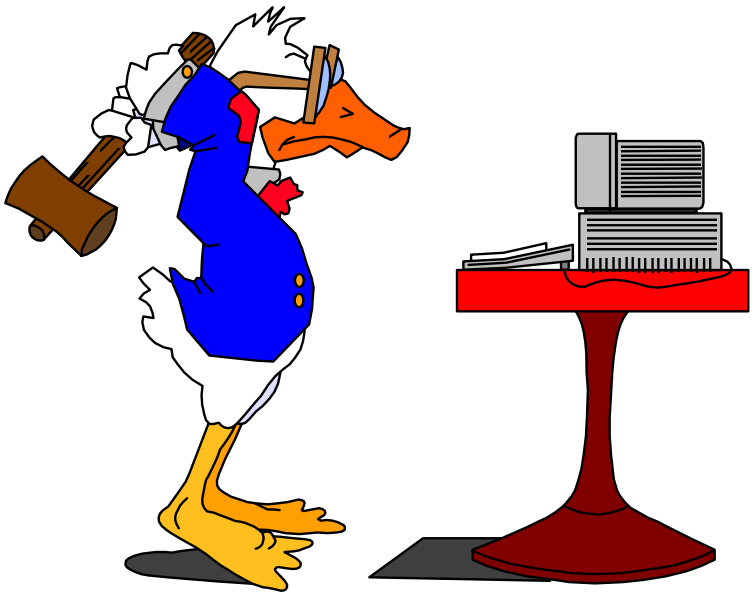
Herzlich Willkommen

Bei der Fortbildung für
RCS

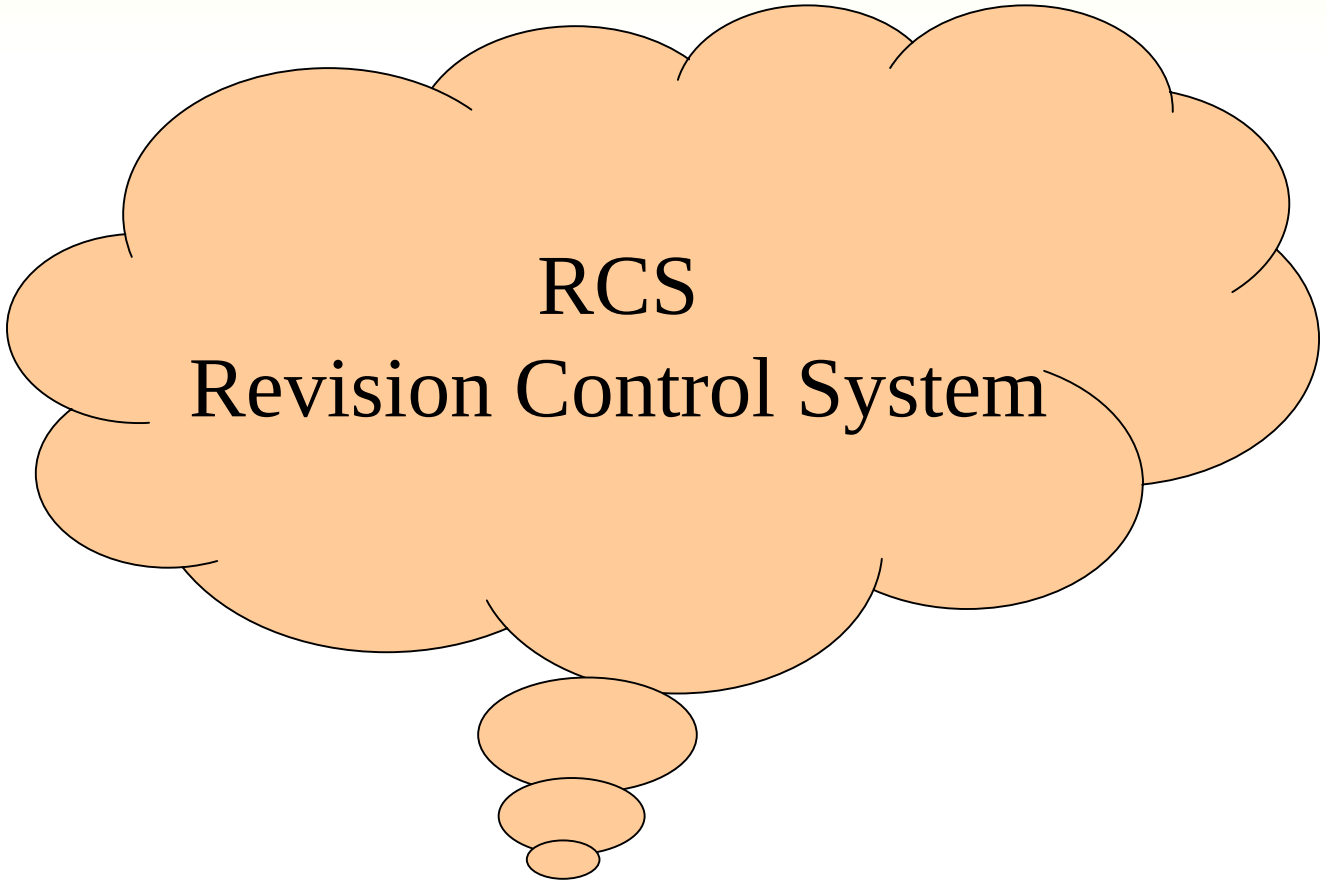
Iabc
nformation GmbH
Phone: +49 30 383983-0
http://www.Berlin.Broker.de

Die Konzentrationsfähigkeit läßt nach 40 Minuten nach!



Möchten Sie in der Mitte jedes Blockes, nach ca. 45 Minuten eine kurze Pause machen?

Iabc
nformation GmbH
Phone: +49 30 383983-0
http://www.Berlin.Broker.de



Iabc
nformation GmbH
Phone: +49 30 383983-0
http://www.Berlin.Broker.de

Zeitplan

- 1.Tag
  - Übersicht CM (Configuration Management) und SCM (Source Control Management)
  - Einführung RCS
  - Funktionen von RCS
  - Das „RCS file“
  - Ein „RCS file“ Beispiel

Zeitplan

- 2.Tag
  - Die wichtigsten RCS Befehle (check in, check out)
  - praktische Übungen
  - Ändern der „RCS file“ Attribute

Zeitplan

- 3.Tag
  - weitere RCS Befehle und Optionen
  - Übungen
  - große Übung RCS gesamt

Übersicht CM

- Configuration Management
  - Quelltextverwaltung (Source Control)
  - Abhängigkeitsprüfungen (Dependancies)
  - Entwicklungsumgebung (Build System)
  - Fehlerverfolgung (Debugger/Bug Tracking)
  - Testsystem (Automated Test Procedures)
  - Dokumentenverwaltung (Entwicklungsdok.)
  - Erstellung der Kunden-Installationsmedien

Iabc
nformation GmbH
Phone: +49 30 383983-0
http://www.Berlin.Broker.de

Übersicht SCM

- Source Control Management
  - Problemstellung
    - Teamorientierte Entwicklung
    - Fehlervermeidung
    - Fehlerverfolgung
    - Dokumentation von Änderungen
    - Gezielte Rücknahme von Änderungen

Übersicht SCM

- Source Control Management
  - Lösungsansatz
    - Festhalten einer Quelltextrevision
    - Fortschreibung mittels Delta-Informationen
    - Zugriffsverfahren mittels Locking-Mechanismus
    - Verschmelzen von Revisionsständen (Merge)

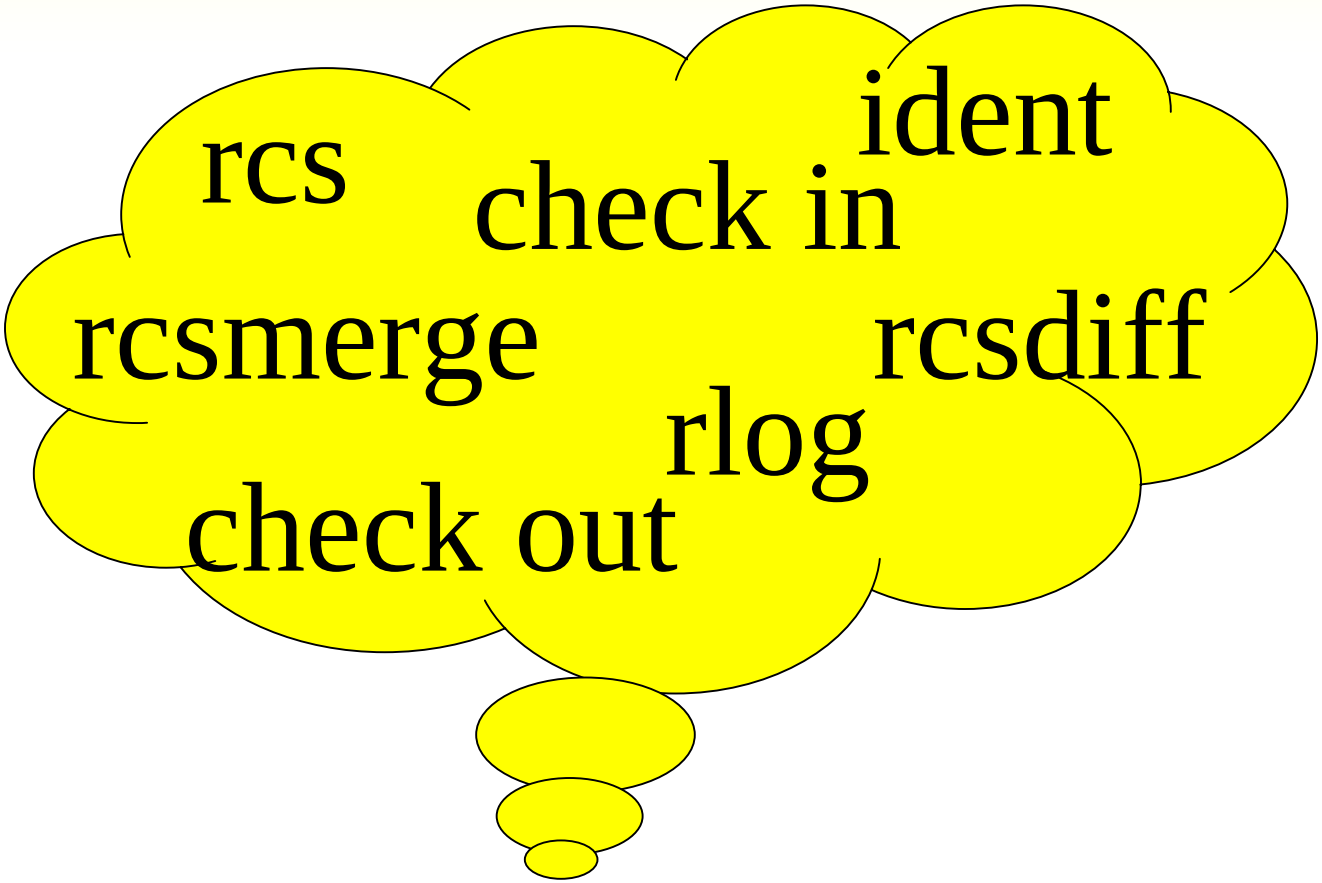
Übersicht SCM

- Source Control Management
  - Vorteile beim Einsatz eines SCM
    - „Teamfähigkeit“ des Projektes
    - Wartung und Entwicklung auf einer Quellcodebasis
    - Entwicklungs-, Release- und Testrevisionen unterscheidbar (Status)

Übersicht SCM

- Source Control Management
  - Verwendung der Begriffe Version und Revision
    - Version -> Auslieferungsstand einer Software
    - Revision -> Entwicklungsstand eines Quelltextes

RCS Revision Control System



Iabc
nformation GmbH
Phone: +49 30 383983-0
http://www.Berlin.Broker.de

Revision Control System RCS

- Das Revision Control System (RCS) verwaltet unterschiedliche Versionen von Dateien
  - automatisches Speichern von Dateien,
  - Wiederherstellen von Dateien,
  - Protokollieren (logging) von Aktionen,
  - Identifizierung von Benutzern,
  - Zusammenführung (merge) von Revisionen

Revision Control System RCS

- Einsatzgebiete
  - ASCII-Dateien
    - Texte, Programme, Dokumentation, ...
  - Binär-Dateien
    - Grafiken, formatierte Dokumente (Word etc.)

Revision Control System RCS

- Einfaches Benutzerinterface für Anwendungen
- Die zwei wichtigsten Kommandos:
  - **ci** (check in)

    sendet den Inhalt einer Datei in die RCS Archivdatei. Diese wird „RCS file“ genannt.
  - **co** (check out)

    stellt die einzelne Revision aus dem „RCS file“ wieder her

  Das „RCS file“ speichert in sich alle Revisionen, bzw. deren Veränderungen und organisatorische Daten.

Iabc nformation GmbH
Phone: +49 30 383983-0
http://www.Berlin.Broker.de

Funktionen von RCS

- Für jede verwaltete Datei existiert eine entsprechende RCS-Datei
- Speichern und Wiederherstellung unterschiedlicher Revisionen von Dateien.
- Effektive Speicherung der Revisionen durch RCS.
  - Die Revisionen werden nur mit den Änderungen zur ihrer vorherigen Version abgelegt.
  - Mit Hilfe der jeweiligen Revisionsbezeichnung kann die Datei wiederhergestellt werden.
- Revisionsnummern, symbolische Namen, Datumsangaben, Autor und Statusangaben können abgefragt werden.

Funktionen von RCS

- RCS liefert eine komplette Historie über den Werdegang der Datei.
  - Alle Veränderungen werden automatisch protokolliert. Die mit geführten Daten machen es möglich, jeden beliebigen Quelltextstand wieder herzustellen und ihn analysieren zu können. Bei jedem „check in“ (ci) werden diese Daten aktualisiert, um auch später die einzelnen Revisionen vergleichen zu können.
- Protokolliert wird:
  - der Autor
  - Datum, Zeit
  - die Revision und Systemdaten (symbolische Namen, Kommentar)

Funktionen von RCS

- RCS vermeidet Bearbeitungskonflikte bei größeren Teams. Dies kommt häufig vor, wenn mehrere Personen an einem Programmbereich arbeiten.
- Gleichzeitiger Revisionsbearbeitung wird durch RCS mittels Warnung und Rechtevergabe verhindert.
- Die einzelnen Revisionen können lesend oder lesend/schreibend zur Verfügung gestellt werden.
- Die Revisionsentwicklung kann in Form eines Baumes aus der „RCS file“ ausgelesen werden, um die Beziehungen zwischen den Revisionen darzustellen.

Iabc
nformation GmbH
Phone: +49 30 383983-0
http://www.Berlin.Broker.de

Funktionen von RCS

- Zusammenführung von Revisionen (merge)
  - RCS bearbeitet einzelne Revisionen und erkennt Konflikte untereinander.
    - Zusammenfügung unterschiedlicher Zeilen werden durch „merge“ ohne Nachfrage durchgeführt.
    - Ein Zusammenfügen gleicher Zeilen mit Änderungen an der gleichen Textstelle führt zu Überschneidungen, wovor RCS warnt. Der Bearbeiter wird durch RCS informiert, um einen Abgleich zu erreichen.

Funktionen von RCS

- Symbolische Bezeichnung der Revisionen
  - Jeder Revision mit einer Nummer, z.B. 1.3.2, kann ein Name zugeordnet werden der eine Beziehung darstellt. Dieser Name kann später an der Stelle der Revisionsnummer beim „check out“ der Datei verwendet werden.
  - Jede Revision kann einem Revisionszustand zugeordnet werden, welcher frei definierbar ist.
    - z.B. experimental
      alpha
      beta
      stabil
      ...

Funktionen von RCS

- Jede RCS Datei ist eindeutig identifizierbar
- Jede RCS Datei erhält einen eindeutigen „Stempel“ mit den Angaben:
  - Version (z.B. 1.3.2)
  - Name
  - Datum
  - Autor

  Das Textfeld ermöglicht es Angaben über den jeweiligen Zustand oder die Veränderungen zu machen. Hierdurch können später Übersichten erstellt werden.

Funktionen von RCS

- Minimaler Speicherbedarf durch RCS
- RCS speichert immer nur die letzte Version komplett im „RCS file“ ab.
- Bei den anderen Versionen werden nur die Deltas zu der letzten Version abgespeichert.
- Gelöschte Versionen werden entfernt und das „RCS file“ komprimiert.

Iabc
nformation GmbH
Phone: +49 30 383983-0
http://www.Berlin.Broker.de

Das RCS file

- In ihm werden alle Revisionen der Datei und Informationen gespeichert.
- Aufbau des „RCS file“ ist im ASCII Format wie folgt:

rcstext::	admin::
delta::	desc::
deltatext::	num::
digit::	id::
sym::	idchar::
string::	newphrase::
word::	

- Die Felder weisen auf Funktionen von RCS hin.

Die „RCS file“ Felder

rcstext ::= admin {delta}* desc {deltatext}*

admin ::= **head** {num};
{ **branch** {num}; }
access {id}*;
symbols {sym : num}*;
locks {id : num}*; {strict ;}
{ **comment** {string}; }
{ **expand** {string}; }
{ **newphrase** }*

Iabc
nformation GmbH
Phone: +49 30 383983-0
http://www.Berlin.Broker.de

Die „RCS file“ Felder

delta ::= num

date num;

author id;

state {id};

branches {num}*;

next {num};

{ newphrase }*

desc ::= **desc** string

Die „RCS file“ Felder

deltatext ::= num

log string

{ newphrase }*

text string

num ::= {digit | .}+

digit ::= **0** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9**

id ::= {num} idchar {idchar | num}*

sym ::= {digit}* idchar {idchar | digit}*

idchar ::= any visible graphic character except special

Iabc
nformation GmbH
Phone: +49 30 383983-0
http://www.Berlin.Broker.de

Die „RCS file“ Felder

special ::= **$** | **,** | **.** | **:** | **;** | **@**

string ::= **@**{any character, with **@** doubled}***@**

newphrase ::= id word* ;

word ::= id | num | string | :

Iabc
nformation GmbH
Phone: +49 30 383983-0
http://www.Berlin.Broker.de

Die „RCS file“ Felder

- Alle Felder des „RCS file“ werden automatisch durch die RCS Befehle gefüllt und bearbeitet (ci, co, rlog, etc).
- Durch neue Revisionen werden Änderungen der Revisionsnummer automatisch, die Kommentare dazu ggf. individuell vom Bearbeiter hinzugefügt.
- Deltas der Revisionen haben das Format:

  z.B. 3.1.1.1, 3.1.1.2, 3.1.1.3 oder 1.1, 1.2, 1.3, etc.
- Die Branch Nummer ist die default Revision eines Astes.

  Nachfolgende Grafik zeigt ein Beispiel eines „RCS file“

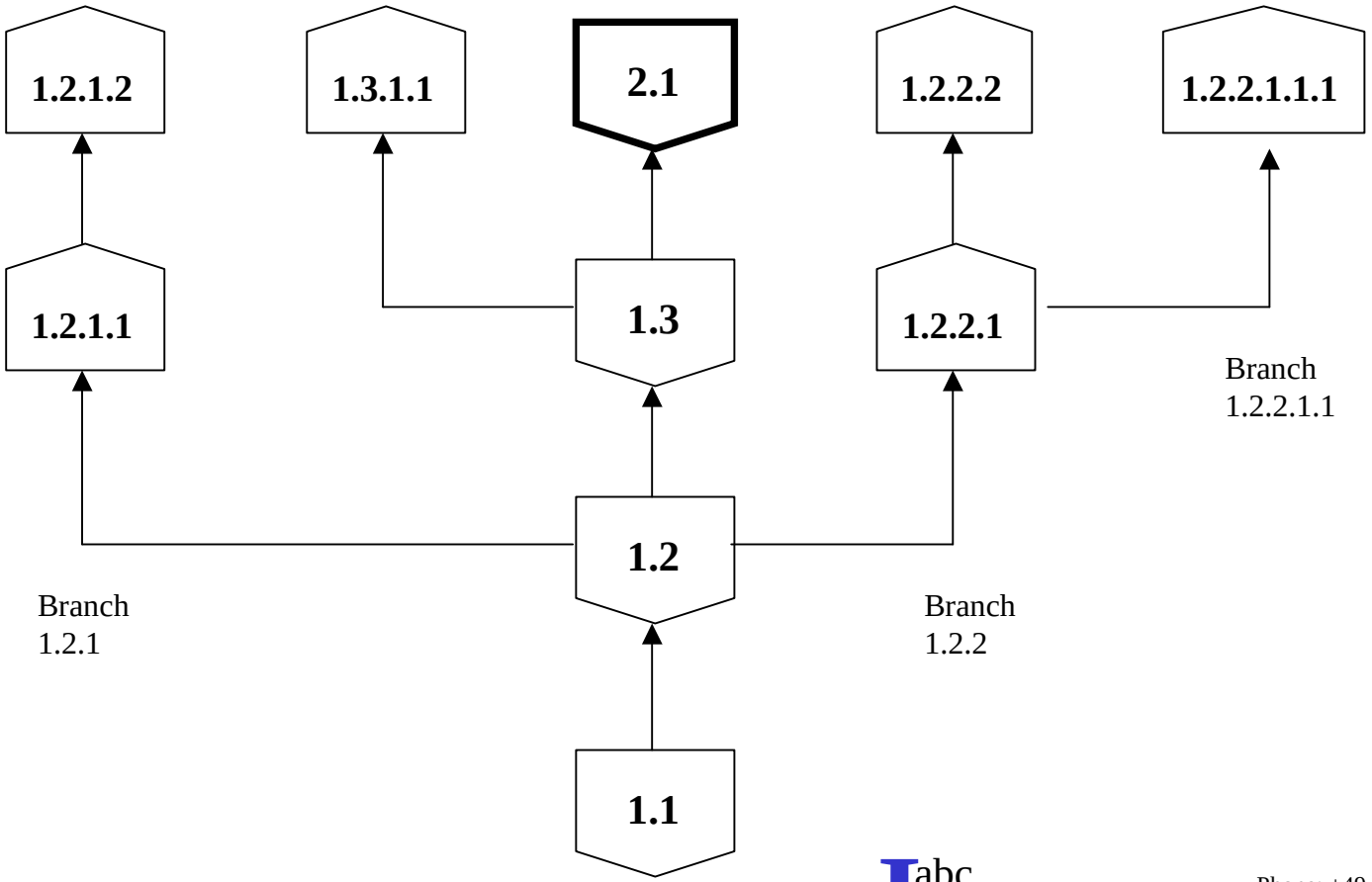
Iabc
nformation GmbH
Phone: +49 30 383983-0
http://www.Berlin.Broker.de

Ein „RCS file“ Beispiel

- Das „Head“-Feld im ersten Teil der Datei gibt den Startpunkt des „Versions“-Stammes an.
- Alle Knoten, die durch zwei Ziffern (2.2, 2.1) dargestellt werden, bilden den Versions-“Stamm“ und sind über das „Next“-Feld verbunden. (2.1->1.3->1.2->1.1)
- Deltas der einzelnen Revisionen werden über das Feld „branch“ referenziert.
- Die weiteren Knoten der jeweiligen Äste werden in aufsteigender Reihenfolge über das Feld „Next“ verbunden.

Iabc
nformation GmbH
Phone: +49 30 383983-0
http://www.Berlin.Broker.de

Ein „RCS file“ Beispiel



Iabc
nformation GmbH
Phone: +49 30 383983-0
http://www.Berlin.Broker.de

Übersicht der RCS Befehle

- **ci** Ablegen von Quelltexten in RCS-Verzeichnis
- **co** Auslesen von Quelltexten aus RCS-Verzeichnis
- **ident** Filtert RCS Schlüsselworte aus Dateien
- **rcs** Ändern der RCS Attribute
- **rcsdiff** Vergleich der RCS Revisionen
- **rcsmerge** Zusammenführung von RCS Revisionen
- **rlog** Log-Datei und Informationen ausgeben

ci - check in RCS Revisions

- Syntax

  ci [options] file ...

  – ci speichert neue Revisionen in dem „RCS file“
  – Ist ein Verzeichnis „RCS“ vorhanden, wird das „RCS file“ in ihm abgelegt.
  – Der Bearbeiter muß in der „access list“ des „RCS file“ eingetragen sein.
    - Ausnahme hierbei ist, wenn:
      die „access list“ leer ist,
      er Superuser- Rechte hat oder
      er Eigentümer des „RCS file“ ist.

ci - check in RCS Revisions

- Eine neue Revision kann nur erstellt werden, wenn die Revision im „locked“ Modus mit „check out“ in Bearbeitung ist.
- Für jede neue Revision wird eine „log message“ verlangt.

  Ende der Message mit einem Punkt (.)

  In Ihr sollten Informationen stehen, welche die Veränderungen an der Revision in Kurzform beschreiben.

  Werden mehrere Dateien gleichzeitig bei „ci“ übergeben (siehe auch die Möglichkeiten von CVS), so kann diese Information an alle „RCS files“ übergeben werden.
- Existiert kein „RCS file“, so legt der „ci“ Befehl diese automatisch an und fragt nach einer Beschreibung der Datei
  - Die default Revision ist: **1.1.**

ci - check in RCS Revisions

- An Stelle der „log message“ kann auch vorher gefertigter Text mit der Option „-t“ übergeben werden.
- Die Revisionsnummer kann als Option mit den meisten Befehlen übergeben werden, um eine eindeutige Zuordnung zu gewährleisten.

ci - check in RCS Revisions

- Die wichtigsten „check in“ (ci) Optionen:

-r[rev]	„check in“ mit Revisionsnummer „rev“
-l[rev]	„check in“ mit Revisionsnummer „rev“ + „check out“
-m[msg]	verwendet das String „msg“ für die „log message“
-s[state]	setzt den Status der Revision (default: Exp)
-t[file]	Die Datei „file“ ersetzt die Beschreibung des „RCS file“.
-T	setzt einen Zeitstempel in das „RCS file“
-w[login]	Das Autor Feld wird mit dem Namen „login“ besetzt.
-z[zone]	Angabe des Zeitformates
-n[name]	Angabe des symbolischen Namens der Revision
-N[Name]	Überschreiben der Option „-n“

co - check out RCS Revisions

- Syntax

  co [options] file ...

  – co stellt eine Revisionen aus dem „RCS file“ wieder her
  – Ist ein Verzeichnis „RCS“ vorhanden, wird das „RCS file“ aus ihm wieder hergestellt.
  – Der Bearbeiter kann die Revision „locked“ oder „unlocked“ herstellen.
  – Ist die Revision von einem anderen Benutzer in Bearbeitung, kann sie nur „locked“ im Lesemodus verwendet werden.
  – Soll eine Revision verändert und später wieder gespeichert werden (check in, ci), muß sie im „locked“ -Modus sein.

co - check out RCS Revisions

- Eine neue Revision kann nur erstellt werden, wenn die Revision im „locked“ Modus mit „check out“ in Bearbeitung ist.

- Voraussetzungen des „locked“-Modus
  - Der Bearbeiter muß in der „access list“ des „RCS file“ eingetragen sein.
    - Ausnahme hierbei ist, wenn:
      die „access list“ leer ist,
      er Superuser- Rechte hat oder
      er Eigentümer des „RCS file“ ist.

co - check out RCS Revisions

- Eine Revision kann nur durch eine eindeutige Identifizierung wiederhergestellt werden. Diese kann als Option des Befehles angegeben werden.
- Ist keine Option angegeben, wird die letzte Revision wiederhergestellt (default branch).
- Mögliche Optionen sind:
  - Revisionsnummer
  - selektierte Branch
  - Datum
  - Status
  - Autor.

co - check out RCS Revisions

- Die wichtigsten „check out“ (co) Optionen:

-r[rev]	„check out“ Revisionsnummer „rev“ oder symbol Namen
-l[rev]	„check out“ Revisionsnummer „rev“ **locked**
-u[rev]	„check out“ Revisionsnummer „rev“ **unlocked**
-s[state]	Wiederherstellung der letzten Rev. mit Status „state“
-w[login]	Wiederherstellung der letzten Rev. mit Login Namen „login“
-z[zone]	Angabe des Zeitformates
-kkv	erzeugt Schlüsselworttext z.B. $Revision: 5.13 $
-kkvl	wie -kkv, gibt des jedoch den Namen an falls „locked“

Keywords in der Datei

- In Ihrer Datei können Schlüsselwörter angegeben werden, welche durch RCS verwaltet werden.
- Diese Variablen werden beim „check out“ von RCS mit den aktuellen Daten gefüllt.

Keywords in der Datei

- Die wichtigsten Schlüsselworte (Keywords) sind:
  - $Author$: Name des Entwicklers bei „Check-In“ der 1. Revision
  - $Date$: Datum des „Check-In“ der Revision
  - $State$: Status der Revision (Release, Alpha, Beta, Exp, etc)
  - Log : Kommentierung der Änderung
  - $Revision$: Revisionsnummer
  - $Source$: Dateiname der RCS-Datei
  - $Locker$: Name des Entwicklers, welcher im Augenblick die Datei in Bearbeitung hat

Übung RCS Befehle „ci“ u.„co“ (1)

- Legen Sie in Ihrem Arbeitsverzeichnis das Verzeichnis „RCS“ an.
- Erzeugen sie eine mehrzeilige Textdatei mit dem Namen „uebung01.c“
- Erzeugen Sie eine „RCS-Datei“ mit „check in“
  - Sie sollen diese Datei weiterhin bearbeiten können
  - machen Sie Angaben über den Inhalt der Datei
- Versuchen Sie, die Textdatei erneut einzu-“checken“
- Ändern Sie jetzt die Textdatei
- Stellen Sie die geänderte Datei wieder ein.

Übung RCS Befehle „ci“ u.„co“ (2)

- Holen Sie Ihre Datei „uebung01.c“ aus der „RCS-Datei“ ohne Bearbeitungsrecht (nur lesen)
  - Welche Informationen werden Ihnen geliefert ?
  - Welche Revision hat Ihre Datei?
- Holen Sie sich das Bearbeitungsrecht für die Datei „uebung01.c“ (Lock)
- Ändern Sie die Datei und stellen Sie sie nochmals ein.
- Geben Sie die Datei „uebung01.c“ wieder frei.
  - Welche Revision hat Ihre Datei jetzt?

Übung RCS Befehle „ci“ u.„co“ (3)

- Holen Sie jetzt die Revision 1.2 zur Bearbeitung
  - Der Inhalt sollte wieder den alten Stand besitzen.
- Ändern Sie die Datei und stellen Sie sie wieder ein
  - Welche Revision schlägt Ihnen das RCS vor?
- Geben Sie Ihre Version der Datei „uebung01.c“ wieder frei
- Stellen Sie Ihre Datei „uebung01.c“erneut aus der „RCS-Datei“ zur Bearbeitung wieder her.
  - Welche Informationen werden Ihnen jetzt geliefert ?
  - Welche Änderung bei der Revisionsnummer hat Ihre Datei jetzt?

Übung RCS Befehle „ci“ u.„co“ (4)

- Wo befindet sich Ihr „RCS file“ (uebung01.c,v) ?
  - Lassen Sie sich diese Datei anzeigen (z.B. more uebung01.c,v)
    - Erkennen Sie Strukturen wieder ?
    - Versuchen Sie, anhand der Daten eine Baumstruktur zu zeichnen.
- Stellen Sie bitte Ihre Datei mit der Revision 1.1 wieder her.
- Versuchen Sie ein „check-out“ aus dem „RCS file“ eines anderen Benutzers.

Übung RCS Befehle „ci“ u.„co“ (5)

- Sehen Sie auf den Man-Pages von ci nach der Möglichkeit, einer Datei einen symbolischen Namen zu geben.
- Setzen Sie einen symbolischen Namen bei Ihrer letzten Revision.
- Setzen Sie einen Status (z.B. alpha) bei Ihrer letzten Revision.
- Stellen sie Ihre Datei mit dem symbolischen Namen wieder her.
- Bitte wiederholen Sie dies mit dem Status.

Übung RCS Schlüsselwörter (1)

- Geben Sie in Ihre Datei „uebung01.c“ einige Schlüsselwörter ein. Beachten Sie bitte den Syntax „$Keyword$“. (Hinweis: Versuchen Sie bitte **Log** .)
- Speichern Sie mit „check-in“ und stellen Sie wieder mit „check-out“ her.
  - Was sehen Sie in Ihrer Datei ?
- Probieren Sie jetzt einige andere Schlüsselwörter innerhalb des Textes und testen Sie die Ergebnisse mittels „check-in“.
  - Diskutieren Sie die Praxistauglichkeit der Schlüsselworte

ident - identifiziere RCS Schlüsselwörter (1)

- Syntax

  ident [-q] [-V] [file ...]

  – **ident** sucht nach allen Instanzen mit dem Syntax $Schlüsselwort:text$ in der Datei „file“
  – **ident** arbeitet bei Textdateien aber auch bei Objektdateien

- Die ident Optionen:

  -q Unterdrückung von Warnhinweisen

  -V Ausgabe der ident Software Version

ident - identifiziere RCS Schlüsselwörter (2)

- Schlüsselworte:
  - $Author$ Autor, Login Name des Benutzers
  - $Date$ Datum und Zeit des Revision „check in“
  - $Header$ Pfad, Rev., Datum/Zeit, Autor, Status, lock
  - Log „log messages“
  - $Name$ symbolischer Name
  - $RCSfile$ „RCS file“ ohne Pfadangaben
  - $Revision$ Revisionnummer
  - $Source$ Voller Pfadname des „RCS file“
  - $State$ Status der Revision

Einrichten des RCS für Mehrbenutzerbetrieb

- Betrieb über Zugriffsrechtevergabe unter Unix
  - Einrichten der Entwickler in einer Benutzergruppe
  - Einrichten des RCS-Verzeichnisses mit rwx für User und rw für Gruppen
  - Problem: Wenn Entwickler in verschiedenen Gruppen eingerichtet sind, können Zugriffsprobleme bei neuen Dateien auftreten.
  - Zusätzlich ist es möglich, daß einzelne Entwickler Dateien löschen oder direkt verändern können.

Einrichten des RCS für Mehrbenutzerbetrieb

- Betrieb über setuid unter Unix
  - Einrichten eines RCS-Administrators (NICHT root !)
  - Legen Sie ein Verzeichnis für von Entwicklern ausführbare Dateien an
  - Kopieren Sie das ci und co Programm als RCS-Administrator in dieses Verzeichnis
  - Führen Sie chmod go-w,u+s für ci und co aus
  - Tragen Sie dieses Verzeichnis bei allen Entwicklern als ersten Suchpfad für Befehle ein.
  - Legen Sie das RCS-Verzeichnis als RCS-Administrator an und vergeben Sie nur Lese und Schreibrechte für den Owner und Leserechte für die Entwicklergruppe.

Iabc
nformation GmbH
Phone: +49 30 383983-0
http://www.Berlin.Broker.de

rcs - Ändern der „RCS file“ Attribute (1)

- Syntax

  rcs option file ...

  – **rcs** erzeugt neue „RCS files“

  – **rcs** ändert die Attribute bestehender „RCS files“

    - Ein „RCS file“ beinhaltet unterschiedliche Revisionen der Datei, die Zugriffsliste (access list), das Veränderungs-Log, die Beschreibung der Datei und weitere Kontrollattribute.

  – Der Bearbeiter muß in der „access list“ des „RCS file“ eingetragen sein.

    - Ausnahmen sind, wenn:
      die „access list“ leer ist, er Superuser- Rechte hat oder
      er Eigentümer des „RCS file“ ist.

rcs - Ändern der „RCS file“ Attribute (2)

- Die wichtigsten „rcs“ Optionen:

-i	Anlegen und Initialisierung eines neuen „RCS file“
-a[login]	Anhängen von Namen an die „access list“
-e[login]	Löschen des Namen von der „access list
-b[rev]	Setzen der default Branch
-c[string]	Setzen eines Kommentares
-l[rev]	„lock“ der Revision mit der Nummer „rev“
-u[rev]	„unlock“ der Revision mit der Nummer „rev“
-m[rev:msg]	Ersetzt Revision „rev“ mit der Log-Meldung „msg“
-n[name:rev]	Verbinde Revision mit symbolischen Namen
-N[name:rev]	wie „-n“, jedoch mit Überschreiben des Namen

rcs - Ändern der „RCS file“ Attribute (3)

- Weitere „rcs“ Optionen:

  -o[rev1:rev2] löscht Revision „rev1“ bis „rev2“

  -s[state:rev] setzt Status „state“ der Revision „rev“

  -t[file] Die Datei „file“ ersetzt die Beschreibung des „RCS file“.

  -V Version vom „rcs“ Befehl

Übung RCS Befehl „rcs“ (1)

- Ziel der Übung ist es, die Teamarbeit mit RCS zu verdeutlichen.
- Ziel der Übung ist es, die Attribute des „RCS file“ zu verändern.
  - Erlauben Sie Ihrem Nachbarn die Bearbeitung Ihrer Datei „text01.c“.
  - Bearbeiten Sie selbst die Datei Ihres Nachbarn durch das Hinzufügen einer Zeile in der „checked out“ Datei „text01.c“.
    - Achtung: ggf. müssen die UNIX Rechte auf „read“ gesetzt werden ! (chmod o+rwx text01.c)
  - Schreiben Sie mit „check in“ die Datei wieder zurück.

Übung RCS Befehl „rcs“ (2)

- Sehen Sie sich an, was Ihr Nachbar mit Ihrer Datei gemacht.
- Ändern Sie den Status Ihrer Revision 1.1 auf „open“.
- Geben Sie in Ihre Revision 1.1 einen neuen Kommentar ein.
- Löschen Sie den Namen Ihres Nachbarn von der „access list“.
- Was geschieht, wenn Ihr Nachbar jetzt auf Ihre Datei zugreift und Sie zurückschreiben will?
- Setzen Sie Ihre Datei in den „locked“ -Modus.

Iabc
nformation GmbH
Phone: +49 30 383983-0
http://www.Berlin.Broker.de

Übung RCS Befehl „rcs“ (2)

- Was geschieht, wenn Sie Ihre Datei jetzt zurückschreiben wollen?
  - Dieser Zustand kann vorkommen, wenn eine Arbeit nicht ordnungsgemäß beendet wurde.
- Stellen Sie den Zustand her, damit Ihre Datei rückschreibbare Änderungen akzeptiert .
- Welche RCS Version wird verwendet ?

rcsdiff - RCS Revisionvergleich (1)

- Syntax

  rcsdiff [-**k**subst] [-**q**] [-**r**rev1 [-**r**rev2]] [-**T**] [-**V**[n]] [-**x**suffixes] [-**z**zone] [diff options] file ...

  – **rcsdiff** verwendet die Betriebssystemsoftware „diff“, um zwei Revisionen einer Datei zu vergleichen.
  – Die Namensangabe erfolgt wie beim „ci“ Befehl.
  – Als Ergebnis liefert „rcsdiff“ die Differenz der Textdateien auf dem Bildschirm mit weiteren Informationen der erfolgreichen oder fehlerhaften Bearbeitung.
  – Die Differenz kann durch ein Umleiten der Standardausgabe in eine Datei umgeleitet werden: „rcsdiff [optionen] file >file.diff“

rcsdiff - RCS Revisionsvergleich (2)

- Werden die Optionen „rev1“ und „rev2“ nicht angegeben, vergleicht „rcsdiff“ die letzte Revision (der default Branch), mit dem Inhalt der Arbeitsdatei.
  - Dies kann verwendet werden, um die Änderungen seit dem letzten „check in“ zu ermitteln.
- Wird „rev1“ und nicht „rev2“ angegeben, vergleicht „rcsdiff“ die Revision „rev1“ mit dem Inhalt der Arbeitdatei.
- Bei Angabe von „rev1“ und „rev2“ werden deren beide Inhalt verglichen.
- Die Angabe „rev1“ und „rev2“ kann nummerisch (z.B. 1.1.2) oder als symbolischer Name erfolgen.

rcsdiff - RCS Revisionsvergleich (3)

- Die wichtigsten „rcsdiff“ Optionen:

-q	keine Diagnoseausgabe
-r[rev]	Revisionsangabe (keine, eine oder zwei), z.B. rcsdiff -1.1 -r2.5 hallo.c
-z[zone]	Angabe des Zeitformates
-kkv	erzeugt Schlüsselworttext z.B. $Revision: 5.13 $
-kkvl	wie -kkv, gibt des jedoch den Namen an falls „locked“

- Die weiteren Optionen werden wie beim „check in“ (ci) und „check out“ verwendet

Übung RCS Befehl „rcsdiff“ (1)

- Stellen sie Ihre letzte Revision wieder her.
- Verändern Sie durch Veränderung einer Zeile und Hinzufügen einer Zeile Ihre Arbeitsdatei „text01.c“
- Stellen Sie Ihre Veränderungen gegenüber dem letzten „check out“ fest und speichern sie dies in der Datei „text01.c.diff01“.
- Erzeugen Sie ein Update von Ihrer Revision 1.1 nach 1.2 der Datei „text01.c“ und speichern das Update in der Datei „text01.c.diff02“
  - Sie haben ein Update erzeugt, welches sie zum patchen versenden können !

rcsmerge - Zusammenführung von RCS (1)

- Syntax

  rcsmerge [options] file ...

  – **rcsmerge** verwendet die Änderungen von zwei Revisionen eines „RCS file“ um sie zusammenzuführen und in der derzeitigen Arbeitsdatei einzuarbeiten.
  – Die Dateiangaben erfolgen wie bei „ci“ Befehl
  – Wird als Option nur eine Revision angegeben, wird als zweite die die letzte Revision (der default Branch) verwendet.
  – Rcsmerge warnt für undefinierten Zuständen, wie es bei Überschneidungen in Zeilen der Datei vorkommen kann.
    - Diese können nur manuell durch den Menschen gelöst werden.

rcsmerge - Zusammenführung von RCS (2)

- Die wichtigsten „rcsmerge“ Optionen:

-A,-E,-e	Ausgabekonflikte werden formatiert
-kkv	erzeugt Schlüsselworttext z.B. $Revision: 5.13 $
-kkvl	wie -kkv, siehe „co“
-r[rev]	Revisionsangabe (eine oder zwei), z.B. rcsmerge -1.1 -r2.5 hallo.c
-p[rev]	Augabe auf die Standardausgabe
-z[zone]	Angabe des Zeitformates

Übung RCS Befehl „rcsmerge“ (1)

- Verändern Sie Ihre letzte Revision und speichern Ihre geänderte Datei unter der neuen Revisionsnummer 2.1.
- Stellen Sie Ihre Revision 1.2 wieder her und führen Sie diese mit der Revision 2.2 zusammen, nachdem Sie zwei Textzeilen hinzugefügt haben.
  - Was ist mit Ihrer Arbeitsdatei „text01.c“ geschehen?
- Wiederholen Sie bitte den letzten Punkt aber senden das Ergebnis der Zusammenführung in die Datei „text01.c.merge01“.
  - Was ist jetzt der Unterschied ?

rcslog - Ausgabe RCS Log (1)

- Syntax

  rcslog [options] file ...

  – **rcslog** gibt die Informtionen des „RCS file“ aus
  – rcslog kann folgende Informationen ausgeben:
    - RCS Pfadname, Arbeitspfadname,
    - RCS Kopf, default Branch,
    - Zugriffsliste (access list), Lock Name(n), symbolischer Name,
    - selektierte Revisionsnummer,
    - Beschreibung des „RCS file“
      – Die Angaben erfolgen chronologisch

rcslog - Ausgabe RCS Log (2)

- Für jede Revision erfolgen die Angaben:
  - Revisionsnummer, Autor
  - Datum/ Zeit,
  - Status,
  - Anzahl der geänderten/ gelöschten Zeilen,
  - Lock Status der Rev.,
  - Bemerkungen zu der Revision
- Wird „rlog“ ohne Optionen verwendet, werden alle Informationen ausgegeben,.

rcslog - Ausgabe RCS Log (3)

- Die wichtigsten „rcslog“ Optionen:

-L	Ignoriere „RCS files“ die nicht „locked“ sind
-R	Name des „RCS file“
-N	ohne symbolische Namen
-ddates	Logs vom Datum /Datumbereich
-r[rev]	Informationen über Revision „rev“
-sstates	Statusangaben
-w[logins]	Informationen der Bearbeitet mit Name(n) „login(s)“
-v	rcslog Version

Iabc
nformation GmbH
Phone: +49 30 383983-0
http://www.Berlin.Broker.de

Übung RCS Befehl „rcslog“ (1)

- Lassen Sie sich den Dateinamen des „RCS-file“ von Ihrer Arbeitsdatei „text01.c“ anzeigen.
- Wie lauten die Informationen Ihrer Revision 1.1 (text01.c)
- Wie lauten die Statusangaben der Revision 1.1, 1.2 u.2.1
- welche rcslog Version verwenden Sie
- Der ganze Entwicklungszyklus Ihrer Arbeitsdatei soll angesehen werden.
- Wie sieht der Entwicklungszyklus bei ihrem Nachbarn aus?

Übung RCS gesamt (1)

- Ziel soll es sein, die Teilprodukte eines jeden Teilnehmers zu integrieren, um ein Produkt zu erhalten!
- Eine gemeinsame Arbeit und Abstimmung ist hierfür als Team notwendig.
- Aufgabenstellung:
  - Eine gemeinsame Datei (Teilnehmer.c) wird erstellt, in der in jeder Zeile ein Teilnehmername steht. Diese Datei existiert nur einmal (Rev.1.1).
  - In der Revision 1.1 soll jeder Teilnehmer zu seinem Namen seinen Arbeitsort hinzufügen durch dezentrale lokale Bearbeitung.
  - Die einzelnen Revisionen (1.1.1.1, 1.1.2.1, 1.1.3.1 etc.) sollen zusammengeführt werden und die Revision 1.2 bilden.

Übung RCS gesamt (2)

- Die Revision 1.2 soll nun von jedem Teilnehmer um seine Telefonnummer erweitert werden.
- Die einzelnen Revisionen (1.2.1.x, 1.2.2.x, 1.2.3.x etc.) sollen zusammengeführt werden und die Revision 1.3 bilden.
- Erstellen sie einen Patch des Produktes von Revision 1.2 nach 1.3.

Hinweis: Arbeiten Sie bitte gemeinsam!