

Endringshåndtering og konfigurasjonsstyring av programvare (SCCM)

Hans Christian
Benestad

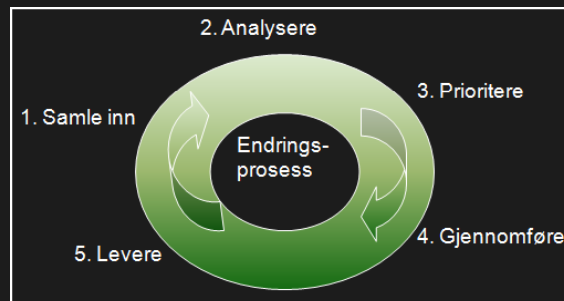
Simula Research
Laboratory



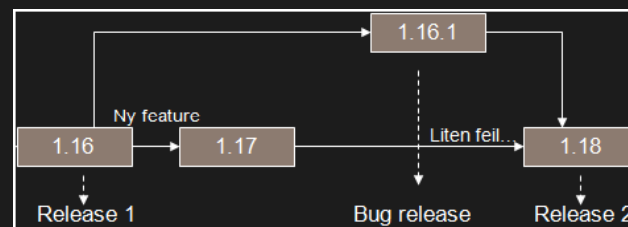
Denne forelesningen beskriver SCCM's rolle under programvare-evolusjon



Behovet for endringer
(og endringskontroll...)



Endringsprosessen
(med knytninger til tidligere forelesninger)



Håndtering av endringer
i systemkomponenter

Målet for SCCM er å holde orden på endringer under utvikling og evolusjon



Hvorfor ble dette endret?
Ser helt feil ut. Jeg endrer tilbake.

Utviklere



Er det noen som vet om vi
har rettet denne feilen?

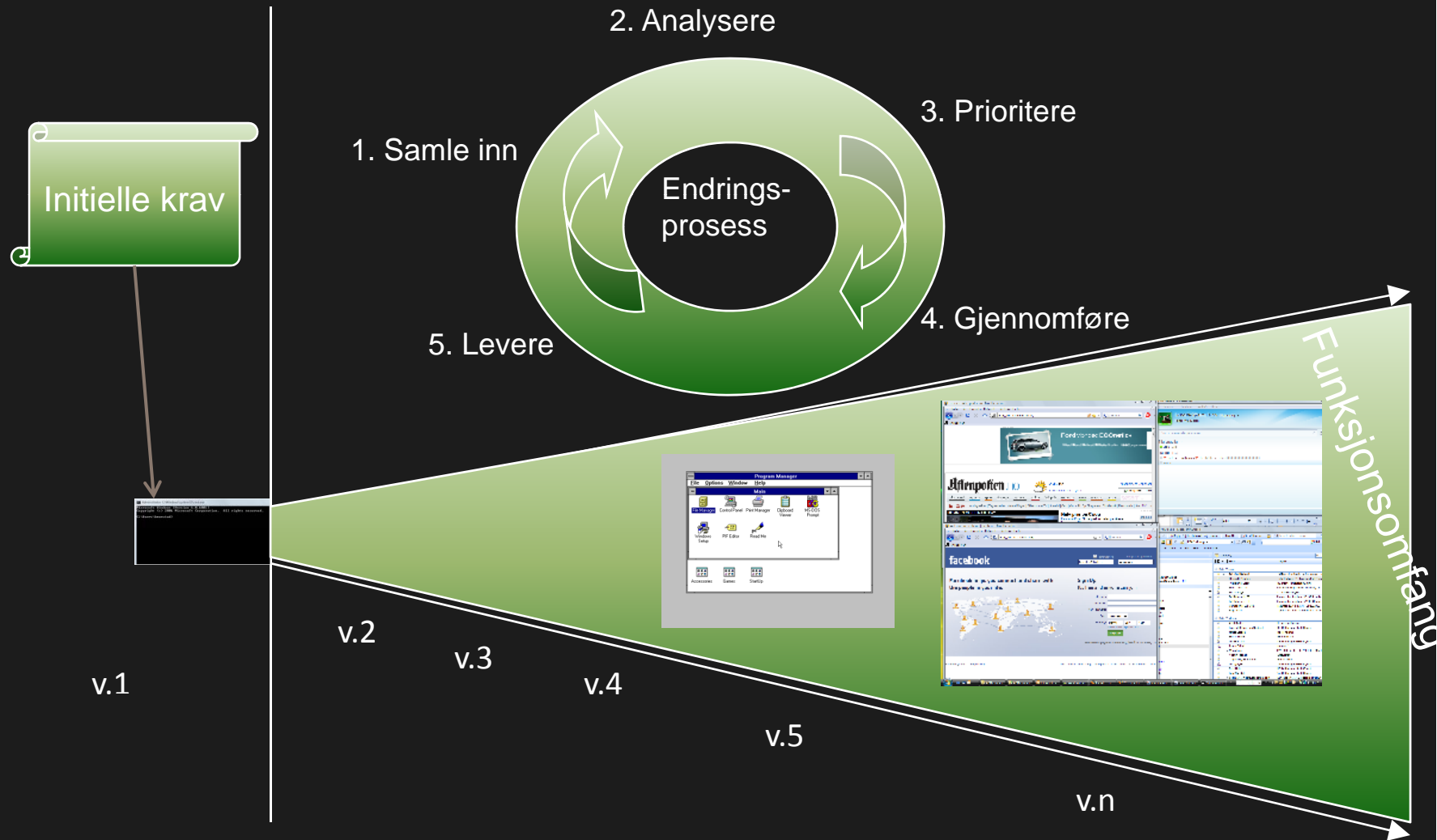
Prosjektet



Denne feilen var rettet men
dukker nå opp igjen etter at
jeg gikk over til Linux!

Brukere

Evolusjonsfasen kan være lang sammenlignet med initiell utvikling



Det er nødvendig og ønskelig å endre programvare



An *E*-type system must be continually adapted else it becomes progressively less satisfactory in use
- Manny Lehman 1974



Embrace change
- Kent Beck 1999

Noen årsaker til endringsbehov:

- Feilretting
- Tilpasning til nytt kjøremiljø
- Nye brukerkrav

"Dimensions of software maintenance" – Swanson 1976

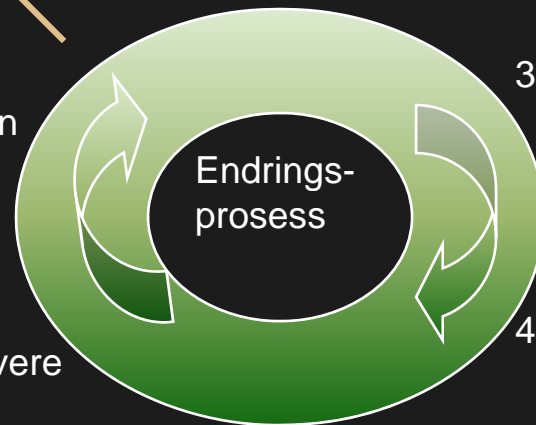
Vellykket evolusjon forutsetter kontrollert håndtering av endringsønsker

Hver endring gjennomgår en systemutviklingsprosess i mikroformat



1. Samle inn

2. Analysere



3. Prioritere



4. Gjennomføre



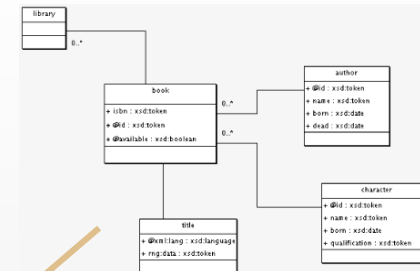
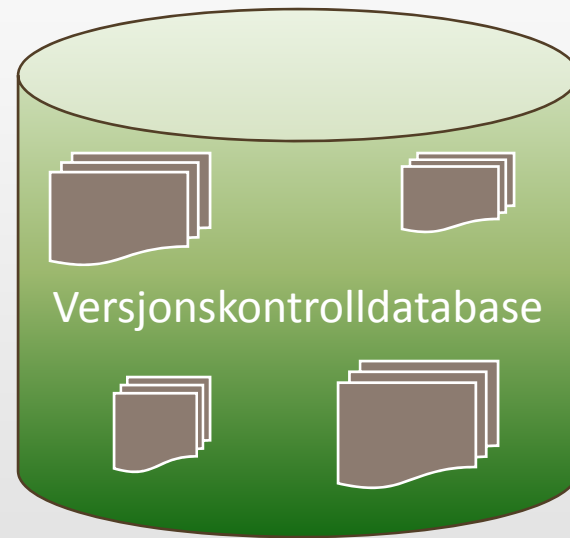
5. Levere

En *endringsprosess* beskriver kommunikasjonsveier, arbeidsflyt, ansvar og beslutningsprosesser

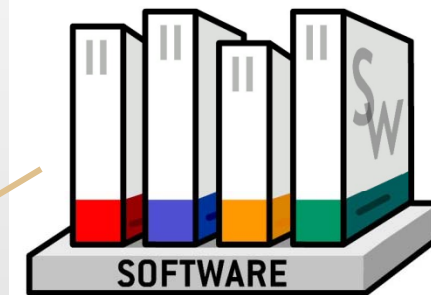
Endringskontroll av programvarekomponenter er sentralt i endringsprosessen

- SOFTWARE REQUIREMENT SPECIFICATION- DHFSC-81433
1. Scope
 2. Reference Documents
 3. Requirements
 - 3.1 Required states and modes
 - 3.2 CSCI capability requirements
 - 3.3 CSCI external interface requirements
 - 3.4 CSCI internal interface requirements
 - 3.5 CSCI internal data requirements
 - 3.6 Adaptation requirements
 - 3.7 Safety requirements
 - 3.8 Security & privacy requirements
 - 3.9 CSCI environment requirements
 - 3.10 Computer resource requirements
 - 3.11 Software quality factors
 - 3.12 Design and Implementation constraints
 - 3.13 Personnel-related requirements
 - 3.14 Training-related requirements
 - 3.15 Logistics-related requirements
 - 3.16 Other requirements
 - 3.17 Packaging requirements
 - 3.18 Precedence and criticality of requirements
 4. Qualification Provisions
 5. Requirements Traceability
 6. Notes
 - A. Appendixes

Krav



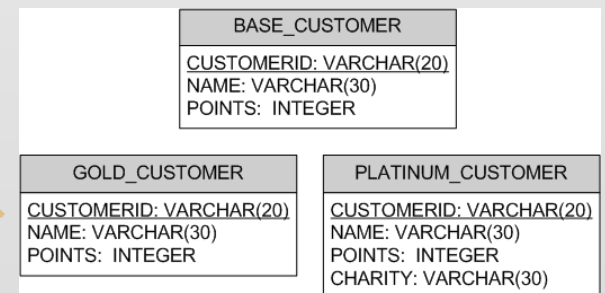
Modeller



Verktøy og biblioteker

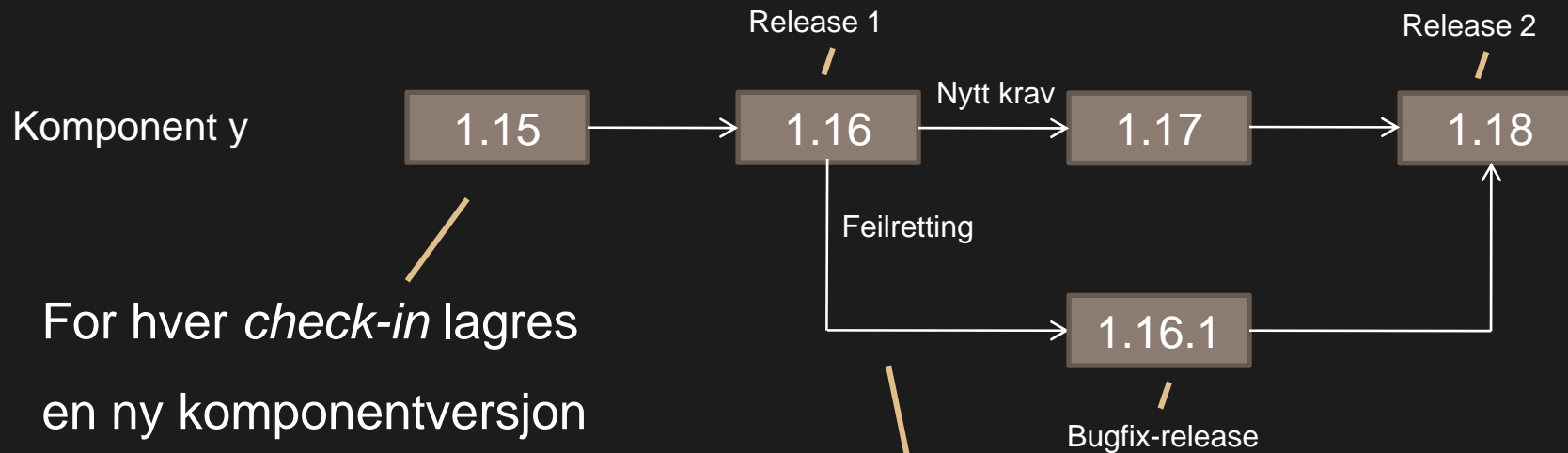


Kildekode



Databaseskjema

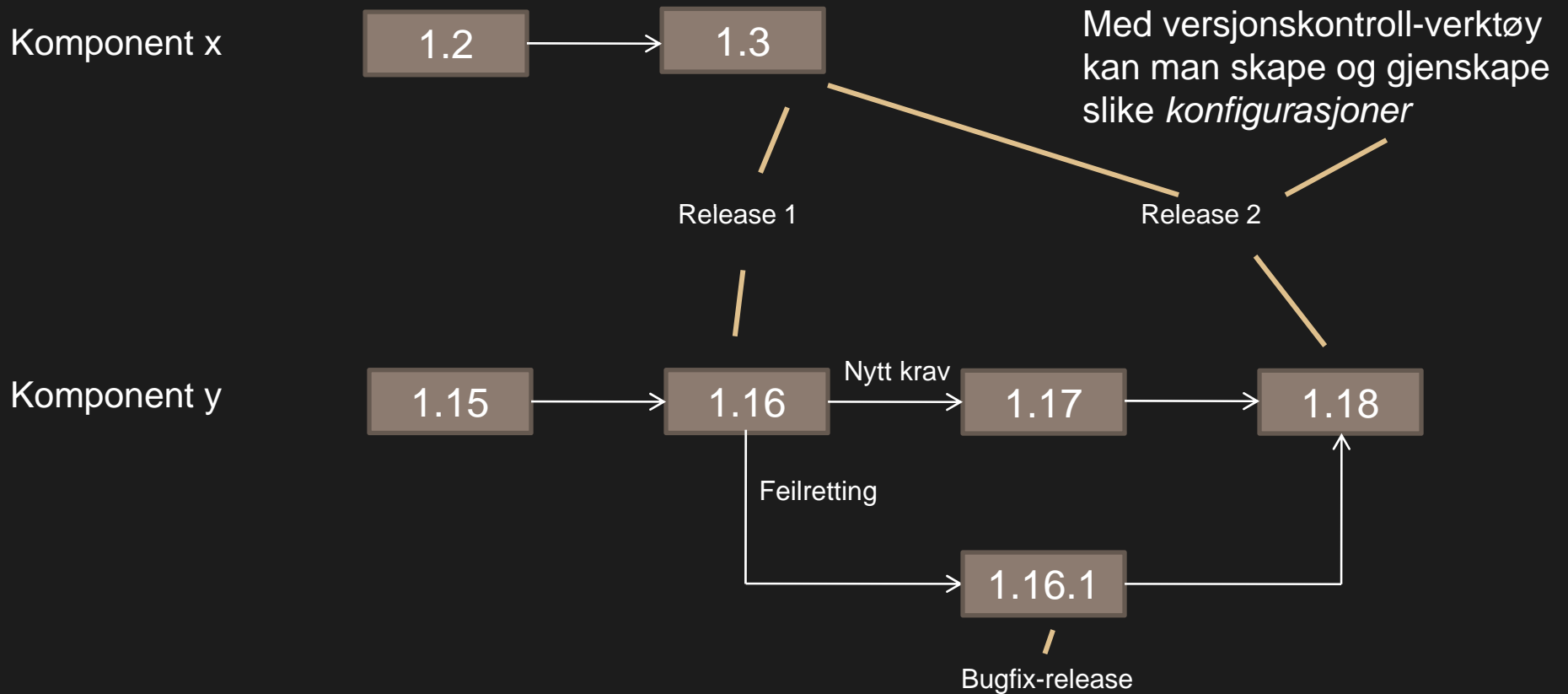
Et versjonstre inneholder hver komponents historikk



For hver *check-in* lagres
en ny komponentversjon

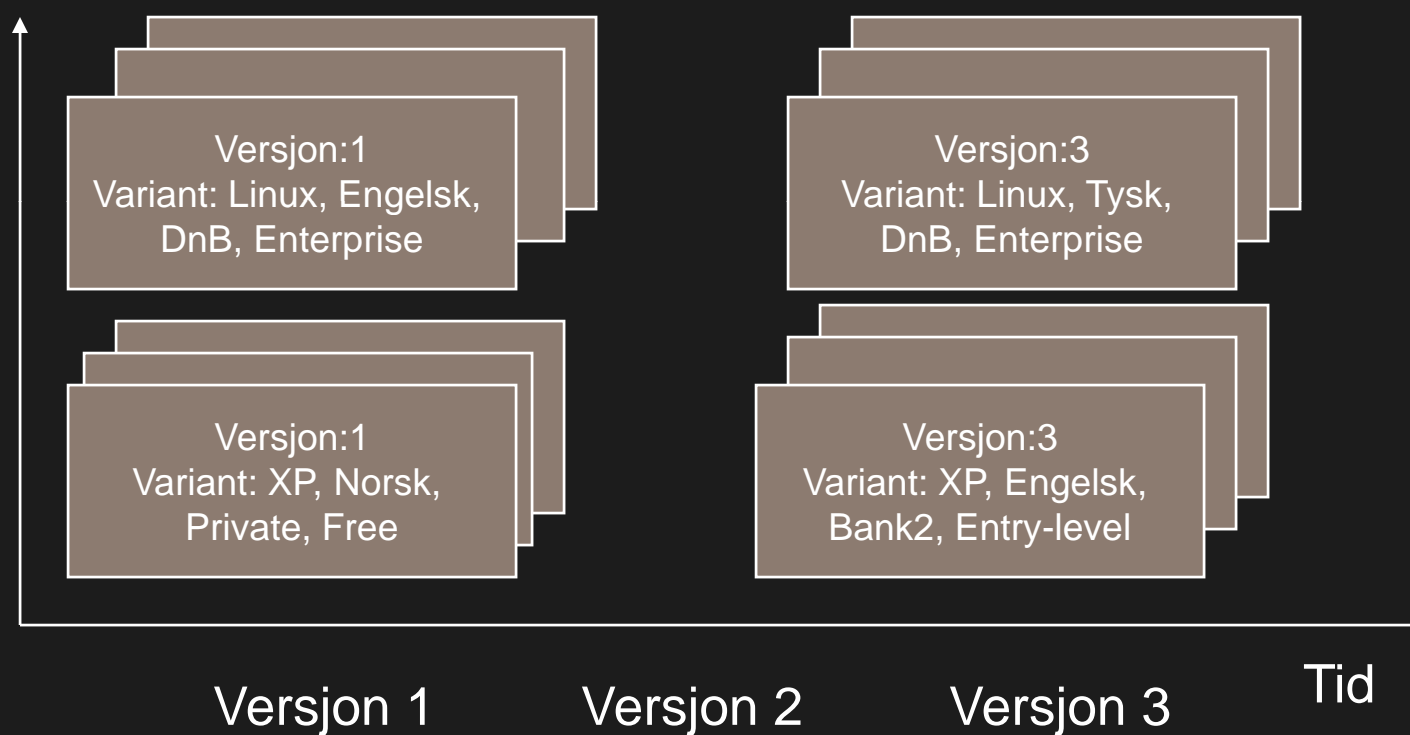
Vanligvis bare en rett linje (trunk,
mainline), men forgreninger kan være
nødvendig

I en *konfigurasjon* inngår nøyaktig én versjon av hver komponent



Varianter og versjoner er ortogonale begreper

Variant (OS, språk, kunde, prissegment)



Varianter kan håndteres *dynamisk* eller *statisk*

Dynamisk: Programvaren tilpasser seg omgivelsene under kjøring

Eksempel: Program leser miljøvariabel LANG, og hent brukermeldinger fra fil med navn LANG.txt

+En felles versjon til alle brukere

Statisk: Den kjørbare programvaren lages i forskjellige varianter

Eksempel: Inkluder ulike kildefiler under bygging av ulike varianter

+Kan gi enklere/mer effektiv kode
- Kompliserende SCCM

Download

SmartCVS requires a Java Runtime Environment (JRE) 1.4.1 or higher to run.

- ⇒ For **Microsoft Windows NT/2000/XP/2003/Vista**, you may choose the installer for your operating system.
- ⇒ For **Mac OS X**, version 10.4 or newer, the necessary JRE is present on your system.
- ⇒ For **other platforms**, you can download platform-specific JRE packages from the Java website.

Java EE 5 SDK (with JDK 6 U13)

Provide Information, then Continue to Download

Select Platform and Language for your download:

Platform:
Language:
Linux
Solaris SPARC
Solaris x86
Windows

En god arkitektur forenkler varianthåndtering

Tre-lags arkitektur
er et godt utgangspunkt



Særegenheter for GUI,
operativ- og databasesystem i
adskilte komponenter

Data-definert oppførsel
(eks. språk, kundesegment)

Kravhåndtering er viktig i endringsprosessen

Hvordan dokumenteres og formidles krav?

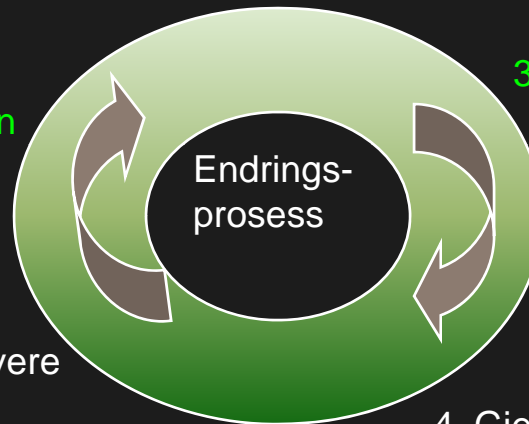
Hvordan passer kravet inn i eksisterende system og produktstrategi?

2. Analysere

3. Prioritere

1. Samle inn

Hvordan motivere brukere til å kommunisere behov?



5. Levere

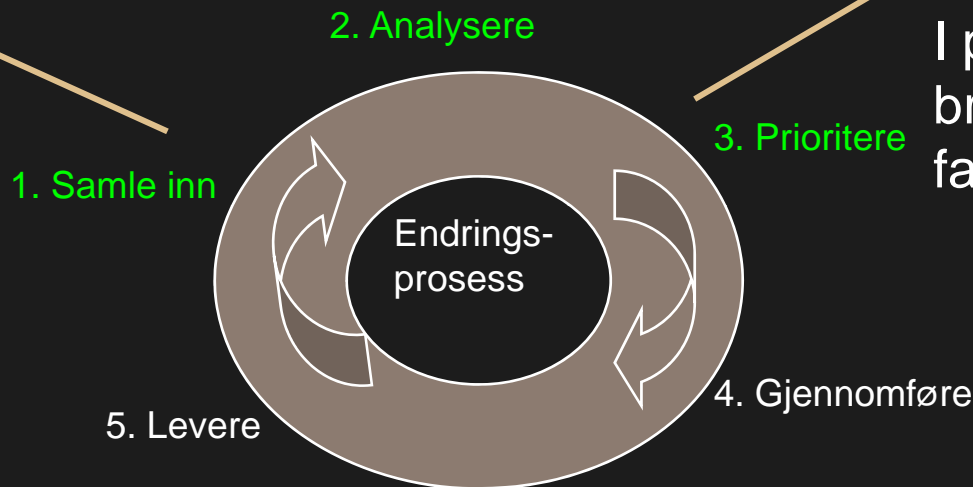
4. Gjennomføre



Populære smidige metoder anbefaler en iterativ kravprosess også under initiell utvikling!

Kontraktstyper som forutsetter veldefinert omfang er vanskelig å bruke under videreutvikling

Nye krav framkommer iterativt



Hvem betaler?

I praksis vil mange bruke kombinasjon av fastpris og timepris

Kommersielle avklaringer må integreres i endringsprosessen

Fra PS2000:

E 2.2 Omfang av garantiytelsene

Garantien dekker feilsøking og utbedring av Feil i Leveransen. Dersom det er tvil om Leverandøren er ansvarlig for feilsituasjonen, skal Leverandøren likevel kontinuerlig bidra i feilsøkingen, frem til det er dokumentert at Kunden eller en tredjepart er ansvarlig for feilsituasjonen og således også for utbedring. Alle Feil som er gjenstående fra Godkjenningprøven skal utbedres i denne perioden.

SCCM er sentralt ved testing før leveranse



Hvilke feil er utestående?

Hvilken versjon skal testes?

Hvilke testcase skal kjøres?

SCCM også av testdata!

God SCCM kommer brukerne til gode

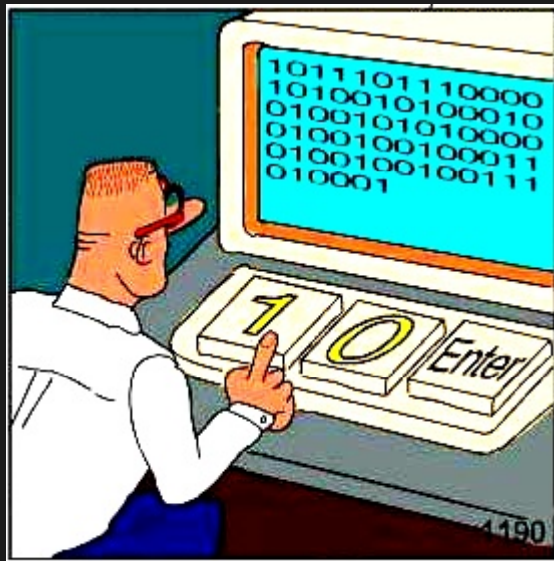
Hvordan oppgraderer jeg fra tidligere versjon?

Hvilke features har den nye versjonen?

Hvordan melder jeg endringsønsker?



God SCCM kommer utviklere til gode



REAL Programmers code in BINARY.

Hvilke endringer har jeg ansvar for?

Hvilken versjon og variant gjelder endringen?

Hvorfor er denne programkoden endret?

God SCCM kommer prosjektet til gode



Når kan vi slippe neste versjon?

Hva må testes før neste versjon?

Hva må oppdateres i brukerdokumentasjonen?

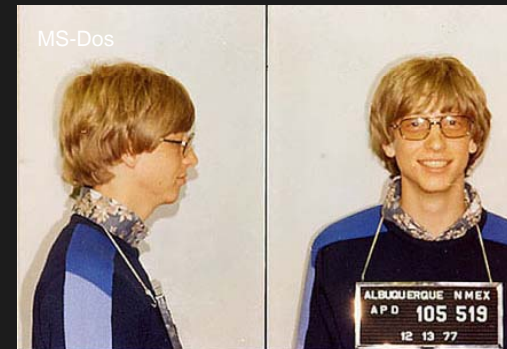
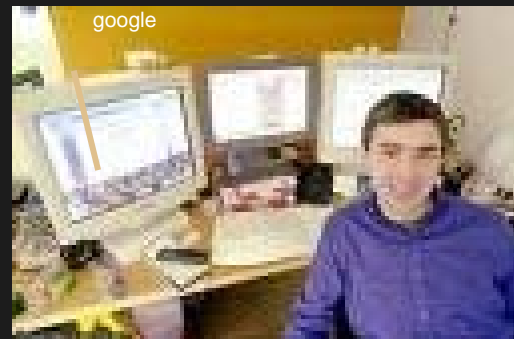
Kognitiv kapasitet er en kritisk begrensning i systemutvikling



- God SCCM frigjør kapasitet til kreativt arbeid

Når er SCCM irrelevant?

Én utvikler lager én versjon av ett system



Men så populært dette ble da

MS-DOS R.I.P.

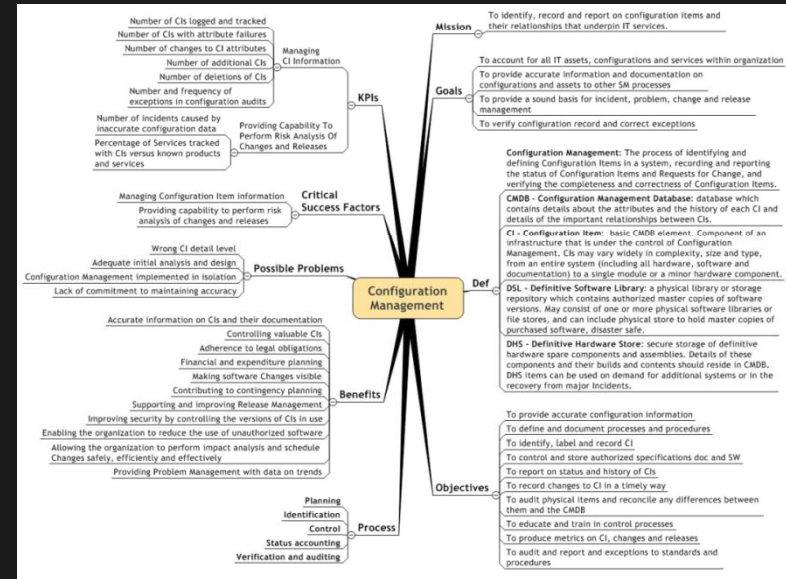
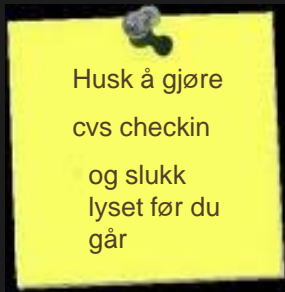
Opprinnelig kodebase lever ofte lenge!

MS-DOS passed away Thursday, October 25, 2001, at the Marriott Marquis Hotel on Times Square in New York City.

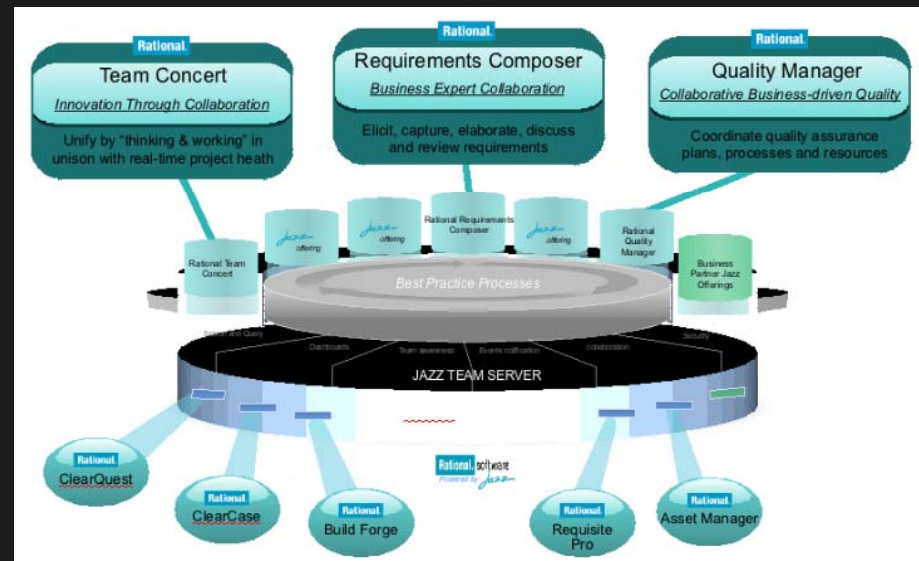
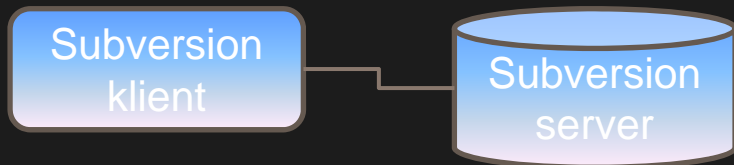
MS-DOS was born in August 1980, in Tukwila, Washington, the creation of Tim Paterson and the Seattle Computer Company. Initially called QDOS 0.10 (short for "Quick and Dirty Operating System"), MS-DOS

Dosering av SCCM må tilpasses behov

Uformelle vs. formaliserte rutiner.



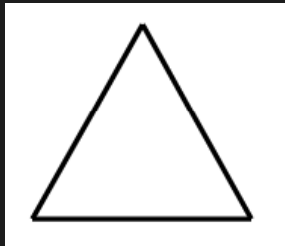
Enkle vs. state-of-the-art verktøy



Behovet er større i store prosjekter

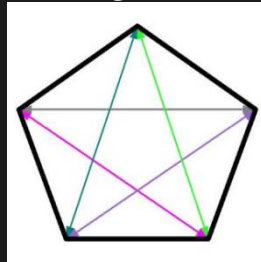
- Antall kommunikasjonsveier øker kvadratisk $(n^2-n)/2$

n=3



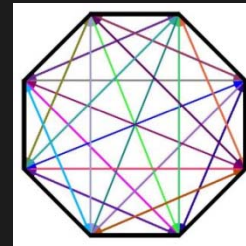
3

n=5



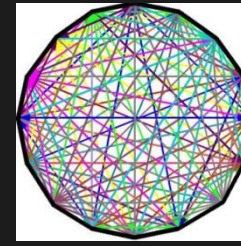
10

n=8

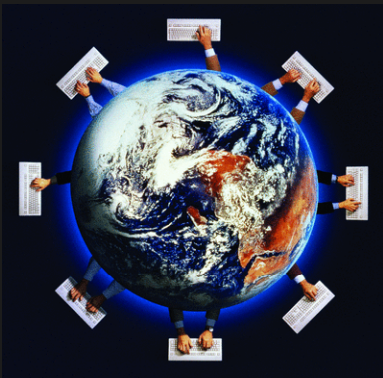


28

n=16



120

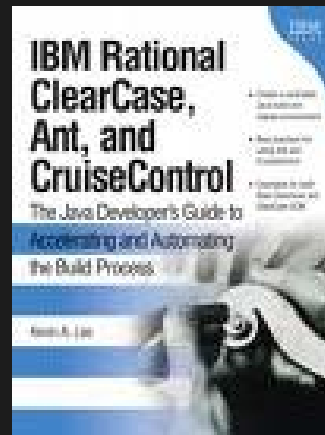


...og i desentraliserte prosjekter

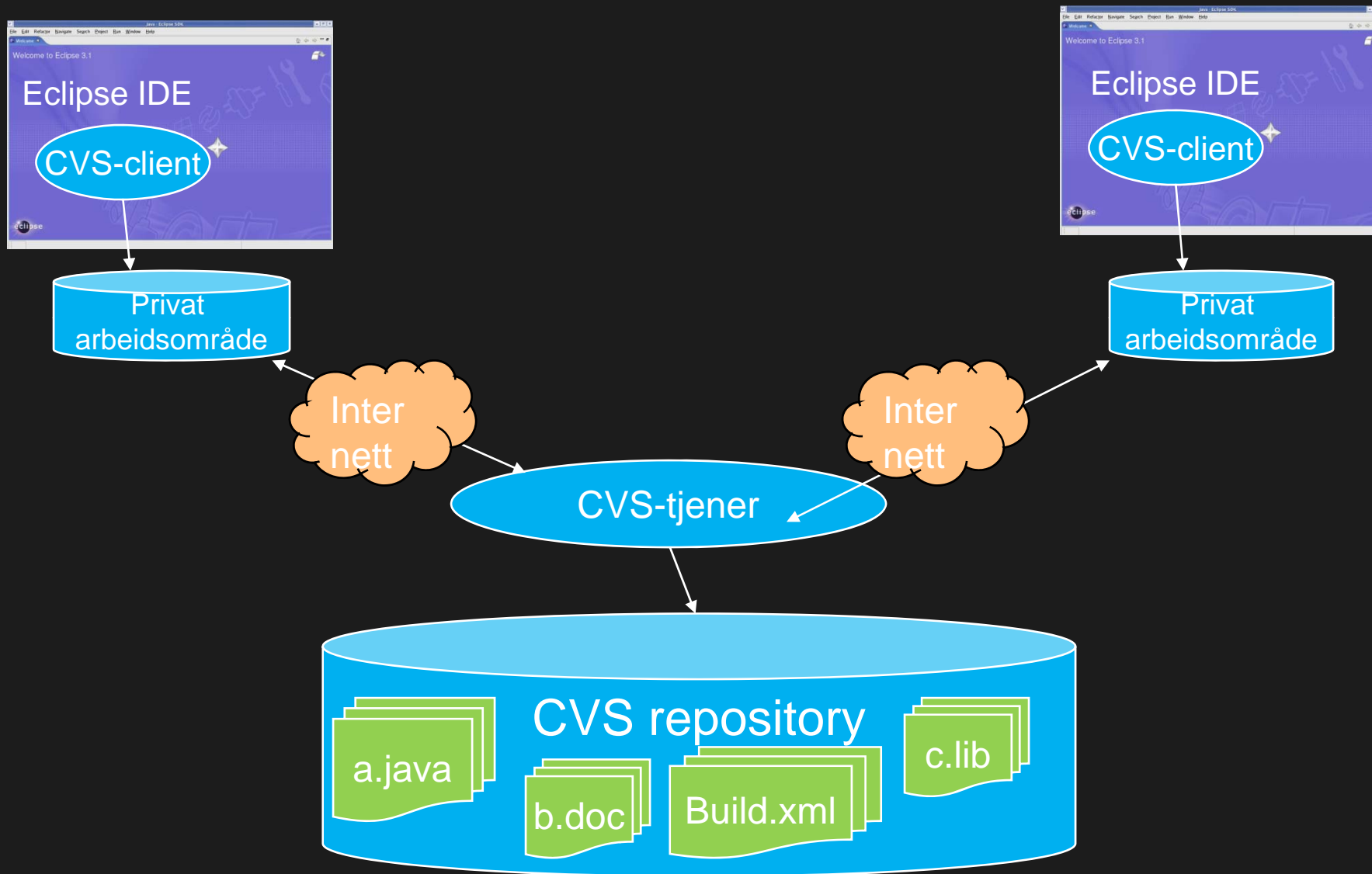
Bruk av verktøy for versjonskontroll



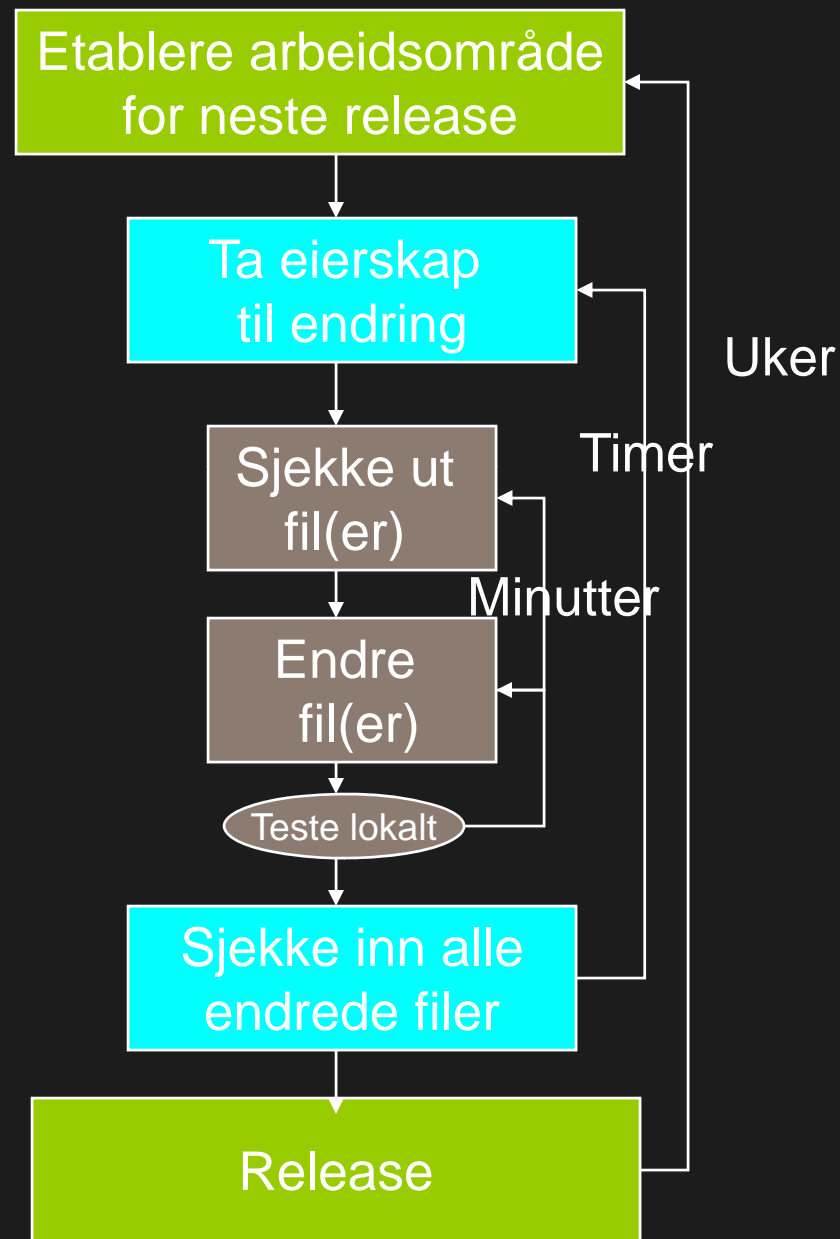
Synergy Suite



Typisk arkitektur i versjonskontroll-systemer



Endringsssyklus sett fra utvikler



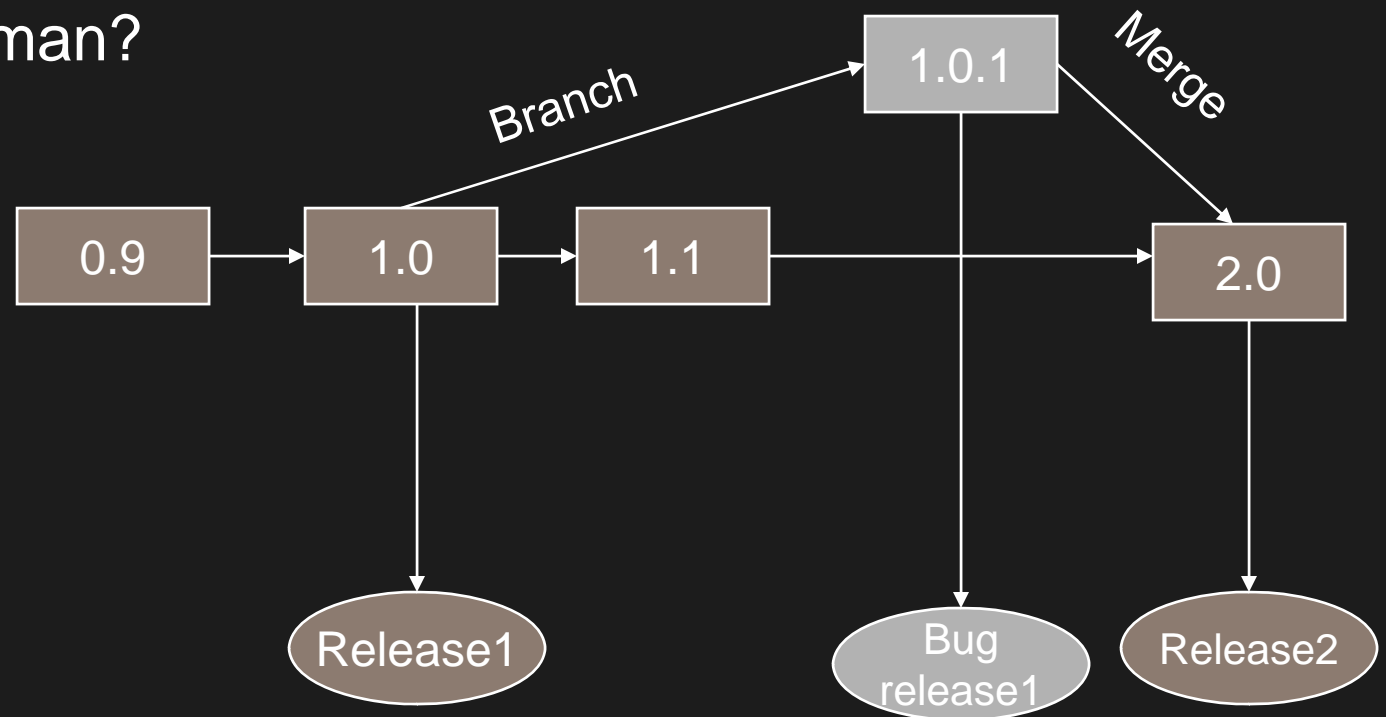
Arbeidsområde (workspace/view) er en utviklerspesifikk konfigurasjon

- For å kunne gjennomføre, bygge og teste må hver utvikler ha sitt personlige arbeidsområde
- Et typisk innhold i arbeidsområder er
 - Siste versjon i Mainline
 - Overstyrt av siste versjon i en aktuell forgrening (for eksempel BugRel1)
 - Overstyrt av egne lokal endringer som ikke er sjekket inn enda
- Holdes synkronisert med innholdet i repository
 - Eksplisitt via kommando "Update"
 - Eventuelt automatisk (støttes av noen verktøy)

Løpende utvikling må kunne skilles fra feilretting

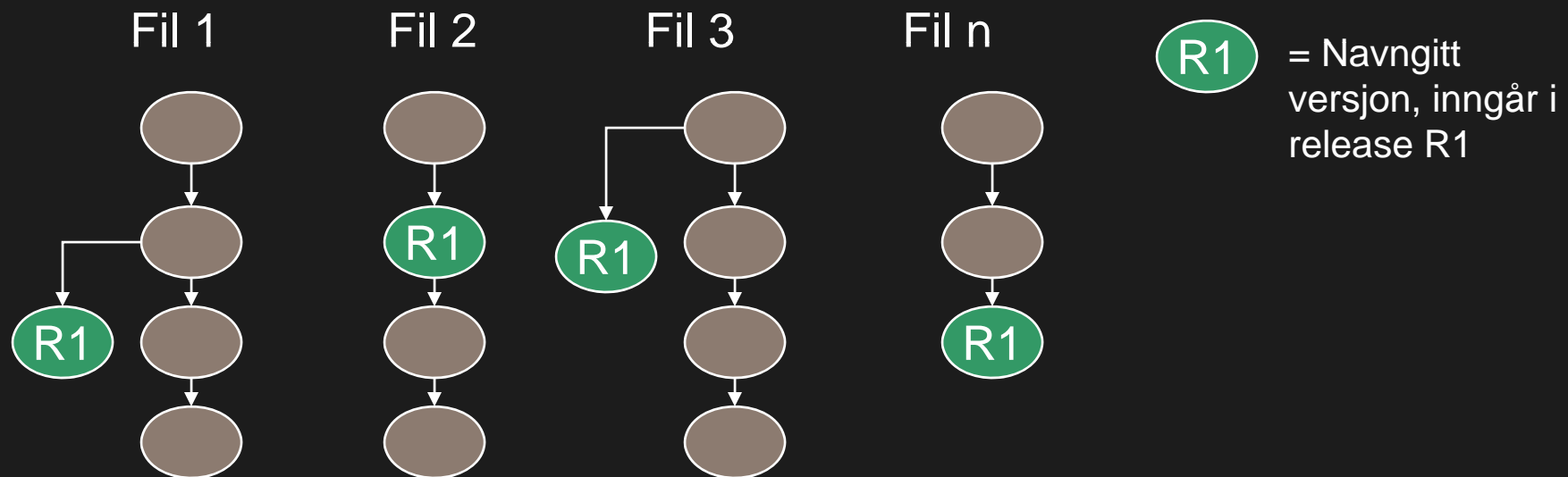
- Scenario:
- Release1 er sluppet, og utvikling pågår for release2
- Kunden opplever kritisk feil i Release1
- Hva gjør man?

HalloVerden
.java



Navngiving (label/tag) gjør det enkelt å gjenskape konfigurasjoner

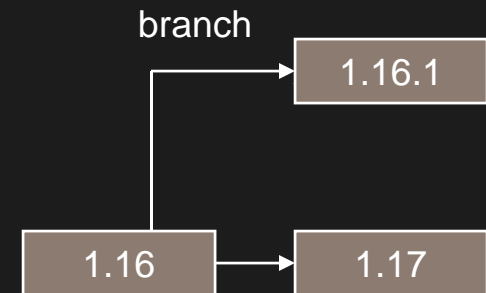
All kode som inngår i en release assosieres med et navn, for eksempel R1 (> cvs tag -R R1)



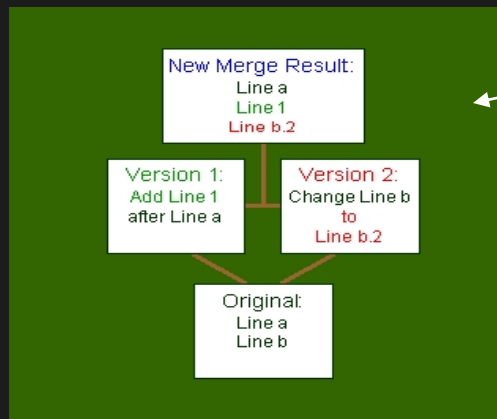
R1 kan da enkelt gjenskapes, med kommando som (> cvs checkout -r R1)

Med forgrening kan man jobbe med flere utgaver av samme komponent

- Aktuelt ved
 - Skille langsiktig utvikling fra kortsiktige releaser
 - Eksperimentell utvikling
 - Utvikling av større features
 - Samtidig utvikling i samme fil
- Kost/nytte må vurderes
 - Forgrening for hver logiske endring er som regel overkill
- Langlivede forgreninger bør unngås

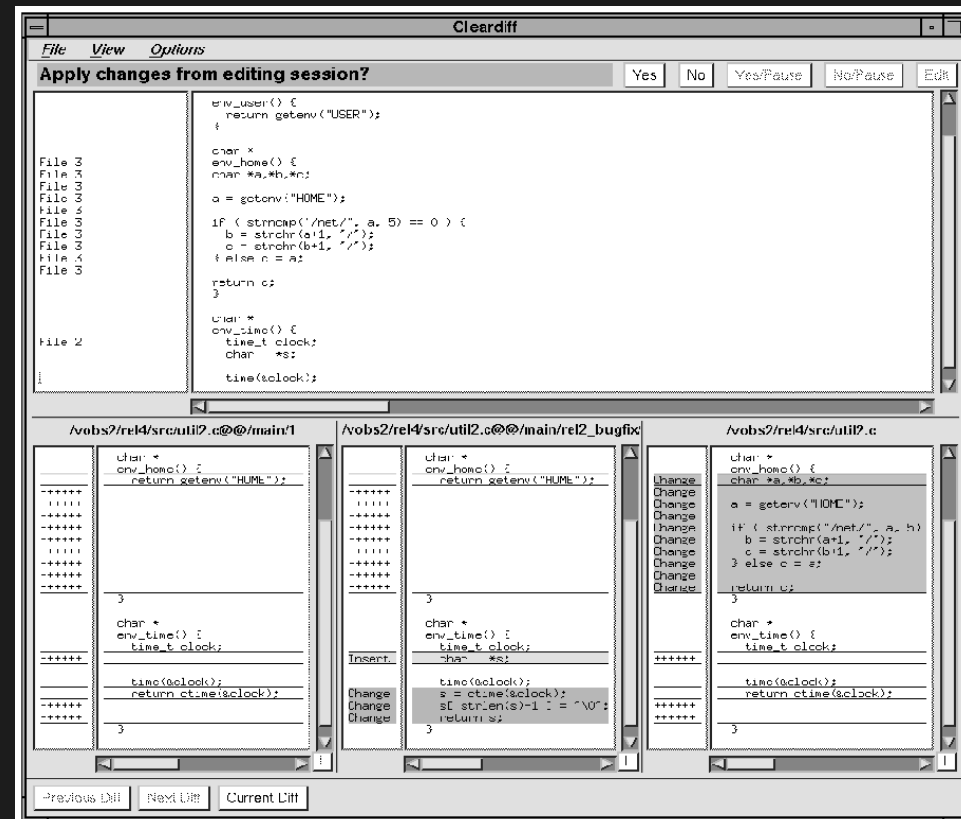


Fletting (merge) slår sammen forgreninger



- 3-veis merge er vanlig
- Konflikter må håndteres manuelt, med støtteverktøy

- Produktnivå merge
Eks: Merge alle Bug-forgreninger inn i Mainline
Her varierer verktøystøtten



Pessimistisk vs. optimistisk håndtering av parallellitet

Pessimistisk (SCCS, Clearcase...)



checkout fil1: **OK**

checkin fil1: **OK**

tid



checkout fil1: **Error**

checkout fil1: **Ok**

Optimistisk (CVS, Subversion)



checkout fil1: **OK**

checkin fil1: **Ok**

tid



checkout fil1: **Ok**

checkin fil1: **Conflict**

Verktøy for endringshåndtering

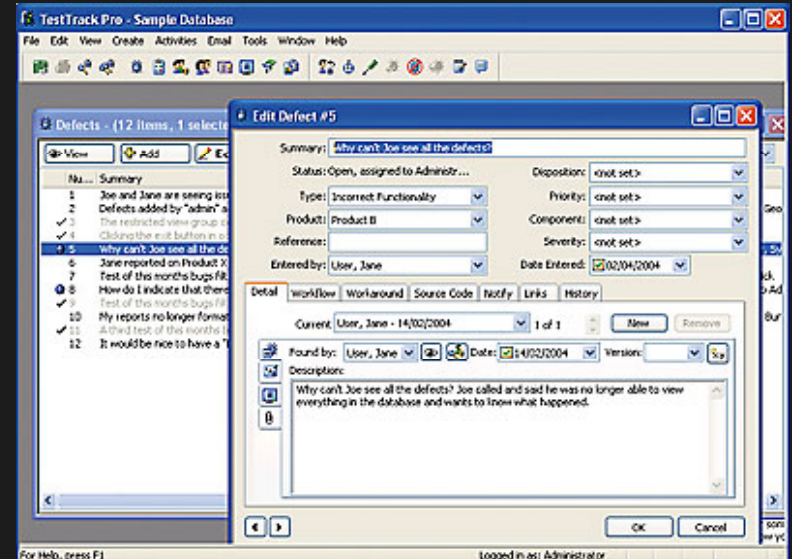
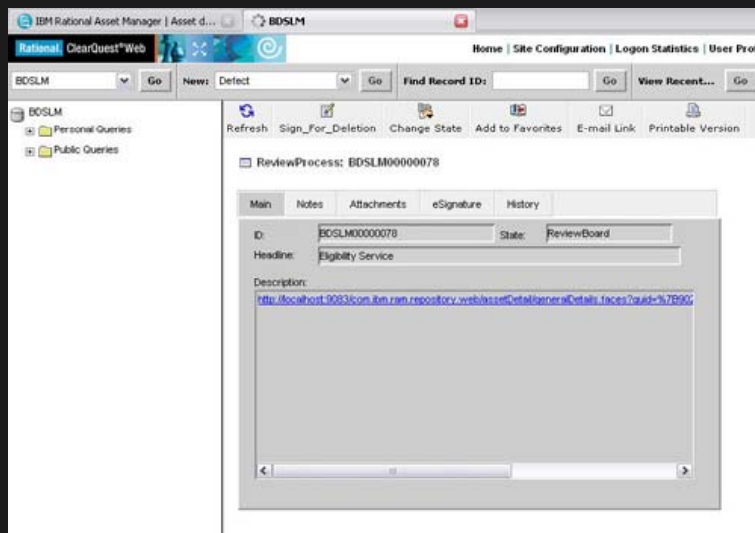
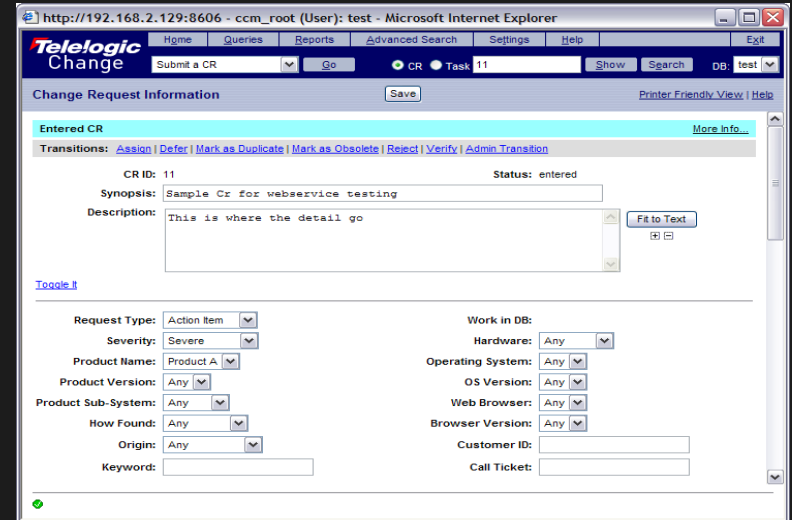
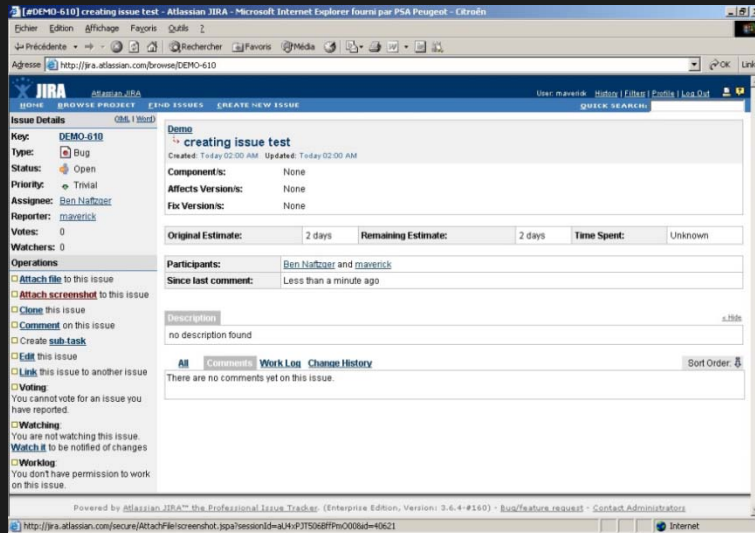
Verktøy gjør det enklere å holde oversikten over endringsforespørsler

jira

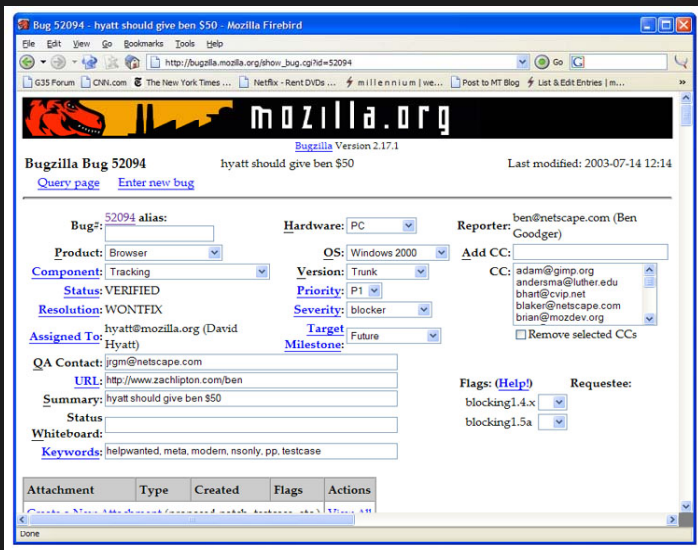
telelogic
change

clearquest

testtrack

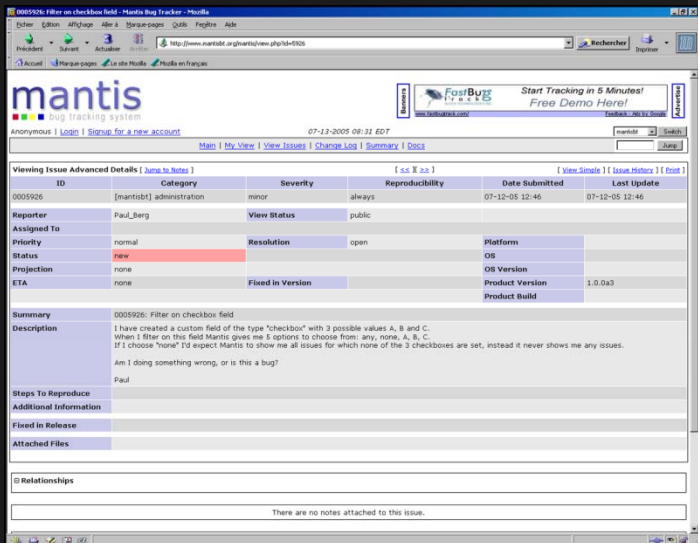
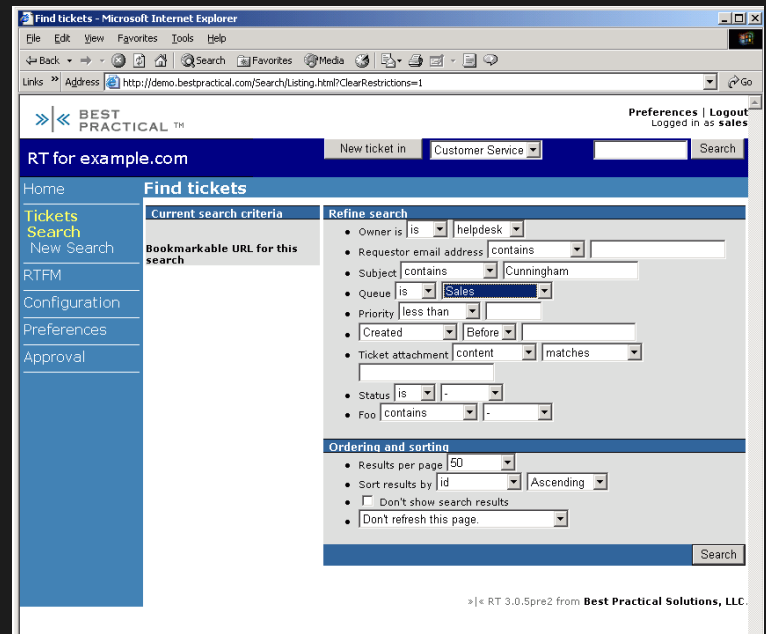


Mange gode open-source alternativer finnes



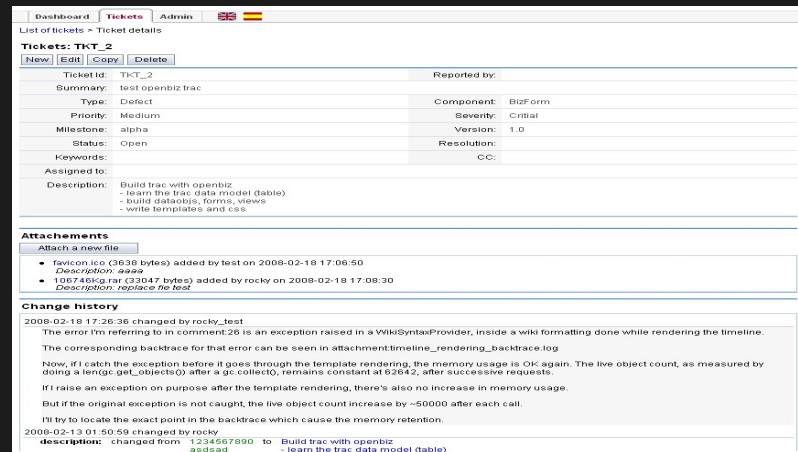
bugzilla

RT



mantis

trac



Innmelding av endringer bør ha en lav terskel

Create and Submit	
Fields marked with asterisks are required	
Area	All ▾
Responsible	(Auto) ▾
Author	Hans Christian Benestad (benestad) ▾
Synopsis*	Galt med feilen
Description* (include here specific	Hei, Det er noe galt med feilen

Innmelding direkte i verktøy gir best struktur

Innmelding via e-mail har lavere terskel for mange brukere

To...	bugzero@simula.no
Cc...	
Subject:	Galt med feilen
Hei, Det er noe galt med feilen	

Eventuelt med direkte link fra programvaren

EDIT

[ation about individuals](#)
[a the pdf externally visible on the Simula website](#)
[ance](#)
[on empirical artifact protocols](#)
[emplates](#)

[View in hierarchy](#)

[nt](#)

the Enterprise Wiki. (Version: 2.3.1 Build:#643 Jan 22, 2007) - [Bug/feature request](#) -

"Mine endringer" og fleksible søk gir god oversikt

Submit Clear

Show Columns

- Header
- Applies to
- Importance
- Description
- Issue type
- field_2
- field_3

Sort Order: Last Modified

Second Sort Order:

Show results per page 999

Ignore Closed

Arrived Date ¹ after || older than 5 day

Last Modified ¹ after || within last day

Closed Date ¹ after || within last day

Issue type

- Bug
- Question
- Change request

Applies to

- None
- All
- System A

State

- new
- Being handled
- closed

Importance

- Medium
- Low
- High

Søkekriterier

Resultat

HOME CREATE QUERY REPORTS PROJECT OPEN #

Hans Christian Benestad / edit simbid

Stored Query ale Go

Query Result (Mar 25, 2008 2:55:50 PM) Sort Order: Last Modified


ID#	Short	Contractor	Testcase#	Type	State	Last Modified
<input type="checkbox"/> 352	Mail always retrieved	Alov	Not specified	Defect	closed	Sep 15, 2005 1:04:45 PM
<input type="checkbox"/> 343	Negative bias not accepted	Alov	Not specified	Defect	closed	Sep 15, 2005 1:04:37 PM
<input type="checkbox"/> 346	The experience level should update	Alov	Not specified	Defect	closed	Sep 9, 2005 1:23:29 PM
<input type="checkbox"/> 356	<bidvalue></bidvalue>	Exiion	Not specified	Defect	closed	Sep 9, 2005 1:22:45 PM
<input type="checkbox"/> 360	More textual errors	Exiion	Not specified	Defect	closed	Sep 9, 2005 1:22:15 PM
<input type="checkbox"/> 357	Where is domain taken from?	Exiion	Not specified	Defect	closed	Sep 9, 2005 9:15:16 AM
<input type="checkbox"/> 245	Something is strange with e-mail (HTML)	Exiion	Test case 10 - Two small bidding rounds - Providers (WEB)	Defect	closed	Sep 8, 2005 9:30:51 AM

Integrasjon med versjonskontrollsystem er nyttig

Utvikler utfører en endring, og sjekker inn kode:

>> CVS commit -m "ID=1 State=KlarTilSystemTest Feil bruk av peker"

CVS oppdaterer endringsdatabase

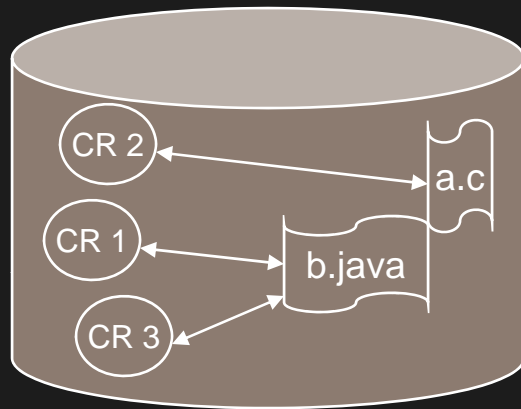


Audit Trail (Change History):	
Version#2	Author: Hans Christian Benestad, Date: May 6, 2008 11:09:42 AM State: KlarTilSystemtest, Responsible: Hans Christian Benestad
Response	Feil bruk av peker
Version#1	Author: Hans Christian Benestad, Date: May 6, 2008 11:00:52 AM State: new, Responsible: Hans Christian Benestad
Description	Fillem Programmer tryner når jeg trykker Lagre

+ Enkelt for utvikler

+ Ta vare på sammenhenger mellom logiske endringer og kodeendringer

Nyttig å vise sammenhenger mellom logiske endringer og kodeendringer



CR = Change Request

Octet string kodes feil på kortet Issue 30 of 17: a(s) [a\(s\)](#)

Created: 26. Sep 2006 09:58 Updated: 19. Dec 2006 10:17 [<< Previous](#) | [MUNA-3562](#) | [Next >>](#)

Component/s: None
Affects Version/s: MT20
Fix Version/s: None

Time Tracking:

Original Estimate:	3 days, 6 hours	<div style="width: 100%;"></div>
Remaining Estimate:	0 minutes	<div style="width: 0%;"></div>
Time Spent:	3 days, 6 hours	<div style="width: 100%;"></div>

Testbeskrivelse: Skriv og les (kjør felddump på kortnivå av) filer som inneholder octet string. Se feks holderData, nrHolderData, iai, nrContractData, eventData, nrEventData.

Description [Hide](#)

Felt av typen octet string skal kodes som et 8 bits lengdefelt og en streng bestående av et helt antall bytes. I CRSI brukes disse feltene til å lagre applikasjonsspesifikke strukturer som ikke nødvendigvis har en lengde på et helt antall bytes. Når disse strukturene leses fra eller skrives til kortet må lengden pad-es og lengde feltet må være satt iht kravene for octet string. Dette er mangelfullt implementert i vår løsning.

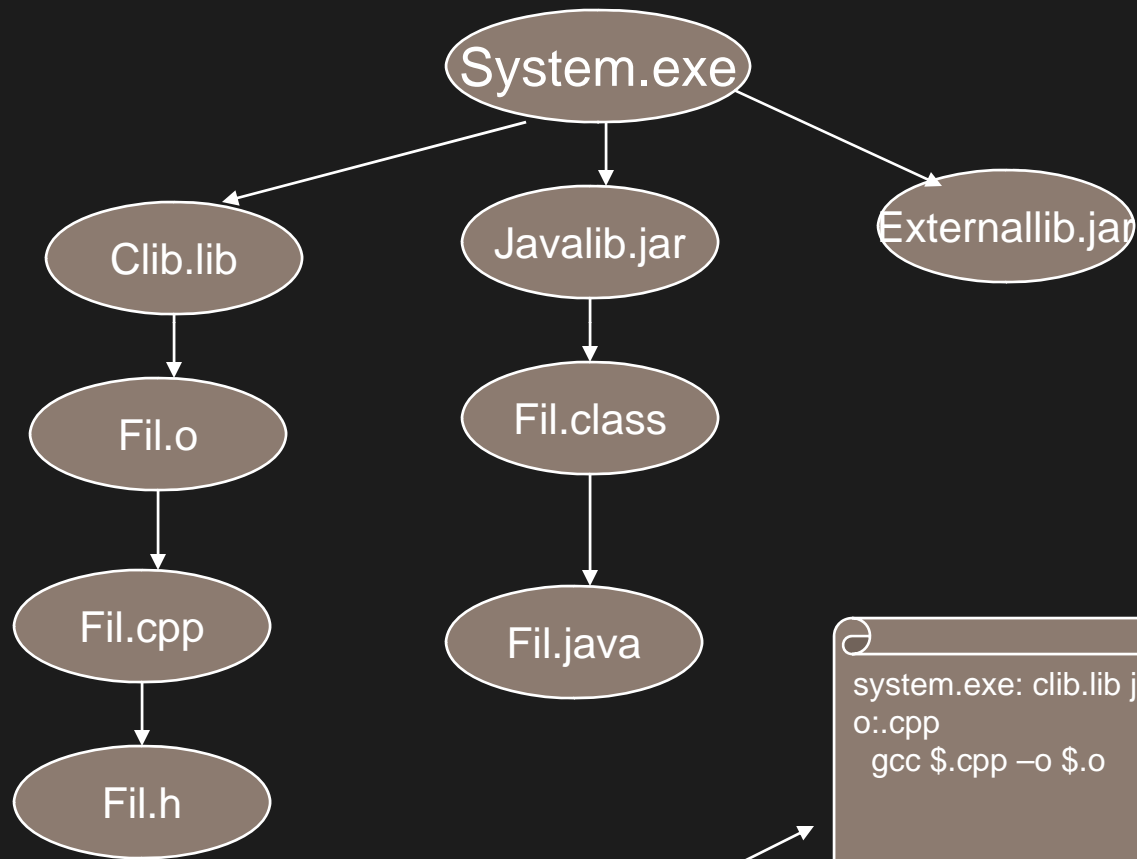
All [Comments](#) [Work Log](#) [Change History](#) [ClearCase Commits](#) [Subversion Commits](#) [Related Builds](#) [Sort Order: ↓](#)

Repository	Revision	Date	User	Message
Trunk	#3122	Tue Sep 26 11:27:29 CEST 2006	tok	Rettet feil ifm koding av octet string. MONA-3562

Files Changed

- MODIFY [/trunk/app/mtclient-jar/src/test/mona/eticketgui/ETicketGUI.java](#)
- MODIFY [/trunk/app/mtclient-dist/src/c++/eticket/fileformats.cpp](#)
- MODIFY [/trunk/app/mtclient-dist/src/c++/eticket/ARMV4Rel/eticket.dll](#)
- MODIFY [/trunk/app/mtclient-dist/src/c++/eticket/encoder.cpp](#)
- MODIFY [/trunk/app/mtclient-dist/src/c++/eticket/eticketvc0](#)
- MODIFY [/trunk/app/mtclient-dist/src/c++/eticket/convertor.cpp](#)
- MODIFY [/trunk/app/mtclient-dist/src/c++/eticket/encoder.h](#)
- MODIFY [/trunk/app/mtclient-dist/src/c++/eticket/eticket.opt](#)
- MODIFY [/trunk/app/mtclient-dist/src/c++/eticket/filestructs.h](#)
- MODIFY [/trunk/app/mtclient-dist/src/c++/eticket/Debug/eticket.dll](#)
- MODIFY [/trunk/app/mtclient-dist/src/c++/eticket/eticketvcb](#)
- MODIFY [/trunk/app/mtclient-dist/src/c++/eticket/util.h](#)
- MODIFY [/trunk/app/mtclient-dist/src/c++/eticket/util.cpp](#)
- MODIFY [/trunk/app/mtclient-dist/src/c++/eticket/eticket.ncb](#)

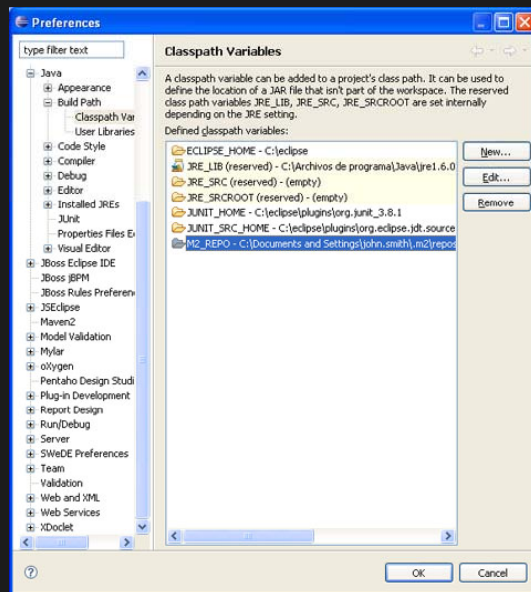
Bygging: Store programsystemer består av komponenter som utgjør en avhengighetsgraf



```
system.exe: clib.lib javalib.jar externallib.jar  
o:.cpp  
gcc $.cpp -o $.o
```

Byggeverktøy hjelper til med å regenerere komponenter ved behov

Byggeregler bør være sentralt definert

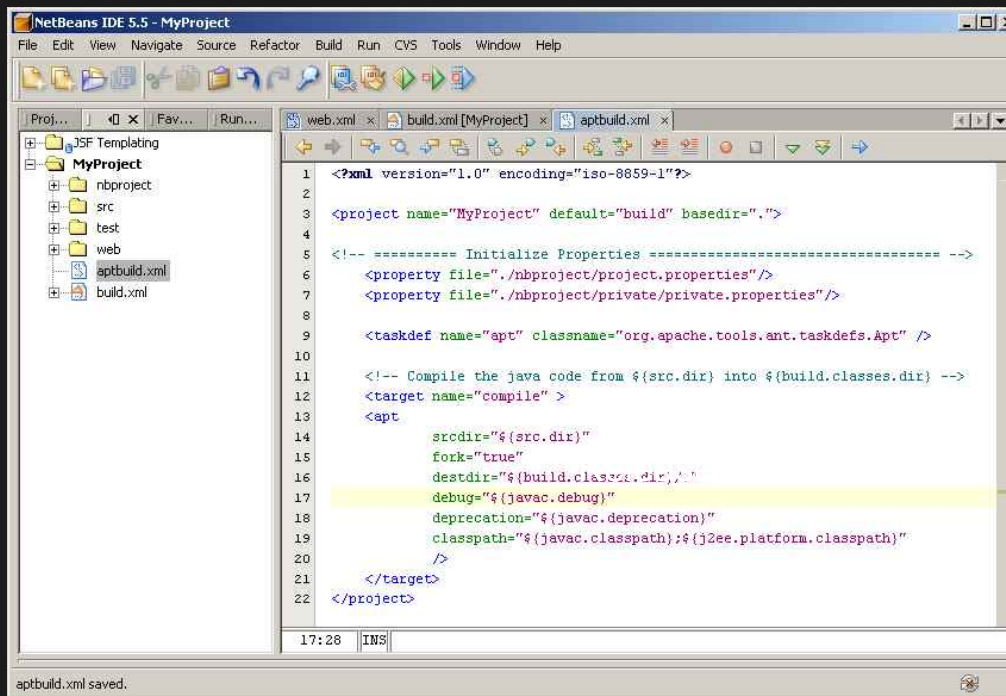


IDE'er som Eclipse oppmuntrer til å definere lokale byggeregler

Dette skalerer dårlig for større prosjekter

Byggeregler kan defineres i byggeverktøyenes beskrivelsesfiler

- Kjente verktøy: Make, Ant, Maven
- Beskrivelsesfiler spesifiserer
 - Hvilke filer inngår
 - Kompilerings- og linkeopsjoner
 - Versjoner av biblioteker og verktøy



The screenshot shows the NetBeans IDE 5.5 interface. The left sidebar displays a project tree for 'MyProject' with folders 'nbproject', 'src', 'test', and 'web', and files 'aptbuid.xml' and 'build.xml'. The main editor window shows the content of 'build.xml' with the following XML code:

```
1 <?xml version="1.0" encoding="iso-8859-1"?>
2
3 <project name="MyProject" default="build" basedir=".">
4
5 <!-- ===== Initialize Properties ===== -->
6 <property file="./nbproject/project.properties"/>
7 <property file="./nbproject/private/private.properties"/>
8
9 <taskdef name="apt" classname="org.apache.tools.ant.taskdefs.Apt" />
10
11 <!-- Compile the java code from ${src.dir} into ${build.classes.dir} -->
12 <target name="compile" >
13 <apt
14     srcdir="${src.dir}"
15     fork="true"
16     destdir="${build.classes.dir}"
17     debug="${javac.debug}"
18     deprecation="${javac.deprecation}"
19     classpath="${javac.classpath};${j2ee.platform.classpath}"
20 />
21 </target>
22 </project>
```

Lurt å la beskrivelsesfiler i seg selv være underlagt konfigurasjonsstyring

Fullt ut repeterbar bygging kan være krevende

- Krever i prinsippet konfigurasjonsstyring av
 - Kildefiler
 - Dokumentasjon
 - Eksterne/interne biblioteker
 - Generatorverkøy, som kompilatorer
 - Byggeregler

Hvem definerer og eier SCCM-systemene?

C 2.3.5 Versjonskontrollsystem

Subversion¹⁰ vil bli brukt som versjonskontrollsystem. Subversion er et open source-verktøy som er gratis i bruk og lisensieres under Apache/BSD-style open source lisens¹¹. Lisensen vil ikke ha betydning, da verktøyet bare vil benyttes under utvikling.

C 2.3.6 Feilrapporteringssystem

JIRA¹² vil bli brukt som feilrapporteringssystem og endringshåndteringssystem. JIRA kan integreres med versjonskontrollsystemet Subversion slik at man i JIRA kan se hvilke versjonerte endringer som retter feilen.

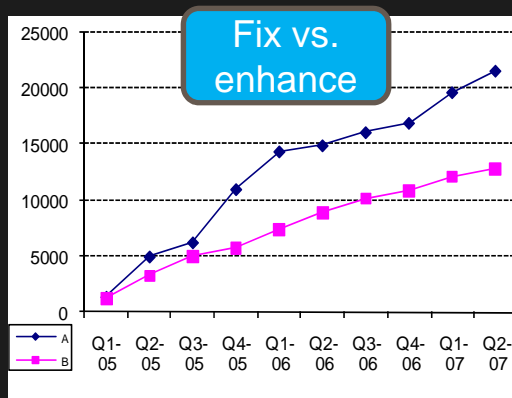
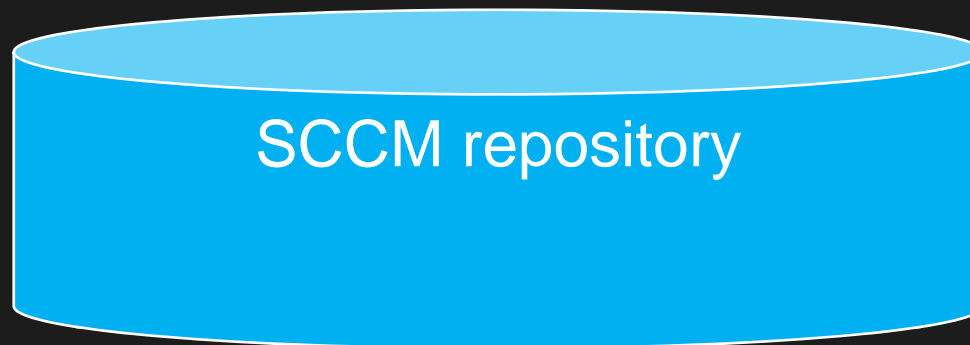
Leverandøren har lisens på JIRA som kan benyttes i dette prosjektet og i et evt. vedlikeholdsprosjekt.

Fra den utfylte
PS2000-kontrakten

Normalt vil utviklingsprosjektet styre versjonskontrollsystemet, men kunde kan og bør sette krav (se over)

Det ligger makt i å eie endringshåndteringssystemet!

Versjonsdatabaser inneholder verdifull informasjon for prosjektevaluering



- Analyse av produktivitet
- Endringsintensitet
- Antall feil, og alvorlighetsgrad
- Andel feil vs. forbedringer
- Identifisere problematiske komponenter

Prosess – og produktforbedring

Oppsummering

- God SCCM gjør livet enklere for prosjektet og utviklerne
- Valg av prosedyrer og verktøy er avhengig av behovene
- Riktig dosert så understøtter versjons- og konfigurasjonsstyring effektiv utvikling og kvalitet i leveranser

Spørsmål?

