



**TOEPASSING VAN SEMANTISCH WEB TECHNOLOGIE VOOR HET  
BEHEER VAN BELEIDSINFORMATIE BINNEN DE VLAAMSE OVERHEID**

**PHILIPPE LUICKX**

Verhandeling aangeboden tot  
het behalen van de graad van  
Handelsingenieur in de Beleidsinformatica

Promotor : Prof. Dr. W. LEMAHIEU





Philippe Luickx

## Toepassing van Semantisch Web technologie voor het beheer van beleidsinformatie binnen de Vlaamse Overheid

Het Semantisch Web wordt de volgende stap in de evolutie van het Internet. Het wordt een Web waar gegevens van semantiek worden voorzien, zodat niet alleen mensen de inhoud kunnen begrijpen, maar ook computers. In deze eindverhandeling wordt het Semantisch Web grondig besproken. Zowel de filosofie erachter, de voor- en nadelen als de technische kant komen aan bod. Na deze bespreking volgt een case study die het gebruik van Semantisch Web technieken onderzoekt binnen de Vlaamse Overheid. Concreet wordt een ontologie opgesteld die het beheer verzorgt van beleidsinformatie van de Vlaamse Overheid. Een kritische reflectie sluit de eindverhandeling af.

Promotor : Prof. Dr. W. LEMAHIEU

# Dankwoord

Graag had ik mijn dank betuigd aan een aantal mensen werkzaam binnen de Vlaamse Overheid. Bedankt dhr. Van Herreweghe voor uw tijd, enthousiasme en goede raad. Bedankt mevr. Verhavert, dhr. Logghe, dhr. Devreese, dhr. Spitz, dhr. Tilkens en dhr. De Boeck voor uw geduld en tijd die deze eindverhandeling brachten tot het huidige resultaat.

Ook had ik graag mevr. Luyten en mej. Kuha bedankt voor het nalezen en de morele steun.

Tot slot kan ik mijn promotor en werkgever niet vergeten. De motivatie, steun en goede raad werden zeker geapprecieerd. Bedankt prof dr. Lemahieu en dhr. Poelmans.

## **Inhoudsopgave**

Hoofdstuk 1	Algemene inleiding .....	1
1.1	Introducerend voorbeeld .....	1
1.2	Definities .....	2
1.3	Het belang van informatie .....	3
1.4	Doelstelling eindverhandeling .....	5
1.4.1	Informatieproblematiek .....	5
1.4.2	Gegevens, informatie, kennis en wijsheid .....	7
1.5	Het verdere verloop van de eindverhandeling .....	9
Hoofdstuk 2	Het Semantisch Web .....	11
2.1	De visie van het Semantisch Web .....	11
2.1.1	Het huidige Web .....	11
2.1.2	Het Semantisch Web: een volgende stap? .....	12
2.2	Belangrijke aspecten van het Semantisch Web .....	15
2.2.1	Decentralisatie .....	15
2.2.2	Ontologieën .....	15
2.2.3	Inference .....	22
2.2.4	Agenten .....	23
2.2.5	Diensten .....	23
2.3	De voordelen van het Semantisch Web .....	24
2.3.1	Het belang van het Semantisch Web in het bedrijfs- en persoonlijk leven .....	24
2.3.2	Spam .....	26
2.3.3	Zoekopdrachten .....	27
2.3.4	Transparantie .....	27
2.3.5	Gecontroleerde woordenschat .....	27
2.3.6	Hergebruik .....	28
2.4	Obstakels voor het Semantisch Web .....	28
2.4.1	Kritiek op het Semantisch Web .....	28
2.4.2	Verzet .....	30
2.4.3	Het Semantisch Web is een hype .....	30
2.5	De toekomst van het Semantisch Web .....	31
2.5.1	Web 3.0 .....	31
2.5.2	Het Semantisch Web, een revolutie? .....	32

2.5.3	Adoptie van het Semantisch Web .....	33
2.6	Het Semantisch Web in actie .....	34
2.6.1	Bibliotheken .....	34
2.6.2	Use cases .....	35
2.6.3	Toepassingen Semantisch Web technologie .....	36
2.7	Het Semantisch Web: state of the art .....	39
2.7.1	Uniform Resource Identifier (URI).....	40
2.7.2	Unicode .....	41
2.7.3	eXtensible Markup Language (XML).....	41
2.7.4	Resource Description Framework (RDF) .....	43
2.7.5	SPARQL .....	46
2.7.6	RDF Schema (RDFS).....	47
2.7.7	Web Ontology Language (OWL).....	49
2.7.8	Rules.....	52
2.7.9	Logic .....	53
2.7.10	Proof.....	53
2.7.11	Trust .....	53
Hoofdstuk 3	Enterprise Architecture.....	55
3.1	Bespreking EA .....	55
3.2	Opbouw .....	57
3.3	Enterprise Architecture Frameworks .....	58
3.3.1	Bespreking EAF .....	58
3.3.2	Raamwerken.....	59
3.4	Enterprise Architecture en ontologieën.....	64
Hoofdstuk 4	Het gebruik van Semantisch Web technieken voor het beheer van beleidsinformatie binnen de Vlaamse Overheid .....	65
4.1	Vorbereiding project .....	65
4.1.1	Inzicht in de Vlaamse Overheid.....	65
4.1.2	Beleidsinformatie .....	66
4.1.3	Tools.....	68
4.1.4	Het oorspronkelijke RDF-model.....	72
4.2	Het nieuwe Protégé-model .....	76
4.2.1	Het gebruik van het Protégé-model gepositioneerd met EAF .....	77
4.2.2	De ontwikkeling van het nieuwe Protégé-model .....	79

4.2.3	Mogelijke uitbreidingen van het model .....	97
4.3	Aanbevelingen Vlaamse Overheid.....	98
4.3.1	Incrementele ontwikkeling en feedback.....	99
4.3.2	Communities of Interest.....	100
4.3.3	Architectuur.....	100
4.3.4	Andere mogelijkheden voor Semantisch Web technieken binnen de Vlaamse Overheid.....	101
4.3.5	Europa en de rest van de wereld .....	105
	Algemeen besluit: kritische reflectie en overgang naar het Semantisch Web .....	106

## Hoofdstuk 1 Algemene inleiding

### 1.1 *Introducerend voorbeeld*

The entertainment system was belting out the Beatles' "We Can Work It Out" when the phone rang. When Pete answered, his phone turned the sound down by sending a message to all the other *local* devices that had a *volume control*. His sister, Lucy, was on the line from the doctor's office: "Mom needs to see a specialist and then has to have a series of physical therapy sessions. Biweekly or something. I'm going to have my agent set up the appointments." Pete immediately agreed to share the chauffeuring.

At the doctor's office, Lucy instructed her Semantic Web agent through her handheld Web browser. The agent promptly retrieved information about Mom's *prescribed treatment* from the doctor's agent, looked up several lists of *providers*, and checked for the ones *in-plan* for Mom's insurance within a *20-mile radius* of her *home* and with a *rating* of *excellent* or *very good* on trusted rating services. It then began trying to find a match between available *appointment times* (supplied by the agents of individual providers through their Web sites) and Pete's and Lucy's busy schedules. (The emphasized keywords indicate terms whose semantics, or meaning, were defined for the agent through the Semantic Web.)

In a few minutes the agent presented them with a plan. Pete didn't like it: University Hospital was all the way across town from Mom's place, and he'd be driving back in the middle of rush hour. He set his own agent to redo the search with stricter preferences about *location* and *time*. Lucy's agent, having *complete trust* in Pete's agent in the context of the present task, automatically assisted by supplying access certificates and shortcuts to the data it had already sorted through. Almost instantly the new plan was presented: a much closer clinic and earlier times: but there were two warning notes. First, Pete would have to reschedule a couple of his *less important* appointments. He checked what they were: not a problem. The other was something about the insurance company's list failing to include this provider under *physical therapists*: "Service type and insurance plan status securely verified by other means," the agent reassured him. "(Details?)" Lucy registered her assent at about the same moment Pete was muttering, "Spare me the details," and it was all set. (Of course, Pete couldn't resist the details and later that night had his agent explain how it had found that provider even though it wasn't on the proper list.)

(Berners-Lee, Hendler & Lassila, 2001)



## 1.2 Definities

Om zeker te zijn dat bepaalde concepten correct geïnterpreteerd worden, volgt hier een lijst met definities. Deze definities zijn niet de “enige juiste” definities en over veel van deze termen bestaat dan ook een uitgebreide discussie. Ze dienen dus eerder als steun en zijn gebaseerd op wat gangbaar wordt aangenomen.

- **beleidsinformatie:** beleidsinformatie is de totaliteit van gegevens die een samenhangende visie geven op al het beleidsmatige bij de Vlaamse Overheid, van de regeringsverklaring tot de beleidsnota's en -brieven, documenten die het beleid definiëren enz.
- **Enterprise Architecture (EA):** de beschrijving van de huidige en/of toekomstige structuur en het gedrag van de processen, informatiesystemen, personeel en organisatorische subeenheden binnen een organisatie zodat deze overeenkomen met de doelen en de strategische richting van de organisatie
- **Enterprise Architecture Framework (EAF):** definieert hoe de structuren en verschillende visies georganiseerd worden in een EA; een EAF identificeert de informatietypes nodig om een EA voor te stellen, organiseert deze types in een logische structuur en beschrijft de relaties tussen deze types
- **gegevens** of **data:** een verzameling geïsoleerde feiten die eigenschappen representeren van objecten, gebeurtenissen en hun omgevingen (cf. §1.4.2)
- **inferentie (inference):** een conclusie afleiden gebaseerd op feiten die al gekend waren
- **informatie:** de betekenis die een mens uitdrukt of ontleent aan gegevens (cf. §1.4.2)
- **informatiesysteem:** het geheel van mensen, middelen en procedures (formeel en informeel (bv. wandelgangen)), dat samenwerkt ten behoeve van het verzamelen, vastleggen, verwerken en verstrekken van informatie; is nooit een doel op zich, maar heeft steeds een ondersteunende rol binnen de organisatie
- **kennis:** informatie over de wereld die ons toelaat een probleem op te lossen (cf. §1.4.2)
- **metadata:** data over data; gestructureerde informatie over gegevens, wat het beheer en terugvinden van informatie verbetert

- **netwerkeffect**: een fenomeen waar een dienst of een goed meer waarde verkrijgt naargelang meer en meer mensen deze dienst of dit goed hebben; bv.: in het begin van het Web, hoe meer mensen een internetpagina hadden, hoe meer hyperlinks gelegd konden worden, waardoor het interessanter werd om zelf ook een pagina te maken
- **ontologie**: een formele en expliciete specificatie van een conceptualisatie; een ontologie definieert een kennisdomein en de concepten en relaties die daarin bestaan; een ontologie maakt het delen en hergebruiken van kennis mogelijk
- **redenering (reasoning)**: het gebruik van de reden om inferences te vormen
- **semantiek**: de wetenschap die de betekenis van taalkundige constructies onderzoekt (bv. woorden en zinnen)
- **Semantisch Web**: een uitbreiding van het huidige Web waar informatie een welbepaalde betekenis heeft gekregen zodat computers en mensen beter kunnen samenwerken; een onderdeel van de *semantische technologie* dat gebouwd is op Web technologieën
- **Service Oriented Architecture (SOA)**: in een SOA worden bedrijfsprocessen aangeboden als diensten (services), die op hun beurt gecombineerd kunnen worden in andere (complexere) bedrijfsprocessen
- **technologie**: gebruik en kennis over hulpmiddelen; veroorzaakt geen verandering, maar maakt verandering mogelijk – schept dus kansen
- **Vlaamse Overheid**: verantwoordelijk voor het voorbereiden, formuleren en uitvoeren van het beleid van het Vlaamse Gewest en de Vlaamse Gemeenschap; bestaat uit 13 beleidsdomeinen (cf. §4.1.1)
- **webservice**: een softwaresysteem ontwikkeld om interactie tussen machines over een netwerk te ondersteunen
- **World Wide Web Consortium (W3C)**: Het W3C is de belangrijkste internationale organisatie voor standaarden op het WWW

### 1.3 *Het belang van informatie*

Informatie speelt in deze maatschappij een ongelooflijk belangrijke rol. Het aanmaken van zinvolle informatie en het delen hiervan met andere partijen, is het veld van menige studie, vooral dan in de ICT (Information & Communication Technology).

Competitieve voordelen worden nu vaak behaald door het efficiënt gebruiken van informatie.

Deze focus op kennis is er gekomen mede dankzij de technologische revolutie. Eén van de meest aanzienlijke evoluties van de voorbije jaren is de ontwikkeling van het Internet. Voor de komst van het Internet konden bedrijven wel communiceren met elkaar, maar hiervoor waren dure investeringen nodig (bv. EDI). Nu kan iedereen met heel de wereld in contact komen, zonder (al te) grote investeringen. Het Internet, vrij en beschikbaar voor de hele mensheid, werd het netwerk om kennis van heel de wereld met elkaar te linken (Klaus Riebschlager, 2005).

Het Wereld Wijde Web (WWW) heeft de manier waarop mensen met elkaar communiceren grondig veranderd. Het WWW is de kern van een revolutie die de ontwikkelde wereld verandert naar een kenniseconomie, een economie waar kennis als productiefactor centraal staat (ten opzichte van arbeid, natuur en kapitaal), en meer algemeen naar een kennismaatschappij (Antoniou & van Harmelen, 2004).

Een groot minpunt van het huidige WWW is dat er nog altijd een grote mate van menselijke betrokkenheid nodig is. Zo kan men het belang van zoekmachines niet ontkennen, maar deze tonen nogmaals aan dat de gebruiker zélf nog heel wat moet verwerken. Zoekopdrachten resulteren niet in een antwoord, maar in een lijst documenten waar *mogelijk* een antwoord te vinden is.

Dat HTML-pagina's vlot door mensen gelezen kunnen worden maar niet door machines, hangt daarmee samen. Webrowsers kunnen verschillende documenten wel verwerken en correct weergeven, maar ze kunnen deze niet zelf interpreteren. HTML-tags beschrijven enkel de presentatie van een document en kunnen slechts in zeer beperkte mate semantiek beschrijven met het meta-element, zoals `<meta name="keywords" content="semantisch, web, eindverhandeling"/>`.

Deze tekortkomingen worden verholpen in het Semantisch Web, het web zoals Sir Tim Berners-Lee (de peetvader van het huidige Internet) het van het begin af voor ogen had, maar wat twintig jaar geleden technisch nog niet haalbaar was. Het Semantisch Web is niet een web van HTML-documenten, maar een web van data, beschikbaar en begrijpelijk voor iedereen, zelfs computers.

## **1.4 Doelstelling eindverhandeling**

Het doel van deze eindverhandeling is te onderzoeken hoe men het beheer en gebruik van beleidsinformatie binnen de Vlaamse Overheid kan verbeteren. Dit gaat uit van het departement Bestuurszaken bij de Vlaamse Overheid, in samenwerking met dhr. Van Herreweghe. In §1.4.2 wordt uitgebreid over het verband tussen gegevens, informatie, kennis en wijsheid.

De reden voor dit onderzoek is dat veel bedrijven te kampen hebben met de informatieproblematiek (cf. §1.4.1). Hoewel de meeste bedrijven dit probleem onvoldoende erkennen, is het prominent aanwezig

Door gebruik te maken van nieuwe, opkomende technologieën zal men trachten deze informatieproblematiek aan te pakken. De eindverhandeling onderzoekt de mogelijkheden van Semantisch Web technieken en meer bepaald ontologieën. Het plaatsen van dit project in de juiste context zal ondersteund worden door Enterprise Architecture Frameworks (EAF).

Het mag duidelijk wezen dat deze eindverhandeling slechts een eerste (kleine) stap vooruit is. Wil men slagen in de opzet het informatieprobleem aan te pakken, zal men dit project moeten blijven opvolgen.

### **1.4.1 Informatieproblematiek**

Deze en volgende paragraaf zijn grotendeels gebaseerd op een boek van Hans van Heghe (2005. Leren zwemmen in informatie. ICMS Group n.v.).

Het loont de moeite even stil te staan bij de informatieproblematiek. Waar men vroeger een tekort had aan informatie, heeft men heden ten dage een overvloed aan informatie. Iedereen kan zonder veel moeite informatie aanmaken en publiceren. Deze mogelijkheden overtreffen echter de mogelijkheden om deze informatie te beheren, met als gevolg dat men vaak de indruk krijgt te *verdrinken* in informatie.

Informatieoverlast ontstaat als iemand

- de beschikbare informatie niet begrijpt
- zich overrompeld voelt door de hoeveelheid te begrijpen informatie
- niet weet of bepaalde informatie bestaat
- niet weet waar informatie gezocht moet worden
- weet waar te zoeken maar er geen toegang toe heeft
- niet weet of informatie betrouwbaar is

De gevolgen van deze overlast zijn dan ook verloren tijd, slechte besluitvorming en het onvermogen om kwaliteit te vinden tussen kwantiteit.

Verder blijkt het integreren van informatiebronnen en het behouden van consistentie een moeilijke opdracht. Vaak ontbreekt een raamwerk of methodologie om gegevens en informatie te beschrijven en worden enorme hoeveelheden gegevens gekopieerd, met redundantie en inconsistenties tot gevolg.

Om deze informatieproblematiek wat meer kracht te geven (bronnen: The Delphi Group, Gartner, IDC en coi.com):

- ongestructureerde informatie en kennis (bv. word-documenten, e-mails, kennis in de hoofden van mensen, enz.) maken 80% uit van informatie en kennis binnen een organisatie
- boekhoud-, ERP-, CRM-toepassingen beperken zich tot de 20% gestructureerde data (bv. data in databases) binnen de organisatie
- een werknemer spendeert gemiddeld 6 tot 12% van zijn tijd aan het zoeken naar informatie - die meestal niet wordt gevonden (27 tot 54 minuten per dag)
- een werknemer is 7 tot 20% van zijn tijd bezig met het herhalen van antwoorden voor collega's (32 tot 91 minuten per dag)
- werknemers spenderen gemiddeld 49 minuten per dag aan het *ordenen* van hun e-mail
- minder dan 20% van de informatie binnen een organisatie wordt effectief gebruikt
- 78% van de managers verklaart veel tijd te verliezen met het zoeken naar informatie

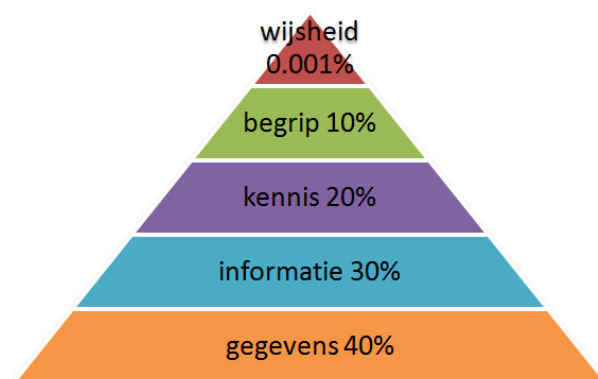
- 45% van de managers ervaart informatieoverlast als een groot probleem
- 35% van de managers verwacht dat dit probleem de komende twee jaar nog zal verergeren
- 50% van de informatie is opgeslagen op de verkeerde plaats
- meer dan 35% van de informatie zijn duplicaten

Wanneer men informatie systematisch begint te beheren, wordt een beter ondersteund beleid mogelijk. Goed informatiebeheer zorgt voor meer flexibiliteit, betere communicatie en een betere productiviteit. Dankzij dergelijk beheer wordt het mogelijk

- sneller en met hoge consistentie informatie te bewaren/op te slaan
- gericht en ergonomisch door informatie te grasduinen vanuit verscheidene invalshoeken
- bij het zoeken enkel de meest relevante informatie terug te vinden
- dynamische verbanden te leggen tussen gelijkaardige informatie
- informatie redundant op te slaan

#### 1.4.2 Gegevens, informatie, kennis en wijsheid

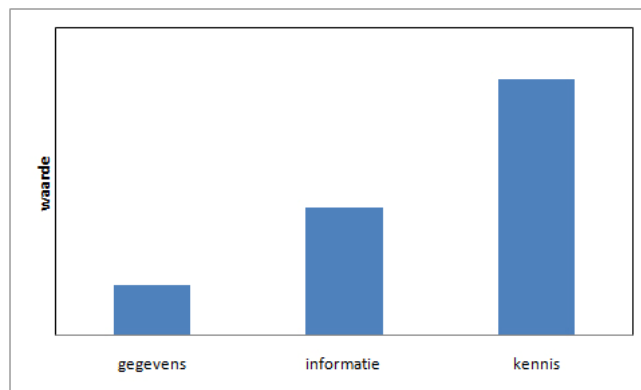
Veel mensen verwarren de termen gegevens, informatie, kennis en wijsheid. Dit is ook niet zo verwonderlijk aangezien de scheidingslijn niet erg strak is. Desondanks is het nodig het onderscheid te maken en de verbanden te leggen tussen deze termen.



**Figuur 1 Gegevens, informatie, kennis**

Figuur 1 toont een duidelijke hiërarchie: kennis is afhankelijk van informatie, wat op zijn beurt afhangt van gegevens.

**Gegevens** worden vaak beschouwd als *betekenisloze* informatie en bedrijven hebben een grote hoeveelheid gegevens opgeslagen in databanken en informatiesystemen. Ze zijn belangrijker dan applicaties, want applicaties kunnen vervangen worden, maar gegevens niet. Hoe meer verbanden een gegeven deelt met andere gegevens, hoe hoger de waarde ervan is.

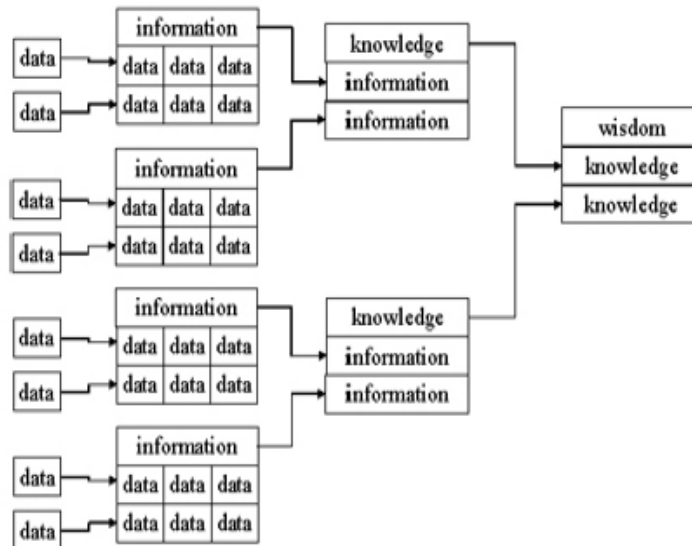


**Figuur 2** Relatieve waarde van gegevens, informatie en kennis

Wanneer gegevens in een context geplaatst worden en dus een betekenis krijgen, bekommt men **informatie**. Deze context hangt echter af van de *gebruiker* van de informatie en niet de *schepper* ervan. Bedrijven hebben een enorme schat aan informatie, opgeslagen in (ongestructureerde) bestanden, grafieken, e-mails enz. Informatie is de drager, het medium, van kennis.

**Kennis** is informatie over de wereld rondom ons die gebruikt wordt om een probleem op te lossen. Informatie wordt geïnterpreteerd om kennis te bekomen. Zo zal men informatie over een specifieke situatie vergelijken met andere, gekende situaties, de implicaties van bepaalde beslissingen anticiperen en interpretaties delen met andere mensen.

Wanneer kennis gebruikt wordt om tussen alternatieven te kiezen, wordt het gedrag *intelligent*. Als waarden en toewijding dit gedrag gaan sturen kan men zeggen dat dit gebaseerd is op **wijsheid**.



**Figuur 3 Data-informatie-kennis-wijsheid stroom (bron: Ahsan S. & Shah A.)**

### **1.5 Het verdere verloop van de eindverhandeling**

In vele bedrijven is er tegenwoordig onvoldoende zicht op de aanwezige gegevensbronnen en verloopt hierdoor de ontsluiting van informatie zeer moeizaam. Om hieraan tegemoet te komen wilde de Vlaamse Overheid onderzoeken wat de mogelijkheden zijn van Semantisch Web technieken voor het efficiënter ontsluiten van beleidsinformatie naar haar werknemers toe.

Na een bespreking van het Semantisch Web, worden Semantisch Web technieken onderzocht en toegepast in een case study van deze eindverhandeling. Eerst zal echter de plaatsing van deze technieken in de organisatie besproken worden aan de hand van Enterprise Architecture Frameworks (EAF).

De eindverhandeling omvat vier delen. Het eerste deel beschrijft de visie van het Semantisch Web en de technologie. Vervolgens omvat het tweede deel een korte literatuurstudie over verschillende Enterprise Architecture Frameworks. Deze bieden de context om de Semantisch Web technieken te plaatsen binnen het kader van een case study. De case study, het derde deel, onderzoekt de mogelijkheden van het gebruik van Semantisch Web technieken binnen de Vlaamse Overheid. Een kritische reflectie over het praktische gebruik en de implementatie van het Semantisch Web sluit als vierde deel af.



De visie, evolutie en state of the art van het Semantisch Web vormt het onderwerp van het eerste onderdeel. Een grondige bespreking van de technologie is noodzakelijk voor het goed begrijpen van het vervolg van de eindverhandeling.

Het Semantisch Web maakt gebruik van verschillende technologieën. Desgevallend spreekt men van de W3C Semantisch Web Layer Cake, een “cake” die bestaat uit een aantal lagen en samen een geheel vormen (cf. §2.7).

In deze volgorde zullen de talen XML, RDF, RDFS, OWL en SPARQL besproken worden. De aandacht zal uiteraard gaan naar het nut van deze talen voor de ontwikkeling van het Semantisch Web, aangezien veel (vooral de onderste lagen) van deze technologieën hun nut al hebben bewezen in andere toepassingen.

Het is echter niet de bedoeling om hier al te diep op in te gaan. Voor een diepere uitwerking kan men zich wenden tot de literatuur. Wel zal er getracht worden om de meest recente evoluties te schetsen.


Het tweede onderdeel behandelt kort enkele van de meer populaire EAF's. Dit onderdeel vloeit voort uit het gebruik van deze raamwerken voor de casestudy.

De casestudy, welke het onderwerp vormt van het derde deel, onderzoekt het gebruik van Semantisch Web technieken voor het beheer van beleidsinformatie binnen de Vlaamse Overheid. In eerste instantie zal er inzicht verworven worden over de organisatie van de Vlaamse Overheid en wordt een reeds bestaand model van gegevensbronnen onderzocht. Dit model zal als basis dienen om in een latere fase uitgewerkt te worden tot een volwaardige ontologie door uitbreiding en toevoeging van nieuwe elementen.

Het oorspronkelijke en nieuwe model zullen ook getoetst worden aan de besproken EAF's om hun plaats in de organisatie te kunnen duiden.

Ten slotte zal het bekomen resultaat kritisch gereflecteerd worden. Met de opgedane kennis zal gekeken worden naar de haalbaarheid van het Semantisch Web en hoe een overgang van web2.0 naar web3.0 (cf. §2.5.1) vlot kan verlopen.

## Hoofdstuk 2 Het Semantisch Web

Het Semantisch Web is het geesteskind van Sir Tim Berners-Lee en is ontwikkeld door het  W3C. Het wordt ook wel bestempeld als een internet waar computers met elkaar kunnen communiceren. In een eerste deel van dit hoofdstuk wordt de visie van het Semantisch Web uiteengezet, waarna een aantal belangrijke aspecten hiervan aan bod komen. Daarna worden de voor- en nadelen van het Semantisch Web tegenover elkaar gezet. Een vijfde deel werpt een blik in de toekomst, terwijl een zesde deel kort bespreekt wat er vandaag de dag al bestaat aan Semantisch Web toepassingen. Dit hoofdstuk wordt afgesloten met een bespreking van de technologieën waarvan het Semantisch Web gebruik maakt.

### 2.1 De visie van het Semantisch Web

#### 2.1.1 Het huidige Web

Het Internet dat in 1969 in de Verenigde Staten geboren werd, heeft de wereld volkomen veranderd. Niemand had zich kunnen inbeelden wat het resultaat zou zijn veertig jaar later. Wat oorspronkelijk louter bedoeld was als communicatiemiddel tussen onderzoekers is nu niet meer weg te denken. Het belang van het Internet vandaag hoeft niet meer aangetoond te worden.

De evolutie van het Wereldwijde Web (WWW) na 1969 is heel snel verlopen. Deze ontwikkelingen bieden veel nieuwe mogelijkheden, denken we maar aan multimediatoepassingen of webservices. Desalniettemin staat het Internet nog in zijn kinderschoenen en bevat het nog veel grondige tekortkomingen. Volgens Tim Berners-Lee, de persoon die beschouwd wordt als de peetvader het WWW, bevindt het Web zich nu in een *embryonische periode* en zal het enkel nog meer revolutionair worden.

De beginperiode van de ontwikkeling van het WWW verliep zeer chaotisch door gebrek aan een overkoepelend orgaan. Dit ontbreken van elk autoritair gezag en de decentrale structuur hebben de revolutie echter mogelijk gemaakt.

Op dit ogenblik ontwikkelen verschillende consortiums standaarden om orde op zaken te stellen. Het W3C (World Wide Web Consortium) is één van deze consortiums. We zien een duidelijke evolutie naar het respecteren van deze standaarden: webontwikkelaars en (ontwikkelaars van) webbrowsers gaan hier meer en meer rekening mee houden.

Ook zien we een opvallende evolutie op sociaal vlak. Mensen komen gemakkelijk met elkaar in contact op verschillende discussiefora, chatrooms en gemeenschappen zoals Myspace en Facebook. Het fenomeen is vooral de laatste jaren toegenomen. Waar voordien het Internet voornamelijk gebruikt werd om informatie van derden op te vragen, beoogt men met het huidige Internet een veel interactievere communicatie tussen de verschillende actoren. Dit *sociale* Internet wordt vaak aangeduid met de term web2.0. Met web2.0 verdwijnt de scheidingslijn tussen aanbieder en gebruiker van informatie en ontstaat er een platform voor wereldwijde samenwerking.

Men kan echter niet ontkennen dat het grote succes van het Internet ondenkbaar is zonder de zoekroboten. (Google, Yahoo,...). Deze vormen vaak het startpunt van een bezoek aan het WWW. Er zijn echter een aantal belangrijke nadelen verbonden aan zoekroboten: ze geven vaak teveel informatie, ze zijn te gevoelig voor woordenschat (synoniemen worden slechts beperkt herkend) en de resultaten blijven nog altijd webpagina's: documenten en dus niet antwoorden. De algoritmes voor het bekomen van resultaten verbeteren elke dag wel dankzij de evolutie in de Artificiële Intelligentie (AI), maar een andere aanpak biedt zich aan.

### 2.1.2 Het Semantisch Web: een volgende stap?

*I have a dream for the Web [in which computers] become capable of analyzing all the data on the Web – the content, links, and transactions between people and computers. A ‘Semantic Web’, which should make this possible, has yet to emerge, but when it does, the day-to-day mechanisms of trade, bureaucracy and our daily lives will be handled by machines talking to machines. The ‘intelligent agents’ people have touted for ages will finally materialize.*

(Berners-Lee, 1999)

Er zijn twee manieren voor machines om informatie te extraheren uit het Web, zodat mensen zelf niet meer moeten zoeken naar zinvolle informatie. Een eerste aanpak is het gebruik van intelligente software agenten, die hun informatie halen uit verschillende websites. Deze agenten worden alsmaar beter naarmate de Artificiële Intelligentie verbetert. Een andere aanpak vertrekt vanuit de data zelf, door deze leesbaar te maken voor machines. Zo bekomt men een verzameling *machinebegrijpbare data*, in tegenstelling tot de huidige verzameling *documenten*.

Vele onderzoekers zijn het erover eens dat het Semantisch Web een volgende logische stap is in de evolutie van het Internet. Het Web zoals het vandaag bestaat, is een netwerk van documenten. Websites worden voorgesteld in HTML-formaat en men kan talloze PDF-bestanden terugvinden, maar al deze informatie ligt dus opgeslagen in documenten. Documenten zijn zeer goed door mensen te interpreteren, maar computers hebben het daar heel wat lastiger mee. Een webbrowser kan wel weten dat een bepaald stukje tekst als een hoofding moet weergegeven worden, maar weet daarom niet waarover het gaat. Of zelfs niet dat het om een titel gaat...

Een netwerk van data lijkt dus het logische gevolg.

Dergelijk netwerk komt in principe neer op wat de Semantisch Web droom beoogt. Het Semantisch Web biedt een raamwerk dat toelaat data te hergebruiken over verschillende applicaties, organisaties en gemeenschappen heen. Het is een samenwerking van het W3C met een groot aantal academische onderzoekers en industriële partners. Met behulp van verschillende technologieën (XML, RDF, RDFS, OWL, SPARQL) zal men metadata construeren rond de gegevens die men wil delen. Deze data zullen dan niet alleen door mensen geïnterpreteerd worden, maar ook computers zullen gebruik kunnen maken van deze metadata. Computers zullen elkaar kunnen “begrijpen” wanneer ze met elkaar communiceren. Het Semantisch Web zal dus de manier om kennis te verwerken fundamenteel veranderen. Waar men vandaag nog gewoon *gegevens* met elkaar deelt, komt men met het Semantisch Web een stap dichterbij om *kennis* te delen met elkaar.

Het uiteindelijke doel is alle kennis expliciet op te slaan in een standaardformaat dat door applicaties “begrepen” kan worden. Applicaties zullen hierover kunnen redeneren en er nieuwe, betrouwbare kennis kunnen uit afleiden. Het Semantisch Web is een opportuniteit om de informatieproblematiek op te lossen in een interactieve omgeving, zodat iedereen de oplossingen kan delen en kan genieten van het netwerkeffect.

Het doel is echter nog altijd om het Web efficiënter te maken voor de gebruiker en het Semantisch Web zal dan ook hierop beoordeeld worden.

Het Semantisch Web zal verder waarschijnlijk nog een grote rol hebben in het toekomstig gebruik van *webservices*. Zoals vermeld zullen computers onderling met elkaar kunnen interageren. Wanneer een gebruiker via zijn browser een webservice aanroept, zal deze autonoom en transparant op zijn beurt de benodigde webservices aanroepen die het op dat moment nodig heeft. Dit zal enkel efficiënt gebeuren als de oorspronkelijke webservice zelf kan beslissen wat het nodig heeft, het benodigde kan opzoeken en tenslotte zelf kan kiezen welke van de gepresenteerde mogelijkheden optimaal is. De huidige technologieën (SOAP, REST, UDDI, WSDL) zijn hiervoor onvoldoende.

Het gaat zelfs nog verder: ook *intelligente omgevingen* zullen baat hebben bij de ontwikkeling van het Semantisch Web. Dankzij de ontwikkelingen in de draadloze communicatie en de miniaturisatie van chips kan men in dagdagelijkse gebruiksvoorwerpen intelligentie inbouwen. Dit kan gaan van lampen en centrale verwarming, tot kleding. Voor het adverteren van de aangeboden diensten moeten deze intelligente voorwerpen elkaar kunnen begrijpen. Een rijke semantiek is dus noodzakelijk.

Welke mogelijkheden het Semantisch Web nog allemaal zal bieden, is op dit moment onduidelijk. Er bestaat weinig consensus over de richting die het vroege Semantisch Web zal inslaan.

Wel blijkt dat informatie veel accurater, sneller en betrouwbaarder ter beschikking zal staan voor gebruikers. Deze informatie zal zo gedefinieerd en gelinkt worden dat ze gebruikt kan worden voor een efficiëntere ontdekking, integratie en efficiënter hergebruik over verschillende applicaties heen. De visie van de “peetvader” van het Internet, Sir Tim Berners-Lee, van het Web als *universeel medium voor de uitwisseling van data, informatie en kennis* zal misschien toch ontwikkeld worden.

## 2.2 *Belangrijke aspecten van het Semantisch Web*

In dit onderdeel worden een aantal fundamentele concepten van het Semantisch Web uitgelegd. De decentrale structuur komt aan bod alsook een uitgebreid onderdeel over ontologieën. Hierna wordt kort iets verteld over inference, agenten en diensten.

### 2.2.1 Decentralisatie

Het succes van het huidige Web is voor een groot deel te danken aan de decentrale structuur waarop het gebaseerd is. Voor de opkomst van het Web waren er al talrijke pogingen ondernomen om iets soortgelijks in gang te zetten, maar deze pogingen faalden hoofdzakelijk omdat ze vertrokken van een centrale structuur.

Als het Semantisch Web in zijn opzet wil slagen, moet het deze decentrale structuur aanhouden. *“Je kan niet van iets het centrum van kennis maken”* (Berners-Lee). Het is goed om in te zien dat het W3C slechts een hoop specificaties van standaarden en richtlijnen aanbiedt, samen met zijn filosofie en een hoop case studies. Maar uiteindelijk is men vrij om te doen wat men wil op het Semantisch Web.

Decentralisatie houdt echter een compromis in. Zo moest het huidige Web de totale consistentie van alle connecties (hyperlinks) laten vallen om te kunnen groeien. Iedereen komt daarom geregeld een *Error 404: Not Found* vermelding tegen.

De opoffering die onderzoekers van het Semantisch Web moeten doen is het aanvaarden van paradoxen en onbeantwoorbare vragen. Paradoxen worden deels opgevangen door de trust-laag van het Semantisch Web (cf. §2.7.11). Het probleem van de onbeantwoorbare vragen wordt voor een deel opgelost door de OWL-DL specificatie (in contrast met OWL-Full, cf. §2.7.7.3.2).

### 2.2.2 Ontologieën

Om het Semantisch Web te realiseren heeft men nood aan machineleesbare informatie. Ontologieën kunnen daarbij helpen. Een ontologie is een formele en expliciete specificatie van een conceptualisatie. Het is een formele en expliciete omschrijving van een set concepten binnen een domein en de relaties tussen deze concepten. Men onderscheidt verschillende klassen, beschreven door hun eigenschappen en attributen en beperkt door restricties.

Een ontologie samen met de instanties vormt een kennisbasis (knowledge base).

Binnen het Semantisch Web worden RDFS (cf. §2.7.6) en OWL (cf. §2.7.7) technologieën voorgesteld om ontologieën aan te maken. Deze bieden een rijke semantiek aan informatie door deze te voorzien van metadata en ze in een context te plaatsen.

#### *2.2.2.1 Nut van een ontologie*

Vooraleer een bedrijf alle data van semantiek zal voorzien en zich zo zal engageren in de Semantisch Web droom moeten de voordelen van het gebruik van ontologieën gekend zijn.

Het meest krachtige voordeel dat een ontologie biedt is een gemeenschappelijk begrip van de structuur van informatie. Zo kunnen mensen, maar vooral ook computers en software agenten informatie van verschillende bronnen analyseren, extraheren en integreren. Deze coherente gegevens kunnen dan gebruikt worden als input voor applicaties en queries.

Een bijkomend voordeel zijn de expliciete assumpties die gemaakt worden in een ontologie, welke niet hard gecodeerd zijn. Hierdoor zijn ze gemakkelijk aanpasbaar in het geval deze assumpties veranderen. Met andere woorden: aangezien de wereld rondom ons verandert, zullen ontologieën ook moeten veranderen. De specificaties die gemaakt worden in een ontologie zijn evenwel niet hard gecodeerd in een bepaalde applicatie, maar in een losstaand document, zodat deze op een flexibele wijze aangepast kan worden.

Dankzij ontologieën is het ook mogelijk de kennis omtrent het domein en de operationele kennis gescheiden te houden en is het ook mogelijk de kennis omtrent het domein te analyseren.

Belangrijk voor de ontwikkeling van ontologieën is dat bestaande ontologieën uit verschillende domeinen (bv. wiskunde en filosofie) gecombineerd worden in plaats van deze combinaties zelf van nul te gaan ontwikkelen. Een generieke basisontologie behandelt één kennisdomein zodat er verschillende, modulaire ontologieën ontworpen worden. Deze ontologieën kunnen dan samengevoegd en aangepast worden tot een ruimere, bruikbare ontologie voor een organisatie (cf. §2.2.2.4).

### 2.2.2.2 Vereisten ontologie

Het doel van een ontologie is expliciete en formele conceptualisaties van domeinmodellen te ontwerpen. Om aan deze doelstelling te voldoen, zijn er enkele vereisten (Thomas R. Gruber (1993). *Toward Principles for the Design of Ontologies Used for Knowledge Sharing*. Stanford Knowledge Systems Laboratory).

#### **Een goed gedefinieerde syntaxis**

Deze vereiste is duidelijk en geldt zowat voor elke programmeertaal. Om informatie te parsen op een computer, moet de syntaxis goed gedefinieerd zijn.

#### **Een formele semantiek**

De betekenis van kennis moet exact gedefinieerd zijn en mag dus niet afhankelijk zijn van subjectiviteit en intuïtie of van verschillende interpretaties. Deze vereiste is nodig om over kennis te kunnen redeneren. Waar mogelijk moeten definities compleet zijn en best ook gedocumenteerd in “normale” taal.

#### **Coherentie**

Gedefinieerde axioma's moeten logisch consistent zijn, maar ook de informeel gedefinieerde concepten (in bv. het machine onleesbare veld “omschrijving”) moeten logisch coherent zijn.

#### **Efficiënte reasoning ondersteuning**

Dankzij reasoning ondersteuning kan men meer informatie afleiden uit bestaande informatie. Dit kan automatisch gebeuren, waardoor men bv. de consistentie van een ontologie kan onderzoeken of instanties automatisch in klassen kan indelen. Ook vergemakkelijkt een efficiënte reasoning ondersteuning de integratie van verschillende ontologieën en de collaboratie van verschillende auteurs voor een grote ontologie.

#### **Voldoende expressieve kracht**

Een ontologie moet voldoende expressief zijn om kennis te kunnen modelleren. Er is echter een afweging tussen deze vereiste en de efficiënte reasoning support. Om gebruikers de keuze te laten tussen deze afweging, kan men bij OWL kiezen tussen drie subtalen: Full, DL en Lite (cf. §2.7.7.3.2).

#### **Gemak van expressie**

Hoewel de meeste ontwikkelaars nooit of amper geconfronteerd zullen worden met code, maar eerder zullen werken met tools, is het toch belangrijk dat de ontologie in een “simpele” expressieve taal uitgedrukt kan worden.



### **Uitbreidbaar**

Tijdens het ontwerp van een ontologie, moet men altijd in het achterhoofd houden dat deze uitbreidbaar moet zijn. Niet alleen moet het mogelijk zijn nieuwe concepten te introduceren, ook een bestaande ontologie implementeren moet mogelijk zijn. Daarom is een degelijke conceptuele basis nuttig, die toelaat de ontologie uit te breiden en te specialiseren, d.w.z. nieuwe termen introduceren gebaseerd op de bestaande ontologie, zonder deze te moeten aanpassen.

### **Minimaliseren van de encodings-bias**

Bij het ontwerpen van de ontologie moet men zich zo weinig mogelijk laten leiden door het gemak van notatie of implementatie. Concepten moeten gespecificeerd worden zonder af te hangen van specifieke encodingsschema's.

### **Minimale ontologische verbintenissen**

Een ontologie maakt best zo min mogelijk ontologische verbintenissen. Door meer algemene concepten te modelleren, kunnen gebruikers deze concepten in de ontologie gemakkelijker delen en zelf verder specialiseren. Op deze manier kan kennis op een consistente wijze gedeeld worden tussen verschillende gebruikers.

#### *2.2.2.3 Ontwerp van een ontologie*

Om een degelijke ontologie te ontwerpen, kan men best een aantal richtlijnen volgen. Deze zijn echter niet verplicht en er bestaat dan ook niet één juiste manier om een domein te modelleren. De beste oplossing zal altijd afhangen van de applicatie waarvoor ze ontwikkeld wordt (Natalya F. Noy & Deborah L. McGuinness (2001). *Ontology Development 101: A Guide to Creating Your First Ontology*. Stanford University).

### **Stap 1: Bepaal domein en werkgebied (domain en scope) van de ontologie**

De eerste stap in het ontwerp van een ontologie is een afbakening van het domein waarin gewerkt zal worden. Er is een kritische reflectie vereist van wat de dekking van de ontologie zal zijn, waarvoor men deze zal gebruiken, aan welke soort vragen ze een antwoord moet bieden en wie er gebruik van zal maken. Deze reflectie dient dan als houvast tijdens het ontwerp van de ontologie.

Het is echter niet een definitief denkkader voor het ontwerp, aangezien het antwoord op bovenstaande vragen op elk moment, tijdens en na de creatie van de ontologie, nog kan wijzigen.

Om zich een beeld te vormen van het werkgebied van een te ontwerpen ontologie kan men een lijst vragen opstellen die beantwoord moeten worden door de toekomstige kennisbasis. Deze vragen noemt men competentievragen en kunnen later ook gebruikt worden om de ontologie te testen.

### **Stap 2: Overweeg het hergebruik van reeds bestaande ontologieën**

Soms is het hergebruik van ontologieën onvermijdbaar wanneer het systeem moet samenwerken met reeds bestaande applicaties of ontologieën. Conversies zijn soms vereist maar veroorzaken over het algemeen geen problemen.

Het hergebruik van bestaande ontologieën kan een hoop werk besparen. Niet alleen moet men zelf niet meer de ontologie van nul af aan ontwerpen, de bestaande ontologie zal vaak ook al in gebruik genomen zijn en dus een zekere mate van betrouwbaarheid opgebouwd hebben. Ook voor integratie met andere ontologieën in een later stadium is het gebruik van generieke en populaire ontologieën een voordeel.

### **Stap 3: Som de belangrijkste termen van de ontologie op**

Deze stap in het proces laat toe om een eerste brainstorming te houden over de termen die in de ontologie moeten opgenomen worden. Men hoeft zich in deze fase nog geen zorgen te maken over overlappingsen, relaties of eigenschappen. Ze dient eerder om al een eerste keer na te denken over de verschillende mogelijke klassen.

### **Stap 4: Definieer de klassen en klassenhiërarchie**

Er zijn verschillende manieren om een klassenhiërarchie te definiëren:

Een *top-down* ontwikkeling begint met de meest algemene klassen binnen de ontologie en gaat deze dan verder specialiseren in verschillende subklassen.

Een *bottom-up* ontwikkeling beschouwt de meest specifieke klassen en tracht deze te groeperen in meer algemene concepten.

Deze twee manieren kan men ook *combineren* door eerst een paar algemene en meer specifieke termen te definiëren en deze dan samen te voegen in klassen tussenin.

De keuze van de aanpak hangt af van persoonlijke voorkeur. Uit de lijst van de vorige stap kiest men de termen die objecten beschrijven die “onafhankelijk” bestaan (bv. *naam* bestaat niet onafhankelijk, aangezien deze term afhangt van o.a. *persoon*; *persoon* is dan wél een object dat onafhankelijk bestaat).

Vervolgens moet men deze klassen hiërarchisch organiseren door te onderzoeken of de instantie van een bepaalde klasse per definitie ook een instantie is van een andere klasse.

### **Stap 5: Definieer de eigenschappen van klassen**

Klassen op zich bieden niet genoeg informatie. De structuur van deze concepten moet daarom beschreven worden met eigenschappen. Dit zijn vaak de overgebleven termen van de lijst in stap 3.

Voor elke eigenschap wordt bepaald bij welke klasse ze hoort. Deze eigenschap wordt best bepaald voor de meeste algemene klasse.

### **Stap 6: Definieer de facetten van deze eigenschappen**

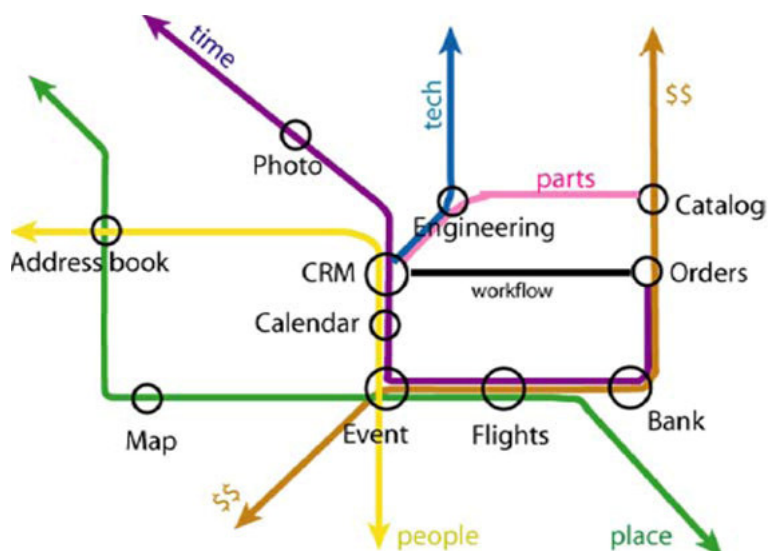
De eigenschap van een klasse bezit verschillende facetten die het waardetype (string, getal, boolean, opsomming, instantie) beschrijven, de toegestane waardes, het aantal waardes (kardinaliteit), enz.

### **Stap 7: Creëer instanties**

De laatste stap omvat de creatie van instanties in de klassenhiërarchie. Hiervoor kiest men eerst een klasse, creëert men een individuele instantie ervan en vult men de waardes in voor de eigenschappen.

#### *2.2.2.4 Ontologieën samenvoegen*

Het nut van ontologieën komt pas echt tot zijn recht wanneer het mogelijk wordt verschillende ontologieën samen te voegen tot een nieuwe, meer uitgebreide en bruikbare ontologie. Deze modulaire ontologieën zullen waarschijnlijk hun opmars vinden binnen hun eigen interessegemeenschappen, waar er het meeste nood is aan een kennisbasis.



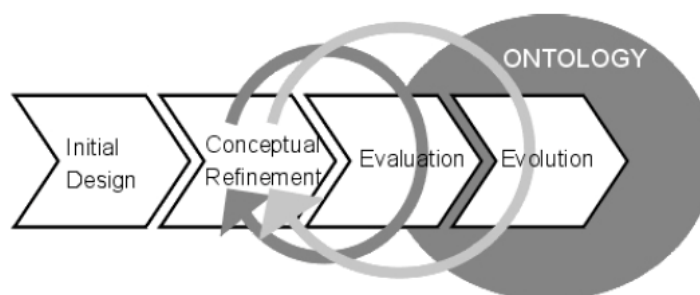
**Figuur 4 Modulaire ontologieën samenvoegen (bron: Bratt S. 2006)**

Zoals op figuur 4 te zien is, zijn er verschillende combinaties van generieke ontologieën mogelijk. Het probleem dat zich meteen aanbiedt is de ontwikkeling van verschillende generieke ontologieën die hetzelfde domein behandelen. Dit probleem zal na verloop van tijd op Darwiniaanse wijze opgelost worden, naarmate bepaalde ontologieën populairder worden. Deze evolutie zal dan de integriteit van het Semantisch Web beter ondersteunen.

Op dit ogenblik gebeurt dergelijke samenvoeging van ontologieën nog grotendeels manueel. Het doel is echter om ook dit proces te gaan automatiseren, maar dit is op het ogenblik van schrijven nog steeds een domein van onderzoek. Wel is het alvast nuttig te vertrekken van een basisontologie voor een gemeenschappelijke definitie van basistermen. Verschillende van deze basisontologieën bestaan reeds, zoals Dublin Core (cf. §4.2.2.1.2).

### 2.2.2.5 Levenscyclus ontologie

Het verloop van het ontwerp van een ontologie volgt een redelijk traditionele cyclus (Jérôme Euzenat (2002). Research challenges and perspectives of the Semantic Web).



**Figuur 5 Levenscyclus ontologie (bron: Euzenat J.)**

In een *behoefteanalyse* onderzoekt men het toepassingsdomein (doel, gebruikers) en maakt men de afweging tussen algemeenheid (voor hergebruik) en specificiteit. In het *initiële ontwerp* worden de verschillende kennisbronnen geanalyseerd en bekomt men informatie door taalanalyse, informatie-extractie, statistische analyse en machine-learning. Dit ontwerp gaat men vervolgens *conceptueel verfijnen*, vaak met de hulp van tools.

Voor de *evaluatiefase* gaat men de ontologie vergelijken met de oorspronkelijke vereisten en specificaties. Wanneer concepten aangepast moeten worden, gaat men een stap terug (verfijning), waardoor er een iteratief proces ontstaat.

Ten slotte moet men de ontologie ook in de tijd gaan beschouwen, waardoor ze onderhevig is aan *evolutie*. Er verschijnen nieuwe concepten of relaties veranderen en dit moet gereflecteerd worden in de ontologie. Zo ontstaat een tweede iteratieve cyclus waar het ontwerp van de ontologie na de evolutie terugkeert naar de conceptuele verfijning.

### 2.2.3 Inference

Inference is een belangrijk principe voor het Semantisch Web. Het houdt in dat nieuwe informatie afgeleid kan worden van reeds beschikbare informatie. Hierdoor ontstaat nieuwe kennis, waardoor een iteratief proces gestart wordt. Uiteindelijk kan het systeem tot een aantal conclusies komen.

Een inference engine is het mechanisme dat reasoning verzorgt. Deze techniek wordt al extensief toegepast binnen de domeinen van medicatie, medisch onderzoek en binnenlandse veiligheid.

Dankzij inference wordt het gemakkelijker applicaties te ontwerpen voor het Semantisch Web. Ook integratie van verschillende databronnen wordt vergemakkelijkt. Zo kan bv. afgeleid worden dat *postcode* en *zip-code* hetzelfde betekenen, waarna ze geïntegreerd kunnen worden.

#### 2.2.4 Agenten

Het Semantisch Web op zich doet niets. Het maakt enkel veel toepassingen *mogelijk*. De echte kracht van het Semantisch Web zal pas ontstaan wanneer programma's of agenten (agents) ontworpen worden, die informatie verzamelen, deze analyseren en de resultaten delen met andere programma's. De efficiëntie van deze agenten zal exponentieel verhogen (netwerkeffect) naarmate meer machineleesbare informatie beschikbaar is. Zelfs agenten die er niet voor geprogrammeerd zijn om met andere software samen te werken, zullen in staat zijn informatie te delen, omdat deze semantiek bevat.

Een andere belangrijke rol die agenten zullen spelen is het leveren van *bewijs* (proof, cf. §2.7.10). Gebruikers van agenten moeten op elk moment kunnen begrijpen hoe bepaalde informatie is afgeleid.

Ten slotte zijn agenten nodig om digitale handtekeningen na te kijken. Agenten zullen de bronnen van informatie nakijken en onbetrouwbare bronnen negeren indien nodig.

#### 2.2.5 Diensten

Dankzij semantiek kunnen aanbieders en gebruikers van webservices of services in een SOA elkaar veel gemakkelijker vinden. Ook bieden ontologieën veel meer flexibiliteit. In dit opzicht kan men ontologieën beschouwen als een stukje middleware zodat applicaties gemapt worden op dataniveau in plaats van op applicatieniveau.

### **2.3 De voordelen van het Semantisch Web**

In de beginfase van het informatietijdperk spitsten bedrijven zich toe op applicaties. Informatie wordt opgeslagen in verschillende formaten (documenten, spreadsheets, e-mails, enz.) en een heel scala aan heterogene databases. Integratie volgde pas daarna. Dit resulteerde in heel wat aanpassingen en legacy systemen, wat resulteerde in een ICT-architectuur zonder degelijke funderingen.

In de filosofie van het Semantisch Web gaat men data structuur en semantiek geven. Integratie bekomt men door gebruik te maken van open standaarden. Met de huidige evolutie naar een meer service-georiënteerde aanpak (SOA en webservices) heeft men misschien meer dan ooit nood aan een gemeenschappelijke en semantiekrijke manier om data te kunnen opslaan.

Het Semantisch Web zou een frisse wind kunnen betekenen doorheen veel bedrijven, die nu lijden onder het juk van legacy-oplossingen en, vaak veelgebruikte, zelf in elkaar geknutselde Excel of Acces programma's zonder enige vorm van integratie, met enorme redundantie als gevolg.

Het Semantisch Web gaat hoofdzakelijk over twee zaken:

- de integratie van data (uit verschillende bronnen) door gemeenschappelijke formaten
- een taal die kan gebruikt worden om te beschrijven hoe deze data zich verhoudt met de werkelijkheid

#### **2.3.1 Het belang van het Semantisch Web in het bedrijfs- en persoonlijk leven**

##### *2.3.1.1 Kennisbeheer*

In het huidige economische landschap is kennis één van de belangrijkste activa voor bedrijven. Kennis laat organisaties toe om productiever, innovatiever en flexibeler te zijn. Het beheer van kennis is dan ook een belangrijke zaak.

**Kennisbeheer** is alle activiteiten en technieken die erop gericht zijn kennis en informatie te verzamelen, te onderhouden en te gebruiken. Het omvat alle processen binnen een organisatie die een vlotte samenwerking trachten te bewerkstelligen tussen data en informatiesystemen enerzijds en de creatieve en innovatieve capaciteiten van de mens anderzijds. Informatietechnologie kan hier duidelijk bij helpen.



**Figuur 6 De plaats van kennis (MITRE 2004)**

Momenteel is informatie vervat in zwak gestructureerde vorm. Dit legt beperkingen op het zoeken, ontdekken, onderhouden en bekijken van informatie. Semantisch Web technieken laten een meer geavanceerde manier toe om met informatie om te gaan: kennis wordt opgeslagen volgens betekenis, inconsistenties worden automatisch gecontroleerd, nieuwe kennis wordt automatisch gevonden, queries worden krachtiger en zullen meerdere bronnen overspannen, enz.

Ook kan men dankzij Semantisch Web technieken verschillende databronnen veel vlotter en consistentier integreeren. Zo heeft men één grote bron aan informatie ter beschikking over verschillende applicaties en departementen heen. Dit levert uiteraard ook een enorm voordeel op bij fusies en overnames.

De grootste uitdaging zullen echter de werknemers zelf zijn. Ze moeten gemotiveerd worden hun kennis te delen en formeel op te slaan. De bedrijfscultuur zal dus aangepast moeten worden en dit is iets waar een technologie niet veel aan kan doen.

### *2.3.1.2 Business-to-Business Electronic Commerce*

In het begin van e-commerce werd voornamelijk Electronic Data Interchange (EDI) gebruikt voor datacommunicatie tussen bedrijven. Deze technologie is zeer complex en duur. Dankzij het Internet konden bedrijven hun communicatie afhandelen via het Web, maar degelijke standaarden ontbraken. HTML verzorgt enkel de visualisatie, maar dient niet als communicatiemiddel en hoewel XML al een stap in de goede richting is, heeft het nog altijd nood aan een a-priori overeenkomst van de gebruikte termen en hun betekenis.



Het Semantisch Web zal bedrijven toelaten met elkaar in contact te treden zonder grote kosten. Dankzij ontologieën weet men duidelijk wat bepaalde termen betekenen en software agenten kunnen (semi-)automatisch acties uitvoeren zoals veilingen, onderhandelingen en opstelling van contracten.

#### *2.3.1.3 Business-to-Consumer Electronic Commerce*

Het zoeken naar producten met gewenste specificaties zal een stuk gemakkelijker verlopen dan tegenwoordig mogelijk is. De gevonden producten vervolgens vergelijken en de beste prijs zoeken, zal met één muisklik kunnen gebeuren.

De huidige generatie *shopbots* hebben te kampen met enkele tekortkomingen. Ze gebruiken een heuristische aanpak gecombineerd met zoeken naar trefwoorden. Hierdoor zijn de resultaten niet altijd even consistent. Vaak laten ze ook veel relevante informatie achterwege (transportkosten, dienst na verkoop, garantie, enz.) zodat het vergelijken van producten nog altijd een moeilijke opdracht blijft.

Het Semantisch Web komt deze tekortkomingen tegemoet. Het zoeken naar en vervolgens integreren en vergelijken van informatie zal vergemakkelijkt worden dankzij ontologieën.

#### *2.3.1.4 Personal agents*

Het introducerend voorbeeld (cf. §1.1) is een mooi voorbeeld van het gebruik van personal agents. Dit zijn software applicaties die gebruik maken van Semantisch Web technieken om ons leven wat eenvoudiger te maken. Er zijn talloze mogelijkheden, zoals kalenderbeheer, contactenbeheer, documentenbeheer, zoekopdrachten over personen, producten, enz.

### **2.3.2 Spam**

Aangezien software beter zal “begrijpen” wat bepaalde informatie betekent, zullen e-mail applicaties kunnen onderzoeken of bepaalde berichten relevant zijn voor de gebruiker of niet. Het gebruik van trust (cf. §2.7.11) in combinatie met digitale handtekeningen, zet nog een stap verder.

### 2.3.3 Zoekopdrachten

Zoekopdrachten zullen sneller en nauwkeuriger uitgevoerd kunnen worden. Dankzij Natural Language Processing (NLP) zullen gebruikers van het Semantisch Web een vraag kunnen intypen en een antwoord ontvangen. Het project TrueKnowledge (cf. §2.6.3) is daar een mooi voorbeeld van.

Dit zal een grote impact hebben op ons leven. Niet alleen zal men sneller informatie kunnen terugvinden, men zal ook informatie kunnen vinden over zaken waar men het bestaan nog niet eens van af wist. Wanneer men grasduint door ontologieën, kan men nieuwe kennis ontdekken.

De slechte beschikbaarheid van correcte en relevante informatie op het juiste moment is anno 2008 één van de grootste frustraties, maar is (misschien binnenkort?) opgelost dankzij het Semantisch Web.

### 2.3.4 Transparantie

Veel van het Semantisch Web zal transparant kunnen gebeuren. Dit is belangrijk aangezien mensen zich vaak verzetten tegen meer werk.

Zo kan een persoon online een jobapplicatie invullen. Deze applicatie kan dan vorige jobs van deze persoon (ervaring) analyseren en verbinden met concepten in een tewerkstellingontologie om semantiek toe te voegen. De eindgebruiker merkt hier echter niets van.

### 2.3.5 Gecontroleerde woordenschat

Een ontologie biedt een formele definitie van termen die in een organisatie worden gebruikt. Op deze manier ontstaat een woordenschat die doorheen het hele bedrijf wordt gebruikt, wat de consistentie van informatie ten goede komt. Omdat werknemers dezelfde termen gebruiken, worden misverstanden voorkomen.

Wanneer bepaalde termen ambigu zijn, kan een ontologie deze termen verklaren. Zo kan de term *Dienst* zowel een departement zijn, als een dienstverlening voor klanten. Bij een zoekopdracht kan de applicatie dan aan de gebruiker vragen welke van de twee termen hij bedoelt.

### 2.3.6 Hergebruik

Omdat gegevens en informatie op een formele wijze wordt opgeslagen, kan deze kennis veel gemakkelijker gedeeld worden. Niet alleen operationele informatie, geïntegreerd over verschillende databronnen, kan hergebruikt worden. Domeinkennis wordt ook herbruikbaar.

Hier duikt het belang van ontologieën op. Een ontologie slaat op een formele wijze verschillende concepten van een bepaald kennisdomein op. Wanneer deze kennisdomeinen op een modulaire wijze van elkaar gescheiden blijven, kan men deze hergebruiken voor verschillende doeleinden (cf. §2.2.2.4).

## 2.4 *Obstakels voor het Semantisch Web*

### 2.4.1 Kritiek op het Semantisch Web

In 2001 omschreef Sir Tim Berners-Lee in een artikel het verwachte verloop van de evolutie van het Semantisch Web. Hoewel de filosofie achter het Semantisch Web niet te ingewikkeld lijkt, is een revolutie uitgebleven en is het gebruik van Semantisch Web technieken nog altijd zeer beperkt anno 2008. Men kan zich dus vragen stellen bij de haalbaarheid van het hele project. Even kort stilstaan bij de kritiek op het Semantisch Web zal daarom helpen in het identificeren en hopelijk vermijden van struikelblokken.

#### 2.4.1.1 *Complexiteit, meer werk en kritische massa*

Mensen die informatie ontwikkelen, moeten deze informatie nu vaak in twee formaten presenteren: één dat door mensen gelezen kan worden, een ander dat door computers begrepen kan worden. Niemand staat te wachten op meer werk. De vraag kan dan ook gesteld worden of het Semantisch Web zal doordringen bij de gemiddelde internetgebruiker, dan wel of het gebruik beperkt zal blijven binnen de academische en wetenschappelijke (medische of industriële) wereld.

In het begin van het Semantisch Web zal deze complexiteit zeker een probleem zijn. Naarmate de technologie doorbreekt zullen meer en meer applicaties transparant kunnen werken. Vergelijk het met webdevelopers die in het begintijdperk van het Internet nog HTML-code zaten te schrijven in Notepad, maar vandaag slechts een Content Management System moeten downloaden en installeren zoals Drupal of Mambo, zonder ooit een HTML-tag te hoeven aanschouwen.

Ook wordt door veel onderzoekers aangenomen dat er een zekere kritische massa nodig is vooraleer de ware revolutie kan beginnen, net zoals het met het WWW het geval was. Toen was het in het begin niet de moeite om een website te publiceren, aangezien er amper andere websites waren om naar te linken. Omdat er zo weinig websites waren, hadden ook maar heel weinig mensen een webbrowser. En waarom zouden ontwikkelaars een betere browser ontwikkelen als deze amper gebruikt wordt? Deze vicieuze cirkel moet op een bepaald ogenblik doorbroken worden door mensen die de stap durven te maken, waarna de populariteit explosief groeit.

Hier kan de Vlaamse Overheid een rol spelen. De Vlaamse Overheid biedt diensten aan personen en bedrijven aan. Bij het implementeren van het Semantisch Web, stijgt niet alleen de kwaliteit van deze diensten, de implementatie kan ook een stimulans zijn voor andere bedrijven om de Semantisch Web technologie te bestuderen en toepassen.

#### 2.4.1.2 *Schaalbaarheid*

Een andere tegenkanting is de schaal van het hele project. Een ontologie ontwerpen binnen één welbepaald domein is nog haalbaar, maar men kan zich de vraag stellen of men *alle informatie van heel de wereld* zal kunnen beschrijven in een ontologie. Het is duidelijk dat, net als Wikipedia, de inbreng van de gemeenschap (community) een belangrijke rol zal spelen. Hier is echter ook kritiek op...

*Doctorow's seven insurmountable obstacles to reliable metadata are:*

- *People lie*
- *People are lazy*
- *People are stupid*
- *Mission Impossible: know thyself*
- *Schema's aren't neutral*
- *Metrics influence results*
- *There's more than one way to describe something*

(Cory Doctorow, 2001)

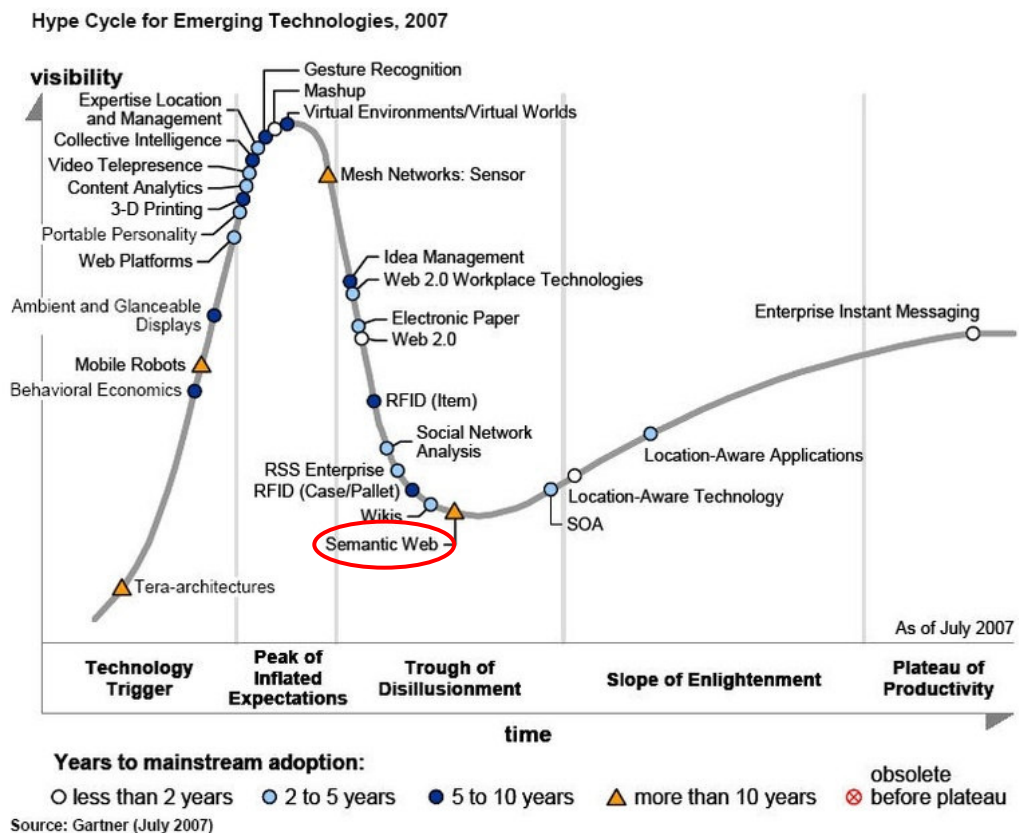
## 2.4.2 Verzet

Het grootste probleem zal de aanpassing van de mensen zelf zijn. Verzet ontstaat vaak vanuit de gebruikers, de werknemers van een organisatie. Ze zien het nut niet in om de dingen anders te gaan doen, als de huidige manier van werken (min of meer) voldoende is. Er ontstaat vooral verzet als de nieuwe methode (tijdelijk) meer werk inhoudt.

Enkele manieren om hiermee om te gaan is het publiceren van use cases om het gebruik en nut van het Semantisch Web aan te tonen. Ook pilootprojecten zijn aangeraden, zodat men uitgebreide feedback kan verzamelen en met concrete resultaten kan afkomen.

## 2.4.3 Het Semantisch Web is een hype

Veel onderzoekers die al verschillende hypes hebben meegemaakt welke nooit de verwachtingen inlossen, staan sceptisch tegenover het Semantisch Web. Hun is al zoveel keren beloofd dat één of andere technologie alle problemen zou oplossen, wat nog nooit gebeurd is. Waarom zou het deze keer anders zijn?



**Figuur 7** Gartner Hype Cycle for Emerging Technologies (bron: Garnter 2007)

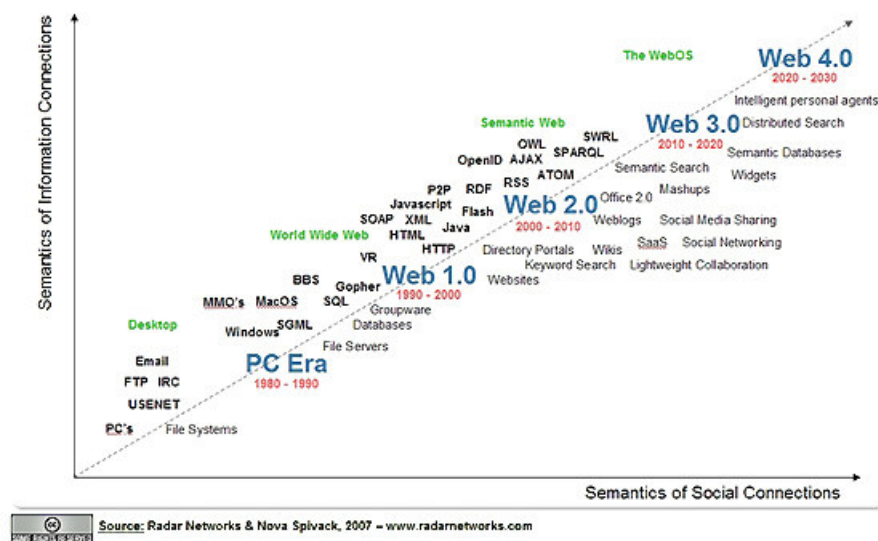
In een jaarlijkse analyse plaatst Gartner opkomende technologieën op hun hype-curve. In de figuur staat het Semantisch Web in het dieptepunt van de hype. De technologie leek veelbelovend en de verwachtingen waren dan ook hoog. Nu zijn de mensen echter ontuchtterd, aangezien er na zoveel jaar nog maar amper sprake is van het Semantisch Web. Nochtans is dit geen reden om alle hoop te verliezen. De technologie zal incrementeel in belang toenemen en langzaam productiever worden. Volgens Gartner kan dit echter nog meer dan tien jaar duren voor het Semantisch Web.

Of het Semantisch Web slechts een hype zal blijven of het beloofde zal waarmaken, zullen we binnen een aantal jaren pas merken. Maar nu al de moed opgeven en het Semantisch Web afschrijven getuigt van een onterecht pessimisme.

## 2.5 De toekomst van het Semantisch Web

Hoe de evolutie van het Semantisch Web zal verlopen, is niet te voorspellen. Naar alle waarschijnlijkheid zullen nog een aantal verrassingen voorkomen. In dit deel wordt de term Web 3.0 kort besproken en wordt hierna de vraag gesteld of het Semantisch Web eenzelfde revolutie zal teweeg brengen als de eerste dagen van het WWW. Een laatste onderdeel sluit af met de vraag hoe het Semantisch Web aanvaard zal worden.

### 2.5.1 Web 3.0



**Figuur 8 Web 3.0 (bron: Radar Networks 2007)**

De term Web 3.0 werd voor het eerst aangehaald in een artikel van de New York Times door John Markoff (2006). Hij verwijst daarmee naar de derde generatie van internetdiensten die samen het “intelligente Web” vormen. Hiervoor worden verschillende technieken gebruikt zoals het Semantisch Web, microformaten, Natural Language Processing (NLP), datamining, machinelere, aanbevelingsagenten en AI-technologieën. Deze benadrukken het begrijpen van informatie, geholpen door de computer, zodat de gebruiker een meer productieve en intuïtieve ervaring beleeft.

Volgens Nova Spivack moet het begrip van deze term nog uitgebreid worden met een paar andere technologieën die zullen bijdragen tot de ontwikkeling van Web 3.0. Zo moet de vooruitgang in connectiviteit (breedband, mobiel), netwerk-computing (grid computing, webservices), open technologieën (open API's, open data formaten, open source software) en open identiteit (OpenID) ook in rekening worden gebracht.

### 2.5.2 Het Semantisch Web, een revolutie?

Volgens Sir Tim Berners-Lee (2001) is het verkeerd het Semantisch Web te beschouwen als een technologie die het huidige Internet zal vervangen. Eerder zullen de mogelijkheden verruimd worden, omdat informatie een beter gedefinieerde betekenis zal krijgen, waardoor computers en mensen beter kunnen samenwerken met elkaar.

Op dit moment is het echter onmogelijk te voorspellen hoe het Semantisch Web het huidige Internet zal beïnvloeden. Semantisch Web applicaties zullen voor een grote verscheidenheid aan taken ingezet kunnen worden, waardoor ook de modulariteit van het Internet zal toenemen. Men ziet dus dat de toepassingsgebieden enorm breed liggen. Het zal dan ook interessant worden te zien hoe verschillende belangengroepen zullen omgaan met het Semantisch Web.

Zo kan men scenario's inbeelden waar Wikipedia 3.0, gebaseerd op het Semantisch Web, een “Ultimate Answer Machine” wordt, een soort van Globaal Brein dat zelf kan redeneren en afleiden en zo antwoorden kan leveren in contrast met Google's zoekmethodes die HTML-documenten aanlevert.

### 2.5.3 Adoptie van het Semantisch Web

Technologische veranderingen zijn vaak disruptief. De voordelen van de Semantisch Web technieken zullen gaan naar zij die begrijpen dat deze technologie een ontwerpstyl is die bedrijven toelaat hun flexibiliteit en wendbaarheid te verhogen. Om deze voordelen te behalen, zullen bedrijven het Semantisch Web moeten aanvaarden als een verandering in de levensstyl.

Nogmaals, Semantisch Web is iets wat een organisatie *doet*, niet iets wat het *koopt* of *uitbestedt*. Semantisch Web is een nieuwe wijze om systemen te ontwerpen en het gaat evenzeer over cultuur als over technologie. (Mitch De Felice, 2008)

Vooraleer het Semantisch Web zich kan ontplooien, moeten mensen die data creëren ervoor openstaan hun gegevens van semantiek te voorzien. Misschien zal dit zelfs de grootste uitdaging zijn voor de ontwikkeling van het Semantisch Web. Een veelbelovend onderzoeksdomein is folksonomie.

#### 2.5.3.1 Tagging en folksonomieën

Dat er meer dan ooit nood is aan metadata bewijst de populariteit van tagging op het WWW. Blog-artikelen, foto's, muziek: alles krijgt verschillende tags om de inhoud ervan in categorieën te definiëren.

Impliciet wordt er gesteld “*category = tag*”. Dit is slechts een zeer algemene context en vaak is er nood aan verduidelijking. Als er staat *category = Turkey*, dan kan er nog verwarring ontstaan. *Food = Turkey* of *Country = Turkey* maken gebruik van metatags (tags die informatie geven over tags, zoals metadata informatie geeft over data). Bepaalde metatags winnen aan populariteit, zoals *location* voor plaatsen en *favorite* voor favoriete boeken, films, enz.

Het proces waar collectief een informeel classificatiesysteem wordt ontwikkeld, wordt **folksonomie** genoemd. Daar waar in de klassieke aanpak van het Semantisch Web onderzoekers proberen overeen te komen over een formeel classificatiesysteem, ontstaan folksonomieën door het collectief taggen van verschillende individuen. Wanneer het systeem van tagging uitgebreid wordt met transparante Semantisch Web technieken, zal een veel rijkere semantiek ontstaan voor bronnen, die tot stand zijn gekomen door een grote gemeenschap gebruikers.



## 2.6 *Het Semantisch Web in actie*

De ontwikkeling van applicaties, ontologieën en databanken die gebruik maken van Semantisch Web technieken is al enkele jaren aan de gang. Desalniettemin is de aanwezigheid van deze toepassing marginaal op het huidige Web. Op dit moment is het nog wachten op de “killer-app” die de doorbraak van het Semantisch Web zal betekenen.

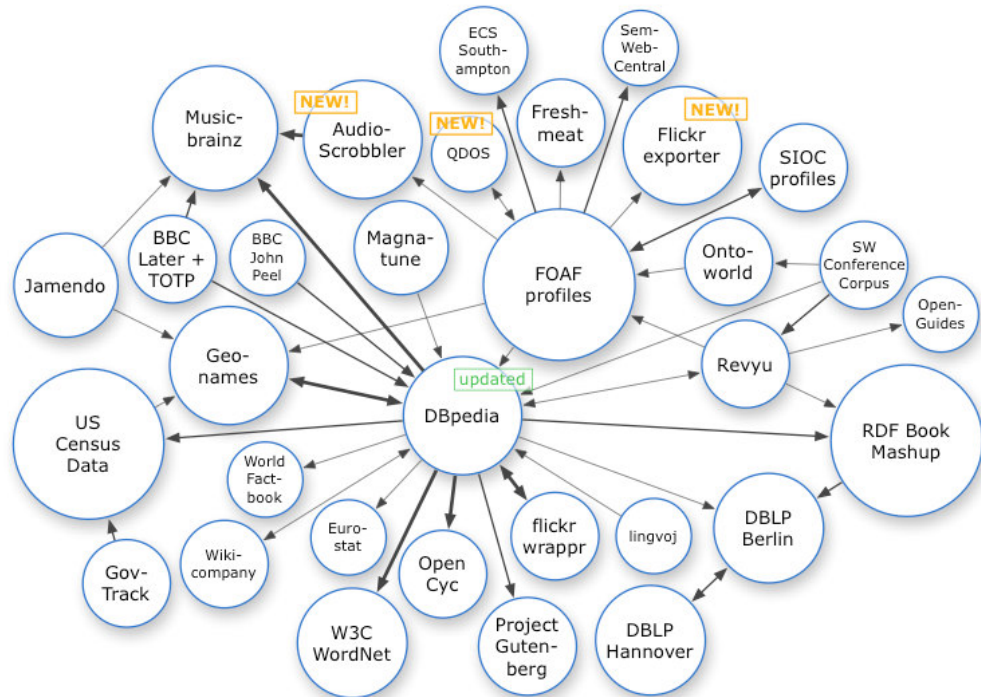
Een aantal projecten lijken veelbelovend, maar zijn nog in ontwikkeling (alpha of beta fase). Eerst zullen enkele bibliotheken (libraries) besproken worden die gegevens opslaan in een machineleesbare taal. Vervolgens worden enkele use cases gegeven, om af te sluiten met een aantal applicaties die draaien op Semantisch Web technologie.

### 2.6.1 Bibliotheken

Alvorens applicaties een nut hebben, is er nood aan gegevens. Voor het Semantisch Web zijn dit machineleesbare gegevens. Hoewel het Internet een gigantische bron aan informatie is, zijn alle gegevens opgeslagen in HTML-pagina's of databases. Twee mogelijkheden dienen zich aan: ofwel wordt er opnieuw begonnen en ontwikkelt men een bibliotheek in machineleesbare taal van nul af aan, ofwel wordt de reeds bestaande informatie omgezet naar het gepaste formaat.

**DBpedia** is een initiatief waar informatie van Wikipedia wordt omgezet. In plaats van een HTML-pagina over een onderwerp aan te bieden, gaat DBpedia gegevens over een onderwerp granulair opslaan.

Een ander project is **Freebase**, een open database die de ambitie heeft informatie van de hele wereld aan te bieden. Iedereen kan op deze database queries uitvoeren, gegevens toevoegen, integreren in websites of applicaties boven op Freebase bouwen.



**Figuur 9 Bibliotheken en interconnecties (bron: LinkingOpenData)**

### 2.6.2 Use cases

Enkele concrete implementaties van Semantisch Web technieken zijn al in gebruik. Een voorbeeld hiervan is **mutualart.com**. Dit project is een globaal initiatief om kunstverzamelaars in contact te brengen met artiesten, musea galerijen en relevante informatiebronnen. Mutualart.com is één van de eerste grote toepassingen die gebruik maken van Semantisch Web technieken om een dienst aan te bieden. Door het analyseren van voorkeuren, biedt de toepassing interessante informatie aan, gebruik makend van relaties tussen gebeurtenissen, artiesten, galerijen en musea.

In dezelfde culturele sfeer vindt men het project **Fundación Marcelino Botín**. Het doel van dit project is om een ontologie te bouwen die elf verschillende soorten cultureel erfgoed vastleggen voor de noordelijke Spaanse regio van Cantabria.

Het probleem was dat de informatie over het erfgoed heterogeen en gefragmenteerd was. Dit verhinderde een vlotte toegang tot informatie, maakte integratie moeilijk en het gebruik ervan op het Web quasi onmogelijk. Dankzij het gebruik van de Semantisch Web technologie is het nu mogelijk op een intelligente manier de relevante informatie over Cantabria's cultureel erfgoed te integreren en toegankelijk maken.

Een laatste toepassing werd gevonden in de **stad van Zaragoza**. De stad biedt meer dan 500 online diensten aan, elk georganiseerd in één van de twintigtal categorieën. Het vinden van een relevante dienst was vroeger niet altijd even simpel voor de inwoners. De informatie werd verschillend weergegeven op de websites van de stad, een geïntegreerde zoekopdracht bestond niet en om een oude wasmachine weg te doen, moest geweten zijn dat deze dienst “grote dingen collectie” heette. Door Semantisch Web technologie te combineren met Natural Language Processing heeft de stad een intelligente zoekrobot ontwikkeld, waardoor de inwoners van Zaragoza op een gemakkelijke en efficiënte wijze naar de relevante dienst kunnen zoeken.

### 2.6.3 Toepassingen Semantisch Web technologie

Het Semantisch Web op zich doet *niets*. Het is een verzameling technologieën met een achterliggende filosofie, maar het is slechts wanneer deze toegepast worden dat ze een nut hebben. Een aantal applicaties die gebruik maken van het Semantisch Web zijn in ontwikkeling, waarvan er hieronder enkele besproken worden.

De meest besproken applicatie is **Twine**, ontwikkeld door Radar Networks. Twine biedt een dienst aan om informatie te organiseren, te delen en te ontdekken. Met Semantisch Web technologie als drijvende kracht organiseert Twine automatisch informatie, leert interessegebieden van gebruikers en maakt aanbevelingen. Hoe meer je Twine gebruikt, hoe beter het je zal “begrijpen”, zodat het meer relevante aanbevelingen kan maken. Twine steunt ook sterk op het sociale gebeuren, door mensen samen te brengen die dezelfde interesses delen. Hier worden mensen niet samengebracht omdat ze elkaar kennen, zoals bij Facebook, maar omdat ze dezelfde interesses hebben. Vandaar dat de applicatie ook wordt beschreven als een tool voor “knowledge networking”.

De applicatie bevindt zich op dit ogenblik nog in bètafase, met een invite-only inschrijving. Door in te schrijven op de wachtlijst, werd na een tweetal maanden een uitnodiging verkregen en was het mogelijk om wat te kunnen “spelen” met Twine. In het begin is het hele gebeuren nogal verwarrend. Het vergt tijd en moeite voordat je de weg en het nut van de applicatie vindt. Uiteindelijk loont het wel, Twine is een nuttige applicatie gebaseerd op deugdelijke principes. Dit project dient verder niet alleen om gebruikers kennis te laten maken om met informatie op een semantische wijze om te gaan, maar ook om ontwikkelaars toe te laten om te analyseren hoe de gebruikers hiermee omgaan.

**Friend of a Friend (FOAF)** is een andere populaire toepassing van het Semantisch Web. Het is een project om een web van machineleesbare pagina's aan te maken die personen beschrijven en verbanden leggen tussen personen. FOAF vergemakkelijkt het delen en gebruiken van informatie over mensen.

Een andere applicatie die al een tijdje bestaat, is **mSpace**. Dit project biedt een interface aan, samen met een interactiemodel en software raamwerk, dat mensen helpt informatie te raadplegen. Een aantal krachtige methodes zijn beschikbaar om informatie te organiseren, zoals slicing, sorteren, swapping, infoViews en preview cues. Een demo is te vinden op <http://demo.mspace.fm/>.

Een interessante dienst die een eerste stap zou kunnen betekenen in de adoptie van het Semantisch Web is **Semantify** van Dapper.net. Met deze dienst kunnen webontwikkelaars hun webpagina's van semantiek voorzien op een simpele en vlotte wijze. Door specifieke velden op een aantal webpagina's te identificeren, zoals titels van boeken, blogposts, vrienden, enz. kan Semantify de rest van de website automatisch voorzien van de nodige semantiek, enkel door een klein stukje PHP-code in de header toe te voegen. Deze metadata kan vervolgens op intelligente wijze gebruikt worden door een semantische zoekrobot (zoals swoogle.com of sindice.com). Semantify is voorlopig nog in bètafase.

Ten slotte is ook **True Knowledge** de moeite van het vermelden waard. Evenals Twine bevindt dit project zich nog in bètafase, met een invite-only inschrijving. Ook hier kon de applicatie uitgeprobeerd worden na een maand op de wachtlijst te staan.

True Knowledge combineert Natural Language Processing (NLP) en Semantisch Web technologie om gebruikers toe te laten een vraag te stellen en een antwoord terug te krijgen. Wanneer een gebruiker een vraag stelt, gaat NLP deze vraag ontleden zodat de computer deze kan begrijpen. Door informatie op te slaan in machineleesbare taal, kunnen afleidingen gemaakt worden die leiden tot de benodigde informatie. Als resultaat krijgt de gebruiker niet alleen het antwoord, maar ook de mogelijkheid om deze afleiding te zien.

Op de vraag “Is Tony Blair older than George W. Bush?” geeft de zoekrobot het volgende antwoord:

No

There were two interpretations of your question, but they both resulted in the same answer. When you said "George W. Bush", I understood you to mean one of the following:

- George Washington Bush (1779-1863), one of the first American settlers and the first African American settler in what would later become the state of Washington
- George W. Bush, the 43rd President of the United States

I used the following facts to provide this answer:

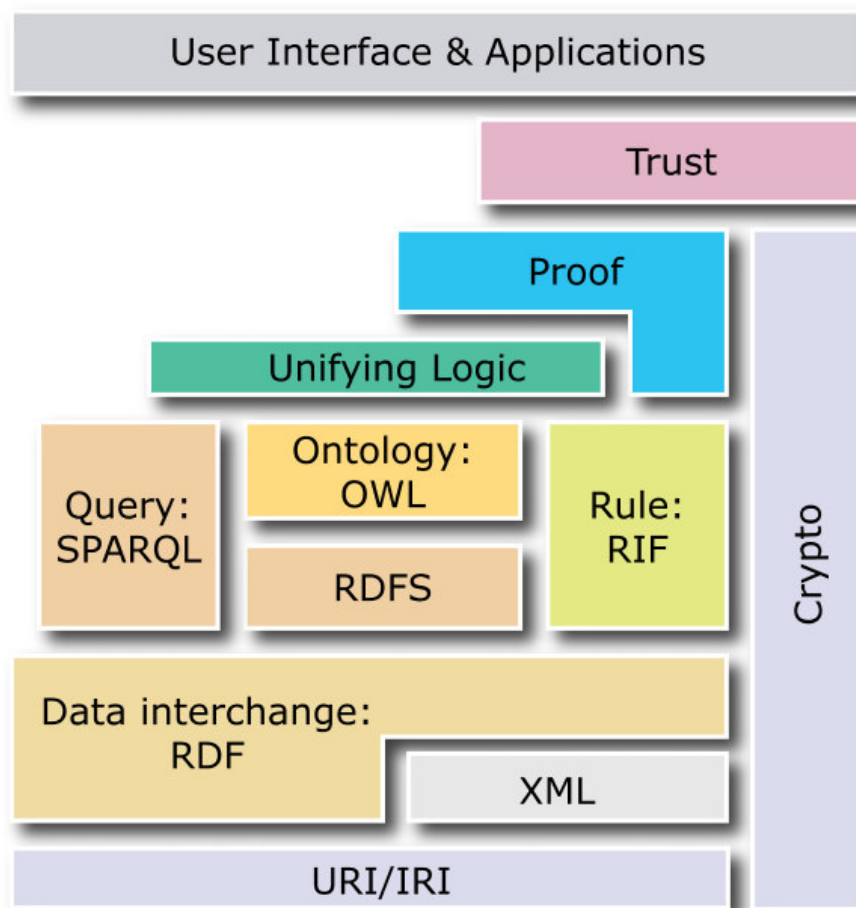
- [thing that was created is the left class of 'is older than'](#)
- [thing that was created is the right class of 'is older than'](#)
- [the 6th of May 1953 is the birthdate of Tony Blair \(endorse\) \(contradict\)](#)
- [1779 is the birthdate of George Washington Bush \(endorse\) \(contradict\)](#)
- [the 6th of July 1946 is the birthdate of George W Bush \(endorse\) \(contradict\)](#)

(I understood your question to mean: Determine whether Anthony Charles Lynton Blair (born 6 May 1953), the British politician who served as the Prime Minister of the United Kingdom from 2 May 1997 to 27 June 2007 is older (has been in existence longer) than George W. Bush (an ambiguous term) at the current time?

In mijn opinie is dit één van de Semantisch Web applicaties die het zullen maken. Uiteraard was het niet moeilijk om vragen te bedenken waar True Knowledge geen antwoord op had. Maar dit lag niet aan de technologie, eerder aan het feit dat de database nog uitgebreid moet worden. Op dit moment is al een vrij actieve gemeenschap hiermee bezig en dankzij een stem-systeem kan onjuiste of onbetrouwbare informatie weggefilterd worden. Dergelijke applicatie zal staan of vallen met de acceptatie van de online gemeenschap, maar als gekeken wordt naar het succes van Wikipedia, kan men verwachten dat projecten zoals True Knowledge heel snel zullen doorbreken.

### 2.7 *Het Semantisch Web: state of the art*

Om data leesbaar te maken voor machines is een laag met semantische informatie nodig. Voor dit doel stelt het W3C volgend raamwerk voor:



**Figuur 10** Semantisch Web layer cake (bron: W3C)

Uit dit raamwerk komt duidelijk naar boven dat het Semantisch Web een combinatie is van verschillende bestaande technologieën. Samen voorzien ze data van de nodige semantiek.

Het Semantisch Web is echter meer dan deze technologieën en anderzijds worden veel van deze technologieën ook buiten het Semantisch Web gebruikt.

Langs de ene kant zijn de technologieën een aanbeveling van het W3C, maar kan men ook andere technologieën gebruiken dan dewelke in de figuur zijn opgenomen. RDF beschrijft bronnen, maar dat kan eventueel ook door een andere taal gebeuren (N3). Figuur 10 geeft dus slechts een mogelijke, aanbevolen manier om het Semantisch Web te implementeren, maar er bestaan verschillende alternatieven. Het is dan ook nuttig om het Semantisch Web eerder te beschouwen als een concept, als een filosofie dat gebruik maakt van verschillende technologieën uit verschillende domeinen, dan als een technisch gegeven.

Langs de andere kant bestaan enkele van deze technologieën al een tijdje en worden ze dus ook buiten het Semantisch Web gebruikt. Zo bewezen XML en XMLSchema hun nut reeds in vele andere toepassingen. In het kader van deze eindverhandeling beperkt de bespreking die nu volgt zich enkel tot de toepassingen van de technologieën binnen het Semantisch Web.

Hierna volgt een beschrijving van elke laag van deze layer cake, te beginnen onderaan. De technologieën Unicode, URI, XML, RDF, RDFS, SPARQL en OWL zijn alle aanvaard als standaard. Formaten voor rules, logic, proof en trust zijn nog een gebied van onderzoek, waarvoor er nog geen standaarden bestaan.

### 2.7.1 Uniform Resource Identifier (URI)

URI staat voor Uniform Resource Identifier. Het is een string van tekens waarmee een bron (resource) wordt geïdentificeerd. Een bron kan vele vormen aannemen: een stuk tekst, een afbeelding, een programma, enz. Het meest bekende voorbeeld van een URI is een URL (Uniform Resource Locator).

```
http://www.w3c.org/
```

Het aanmaken van URI's is gedecentraliseerd, wat wil zeggen dat iedereen een URI kan aanmaken. Dit brengt een enorme flexibiliteit met zich mee, maar ook een aantal negatieve aspecten: verschillende URI's kunnen naar hetzelfde object verwijzen, sommige URI's zijn niet meer up-to-date, enz.

## 2.7.2 Unicode

Unicode is een universele coderings-industriestandaard om tekstuele data te representeren en manipuleren op computers. De standaard zet een reeks bytes, leesbaar voor de computer, om naar een karakter, leesbaar voor de mens. Met Unicode kunnen er ongeveer 100.000 karakters weergegeven worden en elk van deze karakters heeft een uniek nummer. Unicode ondersteunt de meeste talen in de wereld en is niet afhankelijk van platform, applicatie of taal.

XML is gebaseerd op Unicode.

## 2.7.3 eXtensible Markup Language (XML)

### 2.7.3.1 Inleiding

XML (eXtensible Markup Language) is een taal waarmee men informatie zo kan opstellen en structureren zodat ze gemakkelijk gedeeld kan worden. In XML kan men zelf tags aanmaken, om een eigen (boom-)structuur aan te brengen. Deze tags zeggen echter nog niet veel over wat deze structuur eigenlijk betekent. Wel kan men dankzij deze enorme flexibiliteit alle toekomstige scenario's anticiperen en is XML dan ook zeer geschikt als syntaxis voor alle talen in de Semantisch Web layer cake.

XML is een subset van SGML (Standard Generalized Markup Language, een internationale standaard ISO 8879 om informatie weer te geven) en werd door het W3C ontwikkeld om *data te beschrijven* (vergelijk met HTML: beschrijft de representatie van webpagina's). De doelstelling van het XML is om applicaties en webservices gemakkelijker met elkaar te laten communiceren.

XML-documenten die aan alle regels voldoen (bv. correcte nesting) zijn *well formed*. XML-documenten die daarboven nog aan een schema (bv. XMLSchema, cf. §2.7.3.2) voldoen zijn *valid*.



```
<eindverhandeling>

  <onderwerp>Semantisch Web</onderwerp>

  <auteur>Philippe Luickx</auteur>

  <academiejaar>2007-2008</academiejaar>

  <promotor>prof. dr. Lemahieu</promotor>

</eindverhandeling>
```

Uit het voorbeeld blijkt duidelijk de boomstructuur van een XML-document.

### 2.7.3.2 XMLSchema

Een XMLSchema licht niets toe over de semantiek van een document, maar beschrijft enkel een formele vorm waaraan een XML-document moet voldoen om valid te zijn.

XSD (XML Schema Definition) is een W3C recommendation en wordt zelf volledig in XML beschreven. Het biedt een manier om regels te beschrijven zodat machines en mensen de XML document structuur kunnen begrijpen en integreren. Niet alleen gebruikt XSD zelf XML (waardoor geen nieuwe parser moet worden ontwikkeld), maar kan men met XSD ook bestaande schema's hergebruiken en verfijnen.

Zonder in details te treden, definieert een XSD de verschillende element types, attribute types, data types, data type extensions en data type restrictions van een XML-document. Aangezien een groot deel van deze functionaliteit wordt aangeboden in de hogere lagen van de Semantisch Web layercake, wordt XMLSchema enkel (eventueel) gebruikt voor het definiëren van datatypes binnen de Semantisch Web context.

### 2.7.3.3 Namespaces

Dankzij de enorme flexibiliteit van XML als universele markup-taal, kan het voorkomen dat een tag in een XML-document iets anders betekent dan dezelfde tag in een ander document. Om dit te voorkomen, heeft men een oplossing naar voor gebracht om elke tag universeel uniek te maken. Deze oplossing bestaat erin elke tag te definiëren in zijn “omgeving”, de namespace. Concreet doet men dit met behulp van een prefix:

```
prefix:name
```

Een namespace zelf wordt gedefinieerd door een URI, wat vaak een (al dan niet bestaande) URL is.

Deze manier van werken garandeert een unieke verzameling elementen: men gaat eerst de namespace vastleggen en binnen deze omgeving zal men dan unieke elementen kunnen gebruiken. De namespace bepaalt context waarin men werkt, wat ook een belangrijke semantische betekenis kan hebben.

## 2.7.4 Resource Description Framework (RDF)

### 2.7.4.1 *Bespreking*

RDF (Resource Description Framework) verhoudt zich tot data zoals HTML zich verhoudt tot documenten (Berners-Lee). RDF is een raamwerk om datamodellen en hun relaties te definiëren. Het raamwerk kan beschouwd worden als een formaat om kennis weer te geven, waarbij het data voorziet van een eerste laag metadata.

Een RDF-document beschrijft dat bepaalde *resources* (mensen, webpagina's, landen, kortom: alles wat met een URI bepaald kan worden) bepaalde *eigenschappen* (beschrijving van de relatie, op zich ook een resource) hebben met bepaalde *waardes* (specifieke waardes (Unicode string) of ook een URI). Deze drie elementen, of triplets, en hun relaties worden RDF-statements genoemd. Terwijl XML dus structuur geeft aan data, wordt RDF gebruikt om betekenissen uit te drukken. Deze uitdrukkingen creëren een web van informatie over gerelateerde resources.

RDF kan gebruik maken van verschillende syntaxen, “talen” die gebruikt worden om de RDF triplets weer te geven. De syntaxis die aangeraden wordt door het W3C is XML, zodat men in dit verband spreekt over RDF/XML. Alternatieve syntaxen zijn N3 (Notation 3) en Turtle.

Een triplet wordt best beschreven door drie URI's. De uitdrukkingen die ze vormen, leggen de basis voor het Semantisch Web. Ze leggen associaties vast tussen betekenis enerzijds en oorzaak en gevolg anderzijds, van concepten en ideeën die in computertaal voorgesteld kunnen worden.

RDF beperkt zich met uitdrukkingen tot binaire eigenschappen:

```
P(x, y)

<rdf:RDF xmlns:pl="http://www.philippeluickx.be/">

  <pl:isAuthorOf>

    <pl:philippeluickx>

      <pl:eindverhandelingSemWeb>
```

Uit het voorbeeld blijkt duidelijk de graafstructuur van RDF. De resources  $x$  (voorgesteld door de URI <http://www.philippeluickx.be/philippeluickx>) en  $y$  (voorgesteld door de URI <http://www.philippeluickx.be/eindverhandelingSemWeb>) worden gerelateerd door het binaire predikaat  $P$  (voorgesteld door <http://www.philippeluickx.be/isAuthorOf>). Vaak spreekt men in dit verband over object (resource) en subject (value) die gelinkt worden door het predikaat (attribute).

Wegens deze vrij simpele structuur, kunnen zeer krachtige applicaties gebouwd worden. Doordat resources gedefinieerd zijn door URI's, verzekert men dat het niet slechts over "woorden" gaat, maar over concepten die wel degelijk bestaan. Deze structuur heeft echter ook enkele nadelen:

- het is moeilijk relaties te beschrijven met meer dan twee argumenten
- het krachtige mechanisme van reification (cf. §2.7.4.4) lijkt misplaatst in een simpele taal zoals RDF
- de syntaxis gebaseerd op XML is bruikbaar voor machines maar niet gebruiksvriendelijk voor mensen, waardoor men nood heeft aan gepaste tools

#### 2.7.4.2 Voordelen

De structuur van RDF is een zeer natuurlijke manier om het gros van data te beschrijven dat verwerkt moet worden door mensen en machines. Object en subject worden beiden gedefinieerd door URI's en ook de predikaten kunnen geïdentificeerd worden door URI's, waardoor iedereen nieuwe concepten kan ontwerpen.

Het belangrijkste voordeel dat RDF biedt is een deel semantiek, dat gebruikt kan worden door intelligente software en agents. Zo kan men dankzij dit consistent raamwerk gemakkelijker software ontwikkelen die gebruik maakt van semantiek, zal men gemakkelijker gegevens kunnen uitwisselen en zullen zoekresultaten meer accuraat zijn. Kortom, de data die gebruikt worden zullen preciezer en correcter zijn.

Een ander voordeel van RDF is dat het mogelijk wordt zoekopdrachten te volbrengen in geen tijd. Zo hebben onderzoekers van de National University of Ireland, Galway's (NUIG) Digital Enterprise Research Institute (DERI), een zoekmachine ontwikkeld die minder dan een seconde nodig heeft om meer dan zeven miljard RDF-statements te doorzoeken (04/05/2007).

#### 2.7.4.3 *Gebruik*

RDF stelt een taal ter beschikking om gegevens direct en duidelijk naar een model te mappen, terwijl dit model gedecentraliseerd is (belangrijk voor de schaalbaarheid) en waarvoor al vele XML-parsers aanwezig zijn. Hierdoor is het voor elke applicatie duidelijk waar de semantiek in een document zich bevindt en waar de syntaxis.

Een ander meer praktisch probleem is hoe men deze RDF data zal creëren. Het mag duidelijk wezen dat men dit niet allemaal manueel zal doen. Een dergelijke inspanning zou economisch niet haalbaar zijn. Het doel is daarom om zoveel mogelijk RDF-data automatisch te genereren en waar nodig manueel te corrigeren en aan te vullen.

Er bestaan verschillende convertoren om data van applicaties om te zetten naar RDF. In de huidige informatiesystemen wordt veel gebruik gemaakt van relationele databases. Deze databases bevatten al een belangrijk deel semantiek (namen van kolommen, sleutels, enz.) en zijn dan ook uitermate geschikt om te exporteren naar RDF. Men heeft stand-alone convertoren zoals D2R Server<sup>1</sup> of men kan plugins installeren voor bv. Protégé<sup>2</sup>.

---

<sup>1</sup> <http://www4.wiwiss.fu-berlin.de/bizer/d2r-server/>

<sup>2</sup> Protégé is een open source ontologie-editor, welke in de case study gebruikt wordt.

Plugins te vinden op <http://protege.cim3.net/cgi-bin/wiki.pl?ProtegePluginsLibraryByType#nid3TE>

Andere convertoren spitsten zich toe om bestaande HTML, XHTML, XML of XMLSchema om te zetten naar RDF. Zo is GRDDL<sup>3</sup> een standaard (09/2007) ontwikkeld door het W3C om bestaande XML of XHTML om te zetten naar RDF.

#### 2.7.4.4 Reification en context

Een ander krachtig principe bij het gebruik van RDF is reification. Dit wil zeggen dat een uitdrukking, dus de triplet bron-attribuut-waarde, zelf een URI krijgt en op die manier dan een nieuwe bron kan worden in een uitdrukking. Zo kan men zaken modelleren als “*Persoon X zegt dat Persoon Y iets denkt*”. Met deze werkwijze kan men ook de context van een uitdrukking aangeven. Een bepaalde context krijgt een URI en elke uitdrukking kan hiermee geassocieerd worden met een “isTrueIn”-attribuut.

#### 2.7.5 SPARQL

*"SPARQL will make a huge difference"*

(Berners-Lee, 2006)

SPARQL (Simple Protocol and RDF Query Language, spreek uit “sparkle”) is voor het Semantisch Web wat SQL is voor relationele databases. De querytaal maakt het mogelijk om informatie te halen uit heterogene bronnen over het ganse Web. Op 15 januari 2008 werd SPARQL aanvaard als standaard.

SPARQL wordt gebruikt om queries te formuleren over verschillende databronnen heen die zijn opgeslagen als RDF of via middleware aangesproken kunnen worden in RDF. Dankzij de datastructuur van RDF zijn de mogelijkheden van SPARQL uitgebreider dan die van SQL.

---

<sup>3</sup> <http://www.w3.org/TR/grddl/>

## 2.7.6 RDF Schema (RDFS)

### 2.7.6.1 *Bespreking*

Velen zijn het erover eens dat de naam RDFS ongelukkig gekozen is. Het is namelijk *niet* zo dat RDFS zich verhoudt tot RDF zoals XMLSchema (cf. §2.7.3.2) zich verhoudt tot XML. Waar XMLS de structuur van een XML-document vastlegt, bepaalt RDFS de woordenschat die gebruikt wordt in RDF datamodellen. RDFS geeft informatie over de interpretatie van RDF-uitdrukkingen.

RDFS maakt het mogelijk

- een rudimentaire woordenschat te definiëren
- te bepalen welke eigenschappen toegepast kunnen worden op welke objecten en welke waarden deze kunnen aannemen
- relaties tussen objecten te beschrijven

De syntaxis die gebruikt wordt om deze woordenschat te kunnen definiëren is RDF zelf. Met andere woorden, RDFS is geschreven in RDF. RDFS kreeg de status van W3C recommendation in februari 2004.

### 2.7.6.2 *Klassen en eigenschappen*

Het gebruik van RDF laat toe over welbepaalde objecten te praten, maar soms wil men het over het meer algemene, abstracte concept hebben. Zoals voor de meeste programmeertalen is er in het Semantisch Web dan ook een manier voorzien om klassen (classes) te declareren. Een bepaalde resource kan een klasse vormen voor andere resources. Deze zijn dan instanties van de klasse.

De drie belangrijkste klassen zijn Resource (rdfs:Resource), Class (rdfs:Class) en Property (rdf:Property). Verder zijn er ook nog Literal (rdfs:Literal) en Statement (rdf:Statement). Rdfs:Class is zelf een Resource en rdfs:Resource is zelf een Class, wat een recursieve definitie oplevert.

Klassen laten toe om het domein (domain) en het werkgebied (range) van een Resource te bepalen.

```

<rdf:RDF xmlns:pl="http://www.philippeluickx.be/">

  <pl:isAuthorOf>

    <pl:philippeluickx>

    <pl:eindverhandelingSemWeb>

  isAuthorOf rdfs:domain Person

  isAuthorOf rdfs:range WrittenThings

```

Het is in dit voorbeeld duidelijk dat slechts *personen* het object kunnen zijn van de eigenschap “isAuthorOf”. In RDFS kan men bijgevolg bepalen dat het domein van “isAuthorOf” enkel bepaald wordt door “personen”. Men kan ook bepalen dat het bereik van “isAuthorOf” enkel kan bestaan uit publicaties, artikels, boeken, blogs enz., of een meer algemene klasse “geschreven dingen” die een superklasse is van de vorige klassen.

Samengevat:

- rdfs:domain van een rdf:property bepaalt de klasse van een *subject* in een triplet door deze eigenschap te gebruiken als predikaat
- rdfs:range van een rdf:property bepaalt de klasse of het datatype van een *object* in een triplet door deze eigenschap te gebruiken als predikaat

### 2.7.6.3 Class hierarchies en overerving

Met rdfs:subClassOf kan men in RDFS een klassenhiërarchie definiëren. Deze hiërarchie hoeft niet strikt te zijn, een klasse kan meerdere superklassen hebben.

Het belang van subklassen wordt duidelijk als men de overerving van semantiek beschouwt. Subklassen erven de eigenschappen over van de superklasse, wat data van een belangrijk deel semantiek voorziet. Deze eigenschappen bevatten ook het gedefinieerde domein en bereik van de superklassen.

Applicaties hoeven dus niet meer zelf conclusies te trekken in verband met overerving. Wanneer een bepaalde klasse semantiek overerft van zijn superklasse, is deze semantiek intrinsiek aanwezig in de data en geldt ze dus voor elke applicatie.

## 2.7.7 Web Ontology Language (OWL)

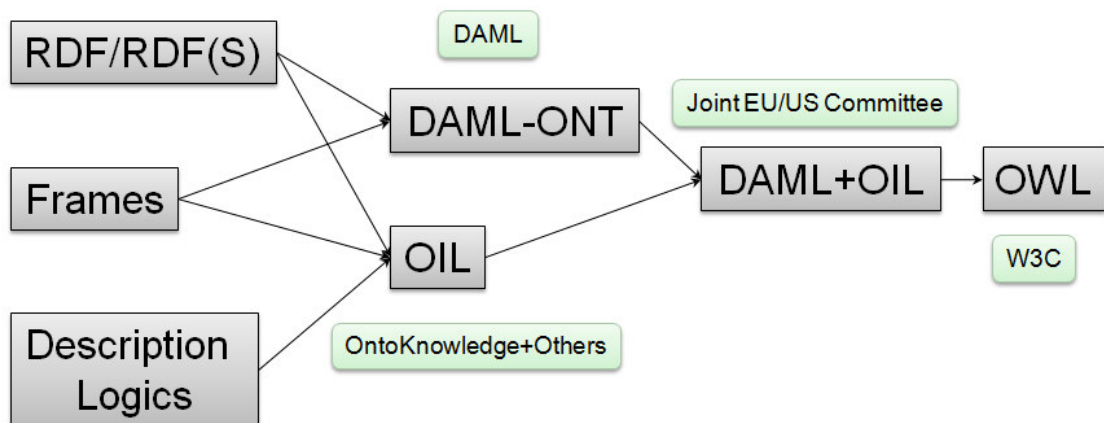
### 2.7.7.1 Bespreking

Hoewel RDF en RDFS al een basis leggen voor een ontologie, blijven de mogelijkheden beperkt. Met RDF kan men enkel *binair* relaties leggen en RDFS beperkt zich tot subklassen hiërarchie met domein- en bereikbeperkingen. Er zijn verschillende gevallen waarin een taal nodig is die meer expressiviteit biedt:

- *lokale scope van eigenschappen*: rdfs:range definieert het bereik van een eigenschap voor *alle* klassen, terwijl het kan voorkomen dat dit bereik enkel voor bepaalde klassen voorkomt en niet voor andere
- *disjointness van klassen*
- *booleaanse combinaties van klassen*: unie, doorsnede, complement
- *beperkingen van kardinaliteit*
- *speciale eigenschappen*: transitief (transitive), uniek (unique), invers (inverse)

Het W3C richtte daarom de Web Ontology Working Group op om een taal te ontwikkelen die meer biedt dan RDF en RDFS. Het uiteindelijke resultaat werd OWL, wat zelf een verderzetting was van DAML+OIL.

### 2.7.7.2 Geschiedenis



**Figuur 11** Geschiedenis van OWL



De ontwikkeling van ontologietalen is al een tijd aan de gang in de computerwetenschappen. Eén van de eerste talen die het mogelijk maakte om ontologieën te ontwerpen voor het web, was **SHOE** (Simple HTML Ontology Extensions), gebaseerd op HTML. SHOE werd echter eerder ontwikkeld als een proefproject, om de toekomstige mogelijkheden van ontologieën aan te tonen.

Het **OIL** (Ontology Inference Layer) project werd ontwikkeld door voornamelijk Europese onderzoekers en was de eerste grote realisatie van het On-To-Knowledge-project. De taal biedt een raamwerk voor het ontwerpen van ontologieën en combineert de framegebaseerde filosofie met Description Logics. OIL houdt sterk vast aan formele semantiek en de syntaxis is gebaseerd op RDFS (op zijn beurt gebaseerd op RDF en (meestal) XML). OIL komt in vier versies: Core, Standard, Instance en Heavy. Elke opeenvolgende versie voegt functionaliteit en complexiteit toe.

**DAML** (DARPA Agent Markup Language) werd in dezelfde periode ontwikkeld als OIL, maar ging uit van het departement van defensie van de Verenigde Staten. Het doel van deze taal was RDF uit te breiden met meer expressiviteit en de interactie tussen software-agenten te verbeteren en te vereenvoudigen. DAML bestaat uit twee talen: DAML-ONT is de ontologietaal, DAML-L is een tweede taal voor redeneringen en logische gevolgtrekkingen.

DAML-ONT baseert zich grotendeels ook op frame-gebaseerde talen, maar ook op concepten uit de OO-wereld (Object Oriented). Het grote minpunt van deze taal is dat de semantiek nogal zwak is, waardoor problemen ontstaan bij de interpretatie door machines en mensen.

Aangezien beide talen (DAML-ONT en OIL) dezelfde doelen nastreefden, maar te kampen hadden met specifieke tekortkomingen, werd er besloten de twee talen samen te voegen tot **DAML+OIL**. De ontwikkeling werd verzorgd door een EU/US comité voor Agent Markup Languages. Hoewel DAML+OIL wel aanvaard en gebruikt werd door de gemeenschap, is de taal nooit doorgebroken. Ze blijft echter wel van belang, aangezien ze de basis legt voor OWL.

In november 2001 werd de *Web Ontology Working Group* opgericht, die **OWL** (Web Ontology Language) ontwikkelden. OWL is gebaseerd op DAML+OIL en werd aanvaard als standaard in februari 2004. Recentelijk is de Web Ontology Working Group heropgestart om te werken aan OWL1.1.

### 2.7.7.3 OWL

OWL is een taal om gestructureerde en webgebaseerde ontologieën te definiëren. Ze voegt mogelijkheden toe om eigenschappen en klassen te beschrijven en bouwt verder op de syntaxis van RDF en RDFS.

#### 2.7.7.3.1 Syntaxis

Klassen worden in OWL gedefinieerd door *owl:Class*. Over deze klasse kan men de semantiek uitbreiden met disjunctie (*owl:disjointWith*) en equivalentie (*owl:equivalentClass*) tegenover andere klassen.

Eigenschappen komen in twee vormen voor: objecteigenschappen en datatype-eigenschappen. Dit betekent dat een klasse gelinkt kan worden via een eigenschap aan ofwel een andere klasse, ofwel een waarde van een bepaalde datatype.

Op deze eigenschappen kunnen beperkingen geplaatst worden. Via *owl:onProperty* kan men de mogelijke waarden van een eigenschap bepalen. *Owl:hasValue* bepaalt de enige correcte waarde voor de relatie die gelegd wordt door de eigenschap, met *owl:allValuesFrom* kan er gekozen worden tussen de mogelijke waarden (universele kwantificatie) en *owl:someValuesFrom* bepaalt dat er minstens één relatie moet zijn die de mogelijke waarde aanneemt (existentiële kwantificatie). Met OWL is het mogelijk kardinaliteiten te bepalen door *owl:minCardinality* en *owl:maxCardinality*.

Ook andere speciale eigenschappen zijn gedefinieerd: transitief (*owl:transitiveProperty*), symmetrisch (*owl:symmetricProperty*), functioneel (*owl:functionalProperty*) en invers functioneel (*owl:inverseFunctionalProperty*).

De booleaanse combinaties complement (*owl:complementOf*), unie (*owl:unionOf*) en intersectie (*owl:intersectionOf*) zijn in OWL ook bepaald. De laatste mogelijkheid die OWL aanbiedt is opsomming (*owl:oneOf*).

### 2.7.7.3.2 Subtalen

Er bestaat altijd een trade-off tussen de expressieve kracht van een ontologie en de ondersteuning voor reasoning: hoe rijker de taal is, hoe minder efficiënt de reasoning zal verlopen. Hoe minder efficiënt de reasoning kan gebeuren, hoe minder schaalbaar de ontologietaal zal zijn. Daarom hebben ontwikkelaars de keuze tussen drie subtalen van OWL, namelijk OWL-Lite, OWL-DL (Description Logic) en OWL-Full.

**OWL-Full** maakt gebruik van alle OWL taalprimitieven, waardoor er een grote expressieve kracht beschikbaar is. Zo kan men de betekenis van voorgedefinieerde primitieven aanpassen: als er een kardinaliteit van *owl:Class* bepaald wordt, kan men het aantal klassen in een ontologie beperken. Deze kracht zorgt er echter voor dat er geen complete en/of efficiënte reasoning mogelijk is. Hierdoor zal OWL-Full slechts in uitzonderlijke gevallen gebruikt worden, waar het belangrijker is om de werkelijkheid zo correct mogelijk te modelleren.

Met **OWL-DL** is het wel mogelijk om op een efficiënte manier aan reasoning te doen. Deze subtaal verliest een deel van zijn expressiviteit doordat er beperkingen zijn op hoe OWL en RDF constructoren mogen gebruikt worden. Zo mogen OWL constructoren niet op elkaar toegepast worden. Een ander nadeel is ook dat het de volledige compatibiliteit met RDF verliest, aangezien RDF deze beperkingen niet kent.

Wanneer ontwikkelaars een gemakkelijk te implementeren ontologie willen modelleren, kunnen ze gebruik maken van **OWL-Lite**. Deze ontologietaal beschikt slechts over een deel van de taalconstructoren en bezit slechts een zeer beperkte expressiviteit.

### 2.7.8 Rules

Om axioma's, of regels (rules), op een formele wijze weer te geven, heeft men nood aan een regel-taal. Veel kennis kan weergegeven worden met regels, zoals procedurele kennis en logische structuren. Waar een ontologie de mogelijkheid geeft klassen te beschrijven met eigenschappen en relaties, kan een regel-taal de logische beweringen en gevolgtrekkingen beschrijven. Zo kan in OWL een relatie als “nonkel” (de broer van één van de ouders) niet gedefinieerd worden.

Het W3C heeft een Working Group opgericht die de ontwikkeling van het **Rule Interchange Format** (RIF) beheert, maar ook andere standaarden zijn al voorgesteld om als regel-taal te gelden binnen het Semantisch Web. De **Semantic Web Rule Language** (SWRL) is in mei 2004 ingediend bij het W3C door het National Research Council of Canada, Network Inference en Stanford University.

### 2.7.9 Logic

De logische laag zal een unificerende taal voorzien om een set afleidingen te kunnen beschrijven die gemaakt kunnen worden over een verzameling gegevens. Deze laag maakt het mogelijk weer een stap verder te zetten in het afleiden van nieuwe informatie en connecties.

### 2.7.10 Proof

Vanaf het ogenblik dat systemen gebouwd worden die logica volgen, is het zinvol deze systemen te gebruiken om beweringen te bewijzen. De bewijstaal zal toelaten de stappen te beschrijven die genomen werden om tot een conclusie te komen. Deze bewijzen kunnen dan op hun beurt doorgegeven en nagekeken worden, waardoor het systeem niet meer elke keer dezelfde deducties moet maken.

In 1997 kwam Sir Tim Berners-Lee af met het concept van de *oh yeah?-button*. Dit illustreert het hele concept van bewijs in het Semantisch Web: op elk moment moet een gebruiker in staat zijn om op de knop drukken om te zien waar bepaalde informatie vandaan komt.

### 2.7.11 Trust

Al het voorgaande lijkt schitterend, maar is eigenlijk vrij nutteloos als *iedereen* iets kan zeggen over *alles*... Dergelijk systeem is niet te vertrouwen en het zal voorkomen dat informatiebronnen op het Semantisch Web elkaar zullen tegenspreken.

Hier komt het belang van digitale handtekeningen naar boven. Deze handtekeningen geven het bewijs dat een bepaalde persoon (of website) instaat voor bepaalde informatie. Wanneer gebruikers aan hun systeem vervolgens vertellen welke personen te vertrouwen zijn, kan deze geloofwaardige informatie beginnen te verzamelen.

Het is echter onwaarschijnlijk dat deze informatie voldoende zal zijn. Daarom zullen *webs of trust* ontstaan: de personen die iemand vertrouwt, vertrouwen op hun beurt ook een reeks personen, die ook een aantal personen vertrouwen, enz. Deze vertrouwensrelaties zullen elk een bepaalde graad van vertrouwen (of wantrouwen) krijgen, die zal dalen naargelang de afstand in het vertrouwensweb groter wordt.

Op deze manier kan op kleinere schaal gewerkt worden, in een bepaalde context. Zo kan iemand een hoop RDF-gegevens van vrienden ter beschikking hebben die films van het voorbije jaar bespreken. Zelf gaat die persoon enkel belang hechten aan de gegevens die afkomstig zijn van vrienden die dezelfde smaak hebben. Er ontstaat een context voor informatie.

## Hoofdstuk 3 Enterprise Architecture

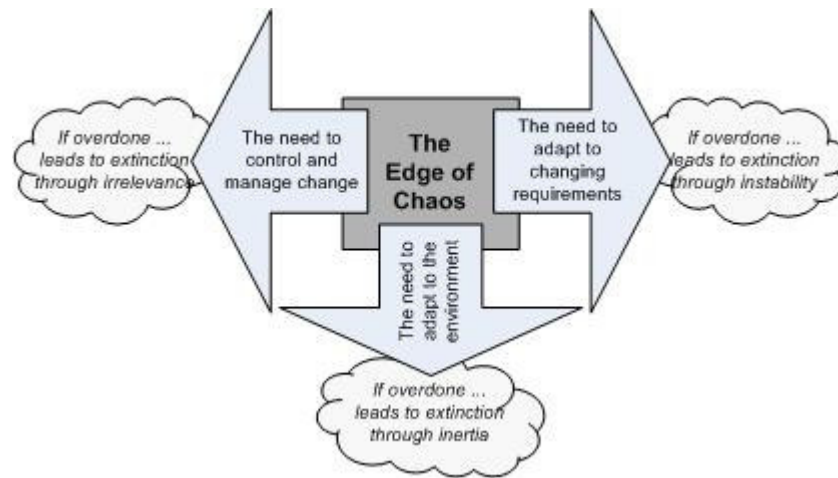
Het concept van Enterprise Architecture wordt in dit hoofdstuk besproken. Het eerste deel behandelt het Enterprise Architecture zelf, terwijl het tweede deel enkele Enterprise Architecture Frameworks bespreekt, namelijk Zachman, TOGAF, FEAF, MoDAF en DoDAF.

### 3.1 Bespreking EA

Een Enterprise Architecture (EA) is de beschrijving van de huidige en/of toekomstige structuur van een organisatie en het gedrag van de processen, informatiesystemen, personeel en organisatorische subeenheden binnen een organisatie zodat deze overeenkomen met de doelen en de strategische richting van de organisatie. EA definieert de missie van een bedrijf en de informatie en technologie die nodig zijn om deze missie te volbrengen. Om dit te beschrijven, omvat een goede EA een basislijn architectuur (*baseline architecture*), een doel architectuur (*target architecture*) en een sequentieplan (*sequencing plan*).

Het gebruik van een EA ondersteunt de beslissingen die genomen worden in een bedrijf door inzicht te geven in de structuur van complexe organisaties, hoe ze werken en welke technologie ze gebruiken. De architectuur helpt om investeringen beter te aligneren met de missie van de organisatie en zorgt ervoor dat deze investeringen niet redundant zijn. Door details van een onderneming te abstraheren, wordt het mogelijk om een overzicht te krijgen van de organisatie. Het wordt ook mogelijk met EA om de organisatie vanuit verschillende standpunten te onderzoeken.

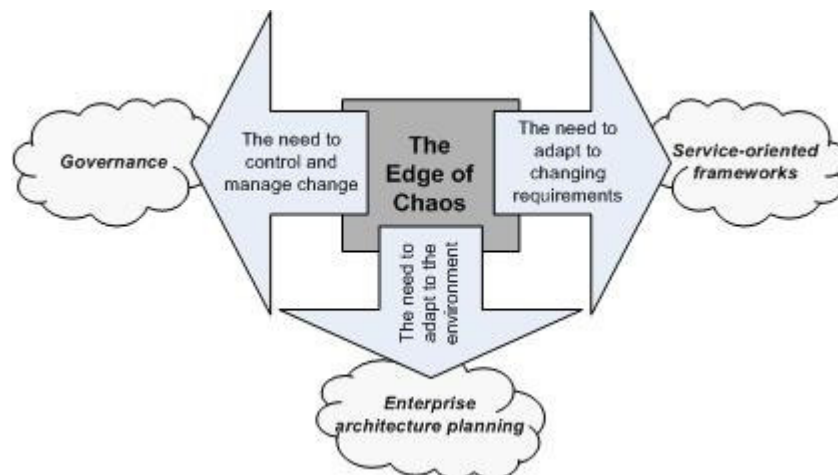
De nood aan EA is gebaseerd op de theorie dat een onderneming onderhevig is aan verschillende krachten (Balchin N. 2007).



**Figuur 12 Krachten onderneming (bron: Balchin N.)**

Een onderneming moet een subtiel evenwicht weten te vinden tussen stabiliteit om alles leefbaar te houden en flexibiliteit om zich te kunnen aanpassen aan veranderende omstandigheden.

Om deze balans te vinden zijn er een aantal hulpmiddelen ontwikkeld die in de huidige maatschappij hun opmars vinden.



**Figuur 13 Oplossingen balans (bron: Balchin N.)**

Een EA planning is dus nodig om te kunnen overleven langs deze scheidingslijn van chaos.

### 3.2 Opbouw

Een EA beschrijft hoe de elementen van een organisatie samenhangen.



**Figuur 14 Elementen Enterprise Architecture (bron: MITRE 2004)**

Een goede architectuur geeft twee modellen: een *as-is* model en een *to-be* model. Het eerste model toont de relaties van eenheden binnen een bedrijf, het “grote plaatje”. Het tweede model wordt voornamelijk gebruikt om de impact van een verandering te analyseren.

Tussen deze twee modellen hoort een sequentieplan om de stappen te beschrijven van hoe een geplande verandering moet verlopen.

Externe en interne omgevingsfactoren veranderen continu en vergen van de organisatie dat zij snel en efficiënt actie te onderneemt. Deze veranderingen hebben een effect op alle aspecten van een organisatie: het **bedrijf**, de **applicaties**, de **informatie** en de **technologie**. Waar mogelijk moeten deze vier aspecten expliciet gerelateerd worden met de doelen en de strategie van de organisatie. Ze zijn dan ook altijd terug te vinden in de Enterprise Architecture Frameworks (EAF).



### **3.3 Enterprise Architecture Frameworks**

#### **3.3.1 Bespreking EAF**

Gezien de omvang van de informatie behandeld door een EA, is het zinvol om over een formele manier te beschikken om deze informatie te klasseren, namelijk een raamwerk. Een Enterprise Architecture Framework (EAF) definieert hoe de structuur en subschema's (views) van een EA georganiseerd moeten worden. Een EAF identificeert de informatie nodig om een EA weer te geven, organiseert deze informatie in een logische structuur en beschrijft de relaties tussen deze informatietypes.

Vaak worden modellen gebruikt om de nodige informatie weer te geven. De Office of Management and Budget (OMB, Verenigde Staten) ontwierp voor hun Federal Enterprise Architecture (FEA) vijf generieke modellen: Business Reference Model (BRM), Performance Reference Model (PRM), Service Component Reference Model (SRM), Data Reference Model (DRM) en Technical Reference Model (TRM).

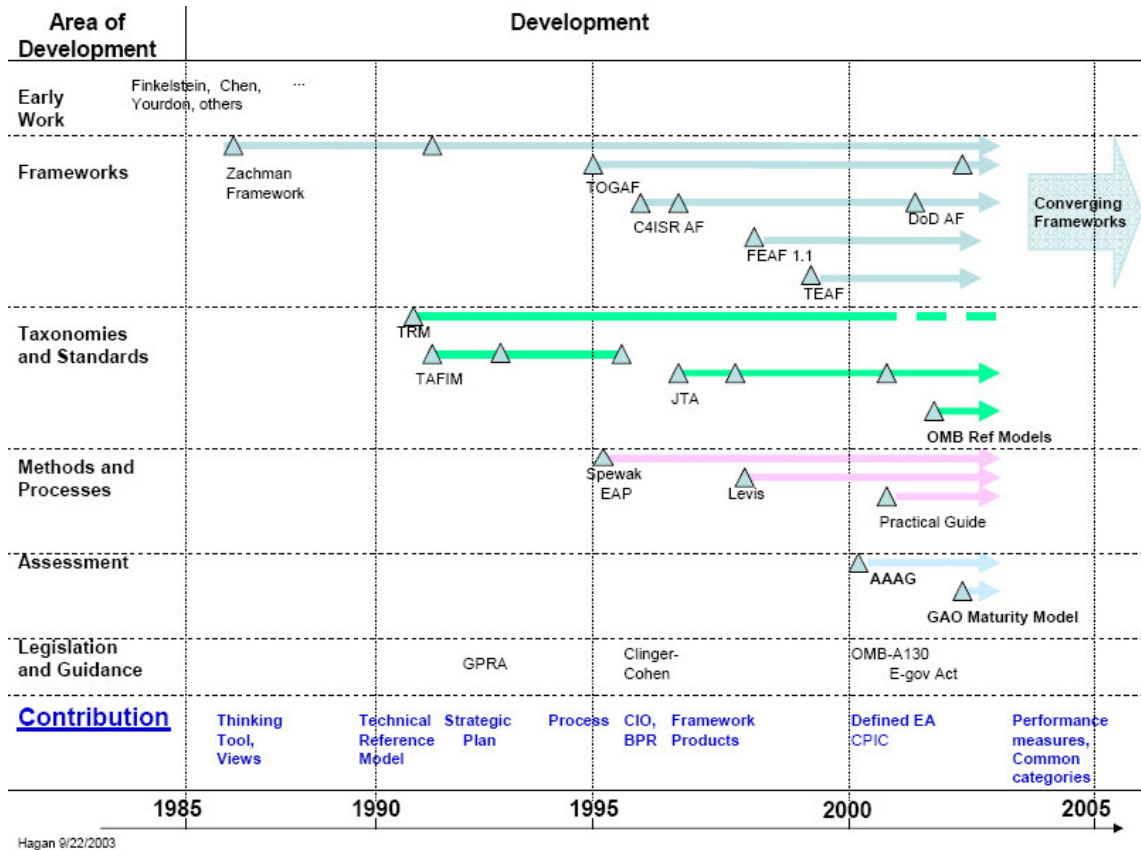
Dankzij een EAF kan de context van een verandering onderzocht worden. Het beeld van *een steentje in het water* kan hierbij helpen: een verandering in één aspect van de organisatie, heeft vaak ook effecten op andere aspecten binnen de organisatie. Als bv. besloten wordt om de toegang tot informatie via het WWW te verbeteren (bedrijfsaspect), zal dit een effect hebben op de andere aspecten van de organisatie (data, applicaties en technologie).

In hoofdstuk 4 wordt onderzocht wat de impact zal zijn van het implementeren van Semantisch Web technieken voor het beheer van beleidsinformatie binnen de Vlaamse Overheid. Wat zijn de effecten op het bedrijf, op de informatie, op de applicaties, op het informatiesysteem?

### 3.3.2 Raamwerken

Hierna worden enkele populaire raamwerken besproken. Dit is geen volledige opsomming en slechts een beperkte bespreking.

Onderstaande figuur schetst de evolutie van de belangrijkste raamwerken.



**Figuur 15** Belangrijkste ontwikkelingen in de geschiedenis van EA (bron: MITRE 2004)

#### 3.3.2.1 Zachman raamwerk

In de jaren '80 ontwikkelde John Zachman het **Zachman Raamwerk** bij IBM. In een ophefmakend artikel enkele jaren later besprak Zachman zijn observatie dat in de vliegtuigbouw blauwdrukken een immens belangrijke rol speelden. Vooraleer men een vliegtuig bouwt, wordt de architectuur formeel beschreven en vaak zelfs getest voordat een prototype gebouwd wordt. Hij trok dit door naar de bouw, waar een constructie ook eerst ontworpen wordt op papier of computer vooraleer het gebouwd wordt. Hij vroeg zich vervolgens af hoe het mogelijk was dat zoiets complex als een bedrijf geen enkele architecturale blauwdruk had.

Een andere observatie die hij maakte, was dat er niet één architecturale representatie bestond van een complex product maar verschillende, elk vanuit een eigen standpunt. Zo is een plan voor de ruwbouw van een huis anders dan het plan voor de elektriciteit of het plan voor het sanitair. In het ontwerpproces zijn bijgevolg verschillende perspectieven of rollen van belang.

In Bijlage 2 is het Zachman raamwerk opgenomen.

Het raamwerk hanteert twee dimensies.

- Elke rij bepaalt een specifiek **perspectief** van een belangenpartij: Planner (Planner), Eigenaar (Owner), Ontwerper (Designer), Bouwer (Builder), Onderaannemer (Subcontractor) en Functionerende Organisatie (Functioning Enterprise).
- De kolommen stellen de hoofdvragen voor, door Zachman beschouwd als **abstracties**: Wat, Hoe, Waar, Wie, Wanneer en Waarom.

De matrix die zo ontstaat, biedt een holistische kijk op de EA die gemodelleerd wordt.

Het raamwerk bevat enkele regels. Zo is de volgorde van de kolommen of rijen van geen belang, aangezien ze elk evenwaardig zijn. Elke kolom heeft een simpel basismodel, dat voor elke kolom uniek is. Aangezien elke rij een verschillend perspectief biedt, is elke cel in de matrix dan ook uniek.

Het voordeel van dit raamwerk is de intuïtieve en complete wijze waarop concepten van de verschillende belangpartijen worden behandeld. Daar staat tegenover dat deze benadering veel documentatie vereist (er zijn dertig verschillende cellen), welke niet allemaal even relevant zijn.

### 3.3.2.2 TOGAF

**The Open Group Architecture Framework (TOGAF)** is een raamwerk dat ontwikkeld en onderhouden wordt door The Open Group. Het originele raamwerk ontstond midden jaren '90 en ondertussen zit het raamwerk aan versie 8.1.1.

TOGAF modelleert vier domeinen: Zaken (Business), Toepassing (Application), Gegevens (Data) en Technologie (Technology).

- De *Zaken* architectuur bepaalt de bedrijfsstrategie, het bestuur, de organisatie en de belangrijkste bedrijfsprocessen.
- De *Toepassing* architectuur beschrijft de individuele applicatiesystemen die gebruikt worden, de interacties tussen deze systemen en de relaties met de bedrijfsprocessen van de organisatie.
- De *Gegevens* architectuur modelleert de structuur van de logische en fysieke gegevensbronnen van een organisatie en hun beheermiddelen.
- De *Technologie* architectuur specificeert de infrastructuur van de software die de kernactiviteiten te ondersteunen.

Het raamwerk komt samen met Architecture Development Method (ADM), welke beschrijft hoe een EA ontwikkeld kan worden. ADM kan aangepast worden aan de organisatie om vervolgens gebruikt te worden om het ontwerp van de EA te ondersteunen. Verder is er nog een Technical Reference Model dat een taxonomie van generische diensten aanbiedt en een Standards Information Base, wat een database is van open industriestandaarden.

### 3.3.2.3 FEAR

De Clinger Cohen Act (1996) vereiste in de USA dat agentschappen een IT architectuur ontwierpen die ervoor zorgde dat de evolutie en het onderhoud van IT en de aanwerving van nieuwe IT in lijn waren met de strategische doelen van de agentschappen.

In 1999 werd daarom de **Federal Enterprise Architecture Framework** (FEAF) gepubliceerd. In de publicatie wordt het raamwerk zelf besproken en ook hoe men het raamwerk ontwikkelt en onderhoudt.

FEAF bevordert de gedeelde ontwikkeling van federale processen, interoperabiliteit en informatiedeling langsheen de verschillende agentschappen.

FEAF omvat acht componenten:

- *Architecturale driver*: bedrijfs- en ontwerpdrivers
- *Strategische richting*: bepaalt de ontwikkeling van de doelarchitectuur van de verschillende agentschappen en omvat de visie, principes, doelen en objectieven van deze agentschappen
- *Huidige architectuur*: beschrijving van de huidige organisatie
- *Doel architectuur*: beschrijving van naar welke toestand de organisatie zou moeten evolueren
- *Overgangsproces*: beschrijving van hoe deze overgang moet verlopen
- *Architecturale segmenten*: focus op specifieke bedrijfsomgevingen
- *Architecturale modellen*: bedrijfs-, data-, toepassingen- en technologie-architecturen
- *Standaarden*

De FEAF was belangrijk om de link te leggen met strategische plannen en doelen en de nood aan een overgang. De zwakheid van het raamwerk is het gebrek aan meer specifieke beschrijvingen, welke nodig waren omdat de ontwikkeling van een EA voor de meeste agentschappen volkomen nieuw was.

#### 3.3.2.4 DoDAF

Het **Department of Defense Architecture Framework** (DoDAF) werd ontwikkeld om de EA te ontwerpen voor het Department of Defense van de USA. Het raamwerk legt de nadruk voornamelijk op defensie-toepassingen, maar is ook toepasbaar op commerciële systemen.

DoDAF is gebaseerd op het C4ISR (Command, Control, Communications, Computers, Intelligence, Surveillance and Reconnaissance) Architecture Framework.

DoDAF wordt ingedeeld in vier basis subschema's (views):

- *Operationeel subschema* (Operational View): omschrijving van de taken en activiteiten, operationele elementen en informatie-uitwisselingen die nodig zijn om de doelen van het Department of Defense uit te voeren
- *Systeem subschema* (Systems View): beschrijving van de systemen en hun interconnecties
- *Technische standaarden subschema* (Technical Standards View): minimale set van regels die de regeling, interactie en onderlinge afhankelijkheid van delen van het systeem bepalen
- *Alles subschema* (All view): omvat producten die op de drie andere subschema's van toepassing zijn, zoals Overzicht en Samenvating

#### 3.3.2.5 MoDAF

Het Ministry of Defense van het Verenigd Koninkrijk ontwikkelde op haar beurt de **Ministry of Defense Architecture Framework** (MoDAF), welke de EA organiseert van defensie-toepassingen. Het is voor een groot deel afgeleid van DoDAF, waardoor het veel gelijkenissen heeft.

MoDAF is ingedeeld in zes gezichtspunten:

- *Standaard gezichtspunt* (Standard Viewpoint)
- *Operationeel gezichtspunt* (Operational Viewpoint)
- *Systeem gezichtspunt* (System Viewpoint)
- *Technische standaarden gezichtspunt* (Technical Standards Viewpoint)
- *Verwerving gezichtspunt* (Acquisition Viewpoint)
- *Alles gezichtspunt* (All Viewpoint)

Er zijn twee gezichtspunten toegevoegd tegenover DoDAF: het Standaard en Verwerving gezichtspunt. UML Profile for DoDAF/MODAF (UPDM) is een standaard die nog in ontwikkeling is en die beide raamwerken samenvoegt.

### **3.4 Enterprise Architecture en ontologieën**

Aangezien een ontologie een concept op een formele wijze weergeeft, kan het zich ook lenen voor de formele beschrijving van een organisatie. Het is dus mogelijk deze organisatie te gaan modelleren met een ontologie. Het consultancybedrijf Topquadrant heeft een ontologie ontworpen die de aspecten van het FEA weergeeft (cf. §4.2.2.1.3).

## Hoofdstuk 4 Het gebruik van Semantisch Web technieken voor het beheer van beleidsinformatie binnen de Vlaamse Overheid

In dit hoofdstuk komt de case study voor de Vlaamse Overheid aan bod. Een eerste deel bespreekt de voorbereiding van het project, waarna een tweede deel de ontwikkeling van een ontologie voor het beheer van beleidsinformatie uiteen zet. Het hoofdstuk wordt afgesloten met enkele aanbevelingen naar de Vlaamse Overheid toe.



### 4.1 Voorbereiding project

In de voorbereidende fase van het project wordt eerst de organisatie van de Vlaamse Overheid uit de doeken gedaan, waarna dieper wordt ingegaan op de term “beleidsinformatie”. Een derde deel bespreekt een beperkt aantal ontologie-editors, waarna de voorbereiding wordt afgesloten met een analyse van het reeds bestaande model van gegevensbronnen binnen de Vlaamse Overheid.

#### 4.1.1 Inzicht in de Vlaamse Overheid

De Vlaamse Overheid in de context van deze eindverhandeling is de verzamelnaam voor de overheidsadministraties van de Vlaamse regering. Na de hervorming van het *Beter Bestuurlijk Beleid* (BBB) op 1 januari 2006 bestaat de Vlaamse Overheid uit 13 homogene beleidsdomeinen, elk met één departement en een aantal agentschappen. De departementen verzorgen voornamelijk de beleidsvoorbereiding, terwijl de agentschappen instaan voor de beleidsuitvoering.

Elk departement en agentschap zijn op hun beurt opgebouwd uit afdelingen en cellen. De agentschappen bestaan in twee vormen. Wanneer een agentschap rechtstreeks aangestuurd wordt door de bevoegde minister, is dit een Intern Verzelfstandigd Agentschap (IVA). Indien er nog een raad van bestuur aanwezig is tussen de bevoegde minister en het agentschap, spreekt men van een Extern Verzelfstandigd Agentschap (EVA).



Drie van de beleidsdomeinen zijn horizontaal, ze hebben betrekking op de hele Vlaamse Overheid. De overige tien zijn verticale beleidsdomeinen, elk met hun eigen specifieke bevoegdheden.

Een aantal Vlaamse Openbare Instellingen werden niet opgenomen in de hervorming, maar behoren wel nog tot de Vlaamse administratie (bv. de VRT).

Zie ook Bijlage 3 voor een verdere uitwerking van de beleidsdomeinen, departementen en agentschappen.

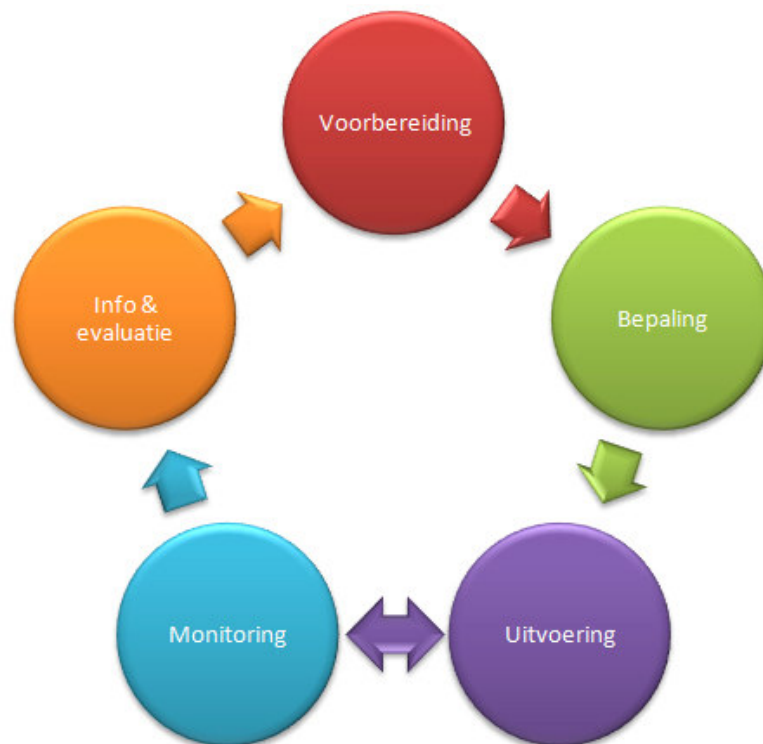
#### 4.1.2 Beleidsinformatie

Om over het beheer van beleidsinformatie te kunnen praten, dient men eerst te kijken naar wat deze informatie juist inhoudt.

Zoals het woord zelf zegt, gaat beleidsinformatie over informatie om een beleid te voeren. Dit is informatie die intern blijft. Wat de burger te zien krijgt (in wetteksten, diensten, enz.) is het resultaat van het verwerken van deze beleidsinformatie.

Beleidsinformatie beperkt zich meestal tot een bepaald beleidsdomein.

Het aanmaken en gebruiken van beleidsinformatie kent een dubbel cyclisch verloop.



**Figuur 16** Beleidscyclus

In de eerste plaats beslaat een beleidsperiode vijf jaar. Na deze vijf jaar zijn er opnieuw verkiezingen voor het Vlaams Parlement en worden nieuwe ministers benoemd. Deze eerste cyclus van vijf jaar kan onderverdeeld worden in vijf fasen.

In de eerste fase wordt het beleid **voorbereid**. Studies en analyses worden uitgevoerd, zoals een socio-economische analyse. In deze fase wordt een grote hoeveelheid informatie verzameld.

De **bepaling** van het beleid komt aan bod in de tweede fase. Hier worden aan de hand van de informatie uit de eerste fase een aantal doelen opgesteld. De resultaten zijn een regeerakkoord en ministeriële beleidsnota's.

De derde fase bekommert zich om de **uitvoering** van de doelen gesteld in de vorige fase. Er is ook een wisselwerking met een volgende (quasi-gelijktijdige) fase, de **monitoring**. Hierbij gaat men bepaalde onderdelen grondiger evalueren, zoals het ziekteverzuim van het personeel.

De vijfde en laatste fase verzorgt de **evaluatie** van de uitvoering door de resultaten van de monitoring te toetsen aan de doelstellingen van de tweede fase.

Deze cyclus van vijf jaar bestaat zelf ook nog eens uit vijf cycli van één jaar. De voorbereidende fase verzorgt nieuwe onderzoeken (want op vijf jaar tijd kan er veel veranderen), waarna er doelstellingen met een tijdshorizon van één jaar worden opgesteld. De uitvoering van het beleid van dat jaar wordt gemonitord en aan het einde van het jaar geëvalueerd. De cyclus van het volgende jaar bouwt dan voort op de vorige cyclus.

Ministers en medewerkers moeten hun visie kunnen baseren op correcte en tijdige informatie. Een goed beheer ervan is dan ook zeer belangrijk binnen de Vlaamse Overheid.

### 4.1.3 Tools

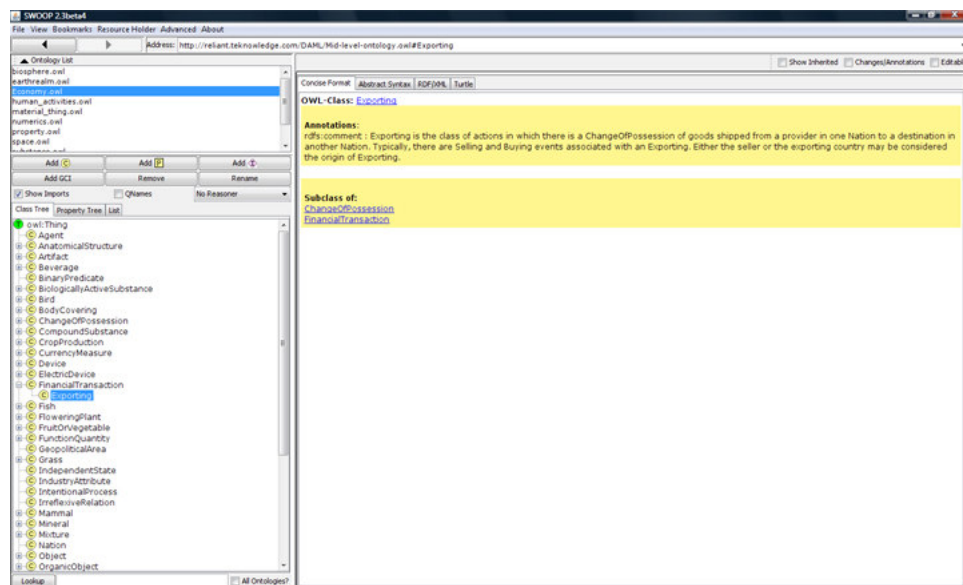
In dit hoofdstuk worden enkele ontologie-editors besproken. Eén van deze applicaties zal gebruikt worden om een nieuwe ontologie te ontwikkelen die de gegevensbronnen binnen de Vlaamse Overheid beschrijft. Deze nieuwe ontologie zal een hulpmiddel worden om een beter zicht te krijgen op de aanwezige gegevensbronnen en zal ook een betere ontsluiting van informatie bevorderen.

Aangezien het oorspronkelijke model (cf. §4.2) in Protégé is opgesteld en aangezien dit een degelijke tool is, werd besloten om hiermee verder te werken. Desalniettemin loont het om eerst naar de alternatieven te kijken. De volgende vier softwarepakketten zijn een persoonlijke selectie.

#### 4.1.3.1 SWOOP

**SWOOP** is een tool ontwikkeld in Java voor het aanmaken van OWL ontologieën. Oorspronkelijk is de software ontwikkeld door het MIND laboratorium van de University of Maryland, maar vandaag is het een open source project.

De mogelijkheden van deze software zijn vrij beperkt in vergelijking met andere editors. Het is mogelijk om klassen, eigenschappen en instanties te definiëren en reasoning is aanwezig, maar veel meer is niet mogelijk. Ook de interface geeft een verouderde indruk en er wordt vermoed dat de gemeenschap dit project verlaten heeft in voorkeur van andere, meer belovende projecten zoals Protégé.



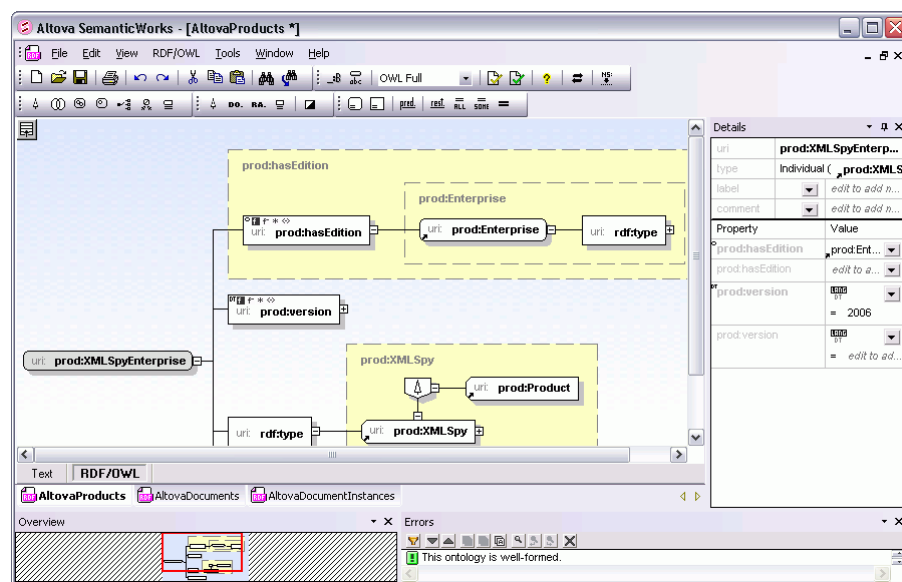
Figuur 17 Screenshot Swoop

#### 4.1.3.2 SemanticWorks 2008

Dit commercieel pakket werd ontwikkeld door Altova, een bedrijf dat al een aantal jaren werkt aan software voor webapplicaties. Een proefperiode van 30 dagen is gratis.

**SemanticWorks** hanteert een meer visuele aanpak. Het definiëren van subclasses en het leggen van verbanden met eigenschappen gebeurt met UML-achtige diagrammen. De filosofie hierachter is dat de gebruiker op deze manier gespaard wordt van het schrijven van ingewikkelde code.

Deze editor laat een veelbelovende indruk achter. Er is vooral nagedacht over het gebruiksgemak, wat niet onbelangrijk is. Langs de andere kant bestaan de argumenten die floreren bij elke discussie over commerciële en open source software. De gebruiker is volledig aangewezen op de ontwikkelaar, waar bij open source een actieve gemeenschap de software verbetert en uitbreidt. Natuurlijk speelt ook de prijs een rol.



**Figuur 18 Screenshot SemanticWorks 2008**

#### 4.1.3.3 Protégé

**Protégé** is het resultaat van een samenwerking tussen de Stanford University en de University of Manchester. De software is uitgegeven onder open source licentie en is geschreven in Java. Met meer dan 90 000 geregistreerde gebruikers kan men zeggen dat Protégé één van de populairste ontologie-editors is.

Protégé komt in twee vormen: Protégé-Frames en Protégé-OWL. In het kader van deze eindverhandeling is de Protégé-OWL editor het meest relevant.

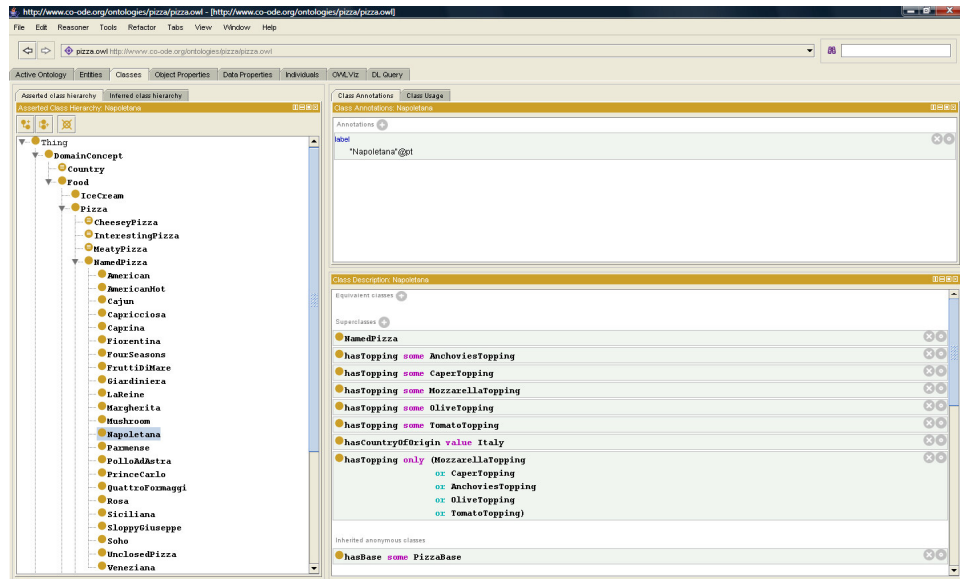
Een groot voordeel van Protégé is de belangstelling die de tool kent. Er is een actieve gemeenschap die voor feedback zorgt en een groot aantal plug-ins zijn ontwikkeld voor Protégé.

Op dit ogenblik zijn er drie versies: 3.3.1, 3.4 beta en 4.0 alpha. Er moest een keuze gemaakt worden tussen deze versies om te gebruiken voor de ontwikkeling van het nieuwe model voor de Vlaamse Overheid. Aangezien de beta al vergevorderd is, kwam de keuze te liggen tussen de beta of de alpha versie.

Beide versies hebben voor- en nadelen. De alpha versie is logischerwijs minder stabiel en ontbeert nog enkele functionaliteiten of plug-ins die nog niet omgezet zijn. Langs de andere kant is de gebruikers-interface grondig aangepast en zit het programma logischer in elkaar. Er zijn standaard twee reasoners ingebouwd, OWL 1.1 wordt ondersteund, ontologie-imports worden beter afgehandeld en de parser is geoptimaliseerd.

De beta versie biedt meer functionaliteiten zoals automatisch gegenereerde forms om instanties in te voeren, SPARQL en SWRL ondersteuning en een heel breed gamma aan plug-ins. De bètaversie voelde echter te “rommelig” aan, zowel in de gebruikers-interface als in de achterliggende code. Dit laatste is geen verwijt en volkomen logisch, aangezien de ontwikkelaars zich moeten baseren op specificaties (RDF, RDFS, OWL, SPARQL) die nog maar recentelijk als standaard aanvaard zijn.

Uiteindelijk werd gekozen om met Protégé 4.0 alpha verder te werken. De extra functionaliteiten die geboden werden door de bètaversie wogen niet op tegen de voordelen van de alpha versie. Bovendien raden de ontwikkelaars zelf aan om met de alpha versie te werken indien er geen directe nood is aan de functionaliteiten van de beta versie.

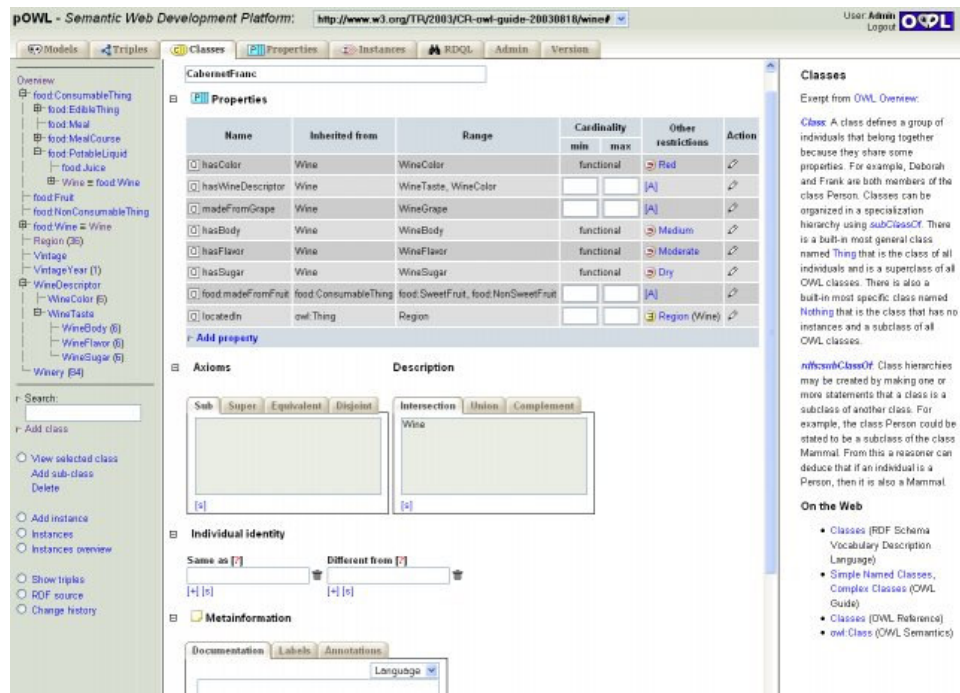


Figuur 19 Screenshot Protégé 4.0a

#### 4.1.3.4 Powl en OntoWiki

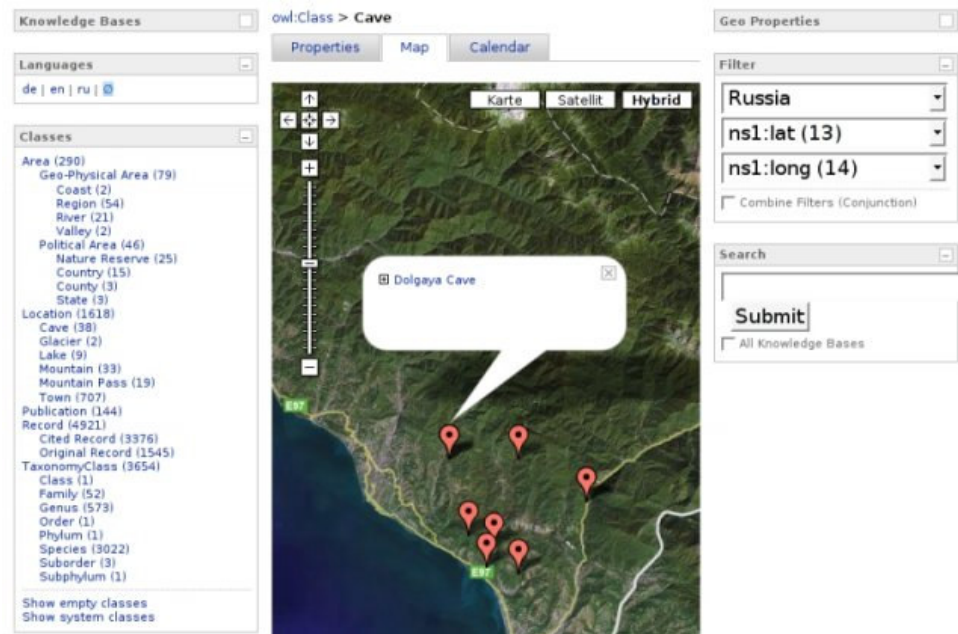
Powl en OntoWiki zijn projecten van de Agile Knowledge Engineering and Semantic Web (AKSW) onderzoeksgroep van de universiteit van Leipzig.

De filosofie achter Powl is een applicatie aan te bieden die het mogelijk maakt op gedistribueerde wijze een ontologie te ontwerpen. Powl levert een open source, webgebaseerde ontologie editor die draait op PHP.



Figuur 20 Screenshot Powl

OntoWiki is een tool die het vervolgens mogelijk maakt om op visuele en gedistribueerde wijze aan kennisbeheer te doen. Deze software maakt het mogelijk gegevens vanuit verschillende standpunten te benaderen en op semantische wijze informatie toe te voegen of aan te passen.



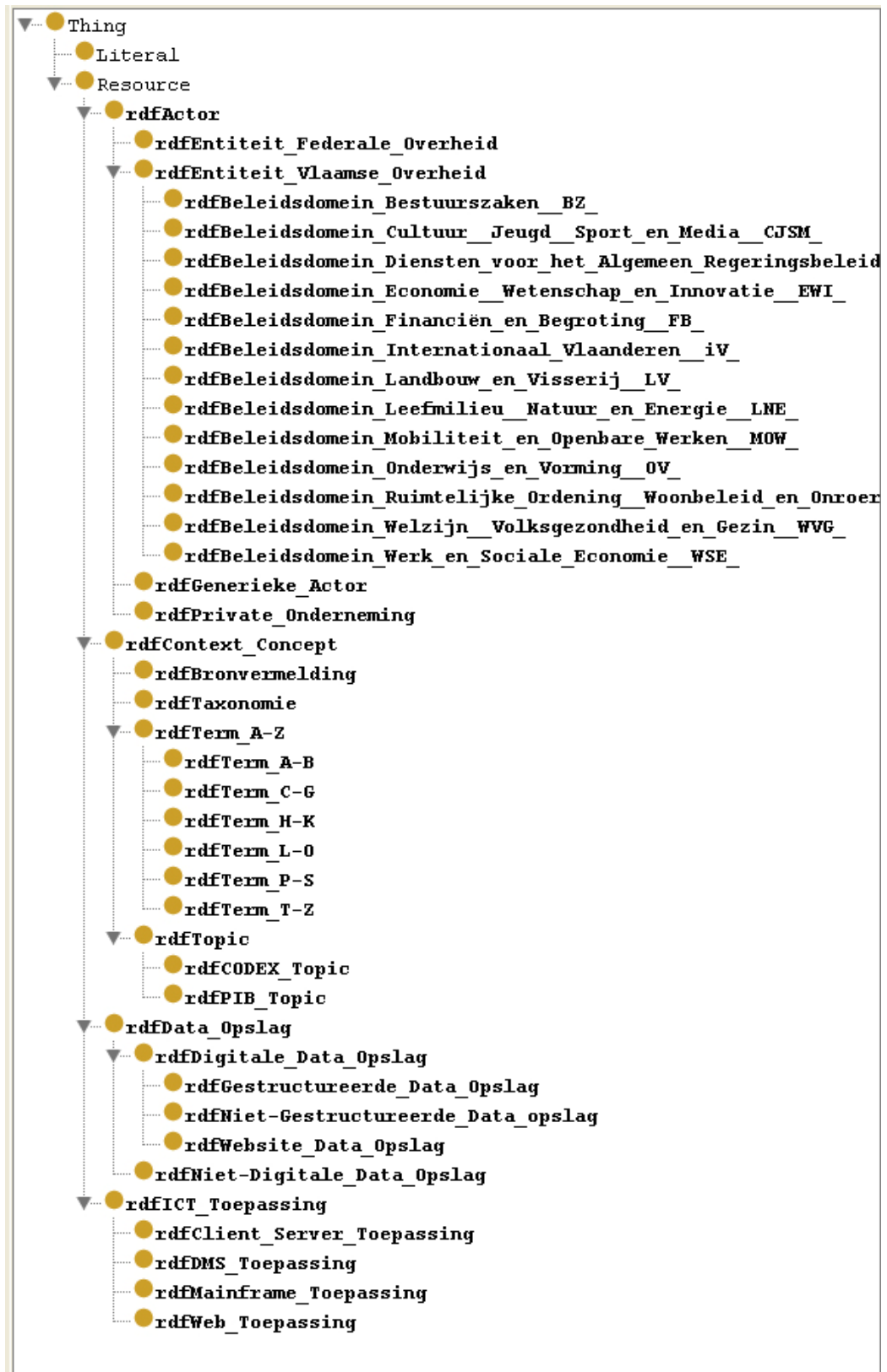
**Figuur 21 Screenshot OntoWiki**

Het is vooral de gedistribueerde manier van werken die voordelen biedt voor deze applicaties, maar wanneer slechts één persoon aan een ontologie werkt zijn er betere alternatieven dan Powl en OntoWiki.

#### 4.1.4 Het oorspronkelijke RDF-model

Ongeveer een jaar geleden werd bij de Vlaamse Overheid besloten alle databronnen in kaart te brengen. De tool die ze hiervoor kozen was Protégé. Achteraf werd er toegegeven dat Protégé niet was gebruikt waarvoor het ontwikkeld was. Het ontworpen model was niet bepaald een ontologie, maar eerder “slechts” een model van de databronnen. Desondanks bezit dit model een schat aan informatie. Daarom werd besloten het model uit te breiden en er een ontologie rond te bouwen.

Vooraleer deze nieuwe ontologie gebouwd kan worden, moet het oorspronkelijke model onderzocht worden.

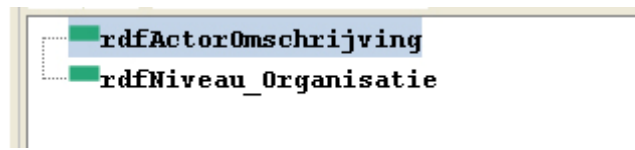


Figuur 22 Klassenhierarchie oorspronkelijk model



rdftype
rdftaxonomie_omschrijving
rdfmnet_Context
rdfdigitaal
rdfororganisatie_omschrijving
rdfbronvermelding_voor_data_opslag
rdfbrontype
rdfheeft_Data_Opslag
rdfmnet_Topics
rdfbeleidsdomein
rdfmnet_gebruiker
rdfororganisatie_aard
rdfinverse_of_beheerd_door_organisatie_
rdftoepassing_ID
rdfgestructureerd
rdftoepassing_Naam
rdfdataopslag_ID
rdfterm
rdfdataopslag_omschrijving
rdfbron_ID
rdfserver
rdfict_toepassing_bronvermelding
rdftaxonomie_ID
rdfdatum_bronvermelding
rdfterm_omschrijving
rdfdataopslag_bronvermelding
rdfgebruikt_ict_toepassing
rdfurl_website
rdftaxonomie_Titel
rdfmet_onderdelen
rdfverantwoordelijke_voor_bron
rdfvan_Taxonomie
rdfict_Toepassing_Technologie
rdfvan_Topic
rdfvlimpersID
rdftoegang
rdfnaam
rdftaxonomie_bronvermelding
rdftoepassing_omschrijving
rdfdataopslag_Naam
rdfdataopslag_Technologie
rdfvan_organisatie
rdfpersoon
rdfmnet_Term
rdfbronvermelding_voor_ict_toepassing
rdfwordt_beheerd_door
rdfwebsite
rdfcontext_voor_dataopslag
rdfbronvermelding_van_taxonomie
rdftitel_bronvermelding
rdftopic_term
rdforderdeel_van
rdfmnet_ict_Toepassing
rdfbeheerder
rdfororganisatie_niveau
rdfbeheert_dataopslag
rdfactor_ID
rdfdrager

Figuur 23 Objecteigenschappen oorspronkelijk model



**Figuur 24 Dataeigenschappen oorspronkelijk model**

De eerste screenshot toont de klassenhiërarchie. De tweede screenshot omvat de objecteigenschappen, dit zijn de eigenschappen die twee objecten met elkaar verbinden. De derde screenshot bezit twee dataeigenschappen die het mogelijk maken om een objecteigenschap in te vullen.

In de klassenhiërarchie kan men vier concepten onderscheiden: actoren, context, dataopslag en ICT-toepassing. Deze vier concepten worden elk op hun beurt uitgebreid.

De term *actor* bestaat uit de Federale Overheid, een Generieke Actor (de burger), Private Onderneming en uiteindelijk de Vlaamse Overheid zelf, die als subklassen de dertien beleidsdomeinen krijgt.

De *context* krijgt enerzijds beschrijvende mogelijkheden door trefwoorden, topics en taxonomieën en anderzijds is er ook een bronvermelding indien van toepassing.

De *dataopslag* is onderverdeeld in digitale data en niet-digitale data. Digitale data omvat gestructureerde, ongestructureerde en website dataopslag.

*ICT-toepassing* krijgt de mogelijkheden van Client-server, DMS (Document Management System), Mainframe en Web toepassingen.

Dit model legt al een degelijke basis voor een nieuwe ontologie en vooral de instanties zijn een belangrijke bron van informatie.

## 4.2 Het nieuwe Protégé-model

In dit hoofdstuk worden de verschillende stappen besproken voor de ontwikkeling van de nieuwe ontologie, gebaseerd op §2.2.2.3 *Ontwerp van een ontologie*. Vooraleer de ontwikkeling van het nieuwe model wordt besproken, zijn hier een aantal algemene principes en beslissingen gepresenteerd die in het achterhoofd werden gehouden.

- Er moet een zekere balans te vinden zijn in het model tussen enerzijds een vorm van simpliciteit om het gebruiksgemak te garanderen en anderzijds een voldoende mate van expressiviteit om de werkelijkheid zo getrouw mogelijk weer te geven. Ook aan de toekomst moet gedacht worden, waardoor het model ook gemakkelijk uitbreidbaar moet zijn.
- De voertaal die gebruikt wordt in het model is voor het grootste deel Engels. Langs de ene kant is dit omwille van ontwikkelingen die in het vakgebied vooral in het Engels verlopen. Ook wordt eventuele samenwerking met andere Europese landen vergemakkelijkt op een later tijdstip. Langs de andere kant voorzien RDF en OWL (en Protégé) de mogelijkheid om *labels* te bepalen voor gedefinieerde termen (klassen of eigenschappen), die louter voor de menselijke gebruiker van de ontologie bedoeld zijn. Hier kunnen Nederlandse vertalingen opgenomen worden.

In het model komen echter ook termen voor in het Nederlands. Het gaat hier dan om benamingen die specifiek worden gebruikt binnen de Vlaamse Overheid, zoals “Beleidsdomein”.

- Er wordt een modulaire opbouw gebruikt voor het nieuwe model. Dit vergemakkelijkt hergebruik van relevante ontologieën. Ook is het logische model en de invulling ervan met instanties gescheiden gehouden. Voor elke modulaire ontologie werd de URI <http://www.vlaanderen.be/ontologie/naam.owl> gebruikt, maar deze URI kan later nog aangepast worden.

In de ontwikkeling van een ontologie worden klassen gedefinieerd die met elkaar gelinkt worden door *objecteigenschappen* (object properties), bepaald worden door *gegevenseigenschappen* (data properties) en ingevuld worden door *instanties* (instances). Met *annotaties* wordt het mogelijk om voor de menselijke gebruiker meer uitleg te verschaffen over klassen, eigenschappen en ontologieën (meta-metadata).

Er wordt aangenomen dat de lezer tot op zekere hoogte vertrouwd is met deze termen, zoniet wordt er aangeraden de literatuur erop na te slaan. Op de website van Protégé<sup>4</sup> zijn verschillende degelijke papers te vinden die de nodige uitleg verschaffen.

De OWL syntaxis die gebruikt wordt bij Protégé is de Manchester OWL Syntax. Deze syntaxis wordt voornamelijk gebruikt voor klassedefinities en beperkingen. Het is een zeer intuïtieve syntaxis, maar indien nodig is meer informatie te vinden op [http://www.co-ode.org/resources/reference/manchester\\_syntax/](http://www.co-ode.org/resources/reference/manchester_syntax/).

Ten slotte wordt aangeraden Protégé te installeren en de ontologieën geopend te hebben, wanneer deze eindverhandeling gelezen wordt. Het is onmogelijk alles in detail te behandelen en het bekijken van de ontologieën zelf verschaft een veel duidelijker beeld dan deze geschreven tekst kan bieden.

#### 4.2.1 Het gebruik van het Protégé-model gepositioneerd met EAF

De eerste stap in het ontwerp van een ontologie (cf. §2.2.2.3) is de bepaling van het domein en werkgebied voor deze ontologie. Deze positionering kan gedaan worden met behulp van de besproken Enterprise Architecture Frameworks (cf. §3.3.2 ).

Wanneer gekeken wordt naar het **Zachman-raamwerk** (cf. Bijlage 2), is meteen duidelijk dat van de zes hoofdvragen (wat, hoe, waar, wie, wanneer en waarom) voornamelijk de wat-vraag van toepassing is op het oorspronkelijke model. Alles draait hier om *data*: waar deze opgeslagen is, wie de databronnen beheerst, enz. Binnen deze kolom zijn vooral de eerste drie standpunten van belang. De *context* van de data is zeker relevant om een duiding te kunnen geven waar deze data voor dient, wie ze kan en mag gebruiken en wie niet. Nog belangrijker zijn het *conceptuele en logische aspect*. Hier worden respectievelijk de bedrijfsentiteiten en hun verbanden en de dataentiteiten en hun verbanden gedefinieerd. Het zijn deze twee cellen die voornamelijk zijn terug te vinden in het model.

---

<sup>4</sup> <http://protege.stanford.edu/doc/users.html>

Maar ook de overige standpunten (rijen) in de datakolom zijn relevant, zij het dan in mindere mate. Zo mag men ook de technische aspecten van de databronnen niet uit het oog verliezen, al zijn deze amper uitgewerkt in het model.

Van de overige kolommen komt enkel nog de wie-vraag in aanmerking, aangezien de databronnen een beheerder hebben. Dit aspect kan men plaatsen in het fysieke standpunt, maar dit is eerder vergezocht.

In de nieuwe ontologie wordt vooral uitgebreid in het technologische standpunt, door het deels opnemen van de Technical Reference Model. Voor de wie-vraag wordt een eerste aanzet gegeven om de ontologie verder uit te breiden. Dit zijn de aspecten die het meest relevant zijn om het model in een eerste aanzet uit te breiden, zonder er al te veel moeite in te steken. Ze zijn dan ook in beperkte mate reeds aanwezig in het oorspronkelijke model en worden bijgevolg uitgewerkt in de nieuwe ontologie. Bovendien is de technische beschrijving van gegevensbronnen zeker relevant, samen met de situering van de gegevensbronnen in de juiste context.

In het licht van het **TOGAF-raamwerk** kunnen dezelfde conclusies getrokken worden. De nadruk wordt gelegd op het modelleren van de *gegevensarchitectuur*. Het model werkt voornamelijk het conceptuele en logische aspect van deze architectuur uit. Maar ook het bestuur en de organisatie, bepaald in de *zaken-architectuur*, komen aan bod in het model, zij het wel nog zeer oppervlakkig. De *technologie-architectuur* is ook vaag van toepassing, doordat de verschillende applicaties van de databronnen (DMS, client-server,...) ook opgenomen zijn in het model.

In de nieuwe ontologie is deze technologie-architectuur meer uitgewerkt en is ook de zaken-architectuur meer uitgebreid. Zo wordt het mogelijk aspecten als een bedrijfsstrategie op te nemen in de nieuwe ontologie.

Voor het **FEAF-raamwerk** is er slechts één component echt van toepassing en dat zijn de *architecturale modellen*. Het model omschrijft niet hoe een overgang moet verlopen naar een doelarchitectuur, maar beschrijft enkel de huidige toestand.

Van deze modellen komen er vooral twee aan bod: het *Business Reference Model* (BRM) en het *Technical Reference Model* (TRM). Het BRM beschrijft hoe de organisatie in elkaar zit en is terug te vinden in de structuur van de Vlaamse Overheid en belangenpartijen die in het Protégé-model zijn opgenomen. Het TRM is van minder belang. Dit model is voornamelijk terug te vinden in de ICT-toepassingen en classificatie van de databronnen.

Zoals eerder al vermeld zijn in de nieuwe ontologie delen van het TRM opgenomen. De nieuwe ontologie is ook deels gebaseerd op het BRM, maar neemt niet veel van dit model letterlijk over, wegens een te Amerikaanse visie.

## 4.2.2 De ontwikkeling van het nieuwe Protégé-model

### 4.2.2.1 Bestaande ontologieën

De tweede stap (cf. §2.2.2.3) in de ontwikkeling van een ontologie is het onderzoeken of reeds bestaande ontologieën hergebruikt kunnen worden. De ontwikkeling van het nieuwe model is uiteraard gebaseerd op het oorspronkelijke model. Ook werd dankbaar gebruik gemaakt van de Dublin Core Metadata Set en een aantal modellen van het FEA-RMO.

#### 4.2.2.1.1 Ontologieën bij de Vlaamse Overheid

Binnen de Vlaamse Overheid bestaan nog geen ontologieën, dus hier moest geen rekening mee gehouden worden. Wel is een grootschalig project aan de gang, het Flanders Research Information Space (FRIS) project, waar gebruik zal gemaakt worden van ontologieën. Dit project zal een soort Kruispuntbank worden voor Vlaamse onderzoekers en onderzoeksprojecten. Het lijkt interessant dit project op te volgen en later te onderzoeken of een bepaalde vorm van samenwerking mogelijk is.

#### 4.2.2.1.2 Dublin Core

Wanneer buiten de grenzen van de Vlaamse Overheid gekeken wordt, komt men onvermijdelijk bij het Dublin Core Metadata Initiative (DCMI). Dit initiatief biedt een set van 15 generieke metadata elementen aan, samen met verfijningen van deze elementen. Deze elementen worden gebruikt om over verschillende domeinen heen gegevensbronnen te beschrijven.

Een passende analogie is deze van een boekenfiche in de bibliotheek waar informatie over het boek (metadata dus) opgenomen is op een steekkaart, zoals auteur en titel. Een lijst van de DCMI metadata elementen kan gevonden worden op <http://dublincore.org/documents/dcmi-terms/>.

Van deze set metadata elementen bestaan verschillende Application Profiles, profielen die meer gericht zijn op één bepaalde toepassing. Zo is er ook een Application Profile voor overheden, de Dublin Core Government Application Profile (DC-GAP<sup>5</sup>). Dit profiel houdt in dat een aantal verfijningen geschrapt worden, een aantal verfijningen toegevoegd worden en dat de definities en beschrijvingen van de elementen beter en meer specifiek gedetailleerd worden.

De leden van deze DC-GAP zijn meestal zelf werkzaam binnen een overheid die actief bezig is met metadata. Twee metadata-initiatieven op grote schaal zijn deze van de Engelse overheid en Nederlandse overheid. De Engelse overheid gebruikt de **e-Government Metadata Standard** (eGMS) wat kadert binnen het e-Government Interoperability Framework initiatief. Dit initiatief helpt de Engelse overheid in hun strategie om de publieke diensten efficiënter te maken en te verbeteren via een betere IT ondersteuning.

Een meer interessant project in het kader van deze eindverhandeling, is deze van de Nederlandse overheid. Het project dat in 2004 van start ging, kreeg de naam **Overheid.nl Webmetadata** en de motivatie was de doelstelling om “*voor [www.overheid.nl](http://www.overheid.nl) en andere overheidsportalen, een zoekrobot te ontwikkelen die snel en efficiënt de burger de overheidsinformatie, diensten en organisaties kan presenteren.*” Het doel was dus alle informatie aanwezig op overheidswebsites van metadata te voorzien, waardoor informatie beter ontsloten kan worden naar de burger toe.

In het Handboek (versie 3.0) dat uitgebracht werd in 2006 is een lijst van elementen te vinden die het mogelijk maken gegevensbronnen van zinvolle metadata te voorzien. Voor de nieuwe ontologie is dit een dankbare bron aan informatie, maar er werd afgezien van het zomaar overnemen van de gedefinieerde termen of het gebruiken van hun URI's. De reden hiervoor is dat een ontologie meer doet dan een gegevensbron van metadata voorzien. Er worden allerlei relaties gelegd en dit is te zeer beperkt in de Dublin Core en zijn afgeleiden. Een voorbeeld kan misschien verduidelijken.

---

<sup>5</sup> <http://dublincore.org/dcgapwiki>

In de Dublin Core kan men een element *Creator* terugvinden. Dit is terecht want weten wie een bepaalde bron aangemaakt heeft, is zinvolle informatie. Binnen een ontologie is echter meer nodig dan deze eenzijdige relatie. Zo bestaat er de persoon *Creator*, welke waarschijnlijk een instantie is van een persoon met een voor- en achternaam en verschillende andere eigenschappen. Het lijkt bijgevolg correcter de relatie tussen de gegevensbron en de persoon te benoemen als *hasCreator*. Op deze manier kan men van elke gegevensbron zien wie deze ontworpen heeft. Maar het is ook nuttig om te weten welke gegevensbronnen een bepaalde persoon allemaal ontwikkeld heeft. Dit is de inverse relatie van *hasCreator*, namelijk *isCreatorOf*. Door deze objecteigenschap te definiëren en te bepalen dat deze de inverse is van *hasCreator* kan Protégé met de reasoner afleiden welke bronnen deze persoon heeft ontwikkeld, zonder dat deze relaties ergens expliciet zijn opgenomen.

Dit voorbeeld wil dus aantonen waarom het nodig was verder te gaan dan de eenzijdig bepaalde termen van de Dublin Core. De termen zijn wel nuttig om op te nemen en later eventueel uit te breiden, en de omschrijvingen (of annotaties) van deze termen die terug te vinden zijn in het Handboek, werden dankbaar hergebruikt.

Concreet werd een ontologie *descriptive.owl* aangemaakt. Hierin zijn de object- en gegevenseigenschappen terug te vinden, samen met de algemene klasse *Resource*.

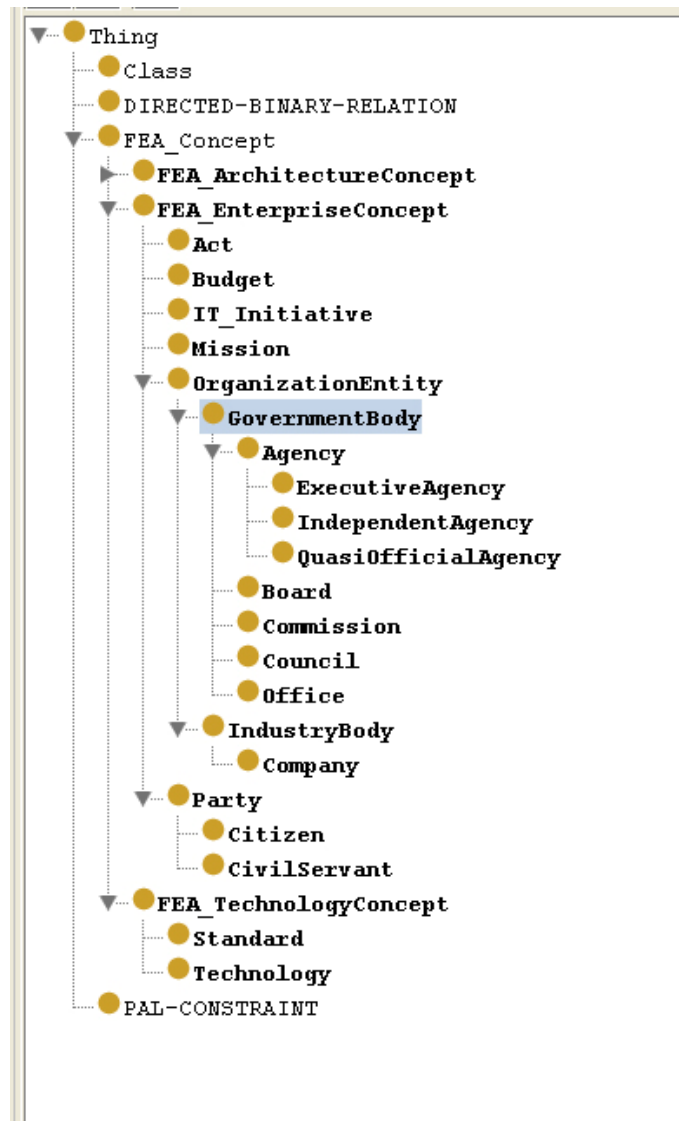
#### 4.2.2.1.3 FEA-Reference Model Ontologies

Ook Enterprise Architecture kan baat hebben bij de ontwikkeling van ontologieën. Het bedrijf Topquadrant formaliseerde de Reference Models van de Federal Enterprise Architecture (FEA) in een aantal integreerbare ontologieën. Ze ontwierpen ontologieën van de *Business Reference Model* (BRM), de *Performance Reference Model* (PRM), de *Service Component Reference Model* (SRM) en de *Technical Reference Model* (TRM). De Data Reference Model (DRM) is op dit moment nog niet opgenomen in de **FEA-Reference Model Ontologies** (FEA-RMO). Een *Government Core* ontologie bepaalt de concepten die niet expliciet gedefinieerd zijn door de FEA, maar die wel gebruikt worden in de modellen.

De paper die hoort bij de FEA-RMO legt uit wat de meerwaarde is van deze ontologieën en beschrijft de beoordeling van de semantiek van FEA, de ontwikkeling van de FEA-RMO ontologieën en een aantal use cases, om af te sluiten met enkele aanbevelingen.



Voor de ontwikkeling van het nieuwe model is vooral de Government Core van belang. Klassen zoals *Company*, *Agency*, *Citizen* of *Mission* krijgen allemaal hun plek in de ontologie.

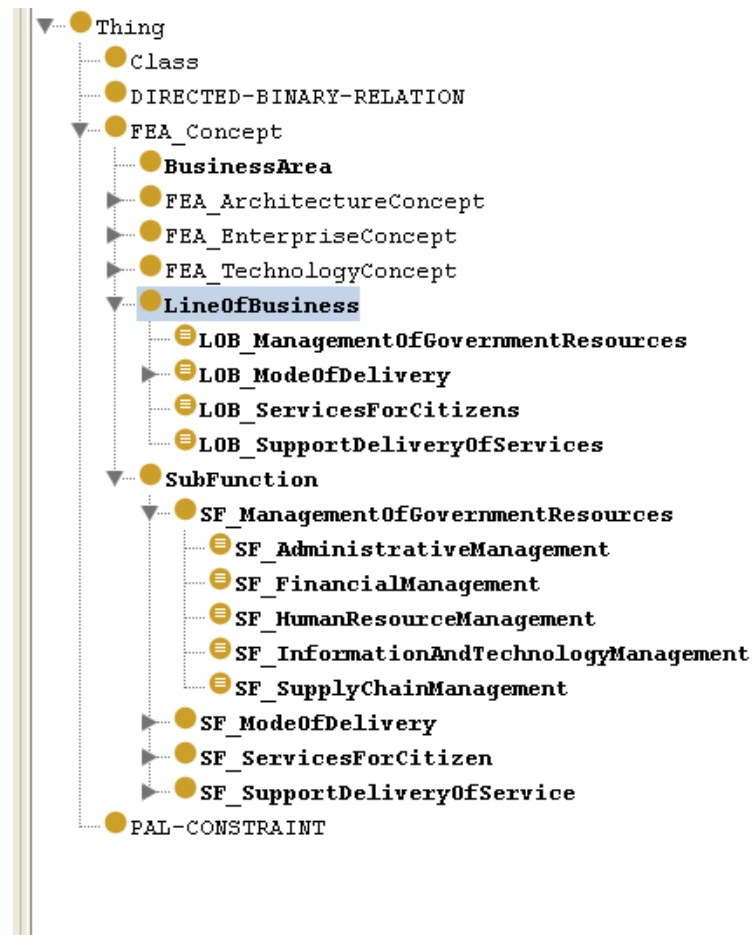


**Figuur 25 FEA-RMO Government Core**

De BRM is niet rechtstreeks te gebruiken, aangezien deze gemodelleerd is naar het Amerikaanse overheidsmodel.

Daar waar de Amerikaanse overheid ingedeeld is in vier Business Areas met elk een aantal Subfunctions, is de Vlaamse Overheid ingedeeld in dertien Beleidsdomeinen met elk één departement en verscheidene agentschappen.

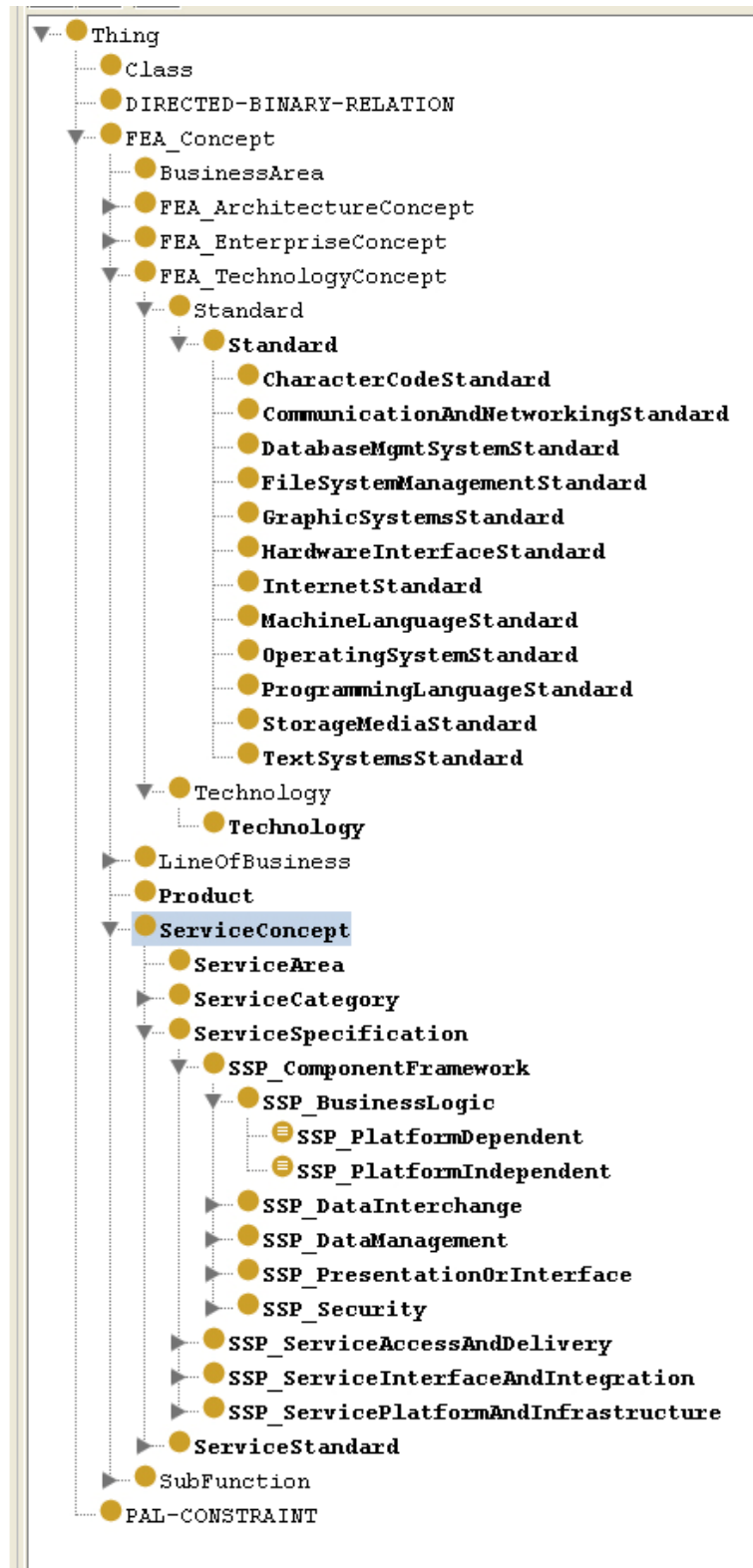
De BRM is wel gebruikt als inspiratie voor de plaatsing van enkele klassen, zoals Beleidsdomein.



**Figuur 26 FEA-RMO BRM**

De Government Core en BRM zijn beide geaggregeerd in een nieuwe ontologie *business.owl*.

De TRM ten slotte is wel vrij letterlijk overgenomen. Er worden vooral generieke concepten gemodelleerd in het FEA-RMO TRM model, waarvan slechts een aantal gebruikt zullen worden in het nieuwe model. De concepten die niet gebruikt worden zijn echter ook aanwezig in het nieuwe model, omdat ze op deze manier gemakkelijk toegankelijk zijn voor eventuele uitbreidingen.



**Figuur 27 FEA-RMO TRM**

De TRM is terug te vinden in *technology.owl*.

#### 4.2.2.2 *Uitwerking van de Descriptive, Business en Technology ontologieën*

Stap 3 en stap 4 van het ontwerp van een ontologie (cf. §2.2.2.3), namelijk het opsommen van de belangrijkste termen van de ontologie en het definiëren van de klassen en de klassenhiërarchie, zijn voor een groot deel al voltooid in enerzijds het oorspronkelijke model en anderzijds het hergebruik van bestaande ontologieën. Deze twee onderdelen moeten echter nog samengebracht en uitgebreid worden.

In de *Descriptive ontologie* is de onderverdeling van taxonomie, categorie en trefwoord opgenomen. Dit zijn drie klassen die niet in een subklasse-superklasse hiërarchie gerelateerd staan. Een trefwoord is namelijk geen specialisatie van een categorie of taxonomie. Ze zijn wel alle drie een subklasse van een beschrijvingsconcept.

Ook werd de klasse Bronvermelding in de ontologie opgenomen.

De *Business ontologie* krijgt als extra klassen voornamelijk de verschillende indelingen van de Vlaamse Overheid: cellen, teams, diensten, afdelingen, departementen, directeurs, n-1 entiteiten, project n-niveau, raden en advies organen en agentschappen (intern en extern). De termen cel, dienst en team zijn als equivalenten gedefinieerd, zodat het niet uitmaakt welke benaming er gekozen wordt. Ze worden als synoniemen behandeld.

Beleidsdomein wordt een klasse apart, zonder subklassen. Het oorspronkelijke model definieerde de dertien beleidsdomeinen als subklassen van de superklasse Beleidsdomein. Deze benadering is niet geheel consistent, aangezien de dertien beleidsdomeinen geen klassen maar instanties zijn. Met de objecteigenschappen *hasPart* en zijn inverse *isPartOf* wordt de hiërarchie binnen de Vlaamse Overheid bepaald.

De klasse Person krijgt ook hier een plaats, welke niet te verwarren is met de klasse Citizen. Person krijgt ook een subklasse, AmbtenaarVO die een ambtenaar bij de Vlaamse Overheid voorstelt. De definitie van deze klasse vereist echter een instantie, want dit zijn alle personen die werken voor de Vlaamse Overheid (een instantie van “Overheid”), dus wordt deze klasse nog verder verfijnd in de ontologie *instances.owl* (cf. §4.2.2.2).

Tot slot wordt ook de *technology.owl* ontologie uitgewerkt. Hier zijn het vooral de gegevensbronnen die verder uitgediept worden.

Een bedenking bij deze uitwerking is om informatie over gegevensbronnen zo granulair mogelijk uit te werken. In het oorspronkelijke model is een gegevensbron bv. geklasseerd als een database en als een digitale bron. Beide gegevens zijn expliciet opgenomen. Als men nu echter enkel zou zeggen dat de gegevensbron draait op een Oracle 8 databasesysteem, dan kan er automatisch meer informatie afgeleid worden. Zo kan er afgeleid worden dat dit een database is, die draait op een product van Oracle en dat dit een digitale bron is. Deze manier van werken levert een meer nauwkeurig resultaat op en is flexibeler.

Er is begonnen met *gespecificeerde gegevensbronnen*: databases, digitale documenten, websites en papieren documenten. Er is voor elke bron een beperking opgelegd. Een database kan enkel op een medium als DB2, Sybase, enz. bestaan (samengevat in de klasse *SSP\_Database*).

```
hasMedium only SSP_Database
```

Op gelijke wijze bestaat een digitaal document enkel op een SAN, NAS, enz. (*SSP\_Storage*), een website enkel op een IIS, Apache, enz. (*SSP\_Webserver*) en een papieren document is alles wat niet een database, digitaal document of website is. Voorlopig werd er bij wijze van experiment (en bij gebrek aan de correcte informatie) ook UnkownDB, UnknownStorage en UnknownWebserver opgenomen, om geen verkeerde kennis weer te geven in het model. Noteer wel dat de concrete invullingen (DB2, SAN, Apache, enz.) allemaal instanties zijn die pas bepaald worden in de *instances.owl* ontologie.

De vorige beperking bepaalt dus dat *indien* er een medium gespecificeerd is, dit voor een database enkel van de klasse *SSP\_Database* afkomstig kan zijn. Een andere beperking zou kunnen zijn:

```
hasMedium some SSP_Database
```

Om een gegevensbron een database te kunnen noemen, moet er minstens één *hasMedium*-relatie zijn met een instantie uit de klasse *SSP\_Database*.

De twee voorbije beperkingen combineren resulteert in de meest correcte semantiek, namelijk dat een database enkel een *hasMedium*-verband kan hebben met instanties uit de klasse *SSP\_Database* en dat ze minstens één dergelijk verband moet hebben.

De reden waarom deze tweede beperking niet opgenomen is in het nieuwe model is tweevoudig. Langs de ene kant bestaat hierdoor nog altijd de mogelijkheid om instanties van de klasse *Database* aan te maken, zonder te moeten specificeren op welk medium deze draait. Langs de andere kant (en samenhangend met het vorige) ontbreekt deze informatie in het oorspronkelijke model, waardoor een migratie van de gegevens zou resulteren in een inconsistente ontologie. Wanneer de gegevensbronnen overgezet zouden worden in de juiste klasse, maar zonder gespecificeerde medium, zou Protégé een foutmelding geven en zou de reasoner niet kunnen werken, aangezien deze enkel met consistente ontologieën overweg kan.

Op dit punt is dan ook besloten het model in twee vormen aan te bieden. De eerste vorm waar alle instanties geïmporteerd zullen worden, heeft de vorige beperking niet, waardoor ook de classificatie in digitale en niet-digitale bronnen niet kan gebeuren (cf. infra). De tweede vorm beperkt zich tot een klein aantal instanties om de mogelijkheden van het model aan te tonen en neemt de vorige beperking wel op in een gedefinieerde klasse.

Er is aangenomen dat een gegevensbron minstens altijd de classificatie database, digitaal document, website of papieren document krijgt (via een rechtstreekse classificatie in de betreffende klassen in de eerste vorm van het model, of via het *hasMedium*-verband in de tweede vorm). De andere indelingen gebeuren met definities. Op dit punt loont het even stil te staan bij de **Open World Reasoning** die gehanteerd wordt bij de ontwikkeling van Semantisch Web applicaties.

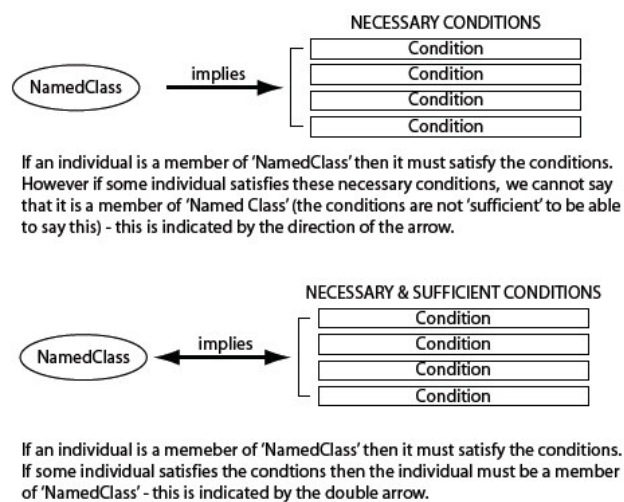
Door beperkingen op te leggen aan een klasse, kan de **consistentie gecontroleerd worden** door de reasoner. Wanneer een gegevensbron als klasse een *Database* krijgt, dan kan de reasoner nakijken of het medium van deze bron afkomstig is van de klasse *SSP\_Database*. Dit is een *necessary condition*. Een andere mogelijkheid van de reasoner is het **klasseren van instanties**. Zo kan er geredeneerd worden dat wanneer een gegevensbron draait op een DB2 installatie, dit dan wel een database moet zijn. Dit is voor ons (mensen) logisch, maar de computer kan dit niet weten. De Open World assumptie gaat ervan uit dat eventueel nog andere beperkingen opgelegd zijn aan de klasse *Database*, die (nog) niet in het model zijn opgenomen.

Een triviaal voorbeeld kan misschien verduidelijken:

Een subklasse OudeMan van de klasse Persoon wordt bepaald. Een beperking is dat het geslacht van instanties van deze OudeMan klasse mannelijk moet zijn. Wanneer er nu instanties aangemaakt worden met als klasse OudeMan, dan kan de reasoner nakijken of het hier wel degelijk gaat om personen van het mannelijke geslacht.

Maar wanneer instanties aangemaakt worden die niet tot een klasse behoren, maar wel mannelijk als geslacht hebben, zou de reasoner kunnen besluiten dat ze ook tot de klasse OudeMan behoren, omdat ze voldoen aan de beperking. Voor mensen is het echter duidelijk dat de beperking geslacht niet voldoende is. Niet alle mannen behoren tot de klasse OudeMan. Het zijn alleen maar mannen die een leeftijd hebben van bv. 70 of hoger. De Open World assumptie houdt dus rekening met de mogelijkheid dat er nog meer beperkingen aanwezig zijn, die niet in het model zijn opgenomen.

Een tweede beperking moet dus worden bepaald voor de klasse van OudeMan, namelijk dat de leeftijd 70 of hoger moet zijn. Daarna wordt aangegeven dat deze twee beperkingen voldoende zijn. Dit worden dan *necessary and sufficient conditions*. Er wordt dus expliciet aangegeven dat een oude man bepaald wordt door het mannelijke geslacht en een leeftijd van 70 of hoger en dat dit de twee enige beperkingen zijn, nodig om de klasse te bepalen. Op deze manier kan de reasoner nu wél alle oude mannen gaan klasseren in de juiste klasse. In Protégé gebeurt dit door een *Primitive Class*, een klasse met een aantal necessary conditions, om te zetten in een *Defined Class*.



**Figuur 28 Necessary en sufficient condities (bron: University of Manchester 2007)**

In de nieuwe ontologie voor de Vlaamse Overheid zijn een aantal soorten gegevensbronnen gedefinieerd in Defined Classes. Een eerste indeling is deze afkomstig van de **ICT-toepassing**, die ook te vinden is in het oorspronkelijke model.

Het oorspronkelijke model was niet geheel consistent bij deze indeling. De classificatie van ICT-toepassingen in client-server-, DMS-, mainframe- en webtoepassingen is zinvol, maar het gaat over dezelfde gegevensbronnen als bij de indeling in database, digitaal document, website en papieren document. Hoewel het om dezelfde gegevensbronnen gaat, zijn dit verschillende instanties in het oorspronkelijke model. In het nieuwe model wordt hiervan afgestapt en krijgen de instanties van de eerste indeling een objecteigenschap *hasApplication* (heeft toepassing) indien relevant. Daarna worden klassen gedefinieerd (dus met necessary and sufficient conditions) zodat de reasoner deze indeling zelf kan afhandelen. De definitie van een mainframe toepassing ziet er als volgt uit:

```
Resource
and hasApplication value Mainframe
```

Ook hier is Mainframe een instantie, die pas later bij de *instances.owl* ontologie ingegeven wordt.

Eenzelfde inconsistentie wordt gemaakt met de klasse **Bronvermeling** in het oorspronkelijke model. Ook hier gaat het om dezelfde gegevensbronnen als database, digitaal document, website of papieren document, maar worden ze in andere instanties opgeslagen. Deze instanties zijn dus ook verwijderd en vervangen door een objecteigenschap *hasSource* op de gegevensbronnen waar dit van toepassing was.

Een derde indeling is deze in **digitale en niet-digitale bronnen**. Een digitale bron (DigitalResource) heeft de definitie:

```
Resource
and hasMedium some (SSP_DeliveryServices
                    or SSP_DatabaseOrStorage)
```



Een niet-digitale bron is disjunct met een digitale bron. Enerzijds wordt bij *disjoint classes* DigitalResource aangeduid en anderzijds krijgt de klasse NonDigitalResource de volgende definitie:

```
Resource
and not DigitalResource
```

Wanneer de optie Edit > Classify gekozen wordt in Protégé, verschijnt de Inferred class hierarchy, waar database, digitaal document en website geklasseerd staan onder DigitalResource en papieren document onder NonDigitalResource.

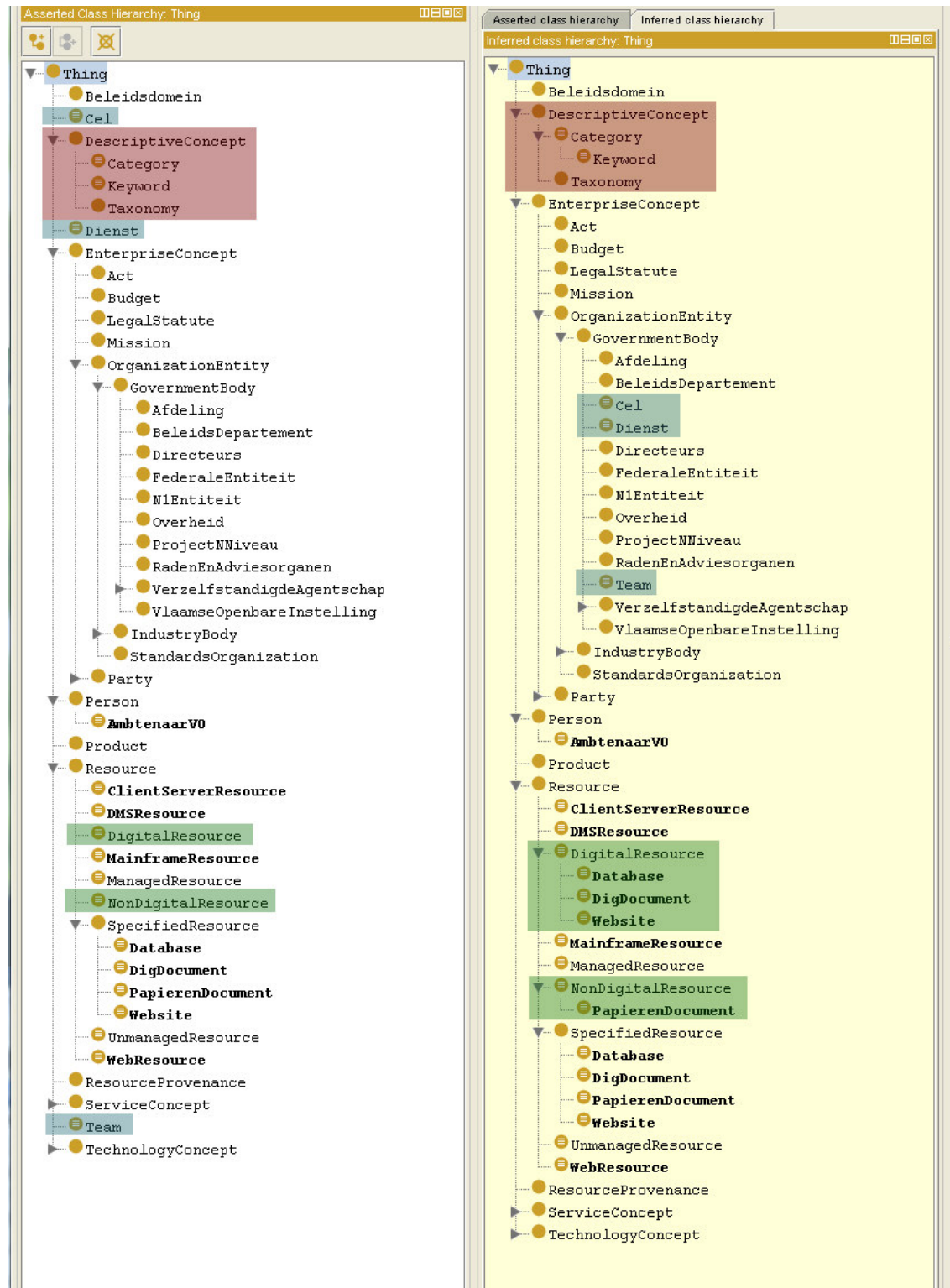
Een laatste indeling is gemaakt om de verdere mogelijkheden van het model aan te tonen. Het kan nuttig zijn te kijken welke gegevensbronnen een beheerder hebben toegewezen en welke niet. Een **ManagedResource** is bijgevolg gedefinieerd als:

```
Resource
and hasResourceManager some Person
```

En een UnmanagedResource is:

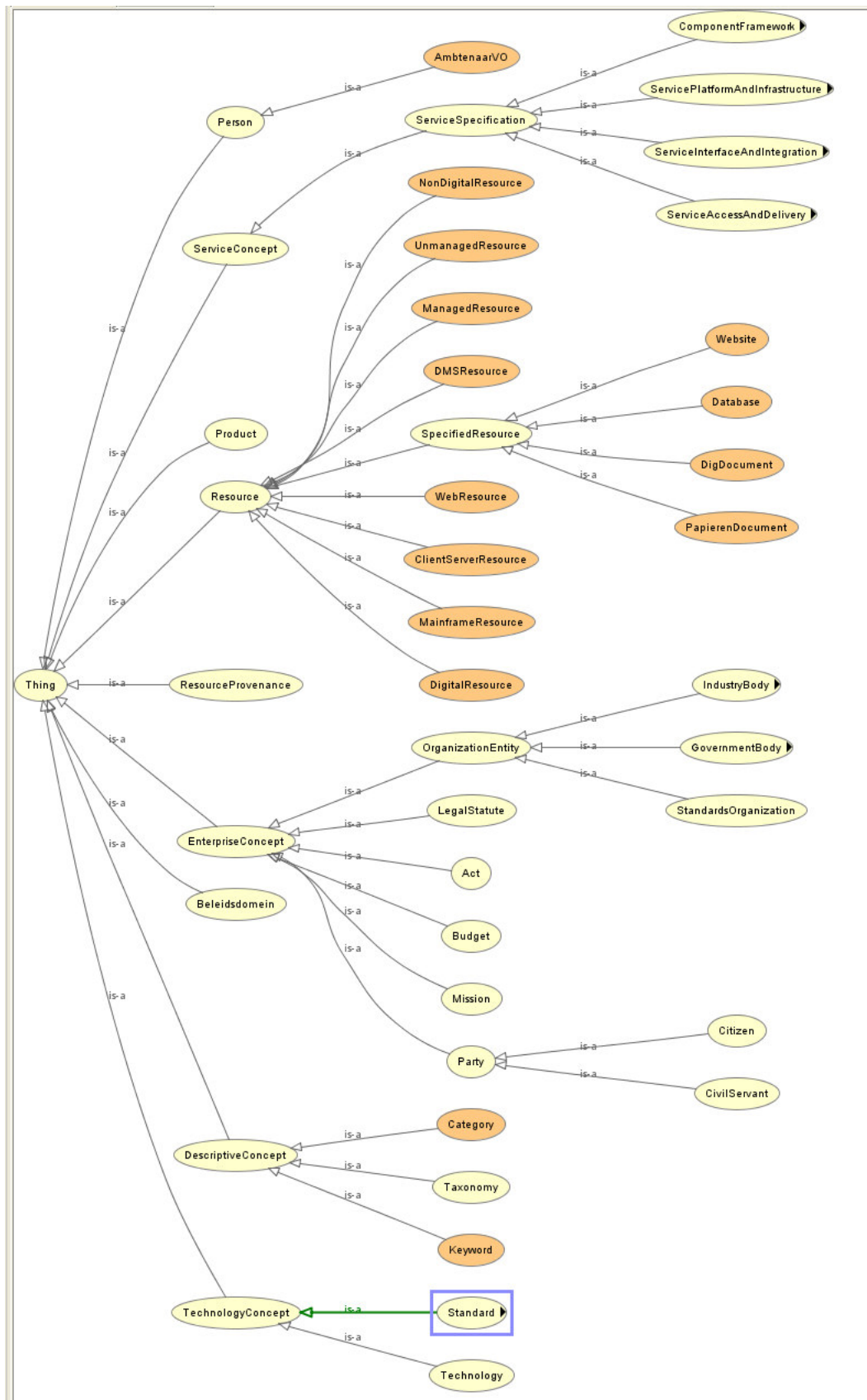
```
Resource
and not ManagedResource
```

Volgende screenshot toont de *Asserted Class Hierarchy* (de klassenhierarchie die ingegeven is) en *Inferred Class Hierarchy* (de klassenhierarchie die afgeleid is) naast elkaar. Drie verschillende kleuren worden gebruikt om de verschillen aan te duiden.



Figuur 29 Asserted Class Hierarchy en Inferred Class Hierarchy

Een alternatieve weergave van de klassen is te vinden in volgende screenshot.



Figuur 30 OWLviz

#### 4.2.2.3 Object- en dataeigenschappen

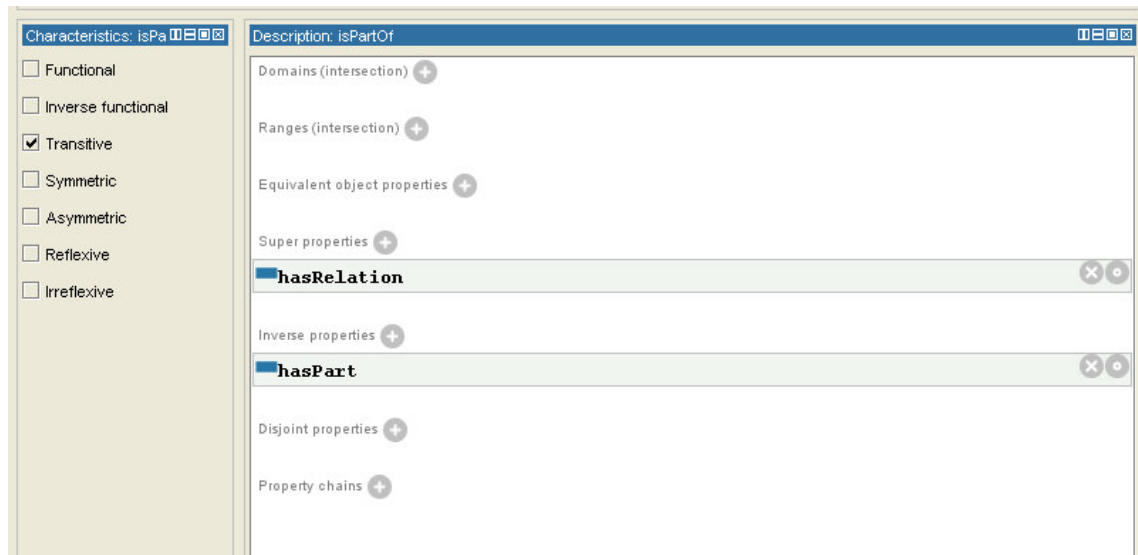
De twee volgende stappen in de ontwikkeling van een ontologie (cf. §2.2.2.3) houden zich bezig met het uitwerken van de eigenschappen van klassen.

In de vijfde stap worden de eigenschappen zelf bepaald. Twee soorten eigenschappen zijn aanwezig, object- en dataeigenschappen. Objecteigenschappen verbinden twee objecten (instanties van een klasse) met elkaar, terwijl een dataeigenschap een waarde geeft voor een bepaalde eigenschap (een string, boolean, enz.).

De meeste eigenschappen zijn al aanwezig door het overnemen van de Dublin Core metadata set. Slechts een beperkt aantal nieuwe eigenschappen werden aangemaakt. In de *descriptive.owl* ontologie werden de eigenschappen *isCategoryOf*, *isKeywordOf* en *isTaxonomyOf* extra opgenomen, samen met de inversen. *Business.owl* krijgt extra objecteigenschappen *isManagerOf*, *isEmployeeOf*, *worksIn* en inversen, en ook nog *hasLegalStatute* om het statuut van de verschillende agentschappen aan te geven. De dataeigenschappen worden uitgebreid met *hasName* (met voor- en achternaam subklassen) en *hasCodeVlinders*. Voor *technology.owl* komt een groter aantal eigenschappen bij: *hasPublisher* legt het verband met de persoon of afdeling die een bron publiceerde, *hasResourceManager* met de beheerder van een bron. De *hasApplication* eigenschap bepaalt de toepassing van een gegevensbron (mainframe, webserver, client-server of DMS). *HasAuthorization* laat toe te zeggen of een bron een read/write of enkel read toegang heeft. Met *usesProduct* wordt de link gelegd tussen een softwareproduct (bv. MS Excel) en een gegevensbron, terwijl deze producten een *Vendor* hebben die bepaald wordt met de eigenschap *hasVendor*. De *hasURL* eigenschap wordt geplaatst als subklasse van *hasIdentifier* en is ook een dataeigenschap.

De zesde stap werkt de eigenschappen verder uit. Van objecteigenschappen kan men bepalen of ze Functional, Inverse functional, Transitive, Symmetric, Asymmetric, Reflexive of Irreflexive zijn. Van dataeigenschappen kan men alleen bepalen of ze Functional zijn.

Daarnaast kan men ook het domein en werkgebied bepalen van eigenschappen en equivalente, inverse en disjuncte eigenschappen.



**Figuur 31** Objecteigenschappen in Protégé

Bij wijze van voorbeeld wordt één aspect van de ontologie verder uitgediept.

De objecteigenschap *isPartOf* wordt gedefinieerd als een subklasse van *hasRelation*. Deze eigenschap wordt onder andere gebruikt om aan te duiden dat een bepaalde cel een onderdeel is van een afdeling, dat een afdeling een onderdeel is van een agentschap, dat een agentschap een onderdeel is van een beleidsdomein en dat het beleidsdomein een onderdeel is van een overheid. De eigenschap *isPartOf* is een transitieve eigenschap, wat wil zeggen dat een cel niet alleen een onderdeel is van een afdeling, maar ook van een agentschap, een beleidsdomein en een overheid.

De inverse van *isPartOf* is *hasPart*. Dit wil zeggen dat wanneer een cel een onderdeel is van een afdeling, de reasoner zal kunnen afleiden dat deze afdeling de cel als onderdeel heeft. Ook zullen het desbetreffende agentschap en beleidsdomein een *hasPart* verband hebben met de cel, vanwege de transitiviteit.

Nu zijn *isPartOf* en *hasPart* beide een subklasse van *hasRelation*. Deze superklasse is transitief en symmetrisch (als A een relatie heeft met B, heeft B ook een relatie met A) en heeft naast de *hasPart* en *isPartOf* subklassen, nog andere subklassen zoals *isEmployeeOf*. Dit heeft verre gaande gevolgen.

Neem nu het volgende concrete voorbeeld:

Persoon X werkt in cel A en persoon Y werkt in cel B. Beide cellen zijn een onderdeel van elk een afdeling, agentschap, beleidsdomein en uiteindelijk de Vlaamse Overheid. Er kan met dit model een antwoord gevonden worden op de vraag: “Hoe relateren persoon X en persoon Y zich tot elkaar?” De reasoner zal afleiden dat beide personen, via de *isEmployeeOf* en *isPartOf* subklassen, een *hasRelation* verband hebben met de Vlaamse Overheid. Nog beter is de mogelijkheid om dit verband weer te geven, om te tonen hoe deze redenering gemaakt werd (dit kan nog niet in Protégé 4.0 alpha maar is voorzien voor latere versies).

#### 4.2.2.4 Creëren van instanties

De laatste stap in de ontwikkeling van de ontologie (cf. §2.2.2.3) betreft de creatie van instanties.

De drie ontologieën die hierboven besproken werden, namelijk *descriptive.owl*, *business.owl* en *technology.owl*, worden geïmporteerd in een nieuwe ontologie *instances.owl*. Deze ontologie beperkt zich tot het beheer van instanties en enkele definities die een bepaalde instantie gebruiken zoals *AmbtenaarVO* (ambtenaar van de Vlaamse Overheid)

```
Person
and isEmployeeOf value VlaamseOverheid
```

waar *VlaamseOverheid* een instantie is van de klasse *Overheid*.

Om te beginnen werden de organisatorische entiteiten ingevuld in het model. Dit zijn alle beleidsdomeinen, agentschappen, departementen, afdelingen, enz. Een import van het oorspronkelijke model was mogelijk, mits een hele reeks aanpassingen aan klassen, eigenschappen en instanties zelf. Hoewel hier veel tijd ingestoken moest worden, is het irrelevant deze migratie te bespreken.

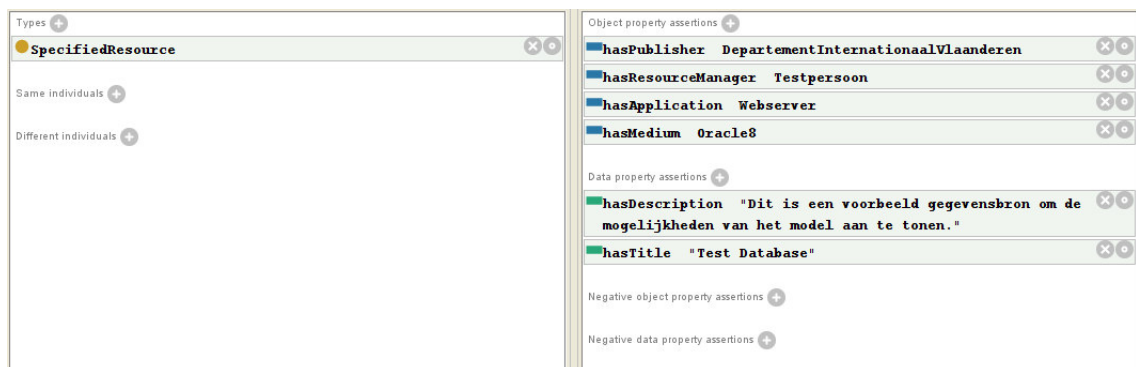
Vervolgens werden ook de gegevensbronnen ingeladen. Deze instanties zijn in een apart bestand gehouden, namelijk *resources.owl*. Op dit moment gaf de reasoner het op en blokkeerde deze indien hij geactiveerd werd. Een zoektocht door de mailing-lists van de Protégé-website resulteerde in het verhogen van het werkgeheugen tot 700 MB.

Dit hielp een beetje (de classificatie van klassen werd uitgevoerd), maar niet genoeg (de andere functionaliteiten van de reasoner werkten nog altijd niet). Een andere posting gaf de verklaring dat de reasoner nog niet goed overweg kan met grote ontologieën, maar dat hier aan werd gewerkt.

De ingeladen gegevensbronnen kregen ook een grondige kuisbeurt, wat wederom veel tijd in beslag nam maar niet interessant is om verder over uit te weiden.

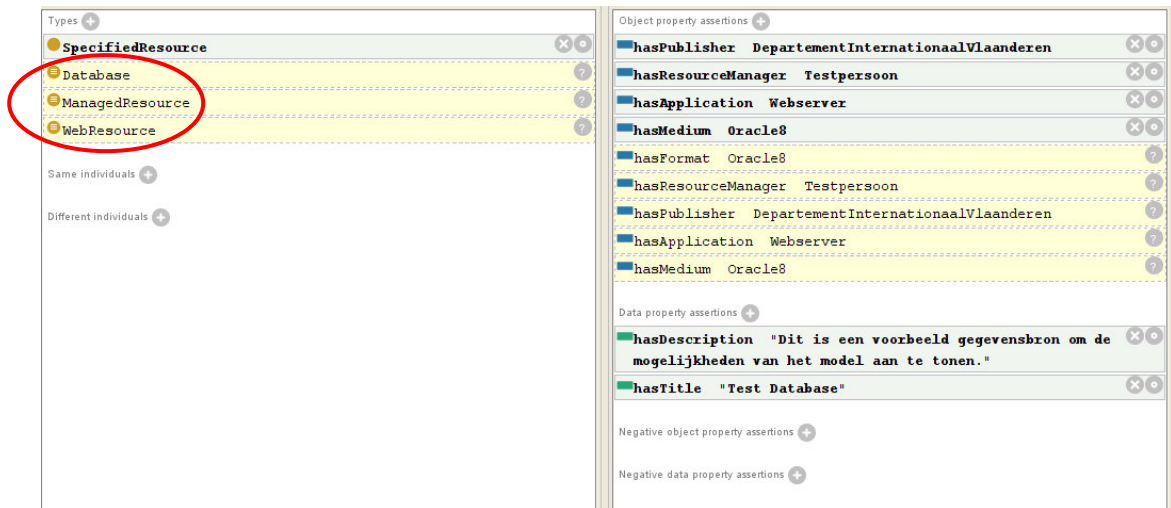
Hier eindigt ook de ontwikkeling van het model. Een aantal zaken zijn redundant in het model, zoals het opnemen van *hasPart* én *isPartOf* verbanden, terwijl één van de twee volstaat en de andere door de reasoner kan afgeleid worden. Er is besloten dit voorlopig nog zo te laten, aangezien de reasoner niet werkt en deze verbanden bijgevolg niet afgeleid kunnen worden. Ook zou het in een latere versie van Protégé mogelijk moeten zijn automatisch redundante verbanden uit het model te halen, zodat het op dit ogenblik geen zin heeft om manueel alle redundantie weg te filteren.

Een laatste voorbeeld wordt getoond om het goed functioneren van het model aan te tonen. Er werd een instantie aangemaakt in de klasse *SpecifiedResource* die enkel de relevante objecteigenschappen bevat.



**Figuur 32** Test-instantie database

Wanneer de reasoner geactiveerd is, wordt de instantie geklasseerd in correcte nieuwe klassen.



**Figuur 33 Testinstantie database, met reasoning**

#### 4.2.3 Mogelijke uitbreidingen van het model

De resulterende ontologie kan een krachtig instrument vormen indien het op een correcte wijze gebruikt wordt. Het is echter beperkt in dekking. Enkel de organisatorische eenheden en gegevensbronnen zijn uitgewerkt.

Langs de andere kant zijn er tal van uitbreidingen mogelijk op het model en vaak is een eerste stap al genomen. Zo kan men documenten uitgebreider beschrijven met de Dublin Core elementen. Dankzij de opname van het Technical Reference Model (TRM) is het mogelijk de technische aspecten van gegevensbronnen uit te werken, maar ook los van gegevensbronnen kan het technologisch kader van de Vlaamse Overheid in een nieuwe ontologie uitgewerkt worden. Dit zou bijgevolg een bredere dekking geven in het Enterprise Architecture Framework (EAF) op het technologische aspect.

Ook de bedrijfsaspecten van het EAF kunnen verder uitgewerkt worden. Door instanties te ontwerpen van de klassen Act, Budget en Mission en deze instanties te verbinden met andere entiteiten, krijgt het model meer waarde.

Het personeelsbestand verder uitwerken is misschien de actie met het meeste directe nut. Zo kan men met rules (op dit moment nog niet ondersteund in Protégé 4.0 alpha) het volgende verband leggen:

```
isEmployeeOf(?x, ?y), isManagerOf(?y, ?z) -> worksIn(?x, ?z)
```



In mentaal wordt dit: als Persoon x werkt voor Persoon y en als Persoon y het departement/team/afdeling z beheert, dan werkt Persoon x binnen z. Aangezien *isEmployeeOf* een transitieve eigenschap is, zal deze regel niet alleen het departement geven van de rechtstreekse werkgever, maar ook van de hele hiërarchie (iemand werkt voor een team, voor een afdeling, voor een agentschap én voor de Vlaamse Overheid). Verder kan het interessante voordelen opleveren om profielen en competenties te koppelen aan werknemers. Bij het vervolgens uitwerken van een projecten-ontologie kan dan gezocht worden naar alle werknemers die voldoen aan een aantal eigenschappen, nodig voor een nieuw project. Zo kan men zorgen voor “de juiste persoon op de juiste plaats”. Een eenvoudig voorbeeld zou kunnen zijn dat een projectmanager op zoek is naar een persoon die kan programmeren. De werknemer die een project achter de rug heeft waar hij in .NET een programma heeft geschreven, kan dan automatisch aanbevolen worden, omdat de ontologie “weet” dat iemand die .NET kent, ook kan programmeren.

Binnen tien jaar zou het model in het ideale geval “volledig” moeten zijn. Dit houdt in dat elk trefwoord dat nu in een taxonomie vervat zit, vervangen wordt door een klasse in een uitgewerkte ontologie, maar ook dat elk mogelijk concept aanwezig in de Vlaamse Overheid formeel bepaald is. Of het zover zal komen is helemaal niet duidelijk en enkel de tijd zal uitwijzen of dit model zijn waarde kan bewijzen.

### **4.3 Aanbevelingen Vlaamse Overheid**

Het in gebruik nemen van nieuwe technologie is nooit een eenvoudige opdracht. De toepassing van ontologieën in een organisatie kan weerstand ondervinden, omdat het nieuw is, omdat het meer werk betekent of omdat het nut ervan niet wordt ingezien. Een ander probleem is dat de behoeften aan nieuwe technologie anders liggen op de verschillende organisatorische niveaus. Een paar bedenkingen worden uiteengezet voor het geval de Vlaamse Overheid zou willen verder gaan in deze ontwikkeling.

#### 4.3.1 Incrementele ontwikkeling en feedback

Binnen een organisatie is vaak een kloof aanwezig tussen het top management en het midden- en lager management. Projecten die volgens het lager management voldoende nut hebben, worden niet aanvaard door het top-management omdat de voordelen niet duidelijk genoeg zijn. De oorzaak hiervan ligt vaak aan de verschillende standpunten waaruit een voordeel bekeken kan worden: wat bv. een tijdswinst betekent voor het beheer, is extra werk voor de ICT (ook hier komt weer het belang van EAFs naar boven).

Het invoeren van Semantisch Web technologie zal dus best gebeuren met kleine, incrementele stappen. Een radicale, breakthrough implementatie heeft een aantal voordelen, zoals het vertrekken van een structuur, een grondige globale analyse kunnen uitvoeren en in één keer alle data en applicaties kunnen omvormen. Het gevaar is echter dat dit in de eerste plaats nooit aanvaard zal worden door het top-management, maar ook dat er weerstand zal ontstaan in alle lagen van de organisatie.

Daarom wordt aangeraden om bij toekomstige projecten te onderzoeken of het gebruik van ontologieën vruchten kan afwerpen voor dat project. Met kleine stappen vooruit kunnen de voordelen van Semantisch Web technieken aangetoond worden. Over de integratie met andere ontologieën hoeft men zich pas later zorgen te maken. Het is mogelijk pas in een later stadium een ontologie-infrastructuur op te zetten, dankzij het gedecentraliseerde en open karakter van Semantisch Web technologie. Op deze manier bekomt men concrete resultaten.

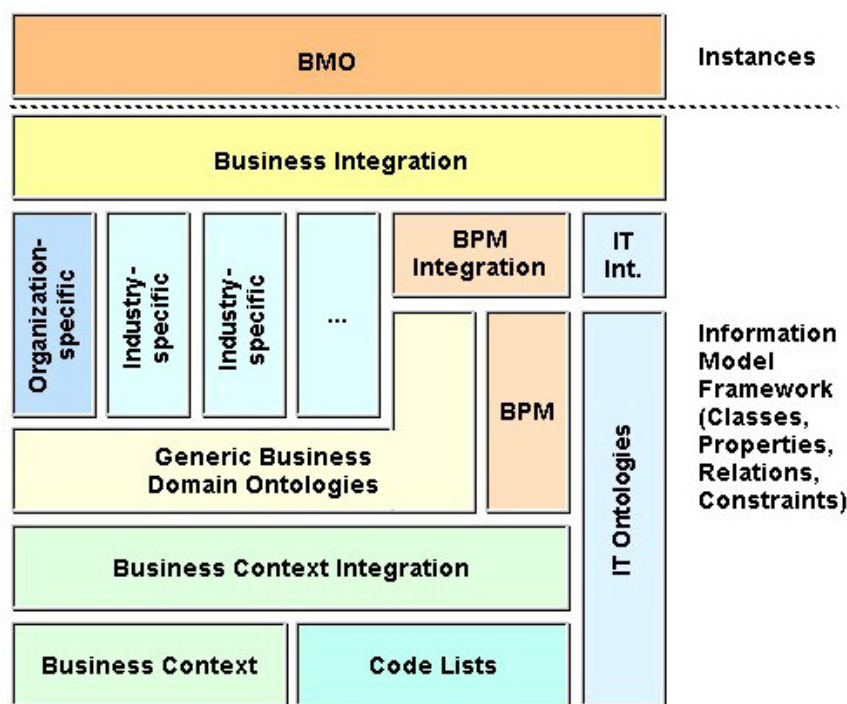
Het Semantisch Web is niet alleen een verzameling technologieën en use cases, maar ook een manier van denken, een filosofie. Dit is zeer moeilijk te ontwikkelen in een bedrijf en lukt al zeker niet met één grootschalig project. Het introduceren van deze manier van denken zal gebeuren met vallen en opstaan, met trial-and-error. Hier komt het belang van feedback naar boven. Het succes van de implementatie van Semantisch Web technieken zal afhangen van de mogelijkheden om feedback te geven en deze effectief te gebruiken. Pilotprojecten zullen niet alleen een eerste stap zetten om de mogelijkheden van Semantisch Web technieken aan te tonen, maar dienen evenzeer om een feedback-loop in gang te zetten. Er zal een continue wisselwerking moeten ontstaan om feedback te verzamelen, deze te gaan toepassen en opnieuw feedback te krijgen.

### 4.3.2 Communities of Interest

Indien een aantal pilootprojecten tot een goed einde worden gebracht, kunnen ontologieën dieper uitgewerkt worden. Dankzij de mogelijkheid ontologieën modulair op te bouwen, is het nuttig om Communities of Interest (CoI) samen te stellen, die elk hun eigen ontologie verder uitdiepen. Zo heeft een boekhoudkundige ontologie niet veel gemeen met een HR ontologie. Wel is er nood aan een centraal aanspreekpunt, die generieke ontologieën beheerst (zoals een ontologie met tijdsaspecten) en ervoor waakt dat de ontologieën van verschillende CoI's elkaar niet tegenspreken.

### 4.3.3 Architectuur

Op het moment dat een aantal ontologieën in gebruik zijn, kan het de moeite lonen een centrale architectuur op te stellen. De Business Management Ontology (BMO) is hier een mooi voorbeeld van. Het biedt een manier om de “semantische” kloof te overbruggen tussen de IT en de business, door een mapping te voorzien tussen bedrijfs- en IT-concepten.



**Figuur 34 BMO architectuur (bron: Jenz & Partner GmbH 2004)**

De BMO architectuur laat toe generische bedrijfsontologieën te ontwikkelen, waardoor hergebruik op een flexibele wijze mogelijk wordt.

#### 4.3.4 Andere mogelijkheden voor Semantisch Web technieken binnen de Vlaamse Overheid

Wat zijn nu enkele concrete voorstellen voor het gebruik van Semantisch Web technieken binnen de Vlaamse Overheid, naast het ontwikkelde model? Hieronder worden enkele ideeën kort toegelicht.

##### 4.3.4.1 *E-government*

*E-government is een manier om de kwaliteit van de openbare dienstverlening aan burgers en ondernemingen te verbeteren door gebruik te maken van de opportuniteiten van moderne technologie, internet en nieuwe media (Informatie- en communicatietechnologie).*<sup>6</sup>

Net als bij het Webmetadata project van de Nederlandse Overheid (cf. §4.2.2.1.2) kunnen ontologieën bijdragen tot een verbetering van de toegang tot online informatie. Dit kan vele vormen aannemen. Zo kan het Protégé-model, dat nu enkel interne informatiebronnen beschrijft, uitgebreid worden naar de informatiebronnen die gebruikt worden door burgers en bedrijven. Zoekrobots zoals Google kunnen daarna gebruik maken van de beschikbare metadata.

Ook is het mogelijk met ontologieën verschillende gegevensbronnen en databanken te aggregeren, waardoor de burger één centraal aanspreekpunt bekommt waar hij alle persoonlijke informatie kan terugvinden. Kortom, ontologieën kunnen een hulp zijn bij het zoeken naar en beheren van nuttige informatie.

##### 4.3.4.2 *Interne communicatie*

Communicatie in de meeste organisaties beperkt zich tot enkele kanalen. Er zijn twee soorten communicatiekanalen, namelijk *push* en *pull*. Bij push-kanalen ontvangt men informatie zonder daar expliciet om gevraagd te hebben, zoals e-mail en telefoon, maar ook een toevallig gesprek met een collega in de wandelgangen.

---

6

<http://www.belgium.be/eportal/application?languageRedirected=yes&navId=5449&origin=aboutBelgium.jsp&pageid=indexPage&languageRedirected=yes&event=bea.portal.framework.internal.refresh>

Bij pull-kanalen ligt het initiatief bij de gebruiker zelf, hij moet zelf actief naar informatie vragen. Denk hierbij aan websites (ook de portalsite van de organisatie zelf) en discussiefora.

Nu wordt e-mail door veel managers gezien als een ware plaag, omdat ze teveel informatie krijgen waar ze niet om hebben gevraagd en die op het verkeerde tijdstip aankomt. Er zijn echter alternatieve communicatiekanalen en één daarvan is *newsfeeds* die gebruik maken van RSS (Really Simple Syndication)- of Atomtechnologie. Newsfeeds is een communicatiekanaal dat tracht het beste samen te brengen van beide werelden en eigenlijk is deze technologie gebaseerd op Semantisch Web technieken. Hoewel newsfeeds al een zekere populariteit kennen in het privé-gebruik, lijkt de opmars van deze technologie in het bedrijfsleven beperkt. Onderzoeken wat de mogelijkheden zijn van newsfeeds binnen de Vlaamse Overheid zou kunnen resulteren in een krachtig communicatiekanaal.

#### 4.3.4.3 SOA-projecten

In een Service Oriented Architecture worden de diensten (services, de functionaliteit die een systeem aanbiedt) formeel gescheiden van de gebruikers (systemen die deze functionaliteiten nodig hebben). Deze scheiding is mogelijk door het mechanisme van dienstencontracten, samen met een mechanisme dat het mogelijk maakt diensten te publiceren en op te zoeken.

Bij nieuwe SOA-projecten kan het lonend zijn te voorzien in een laag semantische “vernis”. Hierdoor zal het zoeken naar diensten vlotter afgehandeld kunnen worden. De diensten zullen beter op elkaar afgestemd zijn en elkaar beter kunnen begrijpen. Het onderhoud van de diensten zal ook vergemakkelijkt worden en de kwaliteit zal verhogen.

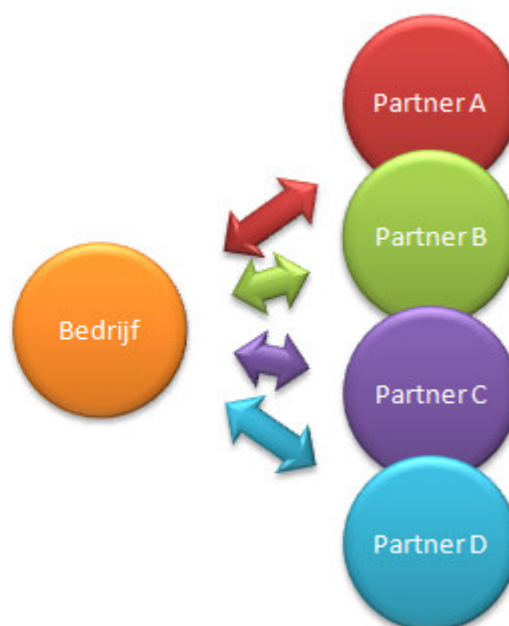
Uiteindelijk is het de bedoeling in een SOA-omgeving te werken met volwaardige Semantic (Web) Services, diensten die autonoom en automatisch met elkaar kunnen samenwerken, wat mogelijk wordt door een machineleesbare semantiek.

#### 4.3.4.4 *Ontologieën als dienst*

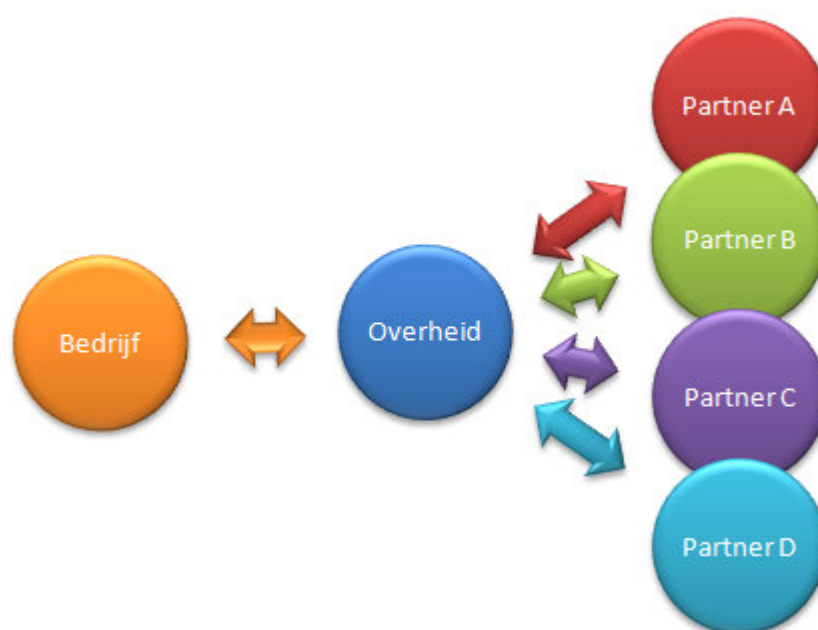
Als laatste bestaat de mogelijkheid Semantisch Web technieken aan te bieden als diensten aan de burger.

In de eerste plaats zou informatie afkomstig van de Vlaamse Overheid, betrouwbare informatie worden in het Web of Trust (cf. §2.7.11). Met andere woorden: alle informatie door de Vlaamse Overheid gepubliceerd is te vertrouwen. De aanwezigheid van een aantal te vertrouwen bronnen op het Semantisch Web zal immers een belangrijke rol spelen. Het is onvermijdelijk dat bepaalde bronnen elkaar zullen tegenspreken en een zekere graad van *vertrouwen* is nodig voor elke bron zodat de reasoners hiermee om kunnen gaan. De rol van de Vlaamse Overheid zou dus kunnen zijn om één van de betrouwbare bronnen te worden in de chaotische online wereld.

Een andere dienst zou kunnen zijn generieke ontologieën aan te bieden naar bedrijven toe. Dit zou een stap vooruit zijn in de B2B communicatie (cf. §2.3.1.2) doordat bedrijven deze generieke concepten als basis zullen hebben om op voort te bouwen. Indien bedrijven reeds eigen ontologieën ontwikkeld hebben, kunnen deze gemapt worden naar de generieke ontologieën van de Vlaamse Overheid. Hierdoor is er slechts één mapping nodig, daar waar zonder deze generieke ontologieën voor elke partner een mapping vereist zou zijn.



**Figuur 35 Communicatie tussen bedrijven met ontologieën**



**Figuur 36 Communicatie tussen bedrijven met ontologieën en het gebruik van een generieke ontologie van de Vlaamse Overheid**

Concreet kan de Vlaamse Overheid een ontologie voorzien met termen zoals klant, bedrijf, werkgever, werknemer, product, dienst, kwaliteit, openbare diensten en overheid.

#### 4.3.5 Europa en de rest van de wereld

Het Semantisch Web zal niet van de ene dag op de andere ontstaan. Er zijn nog veel hindernissen te overbruggen en van sommige Semantisch Web projecten zullen de resultaten teleurstellend zijn. Samenwerking is hier de sleutel. Door ervaringen, bedenkingen en zelfs ontologieën te delen met elkaar, zullen overheden een snellere en betere adoptie kennen van deze nieuwe technologie.

In het Verenigd Koninkrijk, Ierland, Australië, Finland en Denemarken is het gebruik van Dublin Core metadata door de overheid voorgeschreven. Door te gaan polsen naar de resultaten hiervan en een nauwe samenwerking, kan de Vlaamse Overheid heel wat tijd en geld besparen.

Ook wordt aangeraden contact op te nemen met het Semantic Technology Institute (STI) International. STI International is een non-profit organisatie opgestart in april 2007 en staat onder leiding van Dipl. Ing. Wahler (CEO) en Prof. Dr. Fensel (President). De website is te vinden op [www.sti2.org](http://www.sti2.org).

De missie van STI International is

*[...] to establish semantics as a core pillar of modern computer science.*

Dit willen ze bereiken door diensten aan te bieden om het onderzoek, de opleiding en de commercialisering van semantische technologieën te ondersteunen.



## **Algemeen besluit: kritische reflectie en overgang naar het Semantisch Web**

Het Semantisch Web is een veelbelovende en opwindende technologie. Maar het is ook meer dan dat, er zit een mooi uitgewerkte filosofie en een doordachte visie achter. Door gegevens te voorzien van metadata, kunnen computers op een intelligente wijze omgaan met deze gegevens, door deze gegevens te “begrijpen” en nieuwe verbanden af te leiden. Dat we spannende tijden tegemoet gaan, lijkt onvermijdelijk.

Doch is het aangeraden met twee voeten op de grond te blijven door de mogelijkheden, maar ook de beperking van het Semantisch Web te begrijpen en zich niet te laten meeslepen door een te groot enthousiasme (en een tweede bubble-burst te vermijden). Het Semantisch Web gaat niet plots zorgen voor wereldvrede, maar zal vooral in het begin bescheiden gebruikt moeten worden.

In de eerste plaats is het van belang in te zien dat het Semantisch Web slechts een technologie, een hulpmiddel is. Het Semantisch Web op zich doet *niets*.

Dit moet toch even benadrukt worden, zodat veel onterechte verwachtingen vermeden worden. De technologieën die het Semantisch Web ondersteunen, bieden een kans data te voorzien van semantiek. Hierdoor is deze data niet meer afhankelijk van een bepaalde applicatie, maar kan ze gedeeld worden over verschillende heterogene applicaties heen, zoals agents en webservices. Maar het is slechts wanneer deze data gebruikt en verwerkt worden door applicaties, agents of webservices, dat de mens of computer er een nut aan heeft. Dit is er meteen ook de oorzaak van dat op dit ogenblik het Semantisch Web nog maar marginaal is doorgebroken. Vooraleer applicaties gebruik kunnen maken van het Semantisch Web, moeten er gegevens aanwezig zijn die van semantiek voorzien zijn. Maar deze gegevens van semantiek voorzien terwijl geen enkele bestaande applicatie deze semantiek kan verwerken, heeft ook geen zin. Dit is dus het klassieke kip-en-ei probleem waar ook het Internet zelf mee te kampen had.

Eens een zekere kritieke massa bereikt is, zal het Semantisch Web hoogstwaarschijnlijk kunnen doorbreken. Het zal evenwel slechts succesvol zijn op het ogenblik dat niemand meer praat over het Semantisch Web, maar wanneer het huidige Web is uitgebreid en dit nieuwe Web voor iedereen vanzelfsprekend is. Deze doorbraak zou wel eens binnenkort kunnen plaatsvinden.

De nodige infrastructuur is er en een aantal start-ups en grotere bedrijven zijn bezig met de ontwikkeling van Semantisch Web applicaties. Om bij het grote publiek door te breken is er nood aan een killer-app, een simpele applicatie waarvan het nut duidelijk is. Langs de andere kant kan het Semantisch Web voor bedrijven de oplossing zijn om van hun “Toren van Babel” probleem af te geraken. Een standaard protocol om informatie door te geven is op zich al een enorm voordeel.

Om de slaagkansen van het Semantisch Web te verhogen, moet ook aandacht besteed worden aan een zekere mate van opleiding en sensibilisering. Een Web dat leert van zijn gebruikers en een oneindige hoeveelheid kennis beheert is duidelijk waar we heen gaan. De grote uitdaging zal er echter in bestaan gevestigde gebruikers ervan te overtuigen dat een nieuwe vorm van communicatie, samenwerking en kennisbeheer nodig is. Want het zijn de gebruikers die zullen uitmaken of het Semantisch Web een succes is of niet. Daarbij spelen de gebruikers een primordiale rol in het hele project. Het belang van de gemeenschap en van sociale netwerken mag niet onderschat worden. De gemeenschap moet overtuigd worden semantiek toe te voegen aan informatie. Projecten zoals Freebase zouden op eenzelfde manier kunnen evolueren als Wikipedia. Samenwerking is belangrijk voor het Semantisch Web.

Al bij al opent de Semantisch Web technologie de deuren voor vele toepassingen. Door gegevens te verrijken met semantiek, kan er veel meer gebeuren met deze gegevens. Wat al deze mogelijkheden zijn, is zelfs nu nog niet duidelijk. De gebruikers zullen de technologie op de meest nuttige wijze aanwenden om hun noden te bevredigen. Het zal enorm boeiend zijn de ontwikkelingen te volgen.

Ten slotte kunnen cynici zich afvragen waarom het deze keer wél zou lukken, wanneer expert systemen en alle vorige hypes “gefaald” hebben. Het Semantisch Web heeft wel een aantal voordelen tegenover alle vorige hypes. Het is gedecentraliseerd. Het is gebaseerd op open standaarden. En misschien is de wereld er deze keer wel klaar voor. Zelfs indien het Semantisch Web nooit doorbreekt en bloeit, wat uiteraard nog altijd mogelijk is, dan nog zullen er onvermijdelijk belangrijke concepten zijn overgebleven die uitgetest zijn en waarmee geëxperimenteerd werd. In het meest extreme geval zal het Semantisch Web dan een vingeroefening zijn geweest voor waar het Web ons in de toekomst zal brengen.



# Bijlagen

### **Bijlage 1: inhoud CD-rom**

De CD-rom welke hoort bij deze eindverhandeling bevat de Protégé files van het nieuwe model, met een beperkt aantal instanties. Daarnaast bevat de CD-rom ook de DC-GAP en de FEA-RMO modellen.

***Bijlage 2: EA raamwerken***

Zachman raamwerk (Bron: zifa.com)

**PHONE** (810) 231-0531  
**FAX:** (810) 231-8631  
**www.zifa.com**  
10895 Lakepointe Drive  
Plymouth, MI 48169



**ZIFA**  
Zachman Institute  
for Framework Assessment

# ENTERPRISE ARCHITECTURE: A FRAMEWORK™

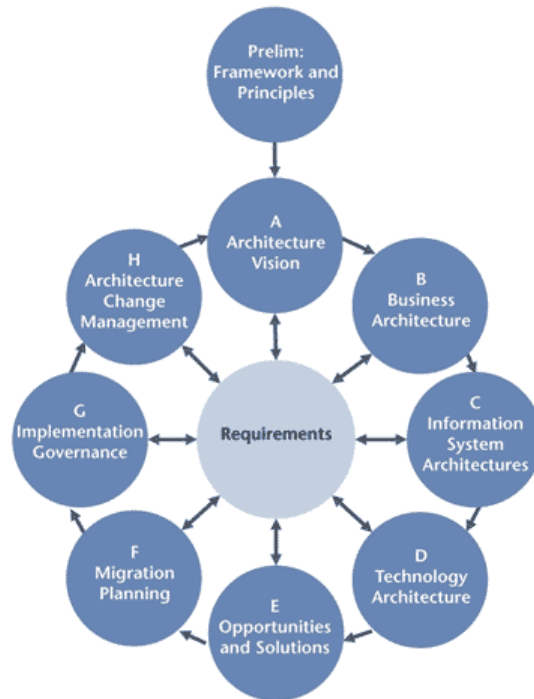
	WHAT		HOW		WHERE		WHO		WHEN		WHY	
	DATA	FUNCTION	NETWORK	PEOPLE	TIME	MOTIVATION	SCOPE (contextual)	BUSINESS MODEL (conceptual)	SYSTEM MODEL (logical)	TECHNOLOGY MODEL (physical)	DETAILED REPRESENTATIONS (out-of-context)	FUNCTIONING ENTERPRISE
Planner	List of Things Important to the Business Entity = Class of Business Thing	List of Processes the Business Performs Process = Class of Business Process	List of Locations in Which the Business Operates Node = Major Business Location	List of Organizations Implicated in the Business People = Major Organizational Unit	List of Events/Cycles Significant to the Business Time = Major Business Event/Cycle	List of Business Goals/Strategies Ends/Means = Major Business Goal/Strategy						Planner
Owner	e.g., Semantic Model Entity = Business Entity Relationship = Business Relationship	e.g., Business Process Model Process = Business Process IO = Business Resource	e.g., Business Logistics System Node = Business Location Link = Business Linkage	e.g., Work Flow Model People = Organization Unit Work = Work Product	e.g., Master Schedule Time = Business Event Cycle = Business Cycle	e.g., Business Plan End = Business Objective Means = Business Strategy	BUSINESS MODEL (conceptual) Owner					
Designer	e.g., Logical Data Model Entity = Data Entity Relationship = Data Relationship	e.g., Application Architecture Process = Application Function IO = User Views	e.g., Distributed System Architecture Node = IS Function (Process, Storage, etc.) Link = Line Characteristics	e.g., Human Interface Architecture People = Role Work = Deliverable	e.g., Processing Structure Time = System Event Cycle = Processing Cycle	e.g., Business Rule Model End = Structural Assertion Means = Action Assertion	SYSTEM MODEL (logical) Designer					
Builder	e.g., Physical Data Model Entity = Segment/Table/etc. Relationship = Family/Key/etc.	e.g., System Design Process = Complete Function IO = Data Elements/bits	e.g., Technology Architecture Node = IBM System Software Link = Line Specifications	e.g., Presentation Architecture People = User Work = Screen Formats	e.g., Control Structure Time = Execute Cycle = Component Cycle	e.g., Role Design End = Condition Means = Action	TECHNOLOGY MODEL (physical) Builder					
Subcontractor	e.g., Data Definition Entity = Field Relationship = Address	e.g., Program Process = Language-Statement IO = Control Block	e.g., Network Architecture Node = Address Link = Protocol	e.g., Security Architecture People = Identity Work = Job	e.g., Timing Definition Time = Interrupt Cycle = Machine Cycle	e.g., Role Specification End = Sub-condition Means = Step	DETAILED REPRESENTATIONS (out-of-context) Subcontractor					
	e.g.: DATA	e.g.: FUNCTION	e.g.: NETWORK	e.g.: ORGANIZATION	e.g.: SCHEDULE	e.g.: STRATEGY	FUNCTIONING ENTERPRISE					

© John A. Zachman

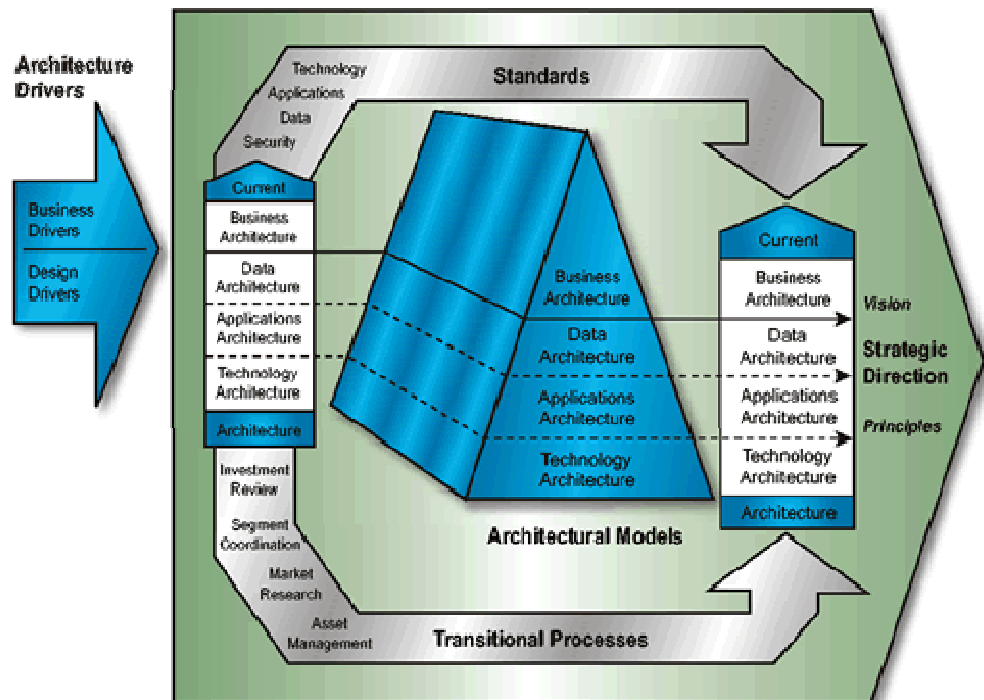
## THE ZACHMAN FRAMEWORK FOR ENTERPRISE ARCHITECTURE

## TOGAF raamwerk (Bron: opengroup.org)

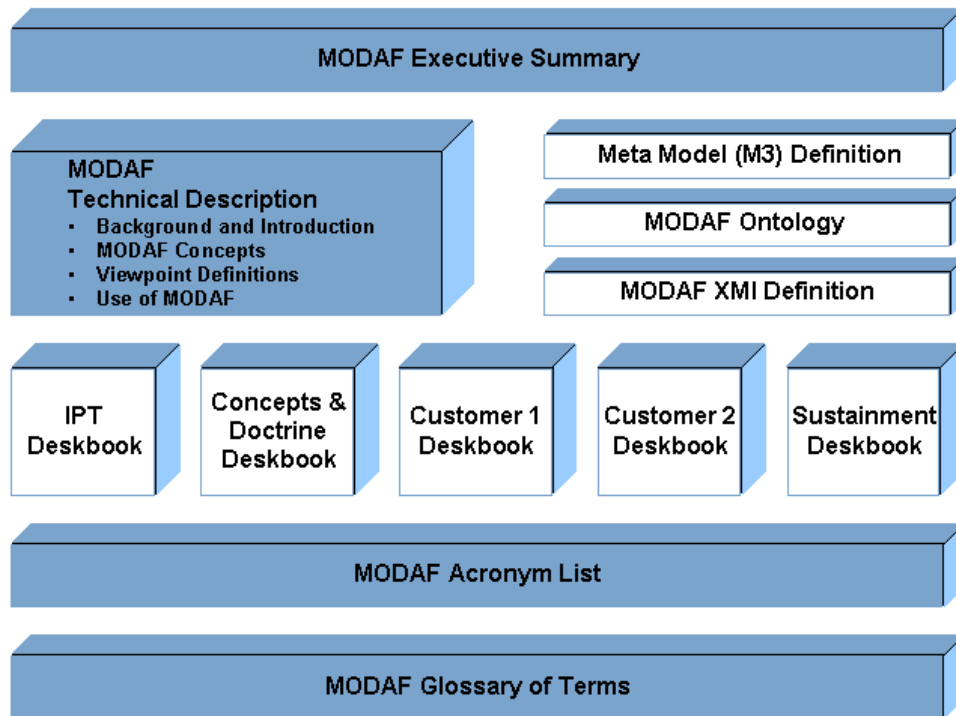
Fig.: TOGAF Architecture Development Method (ADM)



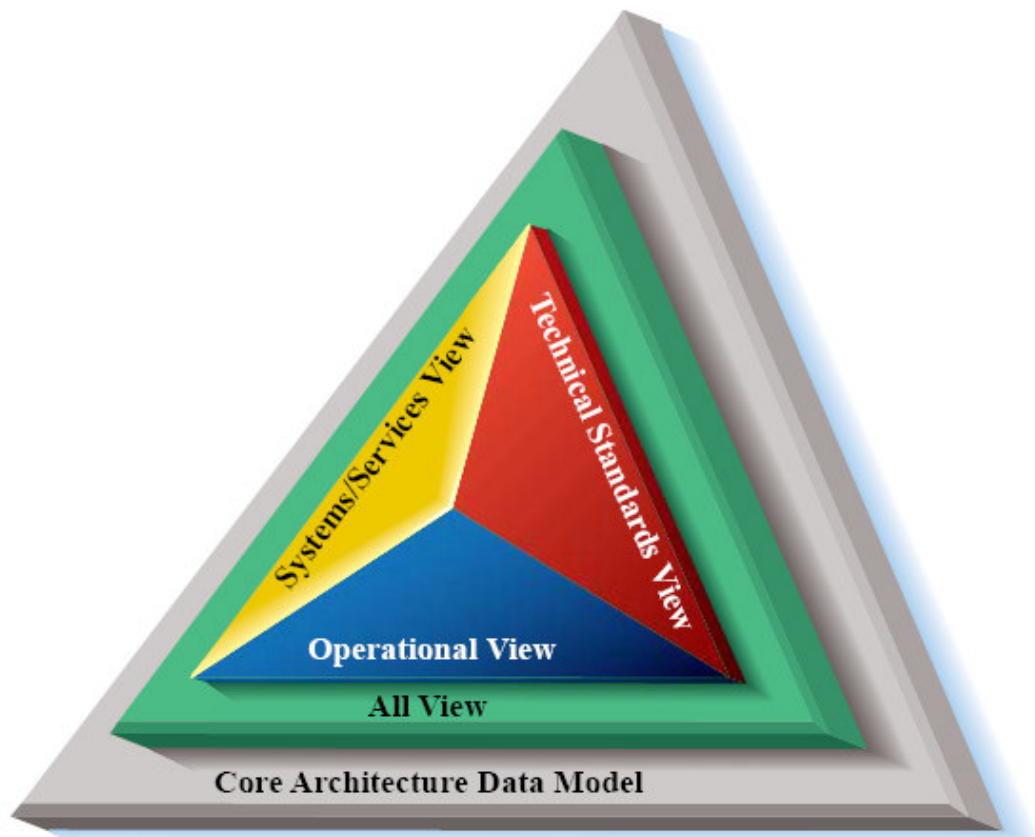
## FEAF raamwerk (Bron: opengroup.org)



**MoDAF raamwerk** (Bron: modaf.org.uk)



**DoDAF raamwerk** (Bron: DoD Architecture Framework Version 1.5)





### **Bijlage 3: Organisatie van de Vlaamse Overheid**

#### 1. Diensten voor het Algemeen Regeringsbeleid (DAR)

- Departement Diensten voor het Algemeen Regeringsbeleid
- Studiedienst van de Vlaamse Regering - *IVA zonder rp*
- Interne Audit van de Vlaamse Administratie - *IVA sui generis*
- Agentschap voor Geografische Informatie Vlaanderen (AGIV) - *EVA publiek*
- vzw De Rand - *EVA privaat*

#### 2. Bestuurszaken (BZ)

- Departement Bestuurszaken
- Agentschap voor Facilitair Management - *IVA zonder rp*
- Agentschap voor Overheidspersoneel - *IVA zonder rp*
- Agentschap voor Binnenlands Bestuur - *IVA zonder rp*
- Vlaams Agentschap voor Recruitering en Selectie (Jobpunt Vlaanderen) - *EVA privaat*

#### 3. Financiën en Begroting (FB)

- Departement Financiën en Begroting
- Centrale Accounting - *IVA zonder rp*
- Vlaamse Belastingdienst - *IVA zonder rp*

#### 4. Internationaal Vlaanderen (IV)

- Departement Internationaal Vlaanderen
- Vlaams Agentschap voor Internationale Samenwerking (VAIS) - *IVA zonder rp*
- Toerisme Vlaanderen - *IVA met rp*
- Vlaams Agentschap voor Internationaal Ondernemen - Flanders Investment & Trade (FIT) - *EVA publiek*

#### 5. Economie, Wetenschap en Innovatie (EWI)

- Departement Economie, Wetenschap en Innovatie
- Agentschap Economie - *IVA zonder rp*
- Vlaams Agentschap Ondernemen (VLAO) - *IVA met rp*

- Instituut voor Innovatie door Wetenschap en Technologie (IWT) - *IVA met rp*
- NV Participatiemaatschappij Vlaanderen (PMV) - *EVA privaat*
- NV Vlaamse Participatiemaatschappij (VPM) - *EVA privaat*
- NV Limburgse Reconvertiemaatschappij (LRM) - *EVA privaat*

#### 6. Onderwijs en Vorming (OV)

- Departement Onderwijs en Vorming
- Agentschap voor Onderwijsdiensten - *IVA zonder rp*
- Onderwijsdienstencentrum voor Hoger Onderwijs en Volwassenenonderwijs - *IVA zonder rp*
- Agentschap voor Onderwijscommunicatie - *IVA zonder rp*
- Agentschap voor Infrastructuur in het Onderwijs (AGION) - *IVA met rp*

#### 7. Welzijn, Volksgezondheid en Gezin (WVG)

- Departement Welzijn, Volksgezondheid en Gezin
- Zorg en Gezondheid - *IVA zonder rp*
- Jongerenwelzijn - *IVA zonder rp*
- Inspectie Welzijn en Volksgezondheid - *IVA zonder rp*
- Kind en Gezin - *IVA met rp*
- Vlaams Agentschap voor Personen met een Handicap (VAPH) - *IVA met rp*
- Vlaams Zorgfonds - *IVA met rp*
- Fonds Jongerenwelzijn - *IVA met rp*
- Vlaams Infrastructuurfonds voor Persoonsgebonden Aangelegenheden (VIPA) - *IVA met rp*
- Openbaar Psychiatrisch Zorgcentrum Geel (OPZ Geel) - *EVA publiek*
- Openbaar Psychiatrisch Zorgcentrum Rekem (OPZ Rekem) - *EVA publiek*

#### 8. Cultuur, Jeugd, Sport en Media (CJSM)

- Departement Cultuur, Jeugd, Sport en Media
- Kunsten en Erfgoed - *IVA zonder rp*
- Sociaal-Cultureel Werk voor Jeugd en Volwassenen - *IVA zonder rp*

- Agentschap ter Bevordering van de Lichamelijke Ontwikkeling, de Sport en de Openluchtrecreatie (Blosso) - *IVA met rp*
- Vlaamse Regulator voor de Media (VRM) - *EVA publiek*

#### 9. Werk en Sociale Economie (WSE)

- Departement Werk en Sociale Economie
- Vlaams Subsidieagentschap voor Werk en Sociale Economie - *IVA zonder rp*
- Vlaamse dienst voor Arbeidsbemiddeling en Beroepsopleiding (VDAB) - *EVA publiek*
- Vlaams Agentschap voor Ondernemersvorming - Syntra Vlaanderen - *EVA publiek*
- NV met sociaal oogmerk Werkholding - *EVA privaat*

#### 10. Landbouw en Visserij(LV)

- Departement Landbouw en Visserij
- Agentschap voor Landbouw en Visserij - *IVA zonder rp*
- Instituut voor Landbouw- en Visserijonderzoek (ILVO) - *IVA zonder rp*
- vzw Vlaams Centrum voor Agro- en Visserijmarketing (VLAM) - *EVA privaat*

#### 11. Leefmilieu, Natuur en Energie (LNE)

- Departement Leefmilieu, Natuur en Energie
- Agentschap voor Natuur en Bos - *IVA zonder rp*
- Instituut voor Natuur- en Bosonderzoek (INBO) - *IVA zonder rp*
- Vlaams Energieagentschap - *IVA zonder rp*
- Vlaamse Milieumaatschappij (VMM) - *IVA met rp*
- Openbare Vlaamse Afvalstoffenmaatschappij (OVAM) - *IVA met rp*
- Vlaamse Landmaatschappij (VLM) - *EVA publiek*
- Vlaamse Reguleringsinstantie voor de Elektriciteits- en Gasmarkt (VREG) - *EVA publiek*

#### 12. Mobiliteit en Openbare Werken (MOW)

- Departement Mobiliteit en Openbare Werken
- Agentschap Wegen en Verkeer - *IVA zonder rp*
- Agentschap voor Maritieme Dienstverlening en Kust - *IVA zonder rp*

- Vlaamse Vervoermaatschappij De Lijn (VVM De Lijn) - *EVA publiek*
- Waterwegen en Zeekanaal - *EVA publiek*
- De Scheepvaart - *EVA publiek*

### 13. Ruimtelijke Ordening, Woonbeleid en Onroerend erfgoed (RWO)

- Departement Ruimtelijke Ordening, Woonbeleid en Onroerend Erfgoed
- Wonen-Vlaanderen - *IVA zonder rp*
- Ruimtelijke Ordening Vlaanderen (RO-Vlaanderen) - *IVA zonder rp*
- Vlaams Instituut voor het Onroerend Erfgoed - *IVA zonder rp*
- Inspectie RWO - *IVA zonder rp*
- Vlaamse Maatschappij voor Sociaal Wonen - *EVA publiek*

Een aantal bestaande Vlaamse Openbare Instellingen werden niet meegenomen in de hervorming Beter Bestuurlijk Beleid. Toch behoren ze ook tot de Vlaamse administratie:

- Vlaamse Instelling voor Technologisch Onderzoek (VITO)
- Fonds voor Wetenschappelijk Onderzoek (FWO)
- Vlaamse Opera
- Vlaamse Radio- en Televisieomroep (VRT)
- Het Gemeenschapsonderwijs
- Universitair Ziekenhuis Gent (UZGent)
- Vlaamse Maatschappij voor Watervoorziening (VMW)

## Bijlage 4: Screenshots model

The screenshot displays the Protege ontology editor interface. The main window shows the 'Ontology Annotations' tab for the file 'instances.owl'. The annotations are as follows:

- contributor:** "mr. Van Herreweghe"
- versionInfo:** "versie 0.1"
- creator:** "Philippe Luickx"
- contributor:** "prof. dr. Lemahieu"
- description:** "Deze ontlogie voegt geen nieuwe classes of properties toe. Ze bevat enkel de instanties."
- date:** "16/04/2008"
- label:** "Ontlogie voor de implementatie van instanties"

Below the annotations, the 'Imported Ontologies' section lists the following URIs:

- Direct imports:
  - <http://www.vlaanderen.be/ontologie/business.owl>
  - <http://www.vlaanderen.be/ontologie/technology.owl>
- Indirect imports:
  - <http://www.vlaanderen.be/ontologie/business.owl>
  - <http://www.vlaanderen.be/ontologie/descriptive.owl>
  - <http://www.vlaanderen.be/ontologie/technology.owl>

On the right side, the 'Ontology Metrics' panel provides a summary of the ontology's structure:

Metrics	
Class count	140
Object property count	52
Data property count	18
Individual count	136
DL expressivity	SHO(F)(D)

Class axioms	
SubClass axioms count	135
Equivalent classes axioms count	17
Disjoint classes axioms count	3
GCI count	0
Hidden GCI Count	6

Object property axioms	
Sub object property axioms count	25
Equivalent object properties axioms count	0
Inverse object properties axioms count	15
Disjoint object properties axioms count	0
Functional object property axioms count	1
Inverse functional object property axioms count	0
Transitive object property axioms count	7
Symmetric object property axioms count	1
Anti-symmetric object property axioms count	0
Reflexive object property axioms count	0
Irreflexive object property axioms count	0

Data property axioms	
Sub data property axioms count	12
Equivalent data properties axioms count	0
Disjoint data properties axioms count	0
Functional data property axioms count	0

Individual axioms	
Class assertion axioms count	136
Object property assertion axioms count	143
Data property assertion axioms count	202
Negative object property assertion axioms count	0
Negative data property assertion axioms count	0
Same individual axioms count	0
Different individuals axioms count	0

Annotation axioms	
Entity annotation axioms count	195
Axiom annotation axioms count	0

http://www.vlaanderen.be/ontologie/instances.owl - (C:\Users\Philippe\Desktop\VO\instances.owl)

File Edit Reasoner Tools Refactor Tabs View Window Help

instances.owl http://www.vlaanderen.be/ontologie/instances.owl

Active Ontology Entities Classes Object Properties Data Properties Individuals OWL Viz DL Query

Asserted class hierarchy Inferred class hierarchy

Asserted Class Hierarchy: Cel

- Thing
  - Beleidsdomain
    - DescriptiveConcept
      - Category
      - Keyword
      - Taxonomy
    - EnterpriseConcept
      - Act
      - Budget
      - LegalStatute
      - Mission
      - OrganizationEntity
        - GovernmentBody
          - Afdeling
          - Beleidsdepartement
          - Cel = Team = Dienst
          - Dienst = Cel = Team
          - Directeurs
          - FederaleEntiteit
          - NIEntiteit
          - Overheid
          - ProjectNiveau
          - RadenEnAdviesorganen
          - Team = Cel = Dienst
          - VerzelfstandigdeAgentschap
          - VlaamseOpenbareInstelling
        - IndustryBody
          - Company
            - Vendor
        - StandardsOrganization
      - Party
      - Person
      - Product
      - Resource

Class Annotations Class Usage

Class Annotations: Cel

Annotations

comment  
"De kleinste entiteit van organisatie binnen de overheid. Is gelijk aan een team of dienst"

label  
"Cel"

Class Description: Cel

Equivalent classes

- Dienst

Superclasses

- GovernmentBody
- isPartOf some GovernmentBody

Inherited anonymous classes

- Team
- Dienst
- Team

http://www.vlaanderen.be/ontologie/instances.owl - (C:\Users\Philippe\Desktop\VO\instances.owl)

File Edit Reasoner Tools Refactor Tabs View Window Help

instances.owl http://www.vlaanderen.be/ontologie/instances.owl

Active Ontology Entities Classes Object Properties Data Properties Individuals OWL Viz DL Query

Asserted class hierarchy Inferred class hierarchy

Asserted Class Hierarchy: AmbtenaarVO

- Thing
  - Beleidsdomain
    - DescriptiveConcept
      - Category
      - Keyword
      - Taxonomy
    - EnterpriseConcept
      - Person
        - AmbtenaarVO
    - Product
    - Resource
    - ResourceProvenance
    - ServiceConcept
    - TechnologyConcept

Class Annotations Class Usage

Class Annotations: AmbtenaarVO

Annotations

comment  
"Is een persoon die bij de VO werkt. Wordt in de ontologie instances.owl gedefinieerd (aangezien er hier nog geen instance VlaamseOverheid aanwezig is)"

label  
"Ambtenaar van de Vlaamse Overheid"

Class Description: AmbtenaarVO

Equivalent classes

- Person and isEmployeeOf value VlaamseOverheid

Superclasses

- Person

http://www.vlaanderen.be/ontologie/instances.owl - (C:\Users\Philippe\Desktop\VO\instances.owl)

File Edit Reasoner Tools Refactor Tabs View Window Help

instances.owl http://www.vlaanderen.be/ontologie/instances.owl

Active Ontology Entities Classes Object Properties Data Properties Individuals OWL Viz DL Query

Asserted class hierarchy Inferred class hierarchy

Inferred class hierarchy: Website

- Thing
  - Beleidsdomein
  - DescriptiveConcept
  - EnterpriseConcept
  - Person
  - Product
  - Resource
    - ClientServerResource
    - DMSResource
    - DigitalResource
      - Database
      - DigDocument
      - Website
    - MainFrameResource
    - ManagedResource
    - NonDigitalResource
      - PapierenDocument
    - SpecifiedResource
      - Database
      - DigDocument
      - PapierenDocument
      - Website
    - UnmanagedResource
    - WebResource
  - ResourceProvenance
  - ServiceConcept
  - TechnologyConcept

Class Annotations Class Usage

Class Annotations: Website

Annotations

Class Description: Website

Equivalent classes

- SpecifiedResource
  - and hasMedium some SSP\_WebServer
  - and hasMedium only SSP\_WebServer

Superclasses

- DigitalResource
- SpecifiedResource

Inherited anonymous classes

- Resource
  - and hasMedium some (SSP\_DatabaseOrStorage or SSP\_DeliveryServices)

Individuals

Disjoint classes

- DigDocument, PapierenDocument, Database

http://www.vlaanderen.be/ontologie/instances.owl - (C:\Users\Philippe\Desktop\VO\instances.owl)

File Edit Reasoner Tools Refactor Tabs View Window Help

instances.owl http://www.vlaanderen.be/ontologie/instances.owl

Active Ontology Entities Classes Object Properties Data Properties Individuals OWL Viz DL Query

Asserted class hierarchy Inferred class hierarchy

Inferred class hierarchy: WebResource

- Thing
  - Beleidsdomein
  - DescriptiveConcept
  - EnterpriseConcept
  - Person
  - Product
  - Resource
    - ClientServerResource
    - DMSResource
    - DigitalResource
      - Database
      - DigDocument
      - Website
    - MainFrameResource
    - ManagedResource
    - NonDigitalResource
      - PapierenDocument
    - SpecifiedResource
      - Database
      - DigDocument
      - PapierenDocument
      - Website
    - UnmanagedResource
    - WebResource
  - ResourceProvenance
  - ServiceConcept
  - TechnologyConcept

Class Annotations Class Usage

Class Annotations: WebResource

Annotations

Class Description: WebResource

Equivalent classes

- Resource
  - and hasApplication value Webserver

Superclasses

- Resource

Inherited anonymous classes

- Test

Individuals

Disjoint classes

http://www.vlaanderen.be/ontologie/instances.owl - [C:\Users\Philippe\Desktop\VO\instances.owl]

File Edit Reasoner Tools Refactor Tabs View Window Help

instances.owl http://www.vlaanderen.be/ontologie/instances.owl

Active Ontology Entities Classes Object Properties Data Properties Individuals OMLViz DL Query

Object Properties: hasSource

- hasType
- isAuthorizationOf
- hasContributor
- hasCreator
- isApplicationOf
- hasAuthorization
- productUsedBy
- hasResourceManager
- hasUpdatingFrequency
- hasLegalStatute
- hasMandate
- isResourceManagerOf
- isSubjectOf
- hasLanguage
- hasOrganisationType
- isRatifiedBy
- hasPublisher
- hasAudience
- hasStatus
- isPublisherOf
- hasSubject
- hasRelation
  - hasEmployee
  - hasVendor
  - isFormatOf
  - isPartOf
  - hasWorkers
  - hasPart
  - worksIn
    - hasSource
    - isSourceOf
    - isManagerOf
    - isEmployeeOf
    - hasManager
    - isVendorOf
  - hasRights
- hasFormat
- hasCoverage
- usesProduct
- hasApplication

Annotations Object Property Usage

Annotations: hasSource

Annotations: hasSource

description

"Maakt het de burger mogelijk overheidsbronnen te vinden die ontwikkeld zijn met behulp van de inhoud van een bepaald ander document (bijv. alle documenten gebaseerd op een bepaalde wetgeving of statistieken).  
Wordt bijvoorbeeld gebruikt om de grondslag van wet- en regelgeving aan te geven.

label

"Bron"

title

"Verwijzing naar een bron waar de huidige bron van afgeleid is"

Characteristics: has

Functional

Inverse functional

Transitive

Symmetric

Asymmetric

Reflexive

Irreflexive

Description: hasSource

Domains (Intersection)

- Resource

Ranges (Intersection)

- Resource
- ResourceProvenance

Equivalent object properties

- hasRelation

Super properties

- isSourceOf
- isSourceOf

Inverse properties

- isSourceOf

Disjoint properties

Property chains

http://www.vlaanderen.be/ontologie/instances.owl - [C:\Users\Philippe\Desktop\VO\instances.owl]

File Edit Reasoner Tools Refactor Tabs View Window Help

instances.owl http://www.vlaanderen.be/ontologie/instances.owl

Active Ontology Entities Classes Object Properties Data Properties Individuals OMLViz DL Query

Data Properties: hasURL

Delete selected properties

- hasName
  - hasLastName
  - hasFirstName
- hasDate
  - isModified
  - isAvailable
  - isValid
  - hasNextVersionDue
  - isIssued
- hasCodeVlmpers
- hasIdentifier
  - hasURI
- hasTitle
- hasDescription

Data Property Annotations Data Property Usage

Data Property Annotations: hasURL

Annotations: hasURL

comment

"Elke website wordt eenduidig bepaald door een URL.  
Opmerking:  
Het zou eigenlijk beter zijn om de URI van deze bronnen te vervangen door deze URL."

label

"Heeft URL"

Data Property characteristics: hasURL

Functional

Description: hasURL

Domains

- Website

Ranges

Equivalent properties

Super properties

- hasIdentifier

Disjoint properties



http://www.vlaanderen.be/ontologie/instances.owl - (C:\Users\Philippe\Desktop\VO\instances.owl)

File Edit Reasoner Tools Refactor Tabs View Window Help

instances.owl http://www.vlaanderen.be/ontologie/instances.owl

Active Ontology Entites Classes Object Properties Data Properties Individuals OWL Viz DL Query

Individuals Individuals By Class Individual Annotations Individual Usage

Individuals By Class: MSAccess

- Product
  - MSPowerpoint
  - MSEXcel
  - MSAccess
  - Documentum
  - Pakket
  - MSWord
- Vendor
  - Oracle
  - Microsoft
- VlaamseOpenbareInstelling
- InternVerzelfstandigdeAgentschap
- SSP\_Database
- ExternVerzelfstandigdeAgentschap
- Act
- BeleidsDepartement
- Person
- SSP\_ServersOrComputers
- SSP\_WebServer
- Overheid
- SpecifiedResource
- ResourceProvenance
- Authorization
- Beleidsdomain
- SSP\_storage
- LegalStatute

Individual Annotations: MSAccess

Annotations

Description: MSAccess

Types

- Product

Same individuals

Different individuals

Property assertions: MSAccess

Object property assertions

- hasVendor Microsoft
- hasRelation MSPowerpoint
- hasRelation MSExcel
- hasRelation MSAccess
- hasRelation Microsoft
- hasRelation MSWord

Data property assertions

- hasAlternative "Access"
- hasTitle "MS Access"

Negative object property assertions

Negative data property assertions

http://www.vlaanderen.be/ontologie/instances.owl - (C:\Users\Philippe\Desktop\VO\instances.owl)

File Edit Reasoner Tools Refactor Tabs View Window Help

instances.owl http://www.vlaanderen.be/ontologie/instances.owl

Active Ontology Entites Classes Object Properties Data Properties Individuals OWL Viz DL Query

Individuals Individuals By Class Individual Annotations Individual Usage

Individuals By Class: Test

- Product
- Vendor
- VlaamseOpenbareInstelling
- InternVerzelfstandigdeAgentschap
- SSP\_Database
  - Database
  - SQLServer
  - Oracle8
  - UnknownDB
  - DB2
- ExternVerzelfstandigdeAgentschap
- Act
- BeleidsDepartement
- Person
- SSP\_ServersOrComputers
- SSP\_WebServer
- Overheid
- SpecifiedResource
- Test
- ResourceProvenance
- Authorization
- Beleidsdomain
- SSP\_storage
- LegalStatute

Individual Annotations: Test

Annotations

Description: Test

Types

- SpecifiedResource
- Database
- ManagedResource
- WebResource

Same individuals

Different individuals

Property assertions: Test

Object property assertions

- hasPublisher DepartementInternationaalVlaanderen
- hasResourceManager Testpersoon
- hasApplication Webserver
- hasMedium Oracle8
- hasFormat Oracle8
- hasResourceManager Testpersoon
- hasPublisher DepartementInternationaalVlaanderen
- hasApplication Webserver
- hasMedium Oracle8

Data property assertions

- hasDescription "Dit is een voorbeeld gegevensbron om de mogelijkheden van het model aan te tonen."
- hasTitle "Test Database"

http://www.vlaanderen.be/ontologie/instances.owl - [C:\Users\Philippe\Desktop\VO\instances.owl]

File Edit Reasoner Tools Refactor Tabs View Window Help

instances.owl http://www.vlaanderen.be/ontologie/instances.owl

Active Ontology Entities Classes Object Properties Data Properties Individuals OWL Viz DL Query

Individuals Individuals By Class

Individuals By Class: MobiliteitOpenbareWerken

- Product
- Vendor
- VlaamseOpenbareInstelling
- InternVerzelfstandigdeAgentschap
- SSP\_Database
- ExternVerzelfstandigdeAgentschap
- Act
- BeleidsDepartement
- Person
- SSP\_ServersOrComputers
- SSP\_WebServer
- Overheid
- SpecifiedResource
- ResourceProvenance
- Authorization
- Beleidsdomein
  - EconomieWetenschapInnovatie
  - MobiliteitOpenbareWerken**
  - RuimtelijkeOrdeningWoonbeleid
  - LeefmilieuNatuurEnergie
  - CultuurJeugdSportMedia
  - OnderwijsVorming
  - LandbouwVisserij
  - DienstenAlgemeenRegeringsbeleid
  - WelzijnVolksgezondheidGezinnen
  - FinancienBegroting
  - WerkSocialeEconomie
  - InternationaalVlaanderen
  - Bestuurszaken
- SSP\_Storage
- LegalStatute

Individual Annotations Individual Usage

Individual Annotations: MobiliteitOpenbareWerken

Annotations

Description: MobiliteitOpenbareWerken

Types

- Beleidsdomein

Same individuals

Different individuals

Property assertions: MobiliteitOpenbareWerken

Object property assertions

- isPartOf VlaamseOverheid
  - hasPart DepartementMobiliteitOpenbareWerken
  - hasPart AgentschapWegenVerkeer
  - hasPart DeScheepvaart
  - hasPart AgentschapMaritiemeDienstverleningKust
  - hasPart WaterwegenZeekanaal
  - hasPart VlaamseVervoermaatschappijDeLijn
- hasRelation DepartementMobiliteitOpenbareWerken
- hasRelation VlaamseOverheid
- hasRelation AgentschapWegenVerkeer
- hasRelation DeScheepvaart
- hasRelation AgentschapMaritiemeDienstverleningKust
- hasRelation WaterwegenZeekanaal
- hasRelation VlaamseVervoermaatschappijDeLijn
- isPartOf VlaamseOverheid

Data property assertions

- hasTitle "Mobiliteit en Openbare Werken"
- hasAbbreviation "MOW"

Negative object property assertions

Negative data property assertions

## **Bijlage 5: DC-GAP**

DC-GAP is een Application Profile van de Dublin Core Metadata Set en is te vinden op:

<http://www.jiscmail.ac.uk/cgi-bin/webadmin?A2=ind0508&L=dc-government&T=0&P=283>

*A government application profile will be a specification that defines the following:*

- *required elements*
- *permitted Dublin Core elements*
- *permitted Dublin Core qualifiers*
- *permitted schemes and values (e.g. use of a specific controlled vocabulary or encoding scheme)*
- *government domain elements used from another namespace*
- *additional elements/qualifiers from other application profiles that may be used (e.g. DC-Education: Audience)*
- *refinement of standard definitions*

*This document proposes a possible application profile that clarifies the use of the Dublin Core Metadata Element Set in governments and government-related applications and projects.*

*It has been reformatted in conformance with the Dublin Core Application Profile Guidelines produced by the CEN MMI-DC Workshop.*

## **Lijst van Figuren**

Figuur 1 Gegevens, informatie, kennis .....	7
Figuur 2 Relatieve waarde van gegevens, informatie en kennis .....	8
Figuur 3 Data-informatie-kennis-wijsheid stroom (bron: Ahsan S. & Shah A.) .....	9
Figuur 4 Modulaire ontologieën samenvoegen (bron: Bratt S. 2006) .....	21
Figuur 5 Levenscyclus ontologie (bron: Euzenat J.).....	22
Figuur 6 De plaats van kennis (MITRE 2004).....	25
Figuur 7 Gartner Hype Cycle for Emerging Technologies (bron: Garnter 2007).....	30
Figuur 8 Web 3.0 (bron: Radar Networks 2007) .....	31
Figuur 9 Bibliotheken en interconnecties (bron: LinkingOpenData) .....	35
Figuur 10 Semantisch Web layer cake (bron: W3C) .....	39
Figuur 11 Geschiedenis van OWL.....	49
Figuur 12 Krachten onderneming (bron: Balchin N.).....	56
Figuur 13 Oplossingen balans (bron: Balchin N.) .....	56
Figuur 14 Elementen Enterprise Architecture (bron: MITRE 2004).....	57
Figuur 15 Belangrijkste ontwikkelingen in de geschiedenis van EA (bron: MITRE 2004) .....	59
Figuur 16 Beleidscyclus.....	66
Figuur 17 Screenshot Swoop .....	68
Figuur 18 Screenshot SemanticWorks 2008.....	69
Figuur 19 Screenshot Protégé 4.0a.....	71
Figuur 20 Screenshot Powl .....	71
Figuur 21 Screenshot OntoWiki .....	72
Figuur 22 Klassenhiërarchie oorspronkelijk model .....	73
Figuur 23 Objecteigenschappen oorspronkelijk model.....	74
Figuur 24 Dataeigenschappen oorspronkelijk model.....	75
Figuur 25 FEA-RMO Government Core .....	82
Figuur 26 FEA-RMO BRM .....	83
Figuur 27 FEA-RMO TRM .....	84
Figuur 28 Necessary en sufficient condities (bron: University of Manchester 2007) ....	88
Figuur 29 Asserted Class Hierarchy en Inferred Class Hierarchy .....	91
Figuur 30 OWLviz .....	92
Figuur 31 Objecteigenschappen in Protégé.....	94

Figuur 32 Test-instantie database.....	96
Figuur 33 Testinstantie database, met reasoning .....	97
Figuur 34 BMO architectuur (bron: Jenz & Partner GmbH 2004) .....	100
Figuur 35 Communicatie tussen bedrijven met ontologieën.....	104
Figuur 36 Communicatie tussen bedrijven met ontologieën en het gebruik van een generieke ontologie van de Vlaamse Overheid.....	104

## **Bronnen**

### Boeken

Ambler S., Jo E., Linn J., McGovern J., Sharan V. & Stevens M. (2003). *Practical guide to enterprise architecture*. Prentice Hall PTR; 1st edition.

Antoniou G. & van Harmelen F. (2004). *A Semantic Web Primer*. The MIT Press.

De Baer, Olivier (2004). *Ontologietalen voor het semantic web : state of the art*. Eindverhandeling K.U.Leuven.

Van Heghe H. (2005). *Leren zwemmen in informatie*. ICMS Group n.v.

Vanthienen J. (2006). *Business Intelligence*. Alta Uitgeverij.

### Papers

Ahsan S. & Shah A. (2006). *Data, Information, Knowledge, Wisdom: A Doubly Linked Chain?* Research and Development Center of Computer Science University of Engineering and Technology, Lahore.

Allemang D., Hodgson R. & Polikoff I. (2005). *FEA Reference Model Ontologies (FEA RMO) - Version 1.1*. TopQuadrant.

Bailey I. (2006). *A Simple Guide to Enterprise Architecture*. Model Futures.

Baker D. & Janiszewski M. (2006). *7 Essential Elements of EA*. Diamond Management & Technology Consultants, Inc.

Blevins T. (2006). *The Architecture of Enterprise Architecture*. MITRE Corporation.

De Rooij R., Graafland-Essers I., Kranenkamp H., Lierens A., Overbeek H., Rothenberg J., Stegeman M., Van Oranje C. & Van Schaik R. (2006). *Overheid.nl Webmetadata – Het verbeteren van de toegankelijkheid van digitale informatie binnen de Nederlandse overheid – handboek CONCEPT Versie 3.0 – recommendation*. Advies Overheid.nl.

Euzenat J. (2002). *Research challenges and perspectives of the Semantic Web - Report of the EU-NSF strategic workshop - held at Sophia-Antipolis, France, October 3rd-5th, 2001*. European Commission - US National Science Foundation.

Fensel D. (2008). *Semantic Technology – More Than Just an Appendix of the Web?* Semantic Technology Institute.

Goethals F. (?). *An Overview of Enterprise Architecture Framework Deliverables - A study of existing literature on 'architectures'*. SAP-leerstoel.

- Graafland-Essers I., Kranenkamp H., Lierens A., Rothenberg J., Van Oranje C. & Van Schaik R. (2004). *Designing a National Standard for Discovery Metadata - Improving Access to Digital Information in the Dutch Government*. RAND Europe.
- Gruber T. (1993). *Toward Principles for the Design of Ontologies Used for Knowledge Sharing*. Stanford Knowledge Systems Laboratory.
- Hagan P. (2004). *Guide to the (Evolving) Enterprise Architecture Body of Knowledge – Draft*. MITRE Corporation.
- Lemahieu W. (2002). *Web Service description, advertising and discovery: WSDL and beyond*. K.U.Leuven.
- McGuinness D. & Noy N. (2001). *Ontology Development 101: A Guide to Creating Your First Ontology*. Stanford University.
- N.N. (2004). *Business Management Ontology (BMO) Version 1.0 Release Notes*. Jenz & Partner GmbH.
- N.N. (2007). *DoD Architecture Framework Version 1.5, Volume II: Product Descriptions*. Department of Defense, United States of America.
- N.N. (2005). *e-Government Interoperability Framework Version 6.1*. CabinetOffice – e-Government Unit.
- N.N. (2006). *e-Government Metadata Standard version 3.1*. CabinetOffice – e-Government Unit.
- N.N. (1999). *Federal Enterprise Architecture Framework Version 1.1*. The Chief Information Officers Council.
- Zachman J. (1996). *Enterprise Architecture: The Issue of the Century*. Zachman International.

### Slideshows

- Maitra A. (2005). *Recommended approach for the FEA Data Reference Model (DRM)*. Washington DC.
- N.N. (2008). *A Tutorial on the Zachman Framework for Enterprise Architecture*.
- Niemann B. (2003). *Implementing Component-Based Government Enterprise Architecture with Semantic Web Services*. US EPA Office of Environmental Information.

Van Herreweghe N. (2007). The “Information Warehouse” challenge - Business Process within the context of a Holistic Enterprise Architecture . Process- and Information Management - Flemish Government.

### Websites

[http://advice.cio.com/mitch\\_de\\_felice/will\\_your\\_company\\_sparkle\\_in\\_2008\\_the\\_state\\_of\\_semantic\\_web?page=0%2C1](http://advice.cio.com/mitch_de_felice/will_your_company_sparkle_in_2008_the_state_of_semantic_web?page=0%2C1) [15/04/2008]

<http://advies.overheid.nl/metadata-standpunt/> [29/01/2008]

<http://aksw.org/Projects/Powl> [22/10/2007]

<http://aperture.sourceforge.net/> [12/05/2007]

[http://aps.vlaanderen.be/straplan/beleidsinformatie/straplan\\_beleidsinformatie.htm](http://aps.vlaanderen.be/straplan/beleidsinformatie/straplan_beleidsinformatie.htm)  
[29/01/2008]

<http://architectureframework.com/> [07/02/2008]

<http://composing-the-semantic-web.blogspot.com/> [15/04/2008]

<http://dbpedia.org/About> [10/11/2007]

<http://dublincore.org/> [18/07/2007]

<http://dublincore.org/groups/government/> [18/07/2007]

<http://en.wikipedia.org/wiki/DoDAF> [05/03/2008]

[http://en.wikipedia.org/wiki/Dublin\\_Core](http://en.wikipedia.org/wiki/Dublin_Core) [22/10/2007]

<http://en.wikipedia.org/wiki/E-Government> [29/01/2008]

[http://en.wikipedia.org/wiki/Enterprise\\_architecture](http://en.wikipedia.org/wiki/Enterprise_architecture) [05/03/2008]

[http://en.wikipedia.org/wiki/Enterprise\\_Architecture\\_framework](http://en.wikipedia.org/wiki/Enterprise_Architecture_framework) [05/03/2008]

[http://en.wikipedia.org/wiki/Frame\\_language](http://en.wikipedia.org/wiki/Frame_language) [24/03/2007]

[http://en.wikipedia.org/wiki/Knowledge\\_economy](http://en.wikipedia.org/wiki/Knowledge_economy) [05/03/2008]

<http://en.wikipedia.org/wiki/MODAF> [05/03/2008]

[http://en.wikipedia.org/wiki/Ontology\\_\(computer\\_science\)](http://en.wikipedia.org/wiki/Ontology_(computer_science)) [24/03/2007]

<http://en.wikipedia.org/wiki/RDFS> [05/03/2008]

[http://en.wikipedia.org/wiki/Reasoning#Cognitive\\_science\\_and\\_artificial\\_intelligence](http://en.wikipedia.org/wiki/Reasoning#Cognitive_science_and_artificial_intelligence)  
[05/03/2008]

[http://en.wikipedia.org/wiki/Semantic\\_Web](http://en.wikipedia.org/wiki/Semantic_Web) [05/01/2007]

<http://en.wikipedia.org/wiki/SPARQL> [22/10/2007]

<http://en.wikipedia.org/wiki/TOGAF> [05/03/2008]

[http://en.wikipedia.org/wiki/Web\\_Ontology\\_Language](http://en.wikipedia.org/wiki/Web_Ontology_Language) [22/10/2007]



[http://en.wikipedia.org/wiki/Zachman\\_framework](http://en.wikipedia.org/wiki/Zachman_framework) [05/03/2008]  
<http://esw.w3.org/topic/SweoIG/TaskForces/CommunityProjects/LinkingOpenData>  
[10/04/2008]  
<http://evolvingtrends.wordpress.com/2006/06/26/wikipedia-30-the-end-of-google/>  
[20/07/2007]  
<http://ewi-vlaanderen.be/ewi/index.php?id=385> [29/01/2008]  
<http://groupme.org/GroupMe/> [20/07/2007]  
<http://infomesh.net/2001/06/swform/> [18/04/2007]  
<http://infomesh.net/2001/08/swtips> [27/02/2008]  
<http://infomesh.net/2001/swintro/> [27/02/2008]  
<http://jena.sourceforge.net/> [18/04/2007]  
<http://lifeboat.com/ex/minding.the.planet> [27/02/2008]  
<http://lifeboat.com/ex/web.3.0> [27/02/2008]  
<http://logicerror.com/semanticWeb-long> [11/09/2007]  
<http://logicerror.com/semanticWeb-webdev> [11/09/2007]  
<http://mspace.fm/> [18/07/2007]  
[http://nl.wikipedia.org/wiki/Vlaamse\\_overheid](http://nl.wikipedia.org/wiki/Vlaamse_overheid) [11/09/2007]  
<http://olp.dfki.de/ontoselect> [11/09/2007]  
<http://ontobroker.semanticweb.org/rdfcrawl/> [17/03/2008]  
<http://ontowiki.net/Projects/OntoWiki> [18/07/2007]  
<http://opencalais.mashery.com/Overview> [18/07/2007]  
<http://planetrdf.com/> [18/07/2007]  
<http://protege.stanford.edu/> [18/04/2007]  
<http://protege.stanford.edu/doc/owl/owl-imports.html> [18/04/2007]  
<http://protegewiki.stanford.edu/index.php/Protege4Migration> [27/02/2008]  
<http://protegewiki.stanford.edu/index.php/Protege4TutorialErrata> [27/02/2008]  
[http://protegewiki.stanford.edu/index.php/Setting\\_Heap\\_Size](http://protegewiki.stanford.edu/index.php/Setting_Heap_Size) [22/04/2008]  
[http://semantic-mediawiki.org/wiki/Semantic\\_MediaWiki](http://semantic-mediawiki.org/wiki/Semantic_MediaWiki) [18/04/2007]  
<http://semanticfocus.com/blog/entry/title/introduction-to-semantic-web-vision-and-technologies-part-4-protege-101-screencast/> [04/10/2007]  
<http://semanticfocus.com/blog/entry/title/introduction-to-semantic-web-vision-and-technologies-part-5-building-owl-ontologies-using-protege-4-screencast/> [12/02/2008]  
<http://swag.webns.net/whatIsSW> [18/07/2007]  
[http://talk.talis.com/archives/2008/02/sir\\_tim\\_berners.html](http://talk.talis.com/archives/2008/02/sir_tim_berners.html) [27/02/2008]

<http://techbiz.blog.com/1730241/> [11/09/2007]  
<http://technorati.com/tag/Semantic%20Web> [17/03/2008]  
<http://togaforblunder.blogspot.com/> [27/02/2008]  
<http://tweakers.net/nieuws/43567/onenigheid-over-toekomstvisie-semantische-web.html>  
[27/02/2008]  
<http://tweakers.net/nieuws/47387/Onderzoekers-bouwen-supersnelle-rdf-zoekmachine.html> [27/02/2008]  
<http://web-services.gov/fea-rmo.html> [17/03/2008]  
<http://webns.net/> [18/07/2007]  
<http://www-ksl.stanford.edu/kst/what-is-an-ontology.html> [11/09/2007]  
[http://www-ksl.stanford.edu/people/dlm/papers/ontologies-come-of-age-mit-press-\(with-citation\).htm](http://www-ksl.stanford.edu/people/dlm/papers/ontologies-come-of-age-mit-press-(with-citation).htm) [18/04/2007]  
<http://www.answers.com/topic/folksonomy?cat=technology> [27/02/2008]  
<http://www.belgium.be/eportal/application?languageRedirected=yes&navId=5449&origin=aboutBelgium.jsp&pageid=indexPage&event=bea.portal.framework.internal.refresh>  
[29/01/2008]  
[http://www.co-ode.org/resources/reference/manchester\\_syntax/](http://www.co-ode.org/resources/reference/manchester_syntax/) [27/02/2008]  
<http://www.cs.umd.edu/projects/plus/SHOE/faq.html> [11/09/2007]  
<http://www.dapper.net/> [27/02/2008]  
<http://www.dapper.net/semantify/> [27/02/2008]  
<http://www.dublincore.org/dcgapwiki> [18/07/2007]  
<http://www.essi-cluster.org/> [23/04/2008]  
<http://www.foaf-project.org/> [18/07/2007]  
<http://www.freebase.com/site/data> [18/07/2007]  
[http://www.gcn.com/research\\_results/entarch6.html](http://www.gcn.com/research_results/entarch6.html) [11/09/2007]  
[http://www.govtalk.gov.uk/schemasstandards/metadata\\_document.asp?docnum=1017](http://www.govtalk.gov.uk/schemasstandards/metadata_document.asp?docnum=1017)  
[27/02/2008]  
<http://www.govtalk.gov.uk/schemasstandards/schemasstandards.asp> [17/03/2008]  
<http://www.iht.com/articles/2006/05/23/business/web.php> [11/09/2007]  
<http://www.internetnews.com/dev-news/article.php/3640361> [27/02/2008]  
[http://www.intranetjournal.com/articles/200709/ij\\_09\\_14\\_07a.html](http://www.intranetjournal.com/articles/200709/ij_09_14_07a.html) [17/03/2008]  
<http://www.jiscmail.ac.uk/cgi-bin/webadmin?A2=ind0508&L=dc-government&T=0&P=283> [11/09/2007]

[http://www.mondeca.com/index.php/en/intelligent\\_topic\\_manager/features/ontology\\_management\\_ontology\\_software](http://www.mondeca.com/index.php/en/intelligent_topic_manager/features/ontology_management_ontology_software) [18/04/2007]

<http://www.nabble.com/Protege---OWL-f14404.html> [15/04/2008]

[http://www.neon-project.org/web-content/index.php?option=com\\_frontpage&Itemid=1](http://www.neon-project.org/web-content/index.php?option=com_frontpage&Itemid=1) [17/03/2008]

[http://www.nytimes.com/2006/11/12/business/12web.html?\\_r=4&adxnnl=1&oref=slogi n&ref=technology&pagewanted=1&adxnnlx=1210411277-pks5SNwnzAWQQk6qD88Jbg](http://www.nytimes.com/2006/11/12/business/12web.html?_r=4&adxnnl=1&oref=slogi n&ref=technology&pagewanted=1&adxnnlx=1210411277-pks5SNwnzAWQQk6qD88Jbg) [18/04/2007]

<http://www.opengroup.org/architecture/togaf8-doc/arch/chap37.html> [27/02/2008]

[http://www.readwriteweb.com/archives/semantic\\_web\\_11\\_things\\_to\\_know.php](http://www.readwriteweb.com/archives/semantic_web_11_things_to_know.php) [17/03/2008]

[http://www.regdeveloper.co.uk/2005/02/17/semantic\\_web/](http://www.regdeveloper.co.uk/2005/02/17/semantic_web/) [18/04/2007]

[http://www.regdeveloper.co.uk/2007/02/11/search\\_and\\_semantics/](http://www.regdeveloper.co.uk/2007/02/11/search_and_semantics/) [17/03/2008]

<http://www.regering.nl/meta/thc/> [27/02/2008]

<http://www.schemaweb.info/> [18/07/2007]

<http://www.sciam.com/article.cfm?id=the-semantic-web&ref=sciam> [11/09/2007]

<http://www.semanticfocus.com/> [11/09/2007]

<http://www.sti2.org/> [27/02/2008]

<http://www.syndicatescape.com/> [17/03/2008]

<http://www.taguri.org/> [18/04/2007]

[http://www.tbs-sct.gc.ca/im-gi/meta/profil/profiltb\\_e.asp](http://www.tbs-sct.gc.ca/im-gi/meta/profil/profiltb_e.asp) [17/03/2008]

[http://www.theregister.co.uk/2007/05/04/semantic\\_web\\_breakthrough/](http://www.theregister.co.uk/2007/05/04/semantic_web_breakthrough/) [11/09/2007]

<http://www.topquadrant.com/sparqlmotion/> [29/01/2008]

<http://www.trueknowledge.com/> [27/02/2008]

<http://www.twine.com/> [29/01/2008]

<http://www.w3.org/2001/sw/> [12/04/2007]

<http://www.w3.org/2001/sw/Activity.html> [12/04/2007]

<http://www.w3.org/2001/sw/SW-FAQ> [12/04/2007]

<http://www.w3.org/2001/sw/sweo/public/UseCases/Zaragoza/> [12/04/2007]

<http://www.w3.org/2001/sw/WebOnt/> [12/04/2007]

<http://www.w3.org/2002/03/semweb/> [12/04/2007]

<http://www.w3.org/2002/07/swint> [12/04/2007]

<http://www.w3.org/2004/ontaria/> [12/04/2007]

<http://www.w3.org/2007/12/sparql-release.html.en> [27/02/2008]

[http://www.w3.org/2007/OWL/wiki/OWL\\_Working\\_Group](http://www.w3.org/2007/OWL/wiki/OWL_Working_Group) [18/11/2007]  
<http://www.w3.org/2007/Talks/0424-Stavanger-IH/> [17/03/2008]  
<http://www.w3.org/DesignIssues/Business> [12/04/2007]  
<http://www.w3.org/DesignIssues/RDFnot.html> [12/04/2007]  
<http://www.w3.org/DesignIssues/Semantic.html> [12/04/2007]  
<http://www.w3.org/DesignIssues/UI.html#OhYeah> [12/04/2007]  
<http://www.w3.org/RDF/Validator/> [12/04/2007]  
<http://www.w3.org/Submission/SWRL/> [27/02/2008]  
<http://www.w3.org/TR/2008/WD-xhtml-rdfa-primer-20080317/> [27/02/2008]  
<http://www.w3.org/TR/grddl/> [27/02/2008]  
<http://www.w3.org/TR/owl-features/> [12/04/2007]  
<http://www.w3.org/TR/owl-ref/> [12/04/2007]  
<http://www.w3.org/TR/rdf-schema/> [12/04/2007]  
<http://www.w3.org/TR/rdf-sparql-query/> [27/02/2008]  
<http://www.w3.org/TR/rdf-syntax-grammar/> [12/04/2007]  
<http://www.w3.org/TR/webont-req/> [12/04/2007]  
<http://www.whitehouse.gov/omb/egov/a-1-fea.html> [17/03/2008]  
<http://www.xml.com/pub/a/2000/10/25/dublincore/index.html> [27/02/2008]  
<http://www.xml.com/pub/a/2001/03/07/buildingsw.html> [11/09/2007]  
<http://www4.wiwiw.fu-berlin.de/bizer/d2r-server/> [17/03/2008]  
<https://mailman.stanford.edu/pipermail/p4-feedback/> [15/02/2008]

