

read_this_first.txt

Moni meistä on kuullut tokaisun “Mä en käsitä sua...” Ainakin näin naisena se kuulostaa varsin tutulta. Olemme erilaisia. Meillä on erilaiset taustat, kasvatus, koulutus ja käytämme aivopuoliskojamme eri tavoin. Käsitämme asiat jokainen omasta näkökulmastamme. Osa tämän lehden artikkeleista keskittyykin aiheeseen merkitysten erilaisuus sekä siihen, miten merkitykset muodostuvat aivoissamme. Tähän aiheeseen liittyen teimme myös pienimuotoisen testin.

Entäpä sitten, kun pitäisi kuvata olevaista maailmaa koneiden ymmärrettävästi. Miten puetaan konkreettiset asiat ykkösiksi ja nolliksi? Tätä ongelmaa on pohdittu ja tutkittu koko tietokoneiden olemassaolon ajan. On olemassa erilaisia käsittekaavioita, toimintamalleja jne. - uusimpana oliomallinnus kaikkine outoine termeineen. Sovellusta rakennettaessa käsitteet muodostuvat sen aihealueesta. Harrastukseen liittyvät käsitteet tuntevat sen aktiiviharrastajat. Kaikkihan tietävät, mikä on händäri, luoviminen ja ylämummo. Sovelluskehitystyössä tulevan sovelluksen käsitteistön asiantuntijoita ovat liiketoimintaosaajat. Yhteisten merkitysten luontiin tarvitaan yleensä dialogia, merkitykset eivät tahdo myöskään pelkän paperin avulla siirtyä tekijältä toiselle. Terminologian avaaminen ymmärrettäväksi on oma taiteenlajinsa kuten myös käsitteistön hallinta. Näistä molemmista teemoista löydät lehdestä myös maistiaisiasia.

Seuraava taso ymmärtämisessä on sovellusten välinen keskustelu. Tietoa siirretään järjestelmästä toiseen ja silti tiedon tulisi olla yhdenmukaista. Suurkoneiden aikaan tietojen kuvaaminen oli helpompaa kuin nyt. Oli yksi yhteinen tietohakemisto, jonne kaikki tiedot kuvattiin. Vaikeusaste kasvaa, kun konealustat ja sovelluskehittimet vaihtuvat. Tulevaisuuden haasteena on myös se, miten tieto kuvataan verkossa eri osapuolten välillä. Tarvitaan standardoituja menetelmiä tiedon kuvaamiseen. Kuvaustiedosta käytettäviä termejä ontologia ja metadata valotetaan tarkemmin lehden artikkeleissa.

ATK-alan termistö on pääsääntöisesti englannin kielistä. Suomessa toimii sanastotyöryhmä, joka kääntää sanoja suomeksi. Osa sanoista on ollut niin vaikeita kääntää, että ne eivät ole koskaan tulleet osaksi pysyvää puhekieltä kuten swap-file, joka suomeksi käännettynä olisi heittovaihtotiedosto. Näistä käännöskokemuksista olemme myös saaneet lehtemme artikkelin. Sanastoja tarvitaan myös aihekokonaisuuksittain, jotta käytössä on yhdenmukainen terminologia. Esimerkiksi testauskerhomme on muodostanut testaamiseen liittyvää sanastoa.

Tietotekniikassa on käytössä niin pitkiä termejä, että niistä käytetään lyhenteitä. Lyhenteiden ongelmana on se, että sama lyhenteen merkitys muuttuu eri asiayhteyksissä. Viime ajat keskusteluissa runsaasti vilahtanut termi ASP on hyvä esimerkki tästä. Lehteä suunnitellessamme käynnistimme lyhenteisiin liittyvän kokeilun. Ryhmätyöalueellemme Flehen on avattu alue, jonne pyrimme keräämään jäsenkunnan avulla lyhenne- ja termitietoa. Tervetuloa tutustumaan!

Mutta ennen sitä...
käsittämättömän nautinnollisia lukuhetkiä.

*Toimituskunnan puolesta,
Minna Oksanen
Sytyke ry:n johtokunnan jäsen*

Käsittämättömät käsitteet

Irmeli Sinkkonen, Adage Oy

Käsitteen käsite

Käsite = ajattelun luoma abstraktinen hahmo, ajattelulla tarkistettu mielle (Nykysuomen sanakirja, lyhentämätön kansanlaitos, WSOY 1966).

Käsitteen käsitettä voi lähestyä monesta näkökulmasta, monesta tieteestä. Itse lähestyn sitä kognitiivisen psykologian kautta ja ajattelen käsitteen olevan ajattelun yksikkö. Käsite vastaa jotain todellisuutta ajatuksissamme ja puheissamme, sillä on yleensä nimi ja joukko ominaisuuksia, jolloin sitä voi ajatella¹. Käsitteessä on myös jokin piirre tai joukko piirteitä erottamassa sitä kaikista muista käsitteistä. Käsitteen muodostus on siis eräänlaista luokitte-
lua. Yksi puumainen kasvi on 'pensas', joku toinen on 'puu'. 'Pensas' ja 'puu' ovat *luonnollisia käsitteitä*. Pensa- ja puun täsmällinen ero löytyy puutarha-
alan lähteestä, mutta varmaan kaikilla on jonkunlainen käsitys mikä 'pensas' on ja mikä on 'puu'. Tämä käsitys joutu-
tuu koetteille silloin, kun edessämme on 'pensastava puu' tai 'puumainen pen-
sas' eli käsitteen epätyypillinen ilmen-
tymä. Käsitteet erotteleva tekijä onkin usein sellainen, joka ei alan maallikolle näy, vaan erottelemme käsitteet jolloin asialle tyypillisellä, mutta ei sen virallises-
sa luokittelussa määräävällä ominai-
suudella.

Miten käsitteet omaksutaan ja mistä ne syntyvät

Käsitteet opitaan joko ulkomaail-
masta tai johtamalla toisista käsitteistä. Vanhemmat opettavat lapsilleen asioiden nimiä sitä mukaa kun lapsi kohtaa niitä. Tieteissä ja ammattiteissa on omat käsi-
teistönsä. Kun tutustuu uuteen ammat-
tiin, on opittava ensin sen peruskäsi-
teistö. Jos tutkii uutta tieteenalaa, jou-
tuu johtamaan uusia käsitteistöjä. Kä-
sitteitä voidaan muodostaa, kun on ole-
massa selkeä asia, jota halutaan ajatella
omana yksikkönään ja jossa on jokin
sellainen ominaisuuksien joukko, joka
erottaa sen muista käsitteistä. Käsite voi
siis olla täysin keinotekoinen.

Käsitteet ovat usein kansainväli-
siä, mutta aina eri kielet eivät ole käsi-
teyhteensopivia. Eri tieteetkään eivät
välttämättä ole käsiteyhteensopivia, ja
varsinkin humanististen tieteiden sisäl-
lä löytyy tyypillisesti tutkijoita, joiden
käsitteet eivät ole yhteensopivia. Var-
sinkin ihmistieteessä on hyvä selvittää,
mitä puhuja/kirjoittaja käsitteellä tarkoi-
taa. Yksi esimerkki on psykologian kä-
sitteet mentaalimalli – sisäiset malli –
kognitiivinen malli². Toinen hankala
käsitevyöhyke on metafora – analogia
käsitteet. Käytettävyydessä menee se-
kaisin, kun alkaa miettiä onko käyttö-
kokemus osa käytettävyyttä vain päin-
vastoin. Kumpikin ratkaisu on perus-
teltavissa.

Jos tieteiden välillä ja jopa tietei-
den sisällä käsitteet muuttavat merki-

tystä tutkimuksesta toiseen, on melko
todennäköistä, että näin käy kahden
ammattikunnan ja etenkin kahden ihmi-
sen välillä. Viesti: teksti, sana, koodi tai
symboli, saa merkityksensä vastaanot-
tajan toimesta, eikä se merkitys ole aina
sama kuin viestin lähettäjän merkitys³.

Skeemat

Skeemoiksi kutsutaan joskus ih-
misen ajatusta jostain käsitteestä. Skee-
ma käsittää paitsi itse käsitteen nimen,
suuren joukon ominaisuuksia, joita kä-
sitteen käyttäjä liittää siihen. Ominais-
sudet joko periytyvät jostain yläkäsi-
teestä, tai ne ovat käsitteen omia omi-
naisuuksia, jotka erottavat ne sen "si-
särkäkäsitteistä" ja jotka kaikki taas pe-
riytyvät kyseessä olevan käsitteen ala-
käsitteille. Esimerkiksi kana on eläin
(hengittää, syö, liikkuu), lintu (lentää,
sillä on höyhenpeite). Toisin kuin olio-
mallinnuksessa, käsitteellä ("lintu") voi
olla ominaisuuksia ("lentää"), jotka ei-
vät periydy sellaisenaan jokaiselle ala-
käsitteelle (esimerkiksi "strutsille").
Skeemojen oletusarvoja käyttäen ajat-
telumme on nopeaa: jos joku puhuu "ta-
lostä", voimme olettaa sen olevan suun-
nilleen tietynlainen. Siinä on "katto",
yleensä vähintään neljä "seinää", "ovia"
ja "ikkunoita". Kun puhutaan "omako-
titalosta", mielikuvamme tarkentuu: se
on luultavasti 1-2 kerroksinen, tietyn
kokoinen jne. Jos taas mainitaan "rin-
tamamiestalo", vanhimmat viestin vas-
taanottajat ymmärtävät, että puhutaan
tietystä 1,5 kerroksisesta "omakotita-
lo"-tyypistä, osa nuorimmista ei ym-
märrä käsitettä lainkaan.

¹ Filosofit keskustelkoot asiasta ovatko käsitteet olemassa ilman niitä muodostavaa subjektia. Eli vaatiiko käsitteen olemassaolo sekä sen pohjalta luotavan objektin että käsitteen muodostajan eli subjektin. Insinööritieteilijöinä voimme lähteä siitä, että käsittelemämme käsitteitä vastaa (tai ainakin pitäisi vastata) jokin reaali maailman asia tai ilmiö ja on joku, joka sitä ajattelee, joko me tai käyttäjä.

² Oma lukunsa on sitten se tapa, jolla maallikot näitä sanoja käyttävät. Esimerkiksi sana 'mentaalimalli' on alkanut käytettävyysskielessä tarkoittaa melkein mitä tahansa ajatusta jostain asiasta. Myös sana englannin kielen sana 'affordance' (suom. affordanssi, mahdollisuus, suunto) sai 90-luvulla käytettävyysskielessä saman merkityksen kuin 'konventio' tai 'alkupalaute' tai jopa 'näkyvyys'. Käsitteet siis rapautuvat helposti.

³ Semiotiikassa puhutaan vielä sanan *denotatiivisista* (sanakirjan, kouluopetuksen mukaisia) että *konnotatiivisista* merkityksistä. Psykologian kielelle käännettynä tämä tarkoittaa sitä, että me assosioimme kukin käsitteeseen omia mielikuviamme ja jopa tunteita. Nämä tunteet ja mielikuvat ovat syntyneet meille oman elinhistoriamme aikana. Konnotatiiviset merkitykset jollekin sanalle voivat olla ihmisillä yhteiset tai ei.

Skeemojen käyttö lukemisessa ja oppimisessa

Käytämme skeemojamme, kun luemme, kuuntelemme toista ihmistä, yritämme käyttää järjestelmää. Meissä viriää lukiessamme/kuunnellessamme asianmukaiset skeemat ja pystymme ymmärtämään tekstin. Mitä lähempänä meidän ja tekstin kirjoittajan skeemat ovat toisiaan, sitä todennäköisemmin ymmärrämme toisillemme ”lähetetyt” viestit. Entäs jos meillä ei olekaan skeemaa asialle, josta kirjoittaja kirjoittaa? Silloin käy luultavasti niin, että kirjoittajan hienot käsitteet, esimerkiksi ”konfiguraation prototypisointi” tai ”serendipiditeetin kasvukäyrä” tai ”propositio-naalinen representaatio”, ei viritä mitään skeemoja, emmekä ymmärrä tekstiä. Emme opi asiaa, jotka näihin käsitteisiin liittyy. Jos tietojärjestelmän käyttäjä ei ymmärrä käyttöliittymän käsitteitä, hän ei pysty lukemaan käyttöliittymää, eikä opi käyttämään sitä.

Istuin vuonna -94 opettelemassa käyttämään 3270-emulaattorin päälle tehtyä aikaraportointiohjelmistoa. En ymmärtänyt kerrassaan mitään. Kaikilla muilla kurssilaisilla oli 3270 käsitteistö selkäytimessä, minulla oli vain yleissivistys asiasta, koska olin tehnyt aina järjestelmiä henkilökohtaisille tietokoneille. Koulutus oltaisiin yhtä hy-

vin pitää portugaliksi, olisin ymmärtänyt yhtä paljon: Tuttuja sanoja siellä täällä. Järisyttävä, mutta opettava kokemus.

Käsitteet systeemityössä

TKKn käytettävyysslaboratorion toiminta-ajalta (7 vuotta) kerätty käytettävyydestiaineisto osoittaa, että huonosti tehty määrittelyvaihe aiheuttaa hankalimmat käytettävyysvirheet. Projektit tuottavat (räätälöityjä) järjestelmiä, joissa käsitteet ovat melkein mutta eivät ihan kunnossa. Järjestelmä näyttävät järkevältä, mutta toimivat sittenkin jotenkin kummallisesti.

Jos käsitteiden merkitykset ovat yksilöllisiä, saako järjestelmistä millään kaikille sopivia? Onneksi tilanne ei ole toivoton. Räätälöidyn järjestelmän tekeminen käsitteistöltään oikeaksi on teoriassa helppoa, koska yrityksen sisäisen kielenkäyttö ja käsitteistö on varsin homogeenista. Jokaiselle ihmisille tarkoitetun järjestelmän käsitteistö on ainakin periaatteessa hankalampi rakentaa, mutta näiden järjestelmien aihepiiri on yleensä yleisempi ja yleensä riittää, että järjestelmä käyttää yleiskieltä. Kaikki ihmiset huomioon ottavan järjestelmän suunnittelu on hankalaa ihan muista syistä, erityisesti esteettömyysongelmien⁴ vuoksi.

Suunnittelijoiden työ ja haaste on muuttaa ihmisen epämääräinen ajattelu formaaliksi tietojärjestelmäksi. Muuntaminen tapahtuu käyttäen eräänlaisia analyysimenetelmiä ja malleja. Täsmällinen kuvaus käsiteltävistä asioista tarvitaan myös siksi, että vastaanottajat, tässä tapauksessa systeemin suunnittelijat, pystyvät tarkastamaan, että heidän näkemyksensä käsitealueesta on yhtenäinen, johdonmukainen ja yhteinen kaikilla. Erilaiset mallit toimivat pohjana sekä tietokantojen ja toimintojen toteutuksessa että yhteisen näkemyksen syntymisessä projektiryhmässä. Erityisesti oliomallinnus on tuonut suunnittelumenetelmät lähemmäksi ihmisen ajattelun todellisuutta, matkiihan se nk. konnektionistien mallia ihmisen skeema (käsite) verkoista. Nämä formaalit mallit eivät kuitenkaan ole oikea tapa siirtää tietoa käyttäjän/asiakkaan ja ATK-suunnittelijan välillä. Miten siis kannattaisi toimia?

Tuotekehitysprosessi ja käyttäjän käsitemaailma

Käyttöliittymän käsitteellinen määrittely

Kun ihmisen tehtävät viedään järjestelmään, sen käsitteet eivät saisi muuttua. Vaikka käyttäjien on mahdollista opetella uusi käsitteistö, on hyvin



Kuva 1: Jos huonosti käy, suunnittelija ei ymmärrä käyttäjää, suunnittelija ei osaa siirtää mallia järjestelmään, käyttäjä ei osaa lukea suunnittelijan tekemää käyttöliittymää.

⁴ Esteettömyys tarkoittaa, että myös iäkkäät ihmiset ja eri tavoin vammautuneet pystyvät käyttämään järjestelmää

epätaloudellista, että 5-5 000 henkilöä opettelee vanhalle työlle uudet käsitteistöt vain siksi, että suunnittelija ei ole ehtinyt omaksua tehtävään liittyvää oikeata käsitteistöä. Väärät termit paitsi hidastavat oppimista, myös aiheuttavat virheitä koko järjestelmän eliniän ajan. Samoin on tärkeää, että käsitteellinen malli vastaa sitä mentaalimallia, mikä käyttäjällä on niistä töistä, jotka asianomainen järjestelmä kattaa, eli tuotteen käsitteellisen mallin täytyy lähteä käyttäjän tehtävistä, ei esimerkiksi tietokannan käsitteellisestä mallista. Tämä malli on sitten viestittävä käyttäjäliittymän kohteiden, toimintojen, termien ja symbolien kautta käyttäjälle. Käytettävyysermeihin järjestelmän ja reaali maailman vastaavuuden (*mapping*) pitää siis olla hyvä.

Suunnittelijaryhmän täytyy siis järjestelmän määrittelyvaiheessa pystyä sekä sisäistämään että dokumentoimaan järjestelmäalueen käsitteet (termit) ja näiden käsitteiden suhteet. Käsitteiden selvitystä tehdään määrittelyvaiheen aikana kolmivaiheisesti:

- Kun käyttäjien nykyinen toimintatapa ja tehtävät tutkitaan käyttäjä tutkimuksessa, kerätään myös käyttäjän käsitteistö määriteltävältä alueelta (taulukko 1)
- Kerätty käsittemateriaali analysoidaan, jne. kerätyn materiaalin vietiin muotoon, jolla ryhmä pystyy tarkistamaan tulosten johdonmukaisuuden (taulukko 2).
- Analyysin tuloksen tarkistus ja täydennys (taulukko 3).

Kolmas vaihe, tuloksen tarkistus, oman ymmärryksen tsekkaaminen on hyvin tärkeä vaihe, eikä sitä saa missään nimessä jättää tekemättä. Myös se väline, jonka avulla sanoma, tässä tapauksessa käsitteet, siirtyy käyttäjiltä suunnittelijoille ja suunnittelijoilta tarkistuksessa käyttäjille. Itse pidän parhaana tarinoin perustuvaa viestintää, sillä tarinoiden kertominen on ihmiselle hyvin luonnollinen tapa kertoa kokemuksistaan ja kuvata toiselle mielikuviin. Tarinat tai skenaariot ovat parempia kuin käytötapauskuvaukset. Käytötapauskuvauksissa tarinat on jo muunnettu kohti formaalimpaa esitystä eivätkä ne enää ole yhtä konkreettisia kertoa ja ymmärtää kuin tarinat. Käytötapauskuvauksia toki on hyvä tehdä, mutta suunnittelun sisäiseen käyttöön.

Taulukko 1: Käsitetiedon keruu	
Haastattelut (interview)	Yksilö- ja ryhmähaastattelut. Termit poimitaan tallenteilta
Havainnointi (observation)	Käyttäjän työpaikalla tapahtuva työn seuranta. Termit poimitaan tallenteilta
Fokusryhmät (focus group)	Projektin käytössä oleva käyttäjäryhmä. Termit poimitaan tallenteilta
Käytettävyytestit (usability tests)	Selvitetään, miten käyttäjät käyttävät nykyisiä järjestelmiään, mitä ongelmia ja vahvuuksia niissä on. Termit poimitaan tallenteilta
Käyttäjien esitykset työstään (usability roundtable)	Käyttäjät esittelevät suunnittelijoille työnsä. Termit poimitaan tallenteilta
Päiväkirjat (diaries)	Käyttäjät kirjaavat valmiisiin päiväkirjapohjiin tutkimuksen kannalta tärkeitä asioita. Termit poimitaan päiväkirjoista
Toimintatarinoiden keruu* (collecting scenarios)	kerätään ja kuvataan konkreettisia ja yksityiskohtaisia kuvauksia käyttäjien arjesta. Termit tarinoista.
Kontekstuaalinen tutkimus (contextual inquiry)	KT yhdistelee muita tässä taulukossa mainittuja menetelmiä. Termit poimitaan tallenteilta ja muista materiaaleista.

Taulukko 2: Käsitetiedon jäsentäminen	
Seinätaulutekniikat, erilaiset mallit, mind map-tekniikat,	Luokitellaan käsitteet, jäsennetään ja järjestetään, kuvataan suhteet
Käyttäjien kanssa tehtävät korttimenetelmät valikoiden ja linkkien selvittämiseksi (card sorting)	Korttien järjestelytehtävä: ryhmiteltävät toiminnot tms. kukin omalle kortilleen, pyydetään käyttäjiä ryhmittelemään kortit ja nimeämään ryhmät. Mahdollisesti syytä sallia kortin kuuluminen useampaan ryhmään samanaikaisesti. Ennustamis- ja etsimistehtävä: näytetään suunniteltu päätaso, pyydetään ennustamaan, minkä päätason kohteen takaa löytyy mikäkin alemman tason kohde

Taulukko 3: Käsitetiedon tarkistaminen, täydentäminen	
Roolileikit, simulaatiot (Dramas, simulations)	Kerätään, simuloidaan ja varioidaan toimintatilanteita. Näissä käytetään termejä, jotka näin ”tarkistetaan”
Termitestit (term tests)	Käyttäjä vertaa ja yhdistelee listan terminimet toisen listan sisältökuvausten kanssa. Myös yhdistelyjärjestys dokumentoidaan.
Fokusryhmät	Oma istuntonsa, jossa termit ja niiden suhteet tarkistetaan
Haastattelut	Oma tapaaminen termien tarkistamiseksi
Toimintatarinat*	Jos suunnittelijat ovat tehneet tarinat, ne on ensin tarkistutettava. termit syytä tarkistaa samalla
Käytettävyytestit	Varsinkin protojen käytettävyystesteissä tulee esiin vielä terminologia virheitä

*) toimintatarinat voi joko kerätä käyttäjiltä tai suunnittelijat voivat rakentaa ne haastattelujen pohjalta. Jälkimmäisessä tapauksessa ne on ehdottomasti tarkistettava käyttäjien kanssa.

Lopuksi muutamia käytännössä hyväksi havaittuja neuvoja. Kun opiskelee käyttäjän käsitemaailmaa ja työtä:

- Käy tapaamassa haastateltavaa tämän työpaikalla. Ihminen muistaa työhönsä liittyvät yksityiskohdat parhaiten omassa työympäristössään.
- Valmistele vierailut huolellisesti, selvitä, mitä pystyt asiaa etukäteen, mutta pysy avoimena sille mitä käyttäjät kertovat
- Mene seuraavaan paikkaan vasta, kun olet hahmottanut ja järjestänyt mielessäsi edellisen paikan annin.
- Älä esitä tietävämpää kuin olet juttellessasi käyttäjien kanssa. Itse olen huomannut, että tyhmemmäksi heittäytyminen saa haastateltavan skarppaamaan sen kanssa, mitä ja miten hän asian esittää.
- Pyydä jatkuvasti varmistuksia, että olet ymmärtänyt asiat oikein, tyyliin ”Tarkoitat siis, että...
- Esitä tulkintasi käsitteiden ja asioiden suhteesta mahdollisimman konkreettisella tavalla. Älä vie käyttäjälle ER- tai muita kaavioita. Pidä ne omina suttuinasi (tai firmasi vaatimina dokumentteina, miten vain). Olen itse käyttänyt mind map-esitystä. Käyttäjä näkee niistä oitis onko terminologia ja asioiden suhteet järkevät

- Voit myös pyytää käyttäjiä tekemään tarinoita tapahtumista, jotka edellyttävät tuotteen käyttöä. Jos valmiiden tarinoiden saaminen ei onnistu, voit pyytää niitä, mitä käyttäjälle tulee haastattelussa mieleen. Mikäli et ole saanut käyttäjältä valmiita tarinoita on hyödyllistä tehdä tarinat itse. Näin voi selvittää, onko ymmärtänyt asiat oikein eli pyytää käyttäjää tarkistamaan niiden järkevyys.
- Ei kannata pitäytyä vain yhdessä taulukoissa mainituista menetelmistä. Eikä kannata ainakaan rajata oman yrityksen käyttämiksi menetelmiksi vain osaa taulukoissa mainituista. Hyvä suunnittelija yhdistelee näitä menetelmiä tilanteen mukaan.
- Käyttäjän käsitemaailmaan perehtymiseen täytyy varata aikaa, myös jopa vaikka olisi nk. toimialaosaaja. Tämä pitäisi järjestelmien ostajapuolenkin ymmärtää, ettei tulisi ostettua halpaa laadukkaan tuotteen sijaan.

Kirjallisuus:

Käsitteistä psykologiassa:

Anderson, J. R. (1995, 4th ed). *Cognitive Psychology and its Implications*. W.H. Freeman and Company, NY

Anderson, J. R. (2000, 2nd ed.) *Learning and Memory. An Integrated Approach*. John Wiley & Sons, NY

Sinkkonen, I., Kuoppala, H., Parkkinen, J., Vastamäki, R. (2002) *Käytettävyyden psykologia*, IT-Press, Helsinki

Smyth, M. M., Collins, A. F., Morris, P. E. Levy, P (1994) *Cognition in action*, Psychology Press, Hove

Thagard, P (1996) *Mind, Introduction to Cognitive Science*. The MIT Press, Cambridge, MA

Wilkes, A.L. (1997) *Knowledge in minds, Individual and collective processes in cognition*, Psychology Press, Hove

Vaativuusmäärittely ja käsitteet: Beyer, H., Holtzblatt, K. (1996) *Contextual Design: A Customer-Centered Approach to Systems Designs*, John Wiley & Sons, NY

Hackos, J., Redish, J. (1998) *User and Task Analysis for Interface Design*, John Wiley & Sons, NY

Mayhew, D. (1992) *Software user interface design.*, NJ: Prentice Hall. Englewood Cliffs

Rosson, M. B., Carroll, J. M. (2002) *Usability Engineering. Scenario-Based Development of Human-Computer Interaction*. Morgan Kaufman Publishers, San Fransisco, CA

Semiotiikkaa: Veivo, H., Huttunen, T. (1999) *Semiotiikka. Merkeistä mieleen ja kulttuuriin*. Edita, Helsinki

Irmeli Sinkkonen, *Adage Oy*

Käsitteiden käsittämistä

Koonnut Minna Oksanen

Lehden teemaa miettiessämme pohdimme ajatusta, miten ihmiset käsittävät asiat. Siksi teimme pienimuotoisen testin. Lähetimme noin 10:lle sytykeläiselle tuttuja termejä kuvattavaksi omin sanoin. Vastauksia saimme kolme, jotka ohessa anonyymeinä. Vertailuryhmään suostui tietotekniikkaa tunteva juristi.

Sinänsä itse tulos ei aiheuttanut yllätystä. Meillä, jotka olemme informaatiotekniikka alalla, on melko sel-

keä ja yhdenmukainen näkemys siitä, mitä eri käsitteet tarkoittavat. Vivahde-eroja toki löytyy. Juristin näkemykset poikkesivat eniten.

Sanat, jotka halusimme kuvattavan liittyvät myös lehden teemaan. Jotta ei tule painotuseroja, sanat ovat aakkosjärjestyksessä. Sanat ovat: arkkitehtuuri, mallinnuskieli, metodi, prosessi ja tieto.

Suurimmat erot löytyivät kuvattaessa sanaa tieto. Tieto on niin yleinen käsite, että sen jokainen mieltää

oman tarpeensa mukaan. Vertailuryhmän edustajan kommentti 'Tieto on valtaa' kuvaa hyvin tiedon merkitystä. Toisaalta asia voidaan johtaa pidemmällekin, kuten Petteri Järvisen Internet Muutoksentehtävä kirjassa mainitaan 'Valtaa ei ole tieto, vaan kokonaisuuksien ymmärtäminen'. (<http://www.pjoy.fi/kirjat/imuutos/luku03c.htm>).

Ohessa kokeilumme lopputulos:

SANA	Sytyke1	Sytyke2	Sytyke3	Vertailu
Arkkitehtuuri	Arkkitehtuuri on rakenteellinen kokonaiskuva tarkasteltavasta kohteesta. Se kuvaa pysyvät tai sellaisiksi tarkoitetut rakenteet. Arkkitehtuureja löytyy systeemityön piiristä lukuisia, esim. järjestelmä-, sovellus-, tekninen, looginen, komponentti-, olio-, tieto-, tietoliikenne-, hajautus- jne arkkitehtuuri.	Loogisten komponenttien kokonaisuus, esim. käyttöjärjestelmä tai verkko.	Arkkitehtuurilla kuvataan jokin kokonaisuus esim. tietoarkkitehtuuri, järjestelmäarkkitehtuuri, tietoliikennearkkitehtuuri, laitteistoarkkitehtuuri.	(De-facto) määrittely, jonka pohjalta "rakentaminen" suunnitellaan edelleen yhdenmukaisen ympäristön toteuttamiseksi.
Mallinnuskieli	Ei-luonnollinen kieli, jonka avulla laaditaan abstrakti malli (yleensä) reaali maailman ilmiöstä tai asiasta. Mallinnuskieli on notaatio, sillä on syntaksi ja semantiikka. Kieli ei kuitenkaan mallinnuksessa ole yksin "riittävä", tarvitaan myös prosessi kielen käytön ohjeeksi. Esim. UML (Unified Modeling Language)	Kuvaustekniikka	Mallinnuskieltä käytetään mallin kuvaamiseen. Tavoitteena on mallinnuskielen avulla välittää mahdollisimman yksiselitteisesti lukijalle se käsitys todellisuudesta, jonka kirjoittaja mallillaan haluaa välittää. Oliolähestymistavassa UML on nykyisin tavallisin mallinnuskieli.	Kieli, jota käytetään järjestelmän määrittämiseen tahi kuvailemiseen.

Metodi	1. Olio-ohjelmointikielellä toteutettu palvelu. Esim. luokan metodi (Henkilöluokan LaskeIkä-metodi). 2. Menetelmä, työtap, tekniikka jonkin yksittäisen asian tekemiseksi tai kokoelma hyväksi havaittuja tekniikoita ja käytäntöjä laajemman tehtävän läpiviemiseksi.	Menetelmä, tapa tehdä jotain, vastaa kysymykseen miten	Oliolähestymistavassa metodilla tarkoitetaan olion toimintaa, joka voidaan kuvata algoritmina.	Tapa tehdä asioita. Käyttökelpoinen sarjamurhaajien selvittämiseksi, koska näillä on tapana standardoida toimintansa. Voi myös tarkoittaa objektin käsittelyyn käytettäviä toimintoja
Prosessi	Prosessi on sarja loogisesti yhteen liittyviä toimintoja. Prosessi muokkaa resurssiensa avulla saamansa syötteen ennalta määritellyksi tulokseksi. Prosessilla on tyypillisesti heräte. Liiketoimintaprosessin tarkoituksena on tuottaa lisäarvoa. "Ohjelmistotuotantoprosessissa käytetään/sovelletaan metodologia A, joka tukeutuu mallinnuskieleen B suunniteltaessa sovelluksen C komponenttiarkkitehtuuria."	Prosessi on sarja toimintoja. Prosessi on jatkuvaa toimintaa, vrt. Projektilla on ajallinen alku ja loppu. Prosesseja ovat mm. tuotekehitysprosessi, toimitusprosessi, myynti- ja markkinointiprosessi, tuotantoprosessi ja johtamisen prosessit.	Toiminto. Prosessi kuvaa, miten organisaatiossa asioita hoidetaan. Prosesseja voidaan tarkastella eri tarkkuustasoilla ja tarkentaa aina työnkulkujen yksityiskohtiin asti. Prosessien kuvaamiseen on kehitetty monia erilaisia mallinnustapoja perinteisistä prosessikuvauksista, tietovirtakaavioista ja lävistäjämatriiseista oliomallinnuksen mukaisiin käyttötapaus- ja yhteistyökaavioihin.	Jotain tapahtuu, jonka suunta tosin ei välttämättä ole selvä tai ennalta arvattavissa kenellekään osapuolista ennakkomääritelmistä huolimatta. Vaikeintentti oikeustieteellisessä..
Tieto	Tieto-sanaa/käsitettä käytetään erilaisissa merkityksissä. Normaalisti yhteydestä selviää, mistä tiedon "tasosta" tai merkityksestä on kysymys. Esim. - merkki, merkkijono, detaljitieto, "data", sisältö merkityksetön (tietokannan taulun puh.nro-sarakkeen numerot) - informatiivista, sisällöllisesti arvokasta, "informaation" (123456 = Minnan puhelinnumero, haluan soittaa hänelle) - ymmärrystä, osaamista, "knowledge" (tieto siitä, miten arkkitehtuureja ja prosesseja kuvataan)	Data. Tietoa välitetään henkilöltä toiselle, järjestelmästä toiseen. Tietoa omaksutaan ja opitaan.	Tieto on vaikea termi suomen kielessä (data, informaatio, knowledge). Järjestelmäkehityksessä sillä tarkoitetaan tietojärjestelmän kohdetta (yksilötyyppejä, entiteettiä) tai oliomaailman termein yksittäistä oliota tai olioluokkaa. Tietojärjestelmät käsittelevät tietoja, joiden tallennukseen käytetään tietokantaa	Vallan toinen esiintymismuoto

Dialogi - yhdessä ajattelua

Olli Leppänen,
Lähde Valmennus

Vuorovaikutuksen kuudes porras

Kun kaksi ihmistä kohtaa, ei heidän välillään vaihdu ainoastaan sanat vaan katseet, tunteet, eleet ja ajatukset. Vuorovaikutukselta on vaikea välttyä tänä päivänä. Päivittäin omassa työyhteisössäni kohtaan tilanteita, joissa pelkästään toisen sanoista ei oivalla mitä hänellä oikein on mielessään. Sellaisia tilanteita tulee varmaan mieleen sinullakin – töistä, kotoa ja harrastuksista. Millainen fakiiri tulisi olla, että voisi yksinkertaisesti ymmärtää mitä toinen ajattelee?

Tekninen osaaminen sekä laitteiden ja ohjelmien hallitseminen helpottaa päivittäistä työtämme, mutta ne ovat vain elottomia apuvälineitä. Organisaatioiden kehittämisessä ei ole oikotietä. Yksilöt muodostavat organisaation, ja aina jokaisen työn takaa löytyy ihminen. Työyhteisöissä etenkin yhteisten jaettujen mielikuvien luomisessa ja käsittämättömien käsitteiden ymmärtämisessä dialogi on olennainen taito.

Mitä dialogi oikein on? Dialogia voisi kuvata mielenkiintoisella tavalla, vuorovaikutuksen kuuden portaalla. Vuorovaikutuksen kuudesta portaasta ensimmäinen on monologi. Eli voit puhua itseksesi, joko itsellesi tai muille, mutta se ei hirveästi ruoki vuorovaikutuksen ilmapiiriä. Sinulla voi olla paljon annettavaa, mutta samalla omalla monologillasi helposti tukahdutat muiden ajatukset. Seuraava taso portailla onkin sitten jutustelu, small talk. Esimerkkinä käy parhaiten normaali kahvipöytäkeskustelu työpaikalla. Jutustelukumppanit tunnetaan, uskalletaan aukaista suumme, mutta syvällistä sisältöä ei puheesta löydy.

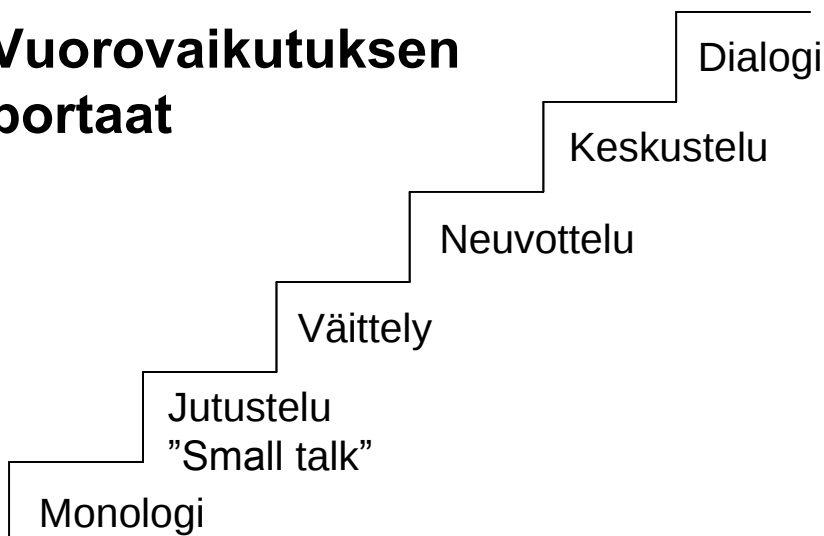
Kun vuorovaikutustilanne kehittyy, on seuraava vaihe väittely. Eli keskustellaan jo asiasta, mutta ainoa tapa edetä on panna vastaan toisen ajatuksiin. Vaikka koulussakin kilpailaan väittelyssä ja väittelyn voittajia kunnioitetaan, on se kuitenkin useimmiten hedelmätöntä eikä vie asiaa eteenpäin. Pikemminkin väittely vie keskustelussa olennaisesta epäolennaisuuksiin ja merkityksettömiin yksityiskohtiin. Seuraava ja hedelmällisempi tapa keskustella onkin sitten neuvottelu. Neuvottelussa tavoitteena on yhteinen lopputulos, jossa jo jokaiselta osallistujalta odotetaan omaa panosta. Itse asia on keskeinen, eivät ihmiset eivätkä heidän ajatuksensa. Kärjistäen voisi sanoa, että neuvottelussa heitetään pöytään 2-4 vaihtoehtoa, joista valitaan se joka vähiten hyödyttää ja vähiten vahingoittaa neuvottelun osapuolia. Eli tavoitteena ei ole ollenkaan paras ratkaisu vaan kompromissi.

Vuorovaikutuksen portailla viides porras on jo avoin keskustelu, jossa rohkeasti heitetään peliin oma persoona ja omat ajatukset. Keskustelun ilmapiiri tukee osallistujien ideoita ja kannustaa eteenpäin. Työyhteisöjen siirtyessä oman ympäristönsä ulkopuo-

lelle, syntyy automaattisesti avoin ja innostunut keskustelu. Ideat lentävät ja ryhmähenki kukoistaa. Mutta kun palaamme seuraavana päivänä takaisin omalle työpöydällemme, on enää rippeet innostuksesta jäljellä. Avoin keskustelu innostaa, mutta ei usein johda pidemmälle. Miksi?

Kuudes vuorovaikutuksen taso on aito dialogi. Dialogi on yhdessä ajattelua ja yhdessä oppimista. Lähtökohtana aidossa vuorovaikutteisessa tilanteessa on, että kaikki osallistujat ovat läsnä. Selkeää johtajaa ei tarvita, vaan jokainen vuorollaan vie keskustelua eteenpäin. Tavoitteena on että keskustelussa syntyy päätöksiä ja se menee koko ajan eteenpäin. Tämän tavoitteen saavuttamiseksi meidän tulisi aina kuvitella, että olemme näyttelijöitä teatterin lavalla jokaisessa keskustelussamme. Jos emme pistä itseämme likoon ja aidosti osallistuu keskusteluun, muuttuu se hyvin nopeasti vain tyhjän päiväiseksi jutusteluksi tai kasvottomaksi neuvotteluksi. Kun emme jaksa keskittyä, tunteet nousevat pintaan ja keskustelu muuttuu väittelyksi, jossa yhdellä sanalla voidaan pahoittaa toisen mieli.

Vuorovaikutuksen portaavat



Valitettavasti liian harvoin pääsemme kokemaan keskustelun, jota voidaan hyvällä mielellä kutsua aidoksi dialogiksi. Kotona, koulussa ja elämän varrella olemme oppineet varjelemaan omia ajatuksiamme ja heiluttamaan sanan säilää. Näiden taitojen poisoppiminen ja keskusteluun antautuminen vaativat todella paljon yksilöltä. Viime talvena työyhteisöille toteuttamisemme Hehkuilloissa lähdimme liikkeelle hyvin pienestä. Hehkuilloissa työporukka kokoontui päivän päätteeksi metsän siimekseen, nuotion ääreen, keskustelemaan työasioista. Yhdessä valitusta temasta keskusteltiin vuoron vaihtuessa ”puhekapulan” avulla. Valmentajina meidän tehtävämme oli auttaa keskustelua eteenpäin kysymysten ja herättelevien ajatusten kautta. Kokonaisuudessaan Hehkuilloissa päästiin aika ajoin lähelle sitä tunnelmaa ja periaatteita, jotka ovat William Isaacsin mukaan aidon dialogin kulmakiviä:

1. Kuuntelu. Ymmärtääksemme mitä toinen sanoo ja tarkoittaa, on meidän kuunneltava häntä. Kuuntelemalla olemme koko ajan läsnä keskustelussa, emmekä tipahda kärryiltä.
2. Kunnioitus. Kuuntelemalla mitä toisella on sanottavaa, osoitamme samalla kunnioitusta häntä kohtaan. Monilla on tapana keskittyä katselemalla muualle, toisille se on kuitenkin osoitus siitä, ettei hänen puheensa kiinnostanut kuulijaa. Toisen kunnioittaminen tarkoittaa myös hänen ajatustensa kunnioittamista. Dialogissa on tavoitteena löytää paras ratkaisu yhteisen keskustelun ja ajatusten esittämisen kautta. Jokaisen ajatukset ja sanat ovat siten tärkeä osa lopputulosta, eikä niitä saa ensimmäiseksi teillata.

3. Odottaminen. Keskustelussa ei tarvitse olla äänessä koko ajan. Hiljaisuudessa on voimaa ja usein on parempi odottaa omaa vuoroaan ja antaa omien ajatusten jalostua, ennen kuin ne suustaan ulos lausuu. Odottaminen on myös toisen sanojen arvostamista, emme keskeytä häntä omalla puheellamme tai häiritse tekemisellämme.
4. Suora puhe. On dialogin tärkein elementti. Dialogi on keskustelua, jossa uskalletaan rohkeasti tuoda kaikki näkökulmat esiin. Jokainen on vastuussa, että keskustelu menee eteenpäin. Suora puhe ei lavertele, selittele tai takerru lillukan varsiin. Suora puhe tarkoittaa myös avoimuutta ja rehellisyyttä.

Miten näitä elementtejä voisi hyödyntää omalla työpaikallaan? Omassa tiimissä?

Ottakaa ilo irti kahvitauoisista! Valitkaa keskustelua nostattava puheenaihe tai ohjaa itse jääräpäisesti keskustelu mieltäsi hiertävään pulmaan. Antakaa kaikille osallistujille tilaa ja käyttäkää vaikka jotain kaunistakiveä puheen vuoron merkinä. Kysykää toisten mielipiteitä ja ajatuksia – koko ajan mielessänne tavoite: Ymmärtää miksi toinen ajattelee tuolla tavalla?

Heittäkää pöydät pois koukushuoneesta! Pitkä neuvottelupöytä kangistaa heti vuorovaikutustilanteen neuvotteluksi. Jokaisella on omat vanhat paikkansa ja joku on aina sihteeri. Pöydän ja papereiden taakse on myös helppo piiloutua. Dialogissa me kaikki olemme tasavertaisia, mikä toteutuu parhaiten istumalla ringissä. Kukaan ei ole piilossa ja kukaan ei ole johtajana pöydän päässä.

Lähtekää metsään paistamaan makkaraa! Tuli luo rauhoittavan tunnelman. Välttämättä kenenkään ei tarvitse puhua, voi vain katella tulta ja olla omissa ajatuksissaan. Vieraassa ympäristössä on myös helpompaa tarttua työpaikalla vaivaaviin kysymyksiin. Älkää kuitenkaan lähtekö nuotioreissulle ilman, että kaikki ovat saaneet reissusta tiedon. Hyvä dialogi vaatii kaikkien ajatuksia, niin puolesta kuin vastaan. Ja kaikki ideat vaativat kaikkia mukaan toteutukseen. Aidon, vuorovaikutteisen dialogin kautta jokainen pääsee tuomaan omat ajatuksensa esille ja on myös sitoutuneempi silloin yhteisiin päätöksiin ja toimenpiteisiin.

Olli Leppänen
gsm. 044 352 3479
email: olli@lahde.net

Olli toimii yrittäjänä ja valmentajana Lähde Valmennuksessa. Yhteisöllinen oppiminen, luovuus ja yrittäjyys ovat keskeisinä teemoina Lähteen valmennuksissa. Lisätietoja ja tarinoita löydät osoitteesta www.lahde.net.

Suosittelavaa kirjallisuutta:

William Isaacs: Dialogi ja yhdessä ajattelemisen taito

Heikkilä & Heikkilä: Dialogi, avain innovatiivisuuteen

Jussi T. Koski: Luova hierre

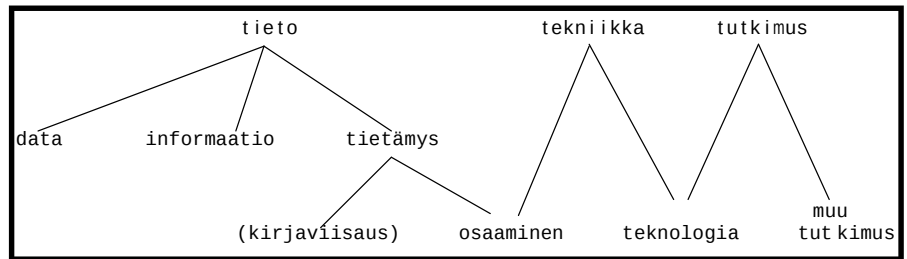
Onko suomella tulevaisuutta?

Ilmari Pietarinen,
Tietotekniikan liitto ry,
Sanastotoimikunta.

Tätä kysyvät huolestuneet kansalaiset kielenhuoltajilta tavan takaa. Huoltajat vastaavat, että suomen kieli voi hyvin ja kehittyä pulskasti. Tietotekniikan kielen kehitystä neljä vuosikymmentä seuranneena en jaksa sen osalta olla yhtä toiveikas, sillä fingelska jyrää aina vain vahvemmin. Olenkin kateellisena katsellut, miten muilla teknisillä aloilla hyvät suomalaiset termit tuntuvat menevän vaivatta läpi viestimiin ja jopa ammattiväen käyttöön. Bagdadiin tiputettiin täsmäaseita, ei älypommeja, kun Saddamilla luultiin olevan ydinkärkiä, ei nukleaarisia sotapäitä. Ja Teksasin taivaalla hajosi avaruussukkula, ei esimerkiksi speissisutteli.

Toisaalta, pitäisikö meidän edes yrittää ylläpitää suomea tietotekniikan kielenä? Jopa oman kielen tukena ja turvana aiemmin tunnettu ns. kielitoimisto tuntuu kääntäneen kelkkansa: kaikki käy. Asia ei ole tietenkään yksioikoinen. Fingelskan puhujalla saatetaan olla matalampi kielikynnys ylitettävänä kansainvälisessä yhteydenpidossa, kun taas omaksumalla ohjelman ja oheislaitteen edistämme kotimaista viestintää ja autamme ugrilaisen suomen säilymistä kartalla. Molemmat päämäärät ovat tärkeitä.

Kun Tietotekniikan liiton, silloisen Tietojenkäsittely-yhdistyksen sanastotoimikunta aloitti työnsä vuonna 1966, dilemma oli jo tunnettu ja ratkaistiin niin, että toimikunta pyrkii luomaan suomenkielistä terminologiaa tavalliselle käyttäjälle, ekspertit puhuivat slangiaan. Tähän se pyrkii edelleen.



Kuva 1. Tieto ja sen lähisanoja

37 vuotta sitten tietotekniikan käyttäjiä oli vähän ja heistäkin suurin osa alan ammatilaisiksi luettavia. Nyt olemme kaikki enemmän tai vähemmän tavallisia käyttäjiä. Onko ATK-sanakirja tänä aikana pysynyt muutosten mukana? Mitkä olivat atk-termityön tavoitteet ja työtavat alkuaikoina ja miten ne ovat kehittyneet? Mitä on opittu - kantapään kautta ja muuten?

Työn tavoitteet eivät ole muuttuneet

Sanastotoimikunnan ensimmäisen julkaisun (18.7.1966) esipuheessa esitettiin tavoitteeksi palvella tavallista käyttäjää. Tämä merkitsi, että ulkomaisia alan lähteitä hallitsevat tekniset ja tieteelliset termit jätettiin vähemmälle, ja niiden sijaan otettiin tietojenkäsittelyn yleiskäsitteitä ja alan kirjallisuudessa usein esiintyviä lähialojen sanoja. Se merkitsi myös, että toimikunta pyrki mahdollisuuksien mukaan suosimaan suomalaisia tai suomalaisista sanajuurista johdettavia termejä.

Kirjasen pitkä johdanto esitteli ja puolusteli toimikunnan ajattelutapaa yli sadan esimerkin osalta. Se otti kantaa "elektronista" vastaan ja automaattisen puolesta: "ratkaisevaa on, suoritetaanko .. käsittely .. muistiin talletetun ohjelman toimesta, ei se, sisältyykö laitteeseen elektroniikkaa." Tätä

sanaista kamppailua käydään edelleenkin, kun jenkkien elektronit tunkevat takaisin suomeen aina uusissa yhteyksissä. Nykyisin suositeltavia käännösvaihtoehtoja ovat myös verkko- ja sähkö-(inen).

Monikollisen "datan" vastineeksi annettiin monikkomuotoinen "tiedot", siis esim. alkutiedot, tulostiedot. Tässä on sittemmin jouduttu osittain perääntymään ja hyväksymään yksikkömuodoksi ymmärretty data suomen sanaksi. ATK-sanakirjan nykynäkemys tiedon, datan ja eräiden lähisanojen suhteista ilmenee kuvasta 1.

Urheilta alkuajoilta on myös peräisin toimikunnan keksimä rekordin vastine tietue, joskin tietojaksolla oli myös ryhmässä kannattajia. Tiedostoa suositeltiin fileen tilalle, mutta se ei ollut toimikunnan keksimä, vaan jo kokeilumielisesti käytössä. Tiedoston "leipelistä" tehtiin nimiö, joka oli uusi termi atk:ssa vaikeikaan kirjapainoalalla. Access käännettiin saanniksi ja on sinä aika hyvin pysynytkin.

Vaikeita olivat silloin ja ovat edelleen offline ja online. Edellinen käännettiin erilliseksi, eikä parempaa liene vielä keksitty. "Sen vastakohta on .. on-line, jolle ehdotetaan käännöstä linja-; on line = linjassa." Ei hassumpi yritys, nykysuositus on "(suoraan) kytetty(nä); verkko-".

60-luvun pop-sanoja olivat multiprogramming, multiprocessing ja time sharing. Näistä oli johdannossa lähes sivun verran tekstiä, jolla perusteltiin suosituksia moniajo, monikäsitteily ja osuuskäsittely. Sanaston myöhemmissä painoksissa osuuskäsittely muuttui osituskäytöksi. Muilta osin nämä varhaiset ehdotukset tulivat jäädäkseen; säästyttiinpä ainakin multiohjelmoinnilta tai multiprosessoinnilta. Mielestäni merkittävän panoksen suomen hyväksii ryhmä antoi myös ehdotuksellaan sivuajo käännöksenä IBM:n SPOOL-tekniikalle, eräs moniajon laji, jota silloin yleisesti kutsuttiin spuulaukseksi.

Softwaren kääntäminen ohjelmistoksi, alalajeina mm. ohjausohjelmisto, tehtäväohjelmisto ja sovellutusohjelmisto, saa laajat perustelut, jotka näköjään ovat vakuuttaneet. Myöhemmin sovellutus lyheni sovellukseksi, kun huomattiin, ettei Sibeliuksen sävellyttänyt, vaan sävelsi sävellyksenä sitä itse.

Nämä ovat vain esimerkkejä kirjajen kymmenistä kehitelmistä. ATK-sanakirjan ensimmäistä versiota edeltänyt puolen vuoden työ on jäljistä päätellen ollut hyvin tuloksekasta. Useimmat ehdotukset ovat vakiintuneet sellaisinaan, toiset vähän muuttuneina. Kovin pitkälle ei kuitenkaan pötkinyt ehdotus volyyymi = tiedokko, joka on nykyisin taltio.

Toisen, heinäkuussa 1967 ilmestyneen atk-sanaston johdannossa todettiin pulma, joka kroonisesti vaivaa termityötä nopeasti kehittyvällä alalla: päiväperhoset. Eli johdannon sanoin: "On .. uskallettua .. kääntää suomeksi uusi termi heti kun se on .. havaittu kirjallisuudessa, sillä uutuukseen kuolleisuus ja muuttuvuus on suuri." Kielen sanavarasto, varsinkin suomen, on rajallinen ja hitaasti uusiutuva luonnonvara. Jos uudelle laitteelle,

ohjelmatyypille tai menetelmälle osoitetaan termi ja itse kohde ei koskaan tulekaan laajaan käyttöön tai häipyä pian markkinoilta, on hukattu niukka resurssi, jota ei enää voi käyttää muualla. Toisaalta, jos uutuudelle ei kohta anneta nimeä, ja se jääkin eloon, voi pian olla liian myöhäistä ja käyttöön jää jokin kiireellä kyhätty fingelska-ilmaisu.

Eloonjäämisen taisto

Vuoden 1971 ATK-sanakirjan alkulause ilmaisee työryhmän ilon siitä, että käyttäjät ovat suurimmaksi osaksi hyväksyneet sen ehdottamat uudissanat. Painos sisältää kuitenkin ainakin yhden uudissan, jota käyttäjät eivät ottaneet omakseen. "„etäiskäsittelyn .. alalta lienee syytä mainita "tiedin", jolla tarkoitetaan laitetta, jonka avulla ihminen voi olla suorassa yhteydessä tietokoneeseen. Ennustusten mukaan tiedonvälitysverkkoon kytketyt tietimet ovat tulevaisuudessa yhtä jokapäiväisiä kuin puhelinverkkoon kytketyt puhelimet tänään." Vaikka ennustukset ovat käyneet toteen, tietimiä ei tänään ole. Sen sijaan on kannettavia, sylimikroja, kämmyköitä, muistikirjoja, lehtiömiikroja ja monia muita erinimisiä pikku vempelaitteita.

Toimikunnan puolustukseksi on sanottava, että tiedin ei ollut sen keksintö, vaan ehdotuksen teki Suomen Pankin atk-päällikkö Kaino Eirola. Hänen puolustukseksi taas voidaan sanoa, että tämä mielestäni nerokas ehdotus epäonnistui kahdesta täysin ulkopuolisesta syystä: d-kirjain ja se, että jenkeillä ei ollut vastaavaa sanaa.

Kolmonen kohtalon koki myös joidenkin vuosia myöhemmin lanseerattu anne, josta toivottiin käännöstä datale. Sana tuli Kielitoimistolta, joka oli johtanut sen moitteettomasti: latinan

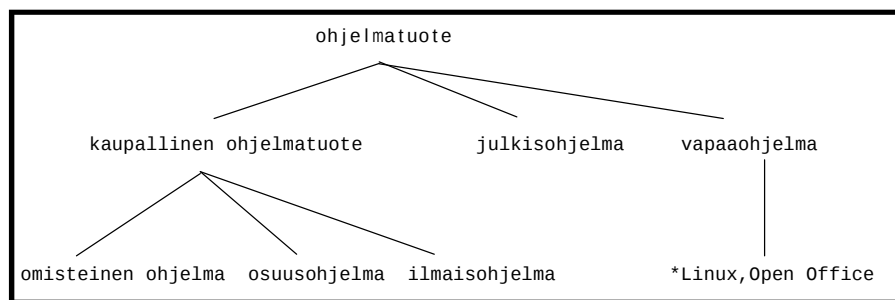
dare = antaa, datum/data = annettu/annetut. Sukukielet olivat jo vastaavan tempun tehneet täysin menestyksekkäästi, virossa on andmed ja unkarissa adat (verbistä adni, antaa). Mutta kun ei. Hanke kompastui siihen, että Miss Suomen nimi oli Anne Pohtamo ja kuukaudesta toiseen ilta-päivälehdet olivat täynnä otsikoita Anne sitä, Anne tätä. Älköön Suomen datakansalla olko kahta anneä.

Kehityksen kelkassa

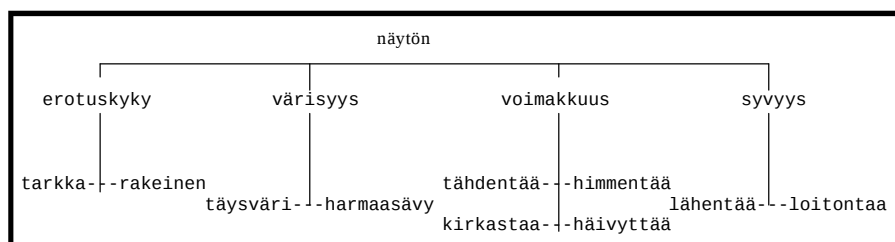
ATK-sanakirjalle on ominaista normatiivinen, käytäntöä ohjaava ote. Sanaa haetaan analysoimalla ja määrittelemällä ensin kyseinen käsite: mikä, millainen, miksi, miten erottuu muista? Onko mahdollista luoda osuva suomalainen termi? Tässä suhteessa ATKS eroaa muista tarjolla olevista sanastoista, jotka tuntuvat olevan kaikkiruokaisia.

Vaikka johtotähti on 37 vuoden ollut samassa paikassa eli kotikoivun latvuksissa, ovat sekä ulkoiset että sisäiset muutokset olleet suuria. Alkuvuosina normatiivisuus puri hyvin, viime vuosina ehkä huonommin. Tästä huolimatta sanastotoimikunta ei ole ratkaisevasti muuttamassa työtapaansa. Liikettä on kuitenkin tapahtunut. Suomalaisuudesta, joka ei ole koskaan ollut ehdotonta tai "supia", on edelleen tingitty ja huonosti menestyvistä ehdotuksista luovuttu aiempaa herkemmin.

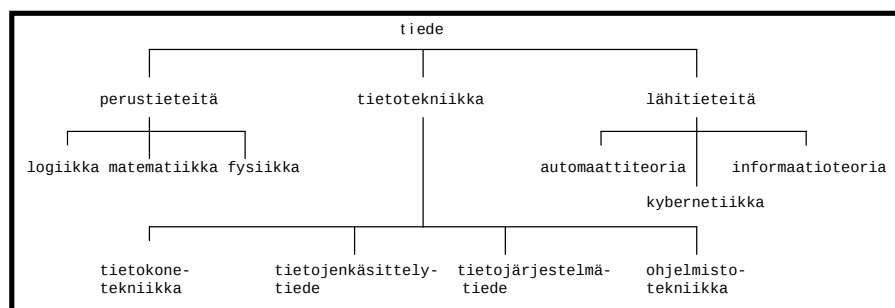
Kilon painoinen kovakantinen ATK-sanakirja mallia 2003 on tietenkin täysin eri opus kuin IBM:n offset-painossa tuotettu keltainen, kahdella nitillä koossa pysyvä Atk-alan sanaluettelo vuodelta 1966, joka sisälsi noin 30 sivun verran pölkkykirjaimista listausta: englannin sana - suomen sana. Tätä seuraava julkaisu tarjosi sanavaraston myös suomesta englantiin, mis-



Kuva 2. Ohjelmatuote käyttöoikeuden mukaan



Kuva 3. Näytön ominaisuuksia ja toimintoja



Kuva 4. Tietotekniikkaan liittyviä tieteenaloja

tä oli lyhyt matka - mutta paljon työtä - ryhtyä varustamaan suomen sanat suomenkielisin määritelmän.

Koska aineisto oli alun alkaen tietokoneella, oli luonnollista pyrkiä tuottamaan sille lisäarvoa ohjelmoimalla sen käsittelyä ja tulostusta monipuolistavia toimintoja. Eräs tällainen on ollut sanojen välisten viitteiden automaattinen luominen. Myöhemmin lisättiin "käsini" tehtyjä viitteitä sekä viitteiden "periytyminen" suomen kielestä muihin kieliin.

Vuosien kuluessa toimikunnalle ilmoittautui termikärpäsen puremia asian harrastajia, jotka olivat askarrelleet jonkin kielen atk-sanaston paris-

sa ja tarjosivat yhteistyötä. Tarjoukset otettiin ilmielin vastaan, ja näin kielivalikoima kasvoi: ranska, ruotsi, espanja, venäjä (opinnäytetyönä), saksa, viro (lahdentakaisena yhteistyönä). Pulmana kielissä on ollut niiden ylläpito alkuunpanijan siirtyessä muihin tehtäviin tai harrastuksiin.

Kuvioita, pääasiassa käsitekaavioita alettiin piirtää sanakirjaan noin kymmenen vuotta sitten. 12. painos sisältää niitä useita kymmeniä. Kokeemus on osoittanut, että on lähes mahdotonta piirtää loogisesti vedenpitävä käsitekaavio, joka ei ole samalla triviaali. Informatiiviseksi tarkoitettu kaavio taas sisältää yleensä aukkoja ja epätarkkuuksia, jopa epäloogisuuksia.

Vasemmalla vielä muutama esimerkki sanakirjan kaavioista.

Tällä hetkellä puolet sanakirjan sivumäärästä kuuluu määritelmäosaan, jossa ovat termit suomeksi ja muilla kielillä, määritelmät, kuvat ja ristiviittaukset. Toinen puoli käsittää seitsemän sanaluetteloa muista kielistä suomeen. Laaja kielivalikoima luo mielenkiintoisen mahdollisuuden tuottaa lisää sisältöä automaattisesti: kaksipuolisia sanaluetteloa pareittain kaikkien kahdeksan kielen välillä. Näin laskien aineisto sisältää piilevänä kaikkiaan $8 \times 7 = 56$ sanakirjaa!

ATK-sanakirjan 12. painosta vastaava aineisto sisältyy nyt myös Kielikone Oy:n tuottamaan ja markkinoimaan MOT sanakirjastoon. Oteita sanakirjasta, samoin kuin sanastotoimikunnan aivoituksista ylipäätään, on jo muutaman vuoden ajan julkaistu Tietotekniikan liiton web-portaalissa www.tt-tori.fi.

Sanastotoimikunta ja muu suomi

Pitkän toimikautensa aikana toimikunnalla on ollut paljon kontakteja tietotekniikan osa- tai lähialueiden termityötä tekeviin tahoihin. Näiden toiminta vaikuttaa viime aikoina laantuneen. ISO/IEC:n puitteissa tehty alan kansainvälinen sanastotyö lopautti jo muutama vuosi sitten rahoituksen puutteeseen. Kotimaisia hankkeita-kaan ei tunnu olevan liiemmästi liikkeellä. Ilahduttava tieto tämän vuoksi on, että Sytykkeellä on tällaisia menossa. Olisi suotavaa saada aikaan yhteistointintaa esimerkiksi niin, että toimikunta voisi saada nähtäväksi ja kommentoitavaksi sanastoversioita ennen kuin ne menevät painoon.

*Ilmari Pietarinen,
Tietotekniikan liitto ry,
Sanastotoimikunta*

Ovatko oliot objekteja?

Lauri Laitinen,
Nokia Research Center

Olio- ja kohde-sanojen käyttö atk-suomessa on eräs esimerkki siitä, miten oleellisen ymmärtävä käännös on alkukielistä ilmausta täsmällisempi.

Object and entity

Sanakirjan mukaan entity on olevainen, itsenäinen kokonaisuus eli tosiole. Suomenkielessä hahmoltaan epämääräistä, mutta kuitenkin itsenäisesti olemassa olevaa kohdetta sanotaan olioksi. Entity sanan suomenkielinen vastine on siten olevainen tai olio.

Object tarkoittaa sanakirjan mukaan yleisen englantilaisen sanastoperaation mukaan vähän kaikkea kuten esinettä, kohdetta, tarkoitusta, päämäärää ja objektia. Havaintojeni mukaan englannin kielisen sanan voi ymmärtää vain jos onnistuu kiteyttämään sanan lukemattomista merkityksistä sen semanttisen perusidean. Object:in perusideana on toiminnan tai tekemisen kohteena oleminen. Tieteellisessä kielenkäytössä objekti on vakiintunut käytettäväksi subjektin vastakohtana. Nykyisin objektin käsite ei oikein ole edes ymmärrettävissä ilman subjektin ideaa. Object on siten suomen kielessä lähinnä tekemisen kohde.

Atk-sanakirja antaa oliolle kaksi käännöstä:

- yleiskieleen perustuvan entity:n eli ympäristöstään erottuvan kokonaisuuden
- atk-kielenkäytöstä lähtevän object:in eli toimitusten muodostaman kokonaisuuden.

Harri Laineen kokoamassa oliosanasto (www.cs.helsinki.fi/u/laine/olio-sanasto) määrittelee olion (object):

- 1) Ympäristöstä erottuvaksi kokonaisuudeksi.
- 2) Tietokokonaisuudeksi, joka tarjoaa myös palveluja sisältämien tietojen käsittelyyn.
- 3) Luokan ilmentymäksi.

Mitä ovat atk-oliot?

Mitä on sitten esimerkiksi object oriented design? Sananmukaisesti käännettynä se on kohdesuuntautunut suunnittelu. Nykysuomeksi kuitenkin lyhyesti olionsuunnittelua. Kuka vaihtoi objektit eli tekemisen kohteet olioiksi? Mistä ne kohteet sitten löytyvät?

Kohteet löytyvät tietokantasuunnittelusta. Esimerkiksi käsiteanalyysissä käytetään kuvausmenetelmänä yleisesti kohdekaavioita eli entity-relationship diagrammeja. Entity eli olio tai olevainen on tietokantapuolella käännetty kohteeksi. Nykyisin tosin käsitekaaviota ei enää kutsuta kohdekaavioksi, vaan luokkakaavioksi (class diagram), joka koostuu luokista (class) ja niiden välisistä yhteyksistä eli assosiaatioista.

Kuka teki virheen?

Virheen tai lähinnä epätasallisuuden aiheuttivat englannin kieliset atk-pioneerit.

Tässä toistuu se yleinen ilmiö, että useimpien asioiden suomenkieliset termit ovat huomattavasti kuvavampia ja täsmällisempiä kuin alkupe-
räiset englanninkieliset.

Kun aika tulee kypsäksi, jonkin uuden asian oivaltamiseen, keksitään perusideat tai löydökset eri puolilla maailmaa riippumattomasti. Tieteen historian alaviitteistä voidaan havaita tämän tapahtuvan yleensä vähintään kolmesti. Keksijäksi katsotaan se, joka ehti tehdä ideansa perusteet riittävän tunnetuksi englanninkielellä yleensä muutamaa kuukautta toisia aikaisemmin. Tosin atk:ssa menetelmät nimitetään niiden mukaan, jotka kirjoittavat myydyimmät oppikirjat. Joka tapauksessa maailman kulttuurivuorovaikutuksen suunta on nykyisin se, että englanninkielisessä maailmassa syntynyt tai englannin kielellä tarpeeksi tehokkaasti dokumentoitu kehityslinja omaksutaan automaattisesti maailman standardiksi.

Alan perusterminologiaksi vakiintuu sitten yleensä alkuperäisen julkaisun keskeiset englanninkieliset sanat. Useinkaan keksijä eli guru ei itse kuitenkaan ole loppuun asti tajunnut, mistä hänen keksinnöstään on pohjimmitaan kysymys. Hän on vain pyrkinyt nopeasti löytämään ideansa jotenkin kuvaavia englanninkielisiä yleisanoja, jotka eivät vielä ole liian varattuja hänen omalla erikoisalallaan. Tämän periaatteen mukaisesti ekonomit ovat voineet huoletta uusiokäyttää esim. insinöörien re-engineering terminkin.

Yleensä nämä englantilaiset sanat elävät suomenkielessä muutaman vuoden yliopistollisten ja konsulttien tarjoamien kurssien nimissä. Jossakin vaiheessa saavutaan suomenkielellä yhteisymmärrys siitä, mistä pohjimmitaan on kysymys. Vasta tämän oivaltuksen jälkeen vakiintuvat suomenkieliset termit.

Menetelmä	Menetelmän perusalkio englanniksi	Perusalkion yleismerkitys englanniksi	Menetelmän perusalkio suomeksi
Oliosuunnittelu	Object	Kohde	Olio
Käsiteanalyysi	Entity	Olio	Kohde

Kuva 1. Suomen ja englannin perusalkioiden nimet ja merkitykset

Näin juuri tapahtui Suomessa entity/kohde- ja object/olio-sanojen suhteen.

Mitä ovat oliosuunnittelun oliot?

Olioajattelussa olioita ei mieletä tekemisen kohteiksi eli objekteiksi vaan pikemminkin tekijöiksi eli subjekteiksi, jotka pyytelevät toisilta palveluja. Itse asiassa kyse on subject oriented -metodologioista.

Oliot voidaan tosin mieltään myös käsiteanalyysinäkökulman mukaisesti tarkastelun kohteina olevien reaali-maailman todellisten toimijoiden edustajiksi.

Ensimmäistä mielikuvaa käyte-tään luotaessa teko-olioita kuten käyt-töliittymäolioita. Jälkimmäinen mieliku-va sopii reaali-maailman fyysisen ja so-siaalisien (yritykset, henkilöt) todellisuus-todellisuuden kuvaamiseen olioilla.

Oliotermi on kohdetta parempi valinta oliosuunnittelun perusalkion nimeksi. Olio ei ota kantaa siihen mistä kulloinenkin abstraktioolio on muodostettu. Englannin mukaisen kohdetermin (object) käyttö sitoisi oliomenetelmät liikaa reaali-maailman olioiden edustajien (kuvien) käsittelyyn.

Objekti- eli kohdetermin käyttö oliosta rajoittaa ajatukset herkästi pel-kästäään todelliseksi koettujen asioiden mallintamiseen. Suomalainen termi olio helpottaa itsenäisten toimijoiden luomiseen perustuvan suunnittelume-

netelmän omaksumista ja sen uusien mahdollisuuksien hyväksikäyttöä. Suomalainen ei siten niinkään mallita kohteita vaan luo itsenäisesti uusia toimintaolioita. Toimintomallituksesta lähtevä oliosuunnittelu traditio voi siten olla Suomessa luonnollisempi kuin anglosaksisessa maailmassa.

Mitä ovat käsiteanalyysin kohteet?

Onneksi tietokantojen ja käsite-analyysin tutkijat ja kouluttajat tekivät Suomessa hyvää työtä ja valitsivat huolella englannin entity-sanana suomalaisiksi vastineeksi kohteen sananmukaisen käännöksen - olio - sijasta. Sanan valinta korostaa hienosti tietokantojen todellisuutta kuvaavaa luonnetta.

Varaamalla kohde-termin käsite-analyysin käyttöön tietokantaihmi-set estivät myös myöhemmin levinneen olio-ohjelmoinnin harrastajia käyttämästä object:in sananmukaista käännöstä kohde. Atk-kulttuuri oli Suomessa sen verran yhtenäinen, ettei samojen sanojen päällekkäinen käyttö ollut yleensä mahdollista.

Olio-ohjelmoijat joutuivat ajatte-lemaan ja huomasivat objektien ole-vankin pohjimmiltaan epämääräisiä mutta itsenäisen olemassaolon omaa-via kohteita eli suomeksi olioita.

Sananmukainen kääntäminen olisi johtanut englanninkielen mukai-sesti suoraan kohdeohjelmointiin, koh-desuunnitteluun jne.

Sitä vastoin relaatiotietokantojen seuraajaksi jo lähes pari vuosikym-mentä kaavailtujen oliotietokantojen suorakäännetty nimi - kohdetietokanta - ei tunnu lainkaan oudolta. Vakiin-tuneen käytännön mukaan tietokanta-tyyppien tarkennin ilmoittaa tapaa millä tiedot on loogisesti organisoitu tietokannassa. Esim. relaatiotietokannassa on relaatioita jne. Sovelluskohtaisesti tietokantoja nimetään myös so-velluksen pääkohteen mukaan esimer-kiksi henkilökannoiksi, vakuutuskan-noiksi jne.

Kohdetietokanta-nimitys olisi kuitenkin jälleen kerran suomessa, kuten on luultavasti englannissakin, hämäävä. Nimitys ei toisi esille olio-tietokantojen oleellista eroa muista tietokantamalleista, sillä kaikki tietokannat tallettavat enemmän tai vähem-män tarkoituksenmukaisesti mallitet-tuna tietoja tarkastelun kohteista. Ai-nakin suomalainen selkeä kielenkäyt-tö on saanut minut näin ajattelemaan. Oliokannoissa tarkastelun kohteet on vain organisoitu olioiksi.

Entäpä sitten kohdeoliot?

Kohde-sanana monitulkintaisuutta tilanteissa, joissa on monia hierark-kisiakin objekti-subjekti-pareja, osoit-taa mm., että Systeemityölehden oliomenetelmät teemanumerossa (1/1995) kaksi kirjoittajaa käytti "kohdeolio"-termiä erimerkityksessä. Nykyisin tämä 1990-luvulla yleinen kohdeolio-termi on jäänyt pois kielenkäytöstä.

Anne Putkonen tarkoitti kohde-oliolla oliomallin oliota, joka on tarkas-telun eli osittamisen kohteena. Koh-deolio oli siis oliomallin suunnittelupro-sessin kohde.

Markku Nokso-Koivisto tarkoitti kohdeoliolla reaali maailman sovelluksen kohdejärjestelmän olion edustajaa riippumatta siitä, onko se tarkastelun kohteena. Kohdeolio oli siis selvästi tietojärjestelmän toiminnan kohde.

Itse pitäisin Anne Putkosen tapaa käyttää "kohdeolio"-termiä luonnollisempana. On selvempää, jos sana kohde viittaa aina lähimmän prosessin eli tässä tapauksessa suunnittelu-prosessin subjekti-objekti pariin. Markku Nokso-Koiviston kohdeolioita kutsuisin siten pikemminkin esimerkiksi sovellusolioiksi. Tosin olisi tietysti loogista, jos kehittämisen kohteena olevan kohdejärjestelmän olioita kutsutaisiin kohteiksi. Sovelluspainotteinen kohdeolion merkitys on kuitenkin joka tapauksessa vain reaali maailman tai siihen kuvitellun kohteen edustaja oliomallissa.

Sovellusolio (eli joidenkin kohdeolio) tulee varsinaisesti ymmärrettäväksi vastakohtansa, teknisen eli teko-olion, kautta.

Teko-olioista tunnetuimpia ovat graafisten käyttöliittymien mukana yleistyneet ns. käyttöliittymäoliot. Käyttöliittymäolio ei varsinaisesti ole

subjekti tai objekti vaan lähinnä viestinviejä, vaikutuksen älykäs välittäjä.

Uutena oliotyyppinä ovat mm. innovatiivisissa kaavion piirto-ohjelmis-sa yleistymässä ns. piirto-oliot. Piirto-oliot eroavat käyttöliittymäolioista siinä, että ne ohjaavat vain itseään. Käyttöliittymäolioiden kautta ohjataan joi-tain kolmatta osapuolta. Piirto-olioilta pyydetään olion itsensä ulkonäköön tai kytkentöihin liittyviä palveluja. Perinteisten piirto-ohjelmien valmiit tai kopiaitavat kuvaelementit ovat varsinaisia tyhmiä objekteja, joita saattoi ai-noastaan käsitellä kohteina, skaalata, värittää jne..

Piirto-olioon sen sijaan voi liittyä esimerkiksi taulukkolaskentamuotoi-sesti kuvattu ohjelma, joka muuttaa oliota piirtäjän antamien etukäteisoh-jeiden mukaisesti ympäristön muuttu-essa. Tulevaisuudessa kaavion piirto-ohjelmien kuvaelementit voivat olla todellisia ohjelmoitavia oliota eivätkä pelkästään graafisia objekteja

Erilaisia todellisuus käsityksiä

Toinen mielenkiintoinen atk-kum-majainen on virtuaalinen-sanan käyt-tö atk-alalla. Englannin sanakirjan

mukaan virtual tarkoittaa yleiskieles-sä tosiasiallista, todellista ja atk-termi-nä näennäistä (WSOY Eng-Suomi suursanakirja). Esimerkiksi virtuaali-muisti käännetään Atk-sanakirjassa näennäismuistiksi.

Käännös kuvastaa ehkä kulttuu-risidonnaisia eroja näkemyksessä to-dellisuudesta.

Yhdysvaltalaisen markkinointi-henkisen näkemyksen mukaan virtu-aalimuisti on todellista, koska se on se muistimäärä, jonka käyttäjä kokee sub-jektiivisesti laitteessaan olevan. Siis käyttäjän näkökulma on todellisuus.

Suomessa on omaksuttu ehkä saksalaisesta vaikutuksesta insinööri-mäisempi objektiivisuuteen pyrkivä näkemys, jonka mukaan ainoastaan se, miten järjestelmä on todella muo-dostettu on todellista. Tämän näke-myksen mukaan virtuaalimuistia, jota ei todellakaan ole olemassa fyysisesti tietokoneen keskusmuistissa vaan ai-noastaan loogisena osoiteavaruutena, on näennäistä.

*Lauri Laitinen,
lauri.laitinen@nokia.com*

Ontologia on tapa esittää tietoa

Helena Venäläinen,
FD Finanssidata Oy

Kreikan kielessä sana logos tarkoittaa tietoa. Filosofiasa ontologiset teoriat pyrkivät löytämään tavan kuvata kaikkia olevaa. Ontologioita on tehty eri alueille ja erilaisten ongelmien ratkaisuun. Yhteisten merkitysten ja esitystapojen sopiminen liittyy oleellisenä osana eri tietojärjestelmien yhdistämiseen ja semanttiseen webiin. Mallinnusteorioiden ja verkkopalveluiden yhteydessä ontologioilla tarkoitetaan yhteistä käsitteistöä koskevia määrittämiä. Tämä artikkeli valottaa ontologioiden merkitystä tietojärjestelmien ja erityisesti verkkopalveluiden rakentamisessa.

Ontologiamääritelmiä ja -luokkia

Ontologia on yleisnimitys monenlaisille kuvaustavoille ja kieliopille. Ontologisilla teorioilla on haettu vuosisatoja selitystä mm. kysymyksiin: ”Millaisia asioita on olemassa, miten ne liittyvät toisiinsa, millaisia tiloja niillä on?”. Geerts ja McCarthy [G&M00] toteavat, että vastaus riippuu tarkastelukulmasta. Hyvin yleisiä (kuten aika ja avaruus) ja alueesta riippumattomia ontologioita he kutsuvat (upper-level) teoriaontologioiksi. Alueontologioiksi (domain) he luokittelevat tietyille, rajatulle alueelle suunnitellut ontologiat. Esimerkkinä he mainitsevat aikataulutukseen, lento-reittien suunnitteluun, lääketieteeseen ja yritysten tarpeisiin tarkoitetut ontologiat.

Ontologia-guru Sowa [Sow97] luokittelee ontologioita niiden muodollisuuden ja määrittelymenetelmän mukaan. Luonnollisella kielellä tehty tyyppiluettelo on hänen luokittelunsa mukaan epämuodollinen ontologia. Muodolliseksi ontologia muuttuu, kun siinä määritellään käsitteiden ni-

met ja suhteet ja kuvataan alityyppien suhteet. Alityyppien erottelu päätyypeistä voi perustua aksioomiin ja muodollisella kielellä tehtyihin määrittämiin (matematiikka, fysiikka, tekniikka) tai alityyppien prototyyppeihin (kasvit, eläimet, normaalin elämän ilmiöt). Suuret ontologia voivat sisältää erilaisia määrittämenetelmiä.

Gruber [Gru93] on määritellyt ontologian selkeäksi kuvaukseksi käsitteellistämistä. Käsitteellistämällä Gruber tarkoittaa ”kohteita, käsitteitä ja muita kokonaisuuksia (entities) joita oletetaan esiintyvän joillakin kiinnostuksen kohteena olevalla alueella sekä niiden välisiä suhteita”. Vaikka määritelmä muistuttaa käsittekaaviota, se eroaa siitä ainakin tavoitteen, laajuuden ja sisällön suhteen:

- Tavoitteena on esittää käsitteellistämisen siten, että se on jaettava ja uudelleen käytettävissä eikä sisällä sovelluskohtaisia erikoisuuksia.
- Ontologia kattaa alueen kaikki sovellukset, ei vain yhtä.
- Ontologian tietämyksen (knowledge) kuvaus ilmaisee tarkasti kuvattujen rakenteiden merkityksen ja tulkinnan sekä säännöt uuden tietämyksen päättelemiseen.

Ontologioita voidaan luokitella myös niihen käyttöalueiden perusteella

- 1) tietohakemistoihin,
- 2) käsitteellisen suunnittelun tukeen ja
- 3) toiminnallisuuteen liittyviin.

Jaottelu ei ole kaiken kattava eikä mikään mainituista alueista ole toisensa poissulkeva. Esimerkiksi tietohakemistolla voi tukea käsitteellistä suunnittelua ja sen tekemisessä voidaan käyttää mallinnuksen apuna toiminnallista ontologiaa [G&M00].

- 1) Tietohakemisto (*Knowledge Dictionary*) ontologia kuvaa alueen käsitteiden merkitykset, niiden väliset suhteet ja käsitteiden muodostamat rakenteet. Tarkasti dokumentoidut määrittäykset parantavat kommunikointia, yhteenliittämistä ja yhdenmukaisuutta
- 2) Käsitteellisen suunnittelun tukena (Conceptual Design Support) ontologiat voivat tarjota merkittävän avun tietyn alueen sovellusten mallien rakentamiselle. Geerts ja McCrathyn lisäksi myös Weber [Web02] toteaa ontologioiden tulevaisuudessa muodostavan tärkeän pohjan käsitteellisen mallinnuksen tutkimukselle, käytännöille ja pedagogialle. Mallinnusteorioiden teoreettisen perustan vahvistamisen ontologoilla uskotaan parantavan niiden laatua ja sitä kautta myös niiden avulla kuvattuja järjestelmiä.
- 3) Toiminnalliset ontologiat (Operational Ontologies) perustuvat käsitteiden ja niiden suhteiden tallettamiseen siten, että niistä tulee osa itse sovellusta. Ontologian suora tallettaminen tietomäärittämisiksi mahdollistaa niiden käyttämisen päättelemiseen.

Alueontologioista

Rajatulle alueelle suunnitellut ontologiat, kuten esimerkiksi yritysten verkkoliiketoimintaa kuvaavat, voidaan luokitella alueontologioiksi. Seuraavassa tarkastellaan hiukan tarkemmin, millaisia verkkoliiketoiminta-alueen yleisiä ilmiöitä (ja miten) ontologioilla voidaan kuvata.

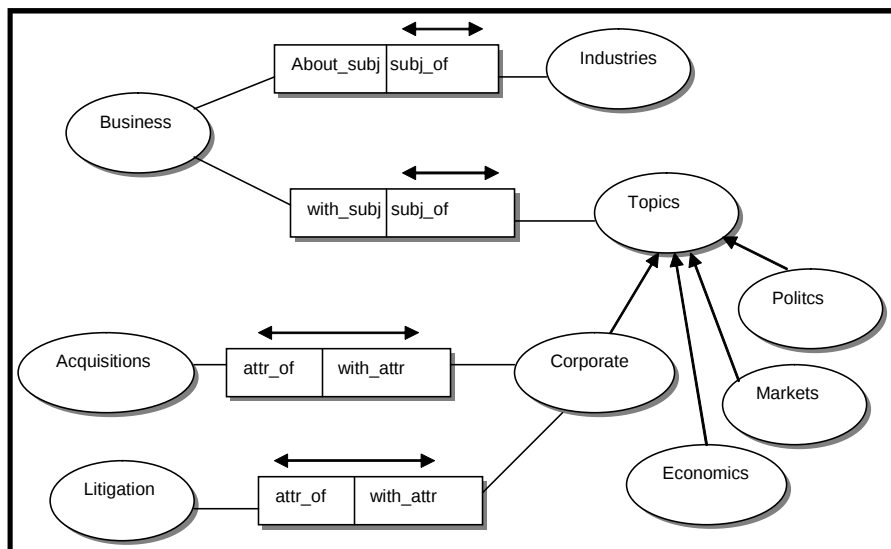
Meersman on määritellyt ja luokitellut semanttisen webin yhteydessä käytettyjä termejä seuraavasti:

TOPICS
CORPORATE
... ..
Accounts Health
Acquisition Human Interest
Asset Transfers Obituaries
Bond Religions
Capacity/Plant Sciency/Technology
Capital Sports
Competition/Anti-Trust Travel/Tourism
... ..

Kuva 1: Esimerkki-thesaurus Yritys. Mukaellen [Mee99: Fragment of Reuters: Corporate Topics]

INDUSTRIES
...
Accountancy
Adhesives
Advertising
Aerospace
...
Baking
Banking
Basic Industrial Making
Brewing
...

Kuva 2: Esimerkki-thesaurus Yritys. Mukaellen [Mee99: Fragment of Reuters, Industries]



Kuva 3: Esimerkki ORM-yhteykskaaviosta Meersmania [Mee99] mukaellen.

Ontologia on oppi olevasta, lista siitä mitä on olemassa, siis käsitteellistämisen määrittely.

Thesaurus on joko aluekohtainen ontologia, luokiteltu lista termejä ja avainsanoja kuten esimerkiksi Yritys_ tai Pankki_toiminta, Sopimus_laki

tai sovelluskohtainen: esim. Lento_varaukset, Conferenssin_järjestäminen.

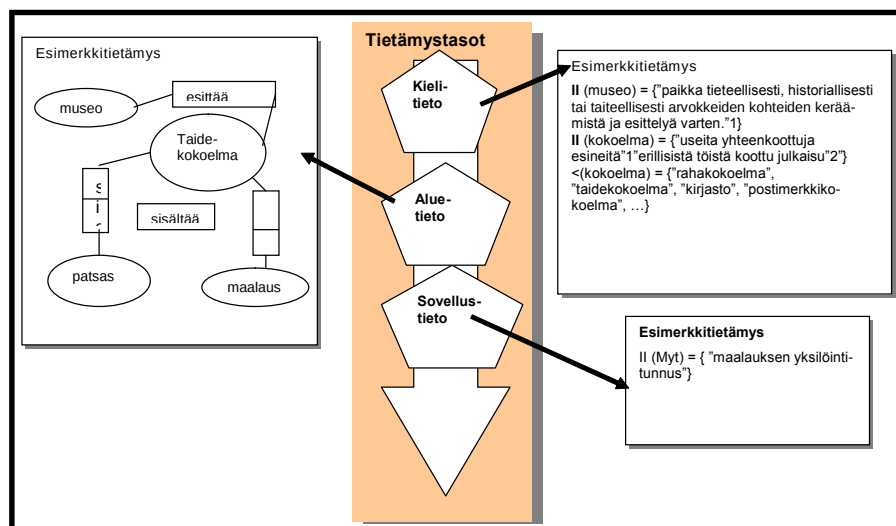
Lexicon on valmis joukko kielien perusmerkitysyksiköitä, siis kieleen erikoistunut ontologia, kuten englanti, esperanto, bosnia.

Meersmanin toteaa, että erityisesti thesaurusten osalta on vaikea vetää rajaa alue- ja sovelluskohtaisen ontologian väliin, varsinkin kun ontologia usein kattaa useita sovelluksia [Mee99].

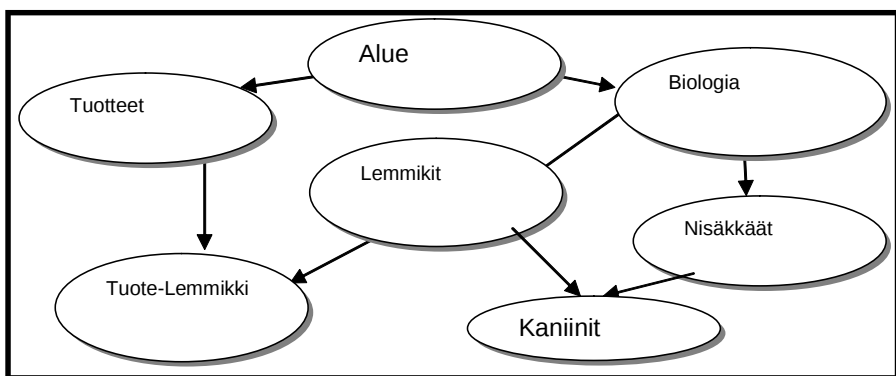
Esimerkkinä aluekohtaisesta thesauruksesta voisivat olla kuvien 1 ja 2 tapainen Reutersin uutistoimiston käyttämän luokittelun pohjalta tehdyt uutisten luokittelutaulut. Ne liittyvät toisiinsa objekti-rooli-mallin (kts. kuva 3) mukaan esimerkiksi siten, että toimialatiedot voivat olla yritystietojen luokkana.

Meersman on kuvannut yleisten (lexicon, thesaurus) ontologioiden muodostamiseen ja merkitysten määritelmien liittyviä periaatteita ja esittänyt niiden pohjalta suorituksia ontologioiden tietosisällöille. Hänen näkemyksensä on, että nykyiset hakemistotyyppiset ontologiat eivät vielä ole rakenteeltaan tai tekniikoiltaan riittävän hyviä kuvaamaan esitettävän aineiston merkityksiä. Meersmanin mielestä olisi tärkeää pitää ontologia latteana ja huolehtia siitä, ettei se sisällä sellaista tietoa, jonka ymmärtäminen ja käsittely kuuluu agenttien vastuulle. Hän toteaa myös, että ontologiaan pitäisi sisällyttää kerrosrakenne, joka mahdollistaisi tietojen jaon yleisiin, alue- ja sovelluskohtaisiin osiin esimerkkipä 4 tapaan.

Monen sovelluksen yleisesti käyttämistä tiedoista pitäisi muodostaa aluekohtaisia thesauruksia. Kielitason perusmääritysten pitäisi olla lähes staattisia ja ehdottomasti alueriippumattomia. [Mee99]



Kuva 4: Tietämyksen tasot ontologiassa [Meermannia mukaellen, Mee99]



Kuva 5: Esimerkki kuvitteellisesta lemmikkikaupan tietohakemisto-ontologia luokittelusta.

Ontologiat ja semanttinen web

Tietokoneiden ja tietojärjestelmien yhdistämisen haasteena on siis saada eri osapuolet ymmärtämään toisiaan. Web-sivut eivät nykyisellään sisällä juurikaan tietoa siitä, mitä merkityksiä niihin tai niiden sisältämiin tietoihin sisältyy. Esimerkiksi thesaurus ja lexicon tyyppiset tietohakemisto-ontologiat (kuva 5) voivat auttaa sivujen merkitysten yhteismitallisessa luokittelussa ja määrittelyssä sekä niihin sisältyvien palveluiden löytämisessä ja käyttämisessä.

Koska nykyiset web sivustot ovat syntyneet hyvin hajautetusti, ei ole ehkä realistista uskoa, että yhdistämisen ratkaisuna voisi ainakaan nopeasti olla yksi tai edes muutama yle-

sontologia. Lisäksi halu luokitella omat verkkosivut tulee vasta hyötyjen realisoitumisen kautta. Avuksi sivujen luokittelulle voitaisiin Hendlerin mukaan tehdä apuneuvoja niiden tuottamisvälineisiin. [Hen01]

Hendler mainitsee pari esimerkkiä siitä, miten sivujen merkitykset voitaisiin tuottaa helposti normaalin sivuntekemisen ohessa. Ensinnäkin kuvakirjastoista haettaviin kuviin voitaisiin liittää valmiita luokittelutekijöitä, jolloin kuva liimattaisiin omalle sivulle ja samalla sen sijainti aiheuttaisi kyseisen sivun luokittelun. Ongelmana on tietysti kuvien merkitysten oikeellisuuden varmistamisesta.

Toinen tapa tuottaa sivujen luokittelutekijät voisi olla ontologioita hyö-

dyntävä sivujen tuottamisen apuväline. Koska välineen toimintaa voisi ohjata mikä (tai mitkä) ontologiat tahansa, olisi termihakemisto mahdollista luoda lukuisilla erilaisilla tavoilla. Helppokäyttöisellä välineellä voisi luoda ihmis- ja harrastushakemistojen avulla kotisivuja, ammatti- tai yrityshakemistojen avulla ammattisivuja sekä toimintaontologioiden avulla agenttien toimintaan perustuvia sivuja. Apuväline tekisi automaattisesti sivuille monipuolisetkin merkinnät ja helpottaisi siten niiden löytymistä jatkossa monenlaisilla hakutekijöillä. [Hen01]

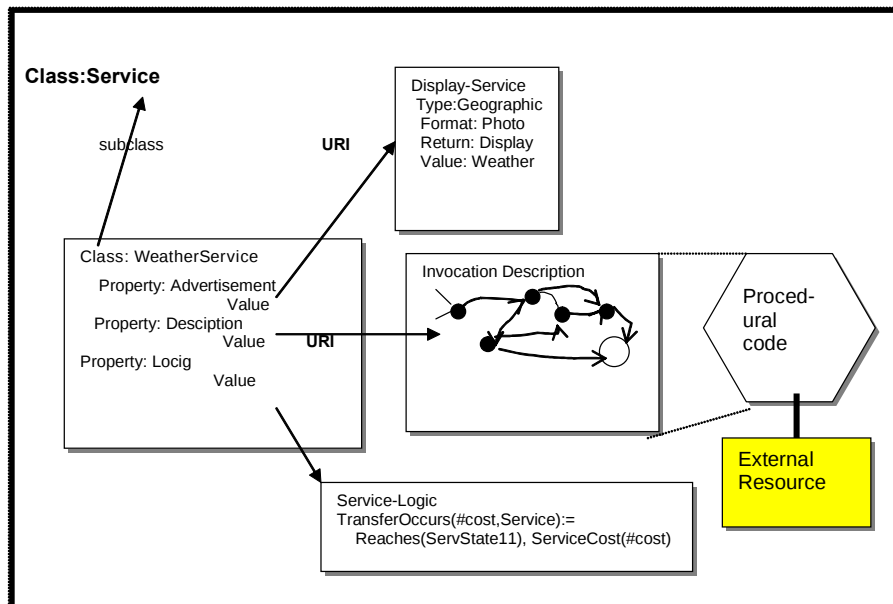
Ontologiat, verkkopalvelut ja agentit

Ontologoiden ensisijainen merkitys verkkopalveluiden rakentamisessa ja koneiden yhteenliittämisessä rakentamisvaiheessa on tiedostettu aiemminkin. Koneiden on ymmärrettävä toisensa käyttämiä termejä, tyypejä ja rakenteita ja yhteenliittämiseen tarvittavat asiat on tehty usien verkon tai palvelun hallintatyökaluilla. Semanttisen webin myötä ontologiat muuttuvat koneella luettaviksi ja palveluagentit voivat hyödyntää niitä palveluiden hakemisessa ja tuottamisessa.

Palveluhakemistoja on jo syntynyt parantamaan tilannetta (esim. UDDI) ja niitä voi pitää jollain tapaa palvelumainoksina. Sopivilla hakusanoilla palveluagentit voivat löytää niiden kautta sopivan palvelun tarjoajan.

Yhdistämällä pointereita, sivujen merkintöjä ja ontologisia kieliä päästään pelkkien mainosten sijasta paljon pidemmälle. Myös palveluiden kuvauksia (siis mitä se tekee) voidaan muuttaa koneella luettavaan muotoon samoin kuin kuvauksia siitä, kuinka palvelua käytetään. [Hen01]

Kuvassa 6 on esitetty graafinen kuvaus tällaisen palveluluokan toteutusperiaatteista. Jokaisella palveluluo-



Kuva 6: Semanttisen Webin valmis palveluluokka [Hen01].

kalla on kolme ominaisuutta: osoitin palvelun mainokseen, osoitin palvelun kuvaukseen ja kuvaus palvelun toteutuslogiikasta.

Nykyisin ostosten tekeminen palvelutarjoajien (esim. verkkokirjakauppa) sivuilla edellyttää yhden tai useamman lomakkeen täyttämistä ja mahdollisten virheiden korjaamista kunnes lomake lopulta on valmis. Yleisin tapa toteuttaa tällaisia sivuja on käyttää Common Gateway Interfacea (CGI). Toimintojen ohjaamista varten on kirjoitettu proseduraalista ohjelmakoodia, joka linkittää lomakesivut ulkoisiksi resursseiksi. Sivulla vierailva agentti ei kuitenkaan pysty koodista helposti selvittämään, millaisiin tietoihin on syytä varautua tai miten palvelu toimii.

Hendler tarjoaa ratkaisuksi deklaratiivista kehystä. Alkuun voisi päästä toki helpommallakin. Kuvassa 6 on esimerkiksi käytetty rajallista tilakoneetta, FSM (Finite State Machine) proseduraalisen koodin herättäjänä. Tällaisen ontologian (ei palvelun) määrittely ei Hendlerin mielestä olisi kovinkaan vaikeaa. Hänen tilakone-ontologiansa voisi sisältää esimerkiksi luokat *Tila* ja *Linkki* aliluokkineen *AloitaTila* ja *LopetaTila*. Rakenteis-

sa ja ominaisuuksissa kuvattaisiin linkit *paa (head)* ja *hanta (tail)* kertomaan miten tiloista pääsee ulos ja nimeämään ne - URI:lla ja tai muulla identifioivalla tavalla. Tämä voisi muodostaa perusontologian, jota palvelun tuottajat voisivat laajentaa ja jolla voisi olla joissakin ontologioissa kuvata tiettyjä termejä yleiseen käyttöön. Tällaisen ontologian pohjalta voisivat sivuille tulevat agentit analysoida kuvauksen ja pystyisivät päättämään palvelun herättämiseen tarvittavat tiedot sekä sen voisivatko ne käyttää tätä palvelua. Koska kielen määrittelyt voisi kytkeä cgi-skripteihin, olisi agenttien mahdollista käynnistää toiminnot käyttäjien puolesta. [Hen01]

Määrittelykielet mahdollistavat siis rakenteisten ontologioiden määrittelyn lisäksi ontologioiden laajentamisen koskemaan myös käsittelysääntöjen määrittelyä kielillä. Tällaisia sääntöjä voisi sisällyttää palveluluokkiin kuvassa 6 esitellyllä tavalla: pointeri osoittaa URIa, josta löytyy tarvittava palvelulogiikka. Esimerkiksi jonkun maksullisen palvelun tuottamisen yhteydessä voisi löytyä sääntö,

```
TransferOccurs(#cost,Service):
=Reached(ServState11),
ServiceCost (#cost)
```

joka kertoo, että tietyn palvelun suorittamiseen liittyy maksun hoitaminen. Tätä tietoa ei tilakone itse tarvitse, mutta siitä on hyötyä erilaisille kaupallisille tapahtumille. [Hen01]

Ontologiatoteutuksia

Ontologoita ja ontologiakieliä ollaan kehittämässä lukuisissa projekteissa ympäri maailmaa. Tähän kapaleeseen on poimittu jotakin yksittäisiä esimerkkejä tarjolla olevista.

Thesaurusityypisiä ontologiatoteutuksia ovat esimerkiksi tuotehakemisto KIF ja yleistietohakemisto Cyc®. WordNet® on esimerkki lexican tyyppisestä sanasto-ontologiasta ja UDDI on yrityksille tarkoitettu palveluhakemisto. TREVI on Reutersin uutispalvelimen hyödyntämä lexiconontologia. Ontologia-kuvauskieliä on myös kehitetty useissa tutkimusryhmissä, SHOE ja DAML+OIL ovat esimerkkejä sellaisista ontologioista. Suomalaista ontologiatutkimusta edustavat Genie sekä tuoreimmat Helsingin yliopiston johtamat Fenix-tutkimushankkeet.

KIF

The Knowledge Interface Format on syntynyt Stanfordin yliopistossa ja se tunnetaan myös nimillä Ontolingua tai KSL. Se toteuttaa ennenkaikkea vähentämisen filosofiaa, mutta ylpeilee hyvin määritellyllä muodollisella semantiikalla. Stanfordin yliopiston Center for Information Technology (CIT) sekä Knowledge Systems Laboratory (KSL) osallistuvat Defense Advanced Research Projects Agency (DARPA) -sponsoroi- tuun CommerceNET-projektiin kokeilemalla ja tutkimalla, mitä lisäarvoa web-pohjaisesta, asiakkaiden hakuehtojen perusteellakin toimivasta tuotekatalogista olisi. Asiakkaat voivat tuotteiden etsimisen lisäksi myös vertailla eri tuotteita. KSL tarjoaa myös editorin tuotetietojen syöttöön. Hyötyjä uskotaan syntyvän tuotteiden sisältöä ja

alla on MS Access Repository for SHOE. [SHO03]

DAML +OIL

Ontologia-kuvauskielien yhdistämiseen ja standardointiin pyrittiin esimerkiksi US Defense Advanced Research Projects Agency (DARPA) tuella DARPA Agent Markup Language (DAML) kehittämistyössä (lue myös KIF). Tämänkin työ on saanut jo jatkoa, sillä DAML +OIL (The Ontology Inference Layer) -kielen kehittäjien tavoitteena on yhdistää ja kehittää standardeja myös Resource Description Frameworkin (RDF, OWL) pohjalta mm. Semantic Web and Agents Projectissa. OIL on web-pohjainen ontologioiden esitystapa ja päättelykerros, jossa käytetään monissa kehyskielissä toteutettuja rakenteita sekä päättelyä ja muodollisia merkityksiä logiikan kuvauksissa. Kuvauksiin sisältyvä ontologisen kielen taso sanastoineen mahdollistaa jatkossa agenttipohjaisten palveluiden tuottamisen. OWL Web Ontology Language on luotu kieleksi, jolla voidaan kuvata webissä oleviin sivuihin ja sovelluksiin liittyviä luokkia ja niiden välisiä suhteita. OWL mahdollistaa alueen muodollisen kuvaamisen määrittelemällä sen luokat ja niiden ominaisuudet. Sillä voidaan kuvata yksilöitä ja liittää niihin ominaisuuksia ja sen mahdollistaa luokkien ja yksilöiden avulla päättelyn muodollisten määrittelyjen rajoissa. Määrittelyt on järjestetty siten, että ne alkavat perusasioilla ja niihin liittyy tarvittavat monimutkaisemmat määrittelykomponentit [Hen01, DAM00, RDF03].

Genie

Yksi kotimainen ontologiaesimerkki löytyy VTT Elektroniikka ja Oulun Yliopiston Älykkäiden järjestelmien tutkimusryhmän agenttipohjaista järjestelmäarkkitehtuuria kehittäneestä hankkeesta Verkon henki, Genie of the NET. Yksi osa arkkitehtuuria oli informaatiopalvelumoduuli, koko

järjestelmän tietokanta. Informaatiopalvelu hakee muiden agenttien tarvitsemat tiedot tietokannasta. Toisaalta se tallettaa muiden agenttien välittämät tiedot tietokantaan. Informaatiopalvelumoduuli muodostuu FIPA ACL-XML -parserista, tietokanta-agentista ja hajautetusta tietokannasta. Tietokantana käytetään IBM DB2-tietokantaa, jossa on XML-laajennin. Kaikki tietokannan tieto on siis talletettu XML-muodossa. Tietokannan tietämys- ja sääntökantamoduuli sisältää tietämyksen järjestelmästä ja kaikki järjestelmässä tarvittavat säännöt. Ontologiatietokantaan on talletettu järjestelmän ontologia, jota agentit käyttävät kommunikoinnin tukena. Ontologiatutkimus projektissa on enemmänkin neuroverkkopohjaista dynaamista käsitteiden klusterointia ja niiden välisiä relaatioita. [Gen02]

Fenix- tutkimusprojekteja

Suomalaiselle semanttiselle webille luodaan perustaa kahdessakin Helsingin yliopiston tietojenkäsittelytieteen laitoksen projektissa. Tutkimusprojektit saavat rahoitusta Tekesin Fenix –Vuorovaikutteinen tietotekniikka –teknologiaohjelmasta. Hankkeissa on mukana monia yrityksiä, julkishallintoa ja tutkimuslaitoksia. Hankkeita johtaa professori Eero Hyvönen Helsingin yliopistosta.

Luettelopalvelut

Yhdessä Helsingin yliopiston tietojenkäsittelytieteen laitoksen hankkeessa kehitetään verkkopalvelua, jossa yhdistetään keltaisten sivujen tyyppiseen hakemistoon yritysmaailman ja julkishallinnon web-palveluita. Hankkeessa rakennetaan älykäästä verkkopalvelua uusilla semanttisen webin tekniikoilla. Ideana on etsiä asiakkaan tarpeisiin liittyviä asioita tai palveluita näiden merkityksen ja keskinäisten suhteiden avulla. Älykäs palvelu osaa esimerkiksi päätellä, että Espoo on osa pääkaupunkiseutua tai ymmärtää, että ”PC” ja ”personal computer” viittaa-

vat merkitykseltään samaan tietokoneiden alatyyppiin.

Hankkeessa uskotaan, että nykyisten luettelopalveluiden businessmallia voidaan vahvistaa älykkäillä ja dynaamisilla verkkopalveluilla. Elektronisen kaupan, palveluiden ja julkisen hallinnon verkkopalveluiden kehittämistä voidaan vahvistaa tekemällä luettelopalvelut monikäyttöisemmiksi loppukäyttäjille.

Semanttinen web – Finnish Ontologies and Thesauri

Toisessa tutkimushankkeessa on tavoitteena kehittää suomalaisen semanttisen webin tarvitsemia kansallista ontologiajärjestelmää. Hankkeessa rakennetaan luokittelumalleja, joiden avulla voidaan yhdistää toisiinsa eri tietojärjestelmien tietoja. Liikkeelle lähdetään museoiden kokoelmatie- doista, otollisesta paikasta, sillä ovat- han museoiden asiasanalistat tuhan- sien sanojen mittaisia. Toisaalta näis- sä listoissa törmätään hyvin nopeasti ongelmaan, joka semanttisen webin luokittelumalleilla on tarkoitus ratkais- ta: jokaisella museolla on omia tapo- jaan luokitella asioita, eivätkä nämä mallit toimi yhteistyössä keskenään. Toisaalta on olemassa ”kuolleita” kä- sitteitä, jotka eivät mahdu nykyisten luokittelujen alle. Esimerkiksi Tsekko- slovakia ei kuulu enää valtioiden luok- kaan, vaikka paljon tietoa on arkistoi- tu kyseisen valtion alle.

Tavoitteena on, että semanttisen webin standardien avulla voidaan yhdistää eri tahojen asiasanalistoja toisiinsa sekä se, että semanttisen webin ontologiat täydentyvät eri käyttä- jin toimesta. Kaikki hankkeen ontolo- giat ja ohjelmistot ovat eri tahojen va- paasti käytettävissä.

Ontologioiden tulevaisuudesta

Nykyinen hajautettu ja hetero- geeninen tietojärjestelmäympäristö sekä nopeasti kasvava kiinnostus kom-

ponenttipohjaiseen ohjelmistokehitykseen ovat osaltaan lisänneet kiinnostusta ontologioita kohtaan.

Yleisesti ollaan sitä mieltä, että ontologiat voivat parantaa kommunikointia, tietämyksen jakamista ja uudelleenkäyttöä. Ontologioiden avulla voidaan määritellä käsitteitä tarkasti ja tämä tekee niistä merkittäviä kommunikoinnin helpottajia. Useiden erillisten (heterogeenisten ja hajautettujen) tietojärjestelmien yhdistäminen on lisännyt kiinnostusta ja tarvetta kehittää tällekin tukea. Yleiset tietohakemistomallit (esim. KIF, UDDI) auttavat tiedon jakamista ja uudelleenkäyttöä. Verkkopalveluiden sivustojen luokittelua on vaikea kuvitella toteutettavaksi ilman sovittua aluekohtaista ontologiaa. Palveluagentit tulevat ilman muuta tarvitsemaan ontologiaan pohjautuvia hakemistoja, jotta palvelut on mahdollista löytää. Yhtä paljon palveluiden dynaaminen kehittäminen edellyttää palveluihin liittyvien työntekijöiden tallettamista ontologiaan ajoaikaista tulkintaa varten.

Ontologioiden kehittäminen ei kuitenkaan vaikuta helpolta eikä missään nimessä kertaluonteiselta tehtävältä. Kun kuvataan muuttuvaa ja kehittyvää maailmaa ja sen ilmiötä, on kuvaamistapojen muuttuminen ja standardien kehittyminen nähtävä osana luonnollista kehitystä. Ontologiat tulevat ja kehittyvät - jatkossakin.

Lähteet:

Cyc03

Cyc® Ontology Guide: <http://www.cyc.com/cyc-2-1/intro-public.html> (26.3.2003)

DAM00

DAML ontology language and OIL. (2000). Status: initial listing of some related OIL documents in progress WWW-sivusto, <http://www.daml.org/> <http://www.daml.org/2000/10/daml-oil> (20.3.2003)

Gen02

VTT Elektroniikka, Oulun yliopisto ja Pehr Brahe Laboratorio. Verkon henki-projektin sivut. <http://www.pbol.org/projects/genie/> (20.4.2003).

G&M00

Geerts, G., McCarthy, W. E. (2000). "The Ontological Foundation of REA Enterprise Information Systems," (Guido Geerts and William E. McCarthy) working paper, Michigan State University (August 2000 and being revised for journal submission) (msword). <http://www.msu.edu/user/mccarth4/Alabama.doc> (19.3.2003)

Gru93

Gruber, T. R. (1993). A Translation Approach to Portable Ontologies. J. on Knowledge Acquisition, Vol 5(2), 199–220.

Hen01

Hendler, J. (2001). *Agents and the Semantic Web*. Department of Computer Science, University of Maryland, College Park, MD 20742. Paper has appeared in the IEEE Intelligent Systems Journal (march/April 2001). <http://www.cs.umd.edu/users/hendler/AgentWeb.html> (16.3.2003)

KIF03

Knowledge Interface Format, (KIF) tunnetaan myös nimillä Ontolingua tai KSL. <http://www-ksl-svc.stanford.edu:5915/doc/ontology-server-projects.html> (25.3.2003)

Mee99

Meersman, R. A. (1999). *Semantic Ontology Tools in IS Design*. Systems Technology and Applications Research Laboratory, Star Lab Technical Report. WWW-sivusto, <http://www.starlab.vub.ac.be/staff/robert/default.htm#Papers> (16.3.2003)

RDF03

Resource Description Framework (2003). The RDF specifications pro-

vide a lightweight ontology system to support the exchange of knowledge on the Web. WWW-sivusto: <http://www.w3.org/rdf> (20.4.2004)

Sow97

Sowa, J. F. (1997). Principles of ontology. <http://www-ksl.stanford.edu/onto-std/mailarchive/0136.html> (20.4.2003)

SHO03

University of Maryland, SHOE-projekti. <http://www.cs.umd.edu/projects/plus/SHOE/> (20.4.2003)

Tekes

Tekes/ Teknologia-ohjelmat WWW-sivusto: <http://akseli.tekes.fi/Resource.phx/tivi/vaui/museot.htx>. <http://akseli.tekes.fi/Resource.phx/tivi/vaui/luettelo.htx> (8.8.2003)

Oas03

Oasis-konsortion sivusto e-kaupan standardien kehittämiseen. <http://www.oasis-open.org/home/index.php>

OasC1

Oasis-konsortion The Web Services Flow Language (WSFL) –sivusto. <http://xml.coverpages.org/wsfl.html>

Web02

Weber, R. (2002). *CONCEPTUAL MODELLING AND ONTOLOGY: POSSIBILITIES AND PITFALLS*. Paper in the 21st International Conference on Conceptual Modeling, ER 2002, October 7-11, 2002, Tampere, Finland. <http://er2002.cs.uta.fi> (20.3.2003)

Wor03

Princetonin Yliopiston Kognitiotieteen laboratorio. WordNet® -sivut. <http://www.cogsci.princeton.edu/~wn/> (13.4.2003)

Helena Venäläinen,
FD Finanssidata Oy,
Helena.Venalainen@osuuspankki.fi

CWM standardoi metadatan

*Mikko Kutvonen,
SAS Institute Oy*

Yritysten it-arkkitehtuurissa on paljon eriaisteisia komponentteja. On tietokantoja, palvelimia, työasemia – ja metatietoa. Metatieto on liima, joka yhdistää kontrollin tietovirran hallintaan tai liiketoiminnan päätökset osaksi datan esitystapaa. Metatieto siis määrittelee sen, mitä raakadata todella liiketoiminnassa merkitsee ja mihin sitä voi käyttää.

Metatieto osana yrityksen arkkitehtuuria ja liiketoimintaa

Metatieto - mitä se todellisuudessa on? Vuosien saatossa metatiedolla on saatettu tarkoittaa tietohakemiston sisältöä, relaatiokannan rakenteita tai ETL (Extract-Transform-Load) -välineen prosessikuvausta. Yhteistä näille on ollut eräänlaisen ”tietoa tiedosta” -kielikuvan käyttö. David Marco totesi kirjassaan tämän metatiedon perinteisen määritelmän olevan liian suppea ja jopa harhaanjohtava. Hän esittikin metatiedon määrittelyksi jotakin tämän tyyppistä: ”Metadataa on kaikki fyysinen data ja tietämystä sisältävä informaatio organisaatiosta ja sen ulkopuolelta koskien organisaation käyttämää dataa. Tähän kuuluu mukaan itse data, liiketoiminnalliset ja tekniset prosessit, säännöt ja eheystarkistukset, rakenteet yms.” Tämä määrittely onkin lähempänä sitä kuvaa, joka eri organisaatioilla metadatasta tänä päivänä on. Nykyaikaiseen, laajempaan metatiedon määrittelyyn kuuluu luontevasti mukaan sekä tekninen metatieto (tekniset prosessit, rakenteet jne.) että liiketoimintaa palveleva metatieto. Liiketoiminnan prosessit, laadulliset tarkastelusäännöt ja esimerkiksi käytetyt laskentakaavat tarjoavat yrityksille paremman kuvan siitä, miten omaa liiketoimintaa johdetaan ja miten se on määritelty.

Metadata ja Balanced Scorecard

Käytännön metatiedon hallinnan kannalta on luontevaa kytkeä metatieto tietovarastointiin. Onhan tietovarastoinnin eräs keskeisimmistä perusteluista mahdollisuus tuottaa operatiivisesta datasta uutta kilpailuvoimaa ja tehoa. Esimerkkinä voi käyttää vaikka Balanced Scorecard –sovellusta, jonka mittaristojen latauksen automatisointi on esimerkki teknisestä metatiedosta – toisaalta mittarien laskusäännöt ja kommentit puolestaan kuvaavat liiketoiminnan tarvitseman tiedon. Yhteiseen metatietovarastoon tallennetaan siis kaikki tekniset prosessit tietovaraston luomiseksi samoin kuin liiketoiminnan säännöstö, kuvaukset ja erilaiset koodistot. Metatieto on näiden yhdistävien ominaisuuksien ansiosta aina keskeisessä osassa kun erilaisia tietovarastoinnin prosesseja hiotaan kuntoon. Toisaalta metatiedon kunnollinen hallinta tuo myös kustannussäästöjä kun asiat on ker-
taalleen määritelty kunnolla.

CWM Standardi

Vuoden 2000 lopussa OMG Group sai aikaiseksi erään metatiedon ja tietovarastoinnin tärkeimmistä standardeista eli Common Warehouse Metamodel (CWM) -määrittelyn. CWM muodostettiin olemassaoloevien standardien päälle: OMG:n Unified Modeling Language (UML), Extensible Markup Language (XML) ja OMG:n omalle XML Metadata Interchange (XMI):lle. CWM mahdollistaa yrityksille liiketoiminnan kannalta kriittisen tiedon keräämisen metamalliin, joka on universaali, skaalautuva ja kattaa kaikki tietovarastoinnin alueet – sekä liiketoiminnan että teknisen puolen.

CWM-välineistö käytännössä

CWM -pohjaiset sovellukset ja välineet ovat nyt tuoneet metatiedon yhä useamman yrityksen saataville. Vice president Doug Laney Meta Groupista on todennut CWM:n saatavuudesta näin: ”Vaikka Common Warehouse Metamodel ei ole vielä läheskään saavuttanut kriittistä massaa implementointien määrässä ja sen kattama alue on ”pienimmän yhteisen tekijän” rajoittama, se on silti standardi, jota data warehouse -toimittajat eivät voi ohittaa. CWM on ainoa toivo yrityksille, jotka haluavat vaihtaa metadataa avoimesti eri työkalujen välillä – täten saavuttaen säästöjä ylläpito-
kuluissa ja lyhentyneissä toteutusajoissa.” Samantyyppinen suuntaus näkyy analyysiyritysten lisäksi käytännön työelämässä. Yhä useampi tarjouspyyntö tai hanke sisältää vaatimuksen standardirajapinnoista (=CWM) myös metatiedon hallinnalle.

Tietohallinnonkin kannalta metatiedon käsittely on kriittistä. Ilman metatiedon avulla tehtävää vaikuttavuusanalyysiä ei muutoshallinta ole mahdollista. Esimerkiksi metatieto eri prosesseista auttaa selvittämään minkä tyyppisiä ongelmia aiheuttaa laajasti päätöksenteon tukijärjestelmissä datana käytetyn operatiivisen tietolähteen muutokset. Käytännössä tilausjärjestelmässä tapahtuva tietokentän pituuden kasvaminen kuudesta merkistä kahteenkymmeneen merkkiin vaikuttaa tietovarastossa kaikkiin niihin tietovirtoihin, tauluihin ja omiin koodeihin, joissa kyseisen järjestelmän vanhaa kenttätietoa on käytetty.

Metatieto on siis nykyaikaiselle liiketoiminnalle elinehto – hallitaan sitä järkevästi ja avoimesti.

*Mikko Kutvonen,
Sales Support Manager,
Technology,
SAS Institute Oy*

Metatietojen välitys eri valmistajien ohjelmistojen välillä: Tarua vai totta?

Mika Naatula,
TietoEnator

Metatietojen hallinta tietovarastoinnin yhteydessä on aina ajankohdainen asia. Metatietojen tärkeydestä on puhuttu aivan tietovarastoinnin alkuajoista lähtien, mutta millainen onkaan tilanne nykyään. Miten toimii metatietojen välitys eri ohjelmistojen välillä?

Tietovarastojärjestelmän metatietoja

Voidaksemme käsitellä metatietojen välitystä ohjelmistojen välillä tulee meidän ensin hieman tarkastella sitä, mitä tietovarastojärjestelmään liittyviä metatietoja on olemassa. Metatiedot luokitellaan yleisesti kahteen pääluokkaan, varsinaisiin sisältökuvauksiin (business metadata) ja teknisiin metatietoihin (technical metadata). Tekniset metatiedot voidaan edelleen jakaa aktiivisiin ja passiivisiin metatietoihin sen mukaan onko tiedoilla suora vaikutus tietovarastojärjestelmään liittyvän ohjelmiston toimintaan (aktiivinen metatieto) vai onko tieto luonteeltaan vain informatiiviseen käyttöön tarkoitettu (passiivinen metatieto).

Tekniset metatiedot syntyvät ja niitä ylläpidetään tyypillisesti jonkin järjestelmään liittyvän ohjelmiston avulla. Kullakin ohjelmistolla on metatiedoille omat tallennusmuotonsa ja paikkansa. Ohjelmistojen metatietojen tallennuspaikoista käytetään englantinkielistä nimitystä ”meta repository”, jolle ehkä yleisimmin käytetty suomenos on metahakemisto. Teknisten metatietojen ohella ohjelmistojen metahakemistot kykenevät tallettamaan enemmän tai vähemmän myös sisältökuvauksia.

Tarve metatietojen välittämiseen

Metatietojen välittämisen tarve on selkeä etenkin sisältökuvauksen osalta. On toivottavaa, että kullakin työvälineellä pääsisi suoraan sen omilla toiminnoilla tarkastelemaan ja päivittämään ohjelmiston käsittelemiin tietorakenteisiin liittyviä sisältökuvauksia, eikä missään nimessä ole järkevää että samat tiedot jouduttaisiin syöttämään uudelleen jokaiseen välineeseen. Kunkin nyt vaikkapa haluaisi kirjoittaa uudelleen tietovaraston hallinnassa käytettävään ETL (Extract Transform Load) välineeseen samoja taulu- ja kenttäkuvauksia, jotka on jo kerran syötetty tietokannan suunnitteluvälineeseen. Sen sijaan olisi erittäin hyödyllistä, jos ETL kehittäjä näkisi kyseiset sisältökuvaukset suoraan käyttämänsä välineen näytöllä.

Suuri osa teknisistä metatiedoista ovat luonteeltaan sellaisia, että niillä ei juuri ole käyttöä muualla kuin sen ohjelmiston sisällä joka ne on tuottanutkin. Otetaanpa esimerkiksi vaikkapa raportointivälineen omana metatietonaan tallettama yksittäisen raportoin ulkoasuun vaikuttava sarakeleveyden määrittäminen. Se on luonteeltaan kyseiseen raporttiin liittyvää aktiivista metatietoa, mutta tiedolle tuskin löytyisi järkevää käyttöä vaikkapa ETL välineen puolella.

Myös teknisten metatietojen osalta on kuitenkin olemassa selkeä tarve tietojen välittämiseen, mutta asia on jo osin hoidossa, kiitos siitä relaatiotietokannoille, joiden kautta erilaiset tietorakenteiden tekniset määrittäykset siirtyvät jouhevasti välineestä toiseen. Harvapa enää nykyisin kirjoittelee kenttänimiä, tietotyypppejä ja pituuksia käsin tietokannan CREATE TABLE lauseisiin, jos kantasuunnittelijan käytössä on ollut jokin suunnitteluun tarkoitettu valmisohjelmisto. Vastaavasti raportointiohjelmistot osaavat jo nykyään yleensä lukea tekniset tietorakenteiden kuvaukset suoraan kantapalvelimelta, työstä käsinsyöttöä ei tarvita. Valitettavasti tietorakenteisiin mahdollisesti liittyvät sisältökuvaukset eivät yleensä kulje mukana samaa reittiä vaikka niitä olisi tietoihin jo josain vaiheessa liitettykin, niiden välittämiseen tarvitaan erikoisratkaisuja.

Lähteestä kohteeseen kuvaukset

Vasta aivan viime aikoina on alettu tiedostaa myös toinen teknisten metatietojen välityksen avulla täytettävissä oleva tarve, joka niin ikään vaatii uudenlaisia ratkaisuja. On alettu vaatia ETL-ohjelmistojen metatietoihin tallettamien lähteestä kohteeseen kuvausten tehokkaampaa hyödyntämistä niin, että kuvaukset toimisivat kattavasti myös eri työkalujen välillä. Käyttäjä voisi esimerkiksi haluta tietää mistä kaikista perusjärjestelmien tauluista raportilla esiintyvä lukuarvotieto on peräisin ja mitä erilaisia käsittelyjä sille on tehty. Vastava tarve on olemassa myös toisinpäin;

järjestelmän ylläpitäjälle on hyödyksi tietää mihin kaikkiin raportteihin jokin tietty perusjärjestelmän luku tietovarastojärjestelmän kautta kulkeutuu.

Kuvatun kaltaisen kyselyn tekemiseen ei enää riitäkään yksinkertainen teknisten metatietojen siirto relaatiokantapalvelimen välityksellä, kuten vaikkapa taulun sarakemääritysten ”import” ODBC:n läpi. Edellytyksenä on teknisten metatietojen välitys niin, että eri ohjelmistojen sisällä sijaitsevien tietorakenteiden vastaavuuksista jää metatietoa talteen. Yhteen ja samaan paikkaan tulee tallettaa paitsi tieto eri ohjelmistojen sisältämien tietorakenteiden vastaavuuksista, myös tieto koko muunnosketjusta. Esimerkkinä taulumäärittäminen olisi ODBC:n sijaan vietävä erillisen metatietojen hallintasovelluksen kautta raportointisovellukseen, jolloin hallintasovellus saa talteen metatiedon siitä mihin raporttisovelluksen rakenteeseen kyseinen taulumäärittäminen menee. Ei siis riitä että raporttisovellus osaisi itse lukea taulumäärittäksen suoraan kantapalvelimelta, silloinhan vain raporttisovellus tietäisi mistä taulusta tieto haetaan, mutta sillä taas ei ole aavistustakaan siitä mistä tieto tuohon tauluun oli tullut, sen tietää ETL ohjelmisto. Tai oikeammin, tieto löytyy ETL välineen hallitsemista metatiedoista. Edellytetään siis kehittyneempää teknisten metatietojen välitystä ja samassa yhteydessä olisi hyvä ratkaista myös sisältökuvausten sujuva välittäminen.

Erilaisia lähestymistapoja metahallintaan

Perinteinen lähestymistapa metatietojen hallintaan on pyrkiä rakentamaan yksi yhteinen metahakemisto tai metatietokanta, joka pystyisi sisältämään kaiken tietovarastojärjestelmän sisältämän metatiedon. Kaikenkattavan metatietokannan tietomallinnus on haastavaa monessakin mielessä ja vaihtoehtona räätälöidylle raken-

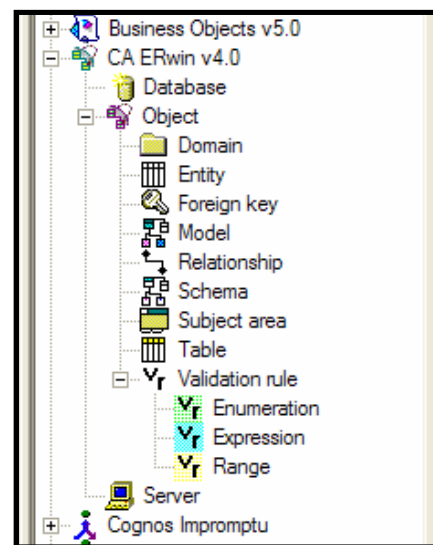
tamiselle löytyy valmiita enemmän tai vähemmän valmiita tietomalleja, joissa yleisin ongelma on monimutkaisuudesta johtuva vaikeaselkoisuus ja sitä kautta seuraava väistämättä seuraava käyttöönoton hankaluus. Ehkä tunnetuin valmiista metatietomalleista löytyy Object Management Groupin (OMG) julkaisemasta Common Warehouse Metamodel (CWM) standardista, joka tuntuu olevan hankalasti implementoitava jopa välinetoimittajille, asiakaskohtaisesta järjestelmäkehityksestä puhumattakaan.

Metatietomallin suunnittelun (tai ostamisen) jälkeen seuraava vaihe on toteuttaa liittymät metadatan siirtämiseksi eri sovellusten ja metatietokannan välillä. Perinteinen keskitetyn metakannan periaate on yleensä pyrkimys siihen, että kaikki metatietoja tuottavat ja käyttävät ohjelmistot pysyisivät itse aktiivisesti jonkin liittymäpinnan kautta tallettamaan ja lukemaan tarvitsemansa metatiedot keskitetystä metakannasta. Yleensä siinä riittääkin jo sitten haastetta jos liittymiä ollaan rakentamassa räätälöidysti. Järjestelmän käyttöönotossa ei paljoa lämmitä, jos työväline tarjoaa liittymärajapinnan, joka on vain ohjelmointirajapinta – sen oikeaan hyödyntämiseen metatietojen siirrossa on vielä pitkä matka.

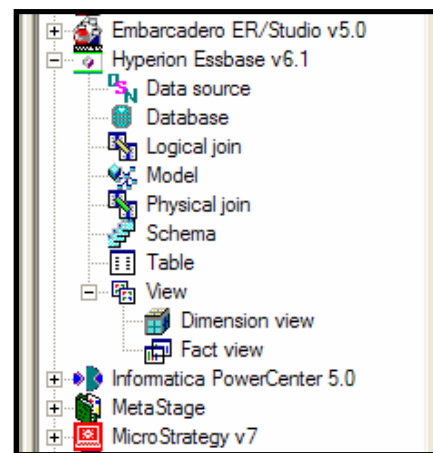
Astetta kehittyneempi tai ainakin helpommin sovellettavissa oleva lähestymistapa on erillisen metatietojen välitysohjelmiston (metabroker) käyttö. Nämä ohjelmistot sisältävät valmiit liittymät yleisimpiin kantasuunnitteluohjelmiin, ETL välineisiin sekä raportointivälineisiin ja osaavat tehdä metatietojen siirrot tukemiensa ohjelmistojen välillä. Välitysohjelmistossa on sisäänrakennettuna tieto siitä miten eri valmistajien ja erityyppiset metatiedot vastaavat toisiaan. Sen tiedon perusteella välitysohjelmisto osaa tehdä siirrot, toimien siis aktiivisena osapuolena. Ohjelmisto lukee metatietoa, tallettaa sen omaan sisäiseen ra-

kenteeseensa ja taas kirjoittaa metatiedon ulos jonkin toisen ohjelmiston muotoon hyödynnettäväksi. Tyypillisesti metatietojen välitysohjelmistot sisältävät myös monipuolista toiminnallisuutta metatietojen analysointiin, kuten esim. vaikutusanalyysi – mihin tämä tieto menee ja lähdeanalyysi – mistä tämä tieto tulee tai on tullut.

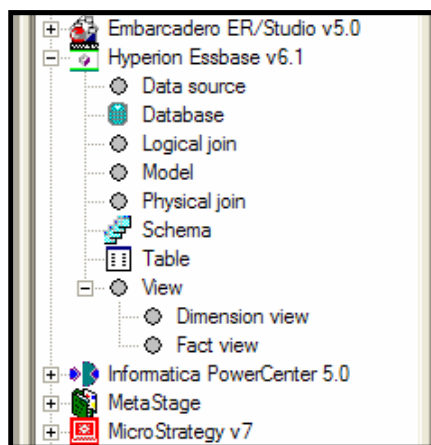
Seuraavassa on esimerkkinä esitetty erään metatietojen välitysohjelman (Ascential MetaStage) hallittomia metatietoja, kuvat 1-3.



KKKKKKKKKKKKKKKKKKKKKKKK metatietoluokat MetaStagen näytöllä.



KKKKKKKKKKKKKKKKKKKKKKKK jelmiston metatietoluokat MetaStagen näytöllä.



Kuva 3. Hyperion Essbase ja CA Erwin metatietojen vastaavuudet MetaStagen näytöllä katsottuna Essbasen kannalta. Kuvakkeet jotka eivät ole harmaina ympyröinä sisältävät vastaavuuksia.

Vaihtoehtona keskitetyille metatietojen hallinnalle on toteuttaa suorat liittymät käytettyjen ohjelmistojen välille. Tämä, ei niin kehittynyt menetelmä, tulee kysymykseen lähinnä joidenkin tiettyjen teknisten metatietojen ja myös yksinkertaisten tietorakenteisiin liittyvien sisältökuvausten siirtämisessä. Kannattaa muistaa että yksittäisten liittymien rakentamiseen tehtävä panostus ei ole kovin yleiskäyttöistä, kun tarve metatietojen laajempaan jakeluun tulevaisuudessa kasvaa, kasvavat myös liittymien rakentamisen ja hallinnan työmäärät.

Ennen kuin aletaan ohjelmoimalla toteuttaa liittymiä, kannattaa aina tutustua millaisia metatietojen välitykseen sopivia liittymiä on saatavana valmiina ko. ohjelmistoihin joko niiden valmistajilta tai kolmansilta osapuolilta.

Metatietojen välityksen suunnittelu

Ensimmäisenä askeleena metatietojen välityksen suunnittelussa on tunnistaa ne ohjelmistot, jotka voivat tuottaa metatietoja toisten ohjelmistojen hyödynnettäväksi ja taas toisaalta

voivat hyödyntää muiden ohjelmistojen tuottamaa metatietoa.

Ollaanpa sitten hankkimassa valmiita liittymiä yksittäisten sovellusten välille tai ostettaessa kokonaisvaltaisempaa ohjelmistoa metatietojen välittämiseen hallintaan, on tarkkaan tutustuttava mitä metatietoja liittymät oikeasti siirtävät ja riittääkö se tarpeeseen. Monesti luvattu tuki saattaakin olla olemassa vain sitä tarkoitusta varten, että voidaan tuotteen esitteessä luvata yhteensopivuus toisen ohjelman kanssa. Niin hienolta kuin metatietojen integraatio saattaakin kuulostaa, se ei missään tapauksessa saa olla itse tarkoitus. Metatietojen välttämisestä on oltava hyötyä!

Metahallintaa suunniteltaessa tärkeä ja muistettava asia on sisältökuvausten hallinnan suunnittelu. Ne olisi hyvä saada siirtymään ohjelmistojen välillä samoilla liittymillä kuin tekninen metatieto. Sisältökuvausten päivittämiseen ja hakujen tekemiseen tarvitaan helppokäyttöinen käyttöliittymä, muuten käyttäjät vieraantuvat tiedoista, niiden ajan tasalla pitämisestä ei koeta tärkeäksi ja koko ajatus sisällöllisistä metatiedoista alkaa romuttua. On tarkkaan arvioitava riittääkö käytettävien valmisohjelmistojen sisältökuvauksien tallettamiseen tarkoitettu toiminnallisuus vai tarvitaanko loppukäyttäjille erillinen selainpohjainen käyttöliittymä metatietojen hakuun ja päivityksiin.

Metatietojen tulevaisuus tietovarastojärjestelmissä

Tulevaisuuden ennustaminen on aina hankalaa, mutta tietovarastojen metatietojen hallinnassa on selvästi nähtävissä muutamia kehityssuuntia. Ensinnäkin tietovarastojärjestelmän hallinnataan käytettävät ETL ohjelmistot ovat yleistymässä voimakkaasti ja niiden käyttöönnoton myötä myös järjestelmän metatietojen hallinta ottaa ison harppauksen eteenpäin.

XML tiedostomuodon käyttö tietojen välityksessä on sekin tehnyt paljon hyvää metahallinnalle ja suuntauksen voi hyvin olettaa jatkuvan. On selkeästi eduksi metahallinnalle, että samaan tiedostomuotoon voidaan varsinaisen tiedon lisäksi liittää standarditavalla myös metatietoa.

Valmisohjelmistojen metaominaisuudet kehittyvät voimakkaasti. Esimerkiksi tietovarastojärjestelmään käytettävien OLAP ja raportointiohjelmistojen metatieto-ominaisuuksia kehitetään kaiken aikaa. Toivomushan olisi, että käyttäjä saisi kaiken tiedon ja siihen liittyvän metatiedon samaa käyttöliittymää käyttäen, unohtamatta myöskään metatietoon kohdistuvia hakutoimintoja. Etenkin sisältökuvausten tehokkaassa hyödyntämisessä on vielä tekemistä monenkin ohjelmiston osalla. Johtavat ohjelmistovalmistajat tiedostavat tämän ja panostavat asiaan.

Erillisistä metatietojen välitys ja hallintaohjelmistoista voidaan todeta, että ne yleistyvät ja halpenevat. Vielä muutama vuosi sitten vaihtoehdot tällä alueella olivat todella vähissä ja nekin harvat erittäin kalliita. Välineet jäivät yleensä hankkimatta, koska hankinnan tekemiseen olisi vaadittu erityistä pioneerihenkeä ja hyvää mielikuvitusta kalliin hankinnan perusteeseen.

Metatiedon hallinnan ja välityksen standardointi on edelleen kysymysmerkki. Lupaavimmalta tällä hetkellä vaikuttaa usean valmistajan ainakin nimellisesti tukema CWM, jota käsiteltiin jo aiemmin tässä artikkelissa. Yrityksenä metatietojen hallinnan standardointi ei ole mikään uusi juttu, historia on nähnyt useita ehdotelmia erilaisilta eri valmistajien yhteenliittymiltä, mutta toistaiseksi yksikään standardiehdotelma ei ole lyönyt itseään kunnolla läpi markkinoilla. Myös CWM:n tulevaisuuden suhteen ollaan

yleisesti edelleen odottavalla kannalla, vaikka muutamat valmistajat (mm. Oracle) kovasti vannovatkin sen nimiin.

Yhteenveto

Metatietojen hallinta tietovarastojärjestelmässä on tärkeä osa-alue, joka vaatii panostusta järjestelmää rakennettaessa. Pienelläkin panostuksella metahallintaan voidaan saavuttaa suuria hyötyjä järjestelmän ylläpidettävyyden ja käytettävyyden parantuaessa.

ETL ohjelmisto antaa hyvän pohjan metahallintaan, mutta se ei yleensä sellaisenaan riitä liiketoiminnan tarvitsemien sisältökuvausten tallettamiseen, mikä puolestaan on erittäin oleellinen osa metatietoa. Todennäköisesti tarvitaan jotain lisäohjelmis-

toa tai räätälöityä ohjelmistokehitystä metahallinnan täydentämiseksi.

Kannattaa myös harkita valmisohjelmistojen käyttöä metahallintaan ja tietojen välitykseen eri sovellusten välillä, metahallinnan valmisohjelmistot ovat kehittyneet ja halventuneet oleellisesti.

Aiheseen liittyviä lähteitä:

Marco, David (2000). Building and Managing the Meta Data Repository. John Wiley & Sons Inc., New York.

The Data Warehousing Institute
luentosarjat:

Mimno, Peter. Selecting the Right ETL and Cleansing Tools.

Marco, David. Fundamentals of Meta Data Management.

Marco, David.. Building a Custom Meta Model.

Reeves, Laura. Integrating Data Warehouses and Data Marts.

www-sivustot:

Data Advantage Group. <http://www.dataadvantagegroup.com/>

Ascential Software. <http://www.ascentialsoftware.com/>

Allen Systems Group. <http://www.asg.com/>

*Mika Naatula,
TietoEnator*

Informaation mallintaminen

*Esko Marjomaa,
Joensuun yliopisto,
Tietojenkäsittelytieteen laitos*

1 Johdanto

Informaatiojärjestelmien suunnittelussa informaation mallintamisella tarkoitetaan yleensä toimintaa, jolla pyritään mahdollisimman luotettavaan esitykseen asiantuntijoiden omaamasta, sovellusaluetta koskevasta tietämyksestä. Tässä artikkelissa lähdetään liikkeelle varsin yleisestä, kattavasta näkökulmasta informaation mallintamiseen, joten esitettävät tarkastelut ja niiden pohjalta tehtävät johtopäätökset koskevat myös ja erityisesti systeemiteoreettisten keinotekoisien järjestelmien suunnittelun periaatteita ja lähtökohtia.

Alalla käytetty terminologia eroaa kulttuurista ja tieteenalakohtaisesta näkökulmasta toiseen, joten muutama rajaava täsmennys heti tähän alkuun: (1) Informaation mallintamisen (Information Modelling) lopputuloksena on usein kohdealuetta koskevan tietämyskokonaisuuden esitys (Knowledge Representation). Sanalla 'tietämys' viitataan inhimilliseen 'tiedon' käsitteeseen, jota on kahdenlaista: implisiittinen (hiljainen, 'tacit') ja eksplisiittinen (väitellään esitetty) tieto. Puhuessamme 'informaatiosta', meidän on erotettava 'tulkittu' informaatio (knowledge by description) 'eitulkitusta' informaatiosta, jota on kahdenlaista - fyysisessä materiaalisessa oleva 'muoto' (in forma) ja 'data', s.o., numeerinen informaatio.

1.1 Mitä on informaatio?

Etymologisesti sana "informaatio" juontuu latinan kielestä (*in forma*), mikä antaa aiheen arvella, että informaatio ei ole ainetta, vaan ennemminkin sen 'muotoa'. Nykyään arkikielessä puhutaan jo 'informaation

muodosta' (tai 'tiedon esitystavoista') ja 'informaatioisällöstä', jolla tarkoitetaan yleensä viestien sisältämää 'tietämystä'.

Tieteen kielessä sanalla "informaatio" voidaan tarkoittaa mitä tahansa seuraavista:

- fyysinen informaatio, s.o., fyysisten järjestelmien negatiivinen entropia (suuri järjestyksen aste),
- yntaktinen informaatio, s.o., tietynlaisten signaalien relationaalinen ominaisuus,
- semanttinen informaatio, s.o., signaalijonon ilmaiseman viestin sisältö,
- pragmaattinen (psykologinen, intersubjektiiivinen, mutta korkeintaan semanttinen) informaatio.

Kysymyksessä "Mitkä tekijät on otettava huomioon mallinnettaessa mitä tahansa erityistä informaatiojärjestelmää?" on kaksi keskeistä käsitettä: 'informaatiojärjestelmä' ja 'mallintaminen'.

1.2 Millainen on informaatiojärjestelmä?

"Informaatiojärjestelmä", hyvin yleisellä tasolla, merkitsee mitä tahansa sellaista objektia (tai "entiteettiä"), jota voidaan tarkastella informaation lähteenä, välittäjänä, vastaanottajana tai näiden yhdistelmänä. Sellaisia ovat esimerkiksi massatiedotusvälineet, asiantuntijajärjestelmät, aivot, ihmis-mielet, solut, elektronit jne.

Suppeammassa merkityksessä "informaatiojärjestelmällä" kuitenkin tarkoitetaan sellaista entiteettiä, jonka luonteenomaisimmat piirteet voidaan ryhmitellä seuraavasti:

- a. teknologiset ominaisuudet, jotka koskevat järjestelmän vakautta ja luotettavuutta tarkasteltuna hardware-näkökulmasta,

- b. funktionaaliset ja semanttiset ominaisuudet, jotka koskevat kysymyksiä, tekeekö järjestelmä oikeita asioita oikealla tavalla ja voiko se sopeutua muuttuviin vaatimuksiin,
- c. ekonomiset ominaisuudet, jotka painottavat koko järjestelmän kustannustehokkuutta ja kontribuutiota yrityksen keskeisiin toimintoihin,
- d. sosiaaliset ominaisuudet, jotka yhdistyvät järjestelmän niihin piirteisiin, joilla on vaikutusta yrityksen työympäristöön tai yhteiskunnan joihinkin osa-alueisiin.

1.3 Mitä on mallintaminen?

Sanaa "malli" on pyritty määrittelemään mm. seuraavasti (FRISCO 1998):

Malli on päämäärähakuisesti abstrahoitu, selkeä, täsmällinen ja yksiselitteinen käsitys.

Käsitys on sellaisen toiminnan tulos (nk. 'aktantti'), jossa ihmistoimija tähtää havainnon tulkitsemiseen mielessään, mahdollisesti jossain spesifissä toimintakontekstissa.

"Metamalli" määritellään seuraavasti:

Metamalli on kielen käsitteellisen perustan malli, joka sisältää joukon peruskäsitteitä ja joukon sääntöjä, jotka määräävät sellaisten mahdollisten mallien joukon, joita voidaan denotoida tuossa kielessä.

"Mallin denotaatio" määritellään seuraavasti:

Mallin denotaatio on täsmällinen ja yksiselitteinen mallin representatio jossain sopivassa formaalissa tai puoli-formaalissa kielessä.

"Mallintaminen" määritellään seuraavasti:

Mallintamisprosessi, ks. mallintamistoiminta.

Mallintamistoiminta on sellainen jono, jossa on jollain alueella ('domain') suoritettu havaitsemistoiminta, ko. havaintoa seuraava kuvittelutoiminta, jonka tuloksena on malli, jota seuraa tuon mallin representointitoiminta, joka tuottaa tulokseksi mallin denotaation.

Me kuitenkin lähestymme asiaa yksinkertaisemmasta päästä: mallintamisen perusideasta. Mallintamiseksi yleisesti ottaen voidaan kutsua mitä tahansa toimintaa, joka perustuu seuraaviin (välttämättömiin) osatekijöihin:

- 1) *modellum* (s.o. mallinnettava entiteetti, objekti, kvantifikaatioavaus jne.);
- 2) *modeller* (s.o. mallintava subjekti);
- 3) *modellens* (s.o. millainen tahanan rakennettu malli, lause, representaatio jne.);
- 4) sekä (4) näiden väliset *vuorovaikutukset*.

Tätä mallintamisen perustilannetta valaisee seuraava kuvio (Kuvio 1), jossa nuolet voidaan tulkita dynaamiseksi vuorovaikutuksiksi.

Termien "*modellum*", "*modeller*" ja "*modellens*" käytön motivoi niiden "konnotaatiovapaus". Esimerkiksi sanalla "malli" on luonnollisessa ja tieteellisessä kielenkäytössä useita eri käyttötapoja:

- 1) ideaaliset mallit, joita on kahdenlaisia:
 - a) matemaattiset mallit, joilla tarkoitetaan yksinkertaistettuja

ja idealisoituja teorioita, jotka koskevat joitain määrättyjä todellisuuden osa-alueita, ja
b) 'karikatyyrit' – kuten pilakuvat, joissa jonkin ihmisen tiettyjä silmiinpistäviä piirteitä on liioiteltu;

- 2) analogia-mallit, jotka esittävät valittua todellisuuden osa-alueita analogiasuhteen avulla - näitä malleja on kahdenlaisia
 - a) fysikaaliset miniatyyrimallit (pienoismallit) sekä
 - b) kaaviot, joita esitetään - esimerkiksi - graafisilla merkeillä tai lingvistisillä, kirjoitetuilla termeillä;
- 3) malliteorian mallit - ts. joukko-opilliset struktuurit, missä annettun formaalisen kielen lauseet on tulkittu;
- 4) esimerkkimallit - kuten idolit, käyttäytymismallit, mallisotilas, valokuvamalli, alastonmalli yms.; ja
- 5) näkemykselliset ("konseptionaaliset") mallit, joilla tarkoitetaan käsityksiä, "mentaalisia representaatioita", "mentaalimalleja", mielikuvia tms., jotka vastaavat mielen ulkopuolisia kiinnostuksen kohteita.

Mallintaminen alkaa yleensä siitä kun mielenkiinnon kohteen havaitsemisen jälkeen kohdetta pyritään kuvailemaan jotenkin. Bertrand Russell esitti aikoinaan tiedon kaksi lajia: Knowledge by Acquaintance ('tieto tuttuuden perusteella') ja Knowledge by Description ('tieto kuvauksen perusteella'). Jos emme entuudestaan tiedä (tai 'tunne') kohdetta, me pyrimme löytämään siitä sellaisia ominai-

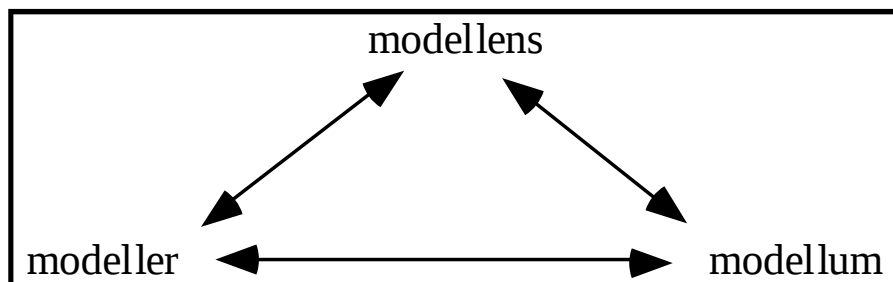
suuksia tai siihen liittyviä relaatioita, jotka ovat meille tuttuja muista yhteyksistä. Olennaista informaatiojärjestelmien rakentamisessa on pyrkiä ilmaisemaan kaikki mielenkiinnon kohdetta koskeva informaatio *eksplisiitisti*, s.o., selkeästi ja sellaisessa muodossa, että muutkin sen voivat ymmärtää (yleensä siis rautalangasta vääntäen) – ja kyseessä on tällöin 'tieto kuvauksen perusteella'.

Tämä ongelma näyttää olevan hyvin yleinen kognitiotieteissä, missä on osoittautunut vaikeaksi muodostaa kognitiivisen mallintamisen konsistentia teoriaa. Edes mitään yleisesti hyväksyttyä viitekehystä ei ole ilmaantunut. *Kognitiivisella mallintamisella* tarkoitamme toisaalta informaation hankkimisprosessia ja toisaalta tuon informaation esittämistä ('representointia'). Semioottisesta näkökulmasta informaation hankkimisen ja informaation representoinnin ongelmat voidaan nähdä olennaisesti samoina ongelmina kuin tiettyjen annettujen *merkkien tulkittamisen* ja *käytön* ongelmat.

"Semioottisella näkökulmalla" tässä tarkoitetaan lähestymistapaa, jolla on seuraavat kolme luonteenomaista piirrettä:

- 1) olemme kiinnostuneita merkkien formaalisista tarkasteluista,
- 2) "merkillä" tarkoitamme "jotain *x* mikä edustaa jollekulle *s* jossain suhteessa tai joltain ominaisuudeltaan jotain muuta kuin itse *x*:ää" ja
- 3) merkinä oleminen on prosessi (*semiosis*). It addresses somebody, that is, creates in the mind of that person an equivalent sign, or perhaps a more developed sign."

Erityisen selvästi nämä ongelmat tulevat esiin *käsitteellisessä mallintamisessa*, jota on kehitetty 1970-luvun lopulta yhdistämään sellaisia alueita kuten tekoäly, ohjelmointikielet, tietokantasuunnittelu ja ihmisen ja tie-



Kuvio 1. Mallintamisen perustilanne.

tokoneen välinen vuorovaikutus. Yleisesti, laajimmassa, ei-teknisessä merkityksessään, käsitteellisellä mallintamisella tarkoitetaan mitä tahansa käsitteitä hyödyntävää representatiivista toimintaa. Tämän merkityksen voi katsoa heijastavan ilmaisun ”käsitteellinen mallintaminen” moniselitteisyyttä: Toisaalta käytämme käsitteitä rakentaessamme mallia jostain kiinnostuksemme kohteesta, mutta toisaalta käytämme merkkejä rakentaessamme jonkin käsitejärjestelmän mallia. Ilmaisun moniselitteisyyttä lisää tietysti sanan ”mallintaminen” monimerkityksisyys.

1.4 Käsitteellisen mallintamisen periaatteet

Kirjallisuudesta löytyy joitain käsitteellisen mallintamisen periaatteita, joita on hyvä pyrkiä noudattamaan rakennettaessa mitä tahansa informaatiojärjestelmää. Periaatteet ovat kuitenkin ’vain’ periaatteita, ja useimmissa tapauksissa niiden tarkka noudattaminen on hyvin vaikeata, jollei suorastaan mahdotonta.

- P1 Käsitteellistämisperiaate: Vain keskusteluavaruuden (UoD, Universe of Discourse) käsitteelliset aspektit on otettava huomioon konstruotaessa käsitekaaviota.
- P2 100% -periaate: Kaikki UoD:n relevantit aspektit on kuvattava käsitekaaviossa.
- P3 Formalisointiperiaate: Ollakseen implementoitava käsitekaavion (tai ehkä ’käsitekartan’, ’miellekartan’ tai ’ajatuskartan’) on oltava formalisoitavissa.
- P4 Semioottinen periaate: Käsitekaavion pitäisi olla helposti tulkittavissa ja ymmärrettävissä.
- P5 Tiedon esittämisen vastaavuusperiaate: Rakennettavan mallin (modellens) tulisi olla sellainen, että sen tunnistettavissa olevilla osilla on yksi-yhteen vastaavuus mallinnettavan kohteen (model-lum) relevantteihin osiin.

2 Uskomusten ja tosiasiaväitteiden mallintaminen

Useimmissa (käsitteellistä) mallintamista vaativissa tehtävissä meidän on jollain tavalla mallinnettava myös propositioita (tai lauseita tai uskomuksia), joiden esittämiseksi käytämme jotain formaalia kieltä – esim. UML:ää tai Sowan käsitegraafeja tai episteenmistä logiikkaa.

2.1 Hintikan Knowledge and Belief

Jaakko K. J. Hintikan teos ”Knowledge and Belief” oli ensimmäinen episteenmistä (s.o., tiedon ja luulon käsitteitä koskevaa) logiikkaa käsitellyt maailmanmainetta saanut kirja, jossa esitettiin ’episteenisiin modaaliteetteihin’ kohdistuva tarkastelu ja yhdistettiin ne nk. lauselogiikkaan ja predikaattilogiikkaan. Hintikan episteenmistä logiikan avulla voidaan esittää täsmällinen uskomisen ja tietämisen looginen järjestelmä.

Logiikka tutkii päteviä argumentteja tai päättelyjä. Logiikassa argumentti on jono lauseita, joissa premisit ovat jonon alussa ja johtopäätös on jonon viimeinen jäsen. Pätevä argumentti on sellainen, että on mahdotonta, että sen premissit ovat tosia ja johtopäätös olisi epätosi; toisin sanoen: jos premissit ovat tosia, niin on välttämätöntä, että johtopäätös on tosi.

Lauselogiikka perustuu Aristoteleen aikoinaan esittämille 40 erilaiselle argumenttimuodolle (nk. argumenttiskeemoille). Logiikka argumenttien tieteenä tutkii argumenttien pätevyyttä tutkimalla argumenttiskeemojen pätevyyttä. Argumenttiskeemat ovat hyvin abstrakteja, koska niistä on hävitetty kaikki sellaiset elementit, jotka eivät vaikuta argumenttien pätevyyteen.

Lauselogiikka tutkii sellaisia argumenttiskeemoja, joiden pätevyys riippuu sellaisten ilmaisujen merkityk-

sestä kuin ’ja’, ’tai’, ’ei’, ’jos...niin’, ’jos ja vain jos’. Kaikki muut ilmaisut, jotka vaikuttavat argumenttien pätevyyteen, on jätetty pois.

Predikaattilogiikka tutkii kaikkia argumentteja, joita lauselogiikkakin tutkii ynnä argumentteja, joiden pätevyys riippuu sellaisista ilmaisuista kuin ’kaikki’ ja ’jokin’.

Episteenminen logiikka tutkii kaikkia argumentteja, joita lauselogiikka ja predikaattilogiikka tutkivat ynnä argumentteja, joiden pätevyys riippuu sellaisista ilmaisuista kuin ’tietää, että’, ’uskoa, että’.

2.2 Russell - Wittgenstein - Carnap - Goodman

Formaalisten kielten käyttö maailman kuvaamisessa (tai ’mallintamisessa’) juontaa juurensa kauas tieteen historiaan, mutta erityisesti viime vuosisadalla harjoitetun logiikan käyttämisen ’aristotelisen tieteen ihanteen’ tavoittelussa tuotti vaikuttavia aksiomaattisten järjestelmien kuvauksia. Carnapin teoksen *Der logische Aufbau der Welt* 1928 (”Maailman looginen rakentuminen”) ideana oli rekonstruoida rationaalisesti logiikan ja positivistisen epistemologian avulla tietäminen prosessina. Kirjan päämääränä oli esittää ’käsitteiden konstruktioaalinen järjestelmä’.

Yhtenä Carnapin innoituksen lähteenä oli Russellin ja Whiteheadin massiivinen *Principia Mathematica*, jossa esitettiin koko (silloinen) matematiikka logiikan avulla.

Carnapin toisena tärkeänä innoituksen lähteenä oli Wittgensteinin *Tractatus Logico-Philosophicus*, jonka yhtenä keskeisenä ideana on käsitys kielestä todellisuuden kuvana. Toinen tärkeä vaikutus *Tractatuksella* oli siinä, että kirja on esitetty kauniisti hierarkkisella tavalla, joka muistuttaa kovasti monien tietokoneohjelmien esitysmuotoa.

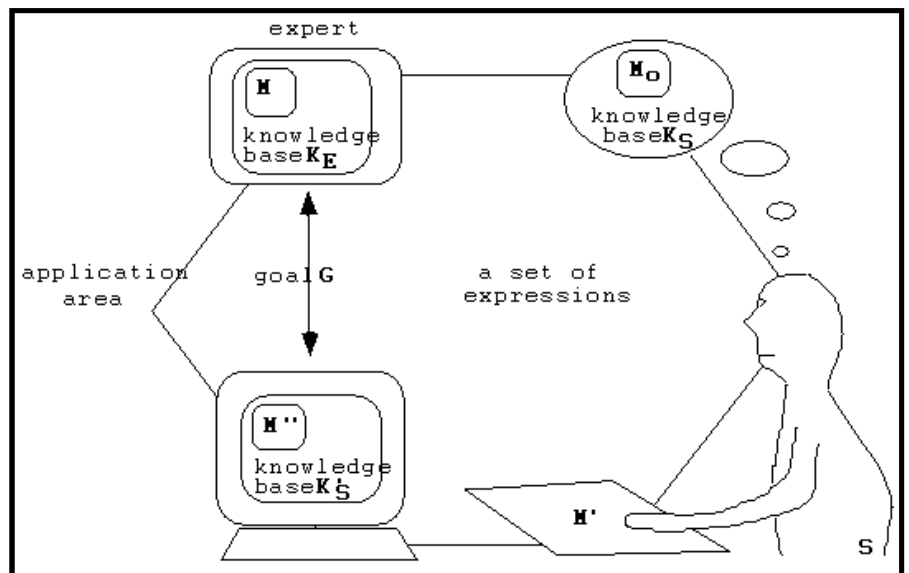
Carnapin työn vaikutukset näkyvät mm. Nelson Goodmanin teoksissa *The Structure of Appearance* ja *Ways of Worldmaking*, joista jälkimmäisessä esitetään jo suuntaviivoja useammanlaisille tavoille hahmottaa tai 'rekonstruoida' maailma käsitteellisesti.

3 Monimutkaisten kohteiden mallintaminen

3.1 Laajojen informaatiojärjestelmien

Lähes aina (paitsi ehkä "absoluuttisen äärettömyyden" kohdalla) mielenkiinnon kohteena olevaa informaatiojärjestelmää on syytä tarkastella vain osana jotain laajempaa informaatiojärjestelmää. Niinpä esimerkiksi jokin tietty tietokoneeseen implementoitu informaatiojärjestelmä on yleensä vain osa sellaista laajempaa informaatiojärjestelmää, johon kuuluu ko. implementoidun järjestelmän lisäksi muitakin informaatioyksiköitä, s.o., erilaisia informaation lähteitä, välittäjiä, vastaanottajia ja näiden yhdistelmiä, kuten esimerkiksi massatiedotusvälineet, kännykät, manuaaliset arkistot, ihmiset jne. Erilaisten informaatioyksiköiden ja niiden välisten suhteiden analyysissä voidaan hyvin noudattaa 'kartesiolaisen' tutkimusmenetelmän perusideaa: Paras käsitys mistä tahansa määrätystä mielenkiintomme kohteesta saadaan jakamalla alue osa-ongelmiin, jotka pyritään strukturoimaan mahdollisimman yksityiskohtaisesti. Tämän jälkeen osaongelmat ratkotaan ja meillä on mielenkiintomme kohteen valmis rakenteinen esitys.

Muodoltaan silmiinpistävän samanlaisia ovat Wittgensteinin *Tractatus* ja monien formaalien kielten, kuten ohjelmointikielten, esitystavat. On kuitenkin huomattava, että tutkimuksemme edistyessä joidenkin osaongelmien ratkaisut voivat vaikuttaa muiden osaongelmien muotoon tai sisältöön siten, että meidän on strukturoi-



Kuvio 2. Mallintaja mallintamassa sovellusalueetta

tava koko alue uudelleen, mikä merkitsee tietysti sitä, että kysymyksessä oleva mallintamisprosessi on *hermeneuttinen*.

3.2 Mallintamisen kokonaistilanteen hahmottaminen

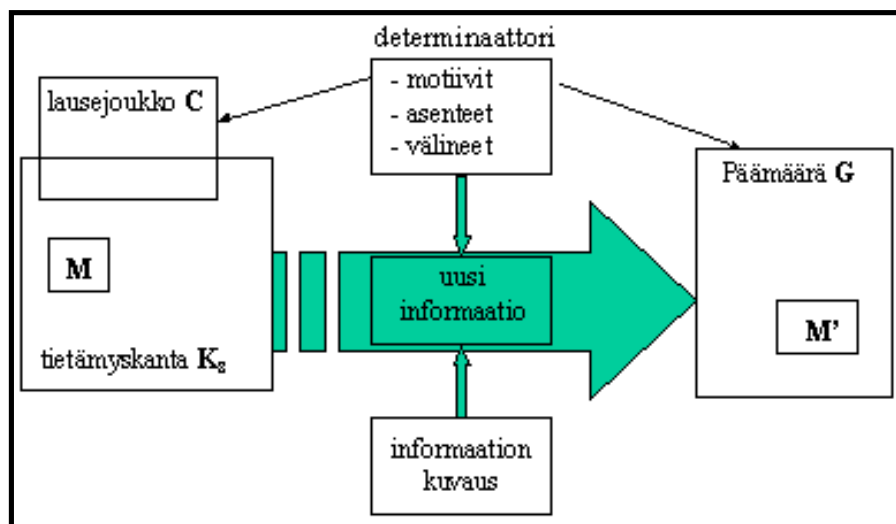
Monimutkaisen informaatiojärjestelmän mallintamisessa kokonaistilanteen hahmottaminen on tehtävä huolella, ettei mitään asiaan vaikuttavia tärkeitä tekijöitä jää huomioimatta.

Hermeneuttinen kehä: Jotta voi ymmärtää jonkin pikkuosan isosta kokonaisuudesta, täytyy ensin ymmärtää kokonaisuus, ja jotta voi ymmärtää kokonaisuuden, täytyy ensin ymmärtää kaikki pikkuosaset. Kun pyrimme ymmärtämään jotain uutta asiakokonaisuutta, meidän on aluksi luotava jonkinlainen alustava yleiskuva tuosta kokonaisuudesta ja sitten keskityttävä sen yksityiskohtiin, mikä jälkeen kokonaiskuvaa saattaa olla syytä tarkistaa ja tämän jälkeen saatamme huomata, että uudessa kokonaiskuvassa on sellaisia osia, joita ei ollut alustavassa yleiskuvassa, mutta jotka on pakko selvittää, jos tarkoituksena on muodostaa mahdollisimman tarkka kokonaiskuva ('ymmärrys')

ko. asiakokonaisuudesta, jne., jne. Hermeneuttista kehää kutsutaankin tämän vuoksi myös *hermeneuttiseksi spiraaliksi*. Kyseessä on usein päättymätön prosessi, jonka myötä kokonaiskuva koko ajan tarkentuu.

Kokonaistilannetta hahmottaessamme huomaamme usein, että kohdealueetta vastaavia malleja ilmenee mallintamisprosessin eri vaiheissa monia olennaisesti toisistaan poikkeavia. Itse mallintamisen varsinainen kohdekin on usein 'episteemiseltä asemaltaan' aivan eri kuin se, minkä voimme havaita. Kuviossa 2 tätä tilannetta vastaa erottelu sovellusalueeseen (application area) ja UoD:hen (a set of expressions). [Varsinaisia malleja voi sitten olla lukuisia – riippuen mallintajasta, esitystekniikasta, työkaluista, jne. – nämä nk. mallintamiseen vaikuttavat taustatekijät on esitetty kappaleessa 3.3.]

Jotta hermeneuttisessa kehässä päästään 'eteenpäin', on meillä tietysti oltava jonkinlainen näkemys siitä, miten uutta informaatiota etsitään. Tässäkin on syytä ensin tarkastella asiakokonaisuutta, jotta 'uuden informaation määrittelyongelman' eri aspektit tulisivat selkeämmin esiin.



Kuvio 3. Uuden informaation määrittelyongelma

100% -periaatteen mukaisesti meidän on pyrittävä ottamaan tarkasteltaviksemme kaikki mahdolliset *relevantit* seikat ja puhummekin nk. 'relevanssiongelma'.

Käsitteellisen mallintamisen relevanssiongelma voidaan jakaa kolmeen osaongelmaan,

- 1) informaation hankkimisen,
- 2) informaation valitsemisen ja
- 3) informaation esittämisen relevanssiongelmaan.

Informaation hankkimisvaiheeseen liittyy läheisesti uuden informaation määrittelyongelma: Miten voimme kuvailla tarvitsemaamme informaatiota, jota meillä ei vielä ole? Kuvio 3, joka on muokattu Kangassalon aiemmin esittämistä (1978: 107; 1979: 86). havainnollistaa tähän liittyvää problematiikkaa.

Kuviossa 3 lausejoukko C vastaa determinaattorin antamia, mielenkiinnon kohdetta koskevia ilmaisuja. Mallintajan S lähtökohtana on tietämuskanta K_s , joka sisältää vähintään osan lausejoukon C lauseista ja mielenkiinnon kohteen 'esiymmärryksen' tai 'mentaalimallin' M. Mallintajan S tarkoituksena on konstruoida sellainen informaatiokanta G joka sisältää sellaisen mallin M' , joka on muodostettu ottamalla huomioon mielenkiinnon kohteesta saatu uusi informaatio.

3.3 Mallintamiseen vaikuttavat taustatekijät

Uuden informaation muotoon ja sisältöön vaikuttavista tekijöistä merkittävimmät ovat 'determinaattori' ja '(uuden) informaation kuvaus', jotka ovat myös koko informaatiokannan G muotoon ja sisältöön vaikuttavien tekijöiden joukossa. Koska käsitteellistä mallintamista voidaan pitää olennaisesti samanlaisena toimintana kuin tieteellisen teorian muodostamista, voimme esittää seuraavat keskeiset tekijät (vrt. Niiniluoto 1980), jotka vaikuttavat käsitteelliseen mallintamiseen:

- (i) *mallintajan profiili*; mitä mallintaja osaa tehdä?
- (ii) *determinaattori*; esim. asiantuntijaprofiili, paradigmat, auktoriteetit,...
- (iii) *käytännön rajoitukset*; ympäristö, aika, raha, työvälineet,...
- (iv) *käyttäjien taidot*; esim. tietotekniset valmiudet
- (v) *mielivaltaiset sopimukset*; valittu terminologia, taustateoriat, merkitsemistavat jne.
- (vi) *empiiriset tosiasiat*; esim. jotain määrättyä UoD:tä koskevat yksittäiset rajoitukset
- (vii) *hypoteesit*; esim. yleistäminen
- (viii) *analysointitavat*; esim. luokittelu
- (ix) *abstrahointitavat*; esim. aggregointi, aksiomatisointi,...

- (x) *yksinkertaisuustarkastelut*; esim. ovatko käsitteerakenteet helposti visualisoitavissa?
- (xi) *hedelmällisyystarkastelut*; esim. onko menetelmä laajemmin sovellettavissa?
- (xii) *idealisaatiot*; esim. estimointi
- (xiii) *metafyysiset oletukset*; esim. ontologiset sitoumukset.

Neljällä ensimmäisellä on ilmeisen tärkeä rooli informaatiojärjestelmien suunnittelussa. Tekijät (v), (vii), (viii) ja (ix) koskevat pääasiassa käsittekaavion formaalisia аспекteja. Tekijällä (x) viittaamme tilanteeseen, missä meillä on kaksi tai useampia vaihtoehtoisia (ekvivalentteja) representaatioita ja joista valitsemme 'yksinkertaisimman'. Hedelmällisyystarkasteluilla - tekijä (xi) - tarkoitetaan esimerkiksi sitä tosiasiaa, että operationalisoidessamme kvantitatiivisia käsitteitä meidän on otettava huomioon niiden systemaattiset suhteet muihin tieteellisiin käsitteisiin, lakeihin ja teorioihin.

Myös idealisoinnin rooli on tärkeä (ks. kappaleet 3.4 ja 3.5). Idealisoinnilla tarkoitamme sitä, että joitain todellisuudessa sovellusalueeseen vaikuttavia tekijöitä ei oteta huomioon konstruoidessa generatiivista mallia (tai tieteellistä teoriaa).

Myös metafyysiset ennako-oletukset voivat vaikuttaa sekä rakennettavan mallin sisältöön että muotoon. Tieteellisen teorianmuodostuksen kannalta ei nimittäin ole lainkaan sama, sitoudummeko esimerkiksi olio-ontologiaan vai prosessi-ontologiaan. Ainakin esitysmuoto on aivan erilainen. Myös keskusteluavaruuden (UoD) erilaisten järjestämistapojen voi monissa tapauksissa katsoa olevan ennalta-asetetut. Tilanteeseen eniten vaikuttava tekijä on tietysti useimmiten 'determinaattori'.

3.4 Keskusteluavaruuden UoD erilaiset järjestämistavat

On olemassa useita erilaisia menetelmiä ja tekniikoita, joiden avulla UoD voidaan uudelleenorganisoida siten, että saadaan aikaiseksi mahdollisimman hyvä kohdealuetta kuvaava käsitteellinen malli. Useimmiten käytämme jotain seuraavista menetelmistä:

- UoD:n osien *luokittelu*; s.o., osien ryhmittely eri luokkiin niiden havaittujen yhteisten piirteiden pohjalta;
- *yleistäminen* on jonkin yleisen huomion esittämistä johonkin luokkaan kuuluvista yksittäisistä tyyppin esiintymistä;
- *aksiomatisointi* on sellainen UoD:n representoimisprosessi, jossa esitetään vain muutama lause ('aksioma' tai 'perusfakta'), joista voimme johtaa järjestelmän muut totuudet ('teoreemat') joidenkin johtosääntöjen avulla; tällaista representointijärjestelmää kutsutaan *aksiomajärjestelmäksi*;
- *aksiomajärjestelmän formalisointi* on askel, jossa lisäämme järjestelmään sellaisia sääntöjä, joita tarvitaan järjestelmän teoreemojen todistamisessa: *formalisoituja aksiomajärjestelmiä* voidaan pitää kohdealuetta koskevan tiivistetyn informaation ihannetapauksina;
- *idealisointi* on sellaisen representaation ('mallin') konstruointia, jossa joitain UoD:n vähemmän tärkeitä аспектеjä jätetään tarkoituksellisesti huomiotta;
- *abstrahointi* on jonkin entiteettien kokoelman joidenkin relevanttien osittaisten aspektien erottamista tuosta kokoelmasta.

3.5 Miellekarttojen käyttö käsitteellisten mallien pohjana

Yhdeksi hyväksi keinoksi lähteä liikkeelle käsitteellisessä mallintamisprosessissa on osoittautunut nk. miel-

lekarttojen ('mielikuvakartat', 'ajatuskartat', 'mindmapit') tekeminen mielenkiinnon kohteena olevasta asiasta. Lyhyt johdatus mindmap-tekniikkaan löytyy esimerkiksi osoitteesta www.cs.joensuu.fi/pages/marjomaa/oppiva/mindmap.htm:

- Liikkeelle lähdetään paperin keskeltä, johon kirjoitetaan käsiteltävän asian nimi, esim. "Tietoverkko".
- Tämän jälkeen piirretään tai kuvailaan aihe nimen ylä- tai alapuolelle tai ympärille.
- Sitten piirretään viivoja keskeltä ulos päin.
- Viivojen päälle kirjoitetaan avainsanoja tai lukuja tai piirretään kuvia tai symboleita. Avainsanoiksi valitaan mieluiten substantiiveja tai verbejä.
- Kaikilla viivoilla on kytkentä keskikuvioon tai johonkin toiseen viivaan.
- Lopputuloksena on yksi, vaikkakin subjektiivinen, malli kyseisestä asiasta.

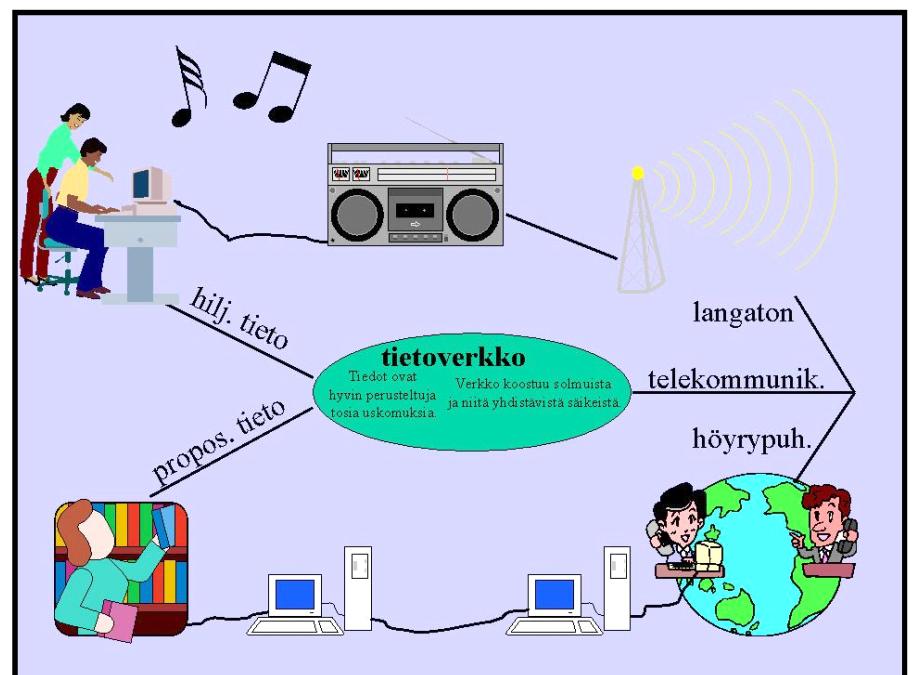
Esim. kts kuva alla.

Miellekarttoja voidaan mallintamisen myöhemmissä vaiheissa tämentää ja muokata siten, että niiden

pohjalta voidaan sitten rakentaa esimerkiksi käsitteekaavioita tai muita käsitteellisten mallien esityksiä. Hankaluutena on usein informaatiolähteiden löytäminen. Yksinkertaisinta on lähteä liikkeelle *hyvistä dokumenteista*.J

Esim. Oletetaan, että meidän pitäisi tehdä miellekartta aiheesta "Mitä on fysiikka?" Yksi tapa on lähteä liikkeelle kirjastosta! Haalitaan pöydälle pinkka fysiikan kirjoja. Kirjat käsittelevät melkoisen laajaa todellisuuden osa-aluetta ja ne ovat itsessäänkin melkoisen laajoja. Ilmeinen vaihtoehto edetä on, että meidän on valittava yksi kirja ja referoitava se. Referoidaan Kraussin kirja *Oleta pyöreä lehmä*. Referaatista (<http://cs.joensuu.fi/~marjomaa/krauss.htm>) etsitään keskeiset käsitteet, jotka sitten esitetään mahdollisimman tiiviissä muodossa. Vaikkapa tähän tapaan: <http://cs.joensuu.fi/pages/marjomaa/CM-kurssi/demot/Mallintaja.htm>

Kraussin kirjan jujuna on, että lähtien liikkeelle yksinkertaisista idealisoinneista saadaan lähes uskomattoman paljon asioita selville. Koetinkivenä läpi koko kirjan Krauss käyttäen nk. lehmä-kaskua:



Fyysikko, insinööri ja psykologi kutsuttiin konsulteiksi maitotilalle, jonka tuotanto oli selvästi alle normien. Ennen raportin antamista kukin tutki tilan toiminnan yksityiskohtia.

Insinööri sai työnsä valmiiksi ensimmäisenä. Hän totesi: "Lehmien pilttuiden kokoa tulee pienentää. Tehokkuus kasvaa, kun lehmät ovat pienemässä tilassa siten, että tilaa on lehmää kohti keskimäärin 9 kuutiometriä. Lypsykoneen putkien halkaisijaa tulee myös kasvattaa neljä prosenttia, jotta lypsyn aikana saadaan suurempi maidon virtauksen keskinopeus."

Psykologi jätti raporttinsa seuraavana. Hän ehdotti: "Navetta tulee maalata sisältä vihreäksi. Se on ruskeaa lempeämpi väri ja sen pitäisi johtaa suurempaan maidontuotantoon. Niityille pitää myös istuttaa nykyistä enemmän puita, jotta laiduntavat lehmät saisivat enemmän vaihtelua eivätkä olisi niin pitkästyneitä kuin nyt."

Lopuksi tuli fyysikon vuoro. Hän pyysi liitutaalua, jolle hän piirsi ympyrän. Hän aloitti esityksensä: "Oletetaan pyöreä lehmä..."

Vaikkei kasku kaikkien mielestä olekaan hauska, se osoittaa, ainakin vertauskuvallisesti, miten fyysikot pitävät kaikkea pallona tai jonakin sen likiarvona. Useimmat fysiikan teorioista ovat syntyneet yksinkertaisista malleista, jotka perustuvat vieläkin yksinkertaisempiin malleihin ja niin edelleen, koska ongelmat, jotka *osaamme* ratkaista, voidaan laskea yhden tai ehkä kahden käden sormilla. Pääosa fyysikoista noudattaa hol-

lywoodilaista sääntöä: Jos se toimii, hyödynnä se. Jos se toimii edelleen, kopioi se.

Krauss esittää myös fysiikassa tärkeän periaatteen, jota voi käyttää muillakin käsitteellisen mallintamisen alueilla:

P6 Abstraktioperiaate: Poista epäolennaiset yksityiskohdat abstraktioilla ennen kuin ryhdyt tositoimiin.

Lauseessa on kaksi tärkeää käsitettä: *abstraktio* ja *epäolennainen*. [Tietysti myös 'tositoimi' on aika keskeinen.] Kraussin esittämä periaate on olennaisesti sama kuin idea tieteessä tehtävistä "osittaisista idealisoinneista ja konkretisoinneista": Abstrahointi tuottaa idealisoidun mallin poistamalla 'epäolennaiset' seikat, ja tiettyyn kontekstiin kiinnittyminen saadaan aikaan lisäämällä idealisoitua malliin semantiikkaa, s.o., rajauksia ja rajoitteita.

Epäolennaisien yksityiskohtien poistamiseksi meidän olisi tietysti hyvä olla selvillä siitä, miten erottaa *relevantit* ('tärkeät') *irrelevantteista* ('vähemmän tärkeistä') seikoista. Asiaan on toki kiinnitetty huomiota (mm. Marjomaa 1997;), mutta tässä yhteydessä voidaan todeta, ettei mikään ole niin tärkeää kuin mökin kunnossapitäminen – paitsi ehkä purjehdus.

Kirjallisuusviitteet

FRISCO - A FRAMEWORK OF INFORMATION SYSTEM CONCEPTS. The FRISCO Report (Web

edition) Edited by Falkenberg, E. D., Hesse, W., Lindgreen, P., Nilsson, B. E., Han Oei, J. L., , C., Stamper, R. K., Van Assche, F. J. M., Verrijn-Stuart, A. A., and Voss, K. ã IFIP 1998. Full text of the FRISCO Report (MS-Word 6.0) (as available 2002-06-24 by anonymous ftp://ftp.leidenuniv.nl/pub/rul/fri-full.zip).

Kangassalo, H., 1978: *Informaatio-tarpeen määrittämisestä ja tyydyttämisestä. (On Defining and Saturating the Need of New Information.* A mimeographed Licentiate Thesis, in Finnish.) Tampereen yliopisto. Matemaattisten tieteiden laitos. Tampere.

Kangassalo, H., 1979: "Logical and Semantical Properties of Environmental Situations for Stating Information Requirements". *Formal Models and Practical Tools for Information Systems Design*. Edited by Schneider, H.-J., pp. 67-91. North-Holland Publishing Company, Amsterdam.

Marjomaa, E. 1997: *Aspects of Relevance in Information Modelling: Methodological Principles and Conceptual Problems*. Fitty 63, Tampere.

Niiniluoto, I. 1980: *Johdatus tieteenfilosofiaan (An Introduction to the Philosophy of Science, in Finnish.)* Otava, Keuruu.

Esko Marjomaa,
Joensuun yliopisto,
Tietojenkäsittelytieteen laitos,
marjomaa@cs.joensuu.fi

Käsitteet, liiketoimintaprosessit ja mallintaminen - outojen otusten kerho

*Claus Günther,
ChangeWare Group*

Kun puhutaan käsitteistä, tarkoitetaan sillä erilaisia asioita - kuten usein homonymi-käsitteiden kohdalla. Tässä Sytyke-lehdessä käsitellään erityisesti systeemityöhön liittyviä käsitteitä, käsite voi olla oman liiketoimintaympäristön termi tai tietovirtoihin ja tietokantarakenteisiin liittyvä käsitetyyppi eli entiteetti (josta IT-slangi käyttää epätarkasti myös sanaa käsite). Liiketoimintaprosessi on itse asiassa kaikissa näissä asioissa erinomainen ohjaustyökalu, koska se erottaa businesslogiikan itse sovelluksista ja webservices-tapaisista kehitysvistä standardeista. Liiketoimintaprosessi tunnistaa myös käsitteiden tarpeellisuuden käsite-sanan laajimmassa merkityksessä.

Liiketoiminnan mallinnuksen tehtävä ei ole tänä päivänä pelkästään kuvata yksittäisiä osia, vain kysymys on siitä, että ymmärretään, kommunikoidaan ja toteutetaan laajoja kokonaisuuksia ja niiden vuorovaikutuksia. Jotta organisaatio tekisi oikeita asioita oikein, tarvitaan malleja prosesseista ja niitä tukevista tietovirroista, mutta myös organisaation rooleista, osaa-

misesta ja käsitteistöstä. Malleihin pitää yhdistää myös strategia, mittarit ja IT-kaupunkisuunnittelu (eli erilaisten IT-arkkitehtuurien konglomeraatti). Ilman sitä mallinnuksen hyöty jää puolittiehen.

Prosessiajattelu tarjoaa erinomaisen abstrahointi- ja absorbointitason uusia IT-virtauksia ajattelen, erityisesti kun kustannus- ja hyötynäkökulma otetaan mukaan tarkasteluun.

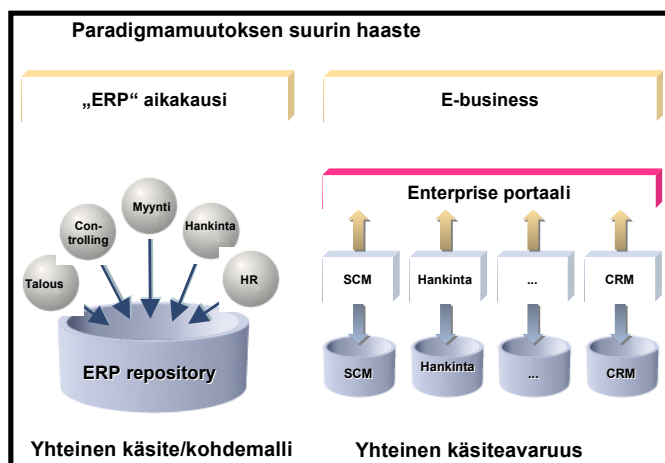
Tässä artikkelissa käsitellään jatkossa lähinnä ”tietoa” lähellä olevaa käsitteistöä, muu prosesseihin liittyvä käsitteistö prosessien rooleista alkaen WPEL4WS- ja muihin mukaviin lyhenteisiin rajataan pois. Lähestymistapa on karkeasta tasosta erikoismallinnukseen.

Käsitteistön ja käsitteiden ymmärtäminen ja hallinta synonyymeineen ja homonyymeineen on jokaisen organisaation keskeinen kyky kehittämiseen. Sanoilla ja erityisesti oman businessslingin ”substantiiveilla” on usein eri merkitys eri henkilöille. Kokonaisoptimoinnin aikakautena terminologian yhtenäistämällä (ja sen ylläpidolla) on keskeinen viestinnällinen merkitys. Eikä se helpota, että käsit-

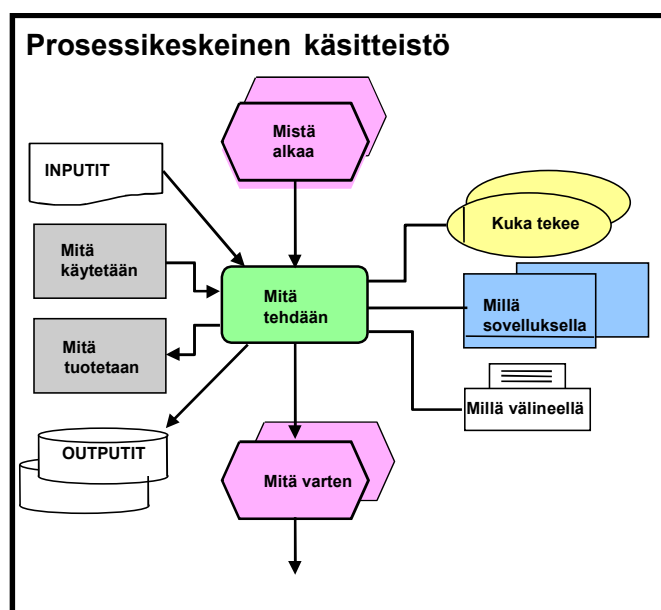
teistön yhtenäistäminen kattaa yhä enemmän myös keskeiset liikekumppanit. Tuohan jo sisäisten käsitteiden standardointi mukanaan laajoja keskusteluja ja ristiriitoja. Tästä syystä standardien kuten SCOR, RosettaNet, ebXML tai ITIL tarjoama yhtenäinen käsitteistö helpottaa yhteistoiminnallista kanssakäymistä merkittävästi.

E EEEEEEEEE EEEE EEEEEEE EE
me siitä opimme.

Pitkään on yritetty luoda ”enterprise data model”. Lähestymistapa oli kuitenkin liian IT-keskeinen, kyseinen malli ei valmistunut Babelin tornin tapaan koskaan (mikä pitää ensin oppia hyväksymään), ja viimeistään valmissovellusten laaja käyttö teki tämän jalon tavoitteen mahdottomaksi. Yhtenäinen tietomalli oli itse asiassa toiminnanohjausjärjestelmien keskeinen pääoma. Valitettavasti myös tämä kausi on päätymässä, monista eri syistä. Kuva 1 havainnollistaa tämän ison muutoksen valmisohjelmistorintamalla – käyttäjäorganisaatiot ovat olleet itse asiassa jo pitkään tässä tilassa. Prosessiajattelu ja sen tietotekniset sisaret kuten EAI, Web Services ja BPML ovat ainoa mahdollinen ulospääsy tästä dilemmasta.



Kuva 1. Toiminnanohjausjärjestelmä ja "Web-services"



Kuva 2. Erilaiset käsitteet prosessin avulla integroituina

Enterprise Data Model ei siis lunastanut lupaustaan, ja sen takia trendi on ollut hieman pois laajasta käsite-mallintamisesta enemmän prosessi-mallintamiseen. Tämä näkyy myös ihan hiljattain tietomallinnusohjelmis-

Prosessiajattelu auttaa tässä monessa mielessä. Se tunnistaa erilaisen käsitteiden (ja tietojen) käytön prosessin toimintojen yhteydessä (ks. kuva 2), ja kuvaa näin niin tietovirtoja

tojen laajennuksena prosessimallinnuksen suuntaan. Käsitteiden käsittämättömyys on sen seurauksena kuitenkin lisääntynyt. Tarvitsemme edelleen tavan ymmärtää ja hallita niin strukturoituja kuin "epämääräisiäkin" käsitteitä, johon sisältyy myös käsitteiden erilaisia ilmentymiä. Lähtökohtana käsitteiden hallintaan tulee kuitenkin olla nimenomaan liiketoimintaprosessi eikä päinvastoin. Tähän sisältyy niin ison organisaation monikielisiä käsitteitä eli sanastoja, kuten myös saman tiedon erilaisia ilmentymiä eri sovelluksissa.

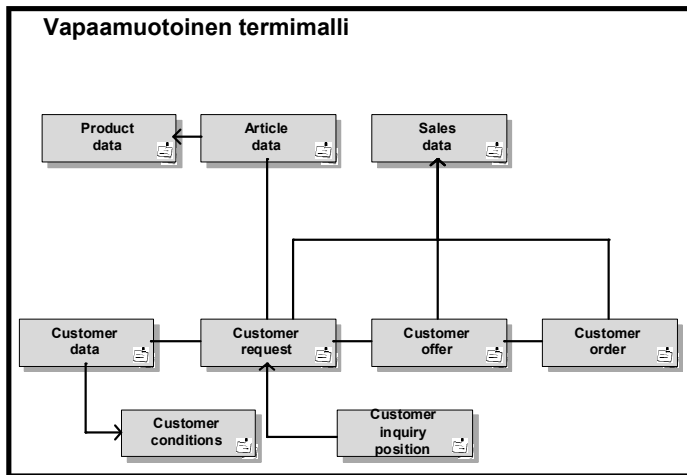
Prosessiajattelu auttaa.

kuin myös mahdollisia roolien välisiä ymmärtämiskykyjä. Prosessiajattelun yhteydessä on opittu ymmärtämään hierarkkisia malleja, mallien variantteja ja niiden yhteydessä erilaisien menetelmien tarpeellisuus. Uimara kun on joskus myrkyä kehittämiseksi. Samat periaatteet ovat sovellettavissa myös käsitteiden hallintaan: karkean tason mallilla ajetaan erilaisia tavoitteita kuin tarkan tason mallilla. Niin integraatio on kuitenkin laadullisen mallinnuksen peruspilari.

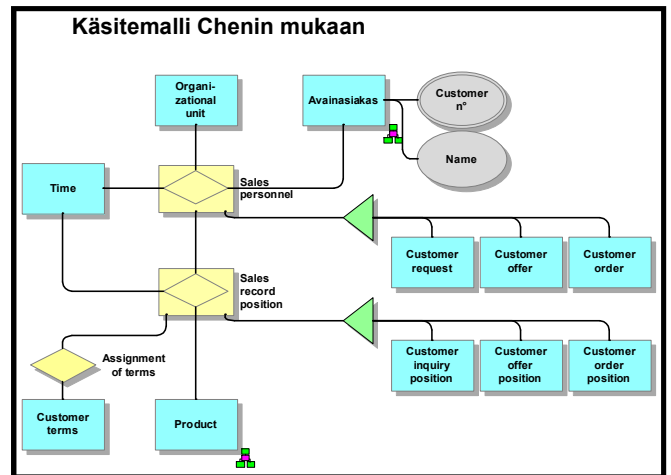
Samasta syystä esim. muutenkin vielä paikkaansa hakevan OMG-standardointiorganisaation (www.omg.org) CWM-standardi (Common Warehouse Metamodel) tarjoaa vain osittaisen ratkaisun.

Tällainen integroitu mallinnus onnistuu parhaiten, kun taustalla on repository eli kuvaustietokanta – ilman että käyttäjien tarvitsee edes tietää, että ne ovat kuvaustietokannan kanssa tekemisessä. Kuvaustietokannan tulee olla myös hieman joustava sen suhteen, että se sallii esim. samanlaisia (samannimisiä) käsitteitä - toisin kuin joskus CASE-työkalussa.

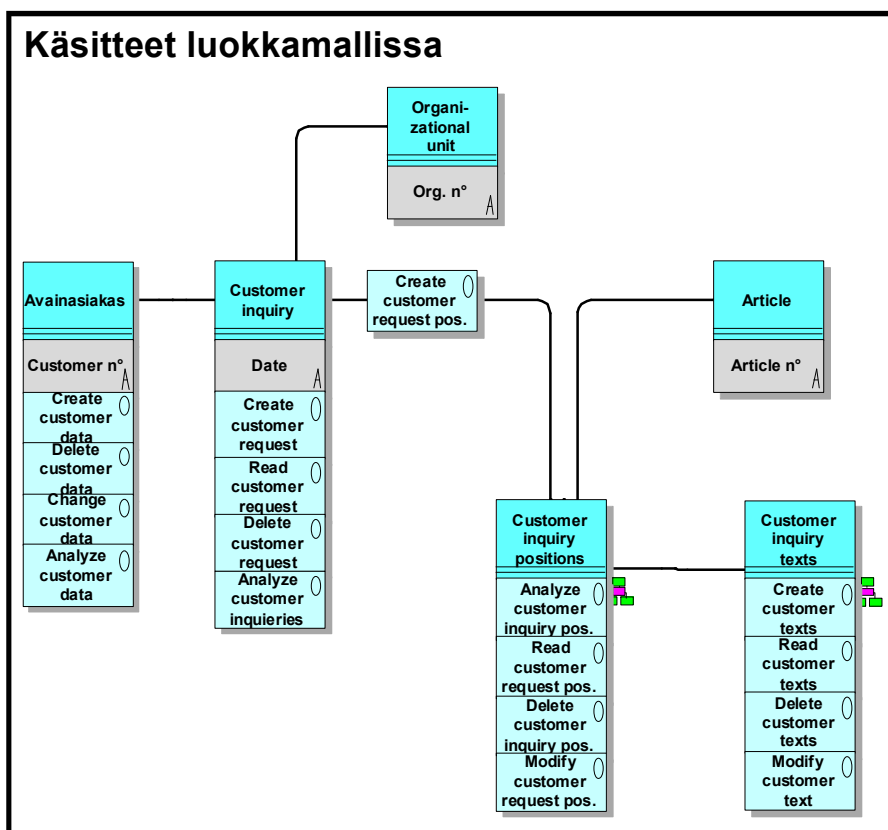
Liiketoimintalähtöinen käsitteistö voidaan mallintaa esim. terminalilla (kuva 3), joka toimii toisaalta pelkkänä graafisena sanastona (selityksi-neen), mutta sallii kuitenkin myös termien välisten suhteiden mallinnuksen ilman relaatiotietokannan "relaation" rajoittunutta tulkintaa.



KKKKKKKKKK K KKKKKKK KKKKKKKKKKKKKKK



Kuva 4. Laajennettu käsitemalli Chenin mukaan



Kuva 5. Käsitteet luokkamallissa, löytyykö?

Kun liiketoimintatarpeita tarkennetaan IT-määrittelyvaiheen yhteydessä, voidaan termejä hienontaa esim. käsitemallilla (eli kohdemallilla huh!) tai luokkamallilla. (kuva 4 ja 5). Tällöin tulee tietysti tietotekninen ulottuvuus mukaan, joka ei ole aina hyväksi. Tästä syystä olen myös sitä mieltä, että luokkamallin käyttö käsitteiden hallintaan ei ole hyväksi. Luokkamalli kuvaa omalla erinomaisella tavallaan tietoteknisen mallinnuksen etenemistä, mutta lopputulos on IT-ammatilliselle tarkoitettu. Käsitteen omistaja on kuitenkin liiketoiminta. Tästä syystä käsitteen tulisi elää myös luokkamallin ulkopuolella, ja integraatio luokkamalliin tapahtuu nimenomaan kuvaustietokannan kautta. Sama pätee tietysti myös käsitemallille. Olen sitä mieltä että kaikkien kolmen mallin olemassaolo on edelleen perusteltu –vain tavoitteesta yhdestä yhteisestä mallista on syytä luopua. Kuvaustietokanta ja prosessi hoitaa in-

tegraation, ja prosessin kautta syntyy myös tietyn hetken (siis hetken!) oikea käsitteistö.

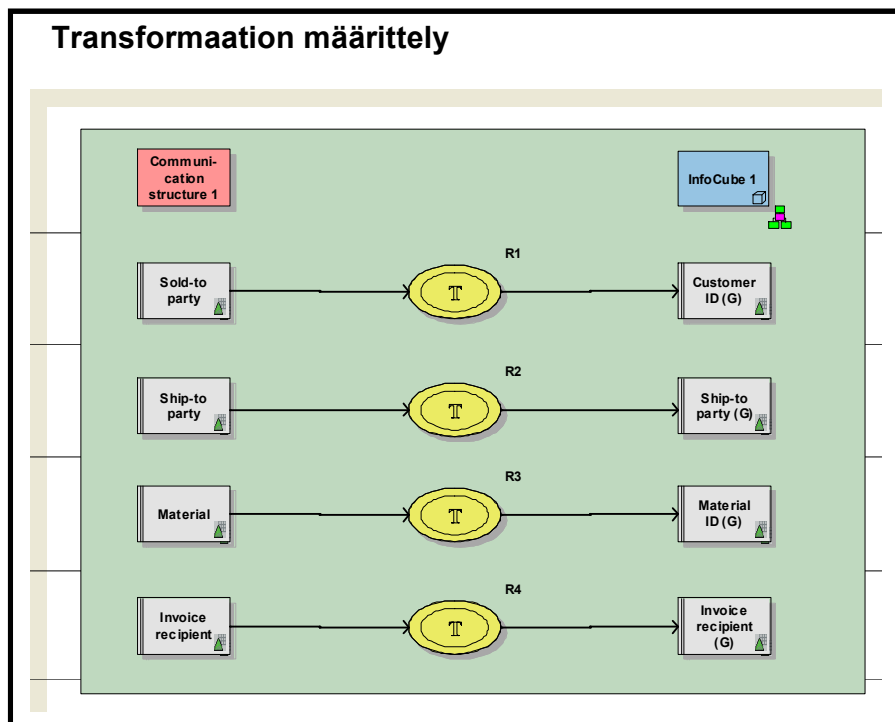
Prosessin poikkitieteelliset ulottuvuudet

Käsitteiden nimeäminen ja tiedon sisältö vaihtelee eri IT-järjestelmissä, ja tietovarastojen ”konsolidoidut” kannat ovat siitä hyvää esimerkkiä. Liiketoimintaprosessi määrittelee tässäkin asiassa tekniset tarpeet. Käytännössä se tarkoittaa myös käsitteiden yhteensovittamista (ks. esim. kuva 6) ja tiedon erilaisten esitystapojen transformaatiota (pieni esim. kuva 7).

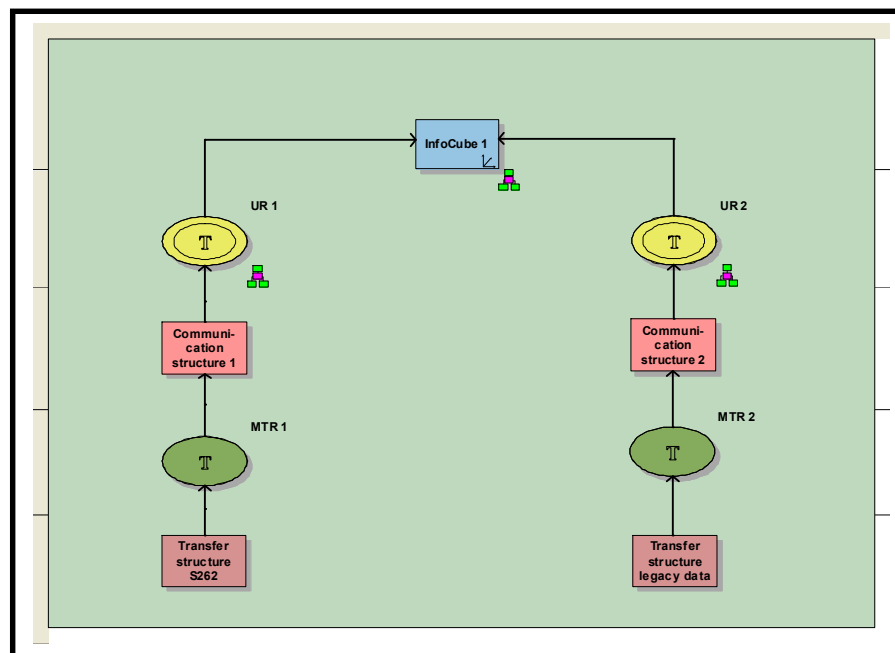
Ihminen on keskeinen osa liiketoimintaprosessia, ja roolien erikoisoaaminen on yksi tärkeä ulottuvuus. Tietämyksen hallinta voidaan niin ikään nähdä tärkeänä käsitteiden hallinnan ulottuvuutena, joka kattaa niin IT-tuettuja tietämyksen varastoja kuin myös hiljaista tietämystä. Molemmat voidaan luonnollisesti mallintaa (jossa myös NLP-konsultit eli mallittajat voivat ottaa omansa). Tietämyksen rakenteen luominen on jo sinänsä käsitteiden määrittelyä, mutta se voi liittyä myös suoraan muihin käsitteisiin. Pieni esimerkki on kuvassa 8.

E2E = End to End = liiketoimintaprosessi

Asiakaslähtöisen liiketoimintaprosessien ”kylkeen” mahtuu paljon käsitteitä. Joskus kannattaa aloittaa peräti ”mielle”-tasolta, joskus pitää porautua hyvin syvälle attribuutti- eli ominaisuustasolle. Kannattaa kuitenkin



Kuva 6. Käsitteiden transformaatio



Kuva 7. Tietovaraston muodostaminen

kin huolehtia siitä, että karkea käsite ja tarkka attribuutti kuvataan eri tasolla. Yleinen käsitteistö kannattaa paimentaa – ja pitää myös rajallisena.

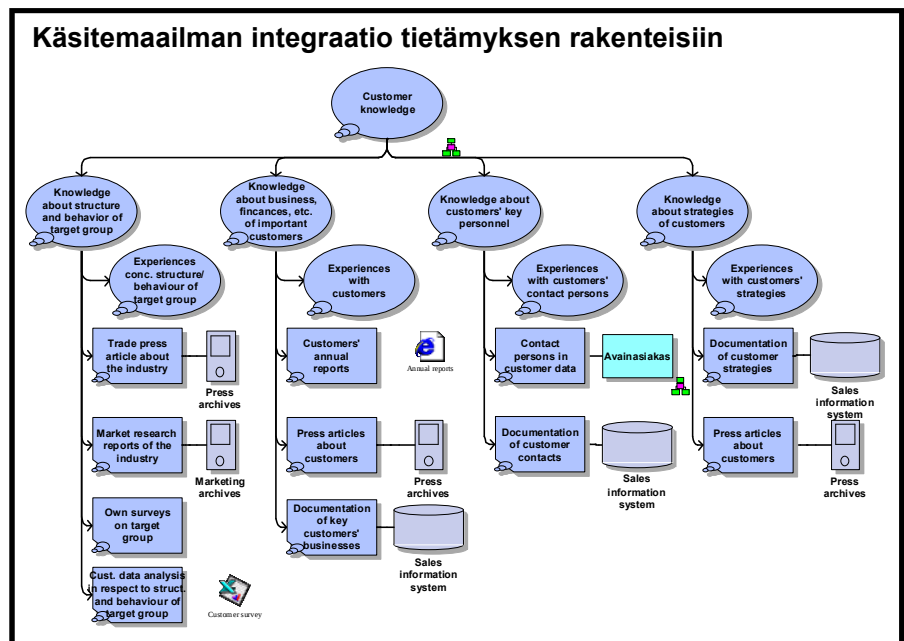
Kuten jo alussa mainittu, muodostavat myös mittarit tärkeitä käsitteitä, joiden merkitystä – mutta myös vaikeutta – ei voi yliarvioida. Ja pienellä pohdinnalla löytynee muitakin käsitteiden ulottuvuuksia.

Kaiken kaikkiaan – ihmisille pitää tarjota mahdollisuus hakea selityksiä käsitteisiin Intranetista, mutta tarvitaan myös change management käsitteiden kehittämiseen ja synonyymien hallintaan – kuinka monesta organisaatiosta löytyy tällainen palautekanava ja kuka on CTO eli Chief Term Officer ?

IT-mielessä on tärkeätä nähdä se ulottuvuus, että prosessi tarvitsee integraatiota eikä pelkästään yhteistoinnallisuutta eli rajapintoja. Sillä on vaikutusta käsitteiden hallintaan, ja harvoin esim. EAI-ratkaisu on siihen riittävä.

Käsitteiden hallinnan kypsyysmalli

En tiedä, tarvitaanko käsitteiden hallinnan kypsyysmallia. Tässä kuitenkin muutama kypsyysmallin mahdollinen ”rajapyykki”.



Kuva 8. Tietämyksen rakenne ja käsitteet

- Organisaatio osaa erottaa strukturoituja ja ei-strukturoituja käsitteitä
- Käsite löytää tiensä (eli on jäljitettävissä) liiketoiminnan termistä relaatiotietokantaan asti
- Käsitteet ovat luokiteltuja (mittarista – attribuutteihin)
- Käsitesanasto löytyy Intranetista
- Käsitteiden palautekanava on olemassa ja muutoshallinta toimii
- Prosessi ohjaa käsitteitä
- ...

Järjestystä ja kaatavuutta ei ole vielä lyöty lukkoon, enemmän on tarkoitus käynnistää tällä ajatuksella keskustelu, ja saa lähettää kritiikkiä ja ehdotuksia. Yhteystiedot ovat alla.

Käsitteiden hallinnan ongelmallisuus tiivistyy sanontaan: enemmän en kehtaa kirjoittaa

Claus Günther,
ChangeWare Group,
claus.gunther@cwgf.fi,
puh. 050-555 1110

Fle³-kokeilu: lyhenteiden avausta ryhmätyöympäristössä

Helena Venäläinen,
FD Finanssidata Oy

Alallemme on tyypillistä erilaisten englanninkielistä alkuperää olevien lyhenteiden viljely. Käynnistimme lehden teemaan liittyen kokeilun lyhenteiden yhteisöllisestä selittämisestä Sytyke ry:n uudella ryhmätyöalueella.

Sytyke ry:n web-toimikunta on kehittämässä sivustomme teknistä alustaa verkossa tapahtuvaa yhteisöllistä tiedonkehittelyä tukevaksi. Computer-Supported Cooperative Work (CSCW) – yhteistyön tietokonetuki tuntuu varsinkin valtakunnallista yhdistystämme ajatellen tarpeelliselta ominaisuudelta. Aluetta on tutkittu tietojenkäsittelytieteessä reilut kymmenen vuotta ja ohjelmistoja on kehitetty tutkimuksen pohjalta. Computer supported collaborative learning (CSCL) – yhteisöllinen opiskelu on toinen mielenkiintoinen ja vastaavantyyppinen tutkimusalue. Ongelmien ratkaisu meidän alalla kun edellyttää hyvin usein uuden tiedon hankintaa ja oppimista.

Web-toimikunta päätti keväällä pilotoida Taideteollisen korkeakoulun ja Helsingin yliopiston psykologian laitoksen toteuttamaa Fle3-nimistä ryhmä-

mätyöohjelmistoa. Tämä tulevaisuuden oppimisympäristö (Future Learning Environment) on rakennettu erityisesti uuden tiedon kehittelyä ja ongelmien ratkaisua tukevaksi ja tuntui siksi meillekin sopivalta välineeltä. Johtokunta ja muutamat toimikunnat saivat omat työalueet ja jaettujen kansiodien tarjoama ”yhteinen muisti” on jo todettu hyväksi ominaisuudeksi.

Päätimme alkukesästä laajentaa kokeilua tiedonrakentelun suuntaan lehtemme Käsittämättömät käsitteet – teemaan liittyen. Avasimme järjestelmään ns. vierailualueen, jonne kutsumme parikymmentä henkilöä kokeilemaan lyhenteiden yhteisöllistä avaamista. Lyhenteiden selvittämisen pelisäännöt kirjattiin seuraavasti:

Jos et tiedä, mitä joku lyhenne tarkoittaa, voit

- 1) etsiä vastausta ongelmaasi olemassaolevien keskusteluiden joukosta
- 2) avata tälle alueelle uuden keskustelun ongelmasi selvittämiseksi. Nimeä keskustelu tällöin lyhenteellä.

Jos sinulla on tietoa, mitä joku lyhenne tarkoittaa, voit vastata kyselyyn ja kertoa mitä tiedät lyhenteestä.

Tyypitä vastauksesi (omaa arvailua, lähteeseen perustuvaa, keskustelun jäsentelyä).

Kokeilu lähti käyntiin ja vierailualueella löytyy jo joitakin lyhenteitä: EAI, BPEL4WS, ebXML, SOAP, WSCL, WSDL ja XP. Kokeilua on häirinnyt aurinkoinen kesä lomineen sekä pilot-järjestelmän piileskely muistamista vaativan linkin takana. Linkki löytyy nyt Sytyke ry:n etusivulta.

Jokainen uusi työkalu edellyttää sen käytön opiskelua, tämänkin järjestelmän käyttöohjeen lukeminen auttaa alkuun. Työpöytä, tiedonrakentelu ja jammailu tulevat toimintoina selviksi ja oman kuvan lisääminenkin selviää näin parhaiten. Ohje löytyy kirjautumissivulta ja sen pääsee lukemaan ilman käyttäjätunnuksia. Tunnuksen Fle3:n vierailualueelle taas saa lähettämällä rekisteröintipyyntönsä sähköpostilla web-toimikunnalle. Java-kerholla ja testauskerholla on omat vastaavat toiminnot sisältävät työalueet, niihin pääsee mukaan (samalla tunnuksella) ottamalla yhteyttä kerhonvetäjiin. Tervetuloa mukaan verkostoitumaan verkkoonkin!

*Web-toimikunnan puolesta,
Helena Venäläinen*