



Ohjelmistoprojektien johtaminen Projektissa käytettävät työkalut

Päivi Kuuppelomäki

1



Työkaluja

Esim.

- MS Project 98
- PMW (Project Manager Workbench)
- Excel
- CVS

2



Milloin kannattaa käyttää projektihallinto-ohjelmistoa

- 50-100 tehtävää
- koko projektin kesto yli 6 kk
- henkilöitä yli 10 ja monet heistä osapäiväisiä
- projektilla kriittinen aikataulu ja vaativat valvontatarpeet
- toimitaan moniprojektiympäristössä

3



MS Project 98

- Tehtävät
- Resurssit
- Resurssien allokointi
- Kalenterit
- Kustannuslaskenta
- Kriittinen polku
- Raportointi

4

Tehtävät

- Tehtävien syöttö esim. Gantt Chart-näkymästä
- Kesto: viikkoja, päiviä, tunteja, minuutteja
- Tehtävien linkitys (FS, SS, FF, SF, None)
- Tehtävien aikataulutus
- Yhteenvedo- ja alitehtävät
- Toistuvat tehtävät
- Tarkistuspisteet

5

Resurssit

- Resurssien syöttö Resource Sheet -näkymän avulla
- Yksilö- ja Ryhmäresurssit
- Kenttiä: Resource Name, Intials, Group, Max. Units, STD Rate, Ovt. Rate, Cost/Use, Accrue AT, Base Calendar, Code

6

Resurssien allokointi

- Gantt Chart -näkömän ja Assign Resources -ikkunan avulla
- Resurssi voidaan liittää tehtävään kokonaan tai vain osa-aikaisesti (Units-sarake kertoo osuuden)
- Liitettäessä samaa resurssia useampi kuin yksi kirjataan sitä vastaava prosenttiluku Units-sarakkeeseen (esim. 2 henkilöä -> 200 %)

7

Kalenterit

- Peruskalenteri
- Henkilökohtaiset kalenterit
- Työajan muuttaminen
 - oletustyöaika (8-17) voidaan muuttaa
 - oletusvapaapäivinä lauantai ja sunnuntai
 - vapaapäiviä voidaan lisätä ja poistaa

8

Kustannuslaskenta

- Komennolla Project/Project Information ja edelleen avautuvasta ikkunasta painamalla Statistics-painiketta
- Kustannuksista voidaan määrittää missä vaiheessa ne erääntyvät (Start, Prorated ja End)
- Earned Value työkalu

9

Krittiinen polku

- Napsauta Gantt Chart -näkyvässä ChartWizard -painiketta ja valitse vaihtoehto Critical path ohjatun toiminnon avulla
- Kriittinen polku näytetään Gantt Chart -näkyvässä punaisella

10

Tapahtumien kirjaaminen

- Kirjaukset Resource Usage ja Task Details Form -näkymien avulla
- Kirjattavia tietoja: Start, Finish, Duration, % Complete

11

Raportointi

- Mikä tahansa näkymä voidaan tulostaa
- Valmiita raportteja
 - Yleisraportit: Project Summary, Top-Level Tasks, Critical Tasks, Milestones, Working Days
 - Tapahtumaraportit: Unstarted Tasks, Tasks Starting Soon, Tasks In Progress, Completed Tasks, Should Have Started Tasks, Slipping Tasks

12

Raportointi jatkuu..

- Kustannusraportit: Cash Flow, Budget, Overbudget Tasks, Overbudget Resources, Earned Value
- Toimeksiantoraportit: Who Does What, Who Does What When, To-do List, Overallocated Resources
- Työkuormaraportit: Task Usage, Resource Usage
- Muokattavat raportit

13

CVS

- Versionhallintajärjestelmä
- Tallentaa vain versioiden väliset erot
 - säästää tilaa
- Helpottaa ryhmätyöskentelyä
 - samanaikaiset muutokset mahdollisia
 - jokainen työskentelee omassa hakemistossaan ja lopuksi lomitetaan muutokset

14

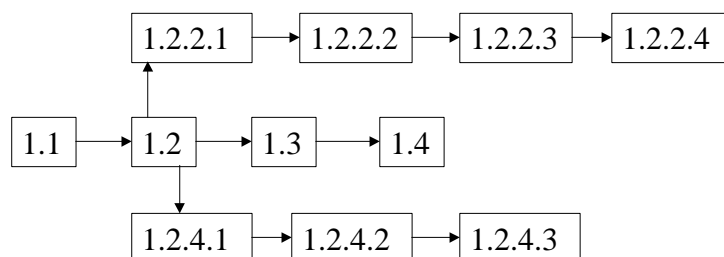
Keskusvarasto

- Kaikki versiot talletetaan keskusvarastoon
- Keskusvaraston tiedostoja ei päivitetä suoraan
- CVS:n komennoilla haetaan oma kopio tiedostoista, tehdään muutokset ja sen jälkeen erillisellä komennolla päivitetään ne keskusvarastoon
- Tiedostot keskusvarastossa järjestetään moduuleiksi

15

Versionumerot

- Jokaisella tiedoston versiolla on oma versionumero, esim. 1.1, 1.2, 1.3.2.2



16

Esimerkki istunto

- Kuitataan moduuli ulos

```
$ cvs checkout tc
```

- Komento luo uuden hakemiston ja tuo sinne moduulin tiedostot

```
$ cd tc
```

```
$ ls tc
```

```
CVS      Makefile  backend.c  
driver.c frontend.c parser.c
```

17

Esimerkki istunto jatkuu...

- Muutetaan tiedostoa backend.c

- Päivitetään muutokset komennolla

```
$ cvs commit backend.c
```

– cvs käynnistää editorin, jotta voidaan kirjata kommentti lokitiedostoon

18

Ympäristömuuttuja CVSROOT

- Ympäristömuuttujassa CVSROOT pitää olla absoluuttinen polkunimi, joka kertoo keskusvaraston juuren
setenv CVSROOT /usr/local/cvsroot
- Keskusvarastoja voi olla useita ja CVSROOT:in arvo määrää mitä käytetään

19

Moduulin luonti

- Jos moduliin tulevat tiedostot sijaitsevat hakemistossa DIR, niin luonti voidaan tehdä seuraavasti:
\$ cd DIR
\$ cvs import -m "Imported sources"
yoyodyne/DIR yoyo start
– kun keskusvarasto on
\$CVSROOT/yoyodyne/DIR

20

Moduulin lisääminen moduulistaan

```
$ cvs checkout modules
```

```
$ cd modules
```

<tiedostoon modules lisätään rivi:

```
DIR yoyodyne/DIR>
```

```
$ cvs commit -m "Added the DIR module."  
modules
```

```
$ cd ..
```

```
$ cvs release -d modules
```

21

Tiedoston/hakemiston lisääminen moduuliin

- Luodaan uusi tiedosto/hakemisto omaan työkopioon

- Moduulin lisäys

```
$cvs add newfile
```

```
$cvs commit newfile
```

- Vastaavasti hakemiston lisäys

- Komennot on suoritettava tiedoston/
hakemiston sijaintihakemistossa

22



Tiedoston poistaminen moduulista

- Tiedosto voidaan poistaa `rm`-komennolla, jolloin se poistuu keskusvarastossa seuraavan `commit:n` yhteydessä
- Tai seuraavasti:
`$cvs remove oldfile`
`$cvs commit oldfile`
- Ongelmia, jos luodaan myöhemmin samanniminen tiedosto

23



Tiedoston tila

- Tiedosto on identtinen keskusvarastossa olevan tiedoston kanssa
- Paikallisesti muutettu
- Tarvitsee päivitystä
- Tarvitsee lomistusta

24

Tiedoston päivittäminen

- Mikäli keskusvarastossa olevaa uutta versiota ja omaa työkopiota on muutettu eri kohdista, tekee CVS muutokset automaattisesti
- Jos on ristiriitaisia muutoksia, lisätään kummatkin työkopioon ja ristiriidat on poistettava ennen uutta yritystä
- Tiedoston voi päivittää ajantasalle update komennolla

25

Kirjallisuutta

- Kai Ruuska, Projekti hallintaan, Suomen Atk-kustannus Oy, Espoo, 1997
- Heikki Stenlund, Projektijohtamisen perusteet, Oy Edita Ab, Helsinki, 1996
- MS Project 98, Projektinhallinta, Teknolit, Jyväskylä 1999

26