

**OPEN SOURCE VERKTYGS STÖD FÖR
CENTRALA EGENSKAPER HOS BUSINESS
PROCESS MANAGEMENT (BPM) SYSTEM**

**OPEN SOURCE TOOLS' SUPPORT FOR
CENTRAL FEATURES IN BUSINESS
PROCESS MANAGEMENT (BPM) SYSTEMS**

**Examensarbete inom huvudområdet
Informationssystemutveckling**

Grundnivå nivå 30 Högskolepoäng

Vårtermin 2018

Namn: Manuel Mazanga Baiao

Handledare: Jonas Gamalielsson

Examinator: Eva Söderström

Sammanfattning

I en komplex informationsteknologisk värld behöver företag och organisationer flexibla affärssystem som följer den tekniska utvecklingen. Av denna anledning har kommersiella BPMS-verktyg (Business Process Management System) blivit populära på grund av sin höga förmåga att snabbt integreras med affärssystemen i nya arbetsmiljöer. Ett alternativ till kommersiella BPMS-verktyg är de OSS-baserade (Open Source Software) BPMS-verktygen med öppen källkod som är tillgänglig för allmänheten att vidareutveckla. På grund av att de OSS-baserade BPMS-verktygen ofta är unga på marknaden och behöver vidareutvecklas blev syftet med denna uppsats att genomföra en granskning av de open source-baserade BPMS-verktygen vad gäller deras skilda egenskaper. En sådan granskning ökar förståelsen för hur verktygen fungerar och underlättar därför vidareutvecklingen av dem. Genom att utgå från en etablerad analysmodell av Delgado et al. (2015) genomfördes en tematisk innehållsanalys för att jämföra egenskaperna i dessa verktyg. Närmare bestämt analyserades insamlade dokument som beskriver BPMS-verktygens egenskaper. Den vägen jämfördes 6 olika BPMS-verktyg utifrån 13 egenskaper. Analysresultatet visade att Bonitasoft, jBPM och JPM5 var de mest välutvecklade BPMS-verktygen som gav stöd för flest egenskaper. Resultatet visade även att verktyg med färre egenskaper ändå kunde vara ett bra alternativ för vissa företag, beroende på vilka egenskaper och funktionaliteter som prioriteras inom företaget.

Nyckelord

Business Process Management (BPM-verktyg), egenskaper, företag, öppen källkod.

Abstract

In a world of developing complex information technologies, companies and organizations need flexible and updated business systems that and match these technological developments. For this reason, commercial BPMS (Business Process Management System) tools have become popular because of their high ability to quickly integrate with business systems in new work environments. An alternative to commercial BPMS tools are the OSS-based (Open Source Software) open source BPMS tools that are available for the public to develop. Since the OSS-based BPMS tools are often new on the market and need further development, the purpose of this paper was to conduct an analysis of open source-based BPMS tools with respect to their various features. Such an analysis increases the understanding of the tools' functionality and therefore facilitates further development of them. Based on an established analysis model by Delgado et al. (2015), a thematic option for some companies, depending on the characteristics and functionalities that are prioritized within the company. Content analysis was conducted to compare the characteristics of these tools. More specifically, collected documents were analyzed describing the properties of BPMS tools. That way, 6 different BPMS tools were compared based on 13 properties. The analysis showed that Bonitasoft, jBPM and JBPM5 were the most well-developed BPMS tools that supported most of the features. In addition, the analysis demonstrated that tools with fewer features could still be a good option for some companies, depending on the characteristics and functionalities that are prioritized within the company.

Keywords

Business Process Management (BPM Tools), Characteristics, Business, Open Source

Förord

Först vill jag tacka Institutionen för informationsteknologi vid högskolan i Skövde som har möjliggjort att skriva denna uppsats. Jag vill även passa på att tacka alla lärare som har uppmuntrat mig och även gett mig den motivation som jag behövde för att klara detta arbete. Jag vill tacka min examinator Eva Söderström som har bidragit till detta arbete. Framförallt vill jag rikta ett stort tack till min handledare och förebild Jonas Gamalielsson som helhjärtat funnits där för mig med sin feedback. Jag hade inte kunnat önskat mig en bättre handledare. Jag vill inte heller glömma att rikta ett tack till hela min familj och mina vänner som alltid trott på mig och stött mig under arbetets gång.

INNEHÅLLSFÖRTECKNING

1	INLEDNING	1
	Arbetets disposition	1
2	BUSINESS PROCESS MANAGEMENT (BPM)	3
2.1	Definition av BPM	3
2.2	BPMS (Business Process Management System)	4
2.3	Konceptuell utveckling av Business Process Management (BPM)	5
2.4	Business Process Management (BPM) livscykel	6
2.5	BPMN (Business Process Management Notation)	7
3	OPEN SOURCE SOFTWARE (OSS)	10
3.1	Historisk kontext	10
3.2	Karakterisering av olika licenser	11
4	PROBLEMFÖRMULERING	13
4.1	Syfte	15
4.2	Frågeställning	15
4.3	Avgränsningar	15
4.4	Förväntat resultat	15
5	METOD	16
5.1	Datainsamling	16
5.2	Urval	17
5.3	Dataanalys	19
5.4	Validitet och reliabilitet	20
5.5	Forskningsetiska principer	21
6	MATERIALPRESENTATION/EMPIRI	22

6.1	BPMS-arkitektur (eng. BPMS-architecture)	22
6.2	Integration med sociala nätverk (eng. Integration with social networks)	23
6.3	Modelleringsstyp (eng. Modeler type)	24
6.4	Stödspråk (eng. Support languages)	25
6.5	Processövervakning (eng. Process monitoring)	25
6.6	Installationspaket (eng. Installation packages)	26
6.7	Tid på marknaden (eng. Time in market)	27
6.8	Community-aktivitet (eng. Community activity)	28
6.9	Stöd för mobila enheter (eng. Support for mobil devices)	29
6.10	Stöd för arbetsflödesmönster (eng. Support of workflow patterns)	30
6.11	Integritet och dokumentssäkerhet (eng. Integrity and document security)	31
6.12	Sökfunktioner (eng. Searching mechanisms)	32
6.13	Tillståndsmekanism för användare (eng. Permission Mechanism for users)	33
6.14	Dokument för granskning av BPM-verktygens egenskaper	34
7	RESULTAT OCH ANALYS	35
8	DISKUSSION	39
8.1	Resultatdiskussion	39
8.2	Om kontextens betydelse	40
8.3	Metoddiskussion	41
8.4	Samhälleliga aspekter	42
8.5	Vetenskapliga aspekter	43
8.6	Etiska aspekter och framtida studier	43
	REFERENSER	44
	BILAGA 1-DOKUMENT ANVÄNDA FÖR GRANSKNING AV BPM-VERKTYGENS EGENSKAPER	49

1 Inledning

I den dagliga informationsteknologiska verksamheten genomför organisationer kontinuerliga förbättringar för att uppnå sina mål (Delgado, 2015). I denna informationsteknologiska värld av ökande komplexitet har företag och organisationer utvecklat och utvärderat ett antal digitala affärssystem (Ling & Xin, 2009). Vanligtvis har dessa affärssystem utvecklats oberoende av varandra, vilket gjort att de skiljer sig från varandra, bland annat vad gäller teknisk kompatibilitet, funktioner och användargränssnitt. På grund av denna brist på samordning mellan dessa olika system har det skapats "öar av information" som försvårar företagens digitala rutiner. För att lösa denna brist på integration behöver företagen utveckla affärssystem som är kompatibla med varandra och snabbt kan svara på förändrad efterfrågan och teknisk utveckling (ibid., 2009).

På grund av dessa inkompatibla "öar av information" undviker många företag att implementera traditionella affärssystem och fokuserar istället på BPMS-verktyg (Thompson, Seymou, & O'Donovan, 2009). BPMS (Business Process Management System) är nämligen mer flexibelt än många andra affärssystem, och denna flexibilitet underlättar den kontinuerliga utvecklingen av organisationens verksamhet (ibid.). BPMS flexibilitet ligger i dess användande av flödesmodeller och processkontroller i programvaran, vilket tillåter företagen att på ett enkelt sätt integrera systemet i arbetsmiljön (Ryan et al., 2009).

Av denna anledning är företagens efterfrågan på BPMS mycket stort (Liang, 2008). På grund av att det finns många variationer av BPMS har många företag svårt att välja det mest lämpliga BPMS-verktyget (ibid.). Exempelvis har olika BPMS-verktyg skilda egenskaper och dessutom är vissa versioner av BPMS inte så fullständiga som de borde vara (Weber, 2009). BPMS kan även baseras på Open Source (OSS), vilket betyder att programvaran baseras på en källkod som är öppen och tillgänglig för allmänheten att vidareutveckla. Open Source är alltså motsatsen till så kallad "proprietär programvara", där åtkomst till källkoden är begränsad till enstaka ägare av källkoden.

BPMS kan alltså baseras på OSS och har en hög potential vad gäller installation, funktion och integration i företagens affärssystem. Samtidigt är OSS-baserade BPMS-system ofta unga och behöver vidareutvecklas. Det motiverar en djupare granskning av open source-baserade BPMS, framförallt vad gäller funktionaliteten i skilda versioner av BPMS, och hur dessa stödjer företagens affärssystem.

Arbetets disposition

Studien är uppdelad i ett antal kapitel. I inledningskapitlet motiveras varför BPMS är värt att undersökas. I andra kapitlet presenteras BPMS-verktygens utveckling, funktionaliteter, livscyklar och definitioner, där ett grundläggande teoretiskt ramverk utvecklas. Det tredje kapitlet presenterar historiken bakom OSS och visar hur denna öppna källkod förvandlar BPMS till ett kraftfullt verktyg. I kapitel fyra presenteras

uppsatsens syfte, avgränsningar och de frågeställningar som ska besvaras. I uppsatsens metodkapitel (kapitel fem) demonstreras hur uppsatsens frågeställningar ska besvaras genom en kvalitativ innehållsanalys (litteraturstudie) som jämför olika BPMS-verktyg. Dessutom diskuteras metodens relevans för att besvara uppsatsens frågeställningar. I kapitel sex presenteras empirin från utvärderingarna av BPMS och i kapitel sju analyseras resultaten i förhållande till tidigare forskning. I kapitel åtta diskuteras slutsatserna från analysen utifrån ett större perspektiv, och ger förslag på framtida forskning.

2 Business Process Management (BPM)

I detta avsnitt definieras samt presenteras Business Process Management System och dess utveckling.

2.1 Definition av BPM

BPM är ett operativt organisationssystem som utgår från ett systematiskt, strukturerat tillvägagångssätt för att modellera, utveckla, implementera, genomföra och utvärdera affärsprocesser i en organisation. Kortfattat kan BPM därför definieras som ett omfattande system för hantering och omvandling av organisatoriska operationer (Hammer, 2015).

BPM innefattar de metoder, verktyg, tekniker (programvara) och icke-tekniker (människor, organisationer, applikationer, dokument och andra informationskällor) som är nödvändiga för att nå organisationens mål genom anpassade affärsprocesser (Delgado, 2015). I dessa affärsprocesser styrs och övervakas de individer, organisationer, applikationer och den information som finns tillgänglig i eller utanför organisationen (Bernroider, 2008).

BPMS kan även betraktas som en uppsättning tekniker för kontinuerlig samt interaktiv förbättring av processer för en organisation (OMG, 2010). Mer exakt kan BPM betraktas som uppnåendet av målen för en organisation genom förbättring, hantering samt kontroll av viktiga affärsprocesser (Jeston & Nelis 2008).

BPM är alltså en modern teknik som med hjälp av programvara stöder affärsprocesser för att specificera, kontrollera, genomföra samt analysera affärsprocesser som involverar människor, företag, applikationer, dokument och andra informationskällor. För närvarande tenderar BPM att fokusera på organisationsprocesser samt de funktioner som ökar mervärdet för företagets kunder (Aalst, Hofstede & Weske, 2003). Närmare bestämt tenderar organisationer att definiera värdet av BPM affärsprocesser genom mervärdet det skapar för kunderna, och utifrån hur hanteringen av dessa affärsprocesser skapar sunda organisatoriska rutiner som leder till effektiva processer och hög ekonomisk avkastning inom företaget (ibid.). Utifrån dessa utgångspunkter bör BPM-metoder innefatta följande steg: att förstå företagsbehovet, att förstå processen, analys och processmodellering, utformning av ny process, granskning och utvärdering av den nya processen och metodiken (Adesola & Baines, 2005). När en organisation kan hantera hela denna processcykel förenklas samarbetet mellan anställda och kunder genom smidiga informationsprocesser (ibid.).

Genom att tillämpa BPM kan organisationen ta sig en mer komplex uppgift genom att övervaka affärsprocesserna på ett snabbt och kostnadseffektivt sätt som är enkelt för organisationens anställda att förstå sig på (Smith & Fingar, 2007). BPM kan vidare utveckla företagets befintliga affärsprocess genom att testa ett stort antal variationer av systemet och den vägen skapa förbättringar eller nya versioner av BPMS (ibid.).

BPM täcker inte bara affärsprocessen som sker genom människor, maskiner, uppgifter och system, men inkluderar även beskrivningar av kostnad, resursförbrukning samt tidsåtgång för varje producerad enhet. Ett sådant system tillåter en bättre simulering av möjliga scenarier, vilket genererar bättre mätvärden i utvärderingsprocessen. På det sättet jämförs scenarierna, och resultatet skickas till företagets IT-enhet som skapar ett automatiserat informationssystem (Smith & Fingar, 2007).

Flertalet kritiska faktorer avgör om det nya systemet blir framgångsrikt, exempelvis förmågan att anpassa systemet till organisationens strategi och personal med varierade kompetenser, samt förmågan att sätta samman det anpassade systemet till en enhetlig struktur (Baldam et al., 2009). Ett sådant anpassat system lägger sig som en väv över företagets avdelningar och sträcker sig utanför organisationsgränserna, hela vägen ut till kunder och leverantörer (Jeston & Nellis, 2008). I en snabbt växande och konkurrensutsatt globaliserad värld utvecklar företag nya strategier för att överleva. I och med att även företagskunder efterfrågar nya tekniska lösningar förvandlas BPM till en potentiell lösning. Anledningen är att BPM-systemet snabbt kan anpassas till nya komplexa affärssystem och organisationer och ändå upprätthålla en samstämmighet i hela systemet och processen. Denna förmåga ökar givetvis organisationens mervärde.

2.2 BPMS (Business Process Management System)

Ett BPMS-system kan betraktas som en uppsättning verktyg vilka förbättrar processhanteringen, i synnerhet genom ökad automatisering av komplexa uppgifter, såsom integration av äldre system med det nya BPMS-systemet (Verner, 2004). Genom sin flexibilitet kan BPMS även uppgraderas kontinuerligt och förändra flödena i de definierade processerna, och upprätthåller därmed organisationens dynamiska affärsmiljö (ibid.). Exempelvis gör BPMS-systemet det möjligt för organisationer att modellera, distribuera samt hantera kritiska processer, vilka kan distribueras över flera företagsapplikationer, företagsavdelningar samt koordineras med affärspartners (Smith & Fingar, 2007). I detta avseende fungerar BPMS som ett processnav mellan människor och system, som definierar arbetsflödet och information mellan dessa deltagare.

BPMS-processerna kan även klassificeras utifrån komplexitet. Det som definierar processens komplexitet är dess livslängd samt hanteringsbehov (nuvarande processtatus och innehåll). En kortvarig process kräver inte implementation, men en process som involverar mänskliga deltagare (hög mängd information) kan kräva implementation (Chang, 2006).

BPMS-system gör det möjligt för företagen att mäta samt standardisera processer såsom organisering av återanvändbara processer som kan ordnas i nätverk. Vidare kan BPMS-system tillåta koordination av verksamheter samt system i process (Chang, 2006) vilket skapar följande möjligheter:

- Öka engagemanget mellan företag och IT-team i affärsprocessdesign.
- Integrera individer och system som deltar i en affärsprocess.

- Simulera affärsprocesser för processoptimering för implementering.
- Övervaka, kontrollera samt förbättra affärsprocesser i realtid.
- Införliva ändringar i befintliga affärsprocesser utan komplicerade omarbetningar.

Ovanstående fördelar med BPMS blir möjliga genom användandet av en arsenal av processer som integreras i den traditionella IT-arkitekturen. Vid tidpunkten för BPR (Business Process Reengineering) var ERP (Enterprise Resource Planing) ett tillgängligt IT-system. Eftersom affärsprocesserna i dessa system var inbyggda i systemet blev det svårt att genomföra enkla modifieringar. De processer som ingår i det mer flexibla BPMS-systemen möjliggör däremot en mer direkt tillgång till IT-applikationer, vilket garanterar flexibiliteten och utvecklingen i affärsprocesserna (Chang, 2006).

2.3 Konceptuell utveckling av Business Process Management (BPM)

För att definiera samt förstå Business Process Management (BPM) som affärssystem kan det vara nödvändigt att betrakta företag som en uppsättning av processer (Mulle, 2003). Processer kan här betraktas som aktiviteter utförda i en logisk sekvens som resulterar i en värdefull tjänst (Hammer & Champy, 1994). Dessa processer kan även beskrivas som systematiskt strukturerade aktiviteter där "input" förvandlas till en mer värdefull "output" för den specifika kunden (Devenport, 1994). Genom process management kan organisationen för en lägre kostnad genomföra sådana processer på ett snabbare och mer flexibelt sätt, samt med högre precision (Hammer, 2010).

Historiskt sett växte denna processhantering fram i tre vågor. Den första vågen kan kallas "total kvalitet", eller "Total Quality Management" (TQM) som det kallas på engelska (Smith & Fingar, 2007). Denna våg initierades på 50-talet. TQM etablerades genom ISO (Internationella Standardiseringsorganisationen) som fastställer standarder för exempelvis kvalitetsstyrningssystem. Den andra vågen inträffade under 90-talet (Hammer & Champy, 1994). Denna våg kan kallas "process reengineering" och innebar en omformning av arbetsrätten som bidrog till omfattande förbättringar av företagets produktionsprocesser, vilka skapade nya organisationsstrukturer som optimerade organisationernas resultat (ibid). Den tredje vågen växte fram genom nya former av digitala affärsprocesser som baserades på ett gemensamt processspråk. Det möjliggör en starkt förbättrad systemintegration och gemensam standard (Smith & Fingar, 2007). En jämförelse mellan de tre metoderna presenteras i tabell 1.

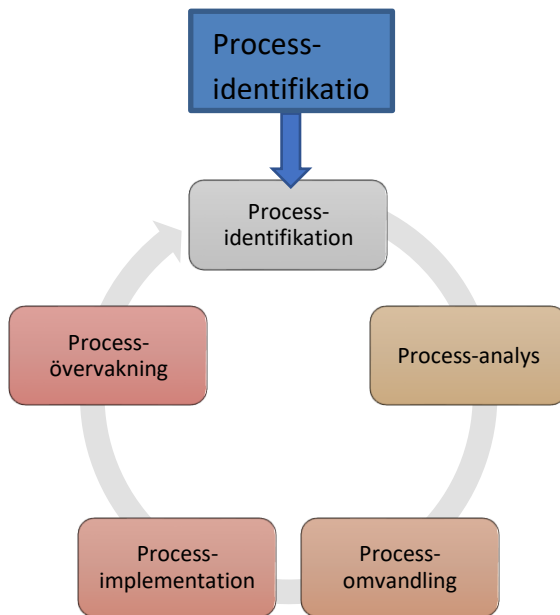
Tabell 1 Tre former av process- eller management-modeller som varit aktuella sedan 50-talet. Källa: Smith och Fingar (2007)

Jämförelsesfaktor	Total Quality Management (TQM)	Innovationsprocess (reengineering)	BPMS
Förändringsnivå	Inkrementell	Radikal	Komplett processcykel
Tolkning av nuvarande och framtida process	Aktuell process förbättras i nya versioner	Gammal process och generations process ny-diskontinuitet	Utan implementation av BPM/BPM implementerad
Utgångspunkt	Befintliga processer	En White board genererar nya idéer	Ny process/befintliga processer
Tid som krävs	Kort	Lång	Beror på processen och tillvägagångssättet
Omfång	Strait, inuti funktioner/avdelningar	Omfattande, korsning av funktioner/avdelning	Bred, inkluderar samtliga processer
Risk	Måttlig	Hög	Låg
Inblandning	Specialister i Industri	Generalister i affärssystem	Teknik som inkluderar alla processer och anställda

Ovanstående tabell visar hur BPMS är ett lågrisksystem som baseras på ett enhetligt processspråk och involverar hela organisationer och processcyklar.

2.4 Business Process Management (BPM) livscykel

Ett stort antal BPM-modeller kan tillämpas på ett företag. De flesta modeller antar ett cykliskt format och betraktas utifrån en livscykel. En klassisk modell delar upp genomförandet i fem faser: organisering inför uppgraderingen, förståelse för processerna, förbättring, mätning, samt kontroll och kontinuerlig uppgradering (Harrington, 1993). En alternativ modell innehåller sju faser: förberedelse, organisatorisk struktur, konfiguration, prioritering, beskrivning, analys, förbättring samt standardisering (Muller, 2003). En av de mer moderna BPM-modellerna baseras på en livscykel i åtta steg: Upptäckt, modellering, distribution, exekvering, interaktion, kontroll, optimering samt processanalys (Smith & Fingar, 2007). En alternativ modell föreslås i nedanstående figur.



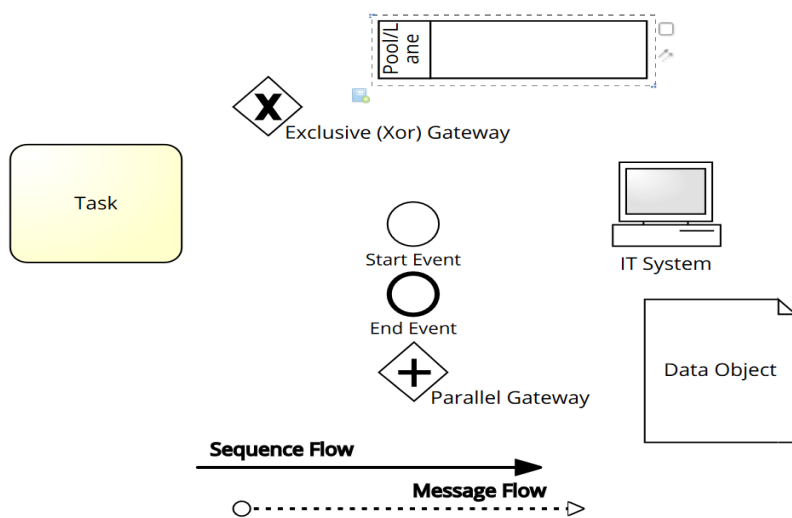
Figur 1 Installation av BPM-modell utifrån ett cykliskt perspektiv. (Baserad) på: Dumas, La Rosa, Mendling & Reijers (2013)

Ovanstående figur visar hur följande faser kan ingå i en BPM-cykel:

- Process-identifikation. I den här fasen existerar ett problem med affärssystemet. För att åtgärda problemet identifieras och avgränsas de processer som är involverade i problemet.
- Process-analys. I denna fas analyseras de problem som identifierats. Om möjligt blir problemen även dokumenterade samt åtgärdade.
- Process-omvandling. Målet med denna fas är att identifiera förändringar i processen som skulle hjälpa till att lösa de problem som identifierats i den föregående fasen.
- Process-implementation. I denna fas implementeras de förändringar som krävs för önskad måluppfyllelse.
- Process-övervakning och kontroll. När den uppgraderade processen är igång samlas relaterade data in och analyseras för att bestämma hur väl processen infördes med avseende på prestationsåtgärder samt prestationsmål.

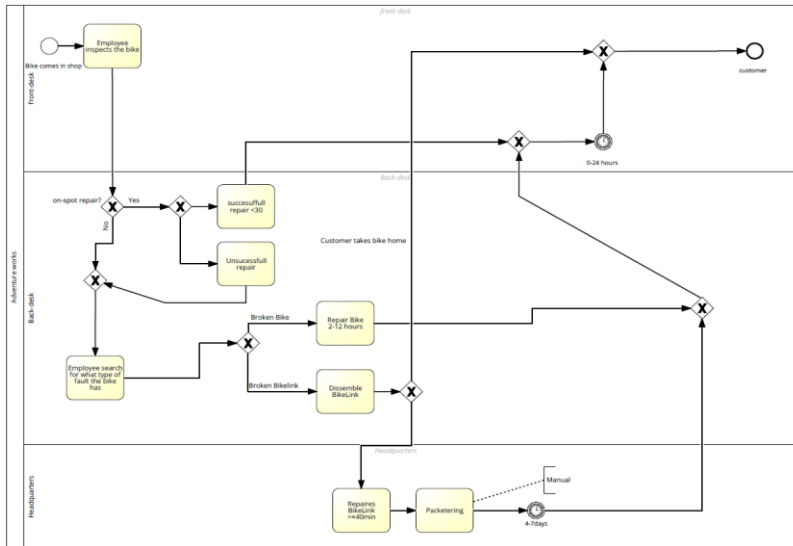
2.5 BPMN (Business Process Management Notation)

BPMN är notationer och symboler som visar hur BPM är strukturerat. Dessa notationer består bland annat av en serie standardikoner vilka underlättar användarens förståelse av systemet. Modellering är viktigt steg i automationsprocessen, eftersom modelleringen identifierar processernas delar. Det är även genom modelleringen som en del av förändringen kan genomföras i syfte att uppnå optimering. Notationer kan användas i denna modellering av processarkitekturen. BPMN utvecklades genom Business Process Management Initiativ (BPMI) och underhålls för närvarande av Object Management Group (OMG). Syftet med nuvarande version av BPMN (BPMN 2.0) är alltså att underlätta användningen av BPM genom att representera processer genom grafiska bilder. Genom dessa notifieringar kan affärsprocessen betraktas i sitt nuvarande tillstånd ("as is") samt i sitt kommande tillstånd ("to be") efter att processen förändrats.



Figur 2 Centrala aspekter av BPMN och dess symboler. Källa: Författarens egen konstruktion.

Som framgår av ovanstående figur består BPMN av en uppsättning grafiska element som baseras på ett format som är bekant för de flesta modellerare (White, 2004). De fyra grundläggande kategorierna av element är flödesobjekt, anslutningar, SIM-banor och artefakter. Flödesobjekt är de mest grundläggande grafiska elementen och definierar processens flöde eller beteende. Det finns tre flödesobjekt: event, aktiviteter samt "Gateways". Flödesobjekt som symboliserar flödet mellan objekt består även de av tre symboler: flödessekvens, meddelandeflöde samt association (ibid.). Det finns två sätt att gruppera elementen i modelleringen genom SIM-banor: genom pooler samt körfält. Artefakter används för att lägga till information till flödet av processen, ett exempel är standardartefakter. Modelleringsverktyg kan användas för att lägga till hur många artefakter som helst. Försök har genomförts för att standardisera en uppsättning artefakter för allmän användning. Denna uppsättning artefakter inkluderar dataobjekt, grupper samt annotationer (ibid.).



Figur 3 Exempel på en BPMN-applikation och dess processer (författarens eget exempel).

Ovanstående figur baseras på grafiska element som påminner om de element som utvecklare använder, med exempelvis rektanglar som symboliserar aktiviteter och diamanter som symboliserar beslut. Huvudidén med BPMN är att skapa en tydlig bild av verksamhetsmodellen (White, 2004). Genom noteringarna blir det lättare att på ett enkelt sätt belysa komplexa variationer i processen.

3 Open Source Software (OSS)

Syftet med detta kapitel är att presentera egenskaperna i open source samt illustrera de internet-communities som har utvecklats i samband med OSS under de senaste decennierna. Dessa communities består av privata och professionella aktörer som deltar i gemensamma diskussioner och samarbeten, vanligtvis on-line, för att förbättra verktyget. Några får betalt, men de flesta deltar på helt frivillig och ideell basis. I korthet kan poängteras att open source är programvara som är befriad från individuell äganderätt och därför gör det möjligt för frivilliga programutvecklare att utveckla mjukvaran (Feller & Fitzgerald, 2002). Denna utveckling brukar ske i gemensamma internet-communities som utvecklar programvaran tillsammans med den ursprungliga upphovspersonen. Genom denna frihet från äganderättsliga begränsningar kan sådana datorprogram bidra till att ett flertal individer kontinuerligt uppgraderar och vidareutvecklar programmen. Därigenom skapas samarbeten mellan programmerare, som utvecklar och anpassar datorprogrammen till nya omvärldsbehov (ibid.).

3.1 Historisk kontext

För att förstå Open Source-fenomenet måste vi gå tillbaka till de tekniska kulturer som skapade denna typ av programvara. Ursprunget till öppen källkod går tillbaka till en hacker-kultur som baserades på obegränsad delning av programvara mellan forskare vid universitet under 60-talet (Perens, 1999). När datorer kom till den akademiska miljön fick programmerarna betalt för sitt arbete och inte för själva mjukvaran. Senare etablerades begreppet "proprietär programvara" i form av en immateriell äganderätt som blockerade utomståendes åtkomst till källkoden, om kunden inte betalat en avgift för att få tillgång till källkoden eller programvaran.

Flera händelser bidrog till utveckling av denna kultur. Inte minst lanseringen av ARPANET 1969, som var det första transkontinentala nätverket av datorer, och skapandet av operationssystemet Unix samt programmeringsspråket C bidrog till open source-kulturen (Raymond, 1999). ARPANET bidrog till en oöverträffad ökning i informationsutbyte som löpte i takt med tekniska framsteg, där flertalet forskningscentra började samverka, exempelvis Artificiell Intelligens (AI) från MIT (Massachusetts Institute of Technology) och Stanford universitet.

80-talet markerade början av epoken med proprietär programvara. Unix blev en kommersiell produkt, licensierad av AT & T sedan 1984, och konkurrerade följande år med flertalet varianter av systemet, bland annat Berkeley Software Distribution (känd som Berkeley Unix eller BSD) och SunOs, distribuerad av Sun Microsystems. Som reaktion på denna trend lanserade Richard Stallman, forskare vid MIT, projektet GNU år 1984, med målet att bygga ett Unix-baserat operativsystem från grunden, bestående av helt icke-proprietär programvara utan äganderättsliga begränsningar. Genom detta initiativ skapades FSF (Free Software Foundation) året därpå.

Begreppet "fri programvara" baseras på öppen källkod. Enligt Free Software Foundation (2009) bör denna fria programvara tillåta fyra friheter:

- 1) Frihet att använda programvaran för all användning.
- 2) Frihet att studera programvarans funktion samt anpassa den till individers behov (för detta behövs tillgång till källkoden).
- 3) Frihet att distribuera kopior av programvaran.
- 4) Fri tillgång för allmänheten (inklusive källkod).

Som juridiskt stöd för detta koncept utvecklade Stallman (1999) en specifik licens till GNU/GPL (General Public License), vilken skyddar upphovsrätten för den distribuerade programvaran, samtidigt som den tillåter kopiering, distribution eller modifiering (Free Software Foundation, 2007). Denna licens begränsar inte kommersiell användning av programvara, exempelvis avgifter för nedladdning. Licensen fastställer dock restriktion av licensiering av verk som härrör från den ursprungliga källkoden, genom att det nya arbetet måste vara licensierat i linje med originalverket. Denna begränsning kallas "Copyleft" och målet är att förhindra att fri programvara integreras eller omvandlas till proprietär programvara (Stallman, 1999).

3.2 Karakterisering av olika licenser

Under 90-talet bevittnade människor hur Internet integrerades med massmedia genom World Wide Web samt genom en starkt ökad tillgång på internetleverantörer (Raymond, 1999). I början av decenniet startade Linus Torvalds utvecklingen av Linux, ett kärnoperativsystem för system Unix, och gjorde den tillgänglig på Internet enligt GPL. Linux utveckling under 90-talet är nära kopplad till den ökade spridningen av Internet, vilket gjorde det möjligt att samordna arbetet bland flertalet projekt och spridas över hela världen i en oöverträffad takt och dimension (ibid.).

Under andra hälften av decenniet gav rivaliteten mellan Netscape och Microsoft upphov till Netscapes beslut att skapa öppen källkod på sin navigator. Denna strategi, motiverad av fördelarna som erhållits genom fri tillgång till källkod, gav upphov till skapandet av Open Source Initiative (OSI) år 1998 (Raymond, 1999). Utan att vara oense med Stallmans (1999) idéer var det tydligt för OSI-medlemmar att det fanns negativa uppfattningar kring begreppet "fri programvara" såväl som FSF-aktiviteter. Genom dessa missuppfattningar om betydelsen i "fri programvara", samt genom FSF:s utsatthet för motstånd mot immateriella rättigheter, blev detta koncept oattraktivt i näringslivet. Beteckningen "öppen källkod" uppstod då som ett pragmatiskt sätt att främja den tekniska och ekonomiska utvecklingen inom mjukvaruutveckling, vilket lockade investerare och företag till sig, i kontinuitet med Netscape (Open Source Initiative, 2006).

Motiverad av framförallt Stallmans (1999) grundarbete med "öppen källkod" etablerade OSI ett dokument med tio kriterier som fastställer programvarulicens som öppen källkod (Open Source Initiative, 2006). Detta dokument visar att öppen källkod betyder mer än bara tillgång till källkoden. Exempelvis möjliggör öppen källkod fri omarbetning och

modifiering av programvara, samtidigt som den ursprungliga upphovsrätten säkerställs. Till skillnad från proprietär programvara är de rättigheter som etablerats genom öppen källkod därför inte begränsade till en specifik produkt (ibid.).

GPL är en öppen källkodlicens. Samtidigt finns numera dussintals licenser som uppfyller OSI-kriterierna. Vissa är "copyleft" licenser som liknar GPL, medan andra etablerat licens av typen "permissive", vilket möjliggör öppen källkod med kommersiella anslag. Licenser som "Apache Licence", "Artistic Licence", "Mozilla Public License" och "BSD License" tillåter integration av öppen källkods derivatverk i proprietär programvara, vilket begränsar åtkomsten till källkoden (Perens, 1999). Nedanstående tabell visar exempel på programvara som distribueras under sådana öppna licenser.

Tabell 2 Exempel på Open Source programvaror med öppna licenser, fördelad utifrån kategorier. Källa: Perens (1999).

Kategori	Exempel	Använd Licens
OperativSystem	Distributioner Linux: Red Hat, Debian, SUSE Linux, etc.	GPL, etc.
DatabasSystem	Mysql, PostgreSQL	GPL respektive BSD
Programmering språk	Perl, PHP	Artistic licens och GNU GPL respektive PHP licens
Webbserver	Apache http server	Apache
Webbläsare	Mozilla Firefox	Mozilla Public License, etc.
Office-applikationer	Openoffice.org	lessor, GPL
Bildbehandlingsprogram	GIMP	GPL

4 Problemformulering

Sedan 80-talet har processhanteringen av digitala affärssystem varit omdiskuterad i branschen samt akademien, främst inom forskningen relaterad till IT-administration och produktionsteknik, och studier har påvisat en kontinuerlig tillväxt av globala BPM-initiativ (Ko, Lee & Lee, 2009). Men trots stora initiativ och investeringar har konstaterats att sju av tio förbättringsprojekt misslyckats (Grover, 1999).

BPM är även starkt beroende av vilka för och nackdelar dessa verktyg har för att modellera eller omvandla skriftliga processer till BPMN (Business Process Management Notation). Vissa företag använder befintliga BPM-system för dessa ändamål, men de rapporterar att dessa verktyg erbjuder lite eller inget aktiv stöd för dessa ändamål (Hammer, 2015). Dessutom har cirka 60–80 procent av BPM-verktygen misslyckats på grund av bristfälligt stöd från toppledningen, låg eller ingen BPM-utbildning för slutanvändarna samt brist på kommunikation och interdepartementalt samarbete (Trkman, 2010). Det finns även en brist på kunskap om hur open source-verktyg stöder centrala egenskaper i BPM-systemen (ibid.). Att hitta den kritiska framgångsfaktor som stöder dessa centrala egenskaper bör därför förbättra företagens BPM-verktygen (Jeston & Nelis, 2008).

För organisationer eller företag som önskar modellera eller etablera affärsprocesser, finns det dock BPM-verktyg med mycket funktionella attribut. Historiskt sett har det framförallt gällt de proprietära BPM-verktygen. De open source-baserade BPM-verktygen har däremot varit problematiska för användaren, eftersom användaren sällan vet hur dessa verktyg stöder företagets övriga företagsapplikationer i BPM-miljöer. Dessutom har ofta felaktigheter och ineffektiva funktioner upptäckts i denna OSS-baserade BPM-lösning.

Kvantitativt baserad forskning säger delvis emot ovanstående påståenden. Det har även visat sig att OSS-baserade BPM-verktyg redan för 12 år sedan kunde ta upp kampen med kommersiell proprietär programvara vad gäller funktionalitet, installation, kompatibilitet och säkerhet (Delgado et al., 2015). Proprietära BPM-verktyg har alltså fått konkurrens av OSS-baserade BPM-verktyg under senare år.

Eftersom open source ofta kan erbjuda kontinuerlig uppgradering, nya funktioner, lägre konsumentkostnader och stort stöd från omfattande programmerings-communities blir frågan om de OSS-baserade verktygen kan komplettera eller ersätta de proprietära verktygen avgörande för användarna. En nackdel med OSS-verktygen kan dock vara vissa brister i säkerhet, kompatibilitet med delar av BPM-systemet och andra oklarheter som kan uppstå i open source-sammanhang (Delgado et al., 2015). Det gäller inte minst BPM-verktygen som består av många programvaror och applikationer (ibid.).

På grund av OSS-verktygens potential och de oklarheter som finns kring de tillämpningarna av BPM-verktygen blir det nödvändigt att fördjupa sig i frågan om hur

OSS-verktygen stöder centrala egenskaper för BPM-systemet. Detta gäller även tillämpningar på företagens affärssystem.

Kvantitativa studier har genomförts av Delgado et al. (2015), där checklistor användes för att identifiera styrkor och svagheter i ett flertal OSS-baserade BPM-system. Dessa styrkor och svagheter jämfördes med en marknadsledande proprietär eller kommersiell BPM-aktör. Granskningen visade att alla BPM-verktyg hade fördelar och nackdelar, samt att det optimala valet av verktyg berodde på organisationens sammanhang och specifika behov. Granskningen visade även att OSS-baserade BPMS som Bonita, Activiti och Intalio CE var, med vissa undantag, lika kompletta och konkurrenskraftiga som motsvarande proprietära produkter från etablerade kommersiella aktörer.

I likhet med Delgado et al. (2015) kom även Bernroider, E. & Bernroider, M. (2008) fram till slutsatsen att BPM-verktygen redan för ett decennium sedan kunde användas på ett effektivt sätt i kombination med eller ersättning för kommersiella produkter. Både Delgado et al. (2015) och Bernroider, E. & Bernroider, M. (2008) konstaterade dessutom att BPMS starkt har förbättrats under senare år. Även det motiverar en fortsatt granskning av dessa system.

Vid en granskning av dessa system är det avgörande att förstå funktionerna i systemens verktyg för att säkerställa att rätt verktyg väljs för rätt företag (Delgado, 2015). Det finns dock åtskilliga varianter av BPMS, vilket gör att alternativen noga måste övervägas (Delgado, 2015). Såväl Bernroider, E. & Bernroider, M. (2008) som Delgado et al. (2015) listade därför ett antal tekniska och icke-tekniska moduler samt deras egenskaper. Denna lista av egenskaper kan användas för att undersöka hur de stödjer företagen i implementeringen av BPM-baserade affärssystem.

Det tycks dock inte finnas några omfattande studier som utvärderar *hur* open source-verktyg stöder viktiga centrala egenskaper hos BPM-verktyg, som är viktiga för specifika typer av företag. I synnerhet inte studier som baseras på kvalitativ metod. Därför blir denna uppsats unik genom att med utgångspunkt från Delgado et al. (2015) utvärdera några av dessa verktyg, främst i syfte att stödja beslutet från företag. Denna studie fokuserar därför på några särskilt utvalda BPM-verktyg för att undersöka hur open source-verktyg stöder centrala egenskaper för BPM system. Det kommer dessutom bli en kontextlös studie av "State.of.the-art" samt kommer utgå från egenskaper som identifierades av Delgado et al. (2015). Studien ska utvärdera ett antal av de mest lovande Open Source-baserade BPM-verktyg som är viktiga för specifika typer av företag. Det som gör denna uppsats unik är alltså att den, till skillnad från Delgado, inte bara undersöker i vilken grad verktygen skiljer sig åt men även beskriver hur verktygens egenskaper skiljer sig. Uppsatsen blir även unik genom sitt kontextlösa sammanhang vilket, till skillnad från Delgados mer specifika sammanhang, tillåter en större generalisering av analysresultaten.

4.1 Syfte

Syftet med denna uppsats är att utvärdera i vilken grad och på vilka sätt konkurrenskraftiga OSS-baserade BPM-verktyg stöder företag med proprietära BPM-verktyg när de tillämpas på affärssystem. Avsikten är att förenkla företagets val av BPM-verktyg genom att utvärdera särskilt utvalda BPM-verktyg, exempelvis vad gäller installation, kompatibilitet, funktionalitet, gränssnitt, användarvänlighet, säkerhet, samt företagets struktur och förväntningar. I en sådan studie är förhoppningen att kunna visa på vilka sätt Open Source stöder centrala egenskaper för BPM-verktyg genom att kategorisera och jämföra dessa utvalda OSS-baserade BPM-verktyg utifrån särskilda kriterier som ställts upp i tidigare undersökningar. Därefter kan uppsatsresultatet kontrasteras mot resultat från tidigare undersökningar.

4.2 Frågeställning

Ovanstående syfte leder till följande frågeställning:

Hur stöder open source-verktyg centrala egenskaper för BPM-system?

4.3 Avgränsningar

Vad gäller urvalet avgränsar sig denna studie till de mest lovande OSS-baserade BPM-verktygen som vanligtvis används av flertalet stora företag med många års erfarenhet. En sådan avgränsning borde bidra till större pålitlighet i undersökningsresultaten (Delgado et al., 2015). Med större resurser skulle denna studie kunna baseras på mer omfattande och globala urval, där analysresultaten exempelvis kontrasteras mellan två internationella urvalsgrupper. Vad gäller kvalitetsgranskningen av BPM-system baseras denna studie på den lista av undersökningsfaktorer som skrivits i rapporten av Delgado et al. (2015).

4.4 Förväntat resultat

Förväntningen är att bygga vidare på Delgado et al. (2015) genom en alternativ kontextlös undersökning som belyser och jämför centrala egenskaper i några väl etablerade OSS-baserade BPM-verktyg. På så sätt skapas en djupare bild av open source licenserade BPM-verktyg och hur de kan stödja större företag. Vidare förväntas studien mer djupgående undersöka hur dessa verktyg stödjer BPM-systemens centrala system, och ge en bättre insikt i vilka verktyg som gör vad och vilka verktyg som är olika bra för olika organisationer.

5 Metod

I vetenskaplig forskning är det nödvändigt att använda en lämplig vetenskaplig metod som guidar forskaren i utförandet av sitt arbete. Denna forskning baseras ofta på kvantitativ eller kvalitativ metod (Oates, 2006). Denna studie baseras på kvalitativ metod. Anledningen är att kvalitativ metod kan ge mer nyanserade och detaljerade svar på forskningsfrågor, i jämförelse med kvantitativ metod som ger mer generella analys svar (Bryman, 2011). En annan anledning är att kvalitativ metod inte behöver baseras på lika stora urval, vilket förenklar insamlingen av forskningsmaterial i denna uppsats (Patton, 2002).

Kvalitativ metod kan bland annat bestå av intervjuer, observationer och kvalitativa enkätundersökningar. Den kan även baseras på tematiska innehållsanalyser som lyfter fram innehållet i vetenskapliga texter och dokument (Bryman, 2011). Denna uppsats utgår från en sådan metod. Anledningen är att tematiska innehållsanalyser kan struktureras kring ett antal teman som enkelt går att jämföra med andra undersökningar. En tematiserad innehållsanalys valdes även för att denna metod tycks vara en etablerad och erkänd analysmetod inom detta forskningsfält (Delgado et al., 2015). Dessutom bör en sådan kvalitativ innehållsanalys kunna utgöra ett alternativ till tidigare kvantitativa studier som gjorts i detta ämne, exempelvis av Bernroider och Bernroider (2008) samt Delgado.

Det är denna kvalitativa aspekt som är unik för denna uppsats i jämförelse med Delgado et al. (2015). Det betyder att medan Delgado lutade sig mer mot kvantitativa jämförelser så baseras denna uppsats på kvalitativa och mer beskrivande analyser. En annan skillnad är att denna uppsats placeras i ett mer generaliserat och kontextlöst sammanhang medan Delgado fokuserar mer på specifika fall där de rent praktiskt testade olika verktyg för utvärdering. Det betyder att denna uppsats blev mer beskrivande, teoretisk och generell, medan Delgados studie är mer praktisk och specifik.

Samtidigt finns stora likheter mellan denna uppsats och Delgado et al. (2015) vad gäller val av BPM-verktyg och egenskaper samt jämförbarheten dem emellan. På grund av dessa dynamiska skillnader och likheter faller valet alltså på en jämförande kvalitativ tematisk innehållsanalys i relation till Delgado. Nedan beskrivs därför tillvägagångssättet för att genomföra uppsatsens kvalitativa tematiska innehållsanalys vad gäller datainsamling, urval och dataanalys.

5.1 Datainsamling

I denna uppsats genomfördes en jämförande tematisk innehållsanalys av OSS-relaterade BPM-verktyg. För att lyckas med detta var det nödvändigt att samla in litteratur och dokument som behandlar tidigare erfarenheter av sådana BPM-verktyg. Genom att använda sökord som *BPM*, *Business Process Management*, *Open Source*, *BPM-tools* och *BPM-egenskaper* i databaser som Google Scholar, Semantic Scholar, IEEE Xplore och

Scopus identifierades relevanta artiklar, dokument och tekniska rapporter. På grund av bristande aktuell dokumentation över BPM-verktygens egenskaper användes även så kallad backward snowballing (Jalali & Wohlin, 2012) där ett relevant dokument pekade vidare till ytterligare relevanta dokument.

Eftersom det forskningsmaterial som valdes ut genom sökningarna är avgörande för studiens forskningsresultat så kvalitetsgranskades materialet (Denscombe, 2009) genom att först läsa igenom artiklarnas och dokumentens sammanfattningar och därefter göra en djupare genomläsning av de mest relevanta artiklarna som framförallt berör OSS-relaterade BPM-verktyg. Med relevanta artiklar avses det forskningsmaterial som ansluter till uppsatsens frågeställning och litteraturstudie. Med relevanta dokument avses de dokument som på ett uppdaterat och tillförlitligt sätt beskriver BPM-verktygens egenskaper.

5.2 Urval

Som framgår av föregående avsnitt gjordes ett urval av dokument och forskningsmaterial genom olika former av sökningar. Dessa urval beskrivs närmare nedan.

Urval av dokument för granskning

På grund av att de sökmetoder som beskrevs ovan lyfte fram bristfälliga och knapphändiga källor blev inte insamlingen av dokument tillfredsställande. Ett återstående alternativ blev därför att finna uppdaterad information om BPM-verktygens egenskaper genom att besöka verktygens hemsidor och Community-bloggar. Dessa hemsidor och hänvisningar från bloggar beskrev verktygens egenskaper på aningen skilda sätt, vilket gjorde det till en utmaning att hitta jämförbara dokument. En stor del av analysmaterialet baseras därför på dokumentation som kommer från BPM-verktygens hemsidor. Denna dokumentation går att finna i uppsatsens bilaga. Anledningen är att BPM-verktygens hemsidor antogs ge den mest tillförlitliga informationen om funktionalitet, kompatibilitet, säkerhet och de andra egenskaper som skulle jämföras. En svaghet kan vara att hemsidorna lär ha ett egenintresse i att framställa det egna verktyget i ett extra positivt ljus och därmed riskerar att vara snedvridet. En annan svaghet är att BPM-verktygen ständigt utvecklas vilket gör det svårare att kontrollera vilka uppgifter som är uppdaterade. Det kan vara bra att ha detta i åtanke vid analysen av dokumenten.

Urval av BPM-verktyg

För att genomföra en innehållsanalys blev det även nödvändigt att välja ut vilka BPM-verktyg som skulle analyseras och jämföras med varandra. Med utgångspunkt från de verktyg som listas av Delgado et al. (2015) valdes de BPM-verktyg som bedöms vara relevanta för denna uppsats. Verktygen ansågs vara relevanta om de var lämpade för de stora företag som uppsatsen fokuserar på samt hade ett visst mått av stabilitet genom att ha vara representerade på marknaden i åtminstone några år.

Det fanns flera skäl till att främst utgå från Delgado i detta urval av BMP-verktyg. Det första skälet var att Delgados undersökning täcker en stor del av marknaden för OSS-baserade BPM-verktyg (Delgado et al., 2015). Ett annat skäl var att detta urval möjliggör en direkt jämförelse mellan denna studie och uppsatsen. Dessutom används de valda verktygen av stora företag, exempelvis Carrefour, Deutsche Bank, Singapor-Airlines, First Financial Bank, BBVA, AREVA, JPROFILER, OXIGEN, med flera. Eftersom dessa verktyg har funnits på marknaden och använts av stora företag under flera år kan man betrakta dem som stabila, vilket ökar analysresultatens reliabilitet (Bryman, 2011). Dessutom fokuserar uppsatsen på större och mer väletablerade företag. Av dessa skäl valdes nedanstående OSS-baserade BPM-verktyg ut för granskning.

XPDL: Bonita CE och Petals

WS-BPEL: Apache ODE och jBPM

BPMN: Activiti och JBPM5

Urvalet av OSS-baserade BPM-verktyg innefattar ett brett omfång av verktyg genom att de baseras på tre skilda standarder, nämligen XPDL, WS-BPEL och BPMN.

Urval av egenskaper

För att genomföra en jämförande innehållsanalys blev det även nödvändigt att välja vilka av verktygens egenskaper som skulle jämföras. Även här utgick uppsatsen huvudsakligen från de egenskaper Delgado et al. (2015) listar. Ett skäl var att uppsatsen ville vara direkt jämförbar med egenskaper som analyserats i tidigare forskning (Tsalgatiidou, 1998). Framförallt fokuserades på egenskaper som berör OSS och storföretag. Dessa egenskaper är samtidigt uppsatsens teman. Därmed förvandlas uppsatsen till en tematisk innehållsanalys (Bryman, 2011).

Utöver att välja egenskaper som nämns av Delgado et al. (2015) fokuserar uppsatsen även på tekniskt (icke-praktiskt) baserade egenskaper/teman. Anledningen var att praktiska moment inte ingick i denna studie. Vad gäller de tekniska egenskaperna utgick Delgado et al. (2015) ofta från flertalet tekniska kategorier och sorterade bort samt lade till ett antal kategorier. Dessa kategorier och egenskaper viktades mot företags speciella behov. Eftersom denna studie är kontextlös (inte riktar in sig mot ett särskilt företag) så baseras den på kategorier som antas vara tillämpbara på stora företag generellt. Därför har nedanstående kategorier och egenskaper valts ut, sammanfattade i tabell 3.

Tabell 3 visar de kategorier och egenskaper som ingår i uppsatsens analys (baserad på Delgado et al., 2015).

Kategorier	Egenskaper
Teknologi, Arkitektur och Interoperability	BPMS arkitektur och integration med sociala nätverk
Process Design och Modellering	Modelleringstyp
Hanterings form	Support för mobila enheter
Arbetsflödesmotor	Stöd för arbetsflödesmönster
Säkerhetshantering	Tillståndsmekanism för användare
Förvaltning, övervakning och granska	Processövervakning
Dokumenthantering system	Integritet och dokumentsäkerhet
Portal	Sökfunktioner
Installation och support	Installationspaket
Mognad	Tid på Marknaden /Community-aktivitet

5.3 Dataanalys

Utmärkande för den tematiska innehållsanalysen är tematiseringen av de texter som analyseras (Berndtsson et al., 2008). Genom att tematisera de utvalda dokument som berör uppsatsens frågeställningar kunde denna studie på ett systematiskt sätt jämföra OSS-baserade BPM-verktyg. Tematiseringen utfördes genom att välja ut ett antal av de teman/egenskaper som presenteras av Delgado et al. (2015, s.84). Dessa teman valdes genom att granska den forskningslitteratur som berör denna frågeställning (främst Delgado et al., 2015; Bernroider & Bernroider, 2008) och den vägen få en uppfattning om vilka teman som är mest relevanta för frågeställningen.

I praktiken innebar det att granska de dokument som beskriver specifikationerna och egenskaperna för varje verktyg. Tematiseringen genomfördes därför genom att notera eller markera de olika BPM-verktygens dokument som anslöt till studiens fastställda teman. Därefter granskades dessa textavsnitt extra noga, för att visa likheter och skillnader mellan dokumenten. På detta sätt kunde dokumenten jämföras, tema för tema, och analyseras i uppsatsens analysavsnitt. Genom detta tillvägagångssätt blev det möjligt att jämföra uppsatsens analysresultat med resultaten i exempelvis Delgado et al. (2015).

Eftersom studien baseras på redan fastställt material är det en empirisk undersökning, och denna empiriska undersökning utförs genom att dra induktiva slutsatser från detta empiriska textmaterial (Marconi & Lakatos, 2010). Genom denna induktiva metod passerade studien tre faser, där studien först observerade och tematiserade de inhämtade dokumenten, därefter identifierade relationen mellan dessa teman, och slutligen identifierade likheter eller skillnader mellan studiens upptäcker slutsatser och tidigare forskning (Delgado et al., 2015). På grund av uppsatsens kontextlösa sammanhang blev det även möjligt att i viss mån generalisera analysresultatet i nya idéer för vidare forskning. Eftersom uppsatsen dessutom är unik genom sin kvalitativa ansats kunde den även beskriva inte bara i vilken grad verktygens egenskaper skiljer sig åt men även mer exakt *hur* de skiljer sig.

5.4 Validitet och reliabilitet

Studiens validitet, reliabilitet, objektivitet och generaliserbarhet är avgörande för dess vetenskapliga kvalitet (Marshall, 1996). En studies validitet handlar om hur väl studien mäter det den är avsedd att mäta (Patton, 2002). Genom att noga välja relevanta BPM-verktyg som redan testats i en tidigare peer reviewed studie (Delgado et al., 2015) samt genom att flera gånger läsa igenom de dokument som behandlar dessa verktyg och granska dem utifrån fastställda kriterier (se avsnitt 5.2.1) så kunde studiens validitet förbättras. Däremot kan ett validitetsproblem uppstå om dokumenten som analyserades inte representerar det som undersöks på ett fullgott sätt. Exempelvis kan man tänka sig att ett flertal källor dokumenterar BPM-verktygen på skilda sätt och med olika djup och nyans. Frågan är alltså hur väl valt uppsatsens dokumentmaterial är.

Reliabilitet handlar om hur tillförlitligt studieresultatet är, genom att studien skall komma fram till samma resultat om den utfördes på exakt samma sätt av andra forskare (Strauss & Corbin, 2008). Genom att vara noggrann och systematisk i att följa de procedurer som liknar Delgado et al. (2015) vad gäller urval, kriterier och tillvägagångssätt så bör studiens reliabilitet förbättras.

En svaghet som kanske riskerar att minska studiens reliabilitet kan vara den brist på objektivitet som uppstår genom att studien granskade dokumenten utifrån forskarens subjektiva synpunkter. Dessa subjektiva synpunkter gjorde att studie fångade upp viss information men uteslöt annan information. En annan person kanske hade fångat upp annan information ur dokumenten (Marshall, 1996). Vad gäller studiens generaliserbarhet bör analysresultatet kunna appliceras på större företag och organisationer. Däremot är det tveksamt om analysresultatet kan tillämpas på organisationer som faller utanför studiens urval, exempelvis internationella småföretag som använder andra verktyg än de som ingår i uppsatsurvalet.

5.5 Forskningsetiska principer

Vetenskapliga studier behöver baseras på åtminstone fyra forskningsetiska principer, nämligen informationskravet, samtyckeskravet, Konfidentialitetskravet och nyttjandekravet (Vetenskapsrådet 2002). Eftersom denna studie baseras på redan publicerade dokument och inte på intervjuer, enkätundersökningar, deltagande observationer eller på något sätt samspelar med organisationer, företag eller individer, så borde denna uppsats inte bryta mot dessa forskningsetiska principer. Angående nyttjandekravet kommer dokumenten och den inhämtade informationen användas till enbart denna uppsats. Från detta perspektiv bedöms denna studie tillmötesgå de fyra forskningsetiska principerna.

Däremot kan ifrågasättas om och på vilka sätt de artiklar och dokument denna studie baseras på tillmötesgår olika etiska riktlinjer. I och med att denna uppsats på flera sätt baseras på och jämförs med Delgado et al. (2015) blir en fråga varför Delgados studie inte för några forskningsetiska resonemang. Trots allt baseras denna studie på flertalet fältundersökningar och praktiska fallstudier som lär innefatta krav på etiska förhållningssätt. Det är alltså oklart om Delgados studie tillmötesgår uppställda forskningsetiska principer, vilket i någon mån kan tänkas ha betydelse för de etiska aspekterna i denna uppsats. Övriga dokument som baseras på BMP-verktygens specifikationer är emellanåt uppställda på olikartade sätt, vilket försvårar en jämförelse dem emellan. På grund av att verktygen dessutom kontinuerligt uppdateras är det inte helt klart hur väl dokumenten beskriver verkligheten. Det kan i sin förlängning innebära att uppsatsresultaten inte är helt korrekta och uppdaterade. Felaktiga analysresultat lär i sin tur innebära att företag efter en granskning av denna uppsats riskerar göra en kostsam felbedömning. Ytterligare en fråga är om användningen av verktygen är etisk för samtliga personer som berörs av verktygen. Framförallt kan tänkas att vissa personer förlorar sina positioner på arbetsplatsen genom att bli ersatta av ett mer effektivt affärssystem. Visserligen kanske det inte kan motivera en regelrätt restriktion mot användandet av BPM-verktyg, eftersom de mest gångbara systemen kan antas öka den totala välfärden. Däremot kan det vara motiverat att åtminstone belysa en sådan aspekt i denna etiska diskussion.

6 Materialpresentation/Empiri

I detta kapitel presenteras hur de OSS-baserade BPM-verktygen stöder olika egenskaper. Denna empiri baseras på de dokument som listas i uppsatsens bilaga 1.

6.1 BPMS-arkitektur (eng. BPMS-architecture)

Som framgått av bakgrundsbeskrivningen kan BPMS-arkitekturen betraktas som ett helt system av processer. Genom att definiera processarkitekturen kan företagen sammankoppla rätt arbetsteam med rätt resurser och teknologier. Därigenom uppnås uppställda mål på ett så effektivt sätt som möjligt. Avsikten med en bra BPM-arkitektur är alltså att optimera resultatet genom förbättrad processhantering. I denna processhantering är det nödvändigt att processarkitekturen i det använda BPM-verktyget ger stora möjligheter till att anpassa systemet för att det uppfyller specifika försiktig användning av "egenskap" eftersom "BPM-arkitektur" är i sig en egenskap.

Bonitasoft: Denna BPMS-arkitektur har en servicebaserad arkitektur som innehåller mer än femtio API (Application Programming Interface). Det inkluderar de API som baseras på Java och EJB3, vilka kan användas för att utveckla och integrera företagets applikationer. I Bonitasofts exekveringsmotor gör det möjligt att anpassa systemet till de flesta typer av informationssystem, från de enklaste till de mycket komplexa. Bonitasofts arkitektur är uppdelad i tre moduler: Bonita Studio, Bonita BPM-engine (BPMN-motor) och Bonita User Experience (Bonitasoft, u.å.).

Petals: Detta verktyg har en nativt distribuerad arkitektur som är anpassad till hela infrastrukturen. Bland annat hanterar den problem med "runtime"-frågor och ansvarsfördelningar i organisationen. Verktyget kan även skapa lösa kopplingar, infrastrukturkontroller och skalbara lösningar i en komplett distribuerad arkitektur. Genom sin skalbara arkitektur eliminerar Petals därför den inflexibilitet som vanligen finns i punkt-till-punkt-integrationer (Petals, u.å.)

Apache ODE: Verktyget har en ODE-arkitektur (Orchestration Director Engine) som består av flertalet komponenter i stil med ODE BPEL-kompilator (Business Process Execution Language), ODE BPEL Engine Runtime, ODE Data Access Object (OAO), ODE Integration Layer och användarverktyg. Dessa komponenter har en hög nivå av kompatibilitet dem emellan. Kompilatorn omvandlar BPEL-dokumentationer till blanketter som kan exekveras på ett tillförlitligt sätt. Kompilatorn omvandla instruktioner till maskinkod som sedan kan direkttolkas i systemet (Apache, u.å.).

jBPM: Processen i jBPM utvecklas i Process Designer, vilken sedan förs in i processdatabasen. JBoss jBPM-motorn interagerar sedan med processdefinitionen för att svara på förfrågningar från jBPM-webbkonsolen, där slutanvändaren använder en processapplikation. jBPM-webbkonsolen lagrar även insamlade data i processdatabasen (jBPM, u.å.).

Activiti: Activiti-arkitekturen är kärnan i Activiti-projektet. Det är en processmotor i Java, där BPMN 2.0-processerna är införda. Den har egenskapen att tillåta att användaruppdateringar kombineras med processuppdateringar i en enda transition. Den kan köras i någon Java-miljö som Spring eller JTA och den är lätt att konfigurera. Den är även byggd för att redan på ett tidigt stadium stödja skalbarhet i molnet. Något som måste tas hänsyn till är att processens virtuella maskin utgör grunden för Activiti-motorarkitekturen. Det gör det möjligt att lägga till nya typer av arkitektur, funktioner och kompletta processpråk (Activiti, u.å.)

JBPM5: Arkitekturen är en kommando-baserad underliggande teknik som består av följande komponenter: Rensa (eng. clear) API, ursprunglig (eng. native) BPMN2-körning, inbäddbarhet (eng. embeddability). Tekniken används för att kunna exekveras som en enkel Java POJO-komponent, varvid applikationerna kan köras oberoende (JBPM, u.å.).

6.2 Integration med sociala nätverk (eng. Integration with social networks)

Den sociala BPM-tekniken integrerar organisationssystem, Business Process Management system (BPMS), social teknologi (Programvara/sociala medier) och organisatoriska kommunikations- och samarbetsmiljöer (intranät, e-post och sociala organisationsnät) för att öka samarbetet mellan processansvariga. Om de två sätten se (sektion 7.1).

Bonitasoft: Genom verktyget går det att koppla processerna till sociala nätverk, bland annat Facebook och Twitter. Verktyget är komplett, dokumentation tillhandahålls och Bonitasofts tekniska experter finns tillgängliga vid problem. Webinarier organiseras dessutom på regelbunden basis för att hjälpa community-deltagare (Bonitasoft, u.å.).

Petals: Genom Petals är det möjligt att kommunicera direkt via Facebook, Twitter, LinkedIn, ett antal bloggar och Viadeo som är en populär plats för diskussion och relationskapande (Petals, u.å.).

Apache ODE: Apache ODE samarbetar med Facebook. Om verktygs stöd för sociala nätverk går utifrån dokumentationen inte att svara på (Apache, u.å.).

JBPM: Detta verktyg integreras med sociala nätverk via interaktioner genom exempelvis Facebook, Twitter och bloggar. På dessa platser finns stöd som är kunniga och hjälper till med problemlösningar i JBPM (JBPM, u.å.).

Activiti: Activiti-communities har formats i syfte att ge en möjlighet för individer att förstå vilka möjligheter som finns på denna plattform. Verktyget syns bland annat på bloggar och Wiki. Det krävs dock registrering för att bli deltagare på Activitis communities (Activiti, u.å.).

JBPM5: Verktöget är integrerat med diverse nätverk, exempelvis ett Twitter-konto och flertalet bloggar. Dessa kommunikationskanaler kan användas för att kommunicera med jBPM-teamet (jBPM, u.å.).

6.3 Modelleringstyp (eng. Modeler type)

Modelleringstyper representerar vilka modelltyper som BPM-verktygen kan skapa. En organisation eller organisationsnivå kan till exempel ha önskemål om hur vissa processer ska tillämpas på deras affärsfunktioner. Verktygen kan då användas för att modellera och matcha dessa önskade affärsfunktioner. Många modelleringstyper har utvecklats för att formalisera BPM-verktygen. I en strävan att vara omfattande och att kunna tillämpas på många olika sätt har de därför blivit komplexa (Skrinjar & Trkman, 2013).

Bonitasoft: Detta verktyg har modelleringstyper i form av Process Modell (BPMN2). Utformningen av affärsflödet BPMN (Business Process Modeling Notation) version 2.0 tillåter användning av grundläggande eller avancerad notering. Användarvänlig processmodellering tillåter delning av processmodeller där exempelvis processägare, intressenter, affärsanalytiker och utvecklare kan sammankopplas (Bonitasoft, u.å.).

Petals: Verktöget finns i flera versioner, bland annat Petals ESB 4.2. Den förbättrar utbudet av de verktyg och möjligheter som finns tillgängliga för Petals-vänliga produktionsmiljöer. Petals CLI 2.1.0 kan handskas med felhanteringar, inklusive felaktigt hanterade loggningar (Java-loggningar med API-stöd). Petals Studio 1.3.2, Studio 1.3.3 och ESB 4.1 är exempel på några av de nya versioner som Petal har lanserat på marknaden. Det kommer ständigt nya versioner och det går därför inte att uppge exakt hur många de är (Petals, u.å.).

Apache ODE: Detta verktyg erbjuder relativt många olika versioner, exempelvis ODE 1.3.8, ODE 1.3.7 (utkom 2017-03-23) och ODE 1.3.6 (utkom 2013). Mellan 2006 och 2018 har därför flertalet modellerings typ med olika funktionaliteter lanserats av Apache ODE. Mer om detta kan läsas på ode.apache.org. (Apache, u.å.).

jBPM: Detta verktyg bygger på en generisk processmotor som innefattar flertalet modelleringstyper. En modelleringstyp är jBPM5 och den utgår från BPMN 2.0 för att utföra processer. BPMN 2.0 är en standardspecifikation som definierar XML-serialiseringar. Verktöget visualiserar även affärsprocesser och kan utvidgas till att inkludera mer avancerade funktioner. jBPM4 stöder istället språken JPDL och BPMN1.2.5 (jBPM, u.å.).

Activiti: Verktöget driver BPMN 2.0-processerna på ett stabilt sätt. Det erbjuder ytterligare funktioner i form av webbaserade modelleringsverktyg för affärsanalytiker och en webbapplikation för att arbeta med och hantera processer (Rademakers, 2012). I standardkonfiguration lagras Activiti-processdefinitionen i filsystemet (Activiti, u.å.).

JBPM5: Detta verktyg innefattar samma grundfunktioner för modellering som jBPM. Mer specifikt så har JBPM5 ett visuellt redigeringsverktyg som gör det möjligt för användare

utan programmeringskunskaper att skapa formulär för att fånga upp och visa processinformation. Detta visuella redigeringsverktyg riktar sig till processanalytiker och utvecklare som bygger och testar formulär, och det innefattar även avancerade funktioner för att skraddarsy formulär (jBPM, u.å.).

6.4 Stödspråk (eng. Support languages)

Språk som stöder BPM-verktyg är programmeringsspråk som ger instruktioner till en dator, och genom en uppsättning syntaktiska och semantiska regler definierar språket BPM-verktyget. Dessa språk understödjer datalagringar eller överföringar och vilka åtgärder som ska vidtas under olika omständigheter.

Bonitasoft: Verktyget har stöd av ett flertal programmeringsspråk, inte minst Java, Angular JS med mera (Bonitasoft, u.å.).

Petals: Har stöd av programmeringsspråken Java och XLM (Petals, u.å.).

Apache ODE: Är programmerat genom och stöds av programmeringsspråket Java, samt bygger på JDK v3.0-funktioner (Apache, u.å.).

jBPM: Detta verktyg stöds av flertalet språk, framförallt Java, HTML, Shell, JavaScript, Studio Eclipse och XSLT Extensible Stylesheet Transformation Language (jBPM, u.å.).

Activiti: Activiti-verktyget har stöd av de tre programmeringsspråken Java och Apache (Activiti, u.å.).

JBPM5: Detta verktyg stöds på samma programmeringsspråk som jBPM. Dock var jPDL det ursprungliga flödesspråket för jBPM, vilken hanterade ett grundläggande gränssnitt (jBPM, u.å.).

6.5 Processövervakning (eng. Process monitoring)

Genom processövervakningen bedöms vanligtvis processens "hälsotillstånd" vad gäller genomsnittlig processcykeltid, antal processer som avslutas för tidigt, antal processfel, genomsnittskostnad för processcykeln, och så vidare. Vid denna processövervakning är det avgörande att i tid upptäcka kritiska situationer för att kunna vidta lämpliga åtgärder inom utsatt tidsram (Chandy & McGoveran, 2004).

Bonitasoft: Verktygets processövervakning ger information i förhållande till en särskild process eller processdel. I denna processövervakning kan exempelvis tabeller filtreras för att visa en viss process, processdel, version eller specifik version. Processhanteraren kan enbart se processerna som hanteras. Från en rad i en tabell exempelvis, kan användaren klicka på listikonen för att se en lista över olika processfall. Genom att klicka på bildikonen för ett processfall öppnas ett diagram över denna enskilda process. Processdiagrammet skapades utifrån den distribuerade processdefinitionen. Det betyder att vissa detaljer som är synliga exempelvis kommentarer, inte är tillgängliga i Bonitasoft (Bonitasoft, u.å.).

Petals: När en förfrågan (eng. request) inkommer till Petals genom ett tjänstanrop från en bindande komponent, avfyra den flera andra tjänsteapplikationer. Alla dessa tjänsteapplikationer utgör ett flöde eller en process. Det finns en mekanism som omskapar dessa flöden eller processer, vilka exekveras från en ingångspunkt. Denna mekanism är baserad på standardloggar. Därigenom lämnas spår från inloggade utifrån varje inmatning/utmatning av tjänsteanrop som ingår i de olika flödes/processstegen (Petals, u.å.).

Apache ODE: Verktöget ODE har ett komplett administrations-API för att kontrollera vilka processer som används, körs och slutförs, samt kontroll av instanser, variabler och flertalet olika värden. För att se vilka processövervakningar som finns tillgängliga går det att granska gränssnitten för processhantering och InstanceManagement. Den dokumentgenerator som verktöget använder, Javadoc, är dock omfattande se bilaga för Apache ODE (Apache, u.å.).

jBPM: Med jBPM levereras kompletta BAM-verktyg (Business Activity Monitoring) som gör det möjligt även för icke-tekniska användare att visuellt komponera företagets instrumentpaneler. Genom jBPM är det alltså enkelt att utveckla och övervaka affärsprocesser och rapporteringslösningar (jBPM, u.å.).

Activiti: Activiti har ingen specifik lösning för processövervakningar. Däremot förklaras i boken "Activiti in Action" (Rademakers, 2012) att ett sätt att hantera denna processövervakning är att integrera Activiti med Esper Framework, vilken är en programvara i Java som gör det möjligt att analysera resultaten från processflödena (Activiti, u.å.).

JBPM5: Detta verktygs processflöden uttrycks vanligen genom XML. Filerna kan enkelt lagras i filsystem under utvecklingen. Om informationen ska göras tillgänglig för en eller flera användare, rekommenderar jBPM användningen av en informationsenhet som centraliserar informationen på ett logiskt sätt. Datamodellen innehåller tre enheter, en för processinstansinformation, en för nodinformation och en för (process-)variabelinstansinformation (jBPM, u.å.).

6.6 Installationspaket (eng. Installation packages)

I datavetenskap är installation (eng: *Installation* eller *Setup*) en process där verktygets alla nödvändiga komponenter placeras och koordineras i ett centralt verktyg (i detta fall BPM-verktyg). Installation är ett av de sätt som möjliggör att verktygen i praktiken kan hanteras av användaren.

Bonitasoft: Verktöget är open source och finns tillgängligt på hemsidan. För att få det installerat på sin dator behöver användaren bara skapa ett konto. För att börja använda verktöget krävs först att användaren väljer vilken version av programvaran som ska installeras. Bonitasoft är tillgänglig på operativsystemen Windows, Mac och Linux. För Bonitasofts designfunktioner erbjuds separata nedladdningar av programvaran. Den

första versionen är en fullständig version som installeras på datorn, vilken innehåller Bonitasoft BPM studio och en integrerad webbportal (Bonitasoft, u.å.).

Petals: Detta verktyg kan installeras manuellt via ett ZIP-arkiv. Det finns även "Debian"-paket för Petals-servern, Petals CLI och deras tillägg. Petals erbjuder olika startlägen med möjligheter för respektive interaktiva kommandon. Petals erbjuder även "Daemon-kommandorader", men dessa är tillgängliga endast i Unix-baserade system (Petals, u.å.).

Apache ODE: För att installera detta verktyg krävs Eclipse IDE för Java EE-utvecklare i version 3.4. Om användaren nyttjar ett annat Eclipse-paket, till exempel Eclipse för Java Developer (utan EE), krävs även webbverktygsplattformen, samt att Tomcat är ansluten till Eclipse. (Apache, u.å.).

jBPM: Installation av jBPM kräver att användaren har Java på sin dator. Verktyget tillhandahåller en länk för att hämta och installera Java. Först och främst måste installationsprogrammet laddas ner och "unzippas" på ett lokalt filsystem. Det erbjuds två installationsversioner. "Full installation" innefattar många av de funktionaliteter som behövs under och efter installationen. "Förenklad installation" innehåller bara installatören och laddar enbart ned de mest nödvändiga funktionerna (jBPM, u.å.).

Activiti: För att installera Activiti behövs en fungerande Java runtime och Apache Tomcat-installation. Systemvariabeln Java home behöver även bli korrekt inställd, vilket genomförs på olika sätt beroende på vilket operativsystem som används (Activiti, u.å.).

JBPM5: För att installera verktyget förutsätts att användaren redan har Java JDK 1.5+ installerad. jBPM tillhandahåller en länk för att hämta och installera Java. JBPM5 har nästan samma installationspaket som jBPM. Först och främst måste installationsprogrammet laddas ner. Även JBPM5 tillhandahåller en komplett "full installation" och en "förenklad installation", där det vanligtvis är bäst att ladda ner det kompletta installationsprogrammet "jBPM-version, installer-full.zip." (jBPM, u.å.).

6.7 Tid på marknaden (eng. Time in market)

"Tid på marknaden" betecknar tiden som produkten varit tillgänglig på marknaden, det vill säga från när första versionen av projektet/verktyget lanserades till dess den senast versionen gavs ut. Utvecklare utgår från produktens "tid på marknaden" för att utvärdera hur verktygen utvecklas på marknaden, inte minst i relation till extern konkurrens. Verktygens tid på marknaden är ett förenklat sätt att mäta BPM-verktygens övergripande produktnärvaro och position på marknaden.

Bonitasoft: Verktyget lanserades i Frankrike år 2001. För närvarande är det en av de mest använda BPM-verktygen i världen. Företaget har mer än 1000 kunder som Orange, Xerox, Accenture, La Redoute och Cisco. Verktyget används i ett av spritt över länder, erbjuder fyra utgåvor av sin programvara, inklusive en open source version. Från 2001 tills idag har Bonitasoft funnits på marknaden i 17 år. Verktyget är därför mycket stabilt eller solitt utifrån ett marknadsperspektiv (Bonitasoft, u.å.).

Petals: Verktöget föddes ur ett forskningsprojekt och har därefter genomgått åtskilliga förändringar och uppgraderingar. Den första stabila version av Petals släpptes den 21 september 2006 och verktöget har därför funnits förhållandevis länge på marknaden (Petals, u.å.).

Apache ODE: Detta verktöget utvecklades under insyn av "Apache Software Foundation" som grundades och etablerades på marknaden år 1999 (Apache, u.å.).

jBPM: Verktöget lanserades 2006 och har alltså funnits på marknaden i tolv år (jBPM, u.å.).

Activiti: Activiti är ett open source-verktöget som använder sig av BPMN (Business Process Management Notation) och skapades år 2010 av de två utvecklarna för jBPM vilka var anställda i Alfresco. Verktöget har därför funnits på marknaden i åtta år (Activiti, u.å.).

JBPM5: Verktöget introducerades på marknaden 2011 och har därför varit representerade i sju år (jBPM, u.å.).

6.8 Community-aktivitet (eng. Community activity)

Verktögens community-aktiviteter utförs av individer eller grupper som, ofta på obetald basis, arbetar för att verktöget utvecklas, exempelvis genom programmering, funktionsutveckling och rådgivning. Community-aktiviteterna utförs dock inte alltid på frivillig basis. Personliga förmåner kan erhållas av bidragsgivare, men sällan genom direkt lön eller betalningar.

Bonitasoft: För att gå med i Bonitasofts community kan individen registrera sig för medlemskap på bonitasoft.org. Därefter kan medlemmen besöka berörda forum on-line och utbyta information med andra användare, vilket inkluderar Bonitasoft-utvecklare på Bonita Community (bonitasoft.org/forum). Alla medlemmar i Bonita Community kan ställa frågor om problem som uppstått med verktöget, samt bifoga en kopia av sin loggfil och eventuell skärmdump för att göra det enklare för andra community-medlemmar eller Bonitasofts tjänsteteam att föreslå konkreta lösningar på problemet. Bonitasofts community innehåller även bloggar som erbjuder förhandsvisningar av teknik under utveckling. Det går även att ladda ner versioner av Bonitasoft som är under utveckling inom communityt. Medlemmar kan dessutom ladda upp egna bidrag för utveckling av process hantering, nya funktionaliteter, samt förslag på nya lösningar som kan vidareutvecklas av andra medlemmar (Bonitasoft, u.å.).

Petals: Petals community är relativt liten och består enbart av personer som tycks arbeta på helt frivillig och obetald basis. Dessa bidrar med sina kunskaper om Petals och om hur vissa tekniska problem kan lösas. Dessa community-medlemmar kan delas in i tre kategorier. Den första kategorin består av tre personer som tidigare bidragit med kod, det vill säga personer som är extra aktiva på Petals och i dess relaterade projekt. Den andra kategorin består av tjugotre personer som kallas "former committers" och den tredje kategorin består av sex personer som kallas "contributors", det vill säga entusiaster som bidrar till Petals projekt genom egna tekniska bidrag. Petals har även en så kallad

"community website" som innehåller ett forum där deltagarna kan interagera med varandra (Petals, u.å.).

Apache ODE: Apache ODE har ett community som består av utvecklare vilka ger ett brett utbud av information genom att svara på vanliga frågor och hjälpa nya medlemmar. De ser även till att tillhandahålla och upprätthålla ett antal verktyg eller funktionaliteter som kan vara användbara för användaren av Apache ODE (Apache, u.å.).

jBPM: Verktygets community är stort i jämförelse med de flesta andra. BPM-verktyg det går att följa community på Twitter (4456 följare) och på "Red Hat"-support samt Facebook (7014 följare) och på Google+. De har även en forumsida, blogg och chatt där medlemmar har möjligheter att kommunicera med jBPM-teamet (jBPM, u.å.).

Activiti: Verktyget har ett eget community med användare som har erfarenheter av teknik som kan integreras med Activiti (Rademakers, 2012). Exempelvis har medlemmar i activiti-communityt implementerat funktioner som SpringSource, FuseSource Mulesoft, Camunda och Spring Integration med Activiti (Activiti, u.å.).

JBPM5: På webben för JBPM5-communityt finns många användbara länkar för de som vill bli en del av gruppen. Där finns även flöden av blogginlägg relaterade till jBPM. Communityt har även Twitter-konto, användarforum för att ställa frågor och ge svar, samt en kontinuerlig byggservare där medlemmar kan dela på snapshots och diskutera med varandra (jBPM, u.å.).

6.9 Stöd för mobila enheter (eng. Support for mobil devices)

En mobilapplikation är ett programmerat system som installeras på mobila elektroniska enheter, exempelvis tablets och smarta mobiltelefoner (Saccol, Schlemmer & Barbosa, 2010). Via de mobila enheterna kan applikationerna exempelvis ge nya virtuella undervisnings- och inlärningsmiljöer i form av mobilt lärande (eng. Mobile learning).

Bonitasoft: Den mobila versionen av Bonitasoft visar bara funktioner för inloggade användare. För att installera denna mobila version behöver användaren en Bonitasoft Pack-utgåva. Denna mobila version visar menyer och funktioner där användaren kan skapa undertexter, lägga till kommentarer, visa och utföra uppgifter, med mera (Bonitasoft, u.å.).

Petals: Utifrån Petals verktygsdokumentation framgår inte om verktyget har stöd för mobila enheter. Däremot konstateras inte att verktyget saknar denna egenskap (Petals, u.å.).

Apache ODE: Även detta verktygs dokumentation tycks sakna information om det stödjer mobila enheter. Trots omfattande sökningar gick det därför inte att avgöra om och hur denna egenskap stöds av Apache ODE (Apache, u.å.).

jBPM: Detta verktyg har stöd för mobila enheter. Kunden kan använda detta stöd genom att installera en app på sin mobila enhet genom något av följande sätt: 1) Ladda ner appen

till datorn och manuellt överföra appen till den mobila enheten. 2) Klicka på en nedladdningsadress i en webbläsare i den mobila enheten. 3) Använda den mobila enheten till att skanna en QR-kod som innehåller nedladdningsadressen. I Android måste användaren tillåta alternativet att installera appar från okända källor. (jBPM, u.å.).

Activiti: Activiti har en så kallad "Microservice Mobile Web Client". Denna webbklient körs genom Angular CLI version 1.4.1 och genom att användaren navigerar till <http://localhost:4200/>. Appen uppdateras automatiskt om någon av källfilernas kodinställningar ändras. Vid en projektmodellering kan arbetet lagras i dist/katalog (Activiti, u.å.).

JBPM5: Detta verktyg har samma mobila stöd som jBPM (jBPM, u.å.).

6.10 Stöd för arbetsflödesmönster (eng. Support of workflow patterns)

Begreppet "arbetsflödesmönster" initierades 1999 av professorerna Wil van der Aals och Arthur ter Hofstede för att beskriva verktygens grundläggande flöden och strukturer. De presenterar en omfattande guide över arbetsflödesmönster vad gäller återkommande och övergripande affärsprocesskonstruktioner, exempelvis kvitton som automatiskt skrivs ut efter att tågbiljetter har utfärdats. Dessa affärsprocesser beskrivs främst utifrån kontrollflödes-, data- och resursperspektiv (Russell, van der Aalst & ter Hofstede, 2016).

Bonitasoft: En applikation kan integreras med Bonitasoft med hjälp av REST (Representational State Transfer). Om applikationen inte baseras på Java kan den istället integreras med Bonitasoft via Web REST API. Detta API ger tillgång till samtliga Bonita-objekt och kan användas för att utföra operationer såsom skapa, hämta, uppdatera och ta bort. Bonita-motorn är central för arbetsflödeslogiken, exempelvis vad gäller funktioner som villkorsstyrda portar, kontakter, meddelande, tidtagning, och så vidare (Bonitasoft, u.å.). Dessa funktioner aktiveras när applikationen sammankopplas med arbetsflödet (Workflow Patterns Initiative, 2017).

Petals: Utifrån dokumentsökningarna gick det inte att hitta information om stöd för arbetsflödesmönster i Petals verktyg. Däremot är det inte sagt att detta stöd inte finns, på ett eller annat sätt (Petals, u.å.).

Apache ODE: Detta verktyg innehåller en katalog över samtliga relevanta implementeringsdetaljer. Denna katalog innehåller installationsbeskrivning och processdefinitioner samt kan även innehålla filer av typen SVG eller XSL. Distributionsbeskrivningen finns i en fil med namnet "deploy.xml" (Apache, u.å.).

jBPM: Verktygets arbetsflödesmönster baseras på fördefinierade block av processelement. En kombination av dessa block kan återanvändas av användaren. Arbetsflödesmönstret består även av sammankopplade noder som skapar ett gemensamt körbart mönster, vilket kan återanvändas i en processmodell. Arbetsflödesmönstret går att finna i avsnittet "Arbetsflödesmönster" i verktygets objektbibliotek och kan klickas, dras och placeras på skrivbordet, precis som alla andra element. För att fästa elementets

mönster på skrivbordet klickar alltså användaren på ikonerna i paletten samt drar och släpper mönstret på duken. Mönstret kopplas då automatiskt till elementet (jBPM, u.å.). jBPM innehåller fördefinierade arbetsflödesmönster. Användaren kan även definiera sina egna arbetsflödesmätare efter behov (Red Hat, 2018).

Activiti: I Activiti kan arbetsflödesmönster visualiseras grafiskt med hjälp av Activiti designer. Detta arbetsflödesmönster kan justeras direkt genom att redigera XML-koden (Activiti, u.å.).

JBPM5: Verktøyets arbetsflödesmönster har samma standard och funktionalitet som jBPM (jBPM, u.å.).

6.11 Integritet och dokumentssäkerhet (eng. Integrity and document security)

Integritet är en informationssäkerhetsegenskap som kontrollerar om verktygens informationsflöden har genomgått ändringar vilka inte kan godkännas ur säkerhetssynpunkt. Informationsflödets integritet innebär därför att tillförlitliga utdata inte snedvrids av otillförlitliga indata (Biba, 1977). Dokumentssäkerhet handlar om hur viktiga dokument är arkiverade, lagrade, bearbetade, säkerhetskopierade, levererade och så småningom eliminerade (Kenwood Moving, 2018).

Bonitasoft: Verktøyet kan konfigureras för att utföra användarautentiseringar gentemot en LDAP-server (Lightweight Directory Access Protocol), exempelvis Active Directory, Apache Directory Server eller OpenLDAP. Denna typ av konfiguration bygger på en specifik implementering av Bonitasofts Engine-autentiseringstjänst, vilken vidarebefordrar det faktiska användarnamnet och lösenordskontrollen till en JAAS-tjänst (Java Authentication and Authorization Service), som i sin tur är konfigurerad genom en LDAP-specifik inloggningsmodul (Bonitasoft, u.å.).

Petals: Petals Web Consoles webbapp ökar säkerheten genom att administratören kan skapa, redigera, ta bort och på andra sätt hantera ett stort antal användare. Användarna av Petals Web Console kan tilldelas flera roller och användarrättigheter, och den vägen komma åt enbart vissa delar av programmet. Verktøyet har en särskild plats som specificerar kraven på Petals Web Console. Eftersom Petals Web Console använder ett antal klasser och gränssnitt som beskriver och definierar kontrakten på en övergripande servletklass (2.5 Servlet Specification API) behöver användarna utgå från den runtime-miljö som tilldelade den (Petals, u.å.).

Apache ODE: Detta verktyg har stöd för en WS-säkerhet (Web Service Security), som säkrar tjänster enligt följande process: I det första processteget tillhandahåller Apache ODE en funktion {<http://sample03.policy.samples.rampart.apache.org>} som förbereder de servicedokument som processen ska hantera. I det andra steget säkerställer "ODE Webapp Classpath" att de resurser som behövs för att anropa tjänsterna är tillgängliga (för Rampart). Typiska resurser är lösenordsåteruppringning med klasshantering, Java keystores och egendomsfiler med keystore-information (Apache, u.å.).

jBPM: Säkerheten i jBPM EAP (Extensible Authentication Protocol) spänner över ett brett fält. Verktøget har väl dokumenterade och standardiserade metoder för att säkra konfiguration, inmatningsdata och slutpunkter (jBPM, u.å.).

Activiti: Activitis säkerhetsegenskaper har inte kunnat identifierats på grund av bristfällig dokumentation, trots noggranna efterforskningar och kontaktförsök med Delgado et al. (2015). Däremot är det inte sagt att Activiti saknar säkerhetsegenskaper (Activiti, u.å.).

JBPM5: Detta verktyg tillämpar samma säkerhetsmetoder som jBPM (jBPM, u.å.).

6.12 Sökfunktioner (eng. Searching mechanisms)

Fritt översatt från engelska kan begreppet "sökfunktion" definieras som "en databas som hjälper människor att hitta information på Internet utifrån ord eller villkor som anges av användare" (Silveira, 2002). Sökfunktioner är komplexa internetverktyg som ofta går att finna på sökwebbplatser. Google driver den kanske mest använda sökwebbplatsen i världen. Mer exakt används webbplatsernas sökfunktioner exempelvis för att söka efter, lagra och presentera information, samt för att kommunicera med andra användare. Sökning av omfattande informationsmängder blev möjligt genom att servrar kopplades samman via Internet. När servrarnas information var publik tilläts överföringar av stora mängder information, samt delning av grafiska resurser och interaktivitet (Hafner & Lyon, 1996).

Bonitasoft: Sökfältet i Bonita Portal kan användas för att söka efter specifika värden, filtrera dem och visa resultatet. Observera att sökningarna i Bonita Portal är beroende av de söknycklar som konfigurerats i Bonita Studio. Om inga söknycklar har angetts under processdesignen så kommer inga sökresultat att visas. Söknycklarna översätts genom ett databasindex i Bonita Engine. För närvarande kan utvecklaren som designar i Bonita Studio skapa max 5 nycklar eller index per process. Söknycklarna visas i en lista över specificerade sökcase. I varje sökcase ges närmare detaljer över sökt information. I denna användning av söknycklar kan utvecklaren lägga till dynamisk och företagsspecifik information som inkluderas i sökningarna (Bonitasoft, u.å.).

Petals: Detta verktyg tillhandahåller en sökpanel som används för att söka innehåll över hela webbplatsen. Resultaten visas sedan i Googles app "OnTheFly". Genom att klicka på det sista sökalternativet i sökpanelen kommer användaren till en panel för detaljerad sökning. På varje söksida finns även en vänsterkolumn med hierarkisk vy (Petals, u.å.).

Apache ODE: Trots omfattande eftersökningar var det inte möjligt att identifiera några sökegenskaper för Apache ODE. Det är dock inte omöjligt att verktyget ändå har någon form av sökfunktion. (Apache, u.å.).

jBPM: Verktøget använder en JPA (Java Persistence API-specifikation) version 2 för att tillåta användare att konfigurera vilken data som ska användas för runtime-data. Denna specifikation ger instruktioner om hur användaren kan konfigurera en datakälla genom

undersystemet JPA på applikationsservern JBoss. Denna konfiguration tar hjälp av en persistence.xml-fil för att konfigurera datakällan och drivrutinen i applikationsserverns standalone.xml. Vidare instruerar API-specifikationen hur användare kan konfigurera andra applikationer med hjälp av JPA. Installatören kan automatisera dessa processer (som kan bestå i att kopiera rätt filer till rätt plats efter installationen). Verktuget använder även en MySQL-databas för att snabbt skapa och lagra användare som kan ingå i sökningarna (jBPM, u.å.).

Activiti: I Activitis uppgiftsprogram kan användaren filtrera efter processinstanser utifrån särskilt angivna formulärfältsvärden och datumintervall. Användaren kan då se alla matchande sökresultat. Användaren ges även tillgång till processöversikten för att kunna granska de filer och den information som sänts. Det verkar som att Activitis nativa funktioner endast tillåter enkla sökningar (Activiti, u.å.).

JBPM5: För detta verktyg gäller samma sökfunktioner som moderversionen jBPM (jBPM, u.å.).

6.13 Tillståndsmekanism för användare (eng. Permission Mechanism for users)

Användartillstånd ges genom en sträng i inloggningsförfrågan eller ett API-anrop. När användaren exempelvis loggar in i sin app via Facebook så kan användaren komma åt en delmängd av en annan persons data som lagras på Facebook. Vanligtvis involverar denna process plattformar med integrerade tillståndsmekanismer. Under tillståndprocessens gång informeras användaren om potentiella risker i olika säkerhets- och säkerhetsbeslut (Felt, Egelman, Finifter, Akhawe & Wagner, 2012).

Bonitasoft: Bonita-portalen, eller applikationer som använder Web REST API, gör det möjligt för användaren att få tillgång till fördefinierad information. Den information som användaren ges tillgång till bestäms utifrån användarens profil. Plattformens tillståndsmekanismer säkerställer att användarna endast får tillgång till de fördefinierade resurserna. Det betyder till exempel att en användare med en grundläggande användarprofil inte kan utföra handlingar som är avsedda för administratören. För att komma åt en given resurs måste en användare alltså ha en viss användarrättighet (Bonitasoft, u.å.).

Petals: I Petals är det möjligt att skapa en tillståndsmekanism genom en routermodul som ansvarar för auktoriseringen. I denna process skickas ett meddelande till en JBI-tjänst (Java Business Integration), som kontrollerar att användaren (som hör till en fördefinierad grupp av användare) är auktoriserad. Detta är möjligt med hjälp av (JAAS-tekniken) Java Authentication and Authorization Service. (Oracle, 2018) som fungerar likt ett säkerhetstillägg till JBI-meddelandet. Den standardmässiga implementeringen av JAAS-auktorisering sinloggningsmodulen baseras på en enkel konfigurationsfil. Utvecklare kan konfigurera routermodulen och definiera en egen behörighetsloggmodul,

vilken ger vissa användare tillgång till datalagringsenheten exempelvis LDAP (Petals, u.å.).

Apache ODE: Utgångspunkten är att användaren samtidigt är administratören. Däremot kan administratören dela ut olika accessvillkor och därmed olika möjligheter att komma åt och hantera data, vilket gör att olika användare ges olika tillgång till data (Apache, u.å.). Verktöget inkluderar servrarna JBoss och Apache Webserver, vilka hanteras av JBoss ON (Operation Network) och beskrivs närmare av Red Hat (2018a).

jBPM: Utifrån Martinez (2016) är det upp till företagets utvecklare att välja system för utdelning av användarrättigheter. Det går att integrera externa system som "Wildfly security management" eller Keycloak, och dessa system har olika stöd för olika former av användarrättigheter. Ett annat alternativ är att bygga ett eget säkerhetssystem som stödjer önskvärda tillståndsmekanismer eller användarrättigheter. Däremot poängteras i Business Process Incubator (2016) att de senare versionerna av jBPM har nativa säkerhetsfunktioner med integrerade tillståndsmekanismer. Med andra ord kan användaren av de senare versionerna av jBPM, exempelvis jBPM 7, installera verktöget och därefter snabbt administrera användare, grupper och användarrättigheter utifrån ett enkelt och intuitivt användargränssnitt (jBPM, u.å.).

Activiti: Utifrån de dokumentökningar som gjorts på Internet, framförallt genom söktermen "Permission Mechanism for users Activiti" ges inga sökträffar som ansluter till Activiti (Activiti, u.å.).

JBPM5: De genomsökningar som gjorts av bloggar, användarguider och böcker on-line visar att JBPM5 tycks följa samma principer som jBPM vad gäller säkerhet och användares access till olika informationsnivåer. Med andra ord verkar JBPM5 i grunden ha samma funktioner som jBPM vad gäller användarrättigheter och liknande tillståndsmekanismer. (jBPM, u.å.).

6.14 Dokument för granskning av BPM-verktygens egenskaper

Nedanstående dokument låg till grund för ovanstående empiri (se även bilaga 1). Uppsatsens kommande analys baserades också på dessa dokument, tillsammans med tidigare forskningsmaterial.

Bonitasoft: <https://documentation.bonitasoft.com/bonita/7.6/>

Apache ODE: <http://ode.apache.org/userguide/>

jBPM och jBPM5: <http://www.jbpm.org/learn/documentation.html>

Petals: <https://doc.petalslink.com/display/welcome/Petals+Documentation>

Activiti: <https://www.activiti.org/docs>

7 Resultat och Analys

I detta kapitel jämförs BPM-verktygens egenskaper och i vilka sammanhang dessa egenskaper är värdefulla.




Genom att jämföra de OSS-baserade BPM-verktygens egenskaper från föregående empirikapitel framträder likheter och skillnader mellan verktygen. I likhet med Delgado et al. (2015) jämförs dessa likheter och skillnader utifrån en tregradig skala. På denna tregradiga skala kunde verktygen ha (1) fullständigt stöd (2) delvist stöd eller (3) otillräckligt stöd för egenskapen. Denna jämförelse illustreras i tabell 1

I andra hand jämfördes även verktygen utifrån tre nivåer av integration. I denna jämförelse kunde egenskapen vara (a) nativ, där egenskapen är en integrerad del av verktyget (b) delvist nativ, där mjukvara behöver utvecklas för att ge fullt stöd för egenskapen, eller (c) ha integrationsbehov, där komponenter från tredje part behöver installeras för att uppnå egenskapen. Denna jämförelse redovisas i text paragrafer efter tabell 1.

Med andra ord granskas *om* vissa egenskaper stöds samt *hur* egenskaperna stöds. Denna bedömning gav nedanstående tabell.

Tabell 4. Utvärdering av BPM-system. Källa (författare egen).

	Bonitasoft	Petals	Apache ODE	jBPM	Activiti 13	JBPM5
Stöd för BPMS-arkitektur	Green	Green	Yellow	Yellow	Yellow	Yellow
Integration sociala nätverk	Green	Green	Yellow	Green	Yellow	Green
Modelleringsfunktioner	Green	Green	Green	Green	Green	Green
Stödspråk	Green	Yellow	Yellow	Green	Yellow	Green
Processövervakning	Green	Green	Green	Green	Red	Green
Installationspaket	Green	Yellow	Yellow	Yellow	Yellow	Yellow
Tid på marknaden	Green	Green	Green	Green	Yellow	Green
Communityaktivitet	Green	Yellow	Green	Green	Yellow	Green
Stöd för mobila enheter	Green	N/A	N/A	Green	Yellow	Green
Stöd för arbetsflödesmönster	Yellow	N/A	Yellow	Green	Green	Green
Integritet och dokumentssäkerhet	Yellow	Green	Yellow	Green	N/A	Green
Sökfunktioner	Green	Yellow	N/A	Green	Yellow	Green
Tillståndsmekanism för användare	Green	Green	Green	Green	N/A	Green

-  (1) Fullständigt stöd, verktyget har de egenskaper som efterfrågas.
-  (2) Delvist stöd, verktyget tillfredsställer vissa specifikationskrav.
-  (3) Otillräckligt stöd, verktyget har inte egenskaperna.

Tabell 4 visar att verktygen har olika *stöd för BPMS-arkitekturen*. Närmare bestämt visade dokumentgranskningen att Bonitasoft och Petals har ett flexibelt och heltäckande stöd som är nativt och oberoende av tredje part-komponenter eller vidareutvecklingar. Även Apache ODE, jBPM, Activiti och JBPM5 är välanpassade för BPMS-arkitekturen, men kräver vanligtvis vidareutvecklingar, extra installationer eller integration av ytterligare komponenter för att vara heltäckande i sitt stöd för BPMS-arkitekturen.

Verktygens *integration med sociala nätverk* granskades på två sätt. Dels kan vissa verktyg rent tekniskt sammankopplas direkt med sociala media och nätverk, dels har verktygen separata kommunikationskanaler på sociala media. Utifrån dokumentmaterialet tycks Bonitasoft, Petals och jBPM vara integrerade på båda sätten. Resterande verktyg verkar enbart ha separat närvaro på sociala media, men sakna teknisk integration med sociala nätverk.

De *modelleringsfunktioner* som BPM-verktygen utgår från har olika komplexitet respektive användarvänlighet. Bonitasoft, Petals, jBPM och JBPM5 har välutvecklade modelleringsstyper som tillåter både enkla och komplicerade lösningar. Användare utan programmeringskunskaper kan därför använda vissa av dessa modelleringsverktyg, vanligtvis genom enkla visuella formulär snarare än genom komplicerad kodning. Dokumentationen för Apache ODE och Activiti och deras modelleringsfunktioner är otydlig, vilket gör det svårt att avgöra hur väl de stöder modellering. Dock verkar Apache ODE ha många versioner som ständigt uppdateras, medan Activiti tycks ha stabila och välutvecklade, webbaserade modelleringsverktyg för bland annat processhantering och affärsanalys. Av dessa anledningar tycks samtliga BPM-verktyg ha ett välutvecklat stöd för modellering, och detta stöd tycks vara oberoende av tredje part-lösningar.

Verktygen baseras på flertalet *stödspråk*, det vill säga de programmeringsspråk som verktygen i grunden baseras på och är kompatibla med. Exempelvis är Bonitasofts stödspråk framförallt programmeringsspråken Java, Angular JS, medan jBPM och JBPM5 framförallt stöds på Java, HTML, Shell, JavaScript och XSLT. Gemensamt för samtliga verktyg är att de stöds på Java. I denna analys ges BPM-verktygen med många stödspråk bedömning på den tregradiga skalan, eftersom de därigenom antas vara mer flexibla och anpassade för fler funktioner.

Utifrån dokumentgranskningen tycks samtliga verktyg utom Activiti ha nativa funktioner för *processövervakning*. Dessa tycks vara visuellt tilltalande, ha hög användarvänlighet och flexibla genom att de enkelt kan modelleras och anpassas till företagets olika behov. Ett exempel på detta är Bonitas processhanterare, där användaren kan klicka direkt på en listikon för att se olika processfall och diagram över dessa.

Verktygens *installationspaket* och enkelheten att ladda ner verktygens skilda delar i ett enda paket bör inverka på företagets vilja att snabbt börja använda verktyget. Utifrån dokumentationen framgick att enbart Bonitasoft kan laddas ned direkt på datorer med

operativsystemet Windows, Mac eller Linux. Petals kräver en mer komplicerad form av nedladdning genom zip-installation, medan resterande program kräver att användaren dessutom använder olika former av Java-program för att kunna installeras. Av dessa anledningar bedöms Bonitasoft ha bäst stöd för installation i jämförelse med resterande verktyg.

Verktygens *tid på marknaden* bedöms främst utifrån hur länge deras produkter varit tillgängliga på marknaden. De bedöms även utifrån deras marknadsnärvaro vad gäller ungefärligt antal marknader de representeras på och hur väl de penetrerat dessa marknader. Dock förekommer många samarbeten mellan verktygen och andra etablerade digitala nätverk, vilket gör det svårt att exakt avgöra hur väletablerade de är enskilt på marknaden, utan samarbeten. Av praktiska skäl läggs därför, kanske felaktigt, den största vikten vid hur länge verktygen existerat på marknaden. Utifrån denna bedömning har Bonitasoft och Apache ODE existerat i uppemot 20 år och har dessutom en förhållandevis hög global marknadsnärvaro. jBPM och JBPM5 har funnits i 12 respektive 7 år men tycks ändå ha en hög marknadsnärvaro. Petals och Activiti har funnits i 12 respektive 8 år och tycks ha lägre marknadsnärvaro än övriga verktyg.

Samtliga verktyg, undantaget Petals och Activiti, har omfattande *community-aktiviteter* där personer kan registrera sig som medlemmar samt bidra till andra medlemmar eller få stöd från professionella verktygsteam. En genomgående skillnad är att de större verktygen erbjuder professionella team som tycks få någon form av kompensation för att upprätthålla verktygens community, vilket verkar öka deras community-service rejält.

Vad gäller *stöd för mobila enheter* så som det beskrivs av Saccol, Schlemmer & Barbosa (2010). Så ges fullt stöd av Bonitasoft, jBPM och JBPM5. Anledningen är att dessa verktyg mycket enkelt tillåter nedladdning av fil eller app direkt på den mobila enheten, och därefter är användaren igång. Activitis funktionalitet för sin mobila version tycks vara omfattande. Däremot tycks denna funktionalitet ges via tredje part-program, vilket gör att verktyget bedöms ha enbart delvist stöd för mobila enheter. Petals och Apache ODE kunde ej bedömas på grund av svårigheter med att hitta dokumentation över denna egenskap.

Verktygens hade olika grad av *stöd för arbetsflödesmönster*. Som framgår av dokumentationen (Red Hat, 2018b) tycks jBPM, JBPM5 och Activiti ha ett omfattande och nativt stöd för arbetsflödesmönster. Dock baseras JBPM-verktygen på ett system medan Activiti upprätthålls genom ett helt annat XML-baserat system. Bonitasoft har ett omfattande stöd som dock kräver manuell installation (Patterns Initiative, 2017), medan det utifrån dokumentationen är oklart hur omfattande och nativt Apache ODE stödjer arbetsflödesmönster. Bonitasoft och Apache bedöms därför delvis stödja arbetsflödesmönster. Dokumentationen för Petals är oklar för denna egenskap och ges därför ingen bedömning.

Eftersom *integritet och dokumentsäkerhet* täcker så många aspekter av verktygens funktioner (Kenwood Moving, 2018) skulle det krävas en mycket omfattande

dokumentation. Eftersom denna dokumentation tycks baseras på så skilda standarder för de olika verktygen blir en regelrätt jämförelse mellan verktygen vansklig. Trots allt föreslår dokumentationen att jBPM och JBPM5 har ett brett och väldefinierat stöd för integritet och dokumentsäkerhet. Bonitasoft, Petals och Apache har skilda säkerhetsfunktioner som kräver mer ingående kunskaper för att bedöma hur omfattande och effektiva de är. Activitis säkerhetsegenskaper kunde inte bedömas på grund av saknad dokumentation.

Verktygen tycks genomgående ha gott stöd för sökfunktioner. Dessutom verkar stöden vara nativa. Verktygens sökfunktioner bygger dock på olika system, där exempelvis Bonitasoft bland annat baseras på Bonitasofts Engine, medan JBPM-verktygen bland annat baseras på Java, JBoss och MySQL. På grund av sådana olika lösningar blir en direkt jämförelse mellan verktygen som vanligt inte helt oproblematiske. Däremot pekar dokumentationen på att Bonitasoft och JBPM-verktygen har välutvecklade söksystem som kan skräddarsys till företagets specifika önskninagar. Även Petals och Activiti tycks ha integrerade sökfunktioner, men det är oklart om de når upp till samma nivå som Bonitasoft och JBPM vad gäller flexibilitet, detalj och omfång. De bedöms därför ha delvist stöd för sökfunktioner, alternativt skulle de kanske kunna bedömas ha fullt stöd. Dokumentation för Apache ODEs sökfunktioner saknas i denna undersökning.

Vad gäller *tillståndsmekanismer för användare* tycks samtliga verktyg ha utvecklat lösningar som liknar varandra, vad gäller utdelning av accessnivåer och användarrättigheter som ger användare olika grad av tillgång till information. Dessa funktionaliteter tycks numera vara nativa och standardmässiga för samtliga verktyg. Ett undantag kan vara Activiti som har mycket knapphändig information om denna egenskap.

8 Diskussion

I detta kapitel sammanfattas och diskuteras analysresultatet. Dessutom diskuteras resultatets vetenskapliga aspekter, samhällseliga aspekter och etiska aspekter.

8.1 Resultatdiskussion

För att besvara uppsatsens frågeställning *Hur stöder open source-verktyg centrala egenskaper för BPM-system* så utvärderade denna uppsats hur konkurrenskraftiga OSS-baserade BPM-verktyg stöder centrala egenskaper som är värdefulla för storföretags affärsverksamhet.

Sammanfattningsvis föreslår uppsatsens resultat att enbart Bonitasoft ger fullt stöd för samtliga granskade egenskaper. Detta resultat går i linje med Delgado et al. (2015) påstående att Bonitasoft ger stöd för flest egenskaper bland de OSS-baserade BPM-verktygen. Analysen visade även att jBPM och JBPM5 har stöd för många egenskaper men tycks brista i stödet för BPMS-arkitektur och enkel installation. Även detta resultat går relativt väl i linje med Delgado et al. (2015) som betonar att jBPM och JBPM5 stödjer de flesta egenskaperna. Det går även i linje med Bernroider & Bernroider (2008) som pekar på den höga utvecklingsgraden i jBPM i jämförelse med andra BPM-verktyg. Däremot visade uppsatsresultatet att Petals inte stödjer de flesta egenskaperna, medan Delgado tvärtom hävdade att Petals stödjer de flesta egenskaperna.

De verktyg som tycks erbjuda minst stöd för de egenskaper som granskats är Petals och Activiti. Utifrån dokumentgranskningen gav dessa verktyg intrycket av att vara förhållandevis små och nyetablerade, med fåtalet egenskaper. De kan därför tänkas vara intressanta för företag som behöver enklare lösningar med färre egenskaper (Bernroider, E. & Bernroider, M., 2008), i synnerhet vad gäller modelleringsfunktionerna. En annan observation är att verktygen med hög community-aktivitet även har hög marknadsnärvaro. Framförallt tycks tiden verktygen funnits på marknaden korrelera med graden av Community närvaro. Exempel på detta är Bonitasoft och Apache ODE.

För att ytterligare jämföra uppsatsens analysresultat med Delgados analysresultat går det att utgå från egenskapen *installationspaket*, eller "*Installation Packages*" som Delgado et al. (2015, s.87) uttrycker egenskapen. Utifrån uppsatsresultatet så är enbart Bonitasoft enkelt att installera, medan resterande BPM-verktyg kunde laddas ned enbart efter förinstallation av funktioner för zip-nedladdning eller diverse Java-program. Även Delgado et al. påstår att flertalet verktyg (exempelvis JBPM och Apache ODE) inte är enkla att installera. Däremot hävdar Delgado et al. (2015) att Petals och JBPM5 är installationsvänliga, utan att närmare motivera varför. Utifrån uppsatsens använda dokument kräver Petals nedladdning via zip-fil medan JBPM5 kräver förinstallation av Java JDK 1.5+. jBPM tillhandahåller visserligen en länk för att hämta och installera Java, samt har en "förenklad installation" och en komplett "full installation" som kan installeras via en arkivfil ("Installer-full.Zip"). Med andra ord framstår inte Petals och JBPM5 som installationsvänliga utifrån uppsatsens analys.

Sammantaget tycks det alltså finnas många likheter och vissa olikheter mellan uppsatsens resultat och Delgado et al. (2015) i tolkningarna av hur väl de OSS-baserade BPM-verktygen stödjer vissa egenskaper. Det kan finnas fler skäl till dessa olikheter. En anledning kan vara att uppsatsen undersökte enbart ett urval av de 94 egenskaper som även Delgado et al. (2015) undersökte. En annan anledning kan vara att tolkningen av egenskaperna i slutändan är subjektiva och skiljer sig beroende på person som analyserar stöd för egenskaper. Exempelvis kanske jag tolkade att tvånget till förinstallation av fil för zip-nedladdning innebär att verktyget är svårt att ladda ned, medan Delgado et al. kanske hade tolkat det som en enkel nedladdning. En tredje anledning kan vara att Delgado et al. slutförde sin undersökning 2015 medan denna uppsats slutförs 2018. Ett fjärde skäl kan handla om att uppsatsen utgick från en mer kvalitativ ansats med små urval medan Delgado et al. (2015) baserades på mer kvantitativa data och större urval. Ytterligare en anledning kan vara att uppsatsen baseras enbart på dokument, medan Delgados undersökning delvis baseras på praktisk undersökning med konkreta testprojekt. Ett avslutande skäl kan vara att uppsatsens analys grundas på dokument som inhämtades från informationssidor på Internet, inte minst BPM-verktygens hemsidor, och att dessa dokument är av varierande kvalitet och årgång. Detta metodologiska problem diskuteras nedan. Avslutningsvis kan en parallell dras till Bernroider & Bernroider (2008) som konstaterar att de OSS-baserade BPM-verktyg uppnått en hög utvecklingsnivå i jämförelse med kommersiella motsvarigheterna, i synnerhet vad gäller jBPM. Utifrån uppsatsens analysresultat kan man därför dra slutsatsen att framförallt Bonitasoft och jBPM är starka OSS-alternativ till kommersiella BPM-produkter.

8.2 Om kontextens betydelse

Precis som Sinur, Schulte, Hill & Jones (2012) baseras uppsatsen på en kontext-oberoende jämförelse av OSS-baserade BPM-verktyg. Därför görs ingen djupgående granskning av vilken kontext olika företag kan befinna sig i och vilka olika behov de därför kan ha i sina val av verktyg. Däremot görs en kort anknytning till Delgado et al. (2015) och påståendet att företagets karakteristika starkt inverkar på vilket verktygsval som är optimalt. Närmare bestämt påpekas att varje företag kan uppge vilka egenskaper som är viktigast för deras affärssystem och därefter finna ett verktyg som matchar dessa egenskaper. Vissa företag kanske har stort behov av gott stöd för BPMS-arkitektur och hög integration med sociala nätverk. Utifrån uppsatsens utvärderings tabell 4 blir då Bonitasoft och Petals de mest relevanta alternativen. För en mer kontext-beroende undersökning blir det alltså nödvändigt att vikta in företagets specifika behov i valet av BPM-verktyg. Bernroider och Bernroider (2008) ger tydliga exempel på detta och visar hur vissa egenskaper ges en vikt på några få procent, medan andra egenskaper tilldelas vikter på omkring femtio procent, beroende på sammanhang. Det visar hur stor inverkan sammanhanget eller kontexten kan ha på egenskapernas betydelse. Hänsyn till detta kan behöva tas i tolkningen av uppsatsens generella analysresultat. Samtidigt kan det motivera en kvalitativ innehållsanalys som viktar ett antal företags specifika behov i valet av BPM-verktyg. Det skulle kunna utgöra en framtida studie.

8.3 Metoddiskussion

Uppsatsens metod och analys innehöll vissa styrkor och brister. En styrka låg i att uppsatsen på många sätt efterliknade de undersökningsmetoder och egenskaper som tillämpades av Delgado et al. (2015) och sedan jämfördes resultaten med Delgado et al. På detta sätt vilar uppsatsen på en källa som tycks vara etablerad och erkänd. Samtidigt tillät jämförelserna med Delgado et al. att de egna resultaten kunde kontrasteras mot etablerad forskning.

Studiens frågeställning hade antagligen kunnat besvaras med andra metoder och därmed kanske fått annorlunda analysresultat. Exempelvis kanske en kontextlös kvantitativ snarare än kontextlös kvalitativ innehållsanalys hade genererat mer generella resultat med högre reliabilitet. Å andra sidan borde uppsatsens kvalitativa ansats ha gett mer nyans, djup och validitet åt analysresultatet (Berndtsson et al., 2008). En svaghet i uppsatsens metod är brister i de dokument som låg till grund för granskningen av BPM-verktygens egenskaper eftersom många OSS projekt har begränsad mängd dokumentation. I och med att BPM-verktygen baseras på Open Source och inte är proprietära var dokumentationen över verktygens egenskaper därför ofta bristfällig, otydlig, eller slumpmässigt spridd över Internet. Det medförde att uppsatsresultatet baseras på ett dokumentmaterial vars reliabilitet emellanåt är svårt att bedöma.

Uppsatsen hade även kunnat utvidga analysen genom att vikta egenskaperna mot olika företag med olika behov av affärssystem. Det bör ha skapat en mer verklighetsanpassad analys. Istället följde uppsatsen samma linje som Delgado et al. (2015) samt Lawlis, Mark, Thomas & Courtheyn (2001) och genomförde en mer generisk och kontextlös analys. Uppsatsen saknar även de praktiska aspekter som en tillämpad och testbaserad analys skulle kunna belysa. Man kan därför tänka sig att en praktiskt tillämpad analys som dessutom viktade varje egenskap mot det enskilda företags särskilda behov hade gett djupare och mer avgränsade analysresultat. Mer konkret verkar det möjligt att vidareutveckla denna uppsats genom att välja ut tre företag med olika behov samt tilldela varje egenskap en unik procentsats som ökar eller minskar egenskapens betydelse i relation till resterande egenskaper. Viss ledning skulle här kunna ges från Bernroider och Bernroiders (2008) tabell över de procentsatser som användes för viktning av varje egenskape.

Uppsatsens kontextlösa och teoretiskt baserade studie gav däremot generaliserbara svar som borde kunna utvecklas i valfri riktning, exempelvis genom att rikta uppsatsens analysmetod mot en viss typ av organisation. Mer specifikt skulle det eventuellt kunna handla om att låta uppsatsen fokusera på internationella organisationer som FN, OECD och Världsbanken. Fokus skulle även kunna ligga på kommersiella internationella företag inom ett visst marknadssegment. En mer omfattande studie skulle eventuellt kunna jämföra organisationer från två sådana marknadssegment och utläsa skillnaderna i behov av BPM-verktyg. En sådan mer riktad analys skulle kunna vara ett ämne för vidare forskning.

8.4 Samhälleliga aspekter

BPM-verktyg tycks ha blivit högaktuella för omvärlden, där organisationer kräver kontinuerliga förbättringar i sina affärssystem för att lättare uppnå sina mål av koordination, flexibilitet och kontroll (Delgado et al., 2015). Dessa verktyg verkar kunna skapa nya möjligheter åt organisationer, där strategi och planering bättre kan prioriteras. Användningen av sådana verktyg tycks även ge företag överraskande inblickar i sina processer. Av dessa anledningar har vissa forskare rentav hävdats att användningen av dessa verktyg kan vara otillåten, men denna användning är svår att bevisa (Jeston & Nelis, 2008). Av just dessa anledningar tycks kritiken mot BPM-verktygen ha växt fram. Det tycks däremot inte finnas några direkta lagar som skulle kunna sätt stopp för användandet av BPM-verktyg. Även om det existerar sådana lagar, verkar det inte finnas många företag som använder dessa BPM-verktyg. I detta sammanhang poängterade uppsatsens tidigare etikdiskussion det eventuellt oetiska i att implementera dessa verktyg av dessa skäl. Ett motsatt argument skulle kunna betona det oetiska i att inte introducera OSS-baserade BPM-verktyg i större skala, eftersom dessa kan antas öka världens totala välfärd och nytta. Visserligen kan tänkas att vissa tjänster försvinner, men samtidigt lär det uppstå nya tjänster som är specialiserade på att upprätthålla BPM-systemen. Denna utfasning av utdöende ineffektiva system och introduktion av mer effektiva system måste ses som en nödvändighet för att ett samhälle ska utvecklas. Eftersom denna studie fokuserade på frågeställningen *Hur stöder open source-verktyg centrala egenskaper för BPM-system* så kan denna fråga därför vara öppen för diskussion även i framtiden.

Med tanke på den senaste tidens utveckling vad gäller användningen och tekniken för BPM-verktyg, med dess Open Source-baserade communities, kan man tänka sig att nästa stora förändring inom området kommer vara att användandet sprids till de små och medelstora företagen i Sverige. De stora, etablerade organisationerna har antagligen redan en dedikerad IT-avdelning som på något vis använder BPM-system. Här är tankegången att de större organisationerna sitter på resurserna att integrera nya BPM-system. Å andra sidan lär större organisationer vara mer trögrörliga och långsamma inför förändringar. Det talar för att små och medelstora företag snabbare kan anpassa sig till tekniska förändringar, inklusive anpassningen till nya BPM-system. Man kan även tänka sig att ju enklare det blir att integrera och använda dessa verktyg, desto mer kan programmen användas och skapa arbetstillfällen för IT-kunniga. I de fall verktygen verkligen blir enkla att integrera i företagets verksamhet kan man tänka sig att de mindre företag som snabbast anpassar sig till de nya förhållandena ges en fördel gentemot övriga företag. Hur utvecklingen kommer gå i verkligheten får framtiden utvisa.

8.5 Vetenskapliga aspekter

Uppsatsens bidrag till forskningen kan ligga i att BPM-verktygen utvärderades utifrån en kvalitativ snarare än kvantitativ jämförelse. En sådan jämförelse bidrog till en mer beskrivande och nyansrik förståelse för inte bara i vilken grad egenskaperna skiljer sig åt men även hur och varför. Uppsatsens mer kontextlösa sammanhang bidrog dessutom till en större möjlighet att generalisera analysresultaten. Eftersom de kvantitativa metoderna visserligen är bra på att göra graderande jämförelser men saknar förmågan att mer exakt beskriva hur skillnaderna mellan verktygen fungerar kan man tänka sig att uppsatsens kvalitativa aspekt skulle kunna fungera som ett värdefullt bidrag till forskningen inom detta område.

8.6 Etiska aspekter och framtida studier

De etiska aspekterna i denna studie har redan diskuterats utifrån vetenskapsrådets fyra forskningsetiska individkrav (Vetenskapsrådet, 2002). Diskussionen visade att denna teoretiska och kontextlösa studie per automatik inte överträder några av dessa individkrav, vilket nämns i ovanstående diskussionen om resultatets vetenskapliga aspekter (sektion 8.3). En mer avgränsad eller riktad analys hade kunnat vara ett ämne för vidare forskning. En annan sak som nämnts är möjligheten att vidareutveckla denna uppsats genom att vikta egenskaperna mot konkreta företag och deras olika behov.

Ytterligare ett förslag för vidare forskning skulle kunna vara att undersöka fler än de 13 egenskaper som granskades i denna uppsats. På grund av uppsatsens begränsade resurser gick det inte att matcha de 94 egenskaper som granskats av Delgado et al. (2015), varav bara tjugofyra presenteras av Delgado et al. i det granskade forskningsmaterialet (ibid., s.84). Det skulle dessutom kunna vara ett intressant bidrag till forskningen att granska ytterligare OSS-baserade BPM-verktyg. En fallstudie med företagsintervjuer med personer som arbetar med BPM-system skulle också kunna vara en intressant fortsättning på denna uppsats. Det förutsätter dock att fler svenska företag använder sig av BPM-system.

Referenser

- Aalst, W.M.P., Hofstede, A. & Weske, M. (2003). *Business Process Management: A Survey. Lecture Notes in Computer Science*. Berlin Heidelberg: Springer-Verlag.
- Adesola, S. & Baines, T. (2005). Developing and Evaluating a Methodology for Business Process Improvement. *Business Process Management Journal*, 11(1), 37–46.
- Apache ODE. (u.å.). *User guide*. Tillgänglig: ode.apache.org/userguide.
- Bryman, A. (2011). *Samhällsvetenskapliga metoder*. Malmö: Liber.
- Baldam, R., Valle, R., Pereira, H., Hilst, S., Abreu, M. & Sobral, V. (2009). *BPM-Business Process Management*. (2 ed.) Sao Paulo: Erica.
- Bernroider, E.W.N. & Bernroider, M. (2008). *A Comparative Study of Business Process Management Tools based on Open Source Software and a Commercial Reference*. Conference paper: 5th International Conference on Cybernetics and Information Technologies, Systems and Applications (CITSA 2008).
- Berndtsson, M., Hansson, J., Olsson, B. & Lundell, B. (2008). *Thesis Projects: A Guide for Students in Computer Science and Information Systems*. London: Springer.
- Biba, K. J. (1977). *Integrity considerations for secure computer systems*. Deputy for Command and Management Systems. Bedford, Massachusetts.
- Bonitasoft. (u.å.). *Bonitasoft*. Tillgänglig: documentation.bonitasoft.com/bonita/7.6.
- Business Process Incubator. (2016). Security management in jBPM & Drools workbenches. Tillgänglig: businessprocessincubator.com/content/security-management-in-jbpm-drools-workbenches.
- Chandy, M. & Mcgoveran, D. (2004). *The Role of BAM. Business Integration Journal: Business Process Management Supplement*, s. 19–21.
- Chang, J. F. (2006). *Business Process Management System. Strategy and Implementation*. Boca Raton: Auerbach Publication.
- Davenport, T. H. (1994). Saving IT's Soul: Human Centered Information Management. *Harvard Business Review*, 72(2), 119–131.
- Delgado, A., Calegari, D., Milanese, P., Falcon, R. & Garcia, E. (2015). *A Systematic Approach for Evaluating BPM Systems: Case Studies on Open Source and Proprietary Tools*, IFIP Advances in Information and Communication Technology.
- Denscombe, M. (2009). *Forskningshandboken - för Småskaliga Forskningsprojekt inom Samhällsvetenskaperna*. Lund: Studentlitteratur AB.
- Duhon, B. (1998). It's All in our Heads. *Inform*, 12(8), 8–13.

- Dumas, M., La Rose, M., Mendling, J. & Reijers, H.A. (2013). *Fundamentals of Business Process Management*. Berlin Heidelberg: Springer-Verlag.
- Feller, J. & Fitzgerald, B. (2002). *Understanding Open Source Software Development*. Australia: Pearson Education.
- Felt, A. P., Egelman, S., Finifter, M., Akhawe, D. & Wagner D. (2012). *How to Ask For Permission*. University of California, Berkeley.
- Fettke, P. (2008). Business Process Modeling Notation. *Wirtschaftsinformatik*, 50(6), s. 504–507.
- Free Software Foundation. (2007). *GNU general public license*. Hämtad 10-02-2018: gnu.org/licenses/gpl.html.
- Hafner, K. & Lyon, M. (1996). *Where wizards stay up late: the origins of the internet*. New York: Simon & Schuster Paperbacks.
- Hammer, M. (2015). What is Business Process Management? In: J.V. Brocke & M. Rosmann (eds.), *Handbook on Business Process Management*. Berlin Heidelberg: Springer-Verlag.
- Harrington, H. J., Esseling, E. K. C. & Nimwegen, H. V. (1997). *Business process Improvement Workbook: Documentation, Analysis, Design and Management of Business Process Improvement*. New York: McGraw Hill.
- Jalali, S. & Wohlin, C. (2012). Systematic Literature Studies Database Searches vs. Backward Snowballin. Karlskrona: Blekinge Institute of Technology.
- jBPM. (u.å.). *Documentation*. jbpm.org/learn/documentation.html.
- Jeston, J. & Nelis, J. (2008). *Business process Management, practical guidelines to successful implementations*. E-bok.
- Kenwood Moving. (2018). *What is Document Security and Storage and Why is it Important?* Tillgänglig: kenwoodmoving.com/what-is-document-security-and-storage-and-why-is-it-important.
- Lawlis, P., Mark, K., Thomas, D. & Courtheyn, T. (2001). A Formal Process for Evaluating COTS Software Products. *IEEE Comput.* 34(5), 58–63.
- Ling, C. & Xin, L. (2009). Achieving Business Agility by Integrating SOA and BPM Technology. In: *International Forum on Information Technology and Applications*. Chengdu.
- Liang, Q. T. (2008), An Improved Framework of Business Process Management System which integrating the strategy management, 15th Annual Conference Proceedings. *IEEE*, 256–261.

- Marconi, M. A. & Lakatos, E. M. (2010). *Fundamentos de Metodologia científica*. (7. ed.) São Paulo: Atlas.
- Marshall, M.N. (1996). Sampling for qualitative research. *Family Practice*, 13(6), 522–526.
- Martinez, R. (2016). *Drools and jBPM*. Tillgänglig: blog.athico.com/2016/04/user-and-group-management-in-jbpm-and.html.
- Muller, C. J. (2003). *Modelo de gestao Integrando Planejamento estrategico, Sistema de avaliacao de desempenho e gerenciamento de processos*. Universidade Federal do Rio Grande do Sul.
- Oates, B. J. (2006). *Researching Information Systems and Computing*. London: Sage.
- Open Source Initiative (2006) – *The Open Source Definition (Annotated)*. Tillgänglig: opensource.org/docs/definition.php.
- Oracle. (2018). *Java SE Security*. Tillgänglig: oracle.com/technetwork/java/javase/tech/index-jsp-136007.html#close
- Patton, M. Q. (2002). *Qualitative Research & Evaluation Methods*. 3. ed. California: Sage Publication, 2002.
- Perens, B. (1999). The Open Source Definition. In: *Open Source. Voices from the Open Source Revolution*. Tillgänglig: oreilly.com/openbook/opensources/book/perens.html.
- Petals. (u.å.). *Petals documentation*. Hämtad: doc.petalslink.com/display/welcome/Petals+Documentation.
- Päivärinta, T. & Munkvold, B. E. (2005). Enterprise Content Management: an Integrated Perspective on Information Management. In: *Annual Hawaii International Conference on System Science*.
- Rademakers, T. (2012). *Activiti in Action*. New York: Manning Publications Co.
- Ravesteyn, P. & Batenburg, R. (2010). Surveing the Critical Success Factors of BPM-systems Implementation. *Business Process Management Journal*, 16(3), 495.
- Raymond, E. S. (1999). A Brief History of Hackerdom. In: *Open Source: Voices from the Open Source Revolution*. Tillgänglig: oreilly.com/catalog/opensources/%20book/raymond.html.
- Red Hat. (2018a). Customer Portal: *User Authorization*. Tillgänglig: access.redhat.com/documentation/en-us/jboss_enterprise_brms_platform/5/html/brms_getting_started_guide/chap-user_authorization.

- Red Hat. (2018b). *Customer Portal: Chapter 6. Advanced Process Modeling*. Tillgänglig: access.redhat.com/documentation/en-us/red_hat_jboss_bpm_suite/6.4/html/user_guide/chap_advanced_process_modeling.
- Russell, N., van der Aalst, W.M.P. & ter Hofstede, A.H.M. (2016). *Workflow Patterns: the definite guide*. Tillgänglig: mitpress.mit.edu/books/workflow-patterns.
- Ryan, K. L. K., Lee, S. S. & Lee, E. W. (2009). Business process management (BPM) standards: a survey. *Business Process Management Journal*, 15(5), 744–791.
- Saccol, A., Schlemmer, E. & Barbosa, J. (2010). *M-learning e u-learning - Novas perspectivas da aprendizagem móvel e ubíqua*. Person ed. São Paulo.
- Silveira, M. (2002). *Web marketing: usando ferramentas de busca*. São Paulo: Novatec.
- Sinur, J., Schulte, W., Hill, J. & Jones, T. (2012). *Magic Quadrant for Intelligent Business Process Management Suites*. Gartner Inc.
- Skrinjar, R. & Trkman, P. (2013) Increasing Process Orientation with Business Process Management. *International Journal of information Management*, 33, 48–60.
- Smith, H. Fingar, P. (2007). *Business Process Management: The Third Wave*. Meghan-Kiffer Press.
- Stallman, R. (1999). The GNU Operativ System and the Free Software Movement. In: *Open Sources: Voices from the Open Source Revolution*. Tillgänglig: oreilly.com/openbook/opensources/book/stallman.html.
- Strauss, A. & Corbin, J. (2008). *Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory* (3rd ed.). Thousand Oaks, CA: Sage.
- Thompson G., Seymour L.F. & O'Donovan B. (2009). *Towards a BPM Success Model: An Analysis in South African Financial Services Organisations*. Berlin Heidelberg: Springer-Verlag.
- Trkman, P. (2010). The Critical Factors of Business Process Management. *International Journal of Information Management*, 30(2), 125–134.
- Tsalgatidou, A. (1998). Selection criteria for tools supporting business process transformation for electronic commerce. In: *Proceedings of EURO-MED NET*.
- Verner, L. (2004). BPM: The Promise and Challenge. *ACM Queue*, 2(1).
- Vetenskapsrådet. (2002). *Forskningsetiska principer inom humanistisk-samhällsvetenskaplig forskning*.
- Weber, B. Mutschler, B. & Reichert, M. (2009), Investigating the Effort of Using Business Process Management Technology: Results from a Controlled Experiment, *Science of Computer Programming*, 75(5), 292–310.

WFMC. (2018). Workflow Management Coalition. Tillgänglig: wfmc.org.

White, S. A. (2006). *Introduction to BPMN*. IBM.

Workflow Patterns Initiative. (2017). The Workflow Patterns Home Site. Tillgänglig: workflowpatterns.com.

Bilaga 1-Dokument använda för granskning av BPM-verktygens egenskaper

Bonitasoft: <https://documentation.bonitasoft.com/bonita/7.6/>

Mycket omfattande dokumentation på flera hundra sidor (beräknat utifrån ett standardmässigt A4-format, teckenstorlek 12 och enkelt radavstånd) som bedöms vara heltäckande för Bonitasofts egenskaper, exempelvis vad gäller installation, applikationsfunktioner, konfiguration, administration och utveckling.

Apache ODE: <http://ode.apache.org/userguide/>

Dokumentation som bedöms motsvara hundratalet sidor vilka behandlar installation, användning och specifikationer för Apache ODE.

jBPM och jBPM5: <http://www.jbpm.org/learn/documentation.html>

Mycket omfattande dokumentation på flera hundra sidor som inte bara täcker installation, applikationsfunktioner och övriga verktygsegenskaper, men som även behandlar konfiguration, utveckling av mjukvara, communities och annan kringverksamhet.

Petals: <https://doc.petalslink.com/display/welcome/Petals+Documentation>

Dokumentation som behandlar installation, konfiguration, underhåll och övriga egenskaper som täcker Petals. Användarguiden bedöms bestå av hundratalet sidor. Dessutom tillkommer en guide för utvecklare på 43 sidor samt diverse komponentguider.

Activiti: <https://www.activiti.org/docs>

Dokumentation som i sitt grundutförande består av omkring 50 sidor och behandlar verktygets egenskaper samt ger användaren guidning i att installera och använda verktyget. Flertalet sidor innehåller dessutom länkar till fördjupande information.