

Resource Description Framework (RDF) -konsepti

Lauri Antikainen

Opinnäytetyö

Tietojenkäsittelyn koulutusohjelma

2013



Tekijä tai tekijät Lauri Antikainen	Ryhmätunnus tai aloitusvuosi 2010
Raportin nimi Resource Description Framework (RDF)-konsepti	Sivu- ja liitesivumäärä 28 + 0
Opettajat tai ohjaajat Arvo Lipitsäinen	
<p>Tämä on opinnäytetyö Resource Description Framework (RDF) -konseptista. Tämä opinnäytetyö on suunnattu lukijoille, joilla ei ole ennalta mainittavaa käsitystä RDF:stä. Tästä syystä tarkasteltavaksi on otettu koko konsepti, sillä yksittäisten seikkojen ymmärtäminen konseptin sisältä on erittäin vaikeaa ilman edes hataraa käsitystä kokonaisuudesta.</p> <p>Työssä pyrittiin kuvaamaan se, mitä RDF tarkoittaa, mitä se sisältää, mikä on sen paikka Semanttisessa Webissä ja miksi sitä kehitetään. Tämä työ avaa myöskin semanttisen webin käsitettä ja selittää kuinka se, ja RDF liittyvät toisiinsa.</p> <p>Sen lisäksi tämä työ käy läpi RDF-syntaksia, RDF-tietokantoja ja SPARQL:ää, RDF-tietokantojen kyselykieltä. Tekstissä on esimerkkejä syntakseista ja tietokantojen rakenteista. Esillä on myös SPARQL:n syntaksia ja sen eroja relaatiotietokantoihin.</p> <p>Tuloksina saatiin yhteenveto eri aiheista ja päätelmiä esimerkiksi RDF-tietokantojen käyttöönoton järjestyksestä yrityksissä, joissa on tähän mennessä ollut relaatiotietokantajärjestelmä.</p>	
Asiasanat RDF, tietomalli, Semanttinen Web, tietokannat	

Degree programme

<p>Authors</p> <p>Lauri Antikainen</p>	<p>Group or year of entry</p> <p>2010</p>
<p>The title of thesis</p> <p>The Concept of Resource Description Framework (RDF)</p>	<p>Number of report pages and attachment pages</p> <p>28 + 0</p>
<p>Advisor(s)</p> <p>Arvo Lipitsäinen</p>	
<p>This thesis is about Resource Description Framework (RDF). This thesis is aimed at those who have no prior knowledge about RDF. This thesis looks at the concept as a whole instead of focusing on certain details. The reasoning behind this approach is the complexity of the subject.</p> <p>The objective of this thesis was to clarify what RDF means, what it entails, how it is connected to the Semantic Web and why it is being developed, both in the first place and still today. Due to the connection to the Semantic Web, it was necessary to explain also what the Semantic Web is.</p> <p>The thesis looked into the RDF-syntax, RDF-databases and the RDF-database query language SPARQL in detail. There are examples of the RDF-syntax and the SPARQL-syntax. In addition, several differences between relational databases and RDF-databases were covered.</p> <p>As results the thesis has a summary of its contents. Results also include several conclusions based on the data shown in the thesis. The thesis concludes that it is a viable strategy to change from a relational database system to an RDF-database system in a company, as long as said company has clearly changing data in their system.</p>	
<p>Key words</p> <p>RDF, data model, Semantic Web, database</p>	

Sisällys

1	Johdanto	1
2	RDF	4
2.1	RDF tietomallin tarkoitus	4
2.2	RDF tietomallin periaate	5
3	Semanttinen Web	7
3.1	Yleisesti	7
3.2	RDF Semanttisessa Webissä	9
3.3	Miksi RDF-syntaksi on valittu XML:n sijasta	10
3.3.1	Samankaltaisuudet	10
3.3.2	Erot	10
3.3.3	Vahvuudet	11
4	RDF:n suunnittelun päämäärät	12
4.1	Yksinkertainen tietomalli	12
4.2	RDF:n muut päämäärät	13
5	RDF tietokannat	14
5.1	Tiedon saatavuus ja yhdenmukaisuus	15
5.2	Välimuisti puskuroidaan SPARQL:ää	16
5.3	Joustavuus RDF-tietokannoissa tietomallin ansiosta	16
6	SPARQL	17
6.1	Kyselyt	18
6.2	RDF Dataset	21
7	Päätelmät	24
8	Johtopäätökset	26
8.1	Yhteenvedo	26
8.2	Työn arviointi	27

1 Johdanto

Suuri osa maailman tiedosta löytyy tavalla tai toisella tietokantojen sisältä. Pääasiassa relaatiotietokantojen sisältä. Vielä suurempia tietomääriä on tavallisissa tiedostoissa, kuten sähköpostiarkistoissa ja vastaavissa lähteissä. Kaikkien näiden tietojen integrointi antaisi valtavasti hyötyä organisaatioille, jotka tällaista dataa omistavat.

Tähän integraatioon pyrkii niin kutsuttu Semanttinen Web ja RDF (Resource Description Framework) on sen työkalu, joka sitä mahdollistaa. Semanttista Webiä ja RDF:ää kehittää eteenpäin muun muassa W3C (World Wide Web Consortium), jota johtaa World Wide Webin kehittäjä, Tim Berners-Lee. Berners-Lee olikin ensimmäisiä, joka valoitti Semanttisen Webin ja tekoälyn eroja (Berners-Lee 1998).

RDF on eri lähteistä tietoa yhdistävä Webin kieli (Lassila 1997), jonka historia alkoi vuonna 1995 (Miller 1998). Helmikuussa 2014 W3C julkaisi uusia versioita RDF dokumenteista ja standardi versio on nyt 1.1, 1.0:n ollessa 10 vuotias. SPARQL on RDF:n kyselykieli, ja sillä voidaan liittää tietoja yhteen mm. eri tietokannoista ja asiakirjoista. SPARQL on erinomainen kieli yhtenäistää data relaatiotietokantojen ja muiden tietokantojen välillä.

Semanttisen Webin synty on innoittanut useita yhdyskäytäviä RDF:n ja tavanomaisten relaatio varastojen välillä. Jotkut järjestelmät uudelleenkirjoittavat SPARQL kyselyitä SQL:ään. Lisäämällä natiivi SPARQL-tuki tietokantaan saadaan sama suorituskyky kuin hyvin optimoiduilla SQL-kyselyillä.

Aiheena RDF on jatkuvasti muuttuva ja kehittyvä, mikä luo oman vaikeutensa sen tutkimiseen. RDF edustaa tulevaisuutta ja on mielenkiintoinen aihe senkin takia.

Tämä opinnäytetyö pyrkii vastaamaan kysymyksiin mikä RDF on, miksi se on olemassa ja miksi sitä kehitetään yhä edelleen. Alussa selvittämme RDF:n peruspiirteet, sitten sen paikan Semanttisessa Webissä ja lopuksi RDF-tietokantoja yleisesti ja sen

kyselykielen SPARQL:n ominaisuuksia. Tämä opinnäyteö ei ole minkäänlainen asennusohje, eikä se opeta koodaamaan RDF-syntaksia tai tekemään tietokantoja.

Tämä opinnäytetyö on teoreettinen tutkimus, eli siinä on lähteistä kerätty tieto jäsennetty yhdeksi kokonaisuudeksi.

Sanasto

IRI – URIn laajennos, joka mahdollistaa Unicode-merkistön käytön.

Literaali – Tietokoneohjelman lähdekoodissa tietyn tietotyypin tiedon esitysmuoto.

NoSQL – ”Not only SQL” tarkoittaa mitä tahansa tietokantaformaattia, joka ei ole relaatiotietokanta.

ODBC – Open database connectivity on avoin rajapinta tietokannoille. Standardoitu.

Ontologia – Semanttisen metatiedon kuvaukseen käytettävien käsitteiden ja ominaisuuksien formaali kuvaus.

OWL – Ontologiapohjaisen tiedon kuvauskieli.

RDF – Web-ympäristössä erityisesti sovellusten välisen tiedon vaihdon malli.

RDFS – RDF Schema.

RDF-syntaksi – RDF koodikieli

SPARQL – RDF kyselykieli tietokannoille.

SQL – Relaatiotietokantojen kyselykieli. IBM standardi.

Tyhjä solmu – Solmu, josta puuttuu informaatio.

URI – URL (tietyn tiedon paikan osoittava merkkijono) tai URN (yksikäsitteinen nimi).

URL – Tietyn tiedon paikan osoittava merkkijono.

W3C – World Wide Web Consortium. Webbiä kehittävä ryhmä MIT yliopistossa.

XML – Laajojen tietomassojen jäsentämiseen W3C:n kehittämä kuvauskieli.

2 RDF

Resource Description Framework (RDF) on kieli, joka on tehty edustamaan tietoa resursseista World Wide Webissä. RDF on W3C:n eli World Wide Web Consortiumin standardi tiedon muuntamiselle koodikielelle. Sitä käytetään semanttisen webin ratkaisuihin. (W3C 2004a.)

Se on erityisesti tarkoitettu edustamaan metatietoja Web-resursseista, kuten otsikkoa, tekijää ja Web-sivun muutoksen päivämäärää, tekijänoikeus ja lisensointi tietoa Web-asiakirjasta tai saatavuuden aikataulua jostakin yhteisestä resurssista. (W3C 2004a.)

RDF:ää voidaan käyttää myös edustamaan tietoa asioista, jotka voidaan tunnistaa verkossa, vaikka niitä ei voitaisi suoraan hakea verkosta. Esimerkkinä voidaan käyttää tietoja saatavilla olevista kohteista verkkokaupassa, kuten tietoja ominaisuuksista, hinnoista ja saatavuudesta, tai vaikka kuvausta Web-käyttäjän mieltymyksistä informaation jakamiseen. (W3C 2004a.)

2.1 RDF tietomallin tarkoitus

RDF on tarkoitettu tilanteisiin, joissa tiedot käsittelee sovellus, ei ihminen. RDF tarjoaa yhteiset kehykset ilmaista nämä tiedot, jotta niitä voidaan vaihtaa sovellusten välillä ilman merkityksen katoamista. Koska se on yhteinen kehys, sovellusten suunnittelijat voivat hyödyntää yhteisten RDF-jäsentimien ja työkalujen saatavuutta. Kyky vaihtaa tietoja eri sovellusten välillä tarkoittaa sitä, että tietoja voidaan vaihtaa muihinkin sovelluksiin, kuin vain mille ne oli alunperin luotu. (W3C 2004a.)

RDF:ssä on yksinkertainen tietomalli, jota sovellusten on helppo käsitellä ja manipuloida. Tietomalli on riippumaton mistään erityisestä sarjoituksesta syntaksista. Predikaattien muoto RDF-tietomallissa on aina URI (W3C 2004c.)

Tietomallin erottaminen verkossa RDF dataa kuljettavasta syntaxista sallii RDF-järjestelmän päästä käsiksi ei-RDF tietolähteeseen. Tämän tehdäkseen RDF adapteri antaa ensin uniikit URI:t resursseille ei-RDF tietolähteessä, jonka jälkeen se luo resurssien ominaisuuksia kuvaavia lauseita. RDF:n erottaa olennaisesti XML:n puu-

mallisesta tietomallista sen URI:en käyttö yksiselitteisesti kuvata objekteja ja niiden välisten suhteiden ominaisuuksia. Tämä yksinkertainen, mutta tehokas mekanismi tukee yleistä lähestymistapaa informaation kuvaamiseen ja integroimiseen, koska se tarjoaa pienimmän yhteisen tekijän kaikille tietomalleille. (Decker, Mitra & Melnik 2002.)

RDF perustuu ajatukseen tunnistaa asioita käyttäen Web-tunnisteita (Uniform Resource Identifier eli URI), ja resurssien kuvaamiseen sekä yksinkertaisilla ominaisuuksilla, että ominaisuuksien arvoilla. Tämä mahdollistaa sen, että RDF voi kuvata resursseja, sekä niiden ominaisuuksia ja arvoja. (W3C 2014a.)

RDF on tarkoitettu yksinkertaiseksi tavaksi kuvata Web resursseja, esimerkiksi web-sivuja. (W3C 2014a.)

2.2 RDF tietomallin periaate

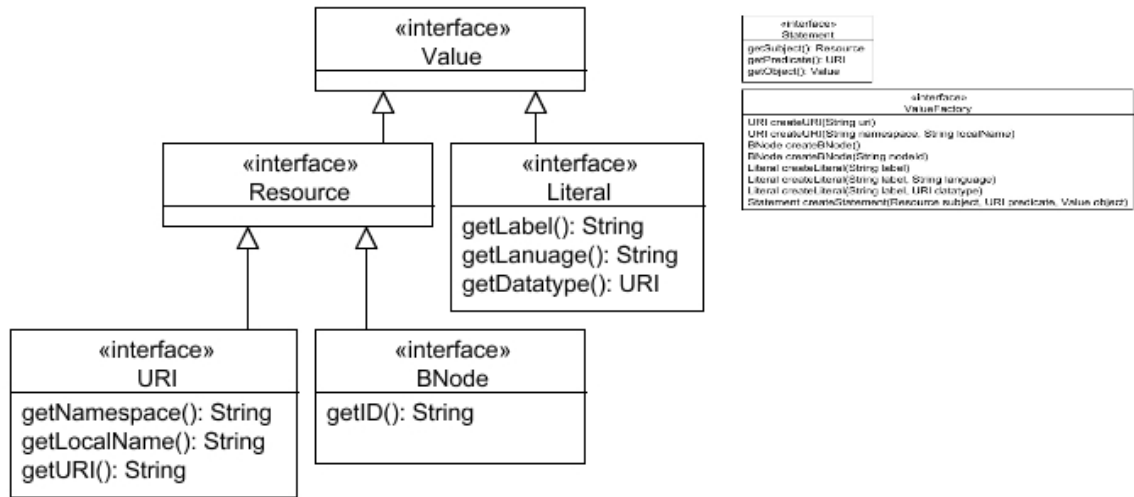
RDF perustuu siihen ajatukseen, että kuvatulla asialla on ominaisuuksia, joilla vuorostaan on arvoja, ja että resursseja voidaan kuvata tekemällä lausuntoja, jotka määrittävät nämä ominaisuudet ja arvot. RDF käyttää erityisestä terminologiaa kuvataksaan lausuntojen eri osia.

Osa, joka osoittaa sen, mistä lausunnossa on kyse, on subjekti. Lausunnon spesifioiman subjektin ominaisuuden osoittama osa on predikaatti. Predikaatin ominaisuuden arvon tunnistava osa on objekti. (W3C 2004a.)

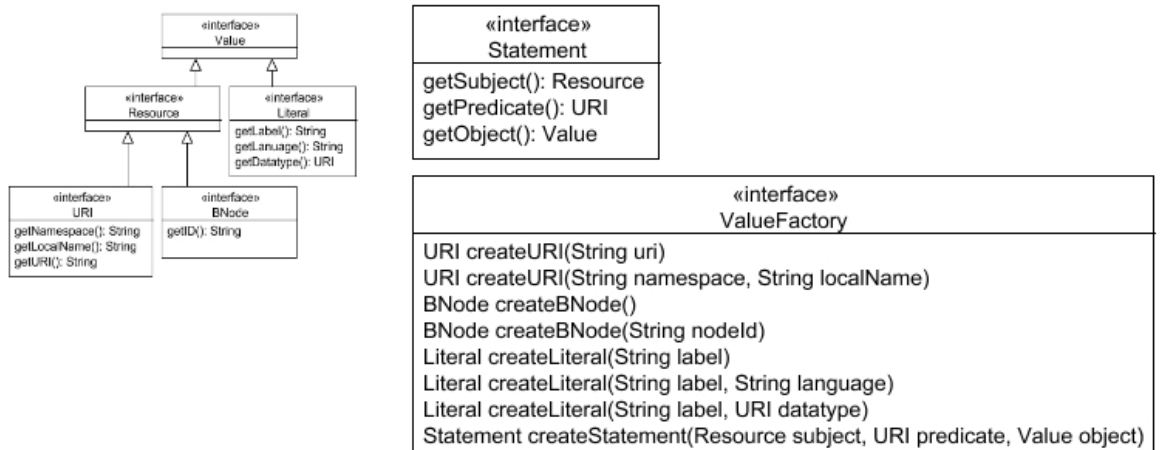
<http://www.esimerkki.fi/home.html> sivulla on luoja, jonka arvo on Eero Esimerkki

Kuva 1. Subjekti, predikaatti ja objekti

Kuvan esimerkissä subjekti on URL ”<http://www.esimerkki.fi/home.html>”, predikaatti on sana ”luoja”, ja objekti on nimi ”Eero Esimerkki”.



Kuva 2. RDF:n hierarkiaa (Vasemmalla Kuva 2. Oikealla kuva 3.)



Kuva 3. RDF:n hierarkiaa (Vasemmalla Kuva 2. Oikealla kuva 3.)

Kuvat 2 ja 3 ovat sama kuva. Kuvassa 2 on pienennetty kuvan oikea puoli ja kuvassa 3 on pienennetty vasen puoli, jotta kuvat mahtuvat kätevästi dokumenttiin. Kuvat 2 ja 3 näyttävät RDF hierarkian eli rakenteen. Value ja Statement arvot ovat ensimmäisinä. Niitä seuraa Resource, Literal ja ValueFactory arvot. (W3C 2014d.)

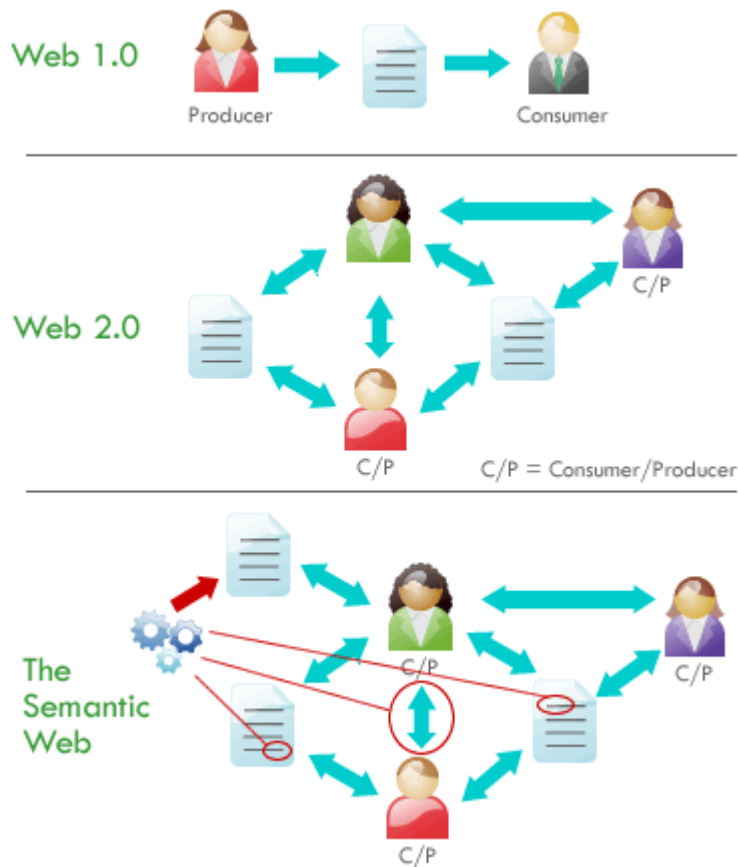
3 Semanttinen Web

3.1 Yleisesti

Semanttinen Web on visio automatisoidusta verkosta. Sitä ei ole olemassa, mutta sitä on kehitetty erilaisissa instansseissa jo 20 vuotta. Pohjana Semanttiselle Webille toimii W3C:n RDF-kehys (W3C 2004d.)

Semanttisessa Webissä haetaan tehokasta ja avointa tapaa löytää ja yhdistää eri tietolähteitä (Allemang & Hendler 2008, 19). Semanttinen Web on jaetulle tiedolle tehty hajautettu alusta, kun nykyinen verkko on jaetuille esityksille tehty hajautettu alusta. Nykyisessä verkossa tietokone ei ymmärrä mitään esimerkiksi wikipedia-sivun sisällöstä, vaan ainoastaan tulkitsee HTML:ää, kuvia, flashia ja niin edelleen. (Breitman, Casanova, Truszkowski 2007, 11-12.)

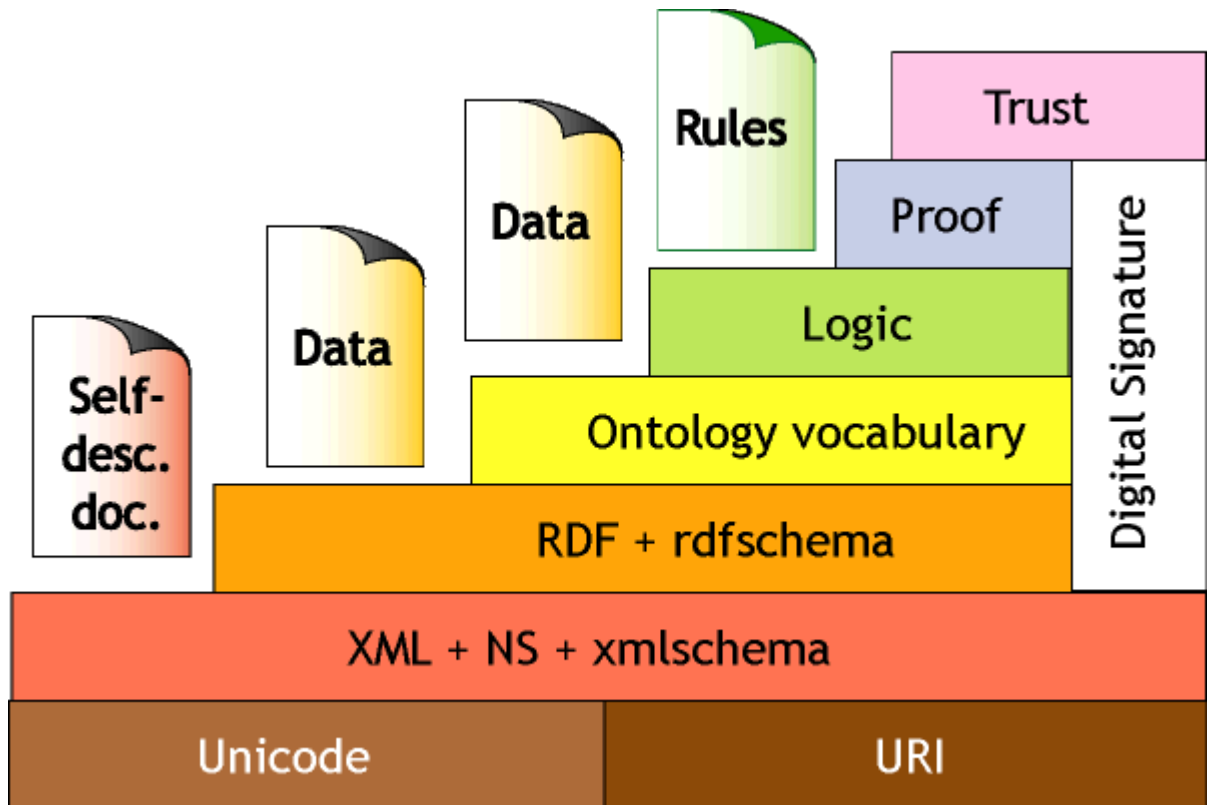
Semanttinen web pyrkii siihen, että tietokone osaisi tulkita esimerkiksi wikipedia sivun sisältämää tietoa eri parametreilla. Tietokone siis osaisi liikutella tiedon loogisia palasia mekaanisesti haluttujen tarkoituksien mukaisesti. (Breitman, Casanova, Truszkowski 2007, 12.)



Kuva 4. Semanttinen Web (WP-Content, 2007)

Semanttisen webin kolme keskeistä seikkaa:

1. Semanttisessa webissä olevien tietojen on kyettävä kuvaamaan tietoa joustavasti (kuva 4). Ei siis voida käyttää siististi pakattuja taulukoita, kuten relaatiotietokantoja, tai hierarkioita, kuten XML:ää.
2. Semanttisen Webin tietojen täytyy liittyä toisiinsa (kuva 4). Myyjän julkaisemat tiedot tuotteen hinnasta ja tiedot käyttäjän itsenäisesti julkaistusta tuotearviosta tarvitsevat indikaattorit siitä, että kyseessä on sama tuote. Pelkkä tuotenimike ei riitä. Semanttinen Web tarvitsee yleisesti ainutlaatuiset tunnistimet, jotka voidaan asettaa hajautetulla tavalla.
3. Väitteiden tekemiseen eri asioista käytetään sanastoja, mutta ne täytyy voida sekoittaa keskenään. Esimerkiksi sanastot TV-ohjelman faneilta ja kriitikoilta on voitava ottaa käyttöön saman tiedoston sisälle, puhuen samasta asiasta, kuten tietyn näyttelijän olemisesta useammassa ohjelmassa. (Appnel 2002.)



Kuva 5. Semanttisen Webin rakenne (Hendler, J. 2009)

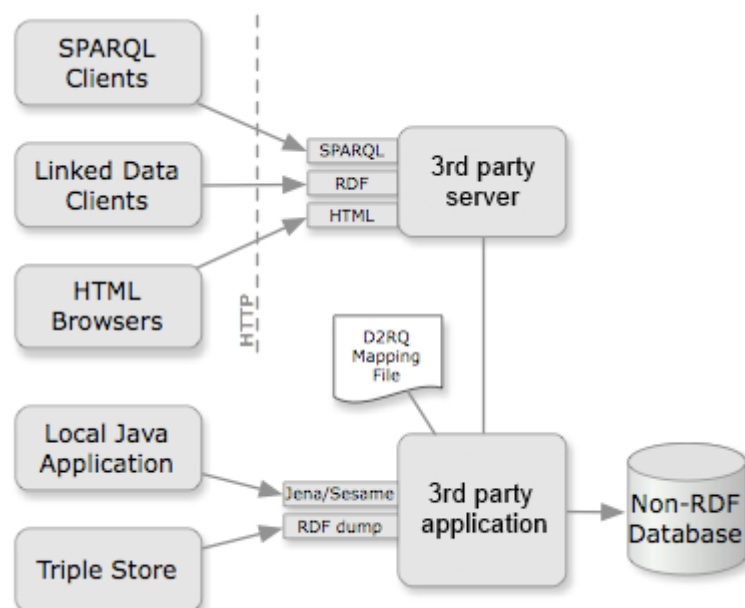
3.2 RDF Semanttisessa Webissä

Kuvan 5 mukaisesti RDF ja sen laajennus RDF Schema ovat Semanttisen Webin ontologian ja XML rakenteiden, syntaksin ja laajennusten välissä.

RDF on yleinen tapa hajottaa tietoa pieniksi paloiksi, säännöillä niiden palasten semantiikasta tai merkityksestä. Menetelmän on oltava sekä niin yksinkertainen, että se voi ilmaista kaikkia asioita, että rakenteeltaan sellainen että tietokoneen sovellukset voivat tehdä hyödyllisiä asioita RDF:ssä ilmaistuilla tiedoilla. Palaset voivat olla monessa eri muodossa ja silti säilyttää alkuperäiset tiedot ja rakenteen, aivan kuten sama asia voidaan ilmaista eri kielillä tai implementoida samaan tietorakenteeseen monin eri tavoin. (Hebeler, Fisher, Blace, Perez-Lopez & Dean 2009, 68-69.)

Myös relaatiotietokantoja voidaan käyttää, mutta tämä vaatii erillisen ulkopuolisen ratkaisun. Näissä on normaalisti oma palvelin ja sovellus kuten kuva 6 esittää. Tällöin tieto siirtyy palvelimelta sovellukseen, joka muuntaa tiedon yhteensopivaan muotoon. Relatiotietokannoilla saadaan tieto yksinkertaisempaan muotoon, mutta se on silti

epäkätevä ja hankala tapa käsitellä Semanttisen Webin tietoa (Hebeler, Fisher, Blace, Perez-Lopez & Dean 2009, 66).



Kuva 6. Kolmannen osapuolen ohjelman käyttö relaatiotietokannan mahdollistamiseksi.

3.3 Miksi RDF-syntaksi on valittu XML:n sijasta

3.3.1 Samankaltaisuudet

RDF:n omaa syntaxia voidaan verrata XML:ään joillain tavoilla. XML on myös suunniteltu yksinkertaiseksi ja soveltamaan kaikenlaisia tietoja. XML on myös enemmän kuin pelkkä tiedostomuoto. Se on perusta hierarkkisten ja itsenäisten tiedostojen käsittelylle. Olivatpa ne sitten levyllä perus sulkukautta formaatissa tallennettavia tai DOM API:n kautta muistista haettavia tiedostoja. (Hebeler, Fisher, Blace, Perez-Lopez & Dean 2009, 66-67.)

3.3.2 Erot

Mikä erottaa RDF:n syntaksin XML:stä on, että RDF on suunniteltu edustamaan tietoa jaetussa maailmassa. Se että RDF on suunniteltu tietoa, eikä dataa varten, tarkoittaa RDF:n keskittyvän erityisesti merkitykseen. Kaikki RDF:ssä mainittu tarkoittaa jotain. Se voi olla viittaus johonkin tästä maailmasta, kuten henkilöön tai elokuvaan, tai se voi olla abstrakti käsite, kuten toisen kanssa ystäväneä olemisen olotilan kuvaus. Laittamalla

kolme tällaista kokonaisuutta yhteen, kertoo RDF-standardi miten päästä siitä johtopäätökseen. RDF:n päälle rakennetut standardit, kuten RDFS ja OWL, lisäävät RDF-semantiikkaan loogista päättelykykyä. (Hebeler, Fisher, Blace, Perez-Lopez & Dean 2009, 67-68.)

3.3.3 Vahvuudet

RDF toimii hyvin hajautetulle tiedolle. RDF-sovellukset voivat koota RDF tiedostoja, joita eri ihmiset ympäri Internetiä ovat julkaisseet ja helposti oppia niistä uusia asioita ilman, että mikään yksittäinen asiakirja kertoo. Se tekee tämän kahdella tavalla, yhdistämällä asiakirjat yhteen yhteisen käytössä olevan sanaston mukaan, ja antamalla minkä tahansa asiakirjan käyttää mitä tahansa sanastoa. Tällä saadaan valtava joustavuus ilmaista asioita monenlaisista asioista, koska hyödynnetään tietoa monista eri lähteistä. RDF:n oma syntaksi on siis XML:ää sopivampi ratkaisu Semanttisen Webin tarpeille. (Hebeler, Fisher, Blace, Perez-Lopez & Dean 2009, 67-68.)

RDF:n jatkuva kehitys on tuonut sille monia eri syntaxeja, joista uudemmat sallivat jo niin paljon erilaisia asioita että ne ovat huomattavasti monimutkaisempia kuin alkuperäinen yksinkertainen tietomalli. Tällaisia syntaxeja ovat esimerkiksi RDF/XML, N3, Turtle ja RDFa. Turtlen syntaksi on ihmisläheinen ja ainoa joka on keskittynyt pelkästään RDF:ään. (W3C 2012b).

RDF on suunniteltu antamaan tieto minimaalisen rajoittavalla ja erityisen joustavalla tavalla. Sitä voidaan käyttää yksittäisissä ohjelmissa, joissa yksilöllisesti suunnitellut formaatit voivat olla suoraviivaisempia ja helpommin ymmärrettävissä, mutta RDF tarjoaa yleisesti paremman arvon tiedon jakamisesta. Tiedon arvo siis kasvaa kasvamistaan kun se tulee useampien ohjelmien saataville koko Internetin laajuudella. (Allemang & Hendler 2008, 27.)

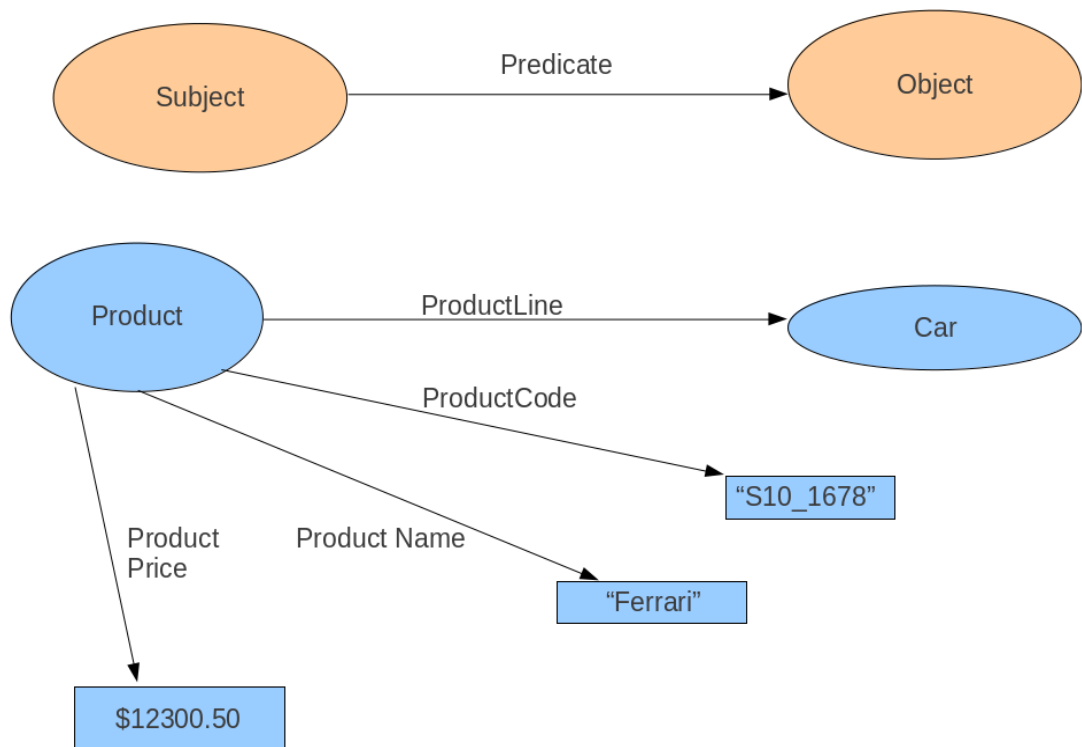
4 RDF:n suunnittelun päämäärät

RDF on suunniteltu sisältämään seuraavat asiat:

- Yksinkertainen tietomalli.
- Sisältää virallisen semantiikan ja todistettavan päättelyn.
- Käyttää laajennettavissa olevaa URI-sanastoa.
- Käyttää XML-pohjaista syntaxia.
- Tukee XML schema data tyyppiä.
- Sallii kenen tahansa tekemän lausuntoja mistä tahansa resurssista.

4.1 Yksinkertainen tietomalli

Yksinkertainen tietomalli on RDF:n tärkein yksityiskohta, jonka ympärille koko asia on rakennettu. RDF vie automaattisesti kohti selkeyttä. Tietomallin formaatti on subjekti, predikaatti, objekti. (W3C 2014a.) Kuva 7 selkeyttää tietomallia, kun mukana on monia arvoja.



Kuva 7. Subjekti – predikaatti – objekti monimutkaisempuna kokonaisuutena

4.2 RDF:n muut päämäärät

RDF:llä on muodollinen semantiikka, joka tarjoaa luotettavan perustan päättelylle RDF lauseen merkityksestä. Se tukee erityisen tiukasti määrittelyitä loogisen seurauksen käsitteistä, jotka tarjoavat pohjan RDF-muotoisen tiedon luotettavien päättelysääntöjen määrittelylle.

RDF:ssä on laajennettavissa oleva URI-pohjainen sanasto. URI viittauksia käytetään kaikenlaisten asioiden nimeämisessä RDF:ssä. Sanasto on täysin laajennettavissa oleva, koska se perustuu URI:iin, joissa on vaihtoehtoiset fragmenttitunnisteet (tunnetaan termillä URI viittaukset tai URIsrefs). URI on merkkijono, jota käytetään resurssin nimeämiseen tai sijainnin osoittamiseen internetissä. Subjektien ja predikaattien muoto RDF-tietomallissa on aina URI. Muut arvot, joita näkyy RDF datassa ovat kirjaimellisia. (W3C 2014b.)

RDF:ssä on suositeltu XML sarjoitusmuoto, RDF syntaksi, jota voidaan käyttää tietojen vaihdon tietomallin koodaamiseen sovelluksissa.

RDF:ssä voidaan käyttää XML Scheman tietotyyppin, XML - SCHEMA2, mukaisia arvoja. Tämä helpottaa tietojen vaihtoa RDF:n ja muiden XML sovellusten välillä. (W3C 2014c.)

5 RDF tietokannat

RDF tietokanta on SPARQL pohjainen, omantyyppisensä tietokantaformaatti, jota suositaan semanttisen webin ratkaisuisissa. Se varastoi tripletejä, minkä vuoksi sitä kutsutaan myös Triplettivarastoksi (Triple store). RDF tietokannat soveltuvat semanttiseen webiin huomattavasti paremmin, kuin relaatiotietokannat. Eroja on muunmuassa tietovarastoinnissa, tiedon saatavuudessa (SQL lukitsee osan), yhdenmukaisuudessa, hinnassa, performanssissa ryhmitetylle datalle, välimuistissa ja siinä miten paljon joustavuutta ne tarjoavat käyttäjille ja kehittäjille. (Bergman 2009.)

RDF-tietokantojen tärkeä tehtävä ei ole pelkästään olla sopiva malliltaan, vaan myös nopea (Rusher 2003).

RDF on ainoa NoSQL ratkaisu, joka on standardoitu. Se on siis käytössä myös Semanttisen Webin ulkopuolella, vaihtoehtona SQL relaatiotietokannoille. (Bendiken 2010.)

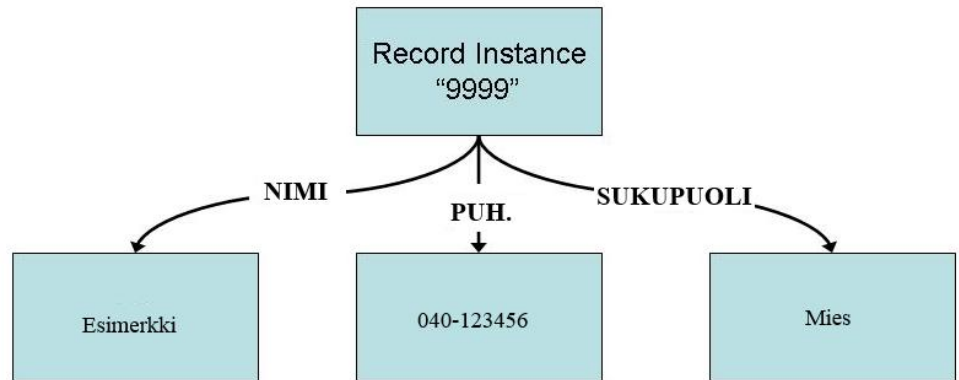
Triplettivarastoja on kolmea eri tyyppiä ja ne jaetaan sen arkkitehtuurin pohjalta miten ne on tarkoitus implementoida. Tyypit ovat muistissa oleva, natiivi ja ei-natiivi, ei muistissa oleva. (The National Center for Biomedical Ontology 2012.)

Kuva 8 selkeyttää eroa relaatiotietokannan ja RDF-tietokannan välillä, jossa tietomalli on täysin erilainen.

RELAATIO

ID	NIMI	PUH.	SUKUPUOLI
9999	Esimerkki	040-123456	Mies

RDF



Kuva 8. Relaatiotietokannan ja RDF:n ero

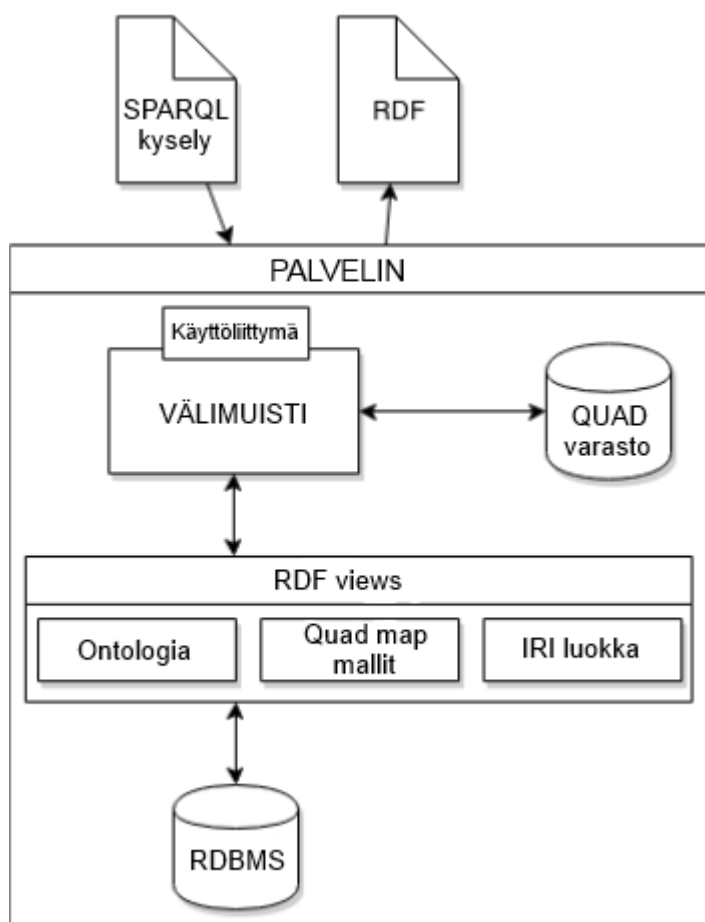
5.1 Tiedon saatavuus ja yhdenmukaisuus

SPARQL:stä löytyy työkalu, jolla voi kartoittaa relaatiotietokanta dataa RDF muotoon. Tällä tavalla saadaan vanhemmista instansseista data siirrettyä uudempaan RDF-tietokanta kokonaisuuteen. Tätä ei ole SQL:ssä vaan se on vain SPARQL:n toiminto. SPARQL:llä voidaan ryhmittää dataa menettämättä erilaisia etsimistöimintoja, joilla tiedon etsintä tauluista nopeutuu valtavasti. (W3C 2012c.)

SPARQL on erittäin pitkälle standardoitu ja se on yhdenmukainen käyttö lähes mitä tahansa ohjelmistoa sen tekemiseen. Vertauksen voi vetää SQL:ään, jossa eri ohjelmistojen välillä on suuriakin eroja eli esimerkiksi MySQL:ää osaava ei suinkaan suoraan hallitse Postgresql tai DB2 ohjelmistoja. SPARQL:aa kohtalaisesti osaava hallitsee sen standardoinnin takia, sen millä tahansa ohjelmistolla. (Di Nunzio, Di Gregorio, Rizzo & Servetti 2010, 151-163.)

5.2 Välimuisti puskuroimaan SPARQL:ää

Verkkosivuille on olemassa monia välimuisti ratkaisuja ja SPARQL:llä saadaan paljonkin irti siitä että tietokannan ja verkkosivun väliin asennetaan ulkoinen välimuisti. Tällöin tiedon käsittely nopeutuu huomattavasti. Kuva 9 esittää välimuistin sijaintia. Välimuisti varastoi pienen määrän tärkeää tietoa ja jakelee sitä tehokkaammin kuin ratkaisu, jossa kaikki tieto haetaan aina tietokannasta asti.



Kuva 9. Välimuisti

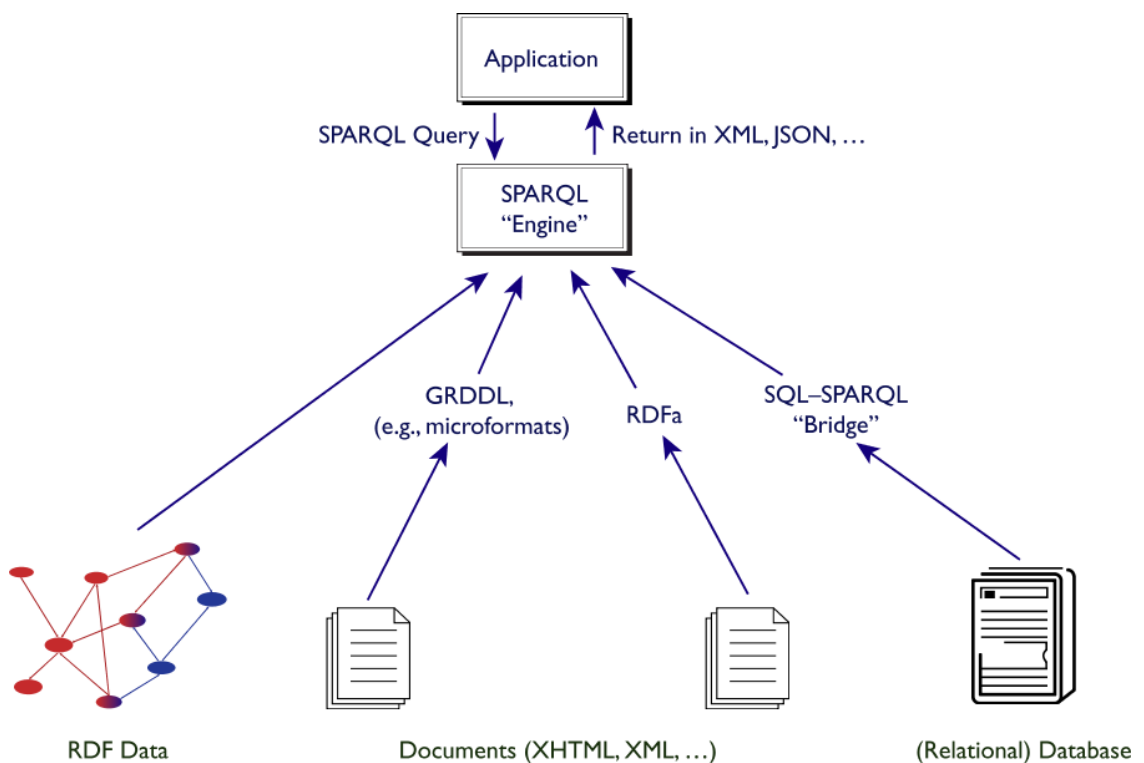
5.3 Joustavuus RDF-tietokannoissa tietomallin ansiosta

RDF:n graph tietomallin joustavuus näkyy tiedon manipuloinnissa ja tiedon jatkamisessa. RDF:ssä käytetty schema on OWL ontologia ja se tuo lisää joustavuutta ja rakeisuutta tiedon esittämiseen. Loogisena formalismina OWL mahdollistaa yksinkertaisten päättelymekanismien soveltamisen (eli luokituksen), jotta määritelmiä voidaan lisätä ontologiaan yhdistämään ja ryhmittämään toiminnan jälkiä automaattisesti. (W3C 2004b.)

6 SPARQL

SPARQL on RDF tietokantojen kyselykieli. Siinä on neljän tyyppin kyselyitä: SELECT, ASK, DESCRIBE ja CONSTRUCT (W3C 2008). SPARQL:stä tuli W3C:n RDF Data Access Working Groupin (DAWG) toimesta standardi. Sitä pidetään yhtenä Semanttisen Webin keskeisistä teknologioista. SPARQL 1.0:sta tuli W3C:n virallinen suositus 15.1.2008 ja SPARQL 1.1:stä maaliskuussa 2013.

SPARQL:stä on olemassa toteutuksia useille ohjelmointi kielille. Jo toukokuussa vuonna 2008 Sir Tim Berners-Lee sanoi SPARQL:n tekevän suuren muutoksen siinä miten webistä saadaan koneiden luettavissa oleva.



Kuva 10. Esimerkki eri tietotyypeistä, joita SPARQL voi käsitellä. (Herman, I. 2007.)

SPARQL ”noudattaa” SPARQL Protocol for RDF:ää sekä SPARQL Query Results XML Formaattia. (W3C 2008a.)

Kuva 10 on esimerkki SPARQL:n monipuolisesta toiminnallisuudesta. SPARQL käsittelee RDF dataa, dokumentteja monissa formaateissa sekä relaatiotietokantoja kunhan välissä on niin sanottu silta, jolla data saadaan yhteneväksi. (Herman, I. 2007.)

SPARQL Protocol for RDF on kehittäjille suunnattu, W3C:n kehittämä ohjepaketti, joka tarjoaa standardin kaikkien käyttöön. Sen tehtävä on auttaa kehittäjiä luomaan yhtenäinen paketti SPARQL:llä ja yhtenäistää eri projektien tulokset keskenään. Siinä on ohjeistukset esimerkiksi HTTP ja SOAP sidoksille (bindings) sekä suositus sille missä formaatissa kyselyiden kannattaisi olla. (W3C 2008b.)

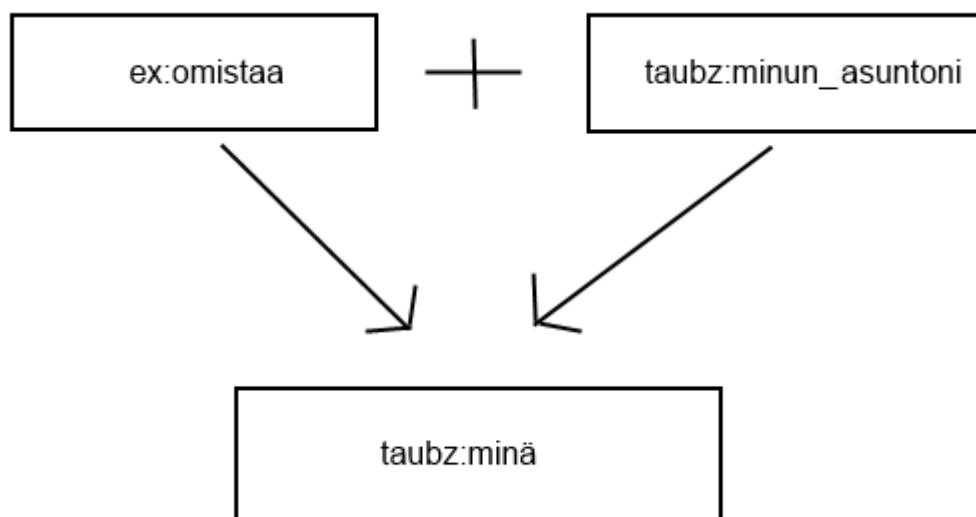
6.1 Kyselyt

SPARQL Query Results XML Format on tarkennus sille, millä tavalla XML asiakirjan olisi hyvä tuoda esille tulokset SPARQL SELECT ja ASK kyselyissä. Se on siis ohje XML asiakirjan formaatille, käyden läpi esimerkiksi otsikon ja tulosten suositellun ulkonäön.

Tietokannoista esimerkkinä SELECT-kysely on lähinnä SQL:ää siinä miten käyttäjä tekee kyselyn ja saa taulukkomuotoisen vastauksen. Siihen yhtäläisyydet kuitenkin loppuvat (kuva 11). SPARQL:n SELECT kyselyssä kysytään resursseista (resources). Esimerkiksi kysely voisi kysyä mitkä resurssit ”omistaa” ”minun asuntoni”. (Tauberer 2008.)

RDF minun asunnostani		
@prefix ex: <http://oppiari.org/> .		
@prefix taubz: <http://jtk.snif.info/index.html#> .		
taubz:minä	ex:omistaa	taubz:minun_asuntoni
taubz:minä	ex:omistaa	taubz:minun_tietokoneeni
taubz:minun_asuntoni	ex:sisältää	taubz:minun_tietokoneeni
taubz:minun_asuntoni	ex:sisältää	taubz:kaverin_rouinat
taubz:minun_asuntoni	ex:sijainti	<http://esimerkki.fi/helsinki>
taubz:minä	ex:omistaa	taubz:minun_pöytäni
taubz:minun_pöytäni	ex:sisältää	taubz:minun_kynäni_ja_kumini

Kuva 11. esimerkki yksinkertaisesta tietokannasta



Kuva 12. ”Omistaa” ”Minun asuntoni”

Vastauksena kysymykseen mitkä resurssit ”omistaa” ”minun asuntoni” kuvan 11 tietokannasta, saataisiin ”taubz:minä” (kuva 12).

Kysely taubz:minun_asuntoni ex: sisältää x (x = kyselyn muuttuja, kuten mitä, kuka, missä jne.), antaisi kuitenkin tulokseksi 2 resurssia minun_tietokoneeni ja kaverin_rouinat. Tässä on ero SQL:ään, jossa tulokset olisivat taulun rivejä, mutta SPARQL:ssä jokainen rivi on resurssi. (Tauberer 2008.)

CONSTRUCT kyselyä käytetään tietojen poimimiseen SPARQL endpoint ohjelmasta ja sitten muuttamiseen tulokset validiin RDF formaattiin. (Tauberer 2008.)

ASK kyselyä käytetään kun halutaan yksinkertainen True / False tulos SPARQL endpoint ohjelmalla. (Tauberer 2008.)

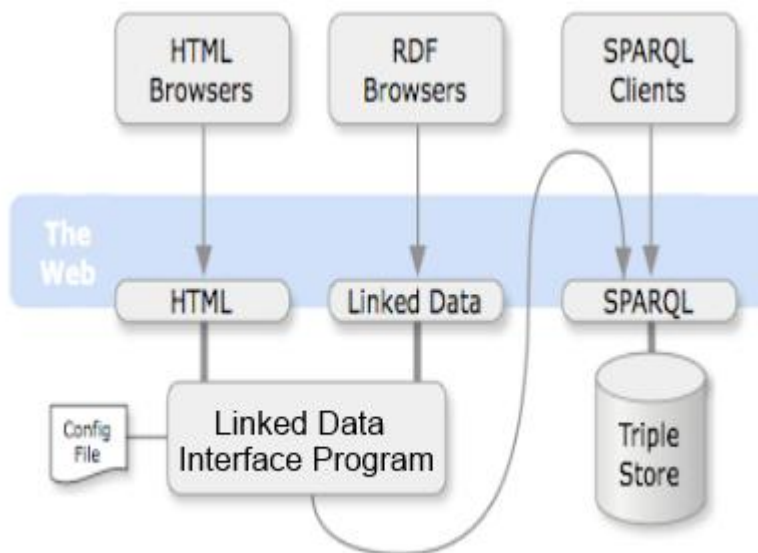
DESCRIBE kyselyä käytetään erottamaan RDF kuvaaja (graph) SPARQL endpoint ohjelmasta, jonka sisältö riippuu siitä, minkä endpoint ohjelma tulkitsee ylläpitäjälle hyödylliseksi tiedoksi. (Tauberer 2008.)

Kukin näistä kyselyistä tarvitsee WHERE kyselyn rajoittamaan tai rajaamaan kyselyn tulosta, joskin DESCRIBE kyselyn kanssa WHERE kysely ei ole pakollinen. SPARQL 1.1 tarkoittaa tietokannan päivittämisen kielen useilla uusilla kysely tavoilla.

SPARQL:ssä on siis SQL:n tapaan WHERE kysely. SPARQL:ssä se on kuitenkin graph kuvio, jossa voidaan käyttää URIa, tyhjiä solmuja (blank node), literaaleja sekä muuttujia. (W3C 2008a.)

SPARQL:ään on myös laajennuksia. Ne ovat lisäominaisuuksia, jotka ovat ominaisuuksiltaan SPARQL kyselykielen tai SPARQL protokollan ulkopuolella. Nämä ovat usein ominaisuuksia, joita kehittäjät tuovat esille asioina joita voitaisiin lisätä viralliseen versioon. Ennen virallistamista niitä voidaan kuitenkin käyttää laajennuksina. Tällaisia ovat esimerkiksi lisä kyselykomennot ja komentojen laajentamiset tai syntaksin laajennukset. (W3C 2008a.)

SPARQL:llä käsitellään yleensä tietoa vain SPARQL Clientin ja Tripletti varaston välillä. On kuitenkin mahdollista käyttää lisäohjelmia, joilla tieto saadaan erinähteistä suoraan SPARQL muotoon ja sitä kautta tripletti varastoon. (kuva 13). Niillä voidaan siirtää tietoa HTML ja RDF selaimista SPARQL formaattiin ja sitä kautta tietokantaan. (W3C 2008a.)



Kuva 13. Missä välissä lisäohjelma hääriä tietoa SPARQL:lle.

6.2 RDF Dataset

Seuraavaksi käymme läpi hiukan RDF aineistoa eli datasettiä ja kuinka se ilmenee tietokannoissa. RDF tietomalli ilmaisee tietoa kaavioina (graphs), jotka koostuvat subjekti, predikaatti, objekti – tripleteistä. Useat RDF tietovarastot pitävät sisällään useita RDF kaavioita ja tallentavat tietoa joka ikisestä kaaviosta, antaen mahdollisuuden ohjelmalle tehdä kyselyjä, joissa on mukana useampi kuin vain yksi kaavio. (W3C 2008a.)

SPARQL kysely suoritetaan RDF Datasetille mikä edustaa kokoelmaa kaavioista. RDF Dataset sisältää yhden vakio kaavion (default graph) sekä nollasta useampaan nimettyä kaaviota (named graphs). Vakio kaaviolla ei ole nimeä. Nimetyt kaaviot tunnistetaan IRillä. SPARQL kysely voi etsiä tietoa eri kyselymallien osilla erilaisista kaavioista, eikä sitä ole rajoitettu. Kysely voi koskea nollaa nimellistä kaaviota, tai pelkkiä nimellisiä kaavioita jättäen vakio kaavion huomiotta. (W3C 2008a.)

```
# Default graph
@prefix dc: <http://tamatyo.uni/dc/elements/1.1/> .

<http://esimerkki.fi/lauri>    dc:publisher  "Lauri" .
<http://esimerkki.fi/eero>    dc:publisher  "Eero" .
```

```
# Named graph: http://esimerkki.fi/lauri
@prefix foaf: <http://xmlns.com/foaf/0.1/> .

_:a foaf:name "Lauri" .
_:a foaf:mbox <mailto:lauri@yritys.esimerkki.fi> .
```

```
# Named graph: http://esimerkki.fi/eero
@prefix foaf: <http://xmlns.com/foaf/0.1/> .

_:a foaf:name "Eero" .
_:a foaf:mbox <mailto:eero@hhamk.fi> .
```

Kuva 14. RDF Dataset

Esimerkkinä kuvassa 14 vakio kaavio sisältää kahden nimellisen kaavion julkaisijan (publisher) nimet. Tripletit nimellisissä kaavioissa eivät ole näkyvissä tämän esimerkin vakio kaaviossa. (W3C 2008a.)

RDF dataa voidaan yhdistää. Tällöin puhutaan termillä RDF merge. Tämä on käytännössä esimerkiksi nimellisten kaavioiden tiedon yhdistämistä vakio kaavioon. (W3C 2008a.)

```
# Default graph
@prefix dc: <http://tamatyo.uni/dc/elements/1.1/> .

_:x foaf:name "Lauri" .
_:x foaf:mbox <mailto:lauri@yritys.esimerkki.fi> .

_:y foaf:name "Eero" .
_:y foaf:mbox <mailto:eero@hhamk.fi> .
```

```
# Named graph: http://esimerkki.fi/lauri
@prefix foaf: <http://xmlns.com/foaf/0.1/> .

_:a foaf:name "Lauri" .
_:a foaf:mbox <mailto:lauri@yritys.esimerkki.fi> .
```

```
# Named graph: http://esimerkki.fi/eero
@prefix foaf: <http://xmlns.com/foaf/0.1/> .

_:a foaf:name "Eero" .
_:a foaf:mbox <mailto:eero@hhamk.fi> .
```

Kuva 15. Nimellisten kaavioiden tieto on vakio kaaviossa.

Kuvassa 15 näkyy kuinka vakio kaaviossa on tyhjät solmut merkitty uudelleen, jotta ne pysyvät erillisinä.

SPARQL kysely voi myös spesifioida mitä tiettyä tai tiettyjä datasettiä halutaan käyttää. Tällöin käytetään FROM ja FROM NAMED komentoja. Tällä keinolla saadaan tehtyä tarkempia kyselyitä. (W3C 2008a.)

Varsinainen kysely tapahtuu siten, että kaavion avainsanaa verrataan malleilla nimellisten kaavioiden sisältöön. Kaavio jolla etsitään, voi käyttää IRIä yhden kaavion valitsemiseen, tai käyttää muuttujaa jolla voidaan etsiä tietoa IRIn parametreilla koko RDF datasetin kaikista nimellisistä kaavioista. (W3C 2008a.)

7 Päätelmät

Tässä on päätelmiä, jotka on tehty teorian pohjalta.

Helpottaakseen toimintaa Internetin mittakaavassa, on RDF avoimen maailman puite, joka sallii kenen tahansa tehdä lausuntoja mistä tahansa resurssista.

Yleisesti ei oleteta, että täydellisiä tietoja kaikista resursseista olisi saatavilla. RDF ei estä ketään tekemästä väitteitä, jotka ovat järjettömiä tai ristiriidassa muiden tehtyjen lausuntojen kanssa tai ristiriidassa sen maailman kanssa, joka kontekstissa ymmärretään. Tämä on haasteellista RDF-suunnittelijoille.

SPARQL pohjaisilla tietojärjestelmillä käyttökustannukset ovat alhaisemmat aloilla, joilla tieto muuttuu nopeasti. Integroituville tiedoille ovat rakennuskustannukset pienemmät. SPARQL mahdollistaa julkisia ja yksityisiä tietoja sisältävien tietokantojen yhdistelyä yrityksessä, niin että voidaan vaikka luoda näkymä siitä keiden yrityksestä potkut saaneiden henkilöiden sähköpostitilit ovat yhä edelleen aktiivisia.

Koska SPARQL:ssä on kuvaaja (graph) tripletin vieressä, voidaan sillä rakentaa avain arvo- tai asiakirjavarasto. Tällaisissa tapauksissa avain on kuvaajan id ja arvo on itse tripletissä. Tämä voi antaa paljon paremman performanssin kuin silkkä relaatiotietokantalähestymistapa. Tämä kuitenkin riippuu varaston implementaatiosta. Kaikki mahdolliset ratkaisut eivät ole täydellisiä kaikkiin vaihtoehtoihin. Yrityksen sisäinen keinojen standardointi on avain tällaisessa ryhmyksessä.

SPARQL tuo hyötyä myös sillä, että sillä on paljon pienempi vaihtokustannus verrattuna kilpailijoihinsa. Tietovarastointi on RDF:n tripleteillä (subjekti, predikaatti, objekti) on kätevää, vaikka tiedostotyyppi vaihtuisi. Relaatiotietokannassa pitäisi muuttaa parseri, tietokantataulukot ja kyselyt. RDF-tietokannassa tarvitsee muuttaa vain kyselyt. ODBC:n avulla voidaan excel-tiedostoistakin siirtää data RDF-tietokantaan.

RDF-tietokantojen ylläpito on siis kätevämpää ja vähemmän vaativaa. Sen tuloksena on todennäköinen ylläpidon kustannusten halpeneminen. Tämä on tärkeää, koska kehittäjiä ja hankkeiden johtajia täytyy tehdä päätökset tällaisista ratkaisuista etukäteen.

8 Johtopäätökset

Yhteenveto, joka koostuu johtopäätöksistä joihin opinnäytetyötä tehdessä on tultu. Työn arviointi on yhteenvetoon perässä.

8.1 Yhteenveto

RDF on sovellusten välisen tiedon vaihdon kieli. Sitä käytetään erityisesti Web-ympäristössä. RDF:n tärkein yksityiskohta on sen yksinkertainen tietomalli, subjekti, predikaatti, objekti. RDF ei rajoita tietomallinsa esitysmuotoa, joten useat eri formaatit ovat avoimena RDF-pohjaiselle tiedolle.

RDF:n keskeisiä tekijöitä ja sen kehitykseen vaikuttavia asioita:

- Webin metadata: Tarjoaa informaatiota verkkoresursseista ja niitä käyttävistä järjestelmistä (Esimerkiksi: sisällön arviointi, saatavuuden kuvaukset, yksityisyyden preferenssit, jne.)
- Ohjelmat jotka vaativat avoimia, mieluummin kuin rajoitettuja tietomalleja. (Esimerkiksi aktiviteettien järjestely, organisaatioprosessien kuvaaminen, verkkoresurssien selitykset, jne.)
- Koneen prosessoitavissa olevan informaation tekeminen (ohjelma data). Mitä Internet on tehnyt hypertextille, eli sallinut datan prosessoinnin sen luomisympäristön ulkopuolella tavalla, joka toimii koko Internetin laajuudella.
- Ohjelmien välinen toimivuus: Useiden ohjelmien datan yhdistäminen, jotta päästään uuteen informaatioon.
- Ohjelmistoagenttien automatisointi prosessoida verkko-tietoa: Internet tulee sisältämään yhä enemmän tietoa joka on koneiden luettavissa ja RDF tarjoaa maailmanlaajuisen kielen sille prosessille.

RDF:ää ja XML:ää vertaillaan paljon, koska niillä on paljon yhtäläisyyksiä. Niiden fundamentaalisin ero on kuitenkin se että RDF on tietomalli, ja XML on

serialisaatioformaatti. RDF:n pointti on siis malli sille kuinka säilöä tietoa, ja XML:n pointti on itse tapa muuntaa haluttu tieto säilytettäväksi sopivaan muotoon.

RDF on käytössä erilaisissa Web-ratkaisuissa, mutta se on erityisesti suunniteltu Semanttisen Webin ratkaisuihin.

RDF syntaxia on kehitetty eteenpäin nimenomaan Semanttisen Webin kyselykieleksi. XML ei soveltunut erityisen hyvin Semanttisen Webin haasteisiin, joten RDF:ää on muunmuassa kehitetty eteenpäin varta vasten vastaamaan niihin tarpeisiin.

Loppujen lopuksi kumpaa tahansa voi käyttää monissa eri tilanteissa, niiden suurin ero ei ole itse syntaksi vaan se, kuinka käyttäjä haluaa alun alkaen tiedon esittää.

Semanttista Webiä pidetään seuraavana Internetin asteena tai Internetin versiona johon tulisi pyrkiä. RDF kehitetään sen iästä huolimatta herkeämättä koska parempaa vaihtoehtoa sitä syrjäyttämään ei ole kukaan keksinyt. Semanttinen Web on kuitenkin vielä enemmän vision asteella, joten yhtään minkään kehitys siihen liittyen on vaivalloista ja hidasta.

Siitä huolimatta, että RDF itsessään on yksinkertainen tietomallissaan, ovat sen käyttö ja käyttöönotto mahdollisesti erittäin monimutkaista.

8.2 Työn arviointi

Tämä opinnäytetyö pyrki vastaamaan kysymyksiin mikä RDF on, miksi se on olemassa ja miksi sitä kehitetään yhä edelleen. Mielestäni olen löytänyt erilaisia kattavia vastauksia kaikkiin tutkimuskysymyksiin, joihin erityisesti yhteenvedossa vastaan.

Vaikeinta oli löytää relevanttia tietoa. Iso osa RDF-tiedosta on hyvin erityisesti rajattua. Tällä tarkoitan sitä että useat tahot, jotka kehittävät RDF:ää, kehittävät sitä omaan tarkoitukseensa. Silloin sen formaatti on erittäin eksakti, eikä siitä saa helposti yleistä kuvaa, jota voisi tällaisessa opinnäyteydessä käyttää hyväksi.

Aiheena RDF on erittäin mielenkiintoinen ja monimutkainen. Tätä työstäessä opin erittäin paljon sen eri seikoista ja Semanttisesta Webistä. Toivottavasti tämän tekemisestä on hyötyä jollain lailla tulevaisuudessa, jos ei suoraan minulle, niin jollekin muulle RDF:stä kiinnostuneelle. Tietoa siitä nyt ainakin on.

Lähteet

Bendiken, A. 2010. How RDF Databases Differ from Other NoSQL Solutions. Luettavissa: <http://blog.datagraph.org/2010/04/rdf-nosql-diff>. Luettu: 15.1.2014.

The National Center for Biomedical Ontology. 2012. Comparison of Triple Stores. Luettavissa: http://www.bioontology.org/wiki/images/6/6a/Triple_Stores.pdf. Luettu: 17.1.2014.

Rusher, J. 2003. Triple Store. Luettavissa: <http://www.w3.org/2001/sw/Europe/events/20031113-storage/positions/rusher.html>. Luettu: 18.1.2014.

Tauberer, J. 2008. What is RDF and what is it good for? Luettavissa: <http://www.rdfabout.com/intro/#RDF%20Schema%20%28RDFS%29>. Luettu: 2.11.2013.

Hebeler, J., Fisher, M., Blace, R., Perez-Lopez, A., Dean, M. 2009. Semantic web programming. Wiley.

Breitman, K., Casanova, MA., Truszkowski, W. 2007. Semantic Web : concepts, technologies and applications. NASA Monographs in Systems and Software Engineering. Springer Publishing. New York.

Di Nunzio, P., Di Gregorio, F., Rizzo & G., Servetti, A. 2010. Reliable SPARQL queries with consistent results over P2P-shared RDF storage. Politecnico di Torino.

Berners-Lee, T. 1998. What the semantic Web isn't but can represent. Luettavissa: <http://www.w3.org/DesignIssues/RDFnot.html>. Luettu: 1.12.2013.

Allemang, D. & Hendler, J. 2008. Semantic Web for the Working Ontologist: Elective Modeling in RDFS and OWL. Morgan Kaufmann. Massachusetts.

Appnel, T. 2002. What's wrong with RDF? Luettavissa:
http://www.oreillynet.com/xml/blog/2002/11/whats_wrong_with_rdf.html. Luettu:
4.12.2013.

Lassila, O. 1997. Introduction to rdf metadata. Luettavissa:
<http://www.w3.org/TR/NOTE-rdf-simple-intro>. Luettu: 29.11.2013.

Miller, E. 1998. An introduction to the resource description framework. Luettavissa:
<http://www.dlib.org/dlib/may98/miller/05miller.html>. Luettu: 26.11.2013.

Decker, S., Mitra, P. & Melnik, S. 2002. Framework for the Semantic Web: An RDF tutorial. Stanford University. California.

WP-Content. 2007. Semantic Web. Luettavissa: <http://www.usingmyhead.com/wp-content/uploads/2007/07/semantic-web.gif>. Luettu: 3.12.2013.

Hendler, J. 2006. Semantic Web @ 5 - Current Status and Future Promise of the Semantic Web. Luettavissa:
<http://www.cs.rpi.edu/~hendler/presentations/SemTech2006-keynote.pdf>. Luettu:
3.12.2013.

W3C 2008b. SPARQL Protocol for RDF. Luettavissa: <http://www.w3.org/TR/rdf-sparql-protocol/#intro>. Luettu: 15.3.2014

W3C 2004a. RDF Primer. Luettavissa: <http://www.w3.org/TR/2004/REC-rdf-primer-20040210/> Luettu: 13.11.2013.

W3C 2014a. RDF Primer 1.1. Luettavissa: <http://www.w3.org/TR/2014/NOTE-rdf11-primer-20140225/>. Luettu: 26.2.2014.

W3C 2014b. RDF 1.1 Concepts and Abstract Syntax. Luettavissa:
<http://www.w3.org/TR/rdf11-concepts/>. Luettu: 5.3.2014.

W3C. 2008a. SPARQL Query Language for RDF. Luettavissa:

<http://www.w3.org/TR/rdf-sparql-query/>. Luettu: 26.1.2014

W3C 2004b. OWL Web Ontology Language Guide. Luettavissa:

<http://www.w3.org/TR/owl-guide/>. Luettu: 26.2.2014.

W3C 2012b. Turtle - terse rdf triple language. Luettavissa:

<http://www.w3.org/TR/turtle/>. Luettu: 2.3.2014.

W3C 2014c. RDF Schema 1.1. Luettavissa: <http://www.w3.org/TR/rdf-schema/>.

Luettu: 9.12.2013.

W3C 2014d. RDF 1.1 XML Syntax. Luettavissa: <http://www.w3.org/TR/rdf-syntax-grammar/>.

Luettu: 14.12.2013.

W3C 2004c. RDF Semantics. Luettavissa:

<http://www.w3.org/TR/2004/REC-rdf-mt-20040210/>. Luettu: 22.11.2013.

W3C 2012c. RDB2RDF Implementation Report. Luettavissa:

<http://www.w3.org/2001/sw/rdb2rdf/implementation-report/>. Luettu: 25.11.2013.

W3C 2004d. Resource description framework (rdf): Concepts and abstract syntax.

Luettavissa: <http://www.w3.org/TR/rdf-concepts/>. Luettu: 12.12.2013.

Bergman, M. 2009. Advantages and Myths of RDF. Luettavissa:

<http://www.mkbergman.com/483/advantages-and-myths-of-rdf/>. Luettu: 3.4.2014.

Herman, I. 2007. State of the Semantic Web. Luettavissa:

<http://www.w3.org/2007/Talks/0424-Stavanger-IH/files/SPARQLUsage.png>.

Luettu: 24.3.2014.