Stadsplan

Projekt
"Stadsplan för lärosäte"
Slutrapport

Innehåll
Inledning
Sammanfattning
Rekommendation och fortsatt arbete
Rekommendation
Bakgrund
Arbetsgång i projektet
Stadsplan för Lärosäte
Beskrivning metamodellen stadsplan
Inledning
Vad är en metamodell
Funktion och förmåga
Metamodell för stadsplan
ArchiMate
Översättningsproblematik – ArchiMate
Definitioner metamodellen
Kärnarkitekturen – funktioner och tjänster
Funktioner
Tjänster
Gemensamma nivåer inom sektorn
Metamodellens indelningar
Nivåer – detaljering av arkitekturer
Arkitekturer
LiU-POC MinIT-tjänst
Förslag på tekniker för att fylla på metamodell
Funktionsarkitekturen
Tjänster/process
Informationsarkitekturen
Applikationsarkitekturen
Bilaga 1 – Detaljering av applikationsarkitekturen, förslag
Bilaga 2 – Metamodell
Bilaga 3 – Förslag på funktionsarkitekturen
Bilaga 4 – LiUs Min-IT

Inledning

Arbetet med att ta fram en metamodell har varit oerhört givande och otroligt utmanande. Arbetet är inte på ngt sätt avslutat genom denna rapport utan behöver fortsätta att utvecklas. Arbetet har gett insikt i många problemställningar men också insikter i ArchiMate som jag inte trodde var möjligt. Min förhoppning är att arbetet skall inspirera arkitekter runt om i landet och öka förståelsen för hur verksamheten fungerar i relation till IT och en ökad förståelse för EA.
Jag vill tacka Jörgen Dahlberg för hans insikter och stöd i detta arbete. Utan Jörgen hade detta arbete inte gått att genomföra!
Jag vill också tacka Johan Peterson som har stått ut med en massa frågor och även lagt mycket tid för att förklara LiUs lösning och arbetet med att översätta deras systemlista till att bli ett utkast på applikationsarkitektur.
Jag vill också ge en eloge till referensgruppen som faktiskt har sett att detta arbete är ngt som är viktigt och som går att använda på respektive lärosäte.
Jag vill tacka Per Hörnblad för stöd och koordinering.
Vill också tacka GU och Chalmers för att jag har fått möjligheten att driva detta arbete och att de har värdesatt mitt deltagande i projektet.
Sist vill jag tacka alla medarbetare som jag varit i kontakt med angående detta arbete och att de stått ut med alla olika förklaringar av metamodellen.
Tack!!
Ola Ljungkrona
IT-arkitekt - GU

Sammanfattning

Länk till wiki där rapporten och fortsatt arbete läggs upp: Stadsplan för lärosäten
En av grundpelarna i en arkitektur är att kunna kommunicera till olika aktörer inom verksamheten, såsom ledningsgrupper, verksamhetsrepresentanter och specialister. Alla har olika behov av att kunna förstå landskapet. För att kunna få helheten att fungera ihop måste det gå att härleda från strategisk nivå till operativ nivå. Idag är digitalisering en viktig drivkraft för statlig förvaltning eller liknande verksamheter och för att stödja digitaliseringen behövs en gemensam struktur att förhålla sig till för att hitta nya tjänster och produkter, samt kunna göra en beskrivning av befintliga tjänster och produkter.
En viktig slutsats som en direkt konsekvens av hur metamodellen är uppdelad i nivåer är att det bör vara möjligt att definiera nivå 1 och nivå 2 på nationell nivå. Detta borde förenkla kommunikation mellan lärosäten och även mot andra myndigheter och departement.
I detta projekt har en metamodell tagits fram som kan användas som ett register över olika komponenter i verksamheten och hur de samspelar. Delarna i registret kan sammanställas i olika vyer för att förklara hur ett initiativ förhåller sig till den befintliga strukturen. Exempel på några olika användningsområden:

1. Övergripande systemkarta
2. Begrepp och informationsdefinitioner
3. En beskrivning av vad verksamheten gör
4. Vilka nyttor stödjer verksamheten
5. Ett funktionsperspektiv som snabbt beskriver verksamheten
6. Funktionshierarkin stödjer riksarkivets klassificeringsstruktur för att definiera handlingsslag
7. Hitta återkommande tillämpningar
8. Jämföra olika lärosätens lösningar för att hitta potentiella samarbetsområden
9. En beskrivning på vad som behövs för att starta ett universitet
10. Härleda strategiska initiativ till påverkan på befintlig arkitektur
11. Livcykelhantering på applikationstjänster

Stadsplanen ger ett långsiktigt stöd för att lärosäten lättare skall hitta samarbeten och gemensamma lösningar samt hur samarbeten kopplat till en molnstrategi kan se ut.
Stadsplan täcker inte heller in alla behov av perspektiv. Dock så täcker den in de flesta komponenter inom verksamheten. För att visa hur komponenterna relaterar till varandra utanför vad som kan beskrivas i metamodellen krävs ytterligare modeller. Dock så bör att andra modeller hämta information om komponenterna baserat på hur de har klassificerats i metamodellen.

Rekommendation och fortsatt arbete

Arbetet med metamodellen och denna rapport är en bra början för att beskriva hur ett lärosätes verksamhet är strukturerad. Arbetet inom detta projekt är en överflygning av området och varje lärosäte behöver själva driva detaljeringsarbetet vidare baserat på metamodellen för stadsplan. Det finns delar i stadsplan som behöver fortsätta drivas på nationell nivå. Arbetet med att utöka den befintliga referensarkitekturen för integration med andra referensarkitekturer är en viktig del i att utveckla stadsplanen. Lämpliga områden som bör bedrivas och utvecklas på nationell nivå:

- Applikationsarkitekturen – standarddetaljering av tillämpningar
- Integrationskatalog – utveckla befintlig katalog med en gemensam begreppsmodell
- Förtydliga referensarkitekturen för integration
- Funktionsarkitekturen - standarddetaljering av funktioner nivå 1 & 2
- Informationsarkitekturen – standarddetaljering av nivå 1 &2
- Processarkitekturen – standarddetaljering av nivå 1 &2

Den förslagna detaljeringen på applikationsarkitekturen (applikationsområden i referensarkitekturen) behöver också kompletteras och gås igenom ytterligare.
Metamodellen för stadsplan är en bra början på ett register över hur verksamheten kan beskrivas på ett enhetligt sätt. Ett register är en grundförutsättning för att börja arbeta strukturerat med arkitektur. Att arbeta fram ett register från början är en inte helt enkel uppgift. Dessutom så skall registerstrukturen kunna vara utformad så att den bör kunna förstås av andra lärosäten för att kunna identifiera och effektivisera samarbeten. För att klara av detta behövs en gemensam modell över registret – vilket metamodellen erbjuder.
Ett annat arbete som bör drivas vidare är ta fram en modell för förvaltning som stödjer stadsplan. Ett visst arbete har gjort inom projektet men detta behöver belysas mer. Många lärosäten använder idag PM3 som förvaltningsmodell. Eftersom PM3 delar in förvaltningsobjekten i vilka system (applikationer i metamodell stadsplan) som finns inom lärosätet och utgångspunkten för indelning av verksamheten i stadsplan är via funktion, behövs en annan modell för att anpassa förvaltningsmodellen till stadsplanstänket.

Rekommendation

Det viktigaste är att utnyttja redan befintlig arbete genom att använda redan gjorda process-, system- och informationskartor. Det som behöver göras är att göra ett arbete med att sortera in dessa entiteter i metamodellens struktur.
Förslag på i vilken ordning det går att börja:

1. Övergripande nivå 1 och nivå 2 för hela stadsplanen
2. Detaljera funktionsarkitekturen
3. Detaljera applikationsarkitekturen
4. Detaljera informationsarkitekturen

Ovan lista är bara en idé och inget måste. Den baseras lite på vad det är för arbete som de flesta lärosäten redan har börjat med eller har en relativt bra uppfattning vilka delar som finns i verksamheten. För andra som arbetat mycket processkartläggning är det naturligt att mappa in det i stadsplanen och koppla dessa till funktionera.
Den rekommendation projektet gör för att initiera arkitekturarbetet är att börja med att dokumentera verksamheten på hög nivå (N1 & N2) i enlighet med metamodellen samt att lägga in detta i ett register Exempel på produkter är: SparxEA, Archi, ARIS &Alfabet, iServer, m.fl. som klarar av att upprätthålla unika objekt och relationer mellan objekt. Detaljera och definiera sedan funktionsarkitekturen eftersom funktionsarkitekturen kommer vara väldigt stabil för verksamheten. Den kommer inte förändras speciellt mycket och arbetet går direkt att använda som KS KS – Klassifieringsstruktur enligt Riksarkivet RA-FS 2008:4 . Det finns material för att göra funktionsinventeringen på följande plats:
The Capability Canvas
Här används "Capability" med meningen funktionella förmågor, dvs. Funktion i metamodellen.
Har lärosätet redan börjat inventera rekommenderas att befintligt arbete mappas in enligt metamodellen. Det borde inte påverka befintligt arbete speciellt mycket då de flesta inventeringar inte är gjorda på Nivå1 och nivå2 utan ofta på en lägre nivå.
Nästa steg är för att förbereda inför detaljering av informationsarkitekturen och det är att detaljera applikationsarkitekturen. Använd stadsplanens förslag på indelning som förslag.
Detaljera sedan vidare Informationsarkitekturen på nivå 3 och nivå 4 i första hand på informationsobjekt och verksamhetsobjekt som hanteras i integrationer. Det handlar inte om att göra fullständiga informationsmodeller utan att göra en klassificering i hierarkisk mening. Utifrån detta kan sedan ett informationsägarskap på nivå 2 enligt metamodellen sättas upp. Detta ligger väl i linje med det arbete som görs inom masterdataarbetet inom ATI. Tänk på att göra arbetet iterativt och börja där verksamheten har ett initiativ eller problem.
Inspiration om hur informationsarkitekturen kan populeras finns i kapitlet Förslag på tekniker för att fylla på metamodell.

Bakgrund

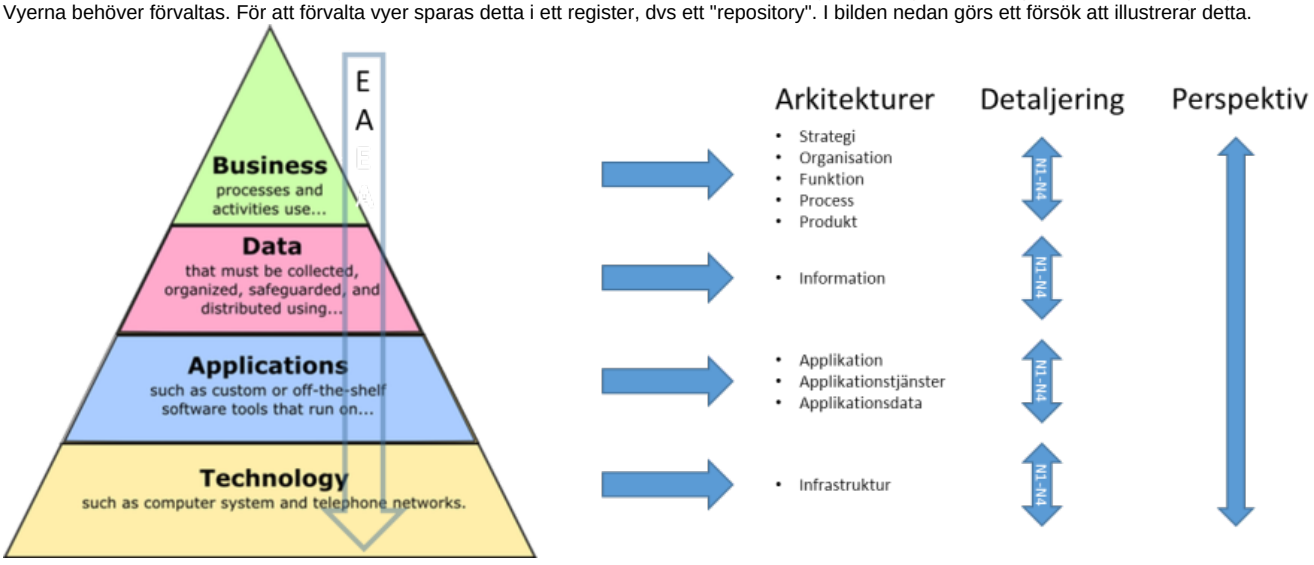
Tanken med projektet var att ta fram en stadsplan ur ett tekniskt perspektiv med system och integrationer i fokus för ett typiskt lärosäte inom högre utbildning i Sverige. Vilka system samt typiska integrationer finns för universitet och högskolor och hur ser de viktiga informationsflödena ut? Vilka typiska källsystem och målsystem finns? Vilka likheter finns och var finns skillnader bland olika lärosäten? Hur kan en tänkt plan kopplat en molnstrategi se ut för olika delar i IT-arkitekturen?
Stadsplanen ger ett långsiktigt stöd för att lärosäten lättare skall hitta samarbeten och gemensamma lösningar samt hur samarbeten kopplat till en molnstrategi kan se ut.
En av grundtankarna med projektet är att hitta potentiella samarbetsområden samt att kunna vara ett stöd i hur ett lärosäte kan beskriva sina ingående komponenter.
Projektplan finns att läsa här:
Projektplan - Stadsplan för lärosäte

Arbetsgång i projektet

Projektet har jobbat med att ta fram en metamodell över ett lärosäte och sedan detaljera den med betoning på Applikationsarkitekturen och Infrastrukturarkitekturen. LiU har använts som ett exempellärosäte. Projektgruppen har haft ett antal WS och fört en kontinuerlig dialog för att säkerställa resultatet.
Projektet har bestått av en projektgrupp och en referensgrupp enligt nedan.
Projektledare:
Ola Ljungkrona, Chalmers
Projektresurser:
Jörgen Dahlberg, JD Konsult
Johan Peterson, LiU
Referensgrupp:
Ingvar Andersson, Chalmers
Joakim Nejdeby, LiU
Stefan Edholm, SLU
Projektägare:
Per Hörnblad - Inkubator projektkoordinator, UmU

Stadsplan för Lärosäte

Utgångspunkten för att beskriva ett lärosäte är att definiera en struktur där sambanden mellan de olika arkitekturerna inom en verksamhet kan beskrivas. Varje arkitektur innehåller ett antal entiteter. Relationer mellan entiterna skär över de olika arkitekturerna och på så sätt fås en beskrivning av hur verksamheten är strukturerad. Stadsplanen är inte en komplett karta utan kan snarare ses som en överflygning av verksamheten som ger stöd för en viss detaljering. Stadsplan ger en ingång på ett register över verksamheten. Detta brukar kallas "repository" Repository är ett vedertaget begrepp som finns definierade i flera arkitekturramverk såsom TOGAF. Vintergatan är ett annat exempel som IRM definierar. IRM har även begreppet Stadsplan och kan ses som en instansering av vintergatan mot ett visst mål. inom olika arkitekturramverk.
En verksamhet och dess olika komponenter beskrivs ibland som en EA-pyramid. Tanken med att illustrera det som en pyramid är att alla komponenter måste finnas och ha en betydelse i systemet. Varje lager i en pyramid representerar en arkitektur. Varje arkitektur måste kunna detaljeras. För att beskriva ett visst samband mellan delarna i varje arkitektur och även samband mellan arkitekturerna görs detta genom olika perspektiv. Dessa perspektiv blir vyer. Vyerna behöver förvaltas. För att förvalta vyer sparas detta i ett register, dvs ett "repository". I bilden nedan görs ett försök att illustrerar detta.



Figur 1 Enterprisearkitektur kopplat till olika arkitekturer

För att beskriva de olika komponenterna och dess samband i arkitekturen har en metamodell tagits fram. Metamodellen gör det möjligt att beskriva en verksamhet på en högre nivå och täcker inte in alla detaljeringar, men är ett bra ingångsvärde för att beskriva verksamheten. Genom att använda metamodellen och dess regelverk kan ett lärosäte relativt snabbt beskriva sin verksamhet. I arbetet med stadsplanen har ett förslag på detaljering i vissa arkitekturer gjorts som kan användas som en mall för att beskriva ett lärosäte. Förslag på detaljering har gjorts i funktions- och applikationsarkitekturen samt det tidigare arbete som är gjort inom referensarkitekturarbetet inom ATI.
I rapporten kommer även ett förslag på hur stadsplanen kan populeras och i vilken ordning detta arbete kan göras.

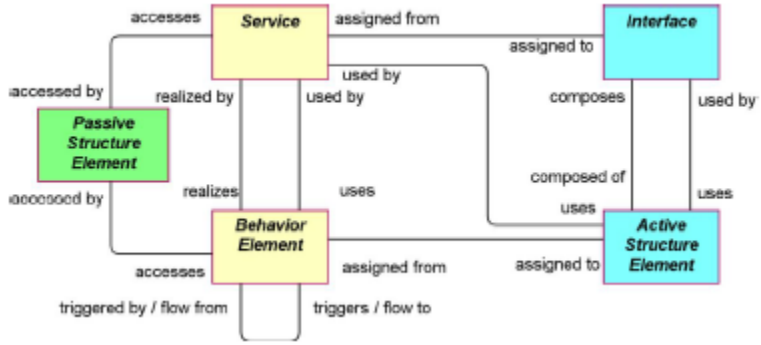
Beskrivning metamodellen stadsplan

Inledning

Grundtanken med en metamodell som bas för att uttrycka stadsplanen för ett lärosäte är att skapa ett gemensamt sätt att beskriva arkitekturen för lärosäten. Genom att använda metamodellens definitioner vid beskrivning av lärosätens arkitektur blir det möjligt att se vilka strukturer och komponenter som är likvärdiga mellan lärosätena.
När lärosäten beskriver sin verksamhet i linje med metamodellen går det att analysera fram vilka potentiella samarbeten som kan skapas med andra lärosäten. Även om fokus för uppdraget är att hitta samarbeten inom IT så behöver IT-stödet sättas i en relation till vilken verksamhet som systemen och infrastrukturen är aktiv inom.

Vad är en metamodell

En metamodell är en generell beskrivning på hur olika generiska element är relaterade till varandra. Med hjälp av en metamodell går det att beskriva en specifik implementation baserat på de element, relationer och regler som definieras i metamodellen. Ett exempel på en metamodell är "core-concept" i modelleringsspråket ArchiMate.



Figur 2 ArciMate v2.1 Core Model
Denna modell kan sedan användas för att beskriva exempelvis hur en specifik process använder/bearbetar information och vilken roll som är tillsatt att utföra arbetet. Det är inte säkert att den specifika implementationen använder alla element i metamodellen.
Stadsplanen är en specialisering av den helt generella metamodellen för att beskriva kontexten stadsplan för lärosäte.

Funktion och förmåga

Två centrala begrepp i metamodellen är Funktion och Förmåga. För att förenkla för läsaren görs här en genomgång på dessa begrepp. Framförallt är det ordet Förmåga som har många olika betydelser i engelskan såsom "Business Skill", "Capability", "Ability", m. fl. Även när ord används i kombination ändras betydelsen. TOGAF referera till "Business Capability" som "Function" och skiljer därmed på "Capability" och "Business Capability".

TOGAF9 defines a Function as:

Function describes units of business capability at all levels of granularity.
The term "function" is used to describe a unit of business capability at all levels of granularity, encapsulating terms such as value chain, process area, capability, business function, etc.
Any bounded unit of business function should be described as a function.
[a Function] Delivers business capabilities closely aligned to an organization, but not necessarily explicitly governed by the organization. Also referred to as "business function".

TOGAF9 defines a Capability as:

A business-focused outcome that is delivered by the completion of one or more work packages.
Using a capability-based planning approach, change activities can be sequenced and grouped in order to provide continuous and incremental business value.

I svenskan har begreppet förmåga har en tvetydig användning som ibland kan ses som funktion och ibland som förmåga. I version 3 av ArchiMate finns stöd för att använda både funktion (Business Function) och förmåga (Capability). I metamodellen lutar vi oss på ArchiMates definition.

ArchiMate v3 defines a Capability as:

[A capability represents an ability that an active structure element, such as an organization, person, or system, possesses.]
In the field of business, strategic thinking and planning delivers strategies and high-level goals that are often not directly implementable in the architecture of an organization. These long-term or generic plans need to be specified and made actionable in a way that both business leaders and Enterprise Architects can relate to and at a relatively high abstraction level.
Capabilities help to reduce this gap by focusing on business outcomes. On the one hand, they provide a high-level view of the current and desired abilities of an organization, in relation to its strategy and its environment. On the other hand, they are realized by various elements (people, processes, systems, and so on) that can be described, designed, and implemented using Enterprise Architecture approaches. Capabilities may also have serving relationships; for example, to denote that one capability contributes to another.
Capabilities are expressed in general and high-level terms and are typically realized by a combination of organization, people, processes, information, and technology. For example, marketing, customer contact, or outbound telemarketing [4].
Capabilities are typically aimed at achieving some goal or delivering value by realizing an outcome. Capabilities are themselves realized by core elements. To denote that a set of core elements together realizes a capability, grouping can be used.
Capabilities are often used for capability-based planning, to describe their evolution over time. To model such so-called capability increments, the specialization relationship can be used to denote that a certain capability increment is a specific version of that capability. Aggregating those increments and the core elements that realize them in plateaus (see Section 13.2.4) can be used to model the evolution of the capabilities.

ArchiMate v3 defines a Business Function as:

[A business function is a collection of business behavior based on a chosen set of criteria (typically required business resources and/or competences), closely aligned to an organization, but not necessarily explicitly governed by the organization.]
Just like a business process, a business function also describes internal behavior performed by a business role. However, while a business process groups behavior based on a sequence or flow of activities that is needed to realize a product or service, a business function typically groups behavior based on required business resources, skills, competences, knowledge, etc.
There is a potential many-to-many relation between business processes and business functions. Complex processes in general involve activities that offer various functions. In this sense a business process forms a string of business functions. In general, a business function delivers added value from a business point of view. Organizational units or applications may coincide with business functions due to their specific grouping of business activities.
A business function may be triggered by, or trigger, any other business behavior element (business event, business process, business function, or business interaction). A business function may access business objects. A business function may realize one or more business services and may be served by business, application, or technology services. A business role may be assigned to a business function. The name of a business function should clearly indicate a well-defined behavior. Examples are customer management, claims administration, member services, recycling, or payment processing.

Begreppet "Capability"

Definitionerna är inte helt tydliga och framförallt exemplen som ges för exempelvis "Capability" i ArchiMate är inte tydliga för att skapa en förståelse hos läsaren. Inom engelska används, på samma sätt som i svenskan, ett flertal begrepp som används för samma betydelse såsom:

- Capability
- Business Capability
- Business Function
- m. fl.

För att göra en distinktion på begreppet förmåga säger vi att det finns två olika förmågor:

- *Funktionella förmågor -> Business Function -> i Stadsplan – Funktion*
- *Framgångsförmågor -> Capability -> i Stadsplan – Förmåga*

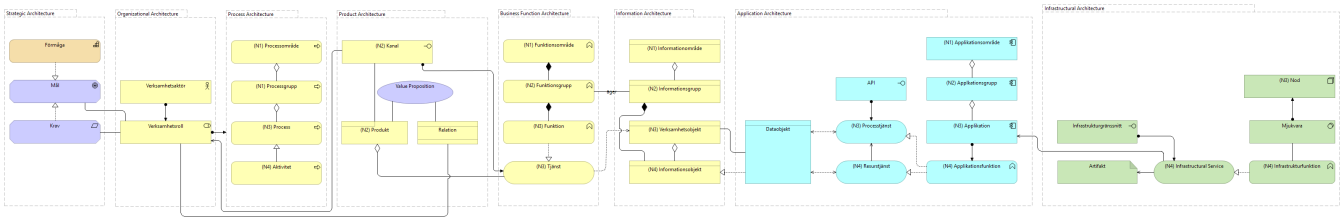
Funktionella förmågor har ett intern perspektiv - det är de förmågor som verksamheten har för att få verksamheten att fungera. Funktionella förmågor/" Business Function" som tas fram är att anse som funktionaliteten som processer behöver för att utföra sitt arbete. Genom att beskriva en verksamhet med förmågor och relatera de till vilka tjänster/produkter verksamheten vill exponera internt och extern kan det visa i en analys att vissa förmågor saknas. I Stadsplan använder vi "Funktion" för begreppet "Funktionella förmågor"
Framgångsförmågor är externa förmågor som realiseras av de funktionella förmågorna. Ett exempel på framgångsförmågor är exempelvis "Vi skall bli världsledande inom ...." eller mer konkret "Vi knyter samman världen bästa forskare inom område elbilsbatteri för att skapa förutsättningarna för innovativ utveckling av räckvidd på elbilar".
I flera av de exempel som lyfts in i rapporten används begreppet "Capability" i betydelsen både som funktionella förmågor och framgångsförmågor. Det är kontexten i modellen som indikerar vilken förmåga modellen avser.

Metamodell för stadsplan

Metamodellen för att beskriva ett lärosäte utgår från ett antal arkitekturer:

- Strategiarkitektur
- Organisationsarkitektur
- Processarkitektur
- Produktarkitektur
- Funktionsarkitektur
- Informationsarkitektur
- Applikationsarkitektur
- Infrastrukturarkitektur

Dessa arkitekturer beskriver hela verksamheten uppifrån förmågor ner till infrastrukturkomponenter. Bilden nedan visar metamodellen i sin helhet. En större bild finns i som bilaga på sid Utöver basarkitekturerna finns även en koppling till mål och krav för verksamheten samt en koppling till kund. Vissa av de begrepp som används i modellen kan verka främmande för den typ av verksamhet som finns inom utbildningssektorn. Exempel kan det verka främmande att uttrycka student som en kund till lärosäte men mönstermässigt så sammanfaller synen på student som en kund till ett lärosäte väl med generella mönster av vad en kund är.
Metamodellen alla arkitekturer visas nedan.



Figur 3 Metamodellen i sin helhet. Metamodellen finns i större format i bilaga 1 på sid
Inom varje arkitektur och i samband mellan arkitekturerna så finns det olika nivåer av detaljering. Nivåerna känns lättast igen genom att titta på en klassisk processkarta där man utgår från processområden som exempelvis stöd-, ledning- och huvud/kärnprocesser för att sedan detaljera vidare ner till lägsta nivån i form av aktiviteter.

För att metamodellen skall bli läsbar finns inte alla tillåtna relationer i ArchiMate utritade i modellen. I flera exempel som följer används relationer som inte syns i metamodellen.

ArchiMate ArchiMate® 3.0 Specification

För att uttrycka metamodellen har projektet valt att göra detta med hjälp av modellspråket ArchiMate. Valet att använda ArchiMate är att modellspråket är relativt spritt inom sektorn och att ArchiMate är väl anpassat för att just uttrycka en högnivåarkitektur. I och med ArchiMate 3.0 täcks behovet väl in för att uttrycka detta.
ArchiMate är inte avsett att användas som det enda verktyget för att uttrycka alla delar i verksamhetens arkitekturer utan som en del av en helhet. Det väsentliga är att det finns en stringens i hur entiteter i arkitekturerna definieras för att kunna använda dem i andra perspektiv oavsett dessa görs i ArchiMate eller annat modellspråk. Genom stringens på entiteterna ökar också friheten i vilka modeller verksamheten kan uttrycka sin verksamhet och vilka analyser på arkitekturen som det finns behov av att göra.

Översättningsproblematik – ArchiMate

Det finns några översättningsproblem med betydelsen i ArchiMate och hur begreppen vanligtvis har för svensk betydelse. Därför är det nödvändigt att göra en tydlig definition av begreppen i stadsplan. Det största problemet ligger i det engelska begreppet "Application" som slarvigt översätts till applikation på svenska. Den korrekta engelska översättningen är tillämpning. Applikation på svenska avser ibland tillämpningen men också ibland produkten eller själva mjukvaran, även en blandning förekommer som exempelvis MS Word – som både är produkten och mjukvaran.
De viktigaste definitionerna som kan anses vara ngt oklara följer här:
Sample Business Capabilities

Definitioner metamodellen

- Application -> (N1-N3) Applikation (den logiska användningen – tillämpningen, se bilaga på sid)
- Application Function -> (N4) Applikationsfunktion (den logiska funktionen Det som faktiskt finns realiserat i applikationen, kräver kod för att finnas; sätta kontoattribut, checka in dokument, hantera kursregistrering, etc)

- Application Software -> Mjukvara (MS Word, Excel, Oracle Database, etc.)
- Business Function -> (N1-N3) Funktion (Funktionell förmåga)
- Capability -> Förmåga (Framgångsförmåga, exempel se Sample Business Capabilities)

(N1-N3) Applikation (ArchiMate Application Component):

Med den logiska användningen menas här tillämpning generellt. I metamodellen är en arkitektur applikationsarkitekturen, dvs applikationsnivå är en abstraktion av underliggande nivå. Jämför "(N3) Inköp", "(N3) Redovisning" etc. är alla en del av "(N2) Ekonomi", som i sin tur är en del av "(N1) Affär".

(N4) Applikationsfunktion (ArchiMate Application Function) – Applikationsarkitektur:

Med den logiska funktionen avses den specifika funktionen som är ofta realiseras av en installerad kod. En kod kan realisera flera Applikationsfunktioner. I metamodellen är det är det "(N4) Applikationsfunktioner" som realiserar "(N3) Processtjänster" och "(N4) Resurstjänster". Exempel på "(N3) Processtjänst" är "(N3) Registrera student på kurs", Exempel på "(N4) Resurstjänst" är "(N4) Hämta Kurstillfälle".

(N1-N3) Funktion (ArchiMate Business Function) –Funktionsarkitektur:

Den interna gruppering av beteendet som krävs för att leverera ett värde i verksamheten. Grupperingen kan typiskt vara "interna förmågor", kompetenser, resurser, kunskaper etc. Kallas även ibland för "funktionella förmågor". Funktion i denna kontext skall **inte** blandas ihop med en organisatorisk funktion. Funktion har inget med organisation att göra alls.

Förmåga (ArchiMate Capabilty) – Strategiarkitektur:

En förmåga metamodellen är en representation av egenskap som övriga komponenter i arkitekturerna behöver ha i en kontext där ett mål skall nås. Ett exempel skulle kunna vara att etablera företaget inom ett nytt produktsegment – vad saknas för komponenter i arkitekturerna för att göra den förflyttningen. I Metamodellen skulle det ge en indikation på vilka "nya" funktioner verksamheten behöver etablera och sedan utifrån dessa funktioner utöka arkitekturerna med nödvändiga komponenter såsom, processer, produkter, applikationer och infrastruktur. Det blir en hjälp att identifiera ett eventuellt gap som den befintliga verksamheten har för att uppnå målet/målen.

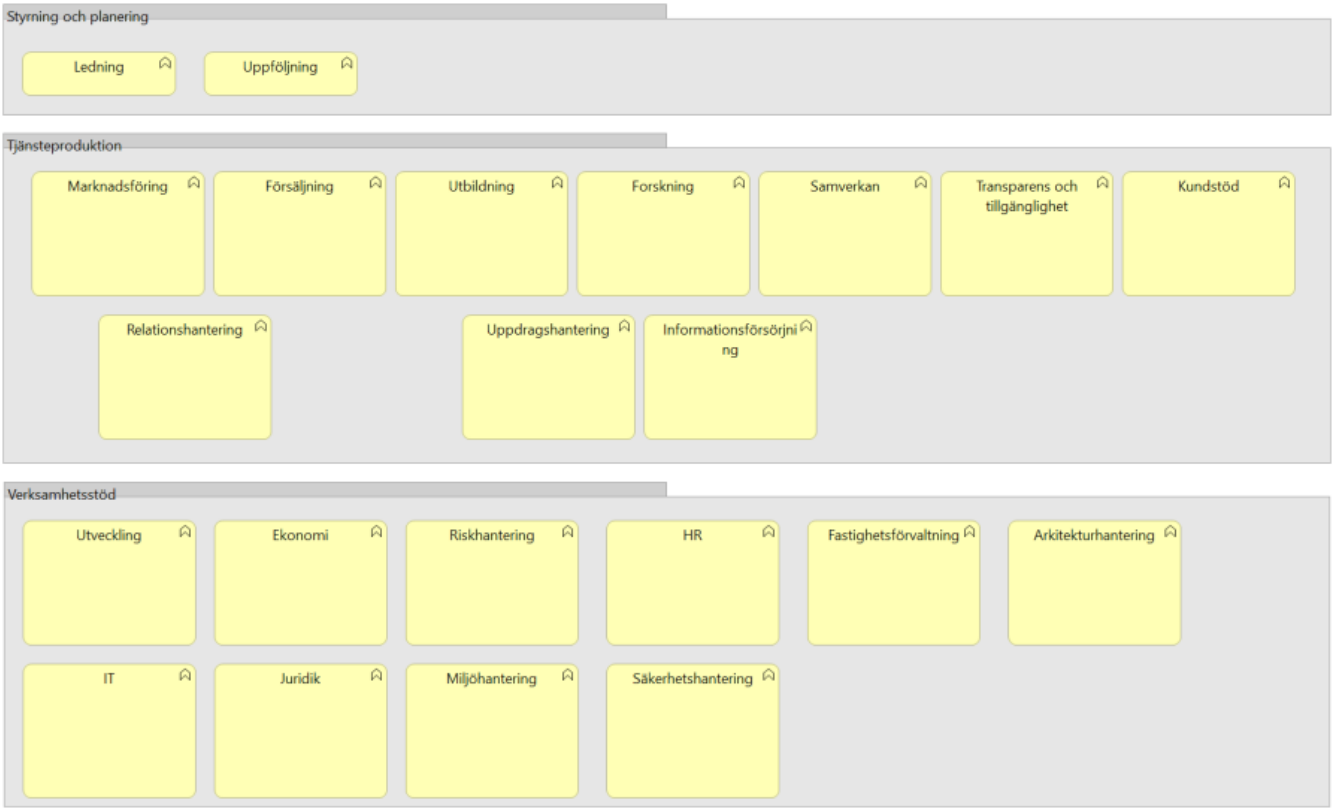
Kärnarkitekturen – funktioner och tjänster

Funktioner

Funktion beskriver på en väldigt övergripande nivå VAD en verksamhet GÖR. Detta i motsats till processer som beskriver HUR en verksamhet GÖR.
För att inte riskera i att hamna i en allt för detaljerad beskrivning av verksamheten har utgångspunkten vart att använda funktioner.
Funktioner har ett internt perspektiv - det är de funktioner som verksamheten har för att få verksamheten att fungera. Genom att beskriva en verksamhet med funktioner och relatera dessa till dessa tjänster och produkter verksamheten vill exponera internt och extern kan en analys påvisa att funktioner saknas alternativt kan återanvändas.
En funktion kan inte själv skapa ett resultat utan det krävs andra komponenter för att funktionen skall kunna leverera ett resultat i verksamheten såsom:

- Processer
- Tjänster
- System
- Aktörer
- Artefakter
- IT-komponenter
- mm.

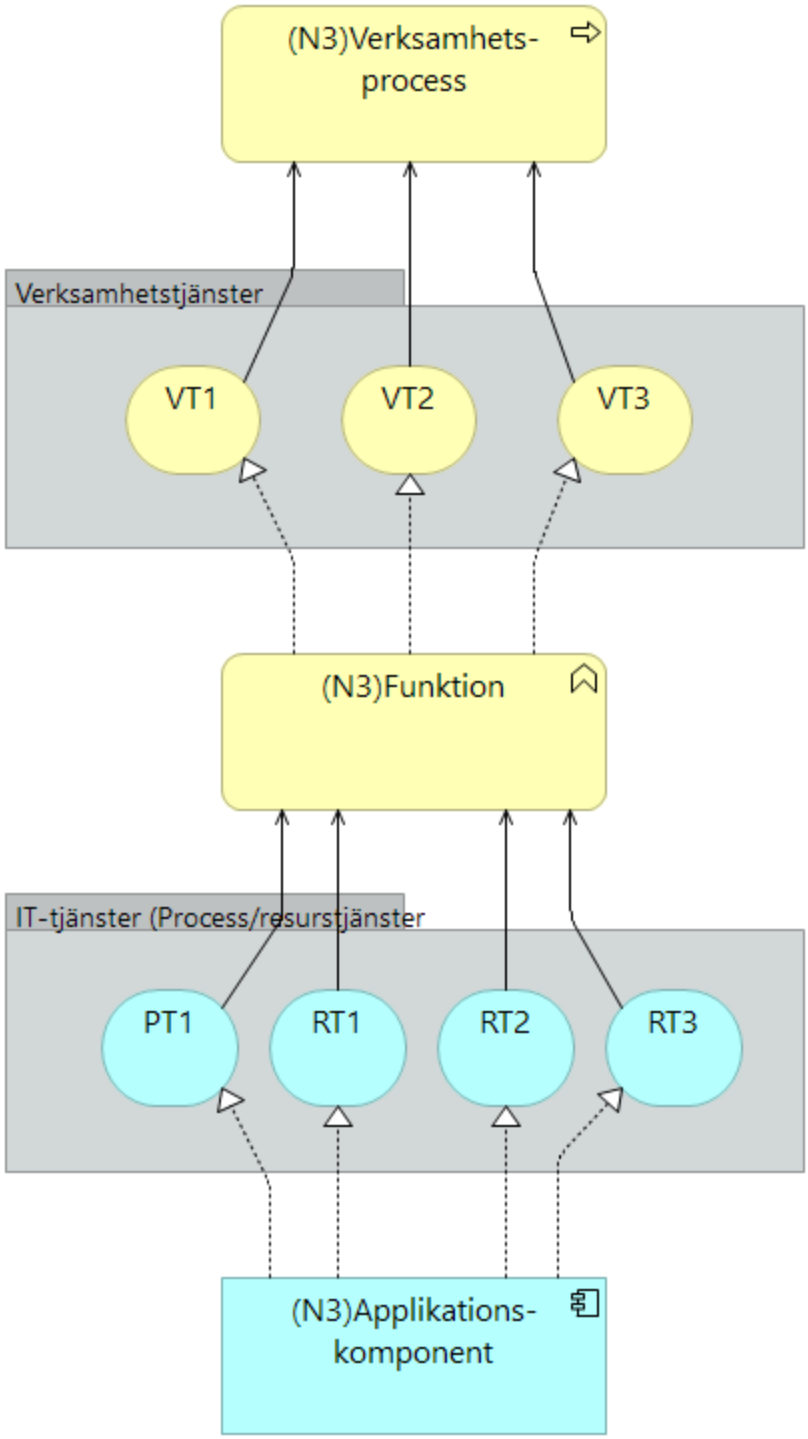
Genom att beskriva verksamheten i termer av funktioner är det inte nödvändigt att detaljera innehållet helt utan det räcker med en övergripande beskrivning av funktionen. Detta kan sättas i relation till att beskriva en verksamhet i processer vilket kan vara väldigt tidsödande och ger ändå inte en helhetssyn av vad verksamheten behöver kunna för att leverera. Det bör gå relativt snabbt (3–4 halvdags fokuserade workshoppar med erfaren facilitator) att skapa en modell över funktionerna inom ett lärosäte. Efter att en modell över funktionerna är på plats kan arbetet fortsätta genom att detaljera verksamheten inom de andra arkitekturerna. Detta skapar snabbt en överblick på verksamheten och ett bra sätt att detaljera vidare där det finns behov av att detaljera – såsom processer som funkar sämre eller upphandling av nya IT-system etc.
Bilden nedan illustrerar en hypotes till funktionskarta som visar funktioner på översta nivån som är aktiv inom högskola- och universitetssektorn. Nivåerna är ett sätt att succesivt öka detaljeringen av funktionerna, där nivå 1 är den högsta nivån och nivå 4 är den lägsta nivån av detaljering inom ramen för metamodellen En mer detaljerad beskrivning av nivåerna finner läsaren i ett senare kapitel på sida . En funktionskarta för nivå 1 och nivå 2 finns i bilaga på sid :



Figur 4 Förslag på funktionskarta nivå 1
En bra post om funktioner (Business Capabilies) finn på:
On Business Capabilities Functions and Application Features

Tjänster

En applikationstjänst (AT) kan användas av många verksamhetsförmågor (VF). Ibland används termen IT-tjänster för applikationstjänster och det förekommer även att IT-tjänster används för verksamhetstjänster som levereras av IT. Därför används endast applikationstjänster (AT) och verksamhetstjänster (VT) inom ramen för detta arbete.
Exempel på en sammanblandning av tjänst begreppet är inom Nya Ladok. Där används termen verksamhetstjänster för produkten Ladoks applikationstjänster. Dessa tjänster representerar givetvis det behov verksamheten har men i metamodellen sorteras de under Applikationstjänster som stödjer en eller flera funktioner. Exempelvis så stödjer Nya Ladoks tjänster olika funktioner inom studieadministrativa funktionen.
Att utgå från tjänster för att beskriva en stadsplan är på samma sätt som att använda funktioner ett sätt att förenkla och reducera detaljnivån i beskrivningen av ett lärosäte. Tjänster är det som en process eller applikation exponerar utåt genom ett gränssnitt. Genom att analysera tjänsterna blir det relativt tydligt vad verksamheten behöver för resurser för att realisera sina förmågor. Det finns både interna och externa tjänster. Externa tjänster är sådana som konsumeras av externa aktörer eller är exponerade utanför sin egen verksamhet. Interna tjänster är de tjänster som funktionerna realiserar. Vissa IT-tjänster har ingen motsvarande verksamhetstjänst utan finns bara för att "hjälpa" andra IT-tjänster. Ett exempel på detta är en integrationsplattform som levererar en IT-tjänst som konsumeras i en integrationskontext.
Tjänster kan levereras av såväl aktörer och processer som av applikationer men också i form av infrastrukturella tjänster. När det gäller applikationstjänster och processtjänster kan man generalisera och säga att det är gränssnitten som skiljer. För processtjänster som inte är automatiserade i ett system är gränssnittet mänskligt medan för processtjänster som är implementerade i ett system är gränssnittet digitalt, exempelvis en hemsida eller liknande. När det gäller infrastrukturella tjänster är det något som användare har ganska svårt att relatera till. Ett tydligt exempel på en infrastrukturell tjänst som användare kommer i kontakt med är sin dator. Med dator som tjänst menas här den fysiska produkten, inte själva användandet av datorn.
För att visa hur tjänster förhåller sig till processer och applikationstjänster kan följande bild vara hjälpsam:

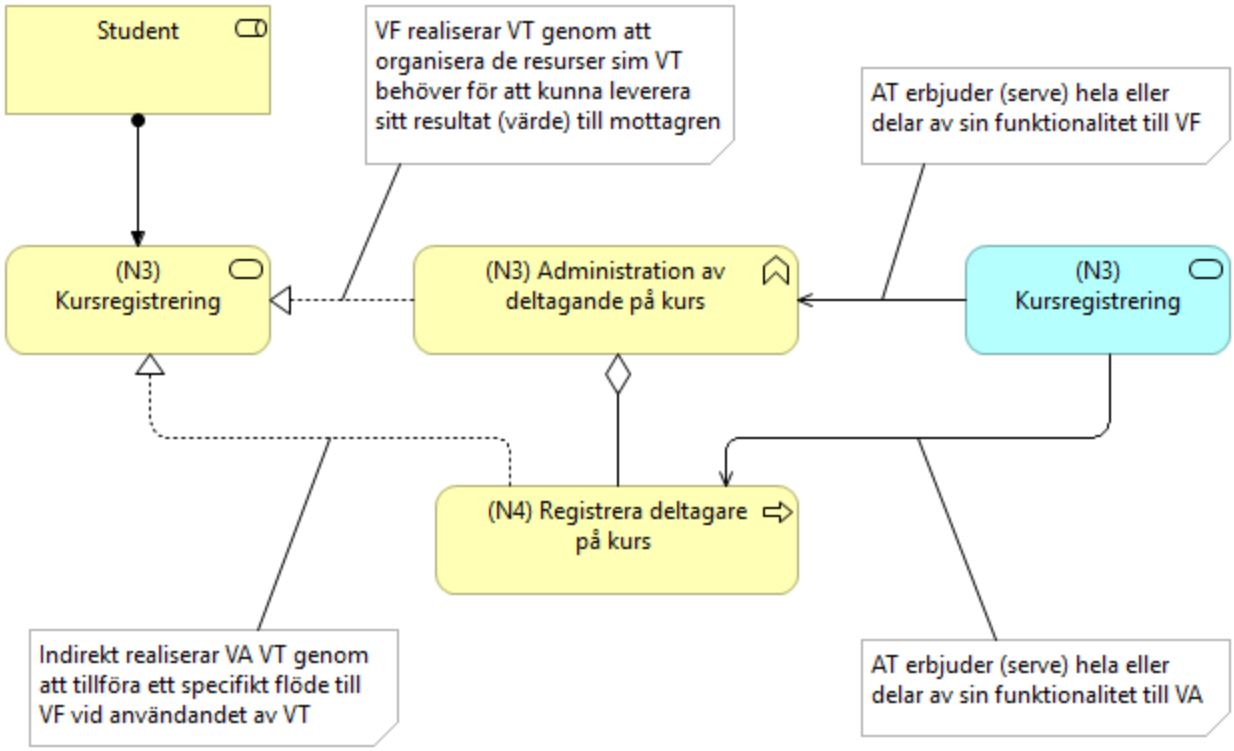


Figur 5 Hur verksamhetstjänster och applikatuionstjänster samverkar

Det är alltså funktionera som realiserar verksamhetstjänsterna (VT), och funktionen (VF) använder sig av IT-tjänster (PT & RT, båda är applikationstjänster, AT) för att göra realisationen. Processerna (VP) använder sig sedan av VT för att skapa en värdekedja. En "(N4) Aktivitet" (VA) som realiserar en VT kan ses som ett specifikt implementerat beteende i en funktion.

Ofta är det så att AT har mer funktionalitet än vad en enkild VF behöver och likaså gäller förhållandet mellan AT och VA.

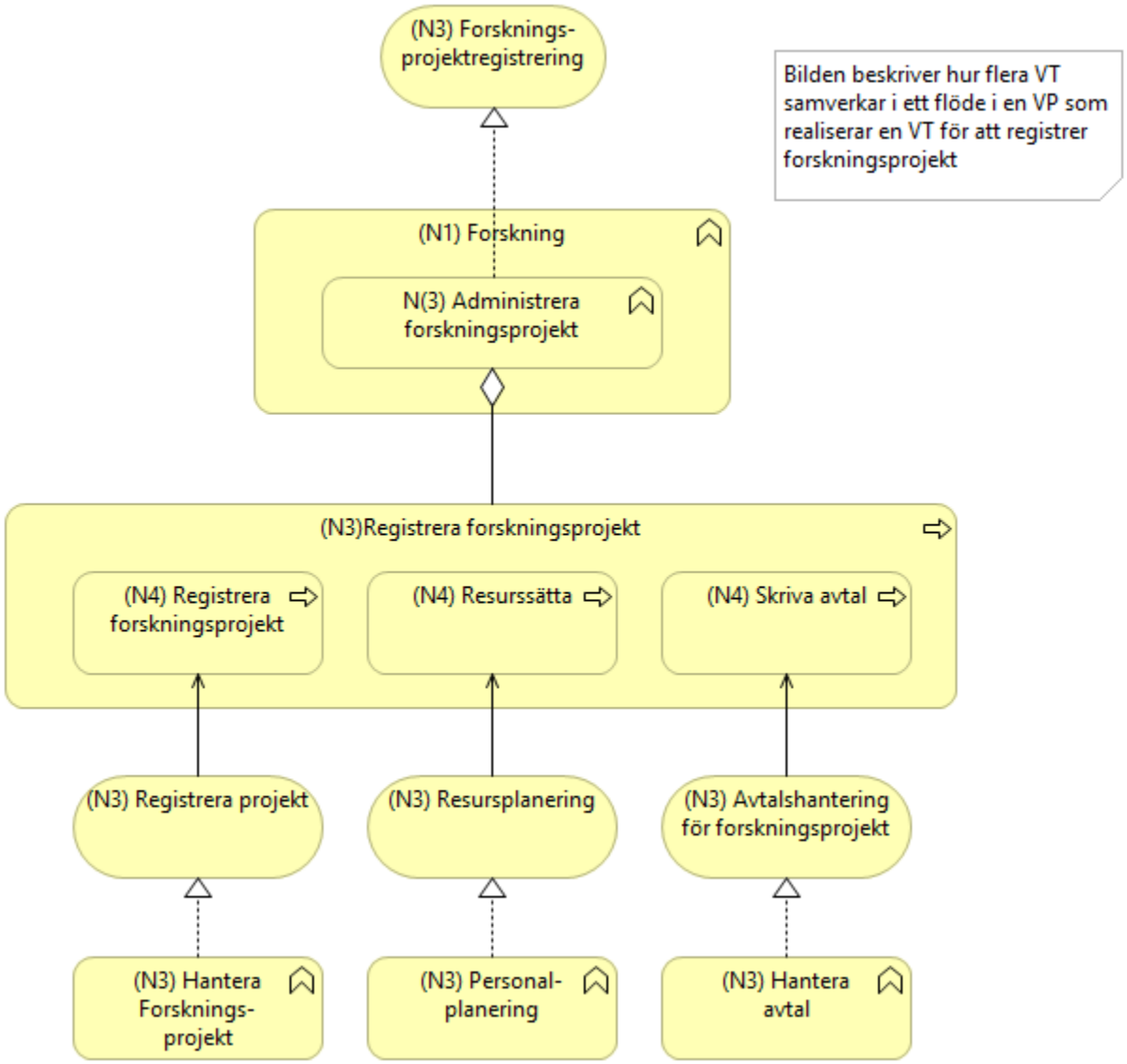
Vad det gäller en VF så har den i många fall behov av flera olika AT för att kunna utföra hela sin funktion och dess tillhörande tjänster.



Figur 6 Både funktion och process kan realisera tjänster men med olika innebörd

I bilden ovan kan realisationsrelationen mellan verksamhetsaktiviteten och verksamhetstjänsten härledas genom att process är en aggregering av funktion som i sin tur realiserar tjänsten. I ArchiMate kallas det för en "derived" relation. Även om relationen går att härledas behövs en tolkning av den relationen och definiera vad den betyder. Bilden ovan försöker göra en sådan definition.

En funktion på verksamhetsnivå exponerar sin funktionalitet via egna verksamhetstjänster. Dessa verksamhetstjänster kan användas av andra verksamhetstjänster på olika sätt. En eller flera verksamhetstjänster kan till exempel koreograferas av en eller flera andra verksamhetstjänster.



Figur 7 Tjänster som koroegraferas för att realisera en annan tjänst

En funktionalitet på applikationsnivå exponerar sin funktionalitet via egna applikationstjänster. Dessa applikationstjänster kan användas av andra applikationstjänster på olika sätt. En eller flera applikationstjänster kan till exempel koreograferas av en eller flera andra applikationstjänster.

Gemensamma nivåer inom sektorn

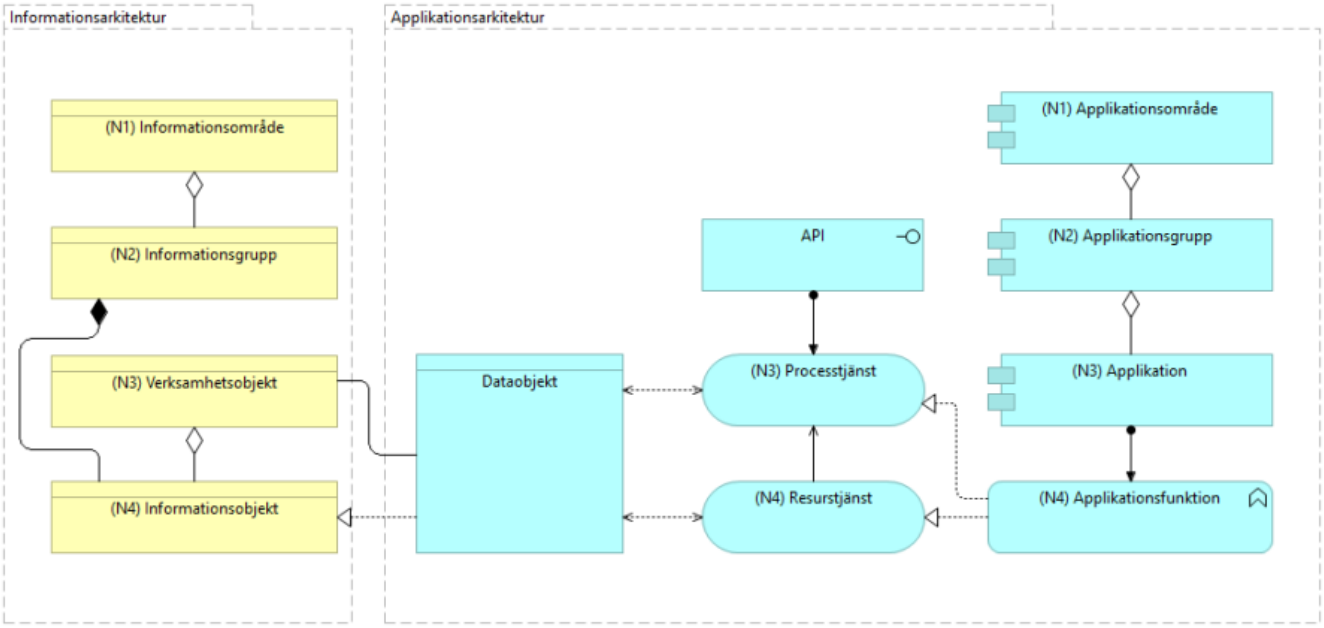
I projektet görs ett antal ansatser om att standardisera vissa nivåer inom metamodellen. I bilagorna finns förslag på indelning ända ner till nivå 3 i applikationsarkitekturen och ner till nivå 2 i funktionsarkitekturen. Det borde gå att argumentera för att nivå 1 & 2 borde gå att definiera gemensamt på nationell nivå utan att det påverkar befintliga arkitekturer inom lärosätena. Det är först på nivå 3 som lärosätena börjar divergera, eftersom det är på nivå 3 som vi hittar de konkreta lösningarna och implementationerna.

På nivå 1 och 2 är det bara en indelning och klassificering. Det är först på nivå 3 som verksamhetstjänster realiseras i funktionsarkitekturen. Likaså är det först på nivå 3 i informationsarkitekturen som verksamhetens begrepp blir synliga. Inom processarkitekturen på nivå 3 är det möjligt att se processer som faktiskt verksamheten kan definiera som flöden. Inom applikationsarkitekturen är det på nivå 3 som de faktiska applikationerna blir synliga som direkt går att härleda till inköpta/utvecklade applikationer. Det är först på nivå 3 som lärosätena kommer skilja sig åt i applikationsarkitekturen beroende på att lärosätena köper eller utvecklar egna implementationer.

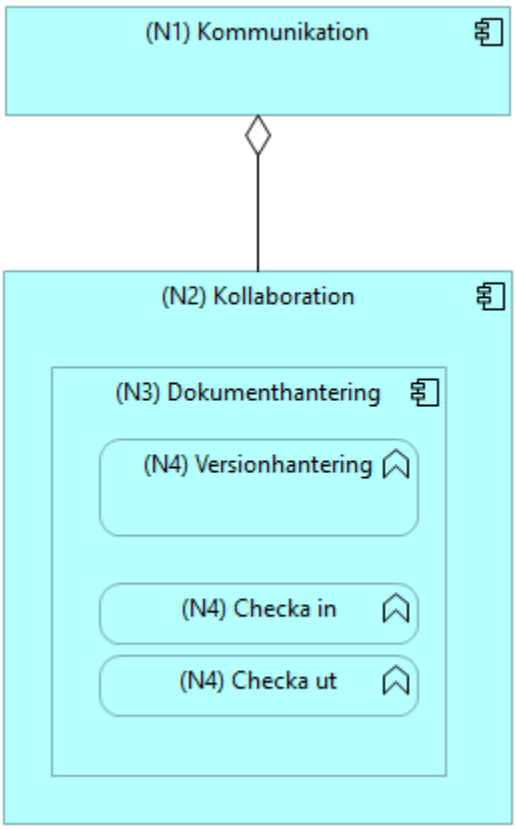
Metamodellens indelningar

Nivåer – detaljering av arkitekturer

Nivåer i arkitekturen är till för att kunna skapa en överblicksbild på hela arkitekturen. Behovet av att detaljera nivåerna beror dels på vilken typ av verksamhet som bedrivs och vilka behov den har av att kunna analysera detaljerna. I en informationsintensiv verksamhet med krav på mycket integrationer blir det naturligt att detaljera informations-, applikations- och infrastrukturarkitekturen men kanske inte processarkitekturen. För en verksamhet med krav på hög standardisering av processer, exempelvis snabbmatskedjor, blir det processarkitekturen som behöver detaljeras mer. Sedan finns det givetvis verksamheter som behöver detaljera flera arkitekturer, såsom flygbolag, för att kunna se sambanden vid en förändring.
För att skapa en ingångsmodell av stadsplanen räcker det med en ganska låg detaljering av arkitekturen men vid ett förändringsbehov kan en eller flera arkitekturer detaljeras mer, givetvis beroende på vad som är viktigt för just den förändringen. Stadsplanen kommer att bestå av flera olika perspektiv med olika grader av detaljering beroende på behoven.
För att skapa en initial bild av verksamheten kan det vara lämpligt att börja med förmågor och tjänster.
Ett exempel på nedbrytning av en arkitektur i nivåer ges av exempelvis informations- och applikationsarkitekturen



Figur 8 Informations- och applikationsarkitekturens nivåer – där N står för nivå
Ett mer konkret exempel på nedbrytning är följande exempel



Figur 9 Ett exempel på nedbrytning av applikationsarkitekturen
I följande stycken följer en förklaring av de olika arkitekturerna och i vissa beskrivningar finns det en ytterligare förtydligande när det gäller element och nivåer där det är otydligt eller kräver ytterligare förklaring.

Arkitekturer

Inom en verksamhet finns det flera områden av struktur. Dessa områden brukar kallas arkitekturområden. Den som vanligtvis brukar beskrivas är:

- Infrastrukturarkitektur – lösning - för att stödja funktionerna
- Processarkitektur – lösning – för att stödja funktionerna
- Produktarkitektur – lösning – för att stödja funktionerna
- Funktionsarkitektur – "funktioner/funktionella förmågor" – det verksamheten behöver kunna utföra för att skapa de produkter och tjänster som motsvarar marknadens behov – du kan inte köra funktioner – lösning på uppdrag (utbildning och forskning)
- Informationsarkitektur – lösning – för att stödja funktionerna
- Applikationsarkitektur – implementation – för att stödja funktionerna
- Infrastrukturarkitektur – implementation – för att stödja funktionerna

Skillnad mellan lösning och implementation är att implementation har en fysisk representation.
Varje område har en struktur som kan beskrivas med kända mönster. Det finns en relation mellan dessa områden och för att förstå hela samspelet krävs en beskrivning av samtliga områden. Varje område behöver inte detaljeras till fullo utan det är utifrån vilka svar som arkitekturen skall ge vid exempelvis en förändring.

Organisationsarkitektur

Detta område beskriver hur organisationen är strukturerad inom en verksamhet. Förutom själva organisationsstrukturen med organisatoriska enheter så är också aktörer och roller definierade. Inom ramen för ArchiMate så går det inte att uttrycka en organisationskarta till fullo men det går att uttrycka vilka aktörer som finns tillgängliga inom en organisation. Detta är ofta tillräckligt för att beskriva en stadsplan då organisation inte direkt beskriver eller tillför någon nytta i funktion- och processperspektivet då dessa går tvärs organisationen.
För att uttrycka hur organisationen är uppbyggd är andra modellspråk såsom UML bättre anpassat för detta ändamål. För att beskriva en organisations uppbyggnad finns flera kända mönster såsom Martin "Fowlers Accountability Pattern". En organisation och dess interna struktur kan uttryckas med en s.k. riktad graf med en toppnod och sedan ett nätverk av noder med relationer under toppnoden. Relationer kan vara av typen linje, samarbete, centrumbildningar mm.
Aktörer och roller är aktiva komponenter i en organisation och visar vad det är för något som utför arbete eller har ägarskap på krav och mål. Om beslut (beteende) skall tas på dessa komponenter så kommer det att krävas en informationsmodell som då kommer att synas i informationsarkitekturen.

Processarkitektur (Utförararkitektur)

Processarkitekturen beskriver en struktur på processerna. Att beskriva en processkarta i detalj är inte målet med ArchiMate utan det är framförallt sambanden mellan processerna som är intressant. Syftet med processarkitekturen är att kunna belysa vilka processer som är påverkade vid en förändring eller att identifiera processer som saknas vid en förändring. För en bättre detaljering av processerna finns det andra modeller som gör detta bättre. ArchiMate är inte ett processkartläggningsverktyg utan snarare en visualisering av hur processerna förhåller sig till sin omgivning på ett förenklat sätt. Den nivå av detaljeringsgrad av processerna definieras vilket samband/perspektiv som skall kartläggas. Det krävs ofta flera perspektiv för att visa "hela" kartan. Inom högskolesektorn är behovet av standardiserade och detaljerade processer begränsat beroende på att verksamheten i de flesta fall är beroende på individbaserat beteende. En bra distinktion på om en process behöver detaljeras är om den kan automatiseras. Med en automatisering menas att även beteendet och besluten i processen automatiseras. För automatiserade processer krävs andra modeller och i arkitekturen bör man representera det som verksamhetsprocesser på en högre nivå som inte är detaljerade. De automatiserade processerna syns istället som tjänster i applikationsarkitekturen men behöver givetvis en relation till en övergripande process i processarkitekturen.

Produktarkitektur (Leveransarkitektur)

Produktarkitekturen är för många inom högskolesektorn ett relativt okänt begrepp. Att börja identifiera produkter och se vilka värden de bidrar till externa aktörer skapar en möjlighet att identifiera externa kanaler som vanligtvis tappas bort. De ger en inblick i vilka tjänster en högskola som faktiskt exponeras externt. Det kan skapa möjligheter att hitta nya behov som idag försvinner om verksamheten bara tittar på sina processer. Andra värden visar sig och motsatsen visas också såsom produkter som inte bidrar till ett externt värde. Produktarkitekturen hjälper till att identifiera "andra värden" som blir svåra att identifiera om verksamheten bara tittar på vilket värde som processerna skall uppfylla. Processperspektivet tenderar att bli något internt även om ambitionen är att se till att det är kunden som skall få nytta. Produktarkitekturen kan hjälpa till med att tydliggöra detta.
Med produkter avses tjänsteprodukter, dvs. produkter som skapas av tjänster och inte fysiska produkter. Fysiska produkter är produkter som inte interagerar med en extern aktör under själva processen. Processen kan ha externa tjänster för att påverka konfigurationen på den fysiska produkten men då är det en separat tjänsteprodukt skilt från den fysiska produkten. I ett EA perspektiv är inte den fysiska produkten intressant eftersom den fysiska produkten inte har några kanaler under själva tillverkningsprocessen.

Funktionsarkitektur

Funktionsarkitekturen bör "mappa" väl mot den i arkivlagen definierade klassificeringsstrukturen.
Framgångsförmågor används vanligtvis när en verksamhet vill göra en förändring som fokuserar på ett tydligt mål eller flera mål. Ett exempel på en framgångsförmåga är "Vi skall vara världsledande inom... eftersom att vi då…". Inom ramen för stadsplan så är det inte dessa förmågor som är intressanta eftersom en stadsplan handlar om att kartlägga nuläget och inte förflyttningen. Stadsplanen kan givetvis användas för att se hur en förändring påverkar stadsplanen och efter förändringen är genomförd så uppdateras Stadsplanen till ett nytt nuläge. Det kan också bli tydligt vilka förmågor som verksamheten har men inte uppfattat att den har.
Funktionella förmågor har ett internt perspektiv och inte ett framgångsperspektiv. Det kan ofta vara enklare att definiera förmågor eftersom de är en löst kategorisering av vad verksamheten skall kunna för att fungera. En förmåga består av alla de övriga arkitekturerna och dess delar. Förmågor är den centrala arkitekturen - det är utifrån förmågorna som stadsplanen skall växa fram.

Regelverk för funktionsarkitekturen

Regelverket:

1. En funktion är unik till namn och samlad funktionalitet inom den avgränsade helhet (system i systemtänkande, där företaget kan vara ett system) som vi undersöker.
2. En funktion har en given mängd tillhörande tjänster.
3. En tjänst tillhörande en funktion exponerar en given mängd funktionalitet ur funktionens hela funktionalitet.
4. En tjänst tillhör endast en funktion.
5. En tjänst kan ha samma namn som en annan tjänst i den avgränsade helheten.
6. En tjänst kan exponera likvärdig funktionalitet som en annan tjänst i den avgränsade helheten.

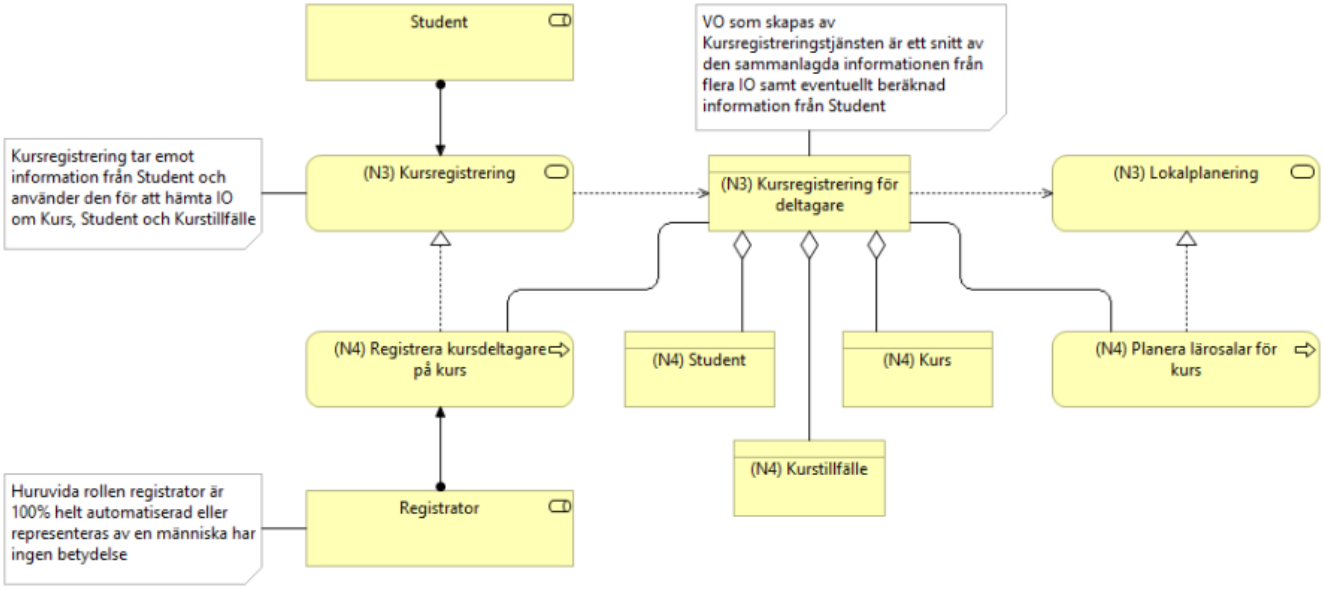
Regel nummer 1 är beroende av att vi använder namnrymder för att särskilja instanser av arkitekturen från varandra.
Exempelvis så finns ju Redovisningsfunktionen både ett externt företag och på GU. Båda är av varandra oberoende instanser av arkitekturen för juridisk person.

Informationsarkitektur

Inom högskolesektorn i Sverige är behovet av korrekt information vid varje tillfälle helt väsentligt för att verksamheten skall kunna fungera effektivt. Det går att hävda att den typ av verksamhet som bedrivs inom utbildning och forsknings är mer förtjänta av korrekt information än att ha standardiserade processer. Detta enligt Ross et al Enterprise Architecture As Strategy: Creating a Foundation for Business Execution: ISBN13:9781591398394. Dock så är detta något som är eftersatt på många myndigheter och det finns mycket att göra för att förbättra detta. En förutsättning för att digitalisera verksamheten är att begrepp och information har en entydig betydelse. I synnerhet när behovet av integrationer mellan system ökar dramatiskt så är det nödvändigt att veta vart information uppstår, vad den betyder och hur den sprids inom verksamheten. I synnerhet är informationsarkitekturen helt betydande inom BI-området. För att kunna få ut rätt mätetal måste struktur och relation mellan information vara helt korrekt annars kommer inte beslutsunderlagen/mätetalen att bli korrekta.
Ofta har behovet av korrekt information drivits från IT-avdelningarna eftersom de är dessa som ansvarat för automatisering och integration. Dock så borde verksamheten bli mer engagerad i definitionen av informationen. Metamodellen hjälper till att se hur informationen kommer in i den totala arkitekturen och också ge indikationer om vilken information som är viktig. MM kan bli ett verktyg att börja strukturera information så att den kan hjälpa verksamheten i digitaliseringen.

Nivåer och Element i informationsarkitekturen

Alla element i Informationsarkitekturen består av ArchiMate Business Object. Det kan upplevas som en bredare tolkning än definitionen i Archimate specifikationen - men som alltid är det kontexten som verkligen sätter definitionen. Dvs. frågan som bör ställas är vad som skall modelleras och vad betyder definitionen just i denna kontext. Metamodellen lyfter in alla nivåer i verksamheten och beskriver också en viss kontext - dvs. en strukturering av information som används i verksamheten, från övergripande nivå till detaljerad nivå.
Metamodellen i sig har inga möjligheter att uttrycka detaljerade informationsmodeller utan är ett sätt att sortera information och se hur den används. För att göra begreppsmodeller, informationsmodeller används andra modellmetoder såsom exempelvis UML. UML kan uttrycka samtliga nivåer i metamodellen modellmässigt. Det finns givetvis en uppsjö av modellspråk som går att använda men gemensamt för alla är att objekten i dessa modeller sedan kan sorteras in i metamodellen.

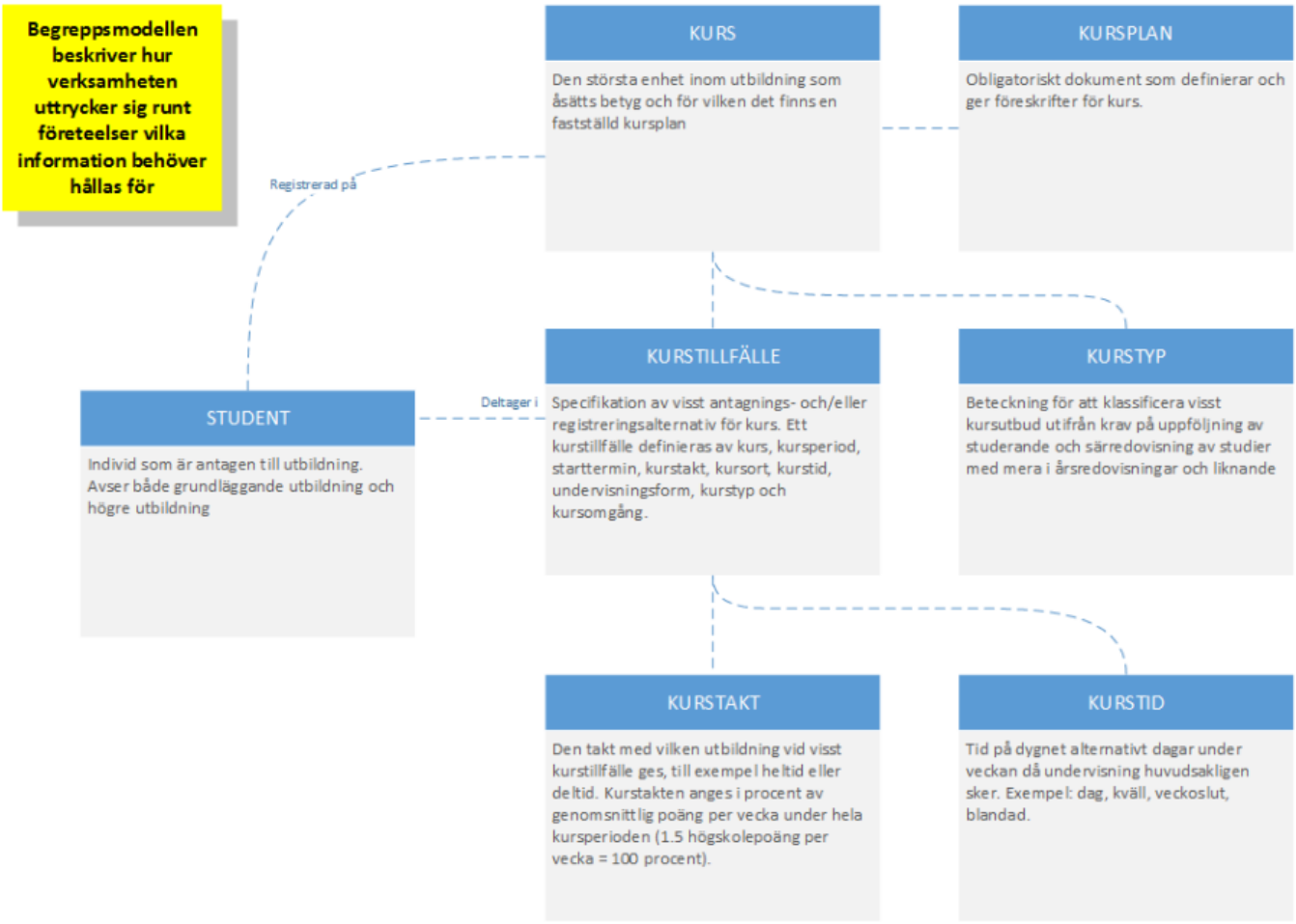


Figur 10 Skillnad mellan verksamhetsobjekt (VO) och informationsobjekt (IO)

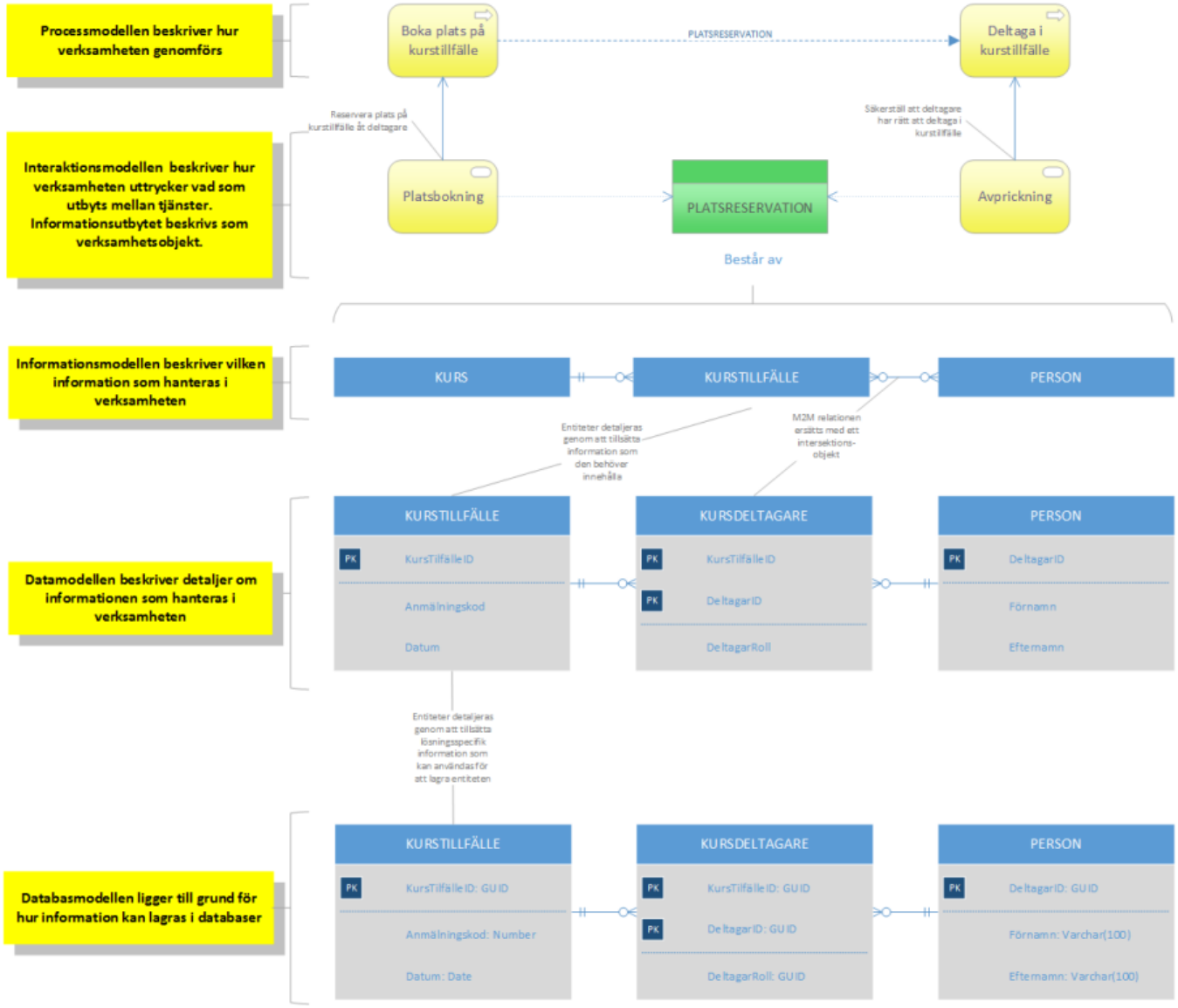
(N1) Informationsområde
En grov indelning av information som visar vilken typ av information som hanteras inom verksamheten. Indelningen kan inspireras av nivå 1 inom applikationsarkitekturen exempelvis: Utbildningsinformation, affärsinformation, myndighetsinformation etc.
(N2) Informationsgrupp
Ytterligare en nedbrytning av informationen till en mer funktionsnära nivå. Det är på denna nivå som informationsägarskapet utövas. Här är det lämpligt att titta på de definierade funktionerna för att inspireras av indelningen. Denna nivå är inte helt olikt den informationsgruppering som i IRM kalls för informationskvarter.
(N3) Verksamhetsobjekt
Denna nivå relaterar till konkreta händelser i verksamheten och som är ett resultat av en (N3) Processtjänst, se figur ovan.
(N4) Informationsobjekt
Nivå 4 är den nivå där vi hittar klassiska entitetsdiagram och klassdiagram. I figuren ovan ser vi att ett VO "(N3) kursregistrering för deltagare" är uppbyggt av minst 3 IO. Dock så finns det ingen möjlighet att inom metamodellen att uttrycka andra relationer än aggregeringar av information. För att visa relationerna mellan IO krävs andra modeller såsom UML-klassdiagram eller IRM informationsmodeller.

Exempel på nedbrytning av informationsarkitekturen

För att förstå vilken information verksamheten har behov av behöver en nedbrytning av informationen göras. En bra utgångspunkt för denna nedbrytning är att förstå vilka begrepp verksamheten använder sig av. Begreppen kan beskrivas och definieras samt att enklare relationer mellan begreppen kan identifieras. Begreppsmodellen nedan visar på en hur några begrepp inom Informationsområde (N1), Utbildningsinformation, är relaterade till varandra och enklare definitioner till begreppen. Begrepp inom verksamheten kan vara väldigt generella och andra kan vara väldigt specifika. Det gör att det är en utmaning att göra nedbrytningen till informationsnivå (N4). Vissa av begreppen realiseras av flera informationsobjekt, ofta lite mer generella begrepp, eller väldigt specifika begrepp som kan vara specialiseringar av enskilda klasser.
En gemensam begreppssyn är en grundbult för att kunna digitalisera verksamheten. Otydliga definitioner på begrepp går inte att entydigt bryta ner till väl definierade informationsobjekt (N4). Då går det inte heller att realisera dataobjekt (del av systemarkitekturen) som applikationstjänster utbyter, bearbetar eller förmedlar. Dessa tjänster och tillhörande dataobjekt är den digitala representationen av verksamheten - det som vanligtvis kallas "digitalisering av verksamheten".



Figur 11 En begreppsmodell som beskriver och visar samband mellan begrepp



Figur 12 Nedbrytning från processmodell till databasmodell

Applikationsarkitekturen

Denna arkitektur är den som de flesta kan enkelt relatera till. De flesta har en relativt god uppfattning som vilka applikationer som finns inom verksamheten och vilken domän som de stödjer. Många lärosäten använder sig av en förvaltningsmodell, PM3, som är en form av applikationsarkitektur. Dock så brukar det vara problematiskt att se hur applikationsarkitekturen påverkar andra delar av verksamheten vid en förändring. Det är också så att applikationsarkitekturen brukar innefatta en del av infrastrukturarkitekturen, vilket leder till att det kan vara svårt att se hur infrastrukturen stödjer flera förvaltningsobjekt (PM3) eller att infrastrukturen förvaltas inom ett objekt vilket leder till suboptimering och en risk för viss stuprörsförvaltning. Genom att abstrahera ut applikationslagret mot infrastruktur och se hur hela kartan ser ut ger en bättre bild av hur infrastruktur kan återanvändas. Det blir också tydligare hur applikationsarkitekturen stödjer flera delar av verksamheten.
Applikationsarkitektur brukar slarvigt kallas för systemkarta och är en förenklad vy av applikationssamband och saknar ofta en relation till verksamhetens tjänster.

Element i Applikationsarkitekturen

Inom applikationsarkitekturen används 2 element i ArchiMate:

- "Application Component"
- "Application Function"

För nivåerna N1-N3 används "Application Component"enligt:

- N1 – Applikationsområde
- N2 – Applikationsgrupp
- N3 – Applikation

För nivå N4 används "Application Function":

- N4 – Applikationsfunktion

Applikationsområden - N1

Idén med vilka applikationsområden som används på nivå 1 inom applikationsarkitekturen grundar sig i en gemensam syn på vilken typ av myndighet som skall beskrivas. Beroende på typen av myndighet finns ett antal gemensamma applikationsområden för samtliga myndigheter samt några specifika applikationsområden som kan härledas till vilken betoning myndigheten har på sin verksamhet. Ett förslag på indelning följer nedan:
Gemensamma applikationsområden som finns inom samtliga myndigheter:

- Affär
- Myndighet
- Fastighet- och lokal
- Infrastruktur, mjuk och hård, och till för ngn annan och inte sig själv
- Kommunikation

Applikationsområden som är kopplade till myndigheter inom utbildning och forskning:

- Utbildning
- Forskningsstöd
- Bibliotek

Applikationsområden som inte förvaltas av myndighet

- Externa applikationer

Skillnader mot applikationsområden i ATI:s referensarkitektur

I referensarkitekturen finns ett antal applikationsområden definierade. Dessa passar dock inte riktigt in i metamodellens nivåer då de spänner sig över flera nivåer. Applikationsområdena och hur de mappar mot de föreslagna applikationsområdena är:

- Ekonomi – Affärsområdet
- HR – Affärsområdet
- Studieadministration – Utbildningsområdet
- Utbildningsstöd – Utbildningsområdet
- Bibliotek – Biblioteksområdet
- Externa relationer – Relationsområdet
- Infrastruktur - Infrastrukturområdet
- Kollaboration - Relationsområdet
- Forskningsadministration - Forskningsstödområdet
- Innehållshantering - Relationsområdet
- Ledningsstöd - Affärsområdet
- Forskningsstöd - Forskningsstödområdet
- Administration - Kan hamna i olika

I tabellform:

Stadsplan (N1)	***Applikationsområde***
Affär	Ekonomi
	HR
	Ledningsstöd
Utbildning	Studieadministration
	Utbildningsstöd
Bibliotek	Bibliotek
Relation	Externa relationer
	Kollaboration
	Innehållshantering
Infrastruktur	Infrastruktur
Forsknings	Forskningsadministration

	Forskningsstöd
"Olika"	Administration

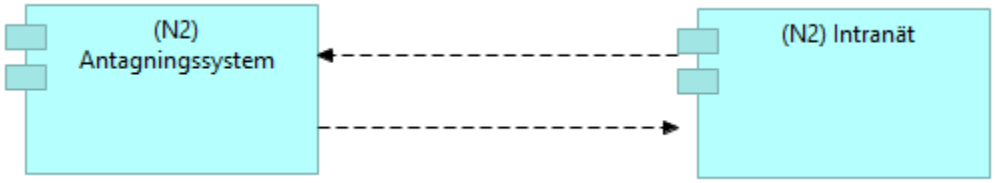
Tabell 1 Översättning av referesnarkitekurens applikationsområden till metamodellens indelning

Ett förslag på vidare detaljering av applikationsarkitekturen finns som bilaga på sid . Detta är ett utkast och förslag som behöver bearbetas några varv till. Mönstret för att bryta göra en detaljering ser ut enligt följande:

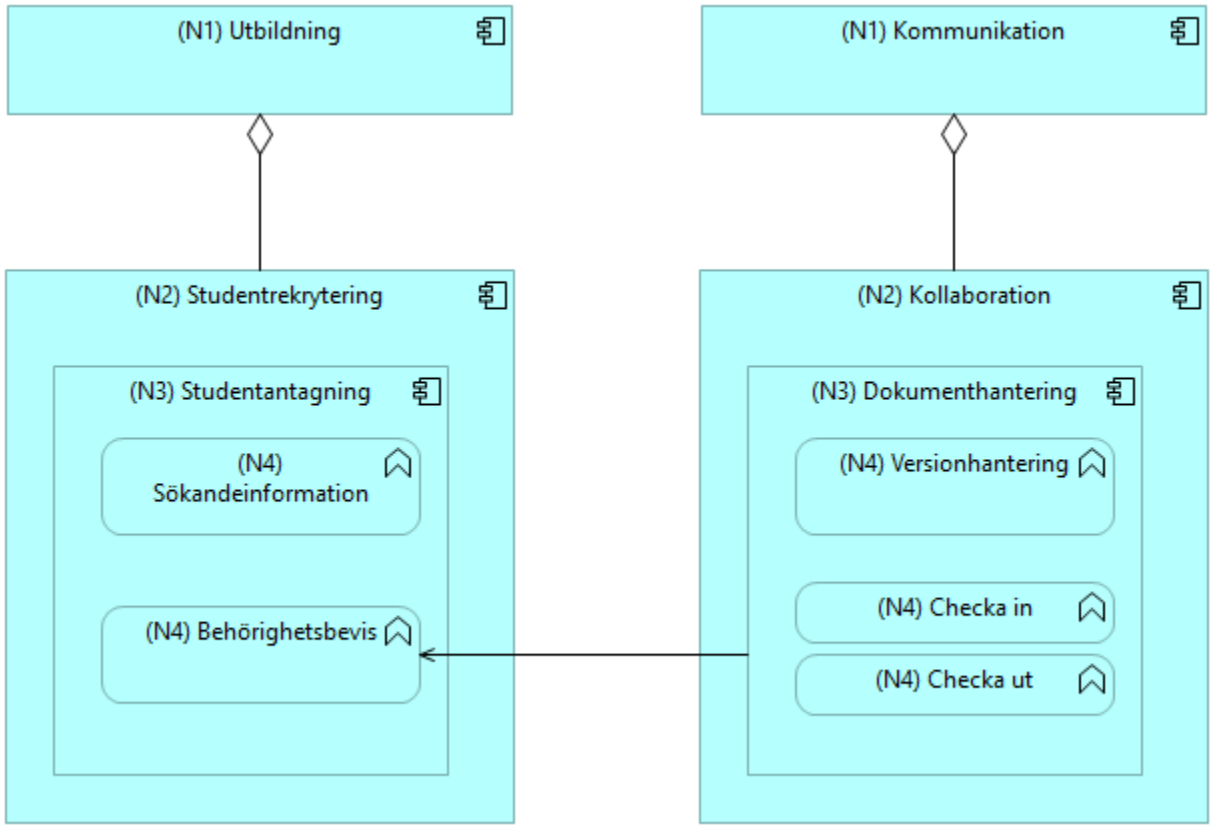
1. Applikationsområde (N1)
    a. Affär
2. Applikationsgrupp (N2)
    a. Ekonomi
3. Applikationskomponent (N3)
    a. Redovisning
    b. Finansiell styrning och uppföljning
    c. Budgetering
    d. Inköp
    e. Lönehantering

Applikationsgrupp (N2) ur ett verksamhetsperspektiv

Ett system (motsvarar applikationsgrupp N2 i metamodell) skall definieras ur ett verksamhetsperspektiv. Ett system är en sammanhängande mängd applikationskomponenter som tydligt kan identifieras från verksamheten. System som finns för att andra system skall fungera såsom exempelvis Active Directory finns men syns inte i ett sambandsperspektiv mellan system eftersom AD ofta har relationer mellan flera system. Informationsutbyte mellan system görs i en vy mellan (N2) SystemA och (N2) SystemB utan att detaljera utbytet. I exemplet nedan sker ett informationsutbyte mellan Antagningssystemet och Intranätet. Dessa har då ett beroende till varandra på systemnivå.



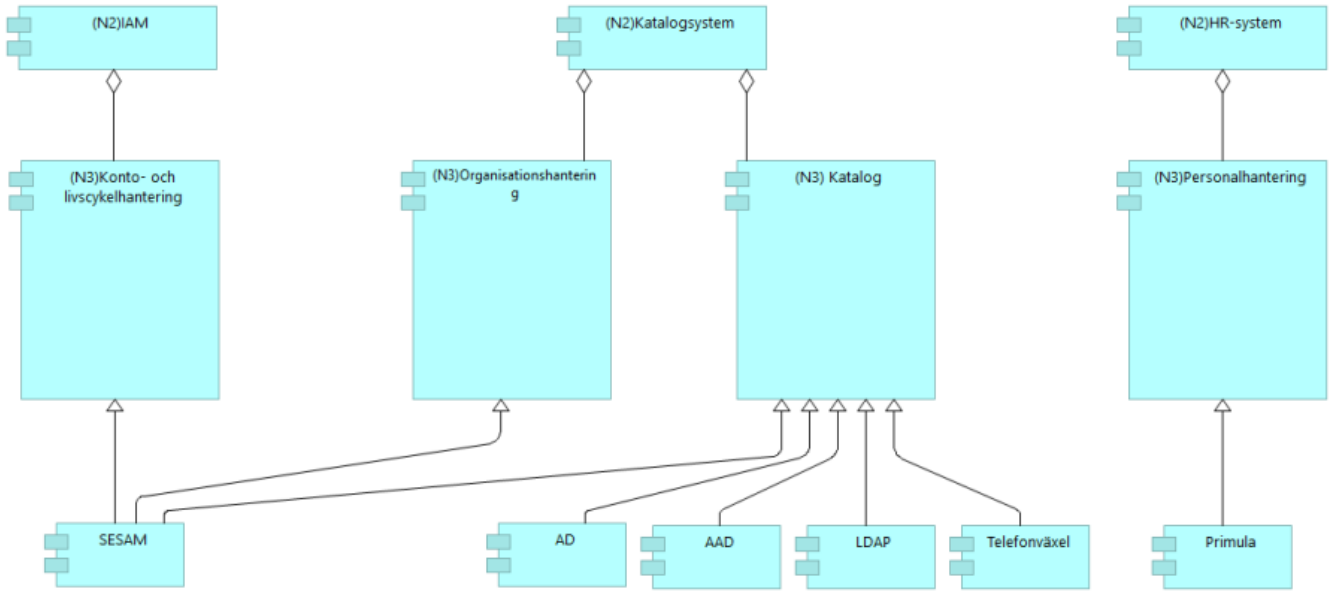
Figur 13 I en detaljerad bild som visar vilka delar i systemen som samverkar på hög nivå



Figur 14 Ett hypotetiskt exempel som visar att behörighetsbevsifunktionen använder sig av dokumenthantering. Modell för att illustrera hur detaljer kan utelämnas för att enklare visa samband

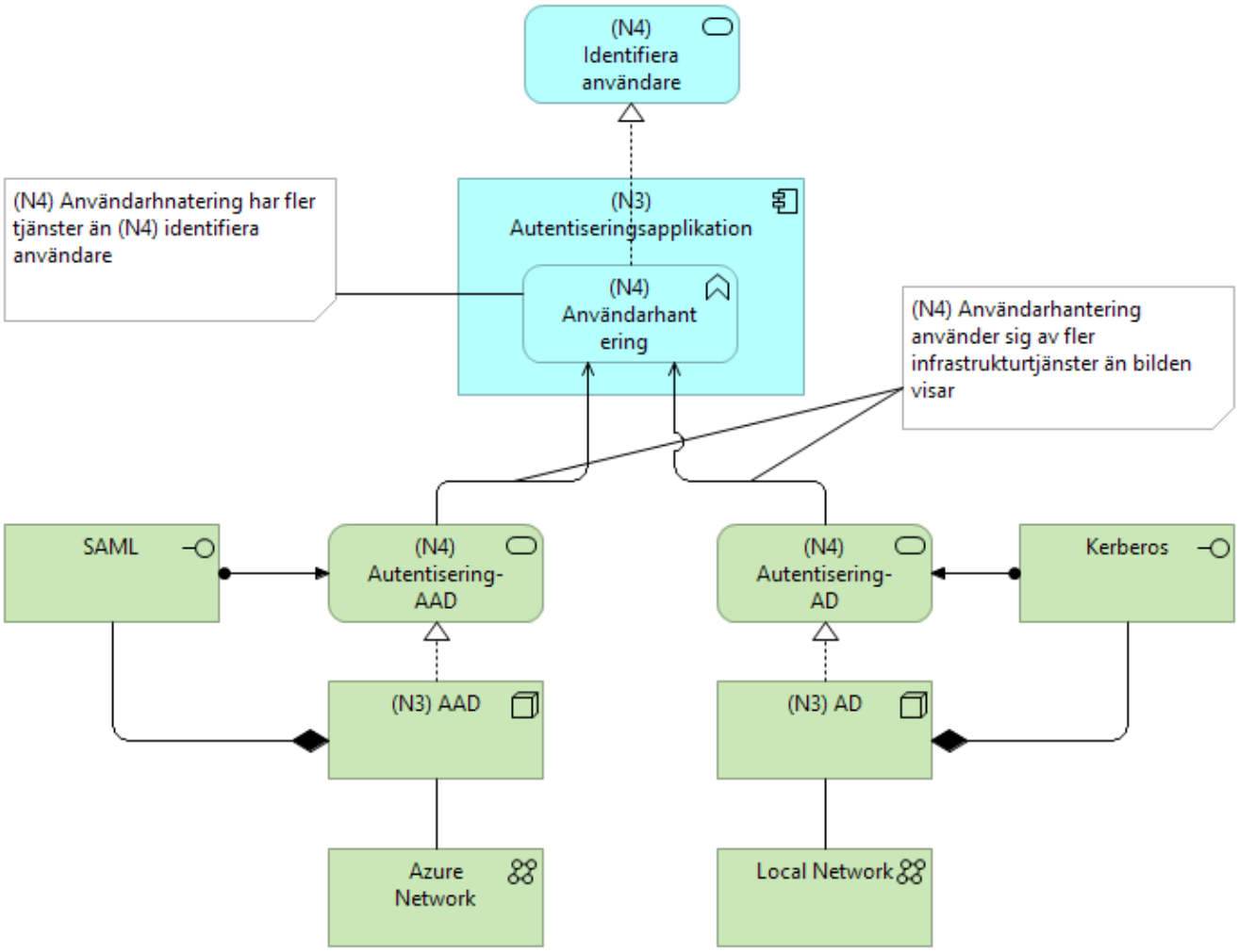
Applikation – (N3)

Med applikations menas tillämpning och genom att standardisera tillämpningarna inom ett lärosäte går det att skapa en överblick hur många olika mjukvaror som realiserar tillämpningarna. Det är på denna nivå som det går att identifiera applikationer som realiserar samma tillämpningar. Som beskrivit tidigare är ett mål med systemarkitekturen att minimera antalet applikationer och i idealarkitekturen endast ha en tillämpning som realiseras av en mjukvara. Givetvis är det inte möjligt i alla fall men det ger en indikation på vart det går att rationalisera. Genom detta arbete går det också att jämföra mellan lärosäten vilka tillämpningar lärosätet använder istället för att titta på vilken mjukvara som används. Det kan vara så att två lärosäten använder samma mjukvara men med olika tillämpning. Ur ett samarbetsperspektiv blir det svårare att hitta gemensamma samarbetspunkter alternativt går det att hitta nya sätt att använda mjukvaran på. En lista på förslag av tillämpningar finns som bilaga på sid
Oavsett om arkitekturen definierar tillämpningar behövs ibland en övergång från tillämpningen till den fysiska representationen av tillämpningen. I synnerhet när lärosätet skall titta på integrationer och informationsflöden mellan fysiska "system". I arbetet med att titta på LiUs "Min-IT" blir det tydligt att det är nödvändigt att gå över i ett perspektiv som beskriver den fysiska lösningen. Dock så ger möjligheten att se att det finns återkommande produkter som samma tillämpningar. Bilden nedan är ett försök att visa detta:



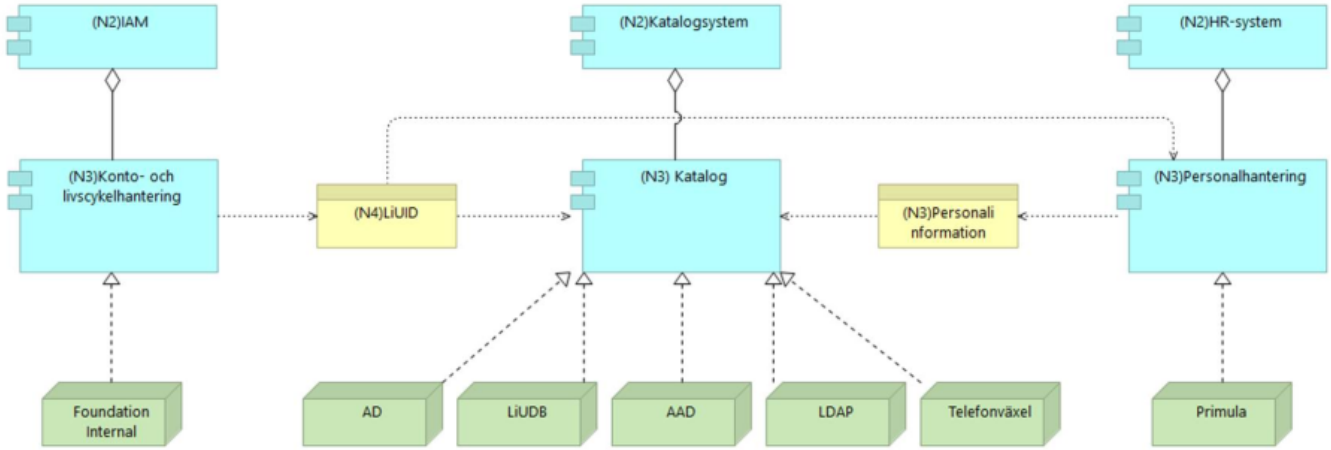
Figur 15 För att koppla från tillämpning till fysisk nivå kan specialisering användas

Detta ger en möjlighet att hitta "onödiga" komponenter, dock behövs detta brytas ner i applikationsfunktioner för att verkligen se om alla fysiska instanser är nödvändiga. Det kommer definitivt finnas överlapp i funktionalitet men att detta kan vara nödvändigt då infrastrukturella egenskaper kräver redundans. Ett enkelt exempel är funktionaliteten att identifiera en användare genom autentisering. Både AD (Active Directory) och AAD (Azure AD) har denna funktionalitet. Dock så skiljer egenskaperna på infrastrukturtjänsterna – den ena gör autentisering "On Site" den andra i molnet (Azure). Så finns verksamheten i molnet måste både infrastrukturtjänsterna existera.
En bild att försöka illustrera detta:



Figur 16 AAD tillhör Azure och AD tillhör lokalt. Båda behövs om verksamheten finns lokalt och i molnet.

Tidigare i rapporten har det tryckts på att informationen i verksamheten måste vara entydigt definierad för att kunna stödja digitaliseringsarbetet. Det kan vara lämpligt att övergripande titta på vilka tillämpningar som utbyter information. Eftersom det finns flera komponenter i infrastrukturen som realiserar de olika tillämpningarna betyder det att antalet integrationer blir fler än de som syns i applikationsarkitekturen när endast tillämpningen beaktas. Dock så visar modellen att informationsmängderna i de olika fysiska komponenterna behöver definieras på samma sätt. Detta är ett sätta att beskriva skillnaden mellan logik och infrastruktur. Exempelvis om lärosätet har definierat att integration skall ske genom meddelandeintegration så skall varje nod i infrastrukturen överföra meddelande mellan varandra. Dessa meddelanden innehåller data som representerar informationsmängden som skall föras över. Modellen nedan är ett sätt att minska komplexiteten på applikationsnivå men samtidigt kunna spåra hur många integrationer (transporter) som måste till för att lösa överföringen av information.

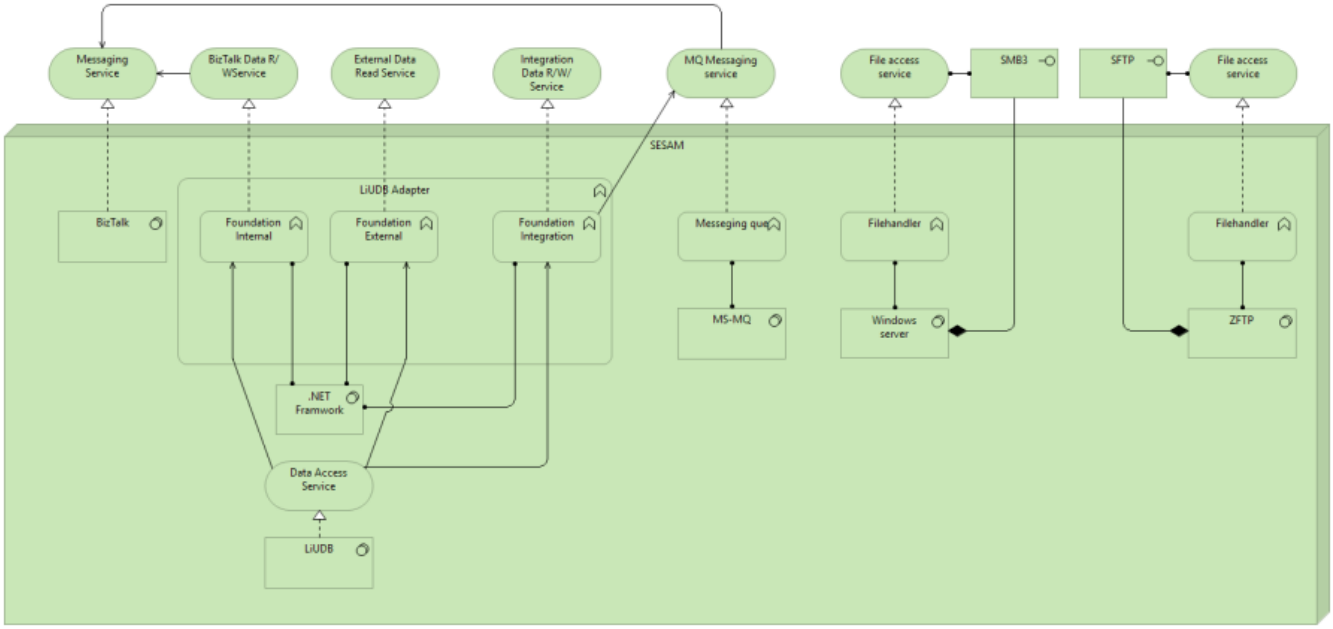


Figur 17 Flera noder realiserar samma tillämpning men informationsbehovet på tillämpningsnivå är samma, vilket leder till fler transporter än det som syns i applikationsarkitekturen

Modellen ovan kan vara abstrakt att förstå och kanske inte är användbar i verkligheten, men författaren tycker det hjälper till att förstå skillnaderna mellan ArchiMate applikationsnivå mot infrastrukturnivån.

Infrastrukturarkitektur

Det kan vara abstrakt att dela på applikationer och infrastruktur och kräver en del tankeverksamhet. Inom infrastrukturarkitekturen hittas "hårdvara och sladdar" tillsammans med filer och databaser mm. Det finns inga logiska beteenden i infrastrukturarkitekturen. Det som kommer fram är gemensam hårdvara för olika applikationer och på vilket sätt data skickas mellan system. Förenklat går det att säga att inom applikationsarkitekturen ses sambanden mellan beteenden och tjänster samt vilken data som utbytes medan i infrastrukturarkitekturen ses hur filerna som innehåller data transporteras och vilken hårdvara detta sker emellan. Det är alltså de fysiska sambanden som visas här och ger en bild över exempelvis i vilken datahall hårdvaran står. Fysisk datasäkerhet eller virtualiserade applikationer är ett exempel på vad som tydligt visas tydligt i infrastrukturarkitekturen. Det kan vara en utmaning att hitta rätt detaljering på denna nivå och det är viktigt att hela tiden tänka på vad som skall beskrivas. Ofta går det att förenkla och beskriva exempelvis hur filer transporteras och inte hamna i en detaljering som beskriver alla komponenter och artefakter i en server.
Det är inom denna nivå som kopplingen till ATI:s referensarkitektur finns och det är på denna nivå som integrationsmönster syns. Dock inte alla integrationsmönster. I ESB fallet finns det komponenter som beskriver hur data flyttas baserat på routinglogik men detta kan implicit ses som att det är samma nod som "hämtar och skickar" filer. I och med att alla filer passerar samma nod så bör det rimligtvis finnas logik för att "fördela" trafiken till olika mottagare. I systemarkitekturen bör det synas att det finns en applikationsfunktion som läser specifika data och sedan använder en tjänst på en annan applikation för att "lämna" data. Återigen är det viktigt att inte detaljera för mycket då bilden fort blir komplex.
Som en del i arbetet med att titta på LiU har här gjort en infrastrukturmodell över LiUs integrationsplattform:



Figur 18 LiUs integrationsplattform i ett infrastrukturperespektiv
Den är inte på komplett utan försöker mer visa vilka infrastrukturtjänster som SESAM exponerar och visa vilka infrastrukturkomponenter som realiserar LiUDBs adapter. Modellen togs fram i just det syftet att visa att LiU inte använder BizTalk för att realisera deras adapter mot LiUDB. Dock så är BizTalk en viktig del i att transformera data i integrationerna.

LiU-POC MinIT-tjänst

För att titta på ett konkret exempel som är relativt klurigt att uttrycka i MM har MinIT valts som ett exempel. Eftersom detta i ett MM-perspektiv inte är en tjänst utan en produkt som aggregeras av ett antal underliggande verksamhetstjänster är det ett bra exempel. Hela modellen finns som bilaga på sid .
Modellen finns även som Archifil på Wiki Stadsplan
En viktig aspekt att beakta är denna vy är begränsad för att beskriva ett snävt perspektiv. Det är nödvändigt att inte försöka visa "för mycket" i varje modell. Då kommer snabbt modellen att bli oläsbara och svåra att använda till något. Det är en svår balansakt att utföra men nödvändig för ett bra arkitekturarbete. Att göra den abstraktionen att MinIT är en produkt hjälper till att identifiera liknande portaler på lärosäten och också då se produkter (det som genererar värde för verksamheten) som ett möjligt samarbetsområde.

Förslag på tekniker för att fylla på metamodell

I detta kapitel finns ett antal olika tekniker och förslag på hur vissa arkitekturer i metamodellen kan populeras. Listan är inte komplett utan det finns flera tekniker. Inom ramen för denna rapport görs inte heller en detaljerad förklaring på de olika teknikerna utan detta är också en del av det fortsatta arbetet. Fokus på rapporten är att förklara metamodellen och dess komponenter.

Funktionsarkitekturen

Ett stöd i att dokumentera funktionerna:
The Capability Canvas
Ett sätt att rita funktionsarkitekturen:
Capability Architecture and Microservices
Fler tekniker:
Finns en processkarta går det att hitta funktionera utifrån den genom att skapa funktionerna på nivå 1, 2 och 3:
Kom ihåg att de Förmågor/Business Function som skapas är att anse som funktionaliteten som processer behöver för att utföra sitt arbete.
Så för att skapa förmågorna på nivå 1, 2 och 3:

1. Titta på processerna på de nivåerna och plocka ut funktionaliteten som processen pekar på, blir hypotes på förmåga på den nivån.
2. Konsolidera förmågorna där så går, blir förmågan som eventuellt flera processer kan dela på.
3. Analysera hela kartan och korrigera där så behövs
4. KLART!

De riktiga förmågorna som ligger i strategy architecture (se metamodellen) kommer för ett lärosäte att vara +/- 7st.
Till exempel: Vi knyter samman världen bästa forskare inom område elbilsbatterier för att skapa förutsättningarna för innovativ utveckling av framtidens elbilar
Ett namn på den skulle kunna vara "Elbilar som går lika långt som bränslebilar"
Ett exempel på en Capability map (Framgångsförmågor):
Sample Business Capabilities

Tjänster/process

En post om hur synen på processarkitektur
Event Based Processing and Capability Architecture

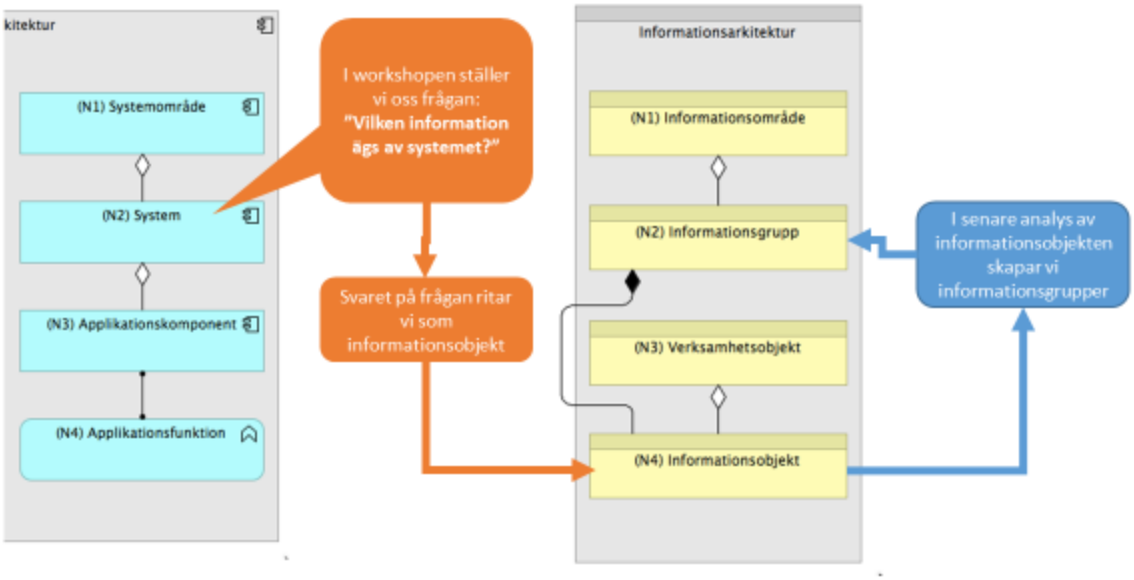
Informationsarkitekturen

Kategorier inom information

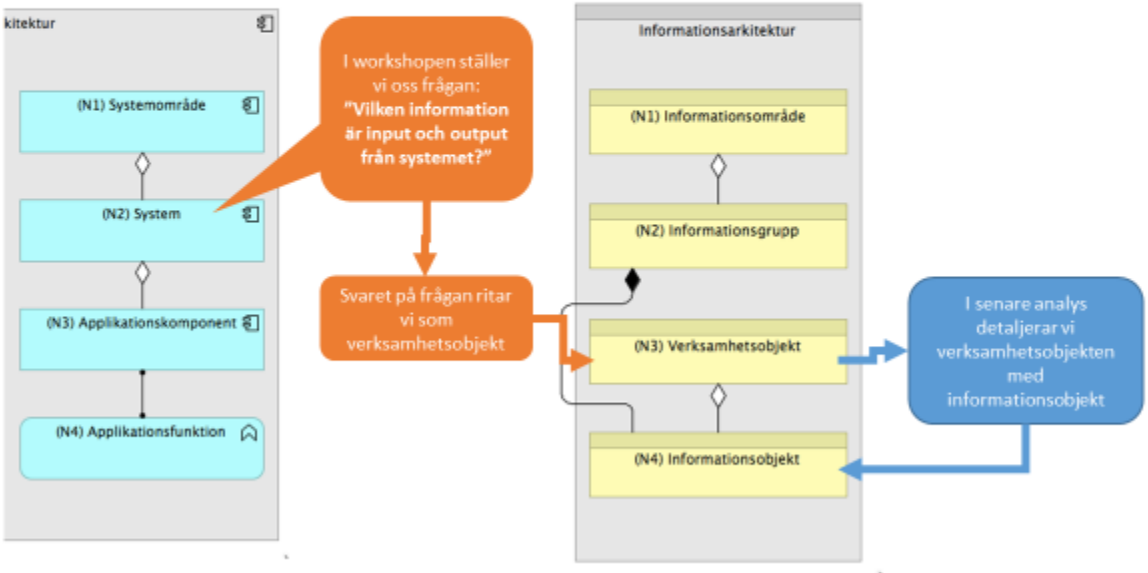
KATEGORI	DEFINITION	EXEMPEL
Vem	• Person eller organisation som är av intresse att känna till saker om. • Frågan vi ställer är "Vem är viktig för oss?" • Ofta har en "vem" en speciell roll, till exempel kund eller leverantör.	• Anställd, Patient, Spelare, Kund, Leverantör, Student, Passagerare, Myndighet, Konkurrent
Vad	• Produkt eller tjänst som är av intresse för oss att känna till saker om. • Frågan vi ställer är "Vad är viktigt för oss?" • Ofta referera detta "vad" till det vi tillverkar och säljer.	• Produkt, Tjänst, Råmaterial, Vara, Kurs, Bild, Kod, Bil
När	• Datum och tid eller intervall som är av intresse för oss att känna till saker om. • Frågan vi ställer är "När händer något som är viktigt för oss?"	• Tid, Datum, Kvartal, År, Kalender, Semester, Rapporteringsperiod, Regnsäsong, Räkenskapsår
Var	• Plats som är av intresse för oss att känna till saker om. • Frågan vi ställer är "Var sker verksamhet?" • Ofta referera detta till fysiska platser men gäller även logiska platser.	• Adress, Koordinat, Distributionspunkt, URL, Nod.
Varför	• Händelse eller transaktion som är av intresse för oss att känna till saker om. • Frågan vi ställer oss är "Vad händer som vi reagerar på?"	• Beställning, Retur, Klagomål, Uttag, Insättning, Kommentar, Förfrågan, Ersättningsanmälan
Hur	• Dokumenterat resultat av en reaktion på en för oss viktig händelse. • Frågan vi ställer oss är "Vad består efter att en händelse har skett?"	• Faktura, Kontrakt, Konto, Inköpsorder, Böteslapp.

Tabellen ovan används för att förstå vilka frågor vi kan ställa i en workshop

Steg 1: Vi skapar underlag till IA från kunskap om SA



Steg 2: Vi skapar underlag till IA från kunskap om SA



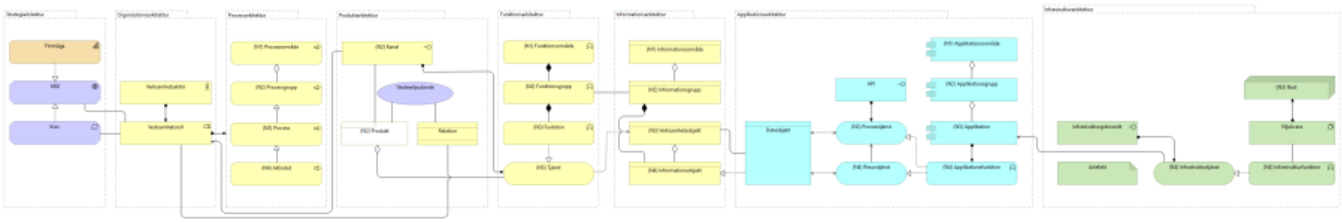
Bilaga 1 – Detaljering av applikationsarkitekturen, förslag

Nedan följer ett förslag på detaljering av applikationsarkitekturen. Detta är ett utkast och behöver bearbetas ytterligare.

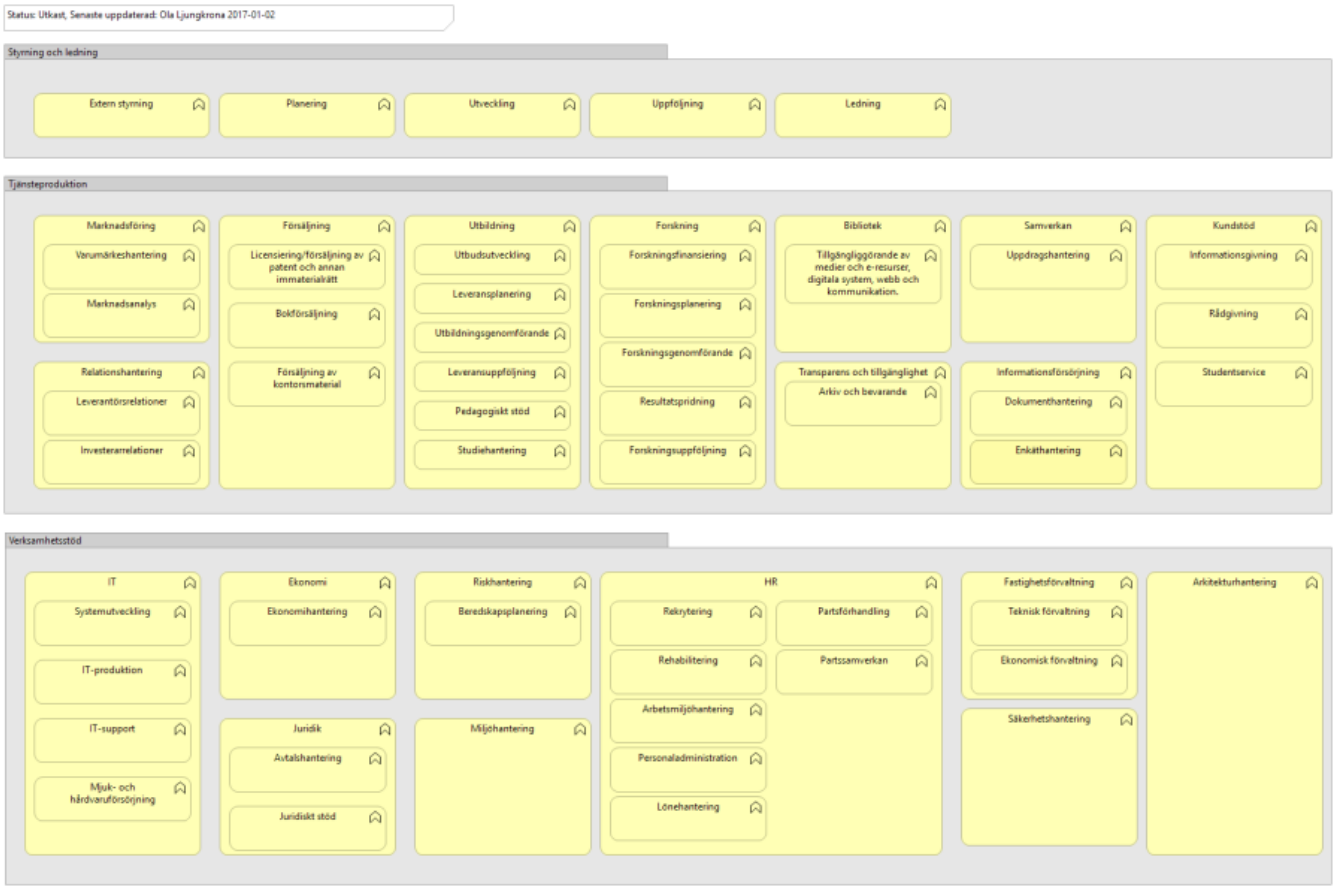
Kursanmälan	Anmälning	Utbildning
Tentaanmälan	Anmälning	Utbildning
Diarie	Arkiv	Myndighet
E-arkiv	Arkiv	Myndighet
Autentisering	Autentisering	Infrastruktur
Extern webb	CMS	Kommunikation
Intranät	CMS	Kommunikation
E-handel	E-handel	Affär
Budget	Ekonomi	Affär
Fakturahantering	Ekonomi	Affär
Finansiell styrning	Ekonomi	Affär
Inköp och upphandling	Ekonomi	Affär
Redovisning	Ekonomi	Affär
Uppföljing	Ekonomi	Affär
Digital examinering	Examination	Utbildning
Campusbussen	Externa transporter	Externt
Personalhantering	HR	Externt
Enkäthantering	Informationsinsamling	Kommunikation
Informationsintegration	Integrationsplattform	Infrastruktur
Tjänsteorienterad integration	Integrationsplattform	Infrastruktur
Applikationsinstallation	IT-infrastruktur	Infrastruktur
Fillagring	IT-infrastruktur	Infrastruktur
Fjärrskrivbord	IT-infrastruktur	Infrastruktur
Hantering av digitala enheter	IT-infrastruktur	Infrastruktur
Mediahantering	IT-infrastruktur	Infrastruktur
Nätverkshantering	IT-infrastruktur	Infrastruktur
OS-installation	IT-infrastruktur	Infrastruktur
Spamhantering	IT-infrastruktur	Infrastruktur
Systemövervakning	IT-infrastruktur	Infrastruktur
Utskrift	IT-infrastruktur	Infrastruktur
Övervakning IT-resurser	IT-infrastruktur	Infrastruktur
Felanmälan	IT-support	Kommunikation
IT-supportstöd	IT-support	Kommunikation
Ärendehantering	IT-support	Kommunikation

Grupphantering	Katalog	Infrastruktur
Konto- och livcykelhantering	Katalog	Infrastruktur
Organisation- och personkatalog	Katalog	Infrastruktur
Organisationshantering	Katalog	Infrastruktur
Chatt	Kollaboration	Kommunikation
Dokumenthantering	Kollaboration	Kommunikation
Onlinemöten	Kollaboration	Kommunikation
Projekt- och verksamhetsstöd	Kollaboration	Kommunikation
Utvecklingsplattform	Kollaboration	Kommunikation
E-posthantering	Kommunikation	Kommunikation
E-postlisthantering	Kommunikation	Kommunikation
Telefoni	Kommunikation	Kommunikation
Kursval	Kursval	Utbildning
Hantering kårmedlemskap	Kår	Externt
Antiplagiathantering	LMS	Utbildning
Byggnads- och rumskatalog	Lokalhantering	Fastighet och lokal
Hantering hyresavtal	Lokalhantering	Fastighet och lokal
Hus- och lokalkatalog	Lokalhantering	Fastighet och lokal
Inpassering lokaler	Lokalhantering	Fastighet och lokal
Kartor	Lokalhantering	Fastighet och lokal
Passerkortshantering	Lokalhantering	Fastighet och lokal
Rekrytering anställda	Rekrytering	Affär
Rekrytering studenter	Rekrytering	Affär
Externa relationer	Relationer	Affär
Interna relationer	Relationer	Affär
Kalender	Resursbokning	Fastighet och lokal
Resursbokning	Resursbokning	Fastighet och lokal
Schemahantering	Schemaläggning	Utbildning
Studentantagning	Studentrekrytering	Utbildning
Examenshandläggning	Studieadmininstration	Utbildning
Kursvärderingsunderlag	Studieadmininstration	Utbildning
Labbanmälan	Studieadmininstration	Utbildning
Studiedokumentation	Studieadmininstration	Utbildning
Utbildningskatalog	Studieadmininstration	Utbildning
Bok- och tidsskriftsutlåning	Utlåning	Bibliotek

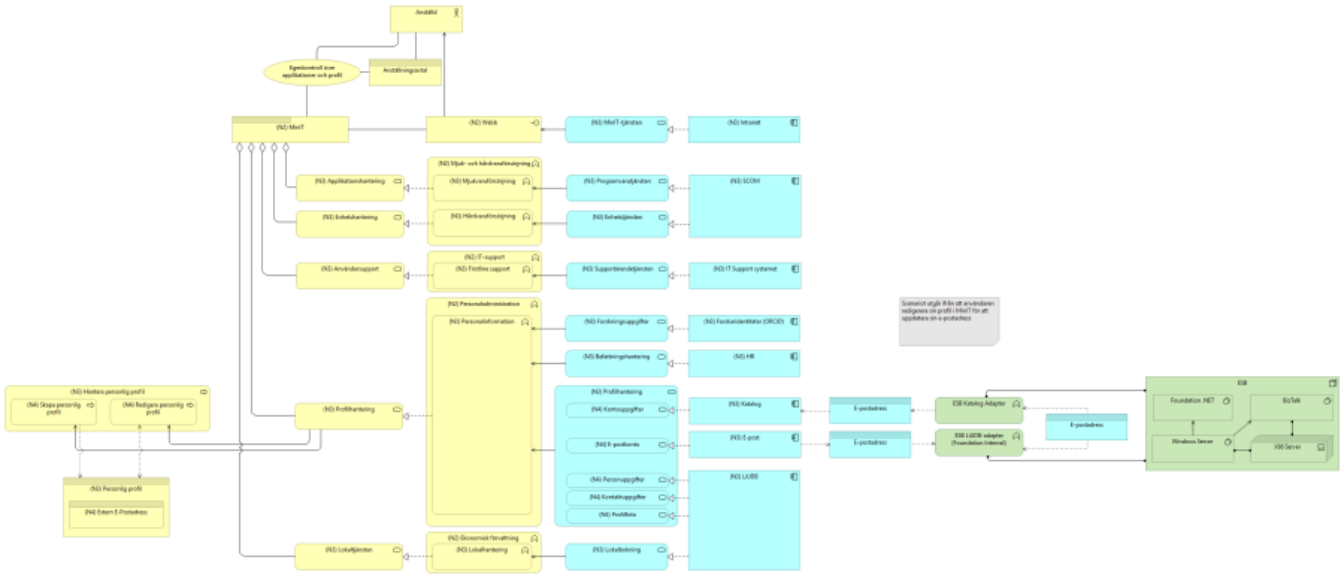
Bilaga 2 – Metamodellen



Bilaga 3 – Förslag på funktionsarkitekturen



Bilaga 4 – LiUs Min-IT


Anställd
Egenkontroll över applikationer och profil
Anställningsortal
(N2) MinIT
(N2) Webb
(N1) MinIT-tjänsten
(N2) Intranet
(N3) Applikationshantering
(N3) Mjuk- och hårdvaruförsörjning
(N3) Mjukvaruförsörjning
(N3) Hårdvaruförsörjning
(N1) Programvarutjänsten
(N1) Enhetstjänsten
(N1) SCOM
(N3) Enhetshantering
(N3) Användarsupport
(N2) IT-support
(N3) Förflow support
(N1) Supportavdelningstjänsten
(N2) IT Support systemet
(N2) Personaladministration
(N3) Personalinformation
(N3) Förfrågningsuppgifter
(N3) Befattningshantering
(N3) Förfrågningsidentiteter (ORCID)
(N2) HR
Scenariot utgår ifrån att användaren redigerara sin profil i MinIT för att uppdatera sin e-postadress
(N3) Hantera personlig profil
(N4) Skapa personlig profil
(N4) Redigera personlig profil
(N3) Profilhantering
(N3) Profilhantering
(N4) Kontouppgifter
(N4) E-postkonto
(N4) Personuppgifter
(N4) Kontaktuppgifter
(N4) Profilstatus
(N1) Katalog
(N1) E-post
(N1) LIA/BI
E-postadress
E-postadress
ESB Katalog-Adapter
ESB LIA/BI-adapter (Foundation Internal)
E-postadress
ESB
Foundation .NET
BizTalk
Windows Server
X86 Server
(N2) Personlig profil
(N4) Extern E-Postadress
(N2) Lokaltjänsten
(N2) Ekonomisk förvaltning
(N3) Lokalhantering
(N1) Lokalbokning