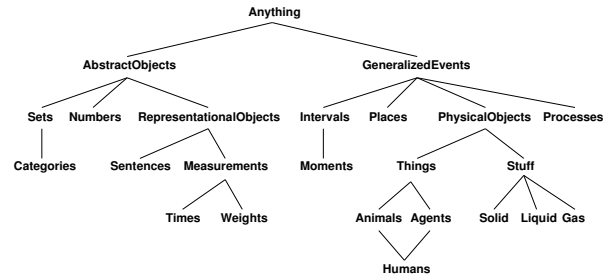


- Nyt käytössämme on riittävän vahva tietämyksenesityskieli päättelymekanismeineen, jotta voimme yrittää esittää tietämystä maailmasta.
- Aiemmin on nähty esimerkkejä sovellusalaakohtaisen tietämyksen esittämisestä, mutta älykkään agentin pitäisi hallita yleiset käsitteet. Eri alojen tietämys pitäisi sitten voida lisätä järjestelmään ja sovittaa yhteisen käsitejärjestelmän alle.
- Tätä varten on pyritty organisoimaan kaikkea maailmassa olevaa *kategorioksi*, jotka jakavat olemassaolevan erillisiin luokkiin. Luokkia voi edelleen hienontaa aliluokiksi, jolloin saadaan *taksonomia* (taxonomy) (tai *taksonominen hierarkia*). Nykyään puhutaan myös *ontologioista*, vaikkakin termi perinteisesti tarkoittaa oppia olevasta.

lukut10.tex -- 17/11/2005 -- 16:35 -- p.1/40



Tässä on esitetty yksi yritys luokitella kaikkea maailmassa olevaa.

lukut10.tex -- 17/11/2005 -- 16:35 -- p.2/40

TAKSONOMIOIDEN ESITTÄMINEN

- Taksonomioita voidaan logiikassa esittää joko predikaatteina tai olioina. Esimerkiksi jos halutaan sanoa että b on koripallo, voidaan sanoa joko $Basketball(b)$ tai sitten voimme käyttää *reifiikaatiota* (reification) ja puhua koripallojen luokasta oliona $Basketballs$ ja sanoa $b \in Basketballs$.
- Tällaisiin hierarkkisiin suhteisiin liittyy *periytymistä* (inheritance), jossa aliluokkien edustajilla on myös ylempien luokkien ominaisuudet. Jos pallojen olemukseen kuuluu pyöreys, niin koripallotkin ovat pyöreitä koska $Basketballs \subset Balls$.
- Esimerkiksi biologiassa luokitukset ovat yleisiä. Myös kirjastoluokitukset ja erikoisalojen aihealueet (esim. ACM:n Computing Classification System) ovat esimerkkejä niistä.

lukut10.tex -- 17/11/2005 -- 16:35 -- p.3/40

Luokkien ja yksilöolioiden suhteita:

- Olio kuuluu luokkaan. Esim. $BB_9 \in Basketballs$
- Luokka on toisen aliluokka. Esim. $Basketballs \subset Balls$
- Kaikilla luokan jäsenillä on jokin ominaisuus. Esim. $x \in Basketballs \rightarrow Round(x)$.
- Olio voidaan tunnistaa luokan jäseneksi ominaisuuksien perusteella. Esim. $Orange(x) \wedge Round(x) \wedge Diameter(x) = 9.5'' \rightarrow x \in Basketballs$.
- Luokalla itsellään on jokin ominaisuus. Esim. $Dogs \in DomesticatedSpecies$.

lukut10.tex -- 17/11/2005 -- 16:35 -- p.4/40

LUOKKIEN SUHTEISTA

- Luokat ovat *erillisiä* (disjoint), jos niillä ei ole yhteisiä jäseniä.
- Luokan jako aliluokkiin on *kattava* (exhaustive), jos jokainen luokan alkio kuuluu johonkin aliluokkaan.
- Jos jako aliluokkiin on kattava ja kaikki aliluokat ovat erillisiä, jako on *ositus* (partition).

Esimerkkejä:

- $Disjoint(\{Animals, Vegetables\})$
- $ExhaustiveDecomposition(\{Americans, Canadians, Mexicans\}, NorthAmericans)$
- $Partition(\{Males, Females\}, Animals)$. (Ei taida olla kovin hyvä esimerkki.)

lukut10.tex -- 17/11/2005 -- 16:35 -- p.5/40

MÄÄRITELMÄT

- Luokkia voidaan myös määritellä antamalla riittävät ja välttämättömät ehdot luokkaan kuulumiselle. Esim. $x \in Bachelors \leftrightarrow Unmarried(x) \wedge x \in Adults \wedge x \in Males$.
- Määritelmiä on erilaisia, perinteisesti on eroteltu nimiä koskevat nominaalimääritelmät ja olioita koskevat reaaliominaisuuksien määritelmät. Edelliset ovat sopimuksia nimien käytöstä, jälkimmäiset faktuaalisia lauseita, jotka voivat olla tosia tai epätosia. ($\rightarrow$ essentialismi?)
- Määrittely on usein ongelmallista, esim.:
 - Natural kinds
 - Essentialismin kritiikki (Wittgenstein, perheyhtäläisyys)
 - Analyttisten lauseiden kritiikki (Quine)

lukut10.tex -- 17/11/2005 -- 16:35 -- p.6/40

FYYSINEN KOOSTUMINEN

- Olioilla voi olla osia, joten tarvitaan relaatio *PartOf*, joka on refleksiivinen ja transitiivinen:
 - $PartOf(x, x)$
 - $PartOf(x, y) \wedge PartOf(y, z) \rightarrow PartOf(x, z)$.
- Tämän avulla voidaan määritellä sekä *koosteisia olioita* (composite objects) että useiden olioiden kimppuja (bunch).
 - $\forall x \ x \in s \rightarrow PartOf(x, BunchOf(s))$
 - $\forall y \ [\forall x \ x \in s \rightarrow PartOf(x, y)] \rightarrow PartOf(BunchOf(s), y)$.

- Määritelmät voidaan myös jakaa deskriptiivisiin ja stipulatiivisiin (ks. Niiniluoto 1984).
- Deskriptiiviset pyrkivät kuvailemaan käytössä olevien termien merkityksiä (analyttinen määritelmä) tai niiden ekstensioita (synteettinen määritelmä). Aiempi poikamiehen määritelmä olisi analyttinen ja vanha esimerkki "ihminen on höyhenetön kaksijalkainen" olisi synteettinen.
- Stipulatiiviset voidaan jakaa nominaalimääritelmiin, joissa sovitaan uuden kielellisen ilmauksen merkityksestä, ja eksplikaatioihin, joissa täsmennetään jonkin moniselitteisen tai epämääräisen termin merkitystä. Esimerkki edellisestä on vaikkapa " $a^n = a \cdot a \cdot \dots \cdot a$ (n kertaa)" ja jälkimmäisestä logiikan totuusmääritelmä.

lukut10.tex -- 17/11/2005 -- 16:35 -- p.7/40

lukut10.tex -- 17/11/2005 -- 16:35 -- p.8/40

- Numeerisia mittoja voidaan ilmaista *yksikköfunktioiden* (units function) avulla, kunhan tehdään ontologinen sitoumus, että on olemassa sellaisia abstrakteja olioita kuin esimerkiksi viivan "—" pituus, jolle voidaan antaa nimeksi vaikkapa 3.81 cm.
- Nyt voidaan sanoa vaikkapa että $Length(L_1) = Inches(1.5) = Centimeters(3.81)$.
- Aina ei tarvita kvantitatiivisia mittoja, vaan päätöksenteon kannalta riittää jos mittoja voidaan vertailla, jolloin voidaan käyttää jotakin vertailurelaatiota >.

lukur10.tex -- 17/11/2005 -- 16:35 -- p.9/40

- Vielä täytyy tehdä erottelu aineiden ja olioiden välille, koska ne käyttäytyvät loogisesti eri tavoin.
- Esim. voin osat ovat myös voita:
 $x \in Butter \wedge PartOf(y, x) \rightarrow y \in Butter$.
- Jaottelu *sisäisiin* (intrinsic) ja *ulkoisiin* (extrinsic) ominaisuuksiin. Sisäiset ominaisuudet kuuluvat itse aineelle josta olio koostuu ja ne säilyvät osiin jaettaessa, kun taas ulkoiset kuuluvat vain kokonaisuudelle oliolle. Sisäisiä ovat esimerkiksi tiheys, väri, kiehumispiste ja ulkoisia esimerkiksi paino, pituus, muoto.
- Esimerkkiluokituksessa oliot, joilla on vain sisäisiä ominaisuuksia, kuuluvat kategoriaan *Stuff* ja muut luokkaan *Thing*.

lukur10.tex -- 17/11/2005 -- 16:35 -- p.10/40

TILANNEKALKYYLI

John McCarthy'n *tilannekalkyyli* (situation calculus) on looginen tietämyksenesityskieli tekojen ja tilanteiden esittämiseen.

- Toimintoihin viitataan loogisilla termeillä kuten aiemminkin:

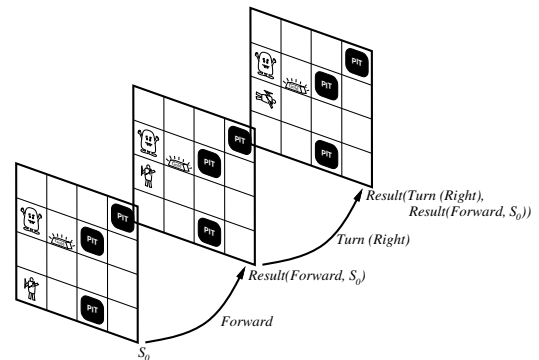
Forward, Turn(Right)

- Tilanteisiin* (situation) viitataan loogisilla termeillä: alkutilanne S_0 ja kaikki tilanteet jotka saadaan kun sovelletaan toimintoja tilanteisiin. Funktio $Result(a, s)$ ilmoittaa tilanteen, johon päädytään kun tehdään teko a tilanteessa s . Ei siis eksplisiittisiä ajanhetkiä $t_1, t_2, \dots$

$$Result(Forward, S_0) = S_1$$

lukur10.tex -- 17/11/2005 -- 16:35 -- p.11/40

TILANTEET



lukur10.tex -- 17/11/2005 -- 16:35 -- p.12/40

AJALLISET JA AJATTOMAT PREDIKAATIT

- Predikaatit, jotka ovat aina voimassa, esitetään kuten predikaattilogikassa tavallisesti:

$$Pit([3, 1]).$$

Näitä kutsutaan *ajattomiksi* (atemporal) predikaateiksi (tai funktioiksi).

- Ajan mukaan muuttuvilla predikaateilla (fluent) on ylimääräinen argumentti, joka kertoo missä tilanteissa predikaatti on voimassa:

$$\neg Holding(G_1, S_0).$$

Näitä voisi kutsua *ajallisiksi* predikaateiksi (fluent).

lukur10.tex -- 17/11/2005 -- 16:35 -- p.13/40

TOIMINTASARJAT

Yksittäisten toimintojen lisäksi voidaan käsitellä toimintasarjoja:

- Tyhjä toimintasarja ei muuta mitään:

$$Result([], s) = s.$$

- Toimintasarjan tekeminen tarkoittaa että ensin tehdään ensimmäinen teko ja sen jälkeen muut:

$$Result([a|seq], s) = Result(seq, Result(a, s)).$$

lukur10.tex -- 17/11/2005 -- 16:35 -- p.14/40

PROJEKTIO JA SUUNNITTELU

Tilannekalkyyliä käyttävän agentin pitäisi pystyä suorittamaan sekä projektio- että suunnittelutehtäviä:

*Projektio*tehtävissä (projection task) agentti päätelee tekojensa seurauksia.

Suunnittelutehtävissä (planning task) agentti pyrkii löytämään toimintasarjan, joka johtaa haluttuun tilanteeseen.

lukur10.tex -- 17/11/2005 -- 16:35 -- p.15/40

ESIMERKKI

- Oletetaan, että agentti voi liikkua mihin tahansa viereiseen ruutuun eikä välitetä agentin asennosta. Olkoon agentti aluksi ruudussa [1,1] ja kulta (G_1) ruudussa [1,2]. Tavoitteena on saada kulta ruutuun [1,1].
- Ajalliset predikaatit ovat: $At(o, x, s)$ ja $Holding(o, s)$.
- Aluksi tietämyskanta sisältää lauseet:
 - $At(o, x, S_0) \leftrightarrow [(o = Agent \wedge x = [1, 1]) \vee (o = G_1 \wedge x = [1, 2])]$
 - $\neg Holding(o, S_0)$
 - $Gold(G_1) \wedge Adjacent([1, 1], [1, 2]) \wedge Adjacent([1, 2], [1, 1])$.

lukur10.tex -- 17/11/2005 -- 16:35 -- p.16/40

- Nyt halutaan todistaa että agentti saavuttaa tavoitteen menemällä ruutuun [1,2], ottamalla kullan ja palaamalla ruutuun [1,1]:
 $At(G_1, [1, 1], Result([Go([1, 1], [1, 2]), Grab(G_1), Go([1, 2], [1, 1])], S_0)).$
- Voisimme myös olla kiinnostuneita löytämään toimintasarjan, joka johtaa tavoitteen toteutumiseen, jolloin meidän pitäisi tehdä kysely:
 $\exists seq At(G_1, [1, 1], Result(seq, S_0)).$

lukur10.tex -- -- 17/11/2005 -- 16:35 -- p.17/40

Tilannekalkyyliissa kuhunkin tekoon liitetään mahdollisuus- ja seurausaksioomat.

Mahdollisuusaksiooma (possibility axiom) kertoo, missä olosuhteissa teko on mahdollista suorittaa.

Ennakkoehdot $\rightarrow Poss(a, s)$

Vaikutusaksiooma (effect axiom) kertoo, mitä tapahtuu teon seurauksena.

$Poss(a, s) \rightarrow$ Seuraukset

lukur10.tex -- -- 17/11/2005 -- 16:35 -- p.18/40

ESIMERKKI

Modifioidun mörkömaailman aksioomat (muuttujat oletetaan universaalisesti kvantifioituiksi):

- $At(Agent, x, s) \wedge Adjacent(x, y) \rightarrow Poss(Go(x, y), s)$
- $Gold(g) \wedge At(Agent, x, s) \wedge At(g, x, s) \rightarrow Poss(Grab(g), s)$
- $Holding(g, s) \rightarrow Poss(Release(g), s)$
- $Poss(Go(x, y), s) \rightarrow At(Agent, y, Result(Go(x, y), s))$
- $Poss(Grab(g), s) \rightarrow Holding(g, Result(Grab(g), s))$
- $Poss(Release(g), s) \rightarrow \neg Holding(g, Result(Release(g), s))$

lukur10.tex -- -- 17/11/2005 -- 16:35 -- p.19/40

- Emme kuitenkaan pysty päättämään, että suunnitelmamme avulla pääsemme tavoitteeseen.
- $Go([1, 1], [1, 2])$ on kyllä mahdollinen ja siitä päätellään, että agentti pääsee ruutuun [1,2]:
 $At(Agent, [1, 2], Result(Go([1, 1], [1, 2]), S_0)).$
- Mutta toiminta $Grab(G_1)$ ei onnistu, koska mahdollisuusaksiooma vaatii, että kullan on oltava samassa ruudussa, siis
 $At(G_1, [1, 2], Result(Go([1, 1], [1, 2]), S_0)),$
 mutta tätä emme voi päätellä.

lukur10.tex -- -- 17/11/2005 -- 16:35 -- p.20/40

KEHYSONGELMA

- Kehysongelmaksi** (frame problem) kutsutaan kysymystä siitä, mitkä asiat maailmassa muuttuvat jonkin teon seurauksena ja mitkä pysyvät ennallaan.
- Tätä voidaan yrittää ratkaista käyttämällä *kehysaksioomia* (frame axiom), jotka kertovat mitkä asiat pysyvät ennallaan:
 $At(o, x, s) \wedge (o \neq Agent) \wedge \neg Holding(o, s) \rightarrow At(o, x, Result(Go(y, z), s)).$
- Tarvittavien aksioomien määrä on luokkaa $O(AF)$, missä A on tekojen määrä ja F on ajallisten predikaattien määrä.
- Kaksi versiota ongelmasta: *representaationaalinen* (aksioomat pitäisi saada ilmaistua $O(AE)$ literaalilla kun E on tekojen vaikutusten määrä) ja *inferentiaalinen* ($t:n$ teon tulokset pitäisi saada evaluoitua ajassa $O(Et)$).

lukur10.tex -- -- 17/11/2005 -- 16:35 -- p.21/40

SEURAAJATILA-AKSIOMAT

- Sen sijaan, että aksioomiksi otettaisiin jokaisen teon seuraukset, tarkastellaankin kutakin ajallista predikaattia ja yhdistetään aksioomat *seuraajatila-aksioomiksi* (successor-state axiom), jotka ovat muotoa
Teko on mahdollinen $\rightarrow$
(Ajallinen predikaatti on tosi seuraajatilassa $\leftrightarrow$
Predikaatti tuli todeksi teon seurauksena $\vee$
Predikaatti oli tosi eikä muuttunut epätodeksi).
- Nyt seuraajatila on täydellisesti määritelty, koska predikaatin totuusarvo määräytyy funktionaalisesti teon ja aiemman totuusarvon perusteella.
- Esim. $Poss(a, s) \rightarrow$
 $(At(Agent, y, Result(a, s)) \leftrightarrow$
 $a = Go(x, y) \vee (At(Agent, y, s) \wedge a \neq Go(y, z)))$

lukur10.tex -- -- 17/11/2005 -- 16:35 -- p.22/40

- Esim. $Poss(a, s) \rightarrow$
 $(Holding(g, Result(a, s)) \leftrightarrow a = Grab(g) \vee (Holding(g, s) \wedge a \neq Release(g)))$
- Nyt aksioomien yhteenlaskettu literaalien määrä on luokkaa $O(AE)$, sillä tekojen (A kpl) vaikutukset (E kpl) mainitaan vain kerran. Aksioomien lukumäärä on F .
- Vieläkään ei voi päätellä tavoitteen onnistumista, sillä aksioomat eivät ilmaise sitä *implisiittistä seurausta*, että kannettu kulta siirtyy agentin mukana. Tätä kutsutaan *ramifikaatio-ongelmaksi* (ramification problem).
- Tässä tapauksessa ongelmasta selvittää ottamalla implisiittiset seuraukset huomioon seuraajatila-aksioomassa: $Poss(a, s) \rightarrow$
 $At(o, y, Result(a, s)) \leftrightarrow (a = Go(x, y) \wedge (o = Agent \vee Holding(o, s))) \vee (At(o, y, s) \wedge \neg(\exists z y \neq z \wedge a = Go(y, z) \wedge (o = Agent \vee Holding(o, s))))$

lukur10.tex -- -- 17/11/2005 -- 16:35 -- p.23/40

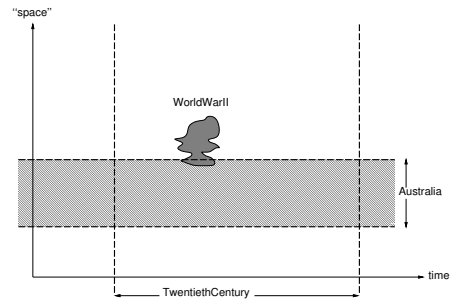
- Tällaisen aksiooman käsittely vaatii, että yksilöliot voidaan todistaa ei-identtisiksi. Logiikassa vakiosymbolit voivat viitata samoihin olioihin, mutta tietämyskannoissa tämä usein kielletään päättelyn yksinkertaistamiseksi. Tämä voidaan tehdä joko lisäämällä tietämyskantaan aksioomat muotoa $C_i \neq C_j$ kaikille i, j tai oletamalla nimien erillisyyss päättelymekanismin tasolla.
- Vastaavasti myös tekojen erillisyyss täytyy esittää aksioomilla.
- Näillä keinoilla voidaan viimein todistaa suunnitelma toimivaksi.
- Inferentiaalinen kehysongelma vaatisi vielä huomiota. . .

lukur10.tex -- -- 17/11/2005 -- 16:35 -- p.24/40

- Tilannekalkyylit toimii hyvin kun agentteja on vain yksi ja teot ovat hetkellisiä, mutta jos toimijoita on useita ja tekojen kestot ovat erilaisia, tarvitaan tarkempaa ajan hetkien esitystä.
- Tapahtumakalkyylissä (event calculus) predikaatit ovat voimassa jostakin tapahtuman alkuhetkestä johonkin loppuhetkeen asti ellei se jostakin syystä keskeytynyt. Tarvitaan lisäpredikaatteja:
 - $Initiates(e, f, t)$ – Tapahtuma e hetkellä t tekee f :n todeksi
 - $Terminates(w, f, t)$ – f lakkaa olemasta tosi
 - $Happens(e, t)$ – e tapahtuu hetkellä t
 - $Clipped(f, t, t_2)$ – jokin tapahtuma ajanhetkien t ja t_2 välillä keskeyttää f :n

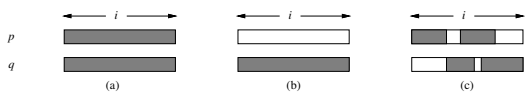
lukur10.tex -- 17/11/2005 -- 16:35 -- p.25/40

Tapahtuman käsitettä voi yleistää koskemaan kaikkia aika-avaruuden kokonaisuuksia, jolloin teot, ajat, paikat ja fysikaaliset oliot käsitetään tapahtumiksi.



lukur10.tex -- 17/11/2005 -- 16:35 -- p.26/40

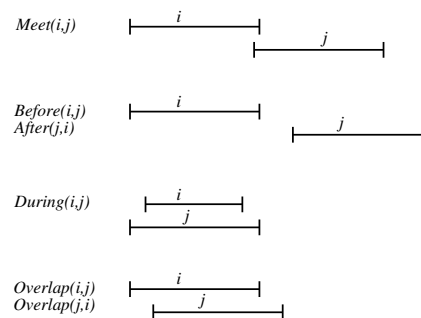
- Tapahtumien käsittelyyn tarvitaan useita predikaatteja ja funktioita kuten *SubEvent*, *Period*, *Duration*, *In jne*.
- Diskreeteillä tapahtumilla on määrätynlainen rakenne, esim. alku ja loppu, mutta lisäksi pitäisi voida käsitellä myös *prosesseja*, joille on tyypillistä että niiden osat kuuluvat samaan prosessikategoriaan.
- Tapahtumien yhdistämistä varten on kehitetty oma esitystapansa, *fluent calculus*, jonka avulla voidaan puhua monimutkaisista tapahtumista.



(a) $T(Both(p, q), i)$ (b) $T(OneOf(p, q), i)$ (c) $T(Either(p, q), i)$

lukur10.tex -- 17/11/2005 -- 16:35 -- p.27/40

- Myös ajanhetket ja aikavälit vaativat määrittelyitä.



- Toinen tapa ajan käsittelyyn on *temporaalilogiikka*.

lukur10.tex -- 17/11/2005 -- 16:35 -- p.28/40

MENTAALISET TILAT

- Moniagenttijärjestelmissä* (multi-agent systems) täytyy voida tehdä päätelmiä toisten agenttien *propositionaalisista asenteista* (propositional attitudes) (uskomukset, halut jne).
- Ongelmana on *opaakisuus* (referential opacity): samaan oloon viittaavia termejä ei voi korvata toisillaan uskomuskonteksteissa: Clark Kent ja Teräsmies viittaavat samaan henkilöön, mutta vaikka Lois uskoo että Teräsmies osaa lentää, hän ei usko että Clark Kent osaa lentää.
- Uskomuksien käsittelyyn on ehdotettu useita erilaisia menetelmiä, sekä syntaktisia että semanttisia. Tärkein lienee Jaakko Hintikan kehittämä *episteeminen logiikka*.
- Sopivan vahvuisen päättelyjärjestelmän löytäminen on vaikeaa: vaarana joko liiallinen heikkous tai toisaalta *looginen kaikkietävyys* (logical omniscience).

lukur10.tex -- 17/11/2005 -- 16:35 -- p.29/40

PÄÄTTELY LUOKKIEN PERUSTEELLA

Hierarkkiset kategoriat ovat yleisiä tietämyksenesityksessä ja niiden esittämiseen ja niistä päättelyyn on kehitetty erityisiä menetelmiä.

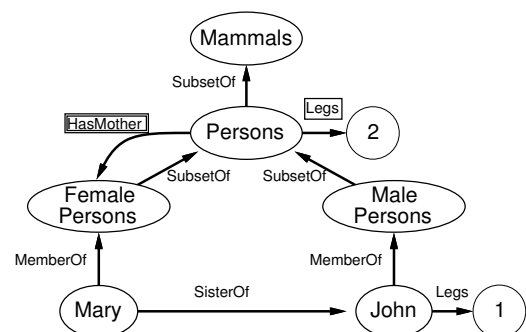
- Semanttiset verkot* (semantic networks) muodostavat graafisen esityksen luokille ja tarjoavat algoritmeja luokkaan kuuluminen perusteella tapahtuvaan päättelyyn.
- Kuvauslogiikat* (description logic) tarjoavat formaalin kielen luokkamäärittelyjen tekemiseen ja luokkien välisen sisältyvyyden esittämiseen.

lukur10.tex -- 17/11/2005 -- 16:35 -- p.30/40

SEMANTTISET VERKOT

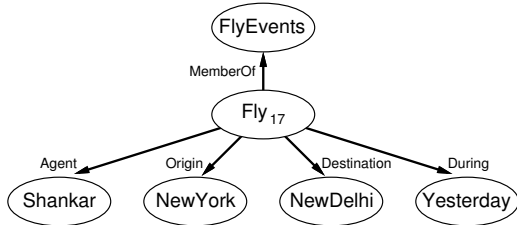
Semanttiset verkot (semantic networks):

- graafisia esityksiä yksilöille, luokille ja relaatioille.
- vaihtoehto logikalle vai logiikan muoto?
- periytymisen esittäminen luontevaa
- useita rajoituksia, joiden korjaaminen on mahdollista mutta monimutkaista, esim. vain binaariset relaatiot, loogiset operaatiot puuttuvat (disjunktio, negatio, eksistenssi)
- oletusarvojen esittäminen luontevampaa kuin logiikassa: alempien luokkien tai yksilöiden arvot kumoavat periytyvät arvot



lukur10.tex -- 17/11/2005 -- 16:35 -- p.31/40

lukur10.tex -- 17/11/2005 -- 16:35 -- p.32/40



lukur10.tex -- 17/11/2005 -- 16:35 -- p.33/40

Kuvauslogiikat (description logics):

- loogisia kieliä määritelmien ja luokkien välisten suhteiden esittämiseen
- tyypillisiä päättelytehtäviä *subsumptio* (luokkien välinen sisältyminen määritelmien perusteella) ja *luokittelu* (olioiden sijoittaminen luokkiin)
- tyypillinen esimerkki CLASSIC, jossa poikamies määritellään seuraavasti: $Bachelor = \text{And}(\text{Unmarried}, \text{Adult}, \text{Male})$
- tyypillisesti predikaattilogiikkaa heikompia ilmaisuvoimaltaan (esim. negaatio ja disjunktio voivat puuttua) mutta joidenkin asioiden esittäminen on helpompaa ja päättely on yleensä tehokasta

lukur10.tex -- 17/11/2005 -- 16:35 -- p.34/40

OLETUSARVOINEN PÄÄTTELY

- Kuinka monta tietojenkäsittelytieteen kurssia on tarjolla seuraavan ilmoitustaululta löytyvän informaation perusteella: "The following courses will be offered: CS 101, CS 102, CS 106, EE 101"?
- Ihmisten ja tietokantojen mukaan neljä, logiikan mukaan jotakin yhden ja äärettömän välillä.
- Erona oletus nimien yksilöllisyydestä (unique names assumption) ja *suljetun maailman oletus* (closed world assumption), jonka mukaan kannassa oleva informaatio on täydellistä ja atomilauseet joita ei esiinny kannassa, oletetaan epätosi.
- $\text{Course}(d, n) \leftrightarrow [d, n] = [CS, 101] \vee [d, n] = [CS, 102] \vee [d, n] = [CS, 106] \vee [d, n] = [EE, 101]$

lukur10.tex -- 17/11/2005 -- 16:35 -- p.35/40

PREDIKAATTIEN TÄYDENTÄMINEN

Tällaisia täydennettyjä predikaatteja (completion) voidaan muodostaa tietämyksen Hornin lauseista seuraavasti:

1. Kokoa kaikki lauseet muotoa $P(t_1, \dots, t_n) \leftarrow \text{Runko}$.
2. Muuta lauseet *Clarkin normaalimuotoon* (Clark normal form): $P(v_1, \dots, v_n) \leftarrow \exists w_1, \dots, w_m [v_1, \dots, v_n] = [t_1, \dots, t_n] \wedge \text{Runko}$.
3. Yhdistä sitten kaikki muotoa $P(t_1, \dots, t_n) \leftarrow B_i$ olevat lauseet muotoon $P(t_1, \dots, t_n) \leftarrow B_1 \vee \dots \vee B_k$.
4. Täydennä ekvivalenssiksi korvaamalla implikaatio ekvivalenssilla (suljetun maailman oletus): $P(t_1, \dots, t_n) \leftrightarrow B_1 \vee \dots \vee B_k$.

lukur10.tex -- 17/11/2005 -- 16:35 -- p.36/40

EPÄMONOTONISET LOGIIKAT

- Tavallinen logiikka on *monotonista*: uuden tiedon lisääminen ei voi kumota aiempia totuuksia.
- Jos $KB \models \alpha$, niin $KB \wedge \beta \models \alpha$.
- Jos on tehty suljetun maailman oletus, niin uusi informaatio voi olla ristiriidassa aiempien päätelmien kanssa. Siksi on ehdotettu *epämonotonisia logiikoita*, joissa aiemmin saatuja seurauslauseita voidaan hylätä. Tunnetuimmat epämonotoniset logiikat ovat
 - McCarthy'n *circumscription* ja
 - Reiterin *oletuslogiikka* (default logic).

lukur10.tex -- 17/11/2005 -- 16:35 -- p.37/40

CIRCUMSCRIPTION

- Taustalla suljetun maailman oletus: alunperin epätodeksi oletetut erikoispredikaatit, jotka ilmaisevat olion olevan epätyypillinen tapaus.
- Esim. $\text{Bird}(x) \wedge \neg \text{Abnormal}_1(x) \rightarrow \text{Flies}(x)$. Tämän perusteella Tipi-lintu lentää, ellei sitten saada jostain tietoa siitä, että Tipi on epätavallinen lintu.
- Mallien suosimisen (model preference) idea: lause on (oletusarvoisesti) tosi, jos se on tosi preferoiduissa malleissa joissa on vähiten epänormaaleja olioita.
- Ratkaisee mm. moniperinnästä johtuvia ongelmia:
 - $\text{Republican}(\text{Nixon}) \wedge \text{Quaker}(\text{Nixon})$
 - $\text{Republican}(x) \wedge \neg \text{Abnormal}_2(x) \rightarrow \neg \text{Pacifist}(x)$
 - $\text{Quaker}(x) \wedge \neg \text{Abnormal}_3(x) \rightarrow \text{Pacifist}(x)$.
- Toisessa preferoidussa mallissa Nixon on pasifisti ja toisessa ei ole joten Nixonin pasifismista ei tehdä päätelmiä.
- Joskus *priorisoidaan* ja sallitaan jompi kumpi päätelmistä.

lukur10.tex -- 17/11/2005 -- 16:35 -- p.38/40

OLETUSLOGIIKKA

- *Oletussäännöillä* (default rules) voidaan päätellä kontingentteja johtopäätöksiä, jotka saatetaan joskus kumota.
- Esim. $\text{Bird}(x) : \text{Flies}(x) / \text{Flies}(x)$ tarkoittaa, että jos x on lintu eikä ole yhteensopimatonta KB:n kanssa, että x osaa lentää, voidaan päätellä, että x osaa lentää.
- Yleisesti säännöt muotoa $P : J_1, \dots, J_m / C$.
- Nixon-esimerkki oletuslogiikassa:
 - $\text{Republican}(\text{Nixon}) \wedge \text{Quaker}(\text{Nixon})$.
 - $\text{Republican}(x) : \neg \text{Pacifist}(x) / \neg \text{Pacifist}(x)$.
 - $\text{Quaker}(x) : \text{Pacifist}(x) / \text{Pacifist}(x)$.
- Oletusteorian *laajennus* (extension) on maksimaalinen seurausten joukko s.e. oletussäännöillä saadut seuraukset ovat keskenään konsistentteja. Esimerkissä on kaksi vaihtoehtoista laajennusta joten oletuspäätelmiltä pidättäydytään ellei niille ole tehty priorisointia.

lukur10.tex -- 17/11/2005 -- 16:35 -- p.39/40

USKOMUSJÄRJESTELMÄT

- Jos tietämyskantaan hyväksytään oletusarvoisia lauseita, jotka voivat osoittautua myöhemmin epätosi, puhutaankin usein mieluummin *uskomusjärjestelmästä*.
- *Uskomusten korjaamisella* (belief revision) tarkoitetaan vanhoista uskomuksista luopumista, kun uusi tieto on niiden kanssa ristiriidassa. Jos KB esimerkiksi sisältää lauseen P ja sille kerrotaan $\text{TELL}(KB, \neg P)$, niin ensin täytyy suorittaa $\text{RETRACT}(KB, P)$. Mutta jos P :tä on käytetty premissinä päätelmissä, täytyisi luopua myös sen avulla saaduista johtopäätöksistä – paitsi jos nämä johtopäätökset voidaan oikeuttaa myös jollain toisella päätelmällä joka ei riipu P :stä.
- Uskomusjärjestelmiä, jotka käsittelevät tällaisia korjausoperaatioita ja pitävät yllä uskomusten välisiä oikeutussuhteita, kutsutaan nimellä *truth maintenance systems* (TMS).
- Vanhoja uskomuksia voidaan korjata monella tavalla: valintaa ohjaavat *konservatiivisuus* ja *koherenssi*.

lukur10.tex -- 17/11/2005 -- 16:35 -- p.40/40