

Ohjelmistoarkkitehtuurit

Arkkitehtuurin kuvaaminen

20.9.2012 1

Arkkitehtuurin kuvaaminen

- Ohjelmistoarkkitehtuurin kuvaaminen on arkkitehtuuritason suunnitteluratkaisujen kuvaamista
- Arkkitehtuuritasoisuus
 - Aihe, ongelma tai ohjelmistoelementti on arkkitehtuuritasolla, jos sillä on merkittävä vaikutus ohjelmiston rakenteeseen tai ohjelmiston laatuominaisuuksiin kuten suorituskykyyn, skaalautavuuteen, tietoturvaan, luotettavuuteen, muutettavuuteen, jne.
 - Osittain myös sopimusasia – riippuu järjestelmästä ja sovellus/teknologia-alueesta, mitä konkreettisia asioita (esim. ohjelmistorakenteita) kuvaukseen otetaan mukaan

20.9.2012 2

Arkkitehtuurin kuvaaminen

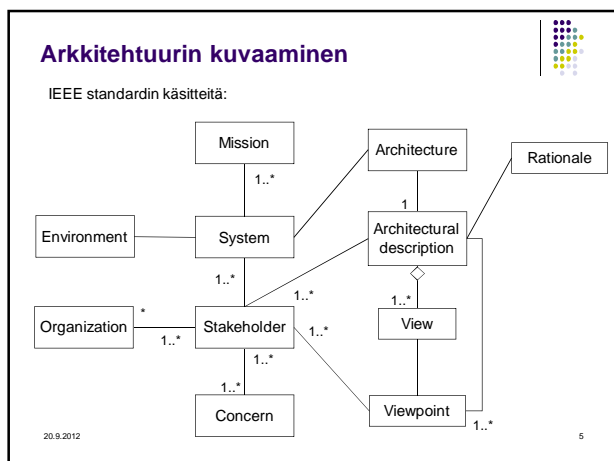
- Kuvaamisen tavoitteita
 - Kokonaisnäkemys järjestelmästä
 - Käsitteistö
 - Perustelut (engl. *rationale*)
- Ohjelmistotekniikan pitkän tähtäimen suuntauksena on ohjelmistojen kuvauksen abstraktiotason nostaminen ja toimivien järjestelmien tuottamien yhä korkeamman tason kuvauksista
 - Todennäköisesti mahdollista vain hyvin spesifeillä sovellusaluekohtaisilla kuvaustekniikoilla, sillä kuvausten on oltava kattavia ja riittävän yksityiskohtaisia

20.9.2012 3

Arkkitehtuurin kuvaaminen

- IEEE 1471 –standardi arkkitehtuurien kuvaamiseen
 - http://www.pithecanthropus.com/~awg/public_html/introducing-p1471.pdf (sivut 14-23)
 - Päivitetty versio standardista on **ISO/IEC/IEEE 42010**: <http://www.iso-architecture.org/ieee-1471>
 - Yhteenveto <http://www.iso-architecture.org/ieee-1471/ads>
 - Standardi saatavissa IEEE Xplore:sta (HY:n tunnukset)
 - Standardi määrittelee arkkitehtuurin kuvaamiseen liittyviä abstrakteja käsitteitä
 - ei esittele yhtäkään standardinäkemää, joka vaadittaisiin kaikissa kuvauksissa
 - Näkömät ja eri näkymissä käytettävät kuvaustekniikat, -kielet ja analyysimenetelmät hyvin sovellusaluekohtaisia
 - 1471-2000 toimii kuitenkin eräänlaisena käsitteellisenä kehyksenä konkreettisen kuvausstandardin laatimiselle
 - Lähtökohtana arkkitehtuurin "käyttäjien" tarpeet ja niiden huomioon ottaminen eri näkymissä

20.9.2012 4



Arkkitehtuurin kuvaaminen

IEEE standardin käsitteitä:

- Jokainen **järjestelmä** (system) täyttää yhden tai useampia **tehtäviä** (missions) **toimintaympäristössään** (environment)
- Järjestelmällä on **sidosryhmiä** (stakeholders), joilla on **konkreettinen panos** (vested interest) pelissä ja tiettyjä **kiinnostuksen kohteita** (concerns)
- Jos järjestelmällä on **arkkitehtuuri** (architecture), sillä on myös yksi **arkkitehtuurikuvaus** (architectural description)
- Arkkitehtuurikuvaus muodostuu **näkymistä** (view). Näkömä perustuu **näkökulmaan** (viewpoint)
 - Näkökulma: tietty esitysmuoto ja tavoite kuvaukselle. Näkömä: yksittäiselle järjestelmälle annettu osittainen arkkitehtuurikuvaus, joka noudattaa näkökulman esitysmuotoa.
- Toisaalta arkkitehtuurikuvaus koostuu **malleista** (esim. UML-kaaviot), joita näkömät hyödyntävät (eivät näy kuvassa). Näkömässä voi olla useita malleja ja sama malli voi olla useassa eri näkyssä.

20.9.2012 6

Arkkitehtuurin kuvaaminen

- Arkkitehtuurin elementtejä yleisellä
 - Komponentit
 - Konnektorit (connector)
 - Konfiguraatiot
 - Liittymät (interface)
 - Perustelut
- Perustuuko kuvaus näihin käsitteisiin vai joihinkin muihin spesifimpiin käsitteisiin?
 - Onko selvää miten näitä käsitteitä pitäisi soveltaa?
 - Onko näihin perustuva malli liian alhaisella abstraktiotasolla?
 - Onko tarjolla näihin perustuvia riittävän ilmaisuvoimaisia kuvauskieliä?
- Tyyli ja ratkaisumallit olisi saatava mukaan korkeamman tason rakenteina
- Pitäisi mallintaa tyyli tai ratkaisumalli yksityiskohtaisesti – kuitenkin kirjallisuudesta löytyy enimmäkseen vain vapaamuotoisia, luonnollisella kielellä laadittuja kuvauksia – osoittaako tämä mallien vajavaisuutta?

20.9.2012 7

Arkkitehtuurin kuvaaminen

- Näkökulma (viewpoint):** yleinen, järjestelmästä riippumaton tapa kuvata tiettyä arkkitehtuurin kannalta merkityksellistä ohjelmistojen ominaisuutta
- Näkymä (view):** varsinainen järjestelmästä riippuva kuvaus, joka noudattaa jotakin näkökulmaa
- Perusta eri näkymille: monimutkaisen järjestelmän kuvaaminen ei ole mahdollista yhdellä kaikenkattavalla kuvauksella, joka olisi ymmärrettävä ja hyödyllinen kaikille sidosryhmille.

20.9.2012 8

Arkkitehtuurin kuvaaminen

Järjestelmän **arkkitehtoninen malli**: kokoelma (arkkitehtuuri-)näkymiä

Järjestelmä

20.9.2012 9

Arkkitehtuurin kuvaaminen

- Näkymien tavoitteita**
 - Asiakokonaisuuksien erillään pitäminen
 - Kommunikointi sidosryhmien kanssa helpottuu
 - Monimutkaisuuden hallinta
 - Keskittyminen olennaiseen
- Näkymien ongelmia**
 - Ristiriitaiset kuvaukset
 - Suorat ristiriidat
 - Tarkennusristiriidat
 - Staatisten ja dynaamisten aspektien yhteensopimattomuus
 - Näkymäjoukon valinta
 - Kattavuusongelmat, mallien huono soveltuvuus
 - Pirstoutuminen

20.9.2012 10

Arkkitehtuurin kuvaaminen

- Mitä näkökulmia?
 - Useita malleja
 - Esikuvana Kruhtenin 4+1 malli:
 - Kruchten P.: *The 4+1 View Model of Architecture*, IEEE Software, Nov 1995, pp 42-50.

20.9.2012 11

Arkkitehtuurin kuvaaminen

Kruhtenin 4+1 malli:

Sidosryhmä
• kiinnostuksen aihe

Käyttäjät:
• toiminnallisuus

Ohjelmistojen integroijat:
• suorituskyky
• skaalautuvuus

Ohjelmoijat
• koodin hallinta

Systemisuunnittelija
• laitetopologia
• toimitus
• asennus
• tietoliikenne

20.9.2012 12

Arkkitehtuurin kuvaaminen – 4+1

- **Skenaariönäkymä** (scenario view)
 - Nitoo yhteen 4 muuta näkymää ("+1")
 - Järjestelmän vuorovaikutus käyttäjien ja muiden järjestelmien kanssa tietyssä käyttöskenaariossa
 - Kuvaa järjestelmän rajapinnat ympäristöönsä
 - Sovellettavissa useimpien järjestelmien kuvaamisessa
- **Looginen näkymä** (logical view)
 - Toiminnallisuuden jakautuminen eri rakenneosien kesken
 - Komponenttien velvollisuudet
 - Koskee sekä staattisia rakenteita (esimerkiksi luokkia) että dynaamisia rakenteita (esimerkiksi olioita)
 - Tärkeä näkymä kaikissa järjestelmissä

20.9.2012

13

Arkkitehtuurin kuvaaminen – 4+1

- **Prosessinäkymä** (process view)
 - Järjestelmän toiminnan jako loogisiin prosesseihin
 - Prosessien kommunikointi - kontrollin kulku
 - Rakenteet ja instanssit - samasta prosessirakenteesta voi olla useita instansseja
 - Tärkeä rinnakkaisuutta sisältävien järjestelmien kuvauksessa
 - Järjestelmän suorituskyvyn ja skaalautuvuuden arviointi
- **Kehitysnäkymä** (development view)
 - Kuvaa järjestelmän jakoa kehitysyksiköihin
 - *pakkaukset, ohjelmistotekninen rakenne*
 - Erityisesti laajojen järjestelmien kuvauksessa

20.9.2012

14

Arkkitehtuurin kuvaaminen – 4+1

- **Fyysinen näkymä** (physical view)
 - Järjestelmän prosessointiyksiköt ja niiden yhteydet
 - Prosessointiyksiköiden sisältämät fyysiset ohjelmistoartefaktit (komponentit, resurssitiedostot, tietokannat, ...)
 - Tärkeä hajautetuissa järjestelmissä

20.9.2012

15

Arkkitehtuurin kuvaaminen – 4+1

- 4+1 ei ole täydellinen → Monesti tarvitaan täydentäviä näkökulmia; esimerkiksi "**muuntelunäkymä**" (Koskimies & Mikkonen 2005)
 - Millaista muuntelua järjestelmä tukee
 - Variaatiopisteet ja muunnelmille esitettävät vaatimukset → järjestelmän **laajennosrajapinta** (tai erikoistamisrajapinta tai uudelleenkäyttörajapinta)
 - Erityisesti tuoterunkojen yhteydessä
 - Järjestelmän uudelleenkäytettävyyden arviointi
- Eri näkökulmia käytetään tarpeen mukaan
 - Esimerkiksi UML tarjoaa kuvaustekniikoita em. näkymien esittämiseksi

20.9.2012

16

Arkkitehtuurin kuvaaminen - näkökulmia

- *Software Systems Architecture: Working with Stakeholders Using Viewpoints and Perspectives*, by Nick Rozanski and Eoin Woods, published by Addison Wesley (2005):
 - Functional
 - *Toiminnalliset elementit, niiden vastuut, liittymät ja vuorovaikutus*
 - Information
 - *Tiedon säilytykset, käsittely, hajautus*
 - Concurrency
 - *Toiminnallisten elementtien jako rinnakkain toimiviin yksiköihin ja rinnakkaisuuden hallinta*

20.9.2012

17

Arkkitehtuurin kuvaaminen - näkökulmia

- Development
 - Järjestelmäkehittäjien rakenteellinen näkymä
- Deployment
 - Miten asennetaan, laitteistovaatimukset
- Operational
 - Käyttö, asennus ja näiden hallinta

20.9.2012

18

Arkkitehtuurin kuvaaminen - näkökulmia

- ASA Views (kirjasta Hofmeister & al.: Applied Software Architecture)
 - Conceptual, Module, Execution, Code
- **Conceptual view**
 - Toiminnallisuus kuvataan käsitteellisinä komponentteina (conceptual components)
 - Yhteistyö ja tietojenvaihto konnektoreina (connector)
 - Kuvaus on vahvasti sidoksissa kohdealueeseen
 - Kuvaus suhteellisen riippumaton ohjelmisto- ja laitteistoteknologioista
- Huomion kohteina:
 - vaatimusten täyttäminen
 - valmiskomponenttien integrointi
 - kohdealueespesifien SW- ja HW-komponenttien mukaan ottaminen
 - uudelleenkäyttö, tuotelinjä-asiat, tuotejulkaisut ja niiden toiminnallisuus
 - vaatimusten ja kohdealueen muutosvaikutusten minimointi

20.9.2012

19

Arkkitehtuurin kuvaaminen - näkökulmia

- **Module view**
 - Käsitteelliset komponentit ja liittymät sidotaan alijärjestelmiin ja moduuleihin
 - Miten käsitteellinen ratkaisu toteutetaan nykyaikaisessa laite- ja ohjelmistoympäristössä
- Huomion kohteena
 - ohjelmistoaalustakytkenät
 - tarvittavat tukipalvelut
 - testausjärjestelyt
 - moduulien välisten riippuvuuksien minimointi
 - uudelleenkäytön maksimointi
 - tuotteen suojaus komponenttien ja alustan muutoksia vastaan

20.9.2012

20

Arkkitehtuurin kuvaaminen - näkökulmia

- **Execution view**
 - Moduulien toteutus suoritusalueella
 - Suoritusajaiset kohteet ja niiden ominaisuudet (prosessit, muistinkäyttö,...)
 - Suoritusajaisen kontrollin kulku
- Huomion kohteina
 - suorituskykyvaatimusten täytyminen, elpyminen, konfiguraatiomuutokset
 - resurssien kuormitus
 - samanaikaisuus ja hajautusratkaisut
 - ajonaikaisen ympäristön muutosten vaikutusten minimointi

20.9.2012

21

Arkkitehtuurin kuvaaminen - näkökulmia

- **Code view**
 - Ajoaikaisen komponenttien jakelukokoonpano (deployment components, executables)
 - lähdekoodin rakenne
 - jakelukokoonpanon tuottaminen lähdekoodista
- Huomion kohteena
 - Päivitysten työmäärän hallinta
 - versioiden ja kokoonpanojen hallinta
 - käännosten hallinta
 - kehitystyökalujen tarve
 - testauksen ja integroinnin tuki

20.9.2012

22

Arkkitehtuurin kuvaaminen - näkökulmia

- Eri näkymät keskittyvät eri kehittämisnäkökohtiin
 - voimakkaasti toisiinsa kytketyt suunnittelupäätökset omissa näkymissään
 - suunnittelupäätösten priorisointi helpompaa
 - vaihtoehtoisten ratkaisujen evaluointi (trade-off analysis) helpompaa
- Näkymien (viitteellinen) laajitusjärjestys
 1. conceptual
 2. module & execution
 3. code

20.9.2012

23

Arkkitehtuurikuvausten tyypit

- Näkökulmasta riippumatta kuvauksen painotus voi vaihdella, esimerkiksi:
 - Rakenteen vs. käyttäytymisen kuvaus
 - Staattisten vs. dynaamisten piirteiden kuvaus
 - Yleinen esitys vs. esimerkki
- Seuraavat kuvaustyytit periaatteessa riippumattomia näkymistä (tietty tyyppi kuitenkin luonteva tietyn näkymän kanssa)
 - Kuvaustyytit eivät ole riippuvaisia kuvattavan järjestelmän tyyppistä

20.9.2012

24

Arkkitehtuurikuvausten tyypit

- Rakennekuvaus (structural view)
 - Järjestelmän rakenteelliset suhteet (viittausuhteet, sisältyvyysuhteet, periytymissuhteet)
 - Ei havainnollista aikaa; voi kuitenkin koskea sekä staattisia että dynaamisia rakenteita
- Käyttäytymiskuvaus (behavioral view)
 - Ohjelmistoyksiköiden (esim. komponenttien) käyttäytyminen vuorovaikutuksissa
 - Ajalla tärkeä merkitys
 - Yhteys rakennekuvaukseen: esimerkiksi käyttäytymiskuvauksen palvelupyyntö edellyttää rakenteellista yhteyttä palvelun pyytäjän ja palvelun tarjoajan välillä

20.9.2012

25

Arkkitehtuurikuvausten tyypit

- Staattinen kuvaus (static view)
 - Järjestelmän staattista esitystä (esimerkiksi lähdekoodia) koskeva kuvaus
- Dynaaminen kuvaus (dynamic view)
 - Järjestelmän suoritusaikana esiintyvän ominaisuuden kuvaus
 - Esimerkiksi olioiden väliset suhteet
- Määrittelykuvaus
 - Kiinnostuksen kohteena kohde kokonaisuudessaan (esimerkiksi täydellinen tietyn luokan olioiden käyttäytymisen kuvaus)
- Esimerkkikuvaus
 - Järjestelmän tietyn ilmiön ilmentymän kuvaus (esimerkiksi tietty oliokonfiguraatio tietyssä tilanteessa)

20.9.2012

26

Arkkitehtuurikuvausten tyypit

- Kuvaustyyppin valintaan vaikuttavat
 - Tarkoituksenmukaisuus
 - Ymmärrettävyys (kyseisessä kuvauksessa)
 - Kuvauksen tekemisen vaatima työmäärä
- Esimerkin antaminen useimmiten helpompaa kuin täydellisen määrittelyn
 - Tärkeät erikoistapaukset voivat kuitenkin jäädä huomiotta
- Usein vältetään samassa kuvauksessa vastakkaisia kuvaustyyppijä
 - Ei samaan kuvaukseen esimerkiksi dynaamisia ja staattisia piirteitä tai esimerkkiä ja määrittelmää

20.9.2012

27

Arkkitehtuuridokumentit

- Arkkitehtuuridokumentin muotoon ja sisältöön vaikuttaa:
 - Kuvattava järjestelmä
 - Kuvauksen tekijän (organisaation) omat käytännöt
 - Organisaation koko
 - Järjestelmän/projektin koko
- Arkkitehtuurikuvaus oltava kuitenkin selkeä osa dokumentointia
- Arkkitehtuuridokumentti viittaa vaatimusdokumentteihin perustellessaan tehtyjä arkkitehtuuriratkaisuja
- Toisaalta yksityiskohtaiset suunnitteludokumentit ja testausdokumentit viittaavat arkkitehtuuridokumenttiin

20.9.2012

28

Arkkitehtuuridokumenttityyppejä

- Alustava arkkitehtuuridokumentti
 - Tärkeimmät arkkitehtuuriratkaisut ja niiden perustelut
 - Vaihtoehtoisten ratkaisujen analyysi
 - Konkreettisen arkkitehtuurin kuvaus + viitteet referenssiarkkitehtuureihin
- Järjestelmäarkkitehtuuridokumentti
 - Arkkitehtuurin ylin taso: sidokset ympäristöön, alijärjestelmien rajapinnat ja keskinäinen vuorovaikutus
 - Pohja alijärjestelmien arkkitehtuurisuunnittelulle, arkkitehtuurien arvioinnille, tarkennetuille työmääräarvioille, projektisuunnittelulle ja järjestelmätestauksen suunnittelulle
 - Konkreettinen arkkitehtuuri (+ alijärjestelmien meta-arkkitehtuurit)

20.9.2012

29

Arkkitehtuuridokumenttityyppejä

- Alijärjestelmäarkkitehtuuridokumentti
 - Eryteisesti suurissa järjestelmissä, joissa alijärjestelmätkin voivat sisältää tärkeitä arkkitehtonisia ratkaisuja
- Tuoterunkoarkkitehtuuridokumentti
 - Eryteisesti: Miten rungon tarjoamaa varianssia käytetään sovellusten rakentamisessa?
 - Esimerkitapaukset, rajoitteet
- Tuotearkkitehtuuridokumentti
 - Miten tuoterunkoarkkitehtuuria on sovellettu?
 - Mitä tuoterungosta riippumattomia ratkaisuja on tehty?
- Rajapintadokumentti
 - Tarjotun ohjelmointirajapinnan (API) dokumentaatio
 - Määrittelyssä voidaan hyödyntää esimerkiksi tulo- ja jätteehtoja

20.9.2012

30