom tematisk analys. Det resulterar i en belysning av problematiken samt ett framtagande av vilka aspekter som kan realiseras och inte realiseras.

**Abstract**

Business Intelligence has become increasingly important among companies today, along with more data being produced. The data produced is something that can be used to analyze the company to make better decisions. It is a necessity to get that information in real time and that is why Real Time BI is a trend right now. The will and need to work with Real Time BI today is large, which requires companies to look at solutions to implement it. Many companies today have a traditional database architecture and are in the starting pit to change their architecture.

This study investigates whether it is realistic to combine Real Time BI in a SOA environment. SOA is an architecture that is service oriented, and the study has looked at the problem of the combination of Real Time BI through a case study conducted on a large organization. Based on the case study, the study addresses the following question:

"How can Real Time BI be realized in a SOA environment?"

Sub-topic: What aspects can / cannot be realized?

To get the best answers as possible, qualitative methods are used with the interview and documentary approach through thematic analysis. This results in an illumination of the problem as well as a development of what aspects can be realized and not realized.
Förord

Jag vill tacka min handledare Jesper Holgersson för all den hjälp och vägledning jag fått för att genomföra studien samt tacka Joeri van Laere för den nyttiga feedback jag fått. Jag vill också tacka de personer som ställde upp på intervjuer och gjorde arbetet möjligt att utföra med ett speciellt tack till Gustaf Westerberg som bidragit med mycket av materialet i studien.
1 Introduktion


Ett sätt att nå framgång med sin databas är att bygga den mer tjänsteorienterat där en databas blir flera små tjänster. Tanken med den typen av arkitektur är att varje tjänst äger sin egen information och att tjänsterna är oberoende och isolerade från varandra. Fördelen blir då att de olika tjänsterna kan skräddarsys utifrån deras behov genom exempelvis prestanda eller programmeringsspråk.


Den här studien tar upp hur/om ett företag kan arbeta med Real Time BI i en SOA-miljö (Service Oriented Architecture) samt vilka eventuella aspekter som i så fall kan realiseras. Rapporten tar upp den grundläggande teori som hela studien vilar på samt det material som tagits fram genom en fallstudie. Det avslutas sedan med forskningens resultat och slutsats.
2 Bakgrund

Bakgrundskapitlet kommer att ta upp sådan kunskap som behövs för att skapa en bättre förståelse kring studiens övriga innehåll. Här reds diverse begrepp ut och förklarars för att läsaren ska ha den teoretiska grund som krävs.

2.1 Business Intelligence


Business Intelligence är ett brett uttryck varav det finns en uppsjö av olika varianter baserade på termen. Det handlar i stort om att ta fram information för beslutsfattare. Informationen ska sedan användas som underlag till att ta beslut (Ng, et al., 2013). Underlaget är ofta statistik och viljan är ofta att statistiken ska vara nära verkligheten (Grossmann & Rinderle-Ma, 2015).

2.2 BI 1.0

BI 1.0 kallas också traditionell BI och är den första generationens Business Intelligence. Det är alltså den nivå många organisationer ligger på idag när det gäller BI (Aufaure & Zimányi, 2012). BI 1.0 innefattar det första steget i Business Intelligence och handlar om att effektivisera hanteringen av stora databaser genom strukturerade data i relationsdatabaser. Att lagra data på det här sättet kallas Data Warehouse (DW) och hanteras av en eller flera lagringsservrar. Att den lagras som relationsdatabaser innebär att information som hör ihop kopplas samman vilket i sin tur skapar tabeller (Chaudhuri, et al., 2011). Det är tabellerna som lagras i ett data warehouse uppbyggt enligt BI 1.0.

2.2.1 ETL

Med BI 1.0 kom även andra begrepp såsom ETL (Extraction, transformation load) vilket uppkom då behovet av att konvertera och integrera sin data utvecklades. ETL innebär att data tvättas genom att den sorteras och likställs enligt en viss angiven princip. Dubbletter renas för att förebygga redundans och informationen sorteras så att allt är läsligt enligt fördefinierade villkor. Data började också hanteras genom Scorecards och Dashboards som verktyg för att analysera och visualisera sina databaser (Chen, et al., 2012).

2.3 BI 2.0

Trenden är i full gång mot BI 2.0 varav många företag redan är där och nu står i startgroparna att gå mot BI 3.0 vilket inte kommer att tas upp i den här studien. Utvecklingen ser ut så bland annat för att det som tidigare nämnt finns en stor önskan av beslutsgrundande material i realtid. Utvecklingen ser även ut så för att företagen samlar
på sig mer och mer data varav dem behöver ramverk för hur den informationen ska hanteras på ett bra sätt (Trujillo & Maté, 2012).

BI 2.0 innebär en effektivisering av BI 1.0 genom att ta till olika koncept som i sin tur leder till en ny övergripande arkitektur för företagets databasstruktur.

Trujillo och Maté (2012) har tagit fram sammanlagt nio koncept för att gå mot BI 2.0 varav den här rapporten kommer att beröra följande två:

- **Real-time BI**
  Tidigare har de olika organisationerna tagit fram sin statistik årsvis eller månadsvis för att ta beslut gällande företagets framtid (Azvine, et al., 2005). Det har länge funnits ett starkt intresse bland flera organisationer att ta fram beslutsgrundande rapporter på ett snabbare sätt nära realtid. Traditionell business intelligence (BI 1.0) är inte längre tillräckligt för företagen idag och det ställs stora krav på att ta fram statistiken på ett snabbare och mer effektivt sätt genom det som kallas ”Real-time BI” (Castellanos, et al., 2008).

  Trenden vände för ca 10–15 år sedan på grund av en större konkurrens och påtryckningar från kunder vilket i sin tur har gjort att beslutsfattarna vill ha sina rapporter i realtid (Azvine, et al., 2005).

  Med Real time BI menas att statistiken hämtas så nära verkligheten som möjligt, alltså nästan exakt när det faktiskt händer. Det handlar om att kolla på den strömmande informationen och att genom den kunna mäta mönster och trender i realtid (Castellanos, et al., 2008). Nyttan med att kunna se trender i realtid är att det då också kan göras en förändring snabbare än att enbart arbeta med äldre historiska data.

- **SaaS (Software as a service)**
  Begreppet används ofta av företag som tillhandahåller molntjänster men också när det diskuteras SOA. Det innebär att en tjänst är tillgänglig och brukbar av en användare men att själva tjänsten ligger installerad externt på en server. Om det exempelvis handlar om en applikation så är själva applikationen installerad på extern enhet men nåbar över internet genom HTML för användaren (Verma,

![Figur 2 Förtydligande exempel hur SaaS fungerar i ett Molnföretag gentemot en Kund](image)

Att isolera olika delar i en serverhall på det här sättet effektiviserar varje applikation då dem är helt oberoende av något annat. De isolerade delarna i serverhallen fokuserar därav helt på den specifika applikationen. Fördelen är också att de olika applikationerna kan styras med olika programmeringsspråk och på helt olika sätt, utan att påverkas av varandra.

Real Time BI och SaaS är två skilda begrepp som har gemensamt att dem båda är koncept för företag att hantera vid övergången till BI 2.0 (Trujillo & Maté, 2012).

### 2.3.1 Big Data

All den data som skapas och lagras från olika källor och på olika sätt kallas Big Data. Företag har i många år producerat stora mängder data och trenden kring digitalisering ökar vilket i sin tur gör att datamängden ökar (Mohanty, et al., 2013). Big data innebär mycket stora datamängder som i sig är svåra att hantera för de traditionella verktygen inom en rimlig tidsram. I början av 2000-talet börjades det hanteras Big data genom att trenden kring användandet av BI-verktyg ökar. Trenden med att börja använda BI-verktyg beror till största delen på att det behövdes ett bra sätt att hantera all
information. Organisationers Beslutsfattare kunde nu omvandla den här stora datamängden så att den blev mer läsbar (Mohanty, et al., 2013).

2.4 ODS och SOA

2.4.1 Operational data store

En ODS sparar bara information i realtid och är inte till för att hantera historiska data och ska inte heller vara byggt för att spara historik. Därför fungerar en ODS bra när det behöver arbetas med att få ut information i nära realtid såsom exempelvis för att få ut realtidsrapporter som en analys av nuläget.

![Diagram](image-url)

**Figur 3** En ODS som tar emot information från olika datakällor, vidarekopplat till realtidsrapporter och DW.
I figur 3 visas hur en ODS tar emot information från flera olika system, informationen hanteras genom ETL och skickas sedan vidare till exempelvis ett Datalager eller till ett system som skriver ut realtidsrapporter.

Tabell 1 Skillnaden mellan ODS och Data Warehouse:

<table>
<thead>
<tr>
<th></th>
<th>Operational Data Store</th>
<th>Data Warehouse</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nätverksstatus</td>
<td>Online</td>
<td>Offline</td>
</tr>
<tr>
<td>Innehåll</td>
<td>Strömmande data i realtid</td>
<td>Historiska data</td>
</tr>
<tr>
<td>Uppdateras</td>
<td>Direkt när aktivitet händer</td>
<td>Schemalagt</td>
</tr>
</tbody>
</table>

Tabellen ovan illustrerar skillnaden på Data Warehouse och ODS i syfte att förtydliga hur ODS fungerar.

2.4.2 Service Oriented Architecture


Figur 4 Beskrivning av övergången till SOA. Vänster är klassisk databasarkitektur (BI 1.0) och höger är SOA.


I den högra figuren är samma databas uppdelad i flera olika databaser som i sig är helt isolerade mot varandra. Var och en av databaserna är som ovan nämnt kopplade mot en varsin API. Varje applikation går mot ett gemensamt gränssnitt som genom API:erna hämtar den information som varje applikation är intresserad av. Att isolera diverse olika tjänster gentemot olika isolerade delar i en databas bygger på samma princip som SaaS varav detta uttryck också ofta förekommer när SOA diskuteras.

Arkitekturen SOA är fördelaktig för de företag som vill jobba mer tjänsteorienterat då idén bygger på att effektivisera varje enskild tjänst på bästa möjliga sätt (Anandamurugan & Priyaa, 2014). Det passar alltså de företag som har flera olika tekniska tjänster som allt inte behöver hantera hela företagets databas. Alltså de företag som har flera tjänster varav merparten hanterar olika delar av företagets lagrade information.

2.4.3 Kombinerat med ODS

Som beskrivet i tidigare kapitel så sker all kommunikation med databaserna i en SOA-arkitektur genom tillhörande API. Vid användande av ODS sker kommunikationen direkt med databasen vilket gör att SOA inte är direkt kompatibelt med ODS. Det skulle då behöva göras vissa ändringar i grundreglerna för SOA så att direkt kommunikation är möjlig enbart när det gäller användande av business Intelligence i den arkitekturen (Wu, et al., 2007).

Tabellen nedan visar skillnaderna mellan Traditionell BI-arkitektur, SOA och kombinationen av SOA och ODS för att förtydliga hur dessa fungerar kommunikativt mellan databaser, ihop med Real Time BI och tjänsternas olika prestationer i de olika arkitekturerna.
Tabell 2 visar skillnaden mellan Traditionell arkitektur, SOA och kombinationen SOA och ODS:

<table>
<thead>
<tr>
<th></th>
<th>Traditionell arkitektur (BI 1.0)</th>
<th>SOA</th>
<th>SOA i kombination med ODS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Direkt kommunikation mellan databaser. Genomförbar/Inte genomförbar</td>
<td>Genomförbar genom gränssnitt för databasfrågor.</td>
<td>Inte genomförbar, kommunikation måste gå via API</td>
<td>Genomförbar genom ODS</td>
</tr>
<tr>
<td>Real-Time BI. Genomförbar/Inte genomförbar</td>
<td>Genomförbar, snabb då kommunikation sker direkt i databasen</td>
<td>Genomförbar men komplex och långsam då kommunikation sker mellan API:er</td>
<td>Genomförbar, snabb då kommunikation sker direkt i databasen</td>
</tr>
</tbody>
</table>

2.5 Senaste forskningen


Forskarvärlden har gått ifrån begreppet SOA och istället börjat använda begreppet Mikrotjänster, vilket bygger på samma princip som SOA (Familiar, 2015). Beskrivningen i en artikel är att SOA kommer i flera olika former varav Mikrotjänster anses vara en av formerna (Lewis & Fowler, 2014). Därför kan SOA och Mikrotjänster också likställas då dem bygger på samma grundkonstruktion.

En annan väl refererad artikel tar upp problematiken kring att kombinera mikrotjänster och Real Time BI där författaren tar fram flera egna tänkvärda lösningar på problemet (Davidson, 2016). Det handlar sammanfattningsvis om att använda sig av en ODS som en meddelandekö. Antingen har varje mikrotjänst en egen meddelandekö som den hanterar, alternativt så finns det en central meddelandekö som prenumererar på tjänsternas information. Detta visas i en egen tolkning genom figur 5 nedan.
Själva meddelandekön prenumererar på all information som mikrotjänsterna ger ifrån sig. Mikrotjänsterna ger ifrån sig den information som kan vara intressant för meddelandekön. Detta sker genom en trigger i varje mikrotjänst som så fort något intressant händer i tjänsten så aktiveras en trigger som i sin tur släpper iväg information om händelsen. Informationen plockas upp av meddelandekön och all data i kön kan sedan projiceras på exempelvis en dashboard eller i en rapport.


I artikeln tas det upp att använda sig av en Data Lake arkitektur för att analysera sin data är mer kostnadseffektivt än andra konventionella sätt att lagra data. Det är också
ett mer flexibelt sätt att analysera informationen då det möjliggör analys av olika typer av data, vilket det kan vara i en SOA eller mikrotjänstarkitektur (Stein & Morrison, 2014).
3 Problemområde


Om en organisation väljer SOA som sin databasarkitektur så kan det uppstå problem eftersom det är komplext att kombinera SOA och BI (Abelló & Romero, 2012).

En del i utvecklingen mot BI 2.0 är också att kunna ta fram beslutsgrundande information i realtid (Real-time BI). Det är just nu en trend att utveckla företaget mot möjligheten till Real-time BI vilket också ses som en av de stora utmaningarna för företag idag (Grossmann & Rinderle-Ma, 2015).

Eftersom SOA är ett sätt att gå mot BI 2.0 och Real Time BI är en av komponenterna för att uppnå BI 2.0 så kan dessa behöva kombineras. Ett byte till arkitekturen SOA innebär därför att det kan uppstå komplikationer om organisationen samtidigt vill använda sig av Real Time BI.

Utifrån detta problem har jag tagit fram den här frågeställningen:

**Hur kan Real Time BI realiseras i en SOA-miljö?**

Delfråga:

**Vilka aspekter kan/kan inte realiseras?**


Studien vänder sig till de organisationer som står i begrepp att byta från sin traditionella databasarkitektur till en SOA-miljö och som samtidigt är i behov av Real-Time BI.

Metoden jag har valt är att utföra en fallstudie på en större organisation som är i processen med att byta till en SOA-miljö och som har problem med införandet av Real-time BI i den nya miljön. Fallstudien kommer att bestå av intervjuer med personer som arbetar med processen samt granskande av dokument som beskriver förändringen i text.
3.1 Förväntat resultat

Inför studien gjordes en del förstudier kring vad som kan vara intressant att bygga forskningen kring. Att kombinera SOA med Real Time BI verkade då svårt och eventuellt inte möjligt. Därav var det förväntade resultatet att det kanske inte skulle framgå något lösningsförslag på problematiken. Hypotesen var att belysa problemet med att arbeta tjänsteorienterad inte fungerar ihop med Real Time BI vilket enbart blev en del av det faktiska resultatet.

Hypotesen byggde på en läsning kring att det enbart går att ta fram realtidsrapporter vid integration direkt med databasen. Vid studerandet av forskningslitteraturen vände min hypotes då det framgick att det går att ta fram genom tjänsternas gränssnitt.

3.2 Avgränsning


Real Time BI har valts eftersom problemet fokuserar på kombinationen av Real Time BI i en SOA-miljö. SaaS tas med som aspekt eftersom det beskriver hur en databas delas upp i en serverhall på samma sätt som en databas delas upp enligt SOA.

4 Metod

I det här kapitlet presenteras vilken typ av metod som har valts och utförts för att få ut så mycket som möjligt från studien. Metoden har i sig utgått från frågeställningen som är ”Hur kan Real Time BI realiseras i en SOA-miljö?” med delfrågan ”Vilka aspekter kan/kan inte realiseras?”. Det är en Fallstudie som undersöker huruvida det är realistiskt att arbeta med Real Time BI i en SOA-miljö.

Det har samlats in information från den aktuella organisationen som är i processen att övergå till en SOA-miljö. Informationen är i form av intervjuer och två dokument som beskriver organisationens vision kring bytet till SOA. Intervjuerna tar upp själva problematiken som företaget står inför med att kombinera Real Time BI med SOA. Dokumenten har som huvudsyrte att bekräfta eller styrka det som har kommit fram i intervjuerna.

4.1 Kvalitativ metod


4.2 Fallstudie

4.2.1 Dokumentgranskning
Det har skett en granskning av två dokument som givits av organisationen och dessa har använts för att fylla ut de kunskapssluckor eller den information som inte framgår i intervjuerna. De dokument som tillgivits är:

- Organisationens femåriga vision som bland annat beskriver hur företagets databas ska vara konstruerad år 2023.
- En databasmodell som en av de anställda gjort. Modellen beskriver hur Organisationen kan lösa problemet med kombinationen mellan SOA och Real Time BI.

Dessa har också jämförts med intervjuerna och utvärderats för att se huruvida dessa stämmer överens med varandra eller inte. En analys har därav gjorts enligt kvalitativ metod (Graneheim & Lundman, 2004).

4.2.2 Intervjuer


4.2.3 Dataanalys
Den analysmetod som valts är induktiv tematisk analys genom att det transkriberade materialet har lästs och placerats i olika teman beroende på samtalens ämnen som tas

När ett ämne har hittats i ett samtal så kontrolleras det med de andra intervjuerna samt med dokumenten som tilldelats. De ämnen som anses relevanta till studien och dess frågeställning sammanfattas i materialpresentationen ihop med de ämnen som återkommer i flera källor. Valet att arbeta induktivt bygger på att analysen pågår jämsides under studiens gång för att vara mer tidseffektiv varav induktiv analysmetod är att föredra. Att analysera tematiskt gör att resultatet blir mer innehållsrikt då analysmetoden kräver att materialet läses flera gånger vilket minskar risken att intressant information förgås. I Figur 7 visas visuellt hur Dataanalysen gått till.
4.3 **Etik**

Etiken i studien har utgått från de fyra huvudkraven inom de forskningsetiska principerna (Vetenskapsrådet, 2002). Intervjuerna spelas in och transkriberas därefter och av etiska skäl hålls det samtal innan och efter inspelningarna. Det som samtalas innan inspelningen är kort vad som kommer att gås igenom under intervjun.

- **Informationskravet** - Forskaren berättar om vad forskningen handlar om och beskriver studiens problemställning. Forskaren förklarar att hur mycket intervjupersonen vill svara är upp till denne, att deltagande därmed är helt frivilligt och att samtalet kan avbrytas när som helst under intervjun.
• **Samtyckeskravet** - Efter en genomgång av forskningen och dess syfte så tillfrågas intervjupersonen om denne fortfarande är positiv till att medverka. Säger denne ja så startar inspelningen och intervjun.

• **Konfidentialitetskravet** – Innan intervjun frågas intervjupersonen om denne vill vara anonym. Efter intervjun samtals det om intervjupersonen önskar censurera något i samtalen samt kontrolleras så att intervjupersonen känner sig komfortabla med intervjun som varit. Samtliga intervjupersoner har ingenting emot om dem kan på något sätt identifieras genom studien då ingen av dem önskar vara anonym. Dem har dock förståelse och medgivande med att forskningen inte tar med deras namn.

• **Nyttjandekravet** - Det informeras vidare om att det inspelade materialet kommer att raderas direkt när forskningen är klar samt att materialet endast kommer att användas i den här specifika studien. Det förklaras också att inspelningarna förvaras på forskarens dator under tiden som studien görs, varav intervjupersonerna tillfrågas om detta känns okej. Det understryks också att när studien i sig kommer att läggas ut offentligt och eventuellt kan användas i annan forskning.

Namn anses inte tillföra något till studien varav dessa har valts att tas bort och istället ersätts med personernas arbetstitel eller position på organisationen. Intervjupersonernas arbetstitel skrivs ut då ingen av intervjupersonerna önskar vara anonym eller har någon invändning mot att deras arbetstitel står med i studien.

**4.4 Forskningsprocess**

Här beskrivs hela upplägget på hur metoden har sett ut för att samla in studiens material.

**4.4.1 Intervjuer**

Det har hållits totalt fyra intervjuer med anställda på organisationen som arbetar med själva övergången till SOA och som har god kunskap om organisationens nuläge och framtida vision. Att personerna har god kunskap i organisationens övergång till SOA har säkerställts genom att forskaren efterfrågade specifikt efter personer som är insatta i organisationens övergång till den nya databasarkitekturen. Att det är kompetenta intervjupersoner anses också framgå i intervjuerna.

**4.4.1.1 Miljö**

Intervjuerna utspela sig på den organisation som fallstudien äger rum. Detta var bra eftersom det då är en miljö som intervjupersonerna är vana vid. Känslan var att dem i den miljön kunde hantera frågeställningen på ett avslappnat sätt vilket eventuellt kunde varit annorlunda i en annan miljö.
4.4.1.2 Etik
Innan varje intervju genomgicks de krav som nämns under rubriken "etik" för att säkerställa att samtliga var med på samma villkor. Samtliga gick med på alla krav och det spelade inte någon roll för någon huruvida deras namn publicerades i studien eller inte. Det var ingen som uttryckte något motkrav vare sig före eller efter intervjun.

4.4.1.3 Frågeställning
<table>
<thead>
<tr>
<th>Fråga</th>
<th>Motivering</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vad är din roll på arbetsplatsen? Vad är din uppgift/funktion på avdelningen?</td>
<td>Viktigt för att få fram vem forskaren pratar med och framförallt för att få fram vilken variation det är på intervjupersonerna</td>
</tr>
<tr>
<td>Hur fungerar Organisationens databasstruktur i nuläget? Följdråga: Vad fungerar bra och inte bra?</td>
<td>För att forskaren ska få en klar bild över nuläget. Det anses nödvändigt att få en bra bild över nuläget i enighet med studiens problemformulering</td>
</tr>
<tr>
<td>Vilka generella ändringar ska göras i databasstruktureren?</td>
<td>Det är viktigt att ha ett holistiskt perspektiv i en kvalitativ studie vilket den här frågan hjälper forskaren att få</td>
</tr>
<tr>
<td>Hur ser du på SOA? Följdråga: Vad är SOA enligt dig/Organisationen?</td>
<td>Dels är det intressant att få en personlig syn på SOA men även att få reda på om den överensstämmer med organisationens definition av SOA</td>
</tr>
<tr>
<td>Ser du några utmaningar med förändringen? Följdråga: Ser du någon utmaning med att kombinera SOA och Real Time BI?</td>
<td>Den här frågan fokuserar på problem i stort varav följdrågan är mycket viktig då den går direkt på studiens kärna</td>
</tr>
<tr>
<td>Hur skulle man kunna lösa utmaningarna?</td>
<td>Frågan är viktig då den kan ge forskaren tips och inspiration till ett eventuellt lösningsförslag i studien. Forskaren styr här in intervjupersonen mot kärnproblemet i studien</td>
</tr>
</tbody>
</table>

Samtliga intervjupersoner gick med på att bli inspelade och varje inspelning har transkriberats. Den avsatta tiden för varje intervju var 30 minuter vilket ansågs tillräckligt för det antal frågor som intervjun skulle innehålla. Det visa sig stämma då varje intervju i genomsnitt har en tid på 20 minuter.

Inspelning är bra som metod då det förbygger att ingenting missas av det som sägs under samtalen. Det är också bra att spela in intervjun då det ger intervjuaren möjligheten att ha ett flytande samtal utan avbrott (Patton, 2015). Att istället anteckna det som sägs i intervjun skulle öka risken för avbrott i samtalen vilket kan leda till ett samtal utan flöde.

4.4.2 Dokumentgranskning

De dokument som användes under genomförandet av studien var noga kontrollerade gentemot Organisationen och dess samtycke att använda dessa i studien. De fyra etiska kraven gicks igenom med behörig personal för att kontrollera att samtliga inblandade var införstådda med användandet av dokumenten. Samtliga dokument analyserades när transkriberingen var klar. Processen har sett ut såhär:

- Det genomfördes flera genomgångar av hela texterna och lokalisering av det relevanta gentemot studiens frågeställning.
- De relevanta delarna i visionen delades upp och plockas ut med hänsyn till den kringliggande texten för att få ett bra sammanhang och förståelse.
- Delarna sammanfattas därefter med egna ord.
- Därefter görs en jämförelse med de sammanfattade intervjuerna.
- Sist delas dem in i teman enligt tematisk metod (beskrivs under rubriken "Analys" nedan).

Det togs enbart ut det relevanta i visionen eftersom stora delar av det dokumentet ej berör den här studiens problemställning. Det anses dock att hela modellen i det andra dokumentet är relevant för studien eftersom den handlar om hela problemställningen. Båda dokumenten sammanfattas i materialpresentationen.

4.4.3 Analys

Efter att intervjuerna transkriberats och lästs igenom så utfördes en analys av det transkriberade materialet samt de dokument som beskrivs under rubriken "dokumentgranskning", enligt tematisk induktiv analysmetod. Forskaren läste noggrant igenom den egenformulerade sammanfattningen av visionen med syftet att hitta det som kan styrka intervjuerna, det som eventuellt talar emot intervju materialet samt information som kan fylla ut det som inte framgick under intervjuerna. Lösningsförslaget analyserades och forskaren för samtidigt anteckningar under granskningen av de två dokumenten som därefter sammanfattas i materialpresentationen.
Analysen fungerade som planerat varav de olika delarna i det transkriberade materialet delades upp i olika teman och de olika delarna i dokumenten delades upp i respektive liknande teman. Dessa jämfördes mot varandra och kunde därigenom stärka sin trovärdighet med hjälp av varandra. Illustrationen i figur 8 visar hur det gick till genom att bilden till vänster föreställer intervju materialet och höger de dokument som användes. Figuren visar på hur dessa har ställts emot varandra för att ta fram likheter sinsemellan samt skildra varandras olikheter.
5 Materialpresentation och Analys

Det här kapitlet sammanfattar det material som intervjuerna och dokumenten har tillfört studien. Materialiet analyseras löpande genom presentationen och informationen kopplas ihop enligt flödesprocessen som beskrivs i metodkapitlet.

5.1 Intervjuer

Intervjuer har hållits med fyra personer som i dagsläget arbetar på den aktuella organisationen.

- **Intervjuperson 1**
  Arbetar som domänarkitekt dels i en grupp som arbetar med den övergripande arkitekturen på organisationen varav han arbetar med domänen Business Intelligence. Han är också med projektet Enhetlig statistikhantering där dem arbetar med att ta fram en ny plattform för beslutstöd.

- **Intervjuperson 2**
  Arbetar som Mjukvaruarkitekt där han sitter i ett av utvecklingsteamen som bygger tjänster. Han arbetar med hur man ska ta fram tjänster och med tjänster menas icke grafiska applikationer som levererar någon typ av information.

- **Intervjuperson 3**

- **Intervjuperson 4**
  Är konsult och arbetar som Lösningsarkitekt och objektarkitekt. Beskriver sin roll på organisationen likvärdigt med Intervjuperson 3.

5.1.1 Nuläget

Intervjuperson 1 beskriver att organisationen i nuläget har en traditionell databasstruktur med en relationsdatabas i botten och flera applikationer kopplade direkt till den, vilket kan bekräftas gällande hur forskningen beskriver en traditionell databasstruktur (Mohanty, et al., 2013). Det går också att utläsa dels genom intervjuerna och dels genom Organisationens visionsdokument.

Samtliga intervjupersoner fick frågan om dom tycker att det fungerar i nuläget och samtliga svarade genomgående positivt. De tillfrågade tycker att det fungerar bra i nuläget varav exempelvis intervjuperson 4 säger: ”Ja det funkar och det kan man väl säga att saker och ting funkar ganska bra”. Vidare kommer intervjuperson 4 in på en förklaring till varför det fungerar bra. Han berättar att det år 2015 hände en del som ledde till stora påfrestningar på verksamheten och som dem med den befintliga databasstrukturen klarade av.
Intervjuperson 3 beskriver nuläget på databasen på följande sätt:

"Det är ju en jättet stor relationsdatabas där från början implementerades den med fasta relationer vilket gjorde att när man började utveckla och förändra i den så slog ju foreighn key an och fördela den här förändringen. Och då börja man och lösa upp det här vilket gör att man har lite motstridigt information i vissa fall."

När intervjuperson 2 får frågan om hur nuläget ser ut så svarar han först att "Jo men det funkar, saker fungerar" varav han följer upp sitt svar med:


Läggs svaren ihop från Intervjuperson 2 och 3 så är nuläget att flera obehöriga kan skriva i databasen och att det från början såg annorlunda ut. Att databasen fungerade bättre innan fler fick lov att lägga till information i databasen.

Att vem som helst kan skriva i databasen ökar självfallet risken för att kvalitén blir sämre. Om obehöriga kan lägga till information i databasen så finns det en risk att den läggs till på olika sätt beroende på vem som lägger till den. Detta kan i sin tur skapa redundans eller på annat sätt försämra datakvalitén (Oliveira, et al., 2005). Det är viktigt att kvalitén på den lagrade datan är korrekt i alla lägen så att den inte riskerar att tolkas på olika sätt. Det tas upp av alla att det finns brister i datakvalitén.

Det är mycket viktigt att kvalitén på informationen i en databas är hög för att det i förlängningen ska tas så bra beslut som möjligt (Oliveira, et al., 2005). Utifrån förklaringen av nuläget från intervjupersonerna så kan problematiken kring datakvalitén förklaras med att olika system registrerar olika information vilket kan göra att olika system visar olika innehåll beroende på vad dem hämtar i databasen. Den teorin utgår dels från en tolkning som kommer från intervjun med intervjuperson 3:

"Ett system säger att man kan registrera att jag är förälder till ett barn medans i ett annat system registrerar man det inte. Och det blir motstridigt för när man tittar i det ena systemet så ser man att någon har barn medans det andra systemet säger att personen inte har barn"

Det kan också bekräftas av intervjuperson 1 som menar att datakvalitén inte har med den tekniska lösningen att göra:

"Det finns datakvalitétutsmaningar men det har ju inte med uppbyggnaden att göra utan det har tillåtits att lägga in och registrera information som leder till brister i våran datakvalité. Det är en stor utmaning."
Att det finns brister i datakvalitén bekräftas av Intervjuperson 4 som intygar att det är en av orsakerna till att organisationen vill byta databasesarkitektur ”Nån har säkert pratat om informationskvalité och det ligger i det här beslutet också”.

Detta problem behöver därför inte lösas genom en ny arkitektur utan kan istället lösas genom att någon går igenom samtliga system så att all registrerar och hämtar samma information i databasen. Samtliga intervjupersoner bekräftar också att det fungerar i nuläget när den frågan kommer.


Slutsatsen dras därför att bytet till den nya arkitekturen görs för att rusta företaget för eventuella framtida utmaningar och att företaget i den förändringen också löser problematiken kring datakvalitén eftersom hela databasstrukturen byggs om.

Genom beskrivningen av nuläget och den framtida databasesstrukturen som kom från samtliga intervjuer och dokumenten så kan det också dras en slutsats att förändringen är mycket lik figur 4 och dess beskrivning i kapitel 2.4.2.

5.1.2 Definition av SOA

Flera intervjupersoner ger sin förklaring av hur den nya arkitekturen ska se ut och ger samtidigt en definition av SOA. Själva övergången till SOA förklaras av Intervjuperson 3 som pratar om att bygga tjänster:

”Vi har insett att den här strukturen för hur vi lagrar information idag medför komplexitet i hur vi kan utveckla och vidareutveckla våra olika typer av tjänster så det vi pratar om är att vi tjänstifierar våran domän.”

Vidare berättar Intervjuperson 3 hur han ser på SOA och hur de olika tjänsterna kommer att delas upp: ”Man pratar om mindre informationsdomän tex person, ärende, handling. Att varje blir en egen tjänst”.

I definitionen pratas det om SOA men också om Mikrotjänster varav intervjuperson 3 kommer in på tjänsternas olika storlek:

”Man skulle kunna vidga begreppet och omfatta hela ärendehanteringsprocessen, att det skulle vara en tjänst. Men nu har man valt att göra väldigt små tjänster istället.”

Intervjuperson 2 ser ingen större skillnad på SOA och Mikrotjänster:

Intervjuperson 1 lyfter fram begreppet Mikrotjänster på följande sätt:

“Mikrotjänster är ju att man isolerar tjänsternas lagring från varandra. En tjänst kan gå mot en objektdatabas och en annan kan gå mot fillagring, en tredje tjänst kan gå en relationsdatabas.”

Vidare förklarar han ändå att SOA fungerar på samma sätt som Mikrotjänster:

“Tjänsterna ska vara isolerade mot varandra mellan sina egna lagringsoch man ska inte kunna ”accessa” dem mot något annat gränssnitt än det API som dem tillhandahåller. Och så fungerar ju SOA också egentligen. Det är en tolkningsfråga.”


Huvudfunktionen är dock att arbeta tjänsteorienterat där varje tjänst äger sin egen information och sina egna funktioner. Att döma av de olika intervjupersonernas definitioner så är det övervägande likadana, vilket gör dem enade i målbilden. Det i sig är självfallet positivt både för utvecklarna och för organisationen generellt.

5.1.3 Kombinera SOA och Real Time BI

Samtliga intervjupersoner tycker att kombinationen SOA och Real Time BI är ett problem eller en utmaning varav exempelvis intervjuperson 1 säger: “Problem som kommer uppstå i framtiden är att få ut information i realtidsuppföljning för BI”.


Det som specifiseras kring problematiken är att dem med dagens modell har kontakt direkt med databasen och genom att ställa frågor direkt till databasen får dem nu ut rapporter i realtid. Vidare pratar intervjupersonerna om problematiken kring att SOA inte tillåter direkttkontakt med databasen.
Intervjuperson 2 ser dock inte något större problem med att kombinera SOA och Real Time BI. När han får frågan så svarar han:

"Egentligen ser jag nog inget problem men vi måste ändå ta fram nya lösning jämfört med hur vi har gjort innan"

Vidare pratar Intervjuperson 2 om hur det finns flera olika lösningar och att det därför inte är något problem:

"Idag går vi direkt på databasen och det kommer inte funka. Men sen hur man löser det, det finns ju massor olika lösningar. Att det går att lösa är jag helt övertygad om men man ska bara komma på hur vi tycker att vi vill göra."

Intervjuperson 1 pratar också om hur lösningen dem har idag och hur den inte kommer att fungera i den nya arkitekturen:

"Men det kommer inte att funka på det sättet när vi går till mikroservicearkitekturen. Vi kan inte göra databasintegrationer och vi kan inte göra den typen av lösning utan vi måste ha en annan lösning. Det är ju det som är utmaningen med mikroservicearkitekturen, att vi då kommer att slå isär mycket data och lägga i olika mikrotjänster."

Intervjuperson 3 är även han inne på att det är en utmaning och han förklarar problemet med att SOA inte får ha direkt kontakt med databasen:

"Det är just på grund av på vilket sätt informationen i databasen relaterar till varandra och framförallt hur regelverket för SOA är implementerat. för tanken är ju att varje tjänst ska äga sin egen lagring. Det innebär att den som frågar efter information inte ska veta hur lagringen ska se ut."

Intervjuperson 4 håller med om att SOA och Real Time BI är ett problem men menar också att den nya arkitekturen kommer att lösa ett stort problem dem har idag. Det finns externa program som dem idag måste uppdatera manuellt när något ändras i databasen. Det problemet menar han att dem slipper med SOA:

"En annan anledning till att man inte vill att man går direkt på databasen är ju att om man lägger till ett fält i tjänsten så vil jag ha rätt att ändra hur det ser ut i databasen utan att behöva förhandla med andra program"

5.1.4 Lösningsförslag

En av intervjufrågorna är en fråga om dem själva har tänkt på någon lösning. Samtliga intervjupersoner har funderat på en lösning av problemet och dem är allomfattande inne på samma lösningsförslag på problemet med vissa individuella skillnader. Efter en analys av de olika intervjupersonernas lösningsförslag kan det dock fastslås att dem har
Intervjuperson 2 tar upp två lösningsförslag:

"Det finns ju två vettiga alternativ. Det ena är ju att tänka events så att man får från en tjänst publicerat att någonting händer. Det andra skulle vara att tjänsten ansvarar för att publicera ändringar vilket också kan ske via databasen".

Den ena handlar om att varje mikrotjänst har ett event med en trigger som skickar ut information så fort det är aktivitet. Det andra är att mikrotjänsten själv publicerar informationen om aktiviteten, antingen i en egen databas eller skickar det till en databas. Intervjuperson 1 är inne på lösningen med events och menar att informationen kan läggas på kö:

"Alla tjänster som publicerar något lägger dem på en kö och alla tjänster som är intresserade av förändringar, prenumererar på den kön, och att man i så fall skulle kunna ha en "tapp" som laddar ut alla förändringar till en BI-databas".

Han berättar vidare att det vore en bra lösning eftersom det då går genom tjänsternas API helt enligt reglerna för SOA. Vidare tar Intervjuperson 1 upp ett annat lösningsförslag "En annan möjlig väg att faktiskt tillåta att man går mot tjänsternas databaser". Han går dock in på att det inte är en lösning som bör väljas: "Men den lösningen är ju egentligen ingenting man vill eftersom det går emot tjänstearkitekturens grundprinciper".

Det Intervjuperson 1 har som lösningsförslag går väl ihop med det som bakgrundskapitlet beskriver som Data Lake kopplat till senaste forskningen. Innebörden för "tapp" som Intervjuperson 1 berättar om är en liknelse med en sjö av information där de anställda kan "tappa ur" nutida information från strömmande data. Arkitekturen som fungerar ihop med en sådan Data lake heter "hadoop architecture" och kan i sin struktur hantera flera olika sorters data vilket i teorin ska fungera ihop med Real Time BI (Stein & Morrison, 2014).

Även Intervjuperson 4 är inne på samma lösningsförslag med att alla aktiviteter hamnar i en egen databas: "Man kan ju se till att allt som kommer in "spruttar" man iväg till någon databas som tar hand om det här. Då äger ju tjänsten hur det ska "spruttas" ut."

Intervjuperson 3 diskuterar samma typ av lösning på följande sätt: "man skulle kunna ha det som en orkestrerande tjänst i form av en transaktionslogg så att man genom den kan följa vad det är som händer". Han menar dock att det kan gå trögt när mycket information skickas samtidigt:
"När jag tänker i dem här termerna så hamnar jag i att det kommer bli rätt så "pratigt" när man hämtar informationen, vilket gör att det kommer att ta tid och bli tungt".

Han poängterar då att han därför inte har någon lösning som han tycker är bra.

5.2 Dokumentationen

Här presenteras de delar i visionen som har påverkat studien varav hela organisationens vision inte sammanfattas då det inte anses bidra med något åt läsaren. Texten i dokumentationen kopplas även ihop med materialet från intervjuerna.

5.2.1 Organisationens Vision

Visionen består av ett dokument på 74 sidor varav det som påverkat den här studien och som är relevant kommer att presenteras under den här rubriken. Databasens namn och organisationens namn har bytts ut för anonymitetens skull.

Visionen beskriver nuläget på följande sätt:

"Utmärkande för Organisationens applikationsarkitektur är att vi har många ärendehanteringssystem som alla helt eller delvis använder och bearbetar samma information."

Detta styrker intervjuersonernas beskrivning av databasen som en traditionell databasarkitektur. Visionen beskriver att Organisationen går mot arkitekturen SOA/Microservices och som anledning beskrivs följande:

"En av anledningarna att utveckla enligt SOA är att få bättre ordning på informationstillgångarna och flytta bort integrationen mellan system från Centrala databasen till tjänster"

Texten beskriver också att detta är en del av digitaliseringen av organisationen i syfte att bli "snabbdristigare":

"Att successivt lyfta bort integrationer från Centrala databasen för att skapa en snabbdristigare och mer följsam IT-arkitektur är en nyckelaktivitet i digitaliseringen av Organisationen."

Visionen beskriver sitt nuläge som olika typer av ärendehanteringssystem och stödsystem som går mot en central databas, likt den vänstra delen i "figur 4" i kapitel 2.4.2.

Målbilden som organisationen har beskrivs som olika tjänster som kommunikerar med varandra i dess tillhörande lager:

"Tjänster kommunikerar alltid med tjänster i underliggande lager i arkitekturen, aldrig med tjänster på samma nivå eller ovanför. Vid behov kan man abstrahera ytterligare en nivå inom ett lager för att slippa duplicera funktionalitet, men det bör ses som ett undantag."
5.2.1.1 Beslutsstöd och analys

Visionen kommer här in på behovet av beslutsstöd som i sin tur leder in på behovet av att få ut information i realtid som beslutsunderlag. Organisationen beskriver att behovet av beslutsstöd och analysmaterial är stort:

"Organisationens behov av uppföljning, analys, prediktion, statistik och beslutsstöd är omfattande. Verket sätts ständigt på prov från olika externa intressenter som kräver olika former av statistik. Media och regering kräver alltmer information om Verkets verksamhet."

Visionen tar därefter upp behovet av Realtids BI genom följande beskrivning:

"En tydlig trend inom BI är att verksamheten i högre grad kräver realtiduppdaterade översikter och analysmöjligheter, en trend som även gäller Organisationen. Rapportering och analys på "nära-realtidsdata" är ett komplement till den mer traditionella BI analysen, där fokus ligger mer på långsiktiga trender."

5.2.1.2 Koppling till Intervjuer


Kravet på Real Time BI är något som både visionen och intervjuersonerna framhäver som något att ta med i den nya arkitekturen. Här är dock intervjuersonerna mer tydliga kring att det faktiskt fungerar med realtidrapporteringen i dagsläget. I visionens beskrivning av nuläget tas det ej upp något kring hur realtidsrapporteringen fungerar i dagsläget.

Visionen tar upp att bytet till den nya arkitekturen görs för att vara mer följsam och för att företaget behöver gå ifrån den centrala databasstrukturen. Det håller även intervjuersonerna med om men det är inte det primära som tas upp i intervjuerna. Intervjuersonerna pratar i första hand om att företaget har en bristfällig datakvalité, något som inte nämns i organisationens vision.

5.2.2 Lösning enligt dokumentationen

Eftersom Organisationens vision enbart sätter en generell målbild över hur organisationens IT-miljö ska se ut så presenteras inte där någon lösning på hur dess realtids BI ska kombineras med SOA. Lösningen tas istället upp i det andra dokumentet som tilldelats, vilket är ett lösningsförslag från Intervjuperson 1. Figur 9 är kopierad från det tilldelade dokumentet "BI Microservices 0.3.ppt" varav inga ändringar har gjorts.
Beskrivningar till de olika siffrorna:

1. **Microservice intern lagring**
   - **Produktval:** Valfritt
   - **Lagringsformat:** valfritt
   - **Lagringsmodell:** valfritt
   - **Meddelandeformat:** valfritt
   - **Ansvar:** lagra tjänstens egen datadomän

2. **Entity service**
   - **Produktval:** Java (utvecklingsspråk)
   - **Lagringsformat:** Ingen lagring
   - **Lagringsmodell:** Ingen lagring
   - **Meddelandeformat:** Json, XML
   - **Ansvar:** Entitetsansvar, speglar informationsmodellens entiteter.

3. **Task service**
   - **Produktval:** Java (utvecklingsspråk)
**Lagringsformat:** Ingen lagring  
**Lagringsmodell:** Ingen lagring  
**Meddelandeformat:** Json, XML  
**Ansvar:** Ansvarar för koordinering och programlogik.

4. **Entity service**  
**Produktval:** Java (utvecklingsspråk)  
**Lagringsformat:** Ingen lagring  
**Lagringsmodell:** Ingen lagring  
**Meddelandeformat:** Json, XML  
**Ansvar:** Ansvarar för att lägga samtliga förändrade entiteter (Json/XML) på central meddelandekö.

5. **Central meddelandekö**  
**Produktval:** Rabbit (?)  
**Lagringsformat:** Json, XML  
**Lagringsmodell:** RAM + fil  
**Meddelandeformat:** Json, XML  
**Ansvar:** FIFO-kö. Garanterar att samtliga meddelanden ligger i rätt ordning i kö. Raderar meddelanden från kö efter ackning från lyssnare. Garanterar att meddelanden kan tas emot i korrekt ordning.

6. **ODS-lyssnare**  
**Produktval:** Java  
**Lagringsformat:** ingen lagring  
**Lagringsmodell:** ingen lagring  
**Meddelandeformat:** Json, XML in, SQL ut  

7. **ODS**  
**Produktval:** Postgres  
**Lagringsformat:** SQL databas  
**Lagringsmodell:** SQL-tabell (speglar info-modell)  
**Meddelandeformat:** inga meddelanden  
**Ansvar:** Lagrar alla entitetsförändringar. Fungerar som logg för alla förändringar som sker i systemen (för de meddelanden som ska in i ODS). Ansvarar för att möjliggöra realtids-rapportering, sammanställningar. Är källa för inkrementell laddning till datalager.
8.

**Datalager**

**Produktval:** Postgres  
**Lagringsformat:** SQL databas  
**Lagringsmodell:** SQL-tabell  
**Meddelandeformat:** inga meddelanden

**Ansvar:** Långtidslagring av historiska förändringar. Ansvarar för att upprätthålla enterprise data modell (speglar informationsmodellen) samt rapporteringsmodell för statistik och uppföljning. Data uppdateras från ODS dygnvis.
6 Slutsats

Detta kapitel tar upp studiens slutsats varav svaret på problemställningen besvaras utifrån den forskning som gjorts.

Arbetets problemställning innefattar huruvida det är realiserbart att kombinera Real Time BI i en SOA miljö med följdfrågan om vilka aspekter som kan respektive inte kan realiseras. För att få svar på dessa frågor så har studien utgått från en fallstudie gjord på en Organisation.

Poängen med SOA är att bygga om den nuvarande databasen till flera olika tjänster som är helt självgående. Med självgående menas att varje tjänst tar hand om sig själv och sin egen information. En tjänst kan exempelvis vara ett visst program som hanterar en viss del i databasen varav den delen då byggs in i tjänsten.

Fördelen eller vinsten med att arbeta tjänsteorienterat är att de olika tjänsterna är oberoende av varann varav om en tjänst lägger nere så kan fortfarande de övriga tjänsterna fungera felfritt. Att tjänsternas är oberoende av varandra är positivt då dem kan se helt olika ut i prestanda eller programmeringsspråk.

*Hur kan Real Time BI realiseras i en SOA-miljö?*

Det är möjligt att arbeta med Real Time BI i en SOA-miljö enligt den här studien. Viktigt är att ta hänsyn till de regler som SOA innebär varav en primär regel att ta hänsyn till är att all information måste gå genom tjänsternas eget gränssnitt. Reglerna som framgår i forskningen kring SOA är väldigt generella och måste därför anpassas till organisationen som arkitekturen ska införas på.

För att svara på frågan så kan SOA och Real Time BI realiseras genom att använda sig av en central meddelandekök som varje tjänst skickar information till. Varje tjänst ska därmed ha en trigger som automatiskt skickar information om olika förändringar som meddelandekön kan vara intresserad av. Meddelandekön prenumererar på den information som är relevant för realtidsrapportering. Informationen skickas sedan vidare till en ODS där Organisationen kan hämta ut data i realtid.

*Vilka aspekter kan/kan inte realiseras?*

De aspekter som inte kan eller inte bör realiseras är att arbeta med Real Time BI på traditionellt vis genom att interagera direkt med organisationens databas eller databaser. Det kan såklart realiseras men då mister organisationen en av SOA:s grundprinciper och det kan då istället vara aktuellt att titta på en annan mer passande arkitektur.

Ska en organisation ha en fungerande SOA-arkitektur så bör den utgå från att hämta informationen genom de olika tjänsternas gränssnitt. Det är alltså realistiskt att kombinera SOA med Real Time BI men aspekten som bör beaktas är kravet på att all kommunikation med tjänsterna ska gå via dess tillhörande gränssnitt (API).
7 Diskussion

7.1 Arbetsprocess

Här diskuterar jag min arbetsprocess med utgångspunkt från den forskningsprocess som beskrivs i studiens metodkapitel.


Intervjuerna gick bra och intervjuersonernas var mycket behjälpliga under intervjuerna. Den generella tiden varje intervju tog var i genomsnitt 20 minuter varav större delen av intervjuerna blev till bra material för studien. Detta gjorde att jag höll min förutsatta tidsram på 30 minuter för varje intervju.

Det blev dock en del prat om annat varav jag gjorde valet att transkribera det irrelevanta för att följa min metod som gick ut på att transkribera allt. Efter Intervjuerna transkriberades materialet vilket tog mycket längre tid än väntat. Det i sin tur gjorde att jag fick höja mitt tempo något för att hinna bli klar med samtliga delar i arbetsprocessen. Samtidigt som jag höll intervjuer erhöll jag också de dokument som skulle ingå i studien.

Att utföra analysen tematiskt var något nytt för mig och därför mycket utmanande och svårt att hantera för första gången. Tematisk analys är mycket tidskrävande eftersom allt material ska läsas om och om igen. Det är dock även mycket effektivt och jag kommer att använda mig av samma metod igen om tillfälle ges.


Jag tycker att de olika materialen som jag fått ut av studien kompletterar varandra på ett tillräckligt bra sätt för att besvara frågan i min problemställning. Intervjuerna står för den mera personliga synvinkeln och dokumenten för de generellt framtagna bestämmelserna. Jag tycker dock att visionen är lite kort i sin beskrivning över hur och varför realtidsrapporteringen och den nya arkitekturen ska implementeras. När jag läser visionen får jag mest en generell bild som inte så mycket går in på förklaringar till
varför det ska göras. Jag får en känsla av att dem till största delen gör förändringen för att företaget vill följa en trend mer än att förändringen görs för att lösa problem.

Jag tror att läsaren kan ha synpunkter på materialets trovärdighet eftersom jag enbart har fyra intervjuer och två dokument. Det kan anses vara för lite material för att kunna dra någon poäng eller slutsats för egen del. Personligen anser jag materialet vara tillräckligt för att bidra med något till forskningen samt för att studien ska kännas tillräcklig. Men självfallet vore det intressant om en likadan studie gjordes på samma företag med egenskapen att eventuellt kunna styrka trovärdigheten i resultatet.

7.2 Studiens Metod

Den metod som valdes var att intervjua anställda på organisationen och jämföra deras svar med vad deras officiella dokument säger. Genom kontakt med en anställd på organisationen togs det fram fyra personer som arbetar med omställningen till SOA. Fyra personer kan anses vara för få men eftersom dem är så pålästa kring projektet så ansågs det inte behövas fler synpunkter. Speciellt inte eftersom det även fanns dokument som kunde styrka det som framkom i intervjuerna.

Hanteringen av materialet var mycket mer tidskrävande än väntat. Transkriberingen tog mycket längre tid än väntat och därav var det antagligen till min fördel att jag inte hade fler än fyra intervjuersoners svar att gå igenom. Genomgången av de tilldelade dokumenten tog också oväntat lång tid med tyngdpunkt på organisationens vision som var mycket längre än väntat.

Fallstudien är gjord på en Organisation och valet gjordes att göra både företaget och de anställda anonyma för att andra företag lättare skulle kunna ta till sig min forskning. Organisationen och de anställda har själva inte bett om anonymitet och dem står för allt som sägs i intervjuerna samt texten i dokumenten. Det gör därför ingenting om någon på något sätt kan lista ut vilka personer som deltagit samt vilken organisation det är eftersom informationen inte är hemlig.

7.3 Etik

Utifrån etiska aspekter tycker jag att studien har gått bra och följt de riktlinjer jag från början satte upp i min metod. Dessa punkter bygger på de råd som ges av vetenskapsrådet vilket är Informationskravet, Samtyckeskravet, Konfidentialkravet och Nyttjandekravet. Innan och efter varje intervju gick jag noggrant igenom de olika etiska punkterna med varje person och var öppen för deras synpunkter och eventuella krav på intervjun. Jag var också noga med min förklaring kring dokumenten och huruvida dessa skulle användas i studien. Ibland kom jag och intervjuersonerna in på sidospår under intervjun varav dilemma kom ifall jag skulle transkribera detta eller ej. Valet gjordes då att transkribera allt men att inte citera det som var irrelevant för studien.
7.4 Samhälleliga aspekter

Med utgångspunkt i samhälleliga aspekter tycker jag att studien kan hjälpa organisationer med att se vilka eventuella hinder och problem som kan uppstå vid bytet av databasarkitektur. Organisationer som står i läget att byta arkitektur kan också känn igen sig i problemen och genom detta se på hur den här fallstudien har valt att se på problematiken. Jag ser också studien som ett bidrag för andra företag som kan använda den som vägledning i sitt eget byte av databasarkitektur samt vilka krav dem kan behöva ställa vid implementationen.

Detta kan ge direktiv kring vilka problem man kan stöta på i syfte att undvika dessa. Organisationerna kan genom den här studien ställa sig frågan om det faktiskt är nödvändigt att byta arkitektur eller om det fungerar bra redan som det är.

Jag ser också studien som ett bra redskap i att utbilda sin personal i eviken av Business Intelligence och primärt Real Time BI. Studien kan användas i samråd med personalen på verksamheten för att ta reda på vilka kunskaper som redan finns samt om det behöver fyllas ut med ny kunskap om BI.

7.5 Vetenskapliga aspekter

Vid sökandet av studier, böcker och artiklar om att kombinera SOA med Real Time BI så var resultatet svagt. Jag tycker att det saknas forskning kring den kombinationen och då framförallt nutida forskning. Därför bidrar den här studien till problemet med kombinationen SOA och Real Time BI generellt.


7.5.1 Senaste forskningen

Samtliga intervjupersoner snuddar vid den senaste forskningen då dom diskuterar Mikrotjänster lika mycket som SOA. Intervjuperson 1 är den som mest är inne på senaste forskningen varav denne nämner lösningsförslag med teorin Data Lake. Även det lösningsförslag som det ena dokumentet beskriver är mycket likt den senaste forskningens lösningar på problemet. Det är bara Organisationens vision som inte hanterar den senaste forskningen på något sätt.

Den senaste forskningen fokuserar väldigt mycket på att SOA och Mikrotjänster är två skilda delar varav kombinationen Mikrotjänster och Real Time BI kombineras istället för SOA och Real Time BI. Den här forskningen kan därför bidra till att visa likheterna mellan SOA och Mikrotjänster och därigenom skildra de samhörigheter dessa har gentemot problemställningen.
Den senaste forskningen fokuserar också mycket på lösningar med molntjänster varav den Organisation jag har arbetat med, valt att inte ansluta sig till molntjänster. Detta för att verksamheten hanterar mycket känslig information och dem anser att det är för riskabelt att lägga den informationen på moln. Därför kan andra organisationer med samma problematik lättare ta till sig den här studiens resultat.

Som nämnt så saknas forskning kring kombinationen SOA och Real Time BI. Detta nämner jag här i diskussionen samt i bakgrundskapitlet. Istället så väljer många att använda begreppet Mikrotjänster vilket när jag läser om det, verkar vara samma princip enligt flera källor. För att spekulera så tror jag att man som forskare vill använda ett nytt begrepp för att teorin ska födas på nytt och inte förknippas med gammal teori och gammal forskning. På flera ställen ser jag meningen ”Microservices is SOA done right” vilket ändå tyder på att teorierna liknar varandra.

Den senaste forskningen hanterar kombinationen Mikrotjänster och Real Time BI med det som jag i bakgrunden beskriver som Data Lake. Det är en teori som ska fungera då företaget Netflix använder sig av en sådan kombination (Watson, et al., 2015). Netflix tas upp som ständigt exempel när man läser om Mikrotjänster och enligt den blogg jag här refererar till så använder dem sig också av verktyg för att analysera sin data i realtid.

Jag har valt att referera till nästan enbart artiklar på internet när jag skriver om den senaste forskningen just för att dessa är väldigt nära i tiden. Jag tror att om man som forskare vill presentera den senaste forskningen så är det viktigt att titta på artiklar som är skrivna så nära nutid som möjligt varav studier ofta är gjorda under en lång tid och blir därför inte lika ”dagsfärska”. Detta kan dock dra ner min studies trovärdighet varv jag också tar upp att informationen inte helt ska hanteras som sann fakta. Den senaste forskningen är sällan väl kontrollerad eftersom den är så pass färdig och inte hunnit hanteras av så många andra forskare. Jag har dock läst samtliga referenser och vill mena att informationen som tas upp i artiklarna har en bra förankring i tidigare studier.

7.6 Framtida forskning

Den här studien har en övergripande inriktning och har helt hoppat över de mer tekniska delarna i problematiken. Ur teknisk synvinkel skulle det kunna vara intressant att titta på vilket programmeringspråk som passar bäst när man ska kombinera SOA med Real Time BI. Det kan också vara intressant att utföra min studie på en annan typ av organisation såsom ett litet företag eller ett privat företag där det eventuellt kan finnas andra krav, problem eller delproblem.

En annan intressant utgångspunkt vore att utveckla något på den här studien som en påbyggnadsstudie. Att bygga en prototyp för hur SOA kombinerat med Real Time BI fungerar rent tekniskt vore intressant eftersom min studie enbart är teorier. Intervjuperson 1 tar i sitt lösningsförslag upp termen ”Data Lake” som jag bara kort tar upp i bakgrundskapitlet. Hade studien haft mer arbetstid så hade det varit intressant att göra en fördjupning kring hur Data Lake skulle kunna vara en lösning på problematiken.
Det kan också vara av intresse att göra en likadan studie men på ett annat företag för att i se likheter och olikheter.
Referenser


[Använd 1 Juni 2018].

