Ohjelmiston vaatimusmäärittely

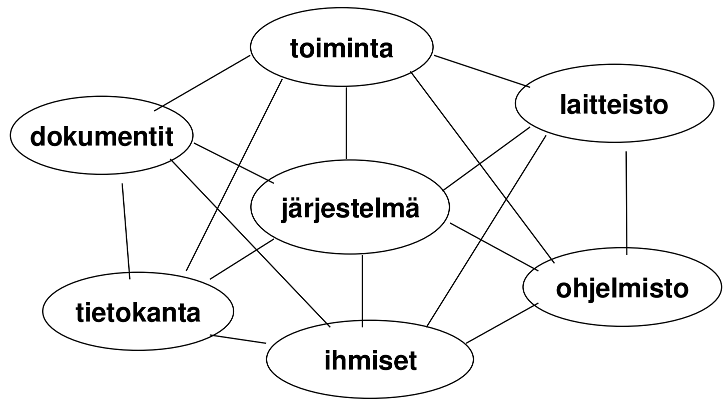
- Määrittelyprosessi
  - tehtävät, kokoustekniikkaa
- Määrittelydokumentti
- Formaalit menetelmät
- Kuvaustekniikat
  - perinteiset tekniikat, oliotekniikat
- OHTU-projektien määrittelydokumentti

Systeemianalyysi

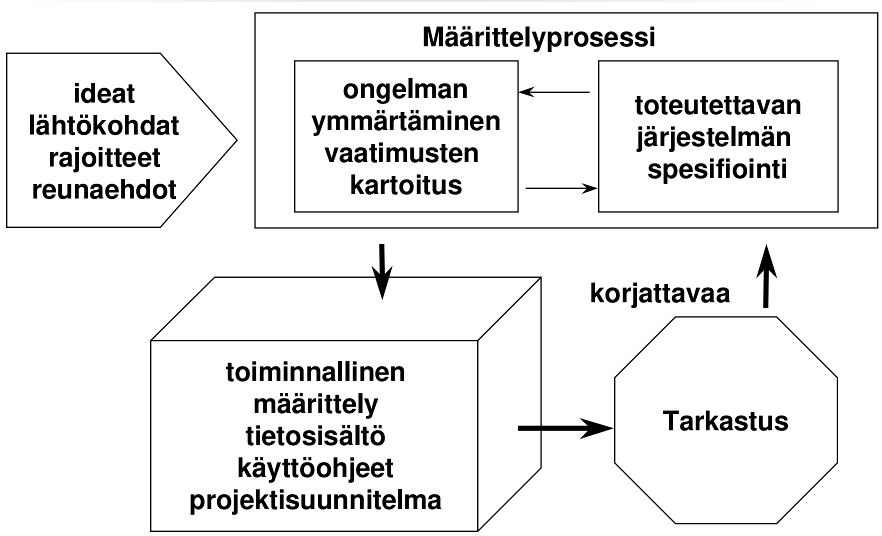
- System engineering ->proj.s.; vaatim.m
- Systeemianalyysi
  - asiakkaan tarpeiden tunnistaminen (käyttö- ja käyttäjänäkökulma)
  - järjestelmän toteutettavuuden arviointi (talous, tekniset seikat)
  - jako laitteistolle,ohjelmistolle, ihmisille
  - järjestelmäkuvaus
  - aika- ja resurssirajoitukset

Ohjelmiston vaatimusmäärittely

- tietoteknisen järjestelmän osat



Määrittelyprosessi



Määrittelyprosessi

- Päävaiheet
  - ongelman analysointi
  - vaatimusten kartoitus
  - järjestelmän määrittely

Määrittelyprosessi - päävaiheet

- ongelman analysointi
  - esitutkimus
  - kannattaako tehdä
  - arvioidut tuotantokustannukset < arvioitu tuotto
  - tarvittava toteutus ja käyttötekniikka saatavilla?
  - onko oikeudellisia tai eettisiä ongelmia
  - potentiaalinen asiakas-/käyttäjäkunta

Määrittelyprosessi - päävaiheet

- järjestelmälle asetettujen vaatimusten kartoitus
  - asiakkaan tarpeet
  - mikä ongelma on tarkoitus ratkaista
  - käyttäjälle tarjottavat palvelut
  - MITÄ …?
  - asiakas ja tuottaja yhteistyössä

Määrittelyprosessi - päävaiheet

- järjestelmän määrittely
  - **MITÄ …?**
  - tuloksena määrittelydokumentti:
    - **sopimus asiakkaan ja tuottajan välillä**
    - **perusta myöhemmille vaiheille**
    - **asiakkaan hyväksyttävä**

Määrittelyprosessi: kokoustekniikkaa

- Kokoustyypit
  - ongelman analysointi = vaatimusten kartoitus
  - määrittelydokumentin laatiminen
  - määrittelyn tarkastus = katselmus
- FAST
  - facilitated application specification techniques, kevennetyt määrittelytknt

Määrittelyprosessi - selvitettävää

- Ongelman analysointi/ vaatimusten kartoitus kokous
  - kuka on asiakas
  - onko käynnissä tarjouskilpailu
  - millaista hyötyä odotetaan
  - millaista tietoa pitäisi käsitellä
  - millainen käyttöympäristö
  - onko suorituskykyä mietitty

Määrittelyprosessi - selvitettävää

- Määrittelydokumentin laatimiskokous / kokoukset
  - lähtökohtana kartoituksen tulokset
  - tavoitteena asiakkaalle ja tuottajalle yhteinen näkemys järjestelmän
    - **yleisistä ominaisuuksista**
    - **pääkomponenteista**
    - **suorituskykyvaatimuksista ja**
    - **rajoitteista**

Määrittelyprosessi - selvitettävää

- Tarkastuskokous
  - täydellisyys
  - yhdenmukaisuus
  - ristiriidattomuus
  - vastaako asiakkaan näkemystä
  - ymmärrettävyys
  - onko tutkittu vaihtoehtoja
  - mitä riskejä liittyy

Määrittelyprosessi - selvitettävää

- tarkastus syytä toteuttaa määrämuotoisena katselmuksena
- jos tulosta ei hyväksytä palataan ed. vaiheeseen ja tehdään uusi tarkastus jos tulos OK, siirrytään seuraavaan vaiheeseen
- on varauduttava määrittelydokumenttiin kohdistuviin muutoksiin
- määrittelydokumentti ei ole **oikeaksi osoitettavissa (**ei voi todistaa, ei voi testata)

Määrittelydokumentti

- Vaatimusten oltava
  - virheettömiä (correct)
  - ristiriidattomia (consistent)
  - täydellisiä (complete)
  - realistisia (realistic)
  - tarpeellisia (needed)
  - todennettavissa (verifiable)
  - jäljitettävissä (tracable)
  - priorisoituja

Ohjelmiston vaatimusmäärittely

- Määrittelyprosessi
- ➨ **Määrittelydokumentti**
- Formaalit menetelmät
- Kuvaustekniikat
- OHTU-projektien määrittelydokumentti

Määrittelydokumentti (Heninger):

1. M:n pitää kuvata vain järjestelmän ulkoista käyttäytymistä.
2. M:n pitää sisältää toteutusta koskevat rajoitukset.
3. M:n pitää olla helposti muutettavissa.
4. M:n pitää soveltua ylläpitäjän työn tueksi.
5. M:n pitää kuvata järjestelmän ajateltu elinkaari.
6. M:n pitää kuvata hyväksyttävät reaktiot odottamattomiin(kin) tilanteisiin.

Määrittelydokun suositusrakenne

- dokumentin lukujen otsakkeet on syytä mukauttaa käytettävän menetelmän käsitteistöön
- on olemassa myös muita perusteltuja tapoja jäsentää määrittelydokumentti
  - ANSI-standardi 830-1984: IEEE Guide to Software Requirements Specification - *määrittelee useita vaihtoehtorakenteita (&hieman seuraavasta poikkeavan yleisrakenteenkin)*

Määrittelydokumentti: Johdanto

- 1.1. Tuotteen tausta ja tarkoitus
  - vrt. Projektisuunnitelma!
- 1.2. Mahdollinen erikoissanasto ja käytetyt lyhenteet
- 1.3 Yleiskatsaus dokumenttiin

2. Nykyjärjestelmän kuvaus

- *[Mikäli tuotettava ohjelmisto perustuu johonkin olemassaolevaan järjestelmään, niin:*
  - 2. Nykyjärjestelmän kuvaus + viitteet sen dokumentaatioon

  ]
- TAI

2. Yleiskuvaus

- 2.1. Yleinen toiminta ("mitä järjestelmän pitäisi tehdä?")
- 2.2. Tuotteen pääasiallinen toimintaympäristö
- 2.3. Tuotteen käyttäjäkunta
- 2.4. Katsaus muihin vastaaviin järjestelmiin

3. Tietokuvaus

- 3.1. Tietosisältö
- 3.2. Tietokantamalli
- 3.3. Kapasiteetti- ja saantiaikavaatimukset

- tekniikoita: luokkakaavio, elinkaarikuvaukset, ER-kaavio, …

4. Toimintokuvaus

- 4.1. Järjestelmän yleisarkkitehtuuri
  - toiminnalliset pääkomponentit
  - pääkomponenttien väliset suhteet ja rajapinnat
- Tekniikoita:
  - osajärjestelmäkaavio,
  - rajaus suhteessa ympäristöön,
  - komponenttien yhteistyötä kuvaavat kaaviot
  - käyttötapausmalli,…

4.2.x Komponentti X

- **yleiskuvaus**
  - **syötteet, tulosteet**
  - **toiminta eli tiedon käsittely**
  - **Komponentti voi olla**
    - **osajärjestelmä**
    - **toiminto tai palvelukokonaisuus**
    - **käyttötapaus**
- **Tekniikoita**: käyttötapaukset, skenaariot, tietovuokaaviot, päätöstaulut, pseudokoodi, …

5. Järjestelmän ulkoiset liittymät

- sekä järjestelmän käyttämät että sitä käyttävät muut järjestelmät
- 5.1. Käyttöliittymä
- 5.2. Laitteistoliittymät
- 5.3. Ohjelmistoliittymät
- 5.4. Tietoliikenneliittymät
- 5.5. Alustustiedot

Käyttöliittymän kuvaaminen

- mahdollisen prototyypin kuvaus
- kuvaus tyypillisistä käyttöskenaarioista = käyttötapauksen toteutuma
- kuvia hahmotellusta käyttöliittymästä (prototyypistä saatuina tai piirrettyinä)

6. Muut ominaisuudet

- 6.1. Suorituskyky
- 6.2. Käytettävyys, virheistä toipuminen, turvallisuus, suojaukset, varmistuskopiointi
- 6.3. Ylläpidettävyys (ohjelmointityyliohjeet yms.)

7. Testaus

- 7. Testaus
  - kuinka kattavasti on testattava?
  - millaisella aineistolla on testattava?

8. Rajoitteet suunnittelulle ja toteutukselle

- 8.1. Noudatettavat standardit
- 8.2. Laitteistorajoitteet
- 8.3. Ohjelmistorajoitteet
  - (ohjelmointikielet, käyttöjärjestelmät, käyttöliittymät,...)

Muut osat

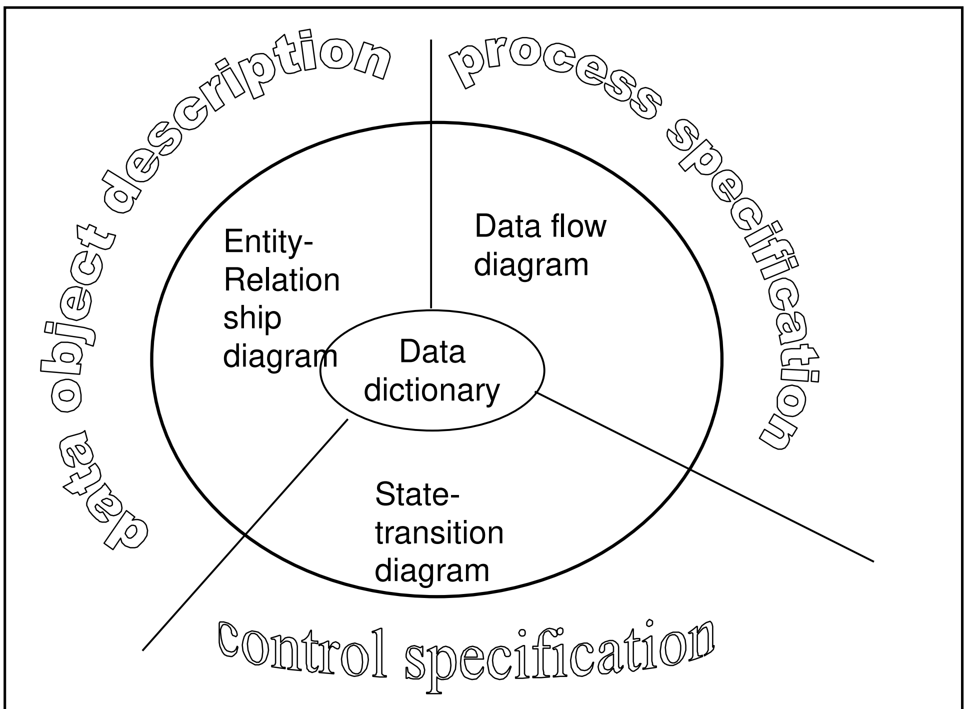
- Lähdeluettelo
- Liitteet:
  - kaavioissa käytettyjen symbolien kuvaus, ellei standardinmukainen
  - syötteiden rakenne
  - tulosteiden rakenne
  - formaalit spesifiointikaavat

Ohjelmiston vaatimusmäärittely

- Määrittelyprosessi
  - tehtävät, kokoustekniikkaa
- Määrittelydokumentti
- Formaalit menetelmät
- Kuvaustekniikat
  - perinteiset tekniikat, oliotekniikat
- OHTU-projektien määrittelydokumentti

Määrittelyn keskeinen tehtävä

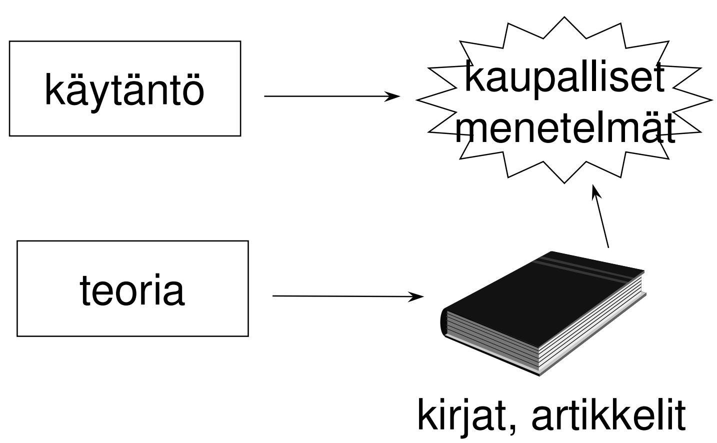
- järjestelmää kuvaavan käsitteellisen mallin (conceptual model) muodostaminen:
  - yksinkertaistettu malli käyttäjien ja suunnittelijoiden kommunikoinnin perustaksi
  - täsmällisessä muodossa paperilla / seinätaululla
  - toimintoja (function)
  - dataa
  - ohjausta (control)



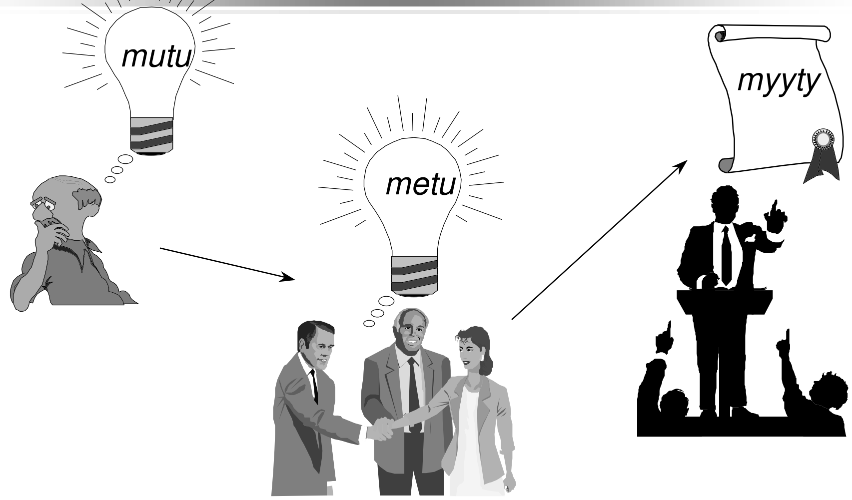
Määrittely - menetelmät

- perusajatus siitä, mikä mallintamisessa on oleellista, on eri menetelmissä erilainen
  - kukin menetelmä vaatii käytettävien käsitteiden, symbolien ja merkintätapojen opettelemisen
  - menetelmät eivät sovellu mihin tahansa tilanteeseen
  - graafisten kaavioiden ylläpito voi olla hankalaa (automatisointi?)
  - suurien kokonaisuuksien hahmottaminen kun yhteen kuvaan ei mahdu kovin paljon

Määrittely - menetelmien synty



Määrittely - menetelmien synty

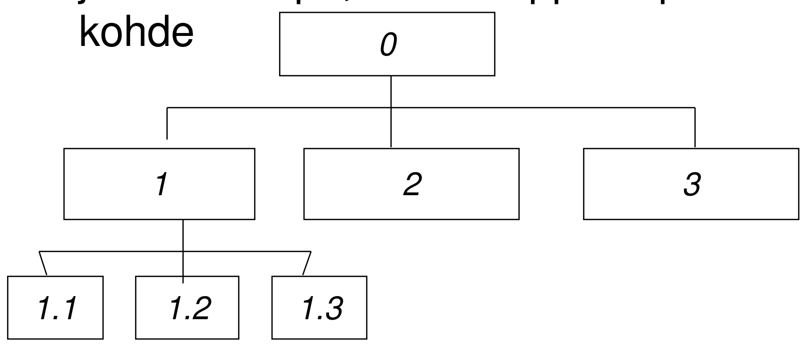


Menetelmien yhteisiä piirteitä

- *Tarvitaan välineet*
  - 1. tiedon analysointiin
  - 2. toimintojen esittämiseen
  - 3. liittymien kuvaamiseen
  - 4. ongelman osittamiseen
  - 5. abstrahoinnin tukemiseen

Ongelman osiinjako - ositus

- Muodostuu jakohierarkia
  - kullakin tasolla sama tarkastelukulma ja kuvaustapa, mutta suppeampi kohde



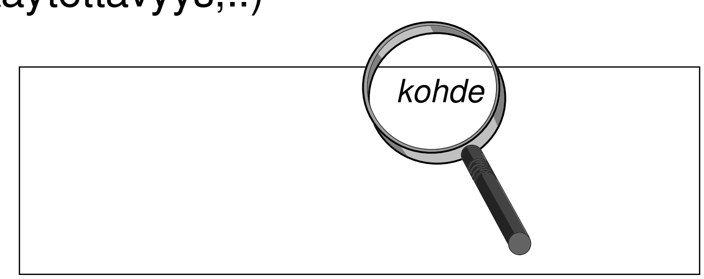
Järjestelmien osiinjako - abstrahointi

Järjestelmien osiinjako - abstrahointi

- tarkastellaan aluksi karkeammalla tasolla
- lisätään yksityiskohtaisuutta siirryttäessä seuraavalle tasolle
  - esim.
    - **käsitetaso**
    - **rakennetaso**
    - **fyysinen taso**
  - kuvaustapa tasoilla voi vaihdella

Järjestelmien osiinjako - projisointi

- Projisointi
  - järjestelmää tarkastellaan jostain tietystä näkökulmasta (esim. suojaus, käytettävyys,..)

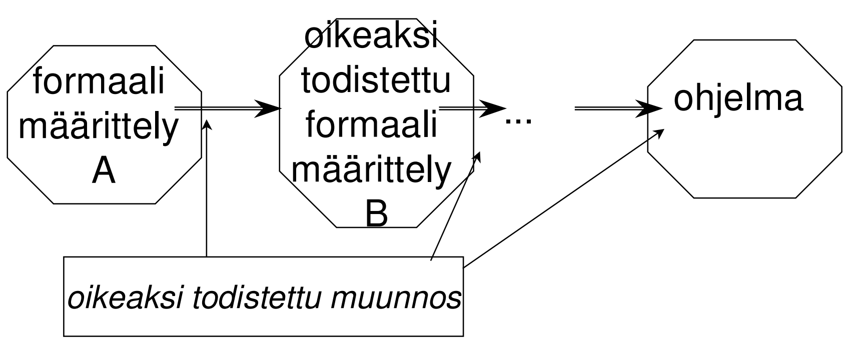


Ohjelmiston vaatimusmäärittely

- Määrittelyprosessi
- Määrittelydokumentti
- ➨ **Formaalit menetelmät**
- Kuvaustekniikat
  - perinteiset tekniikat
  - oliotekniikat

Määrittely - Formaalit menetelmät

- Formaali näkemys
  - ohjelmisto tuotetaan jonona asteittain tarkentuvia spesifikaatioita



Määrittely - Formaalit menetelmät

- formaali määrittely = ohjelmiston ominaisuudet määrittelevä matemaattinen kaava
- Esim: lajittelu

  lajiteltu([x,L]) = ($\forall$ *y*$\in$*L: x*$\leq$*y) and lajiteltu(L).*

  lajiteltu([]).

Formaalit menetelmät - plussia

- **täsmällisyys**
- **tuotettu ohjelmisto vastaa alkuperäistä määrittelyä, eli tekee sen, mitä vaaditaan**
- **automatisointi: spesifikaatiota voidaan analysoida ja havaita sisäiset loogiset virheet**
- **spesifikaatiota voi osittain suorittaa jo ketjun alussa**
- **abstraktiotason voi säätää sopivaksi**
- **erikoiskieliä (VDM,Z, LOTOS, Prolog,…)**

Formaalit menetelmät - miikkoja

- miten varmistetaan, että ketjun 1. spesifikaatio on “oikein” (suhteessa epäformaaliin asiakkaaseen)?
- miten todistetaan, että muunnosten oikeaksitodistukset ovat oikein?
- entä oikeaksitodistuksen oikeaksitodistukset, ...

Formaalit menetelmät - miikkoja

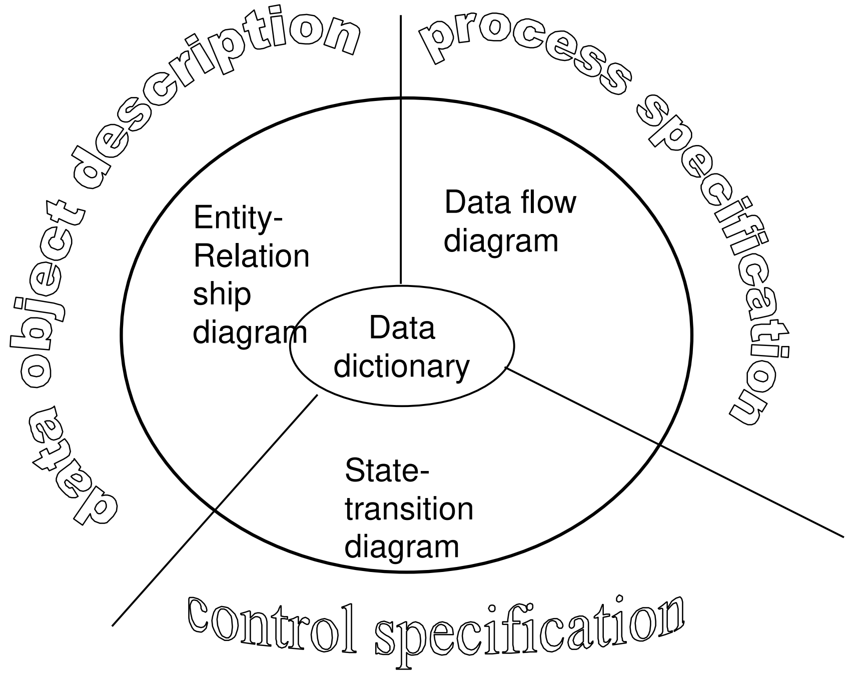
- **prosessin raskaus**
  - **lyhyt ketju -> suuria semanttisia välejä spesifikaatioiden välillä -> todistaminen vaikeaa**
  - **pitkä ketju -> paljon spesifikaatioita, muunnoksia ja todistuksia**
- **epähavainnollisuus (huono kommunikaatioväline, notaatio?)**
- **teollisuudelta puuttuva formaali osaminen**
- **vähän vakuuttavia näyttöjä**
  - **käytössä erityisaloilla: tietoliikenne, kääntäjät**

Ohjelmiston vaatimusmäärittely

- Määrittelyprosessi
- Määrittelydokumentti
- Formaalit menetelmät
- ➨ **Kuvaustekniikat**
  - perinteiset tekniikat
  - oliotekniikat

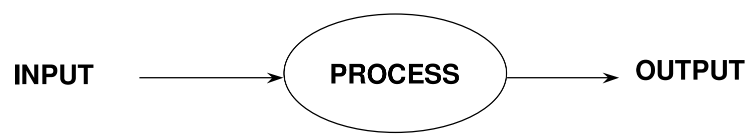
Lähestymistapoja mallintamiseen

- toimintokeskeinen
  - **syötteistä tulosteita muokkaava systeemi**
- tietokeskeinen
  - **kohdealueeseen liittyvää tietämystä ylläpitävä ja tarjoava systeemi**
- oliokeskeinen
  - **olioiden joukkio, joka yhteistyössä toimien tuottaa halutut palvelut (vrt. organisaatioyksikkö)**



Toiminnallinen lähestymistapa

- Systeemiteoreettinen lähestymistapa



- **systeemi on prosessi, joka saa syötteitä ja tuottaa tuloksia**
- **systeemi voidaa jakaa osasysteemeihin**
- **tietojärjestelmissä syötteet ja tuotteet ovat tietoa**
- **prosessit muokkaavat tietoa**

Toiminnallinen lähestymistapa ...

- tiedot kulkevat järjestelmän läpi jalostuakseen alkuperäisistä syötteistä lopullisiksi tulosteiksi
- yleisin kuvaustekniikka on **tietovuokaavio** *(data flow diagram, DFD*
  - HIPO-kaavioiden (Hierarchy-Input-Process-Output) sisältö on likimain sama kuin tietovuokaavioissa

Tietovuokaaviot (tietovirtakaaviot)

- DeMarco & Yourdon 1979, Gane & Sarson 1979
- Kuvauksen sisältö
  - toiminnan hierarkkinen osiinjako
  - tiedon kulku toimintojen välillä
  - tiedon säilytys
  - tiedon käyttäjät ja tuottajat

Tietovuokaaviot - peruskäsitteet

- PROSESSI (process)
  - toiminto = tehtäväkokonaisuus,
  - kuvaa tekemistä (tietoa muokataan jollain tavoin),
  - prosessi saa syötteitä ja tuottaa tulosteita.

Tietovuokaaviot - peruskäsitteet

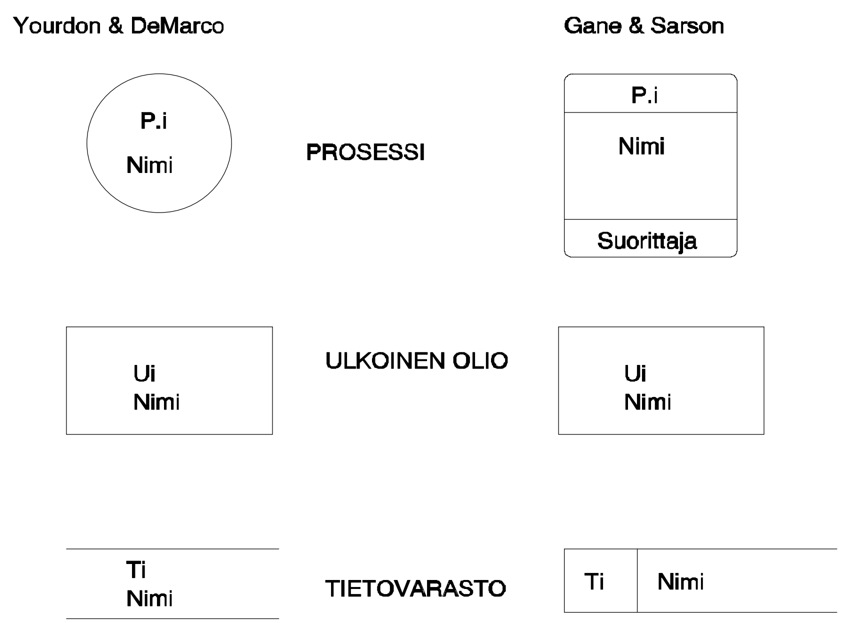
- TIETOVARASTO (data store)
  - kuvaa tiedon tilapäistä tai pitkäaikaista säilytystä,
  - saa tietoa yhdeltä tai useammalta prosessilta,
  - prosessit voivat hakea tietoa tietovarastosta.

Tietovuokaaviot - peruskäsitteet

- ULKOINEN OLIO (external entity)
  - järjestelmän ulkopuolelle rajattu järjestelmä, käyttäjä tai muu olio, joka joko saa systeemiltä tietoa ja/tai antaa systeemille tietoja,
  - suunnittelija ei voi vaikuttaa ulkoisten olioiden toimintaan,
  - joskus erotellaan ulkoiset oliot tiedontuottajiksi (source) ja tiedon hyväksikäyttäjiksi (sink).

Tietovuokaaviot -peruskäsitteet

- TIETOVUO (tietovirta) (data flow)
  - kuvaa suoraa tiedonkulkua prosessin ja toisen prosessin / tietovaraston / ulkoisen olion välillä,
  - tietovuon toisena osapuolena on aina prosessi,
  - tietovuolla on suunta.



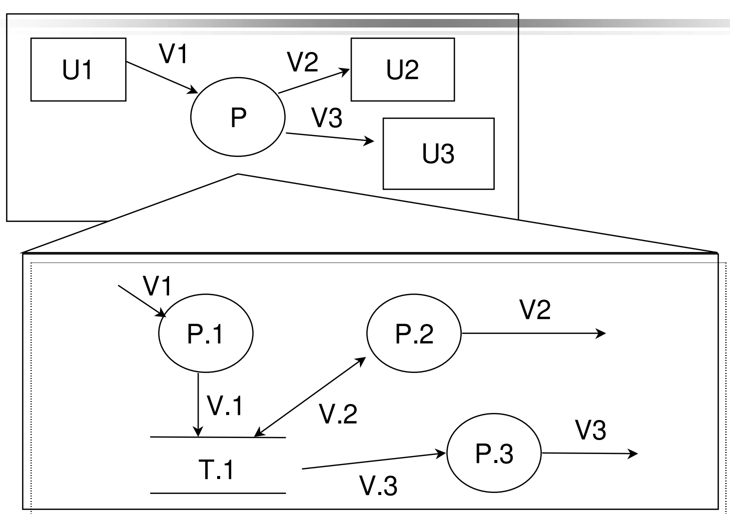
Tietovuokaaviot - peruskäsitteet

- mallintaminen perustuu toiminnan hierarkkiseen tarkentamiseen:
  - prosessi kuvataan jakamalla se osaprosesseihin, toistuvasti
  - ylimmän tason kaaviossa (<u>yhteyskaavio</u>, taso 0, context diagram) esitetään järjestelmän yhteydet ympäristöön,

Yhteyskaavio

- järjestelmä esitetään yhtenä prosessina
- kaikki sidosryhmät mukana kaaviossa
- kaikki sidosryhmille menevät ja niiltä saatavat tietovuot mukana, mutta ei välttämättä erillisinä vaan sopivasti ryhmiteltyinä.
- Tärkeimmät tietovuot mukaan erillisinä.

Tietovuokaaviot - yhteyskaavio



Tietovuokaaviot - peruskäsitteet

- Tarkentaminen
  - jaetaan tason *i* kaaviossa oleva prosessi osaprosesseihin. Osat ja niiden väliset tietovuot sekä jaetun prosessin sisäiset tietovarastot esitetään tason *i+1* alikaaviossa.

Yleiskaavio

- **yleiskaaviossa (taso 1) näkyvät järjestelmän keskeiset osasysteemit sekä tiedon kulku niiden välillä**
  - tietovarastot esitetään kaaviossa vain jos kaksi tai useampia prosesseja käyttää niitä
  - Jos prosessilla ei ole tarkennusta otetaan kaavioon myös prosessin sisäiset tietovarastot
- alikaavio yhdelle A4 arkille=> kaaviossa alle 10 toimintoa

Tietovuokaaviot - peruskäsitteet

- Tarkennusta jatketaan kunnes päästään niin pieniin osatoimintoihin, että ne on täsmällisesti kuvattavissa vajaa sivun 'tekstikuvauksella':
  - luonnollista kieltä,
  - pseudokieltä,
  - päätöspuita,
  - päätöstauluja
  - tila-automaatteja

- Alimman tason prosessien tekstikuvauksessa kuvataan
  - mitä prosessissa tehdään
  - prosessiin liittyvät säännöt ja vaatimukset
  - suorittajatietoja
  - ajoitustietoja yms.

Tietovuokaaviot - peruskäsitteet

- tietosanasto (*data dictionary)* , jossa tiedot kuvataan
  - luettelona tai määrittelykielellä
- *käsiteanalyysi*
  - kunkin tietovuon ja tietovaraston perusteella laaditaan käyttäjänäkemys sen sisältämistä tiedoista

Tietovuokaaviot - laajennos

- tietovuokaavion kuvaesityksessä ei perusmuodossa esitetä toimintojen aikariippuvuutta eikä toimintaan liittyvää kontrollia, näidenkin kuvaamista varten on kehitetty muunnelma (Ward & Mellor - reaaliaikalaajennos)
  - kontrolliprosessit
  - kontrollivuot (-signaalit)

Tietovuokaavioiden laatiminen- 1

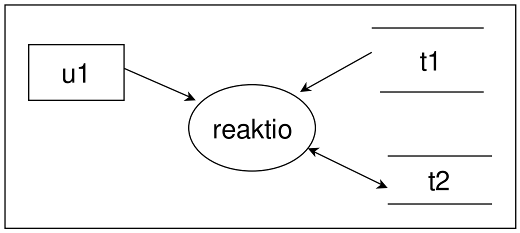
- **hierarkkinen ositus (functional decomposition):**
  - aloitetaan yleiskaaviosta ja edetään 'top-down' -tyylillä toimintaa osittaen kunnes saavutetaan riittävän pienet prosessit
    - kuvaus riittävän tarkka esim. työmäärä-arvioiden tekemiseksi vasta muutaman tason jälkeen

Tietovuokaavioiden laatiminen -2

- **tapahtumaperustainen koostaminen (event partitioning) / uudempi tapa**
  - rajataan järjestelmä yhteyskaavion avulla
  - luetellaan kaikki järjestelmän piiriin kuuluvat **tapahtumat**
    - tapahtuma on asia, johon pitäisi reagoida (esim. tilauksen teko, raportin pyytäminen)
  - tapahtumalle **reaktioprosessi**
    - laaditaan kaavio prosessin tietotarpeista

Tietovuokaavioiden laatiminen -2

Reaktioprosessi



- reaktioprosessin kaaviossa esiintyvät tietovarastot ovat käsitemallin yksilötyyppejä tai niiden kokoelmia
- käsitemallia tuotetaan samanaikaisesti - malli esittää myös yksilötyyppien välisiä yhteyksiä

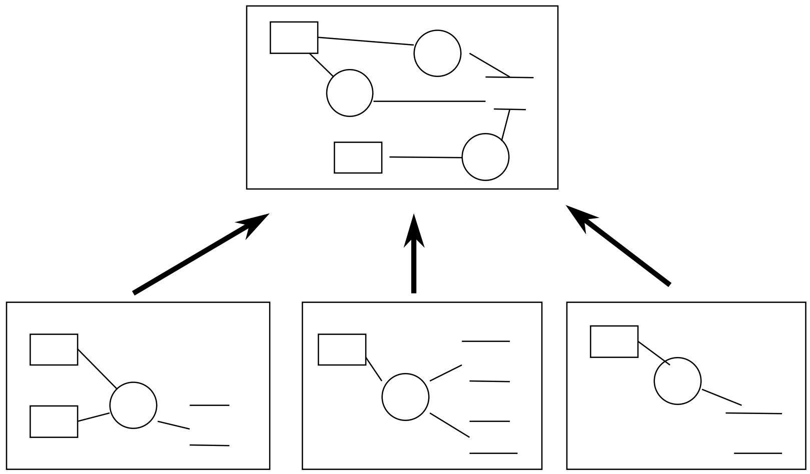
Tietovuokaavioiden laatiminen -2

Prosessihierarkkia

- kun reaktioprosessit on kuvattu, rakennetaan prosessihierakia
  - kootaan yhteen yhteenkuuluvat prosessit ja määritellään kokoelma ylemmän tason prosessiksi
  - yhteenkuuluvuus (vaihtoehtoja):
    - samat ulkoiset oliot
    - tietovarasto piiloon ylemmän prosessin sisään
    - prosessien välisen tiedonsiirron minimointi

Tietovuokaavioiden laatiminen -2

Prosessihierarkkia



Tietovuokaavioiden laatiminen -2

Prosessihierarkkia

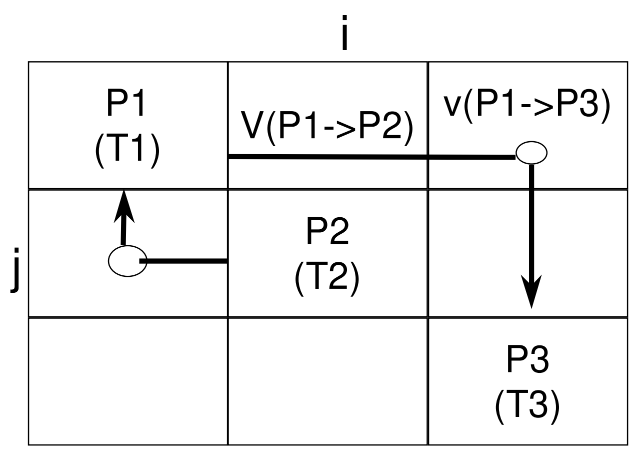
- Hierarkkisessa koostamisen bottom-up kokoamisessa ei synny prosessien välisiä suoria tietovirtoja, vaan tieto välitetään prosessista toiseen tietovarastojen kautta (vrt. nykyaikainen tietokantajärjestelmä ja tapahtumakäsittelyprosessit)

Toimintomatriisi

- Tietovirtojen esitys kaaviona on vain eräs esitystapa
- Toimintomatriisia voi käyttää vaihtoehtona

prosessi P1 (sisäiset tieto varastot T1)		
	P2 (T2)	
		P3 (T3)

ruudussa (I,j) I=j: prosessilta Pi prosessille Pj välitettävät tietovirrat / tietovarastot



Ohjelmiston vaatimusmäärittely

- Kuvaustekniikat
  - tietovuokaaviot
    - hierarkkinen tarkentaminen
    - reaktioprosesseista kokoaminen
  - toimintomatriisi
  - tietokeskeinen menetelmä
    - esim. ER-mallit (HYTKY)

Tietokeskeinen menetelmä

- lähtökohtana järj. tietosisällön määrittely
  - **tiedon rakenne ja riippuvuudet ovat ‘pysyvämpiä’ kuin tietoon kohdistuvat toimenpiteet**
  - **toiminnot hyödyntävät tietoa, toimintoja voidaan kytkeä myöhemmin**
- tietokantajärjestelmiin
  - tietojen väliset yhteydet/riippuvuudet keskeisiä

Lähestymistapa - tietokeskeinen

- keskeiset ongelmat eivät liity toimintoihin, vaan tiedon hallintaan ja jäsentämiseen
- käsiteanalyysissa (data modelling, conceptual modelling) määritellään järjestelmän keskeiset tieto-objektit *(entities, items, objects)* ja niiden väliset riippuvuudet,
- toteutustavasta riippumaton kuvaus

Lähestymistapa - tietokeskeinen

- käsiteanalyysin kuvaustekniikkoista yleisimpiä ER (Entity - Relationship) -malliperheen mallit, tai muut ns. semanttiset tietomallit ja oliomallit

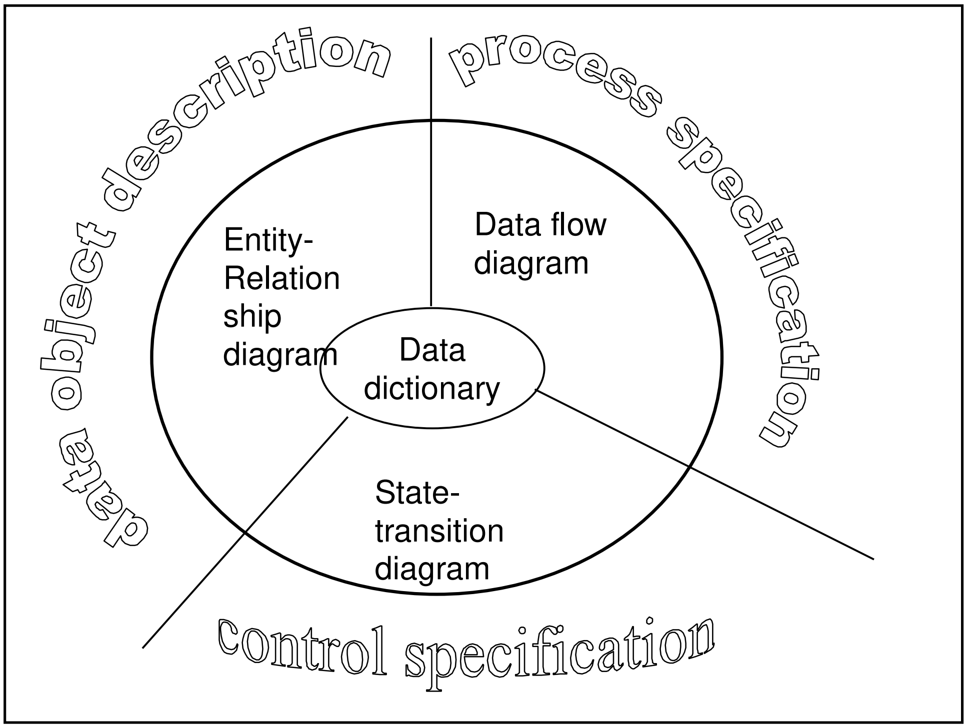
- käsitteitä:
  - **yksilötyypit (entity types)**
  - **yhteystyypit (relationship types)**
  - **ominaisuudet (attributes)**
  - **yleistyshierarkia (generalization)**
  - **rajoitteet (constraints)**

- graafisia esityksia Chen:in ER-tekniikka, Bachman diagrammit (esim HYTKY -työkalut), yms.
- kuvaesityksen lisäksi tarvitaan tekstimääritelmät

Lähestymistapa - tietokeskeinen

- analyysin tukiohjelmistoja tarjolla
  - tulosten perusteella voidaan tietokannan rakenne johtaa automaattisesti
- kuvaus ei ole hierarkkinen => isot kaaviot

- tietokeskeisen lähestymistavan yhteydessä käytetään usein toiminnan selvillesaamiseksi **tapahtuma-analyysia**
  - selvitetään mitkä tapahtumat luovat, muuttavat tai hävittävät tieto-objekteja
.



Oliot ja määrittely

- Mitä määrittelyvaiheessa on välttämättä kuvattava: järjestelmän toiminnalliset vaatimukset
  - tarvitaanko tähän olioita - EI
  - auttavatko oliot - VÄHÄN
    - olioita käytettäessä toiminta ja siihen liittyvät vastuut saadaan jaettua eri luokille => oliokäsitteestä olisi apua kuvaamaan esim. osajärjestelmiä

- olioiden yläpuolelle tarvitaan toiminnan kokonaiskuvaus
- jos ollaan oliokeskeisiä, pitäisikö koko järjestelmä nähdä yhtenä oliona, joka tarjoaa palveluja - KYLLÄ
  - käyttötapaukset (use case) ovat järjestelmän käyttäjilleen tarjoamien palvelujen määrityksiä

Oliot ja määrittely

- Mitä määrittelyvaiheessa on välttämättä kuvattava:yleiskuva järjestelmän tietosisällöstä
  - siinä laajuudessa, että paljelut tulevat ymmärretyiksi
  - voiko tässä käyttää olioita - KYLLÄ
    - keskeiset luokat ja niiden väliset yhteydet