

Borland[®] StarTeam[®] 2009

StarTeam Command-line Tools-Hilfe

Borland[®]

Borland Software Corporation
8310 N Capital of Texas Hwy, Bldg 2, Ste 100
Austin, Texas 78731 USA
www.borland.com

Die Borland Software Corporation verfügt möglicherweise über Patente und/oder anhängige Patentanmeldungen, die den Inhalt dieser Dokumentation betreffen. Eine Liste vorhandener Patente finden Sie auf der Produkt-CD im Dialogfeld "Info". Durch die Bereitstellung dieses Dokuments erhalten Sie keine Lizenz für diese Patente.

Copyright © 1995–2009 Borland Software Corporation und/oder deren Tochterunternehmen. Alle Marken- und Produktnamen von Borland sind Marken oder eingetragene Marken der Borland Software Corporation in den USA und anderen Ländern. Alle anderen Marken sind das Eigentum ihrer jeweiligen Inhaber.

Juni 2009
PDF

Befehlszeilen-Tools

Referenz	7
Befehlszeilenoperationen	8
Befehlszeilenooptionen des Dienstprogramms 'Bulk Check-out' (BCO)	9
Befehlszeilenooptionen für das Checkout Trace-Dienstprogramm	18
Vault Verify - Befehlszeilenooptionen	20
Weitere Client-Befehlszeilenoperationen	23
Allgemeine Optionen	25
Sonderzeichen	34
Exit-Codes	35
Dateien hinzufügen: stcmd add	36
Ordner hinzufügen: stcmd add-folder	38
Projekte hinzufügen: stcmd add-project	40
Ansichten hinzufügen: stcmd add-view	43
Labels übernehmen: stcmd apply-label	45
Dateien einchecken: stcmd ci	47
Dateien auschecken: stcmd co	50
Dateirevisionen vergleichen: stcmd diff	54
Dateibeschreibungen ändern: stcmd dsc	57
Labels erstellen: stcmd label	58
Arbeitsordner erstellen: stcmd local-mkdir	60
Lokale Dateien löschen: stcmd delete-local	61
Dateihistorien anzeigen: stcmd hist	63
Dateien auflisten: stcmd list	64
Dateien sperren und entsperren: stcmd lck	66
Server sperren und entsperren: stcmd server-mode	68
Dateien entfernen: stcmd remove	70
Persönliche Optionen anpassen: stcmd set-personal-options	71
Dateistatus aktualisieren: stcmd update-status	73
VCM-Befehlszeilenprogramm	75
Übersicht über das VCM-Befehlszeilendienstprogramm (VCMUtility)	76
VCMUtility-Befehle	80
VCMUtility-Verbindungsoptionen	83
Sitzungsoptionen von VCMUtility	85
Weitere Optionen des VCM-Dienstprogramms	98
Beispiele für das VCM-Dienstprogramm	99
Befehlsreferenz	101
Syntax der zusammengesetzten Optionen des VCM-Dienstprogramms	105
<Aktion>	106
<Auscheckoptionen>	107
<Änderungsanforderungen>	109
<Dateien>	110
<Ordner>	112
<Elementtyp>	114
<Bedingungen für Übereinstimmung>	115
<Prozesselement>	117
<Anforderungen>	118
<Revisions-Labels>	119
<Tasks>	120
<Zeitstempel>	121
<Themen>	122



Befehlszeilen-Tools

In diesem Abschnitt werden die verschiedenen Befehlszeilen-Tools dokumentiert.

In diesem Abschnitt

[Referenz](#)

Dieser Abschnitt enthält alle Referenzthemen.

Referenz

Dieser Abschnitt enthält Referenzthemen.

In diesem Abschnitt

[Befehlszeilenoperationen](#)

Dieser Abschnitt enthält Referenzthemen, die sich mit Befehlszeilenoperationen befassen.

[Weitere Client-Befehlszeilenoperationen](#)

Dieser Abschnitt enthält Referenzthemen, die sich mit Client-Befehlszeilenoperationen befassen.

[VCM-Befehlszeilenprogramm](#)

Erläutert das Befehlszeilenprogramm zum Vergleichen und Zusammenführen von Ansichten (`VCMUtility`).

Befehlszeilenoperationen

Dieser Abschnitt enthält Referenzthemen, die sich mit Befehlszeilenoperationen befassen.

In diesem Abschnitt

[Befehlszeilenoptionen des Dienstprogramms 'Bulk Check-out' \(BCO\)](#)

In diesem Thema werden die Befehlszeilenoptionen für das Bulk Check-out-Dienstprogramm beschrieben.

[Befehlszeilenoptionen für das Checkout Trace-Dienstprogramm](#)

In diesem Thema werden die Befehlszeilenoptionen für das Checkout Trace-Dienstprogramm beschrieben.

[Vault Verify - Befehlszeilenoptionen](#)

In diesem Thema werden die Befehlszeilenoptionen für das Vault Verify-Dienstprogramm beschrieben.

Befehlszeilenoptionen des Dienstprogramms 'Bulk Check-out' (BCO)

Dieses Thema enthält folgende Informationen:

- ◆ Syntax
- ◆ Befehlszeilenoptionen
- ◆ BCO-Anwendungsbeispiele

Syntax

Für das Dienstprogramm "Bulk Check-out" (BCO) gilt die folgende Syntax:

```
-p "Projektangabe" [-pwdfile "Dateipfad"] [-autoLogon] [-cwf] [-is] [-rp "Ordnerpfad" | -fp "Ordnerpfad" ] [-cmp] [-dryrun] [-vb] [-useCA] [-encrypt Verschlüsselungstyp] [-cfgl "Label-Name" | -cfgp "Statusname" | -cfgd "Datum"] [-filter "Dateistatus"] [-o] [-ro] [-ts] [-fs] [-eol on | off | cr | lf | crlf ] [-exclude <Muster> | @<Musterdatei>] [-netmon] [-t] [-h | -help] [Dateien...]
```

Optionen

Die folgende Tabelle enthält die BCO-Optionen.

Option	Beschreibung
<code>-autoLogon</code>	Wenn mit der Option <code>-p</code> kein Benutzername angegeben ist, wird unter Verwendung der Benutzer-ID und des Passworts beim angegebenen Server eine Anmeldung ausgeführt, wie im Toolbar-Dienstprogramm konfiguriert. Dies ist nur auf Windows-Betriebssystemen möglich.
<code>-cfgd</code>	Konfiguriert die Ansicht mit den angegebenen Datums-/Uhrzeitangaben. Folgende Beispiele sind u. a. möglich: <ul style="list-style-type: none">■ "12/29/01 10:52 AM"■ "December 29, 2001 10:52:00 AM PST"■ "Montag, 3. Januar 2005 11:25 Uhr CET"
<code>-cfgl</code>	Konfiguriert die Ansicht mit dem angegebenen Label. Ohne Angabe von <code>-cfgl</code> , <code>-cfgp</code> oder <code>-cfgd</code> verwendet BCO die aktuelle Konfiguration für die Ansicht.
<code>-cfgp</code>	Konfiguriert die Ansicht mit dem angegebenen Heraufstufungsstatus.
<code>-cmp</code>	Komprimiert alle zwischen der Workstation und dem Server ausgetauschten Daten und dekomprimiert eingehende Daten. Ohne die Option erfolgt keine Komprimierung. Die Komprimierung ist besonders hilfreich, wenn der Client und der Server über eine langsame Verbindung kommunizieren. Mithilfe eines kleinen Testfalls können Sie feststellen, ob die Verwendung der Komprimierung empfehlenswert ist. Dabei sollten Sie prüfen, ob es vorteilhafter für Sie ist, Zeit für die Komprimierung bzw. Dekomprimierung der Daten aufzuwenden oder längere Zeiten zum Senden unkomprimierter Daten über eine langsame Verbindung in Kauf zu nehmen.

<code>-cwf</code>	Erstellt den Arbeitsordner für den Basisordner und bei Angabe der Option <code>-is</code> alle Unterordner des Basisordners, selbst wenn während der Ausführung keine Dateien ausgecheckt werden müssen. Es werden nur sichtbare Ordner erstellt.
<code>-csf</code>	<p>Wenn der Befehl den Ordner, der mit der Option <code>-p</code> angegeben wird, dem zugrunde liegenden StarTeam-Ordner zuordnet, können Sie mithilfe der Option <code>-csf</code> die StarTeam-Ordner anhand der Unterschiede in der Groß-/Kleinschreibung auseinander halten. (Diese Option hat keinen Einfluss darauf, ob die Groß-/Kleinschreibung der Dateinamen in den Ordnern beachtet wird.) Wenn Sie den Befehl <code>-csf</code> verwenden, werden die StarTeam-Ordner "doc" und "Doc" beispielsweise als unterschiedliche Ordner erkannt. Ohne diese Option werden beide Ordner als "doc"-Ordner erkannt.</p> <p>Standardmäßig erfolgt bei Namen von StarTeam-Ordnern keine Unterscheidung aufgrund divergierender Groß-/Kleinschreibung.</p> <p>Mit oder ohne Verwendung des Befehls <code>-csf</code> tritt bei nicht eindeutigen Ordernamen ein Fehler auf. Bei Verwendung von <code>-csf</code> sind die Namen von zwei Ordnern nicht eindeutig, wenn ein Ordner mit "Doc" und ein Ordner mit "doc" bezeichnet ist. Wenn Sie <code>-csf</code> nicht verwenden, sind Ordernamen bei gleicher Rechtschreibung nicht eindeutig.</p>
<code>-dryrun</code>	<p>Checkt Dateien nicht aus, zeigt aber eine Liste der Dateien an, die ausgecheckt würden, wenn <code>-dryrun</code> nicht angegeben wäre. Die Pfade verweisen auf die Arbeitsordner, in die die Dateien kopiert würden. Wenn Sie die Option in Verbindung mit <code>-vb</code> verwenden, erhalten Sie eine vollständige Übersicht.</p>
<code>-encrypt RC4 RC2_ECB RC2_CBC RC2_CFB</code>	<p>Verschlüsselt alle zwischen der Workstation und dem Server ausgetauschten Daten und entschlüsselt eingehende Daten. Ohne die Option erfolgt keine Komprimierung. Die Verschlüsselung schützt Dateien und andere Projektinformationen vor dem Zugriff durch nicht autorisierte Dritte über ungeschützte Netzwerkverbindungen.</p> <p>Die vollständige Syntax lautet: <code>-encrypt Verschlüsselungstyp</code></p> <p>BCO unterstützt folgende Verschlüsselungstypen:</p> <p>RC4: RSA RC4 Stromverschlüsselung (schnell)</p> <p>RC2_ECB: RSA RC2 Blockverschlüsselung (Electronic Codebook)</p> <p>RC2_CBC: RSA RC2 Blockverschlüsselung (Cipher Block Chaining)</p> <p>RC2_CFB: (Nur auf Windows-Plattformen) RSA RC2 Blockverschlüsselung (Cipher Feedback)</p> <p>Die genannten Verschlüsselungstypen sind in der Reihenfolge vom schnellsten zum langsamsten hin geordnet. Je langsamer der Verschlüsselungstyp um so sicherer die Verschlüsselung.</p>
<code>-eol { on off cr lf crlf }</code>	<p>Führt die Zeilenendekontvertierung von Textdateien durch.</p> <p>Wenn die Option <code>-on</code> mit angegeben wird, werden Textdateien aus dem Repository des Servers an den Arbeitsordner der Workstation übertragen, wobei die Zeilenendekonvention der Plattform verwendet wird, die den Befehl ausführt. wie von der JVM festgelegt.</p> <p>Wenn Sie diese Option nicht angeben, führt BCO standardmäßig keine Zeilenendekontvertierung durch. Die Verwendung des Standards hat dieselbe Auswirkung, als würde <code>-eol</code> nicht angegeben.</p> <p>Wenn Sie das Zeilenendezeichen (<code>cr</code>, <code>lf</code> oder <code>crlf</code>) angeben, überträgt BCO die Textdateien aus dem Repository des Servers an den</p>

Arbeitsordner auf der Workstation, wobei die angegebene Zeilenendekonvention verwendet wird.

Auf Windows-Plattformen wird die EOL-Markierung "Carriage Return / Line Feed" (`\r\n`) verwendet; auf UNIX-Plattformen wird "Line Feed" (`\n`) verwendet.

Sie aktivieren diese Option beispielsweise, wenn Sie eine Datei aus dem Repository mit einer Arbeitsdatei auf einem UNIX-System vergleichen möchten (wenn das Repository Textdateien mit `\r\n` speichert).

```
-exclude <Muster> | @<Musterdatei>
```

Schließt Dateien aus, deren Name oder übergeordneter Ordner dem angegebenen Muster entspricht. Sie können entweder den exakten Datei- oder Ordnernamen als Muster eingeben oder Platzhalterzeichen verwenden (z. B. `*.class`). Bei der Angabe eines Ordnernamens muss vor dem Muster ein Schrägstrich (z. B. `./bin`) stehen. Jeder `-exclude`-Befehl kann ein einzelnes Muster enthalten, das wiederholt werden kann. Alternativ können Sie in der vorgegebenen `<Musterdatei>` (der ein `@` vorangestellt ist) ein oder mehrere Muster in separaten Zeilen angeben.

Dateien...

Gibt die im Befehl zu verwendenden Dateien über den Namen oder über Dateinamensmuster an (z. B. `*.c`). Alle hier genannten Optionen entsprechen den semantischen Konventionen unter UNIX, da diese spezifischer als die Windows-Konventionen sind. Dies bedeutet, dass `**` und nicht `*.*` "Alle Dateien" bedeutet. Das Muster `*.*` bedeutet "Alle Dateien mit Dateinamenserweiterungen". Wenn Sie beispielsweise `star*.*` eingeben, werden `starteam.doc` und `starteam.cpp`, aber nicht `starteam` gefunden. Wenn Sie alle Vorkommen mit "starteam" finden möchten, verwenden Sie stattdessen `star*`.

Wenn diese Option nicht verwendet wird, lautet die Vorgabe `**`.

Wenn Sie `*` statt `**` als Platzhalter für alle Dateien verwenden, erweitert die UNIX-Shell diesen Befehl auf mehrere Elemente und übergibt diese als eine Optionengruppe an den `bc`-Befehl. Dies kann zu Problemen führen, z. B. beim Auschecken fehlender Dateien. Verwenden Sie daher `**`, um Komplikationen zu vermeiden.

Wenn Sie eine Gruppe von Dateimustern verwenden, sollte jedes Muster separat zwischen Anführungszeichen gesetzt werden. So können Sie beispielsweise `".bat" *.c`, jedoch nicht `".bat *.c` verwenden.

Hinweis: Es wird empfohlen, dass Sie diese Option aus den folgenden Gründen, unabhängig von der Plattform, in Anführungszeichen angeben. Unter Windows werden Datei- und Ordnernamen, die Leerzeichen enthalten, nicht korrekt erkannt. Unter UNIX wird die Option von der Shell erweitert und anschließend wird die aufgrund der Erweiterung erstellte Elementliste an den Client übergeben. Diese Vorgehensweise führt jedoch häufig zu unerwarteten Ergebnissen. Um entsprechende Fehler zu vermeiden, sollten Sie diese Option immer zwischen Anführungszeichen setzen. Nur in den Fällen, in denen es unbedingt erforderlich ist, dass die Option von der UNIX-Shell erweitert wird, empfiehlt es sich, die Anführungszeichen wegzulassen. Wenn Sie die Anführungszeichen weglassen, kann es auf beiden Plattformen bei der Ausführung von Batch-Skripts zu Fehlern kommen.

In der Dateispezifikation können mehrere Sonderzeichen verwendet werden:

* Entspricht einem beliebigen String, einschließlich einem leeren String. Beispiel: `**` entspricht einem beliebigen Dateinamen mit oder ohne

Erweiterung. "xyz*" entspricht "xyz", "xyz.cpp" und "xyzutyfj".

? Entspricht einem beliebigen einzelnen Zeichen. Beispiel: "a?c" entspricht "abc", aber NICHT "ac".

[...] Entspricht allen Zeichen, die von der linken und der rechten Klammer eingeschlossen werden.

Mehrere Zeichen, die durch Bindestriche (-) getrennt werden, geben die Zeichen an, für die eine Übereinstimmung gefunden werden muss.

Wenn das erste Zeichen nach einer öffnenden, eckigen Klammer ([) ein Ausrufezeichen (!) oder ein Caret-Zeichen (^) ist, dürfen die nachfolgenden Zeichen nicht übereinstimmen. Alle Zeichen außerhalb der Klammer müssen übereinstimmen. Beispiel: "x[a-d]y" entspricht "xby", aber nicht "xey". "x[!a-d]y" entspricht "xey", aber nicht "xby".

Ein Bindestrich (-) oder eine schließende eckige Klammer (]) werden gefunden, wenn er bzw. sie als erstes oder als letztes Zeichen der in Klammern gesetzten Zeichen steht.

Wenn Sie ein Sternchen (*), Fragezeichen (?) oder eine öffnende, eckige Klammer ([) in einem Muster verwenden möchten, muss vor diesen Zeichen ein Escape-Zeichen, d. h. ein umgekehrter Schrägstrich (\), stehen.

-filter

Gibt einen String mit einem oder mehreren Zeichen an, von denen jedes einen Dateistatus darstellt. In diesem String dürfen keine Leerzeichen oder andere Whitespace-Zeichen verwendet werden. Es werden nur Dateien mit dem angegebenen Status ausgecheckt. Dateien des Typs *Nicht in Ansicht* können nicht ausgecheckt werden.

Zur Darstellung der einzelnen Statuswerte werden die folgenden Buchstaben verwendet: **C** für "Current" (Aktuell), **M** für "Modified" (Geändert), **G** für "Merge" (Zusammenführen), **O** für "Out of Date" (Veraltet), **I** für "Missing" (Fehlt) und **U** für "Unknown" (Unbekannt).

Wenn Sie **C**, **M**, **G**, **O** oder **U** verwenden, müssen Sie auch **-o** angeben, um das Auschecken zu erzwingen. Wird die Option **-o** nicht verwendet, überspringt BCO Dateien mit dem Status **C**, **M**, **G**, **O** und **U**. Es wird jedoch eine Warnung in die Protokolldatei eingetragen.

-fp "Ordnerpfad"

Überschreibt den Arbeitsordner oder das Arbeitsverzeichnis des angegebenen Ordners. Dies entspricht dem Festlegen eines alternativen Arbeitspfads für den Ordner.

Mithilfe dieser Option können Sie einen anderen Arbeitsordner verwenden als den, der durch den Ordner angegeben ist, und zudem die plattformübergreifende Kompatibilität sicherstellen. Bei der Angabe von Laufwerks- und Verzeichnispfadnamen sind UNIX- und Windows-Systeme beispielsweise nicht kompatibel. Der Pfad "**D:** \MEINPRODUKT\ENTWICKLUNG\QUELLE" wird unter Windows, jedoch nicht unter UNIX korrekt interpretiert. Mithilfe dieser Option können Sie den Pfad des Arbeitsverzeichnisses angeben, wenn der im Projekt angegebene Pfad von Ihrer Plattform nicht erkannt wird.

Umgekehrte Schrägstriche (\) werden als Escape-Zeichen interpretiert, wenn sie Anführungszeichen vorangestellt werden. Aus diesem Grunde tritt im folgenden Beispiel ein Fehler auf:

```
bco -p "xxx" -fp "C:\ " "*"
```

wird interpretiert als:

```
bco -p "xxx" -fp "C: "*" .
```

Um einen solchen Fehler zu vermeiden, geben Sie "C:\\" wie folgt an:

```
bco -p "xxx" -fp "C:\\ \" "*" .
```

Oder geben Sie Folgendes ein, wenn der `-fp`-Pfad nicht mit dem Stammordner endet wie beispielsweise bei "C:\orion\`":`

```
bco -p "xxx" -fp "C:\orion" "*" .
```

Beachten Sie außerdem, dass UNC-Netzwerkpfade ein zusätzliches Escape-Zeichen erfordern, z. B.:

```
bco -p "xxx" -fp "\\Server\Pfad".
```

Die vollständige Syntax lautet: `-fp "Ordnerpfad" .`

Ordner ist der Windows-Begriff, der auch in der Benutzeroberfläche verwendet wird. Verzeichnis ist der Begriff, der für die UNIX-Plattform verwendet wird.

`-fs`

Verhindert, dass der Dateistatus nach dem Auschecken beibehalten wird. Nachfolgende Statuswerte für diese Dateien sind falsch und unklar. Verwenden Sie diese Option, wenn der Status einer Datei irrelevant ist. Wenn Sie die Arbeitsordner beispielsweise routinemäßig vor dem Auschecken der Dateien für einen Build löschen, sind keine Dateien vorhanden und ihr Status spielt demnach keine Rolle.

Bedenken Sie, dass dann der Status der Dateien niemals bekannt ist, selbst dann nicht, wenn Sie zu einem späteren Zeitpunkt den Befehl zum Aktualisieren des Status verwenden. Um die aktuellen Dateien mit ihren richtigen Statuswerten zu erhalten, können Sie das Auschecken ohne die Option `-fs` erzwingen.

`-h`

Zeigt Informationen zu den Befehlszeilenoptionen an.

`-help`

`-is`

Wendet den Befehl rekursiv auf alle untergeordneten Ordner an. Ohne diese Option wird der Befehl nur für den angegebenen Ordner übernommen.

`-netmon`

Gibt SDK NetMonitor-Informationen im Konsolenfenster aus. NetMonitor zeigt Statistiken für Serverbefehle an. Weitere Beispiele zu BCO finden Sie im nächsten Abschnitt dieses Themas.

`-o`

Erzwingt das Auschecken der Dateien, deren Status dies in der Regel nicht zulassen würde. Diese Statuswerte sind *Geändert*, *Zusammenführen* und *Unbekannt*.

`-p`

Zeigt die zu verwendende Ansicht oder den zu verwendenden Ordner an und stellt den Benutzernamen und das Passwort für den Serverzugriff bereit.

Die vollständige Syntax lautet:

```
-p "Benutzername:Passwort@Host-Name:Port-Nummer/  
Projektname/[Ansichtsname/][Ordnerhierarchie/]"
```

Beispiel: `-p "JMarsh:Passwort@orion:49201/StarDraw/StarDraw/SourceCode/"`.

Wenn der Benutzername weggelassen wird, wird der aktuelle Benutzername verwendet. Der Benutzername im Beispiel lautet "JMarsch".

Wenn das Passwort weggelassen wird, wird der Benutzer aufgefordert, das Passwort einzugeben. Wenn der Benutzer ein Passwort eingibt, werden die Zeichen nicht auf dem Bildschirm angezeigt. Das Passwort im Beispiel lautet `"Passwort"`.

Wenn der Host-Name weggelassen wird, wird als Vorgabe `localhost` verwendet. Der Host-Name im Beispiel lautet `"orion"`.

Die Port-Nummer ist erforderlich. In diesem Beispiel wird die Standard-Port-Nummer `49201` verwendet.

Der Projektname muss immer angegeben werden. Der Projektname im Beispiel lautet `StarDraw`.

Kennzeichnen Sie die Ansicht mithilfe der Ansichtshierarchie. Verwenden Sie einen Doppelpunkt (`:`) als Begrenzungszeichen zwischen Ansichtsnamen. Die Ansichtshierarchie sollte stets die Stammansicht einschließen. `"StarDraw:Release 4:Service Packs"` gibt beispielsweise an, dass die Ansicht `Service Packs` verwendet werden soll, die der Ansicht `Release 4` untergeordnet ist, die wiederum eine untergeordnete Ansicht der zweiten Ebene der Stammansicht `StarDraw` ist. Ohne Angabe eines Ansichtsnamens wird die Stammansicht (Standardansicht) verwendet. Wenn die Ansicht die einzige Ansicht dieses Projekts mit diesem Namen ist, reicht es aus, nur den Ansichtsnamen zu verwenden. (Diese Vorgehensweise empfiehlt sich jedoch nicht, da im Verlauf des Projekts eine weitere Ansicht mit demselben Namen erstellt werden könnte, was zu Verwechslungen führen kann.) Der Ansichtsnamen im Beispiel lautet `StarDraw`. Da dies die Stammansicht des `StarDraw`-Projekts ist, können er auch weggelassen werden.

Kennzeichnen Sie den Ordner mithilfe der Ordnerhierarchie. Verwenden Sie den Schrägstrich (`/`) als Begrenzungszeichen zwischen Ordnernamen. Der Stammordner ist in der Ordnerhierarchie niemals enthalten. Lassen Sie die Ordnerhierarchie weg, wenn die Datei sich im Stammordner der Ansicht befindet. Wenn der Name des Stammordners beispielsweise `StarDraw` lautet und Ihre Dateien sich in der Hierarchie `"StarDraw/SourceCode/Client"` befinden, verwenden Sie nur `"SourceCode/Client"`.

Wenn einer der mit dieser Option verwendeten Parameter (z. B. Benutzernamen, Ansichtsnamen, Passwörter, Projekt- oder Ordnernamen) Zeichen enthält, die als Begrenzungszeichen eingesetzt werden, verwenden Sie für das Zeichen das Prozentzeichen (`%`) gefolgt vom Hex-Wert des entsprechenden Zeichens. Wenn beispielsweise `"@"` als Zeichen in einem Passwort verwendet wird, müssen sie es durch `"%40"` ersetzen .

Ersetzen Sie `":"` durch `"%3a"`

Ersetzen Sie `"/"` durch `"%2f"`

Ersetzen Sie `"@"` durch `"%40"`

Ersetzen Sie `"%"` durch `"%25"`

Unter UNIX und anderen Betriebssystemen muss einigen Sonderzeichen ein umgekehrter Schrägstrich (`"\"`) oder ein anderes Escape-Zeichen vorangestellt werden. Wenn Sie die Option `-p` verwenden, können Sie diese Zeichen durch den entsprechenden Hexcode ersetzen. Beispielsweise kann unter UNIX `"%3c"` anstelle von `"\<"` verwendet werden.

Ersetzen Sie Leerzeichen durch `"%20"`

	Ersetzen Sie "<" durch "%3c"
	Ersetzen Sie ">" durch "%3e"
<code>-pwdfile</code>	<p>Gibt den Pfad zu der Datei an, in der das Passwort des Benutzers gespeichert ist. Diese Option hat Vorrang vor dem mit der <code>-p</code>-Option verwendeten Passwort. Sie verhindert, dass andere Benutzer das Passwort des Benutzers in der Befehlszeile sehen können. Es muss im UTF-8-Format gespeichert werden.</p> <p>Die vollständige Syntax lautet:</p> <pre><code>-pwdfile "Dateipfad" .</code></pre>
<code>-ro</code>	<p>Definiert nach dieser Operation die Arbeitsdatei als schreibgeschützt. Ohne diese Option bliebe der Status der Datei derselbe wie vor der Operation. In der Regel verwenden Sie <code>-ro</code>, um zu verhindern, dass Sie selbst eine Datei bearbeiten, die Sie nicht gesperrt haben. Ohne die Option <code>-ro</code> können die Dateien gelesen und geschrieben werden.</p>
<code>-rp</code>	<p>Gibt den Arbeitsordner oder das Arbeitsverzeichnis für den Stammordner der Ansicht an oder überschreibt diese. Dies entspricht dem Festlegen eines alternativen Arbeitspfads für die Ansicht.</p> <p>Weitere Informationen hierzu finden Sie unter der Beschreibung der <code>-fp</code>-Option in dieser Tabelle.</p> <p>Die vollständige Syntax lautet:</p> <pre><code>-rp "Ordnerpfad".</code></pre>
<code>-t</code>	Zeigt das Auscheck-Volumen und Zeitstatistiken an.
<code>-ts</code>	Stellt den Zeitstempel für jede einzelne Arbeitsdatei auf die Auscheckzeit ein. Ohne diese Option erhält die Datei denselben Zeitstempel wie die eingecheckte Revision der Datei.
<code>-useCA</code>	<p>Versucht, Dateien unter Verwendung des StarTeamMPX-Cache-Agenten auszuchecken.</p> <p>Die vollständige Syntax lautet:</p> <pre><code>-useCA Host:Port-Nummer autolocate.</code></pre> <p>Die Syntax <code>Host:Portnummer</code> gibt den Host-Namen (oder die IP-Adresse) und die Port-Nummer des zu verwendenden Cache-Agenten an.</p> <p>Andererseits kann <code>autolocate</code> angegeben werden, um automatisch den nächsten Netzwerk-Cache-Agenten zu suchen. Voraussetzung für die Verwendung von <code>autolocate</code> ist, dass der StarTeam-Server MPX-fähig ist.</p>
<code>-vb</code>	Die Ausgabe ist ausführlich. Es wird jede Datei angezeigt, während sie ausgecheckt wird. Der Ordnerpfad ist der StarTeam-Ordnerpfad und nicht der Pfad des Arbeitsordners.

BCO-Anwendungsbeispiele

Im folgenden Beispiel wird BCO dazu verwendet, das Auschecken aller Dateien aus *Source Code*, einem untergeordneten Ordner des Stammordners *StarDraw*, zu erzwingen (in der *StarDraw*-Ansicht des *StarDraw*-Projekts).

```
bco -p "JMarsch:Passwort@Orion:1024/StarDraw/StarDraw/Source Code" -is -o "*" 
```

Das nächste Beispiel zeigt einen BCO-Befehl mit der `-netmon`-Option und die vom Netzmonitor angezeigte Ausgabe.

```
bco -p "Administrator:Administrator@10.50.6.91:49201/StarDraw/WebSite" -fp D:\ Test  
-netmon -o "*.htm"
```

Beispiel für Ausgabe:

```
StarTeam BulkCheckOut Utility version 9.0.xxx  
Copyright (c) 2006 Borland Software Corporation. Alle Rechte vorbehalten.  
Start: (rev 100) SRVR_CMD_GET_PROJECT_LIST Time: 62 millis; Sent: 42 bytes;  
Got: 1834 bytes  
Start: (rev 100) SRVR_CMD_GET_PROJECT_VIEWS Time: 47 millis; Sent: 46 bytes;  
Got: 186 bytes  
Start: (rev 100) SRVR_CMD_GET_PROJECT_VIEWS Time: 15 millis; Sent: 46 bytes;  
Got: 186 bytes  
Start: (rev 100) SRVR_CMD_PROJECT_OPEN Time: 188 millis; Sent: 70 bytes;  
Got: 120 bytes  
Start: (rev 100) PROJ_CMD_GET_VIEW_PROPERTIES Time: 31 millis; Sent: 42 bytes;  
Got: 2556 bytes  
Start: (rev 100) PROJ_CMD_GET_FOLDERS Time: 63 millis; Sent: 42 bytes;  
Got: 1112 bytes  
Start: (rev 100) PROJ_CMD_GET_FOLDER_ITEMS Time: 16 millis; Sent: 50 bytes;  
Got: 40 bytes  
Start: (rev 100) PROJ_CMD_REFRESH_ITEMS Time: 3562 millis; Sent: 122 bytes;  
Got: 414 bytes  
Start: (rev 100) SRVR_CMD_GET_PROJECT_VIEWS Time: 16 millis; Sent: 46 bytes;  
Got: 186 bytes  
Start: (rev 100) PROJ_CMD_GET_PROJECT_PROPERTIES Time: 31 millis; Sent: 42 bytes;  
Got: 4797 bytes  
Start: (rev 100) FILE_CMD_CHECKOUT Time: 47 millis; Sent: 78 bytes;  
Got: 108 bytes  
Start: (rev 100) FILE_CMD_CHECKOUT Time: 31 millis; Sent: 78 bytes;  
Got: 1767 bytes  
Start: (rev 100) FILE_CMD_CHECKOUT Time: 31 millis; Sent: 78 bytes;  
Got: 1140 bytes  
Start: (rev 100) SRVR_CMD_PROJECT_CLOSE Time: 15 millis; Sent: 62 bytes;  
Got: 16 bytes  
Start: (rev 100) SRVR_CMD_RELEASE_CLIENT Time: 31 millis; Sent: 42 bytes;
```


Got: 16 bytes

Befehlszeilenoptionen für das Checkout Trace-Dienstprogramm

In diesem Thema werden die Befehlszeilenoptionen für das Checkout Trace-Dienstprogramm beschrieben.

Im Allgemeinen können Sie das Dienstprogramm von der Befehlszeile aus wie folgt aufrufen:

`CheckoutTraceDump.exe -go` . In der folgenden Tabelle werden gültige Optionen für Checkout Trace beschrieben.

Option	Beschreibung
<code>-go</code>	Geben Sie diesen Parameter an, wenn das Programm mit den Standardeinstellungen ausgeführt werden soll.
<code>-path:<Pfad></code>	Der Ordner mit den binären Checkout Trace-Dateien (<code>.cotrc</code>). Standardmäßig ist dies der aktuelle Ordner.
<code>-outpath:<Pfad></code>	Der Ordner, in dem die Ausgabedateien (<code>.csv</code>) gespeichert werden. Standardmäßig wird derselbe Ordner verwendet, in dem die binären Trace-Dateien (<code>.cotrc</code>) gespeichert sind.
<code>-file:<Dateispezifikation></code>	Die als Input verwendeten binären Checkout Trace-Dateien. Es werden die Standard-Platzhalterzeichen (*,?) des Dateisystems unterstützt. Die Standardeinstellung ist <code>"*.cotrc"</code> (alle Checkout Trace-Dateien im Ordner). Sie können mit dem Parameter nur einen Pfad verwenden und sie können diesen Parameter im Befehl nur einmal angeben.
<code>-ext:<Erweiterung></code>	Die für Checkout Trace-Dateien verwendete Dateierweiterung. Die Erweiterung wird an den Dateinamen des binären Checkout Trace (<code>.cotrc</code>) angehängt, wodurch der Name der Ausgabedatei erzeugt wird. Die standardmäßige Erweiterung lautet <code>.csv</code> .
<code>-start:<Startzeit></code>	Frühester Zeitpunkt von Interesse. Bei der Ausgabe werden nur Auscheckvorgänge berücksichtigt, die nach diesem Zeitpunkt stattfanden. Standardmäßig filtert das Dienstprogramm nicht nach Zeit.
<code>-end:<Endezeit></code>	Der aktuellste Zeitpunkt von Interesse. Bei der Ausgabe werden nur Auscheckvorgänge berücksichtigt, die vor diesem Zeitpunkt stattfanden. Standardmäßig filtert das Dienstprogramm nicht nach Zeit.
<code>-project:<Projektname></code>	Name des Projekts, für das die Informationen zum Auschecken gefiltert werden. Bei der Ausgabe werden nur Auscheckvorgänge berücksichtigt, die dieses Projekt betreffen. Standardmäßig filtert das Dienstprogramm nicht nach Projekt. Alle Projekte werden bei der Ausgabe berücksichtigt. Wenn sowohl <code>-project</code> als auch <code>-projectid</code> angegeben werden, hat <code>-projectid</code> Vorrang.
<code>-projectid:<Projekt-ID></code>	ID des Projekts, für das die Informationen zum Auschecken gefiltert werden. Bei der Ausgabe werden nur Auscheckvorgänge berücksichtigt, die dieses Projekt betreffen. Standardmäßig filtert das Dienstprogramm nicht nach Projekt-ID. Alle Projekte werden bei der Ausgabe berücksichtigt. Diese Option hat Vorrang vor <code>-project</code> , wenn beide Optionen angegeben werden.
<code>-separator:<Trennzeichen></code>	Zum Trennen von Werten in der Ausgabedatei verwendeter String. Standardmäßig verwendet das Dienstprogramm <code>" , "</code> .

`-overwrite`

Mit diesem Parameter können Sie vorhandene Checkout Trace-Dateien überschreiben. Falls nicht angegeben, werden Checkout Trace-Binärdateien übersprungen, wenn bereits eine Trace-Dump-Datei mit dem Zielnamen vorhanden ist.

Vault Verify - Befehlszeilenoptionen

In diesem Thema werden die Befehlszeilenoptionen für das Vault Verify-Dienstprogramm beschrieben.

Im Allgemeinen können Sie Vault Verify von der Befehlszeile aus wie folgt aufrufen: `VaultVerify [Optionen] "Konfiguration"`.

Die Datenspeicher-Archivdateien für die angegebene StarTeam-"Serverkonfiguration" werden anhand den verwendeten `-check`-Optionen auf Integrität geprüft. Wenn Sie die `-repair`-Option angeben, versucht Vault Verify, gefundene Probleme zu korrigieren. Vault Verify öffnet die Datenbank für die Serverkonfiguration, ändert sie aber nicht. In der folgenden Tabelle werden gültige Optionen für Vault Verify beschrieben.

Option	Beschreibung
<code>check {missing corrupt stray all}</code>	<p>Mit den check-Optionen legen Sie fest, welche Integritätsprüfungen durchgeführt werden:</p> <p><code>missing</code>: Prüft auf fehlende Dateien, indem die Datenbank mit den tatsächlich vorhandenen Archivdateien abgeglichen wird.</p> <p><code>corrupt</code>: Prüft die Integrität der vorhandenen Archivdateien (MD5, Name, Ordner und .gz-Dateiformat).</p> <p><code>stray</code>: Prüft anhand der Datenbank auf verwaiste Dateien. Diese Option kann nicht verwendet werden, wenn die Serverkonfiguration in Gebrauch ist.</p> <p><code>all</code>: Führt alle Integritätsprüfungen durch.</p> <p>Es können mehrere <code>-check</code>-Optionen angegeben werden. Beachten Sie auch die <code>-repair</code>-Option.</p>
<code>-cf <Ordnerpfad></code>	<p>Pfadname für den <i>Ordner mit beschädigten Dateien</i>, in den bei Verwendung von <code>-repair</code> Problemdateien verschoben werden, die von der <code>corrupt</code>-Option gefunden wurden. Der Standardordner mit beschädigten Dateien ist <code>C:\Temp\VVCorruptFiles</code>.</p>
<code>-dbhost <Host></code>	<p>Gibt den Host-Namen der Datenbank für die angegebene <code><Serverkonfiguration></code> an. Dies ist unter Windows nur sinnvoll, wenn auch <code>-dbinstance</code> angegeben wird. Verwenden Sie unter Windows und Linux diese Option nur dann, wenn der Datenbankserver auf einem anderen als diesem Host ausgeführt wird.</p>
<code>-dbname <Name></code>	<p>Gibt den Datenbanknamen für die angegebene <code><Konfiguration></code> an. Unter Windows ist dieser Parameter nur dann sinnvoll, wenn auch <code>-dbinstance</code> angegeben wird, und wird nur dann benötigt, wenn sich der Datenbankname von dem ODBC-DSN unterscheidet. Verwenden Sie diese Option nur dann unter Linux, wenn <code>-dbinstance</code> nicht verwendet wird und sich der Oracle-Dienstname von dem TNS-Namen unterscheidet.</p>
<code>-dbinstance <Name></code>	<p>Dieses Option ist nur unter Windows sinnvoll. Wenn sie verwendet wird, sorgt sie dafür, dass VaultVerify die Datenbank direkt anstatt über den in der Konfigurationsdatei angegebenen ODBC-DSN öffnet. Bei SQL Server muss der <code><Name></code> eine Instanzname sein (z. B., 'SSE2005_ST'). Beachten Sie, dass der Standard-Instanzname für SQL Server 'MSSQLSERVER' und für SQL Server Express 'SQLEXPRESS' lautet. Bei Oracle wird der Dienstname verwendet (z. B., 'ORCL').</p> <p><code>-dbinstance</code> muss zusammen mit <code>-dbhost</code> angegebenen werden, wenn der Datenbankserver auf einem anderen Host ausgeführt wird. Bei SQL Server sollte <code>-dbname</code> auch verwendet werden, falls sich der</p>

	Datenbankname von dem ODBC-DSN unterscheidet. Oracle ignoriert <code>-dbname</code> , wenn <code>-dbinstance</code> angegeben wird.
<code>-dbpassword <Passwort></code>	Gibt das Passwort für die Datenbank anmeldung an. Wenn nichts angegeben wird, wird ein leeres Passwort verwendet. (Das in the Konfiguration gespeicherte Passwort ist verschlüsselt und kann von Vault Verify nicht verwendet werden.) Bei Serverkonfigurationen für Oracle-Server muss diese Option angegeben werden, da das Oracle-Passwort nicht leer sein darf.
<code>-dbport <Port></code>	Gibt den TCP/IP-Port an, der für die Verbindung zum Datenbank-Server verwendet werden soll. Dieser Parameter wird nur auf Nicht-Windows-Plattformen verwendet, wenn ein anderer Port als der Standard-Datenbank-Port des Herstellers (z. B. 1521 für Oracle) verwendet wird.
<code>-dbuser <Benutzer></code>	Gibt die Anmelde-ID für die Verbindung zur Datenbank an. Wird dieser Parameter angegeben, hat er Vorrang vor dem Benutzer, der in der StarTeam-<Konfiguration> angegeben ist. Der einzige gültige Benutzer, der mit dieser Option angegeben werden kann, ist der Benutzer, der Eigentümer der StarTeam-Tabellen ist.
<code>-help (oder -h oder -?)</code>	Zeigt die Nutzungsinformationen an.
<code>-path <Ordnerpfad></code>	Gibt den Ordnerpfad der Datei <code>starteam-server-configs.xml</code> an. Diese Datei muss vorhanden sein und die angegebene <Serverkonfiguration> enthalten. Standardmäßig wird diese Datei im übergeordneten Ordner des aktuellen Arbeitsverzeichnisses geöffnet, wenn sie nicht im aktuellen Arbeitsverzeichnis gefunden wird.
<code>-nosharereport</code>	Unterdrückt die Aufnahme von Freigabeinformationen in die Berichte. In der Regel werden alle Freigabepfade einer jeden beschädigten Datei in die Berichte aufgenommen. Mit dieser Option werden die Informationen über die Freigabepfade unterdrückt. Dies dient der Beschleunigung der Ausführung von Anwendungen sowie der deutlichen Reduzierung der Berichtsgröße.
<code>-repair</code>	Gibt an, dass versucht werden soll, die Probleme mit den Archivdateien zu beheben. 'Beschädigte' Archive werden in den Ordner für beschädigte Dateien verschoben (siehe die <code>-cf</code> -Option). Wenn sie gültigen Dateirevisionen entsprechen, werden sie anschließend als "fehlend" behandelt. Die Wiederherstellung fehlender Archive wird von anderen Datenspeicherdateien versucht bzw. von einem Cache-Agenten, wenn die <code>-useca</code> -Option angegeben wird. Verwaiste Archive werden in den Ordner für verwaiste Dateien verschoben (siehe die <code>-sf</code> -Option). Hinweis: <code>-repair</code> wird ignoriert, wenn die StarTeam-<Konfiguration> verwendet wird.
<code>-sf <Ordnerpfad></code>	Pfadname für den 'Ordner mit verwaisten Dateien', in den bei Verwendung von <code>-repair</code> verwaiste Dateien verschoben werden, die von der "stray"-Prüfung gefunden wurden. Der Standardordner mit 'verwaisten Dateien' ist <code>C:\Temp\VVStrayFiles</code> .
<code>-t</code>	Zeigt die Zeit an, die für die Verifizierung benötigt wurde.
<code>-useca <Host>:<Port></code>	Wenn <code>-repair</code> angegeben wird, wird versucht, fehlende Dateien über den angegebenen MPX-Cache-Agenten wiederherzustellen. Mit <Host> und <Port> muss ein Remote-Cache-Agent festgelegt werden, da er einen unabhängigen Cache-Speicher verwaltet.
<code>-verbose</code>	Zeigt bei fortschreitender Verifizierung zusätzliche Statusinformationen an.

"Konfiguration"

Gibt den Konfigurationsnamen an. Bei dem an Vault Verify übergebenen Namen der Konfiguration muss die Groß-/ Kleinschreibung beachtet werden. Wenn der Name Leerzeichen enthält, müssen Sie den an Vault Verify übergebenen Konfigurationsnamen in Anführungszeichen setzen.

Weitere Client-Befehlszeilenoperationen

Dieser Abschnitt enthält Referenzthemen, die sich mit Client-Befehlszeilenoperationen befassen.

In diesem Abschnitt

[Allgemeine Optionen](#)

Beschreibt die Optionen, die in allen oder fast allen Befehlen zum Einsatz kommen.

[Sonderzeichen](#)

Beschreibt die Sonderzeichen, die für die Suche nach Dateien verwendet werden können.

[Exit-Codes](#)

Beschreibt die Exit-Codes, die von den stcmd-Befehlen zurückgegeben werden.

[Dateien hinzufügen: stcmd add](#)

Beschreibt den Befehl zum Hinzufügen von Dateien zu einem Projekt.

[Ordner hinzufügen: stcmd add-folder](#)

Beschreibt den Befehl zum Hinzufügen von StarTeam-Ordnern zu einer Ansicht.

[Projekte hinzufügen: stcmd add-project](#)

Beschreibt den Befehl zum Hinzufügen eines Projekts zu einer StarTeam-Server-Konfiguration.

[Ansichten hinzufügen: stcmd add-view](#)

Beschreibt den Befehl zum Hinzufügen einer Ansicht zu einer StarTeam-Server-Konfiguration.

[Labels übernehmen: stcmd apply-label](#)

Beschreibt den Befehl zum Übernehmen eines Ansichts- oder Revisions-Labels für Dateirevisionen.

[Dateien einchecken: stcmd ci](#)

Beschreibt den Befehl zum Einchecken von Dateien aus einem Arbeitsordner in ein Repository (oder einen Dateispeicher) von StarTeam.

[Dateien auschecken: stcmd co](#)

Beschreibt den Befehl zum Auschecken von Dateien aus einem Repository (oder einem Dateispeicher) von StarTeam in Ihren Arbeitsordner.

[Dateirevisionen vergleichen: stcmd diff](#)

Beschreibt den Befehl zum Anzeigen von Unterschieden zwischen zwei Dateirevisionen.

[Dateibeschreibungen ändern: stcmd dsc](#)

Beschreibt den Befehl zum Ändern einer Dateibeschreibung.

[Labels erstellen: stcmd label](#)

Beschreibt den Befehl zum Erstellen von Ansichts- oder Revisions-Labels.

[Arbeitsordner erstellen: stcmd local-mkdir](#)

Beschreibt den Befehl zum Erstellen des Arbeitsordners oder des Arbeitsverzeichnisses eines angegebenen StarTeam-Ordners auf Ihrer Workstation.

[Lokale Dateien löschen: stcmd delete-local](#)

Beschreibt den Befehl zum Löschen von Dateien aus einem Arbeitsordner.

[Dateihistorien anzeigen: stcmd hist](#)

Beschreibt den Befehl zum Anzeigen der Revisionshistorie von Dateien.

[Dateien auflisten: stcmd list](#)

Beschreibt den Befehl zum Auflisten aller Dateien eines Ordners, der mit der Option "-p" angegeben wird.

[Dateien sperren und entsperren: stcmd lck](#)

Beschreibt den Befehl zum Sperren oder Entsperrern von Dateien.

[Server sperren und entsperren: stcmd server-mode](#)

Beschreibt den Befehl zum Sperren und Entsperren einer Serverkonfiguration.

[Dateien entfernen: stcmd remove](#)

Beschreibt den Befehl zum Entfernen von Dateien aus der Versionskontrolle.

[Persönliche Optionen anpassen: stcmd set-personal-options](#)

Beschreibt den Befehl zum Anpassen und Auflisten der persönlichen Optionen.

[Dateistatus aktualisieren: stcmd update-status](#)

Beschreibt den Befehl zum Aktualisieren des Status einer Datei.

Allgemeine Optionen

Einige der im Folgenden genannten Optionen kommen in allen oder fast allen dieser Befehle zum Einsatz. Sie haben für jeden Befehl dieselbe Bedeutung, daher werden sie nur an dieser Stelle erläutert. Eine wiederholte Erklärung an späterer Stelle erfolgt nicht. Optionen, die nicht in allen Befehlen verwendet werden, oder deren Bedeutung von Befehl zu Befehl unterschiedlich ist, werden zusammen mit diesen Befehlen erläutert.

Die hier verwendete Befehlszeilensyntax entspricht eher den UNIX- als den Windows-Konventionen.

Alle Optionen können entweder mit einem Bindestrich (-) oder mit einem Schrägstrich (/), geschrieben werden, je nachdem, welche Konventionen für das Betriebssystem gelten. Bei Optionen, die laut diesem Kapitel mit Anführungsstrichen versehen werden, wird die Verwendung von Anführungszeichen empfohlen. Es kann jedoch vorkommen, dass die Option auf Ihrem Betriebssystem auch ohne Anführungszeichen verwendet werden kann. Wenn Sie die hier genannten Richtlinien berücksichtigen, sollten keine Fehler auftreten.

Windows-Betriebssysteme erfordern die Verwendung von Anführungszeichen, wenn ein Leerzeichen Teil der Option ist. Ein Revisionskommentar beispielsweise, der aus mehreren Wörtern besteht, muss zwischen Anführungszeichen gesetzt werden. Bei einem Kommentar, der aus einem Wort besteht, ist dies zwar nicht notwendig, die Anführungszeichen können jedoch benutzt werden. Beachten Sie, dass Befehle, die über die Java-Befehlszeile übergeben werden, nicht akzeptiert werden, wenn sich in den Namen von Arbeitsordnern und -dateien Leerzeichen befinden. Leerzeichen können in UNIX-Namen nur verwendet werden, wenn sie zwischen doppelte Anführungszeichen gesetzt werden.

Syntaxkonventionen

Die Syntaxkonventionen für die Befehlszeile lauten wie folgt:

[]

Eckige Klammern umgeben optionale Syntax.

|

Ein senkrechter Strich trennt Optionen, die sich gegenseitig ausschließen. Nur eine der Möglichkeiten kann ausgewählt werden.

-?

Listet die Syntax des Befehls und eine Beschreibung aller Optionen in der Befehlszeile auf.

Auch durch Eingabe von -? wird die Hilfe aufgerufen. -help und -h sind Synonyme für-?. -? kann mit jedem beliebigen Befehl verwendet werden, obwohl dies in der Syntax nicht extra angegeben wird. Diese Informationen werden an `stderr` und nicht an `stdout` gesendet. Um `stderr`-Informationen von der Windows-Befehlszeile aus in einer Datei zu erfassen, verwenden Sie "`2>`" und nicht "`>`". Mit Letzterem werden `stdout`-Informationen erfasst.

-active

Gibt das aktive Prozesselement an.

-cmp

Komprimiert alle zwischen der Workstation und dem Server ausgetauschten Daten und dekomprimiert eingehende Daten. Ohne diese Option erfolgt keine Komprimierung.

Die Komprimierung ist besonders hilfreich, wenn der Client und der Server über eine langsame Verbindung kommunizieren. Mithilfe eines kleinen Testfalls können Sie feststellen, ob die Verwendung der Komprimierung empfehlenswert ist. Dabei sollten Sie prüfen, ob es vorteilhafter für Sie ist, Zeit für die Komprimierung bzw.

Dekomprimierung der Daten aufzuwenden oder längere Zeiten zum Senden unkomprimierter Daten über eine langsame Verbindung in Kauf zu nehmen.

-csf

Wenn der Befehl den Ordner, der mit der Option `-p` angegeben wird, dem zugrunde liegenden StarTeam-Ordner zuordnet, können Sie mithilfe der Option `-csf` die StarTeam-Ordner anhand der Unterschiede in der Groß-/Kleinschreibung auseinander halten. Diese Option hat keinen Einfluss darauf, ob die Groß-/Kleinschreibung der Dateinamen in den Ordnern beachtet wird. Wenn Sie den Befehl `-csf` verwenden, werden die StarTeam-Ordner "doc" und "Doc" beispielsweise als unterschiedliche Ordner erkannt. Ohne diese Option werden beide Ordner als "doc"-Ordner erkannt.

Standardmäßig erfolgt bei Namen von StarTeam-Ordnern keine Unterscheidung aufgrund divergierender Groß-/Kleinschreibung.

Mit oder ohne Verwendung des Befehls `-csf` tritt bei nicht eindeutigen Ordernamen ein Fehler auf. Bei Verwendung von `-csf` sind die Namen von zwei Ordnern nicht eindeutig, wenn ein Ordner mit "Doc" und ein Ordner mit "doc" bezeichnet ist. Wenn Sie `-csf` nicht verwenden, sind Ordernamen bei gleicher Rechtschreibung nicht eindeutig.

-encrypt

Verschlüsselt alle zwischen der Workstation und dem Server ausgetauschten Daten und entschlüsselt eingehende Daten. Ohne diese Option erfolgt keine Verschlüsselung. Die Verschlüsselung schützt Dateien und andere Projektinformationen vor dem Zugriff durch nicht autorisierte Dritte über ungeschützte Netzwerkverbindungen.

Die vollständige Syntax lautet:

```
-encrypt Verschlüsselungstyp
```

Die folgenden Verschlüsselungstypen sind verfügbar:

RC4: RSA RC4 Stromverschlüsselung (schnell)

RC2_ECB: RSA RC2 Blockverschlüsselung (Electronic Codebook)

RC2_CBC: RSA RC2 Blockverschlüsselung (Cipher Block Chaining)

RC2_CFB: (Nur auf Windows-Plattformen) RSA RC2 Blockverschlüsselung (Cipher Feedback)

Die genannten Verschlüsselungstypen sind in der Reihenfolge vom schnellsten zum langsamsten hin geordnet. Je langsamer der Verschlüsselungstyp um so sicherer die Verschlüsselung.

Hinweis: Mit Ausnahme von Windows-Plattformen werden die public- und private-Schlüssel, die im Verschlüsselungsprozess verwendet werden, nicht automatisch erstellt. Sie werden in einer Datei im Stammverzeichnis des Benutzers gespeichert. Die Datei mit diesen Optionen lautet `.starteam`. Sie enthält eine Variable oder Shell-Variable namens `keyfile`. Die `keyfile`-Variable gibt den Speicherort der Datei an, die die public- und private-Schlüssel enthält. Wenn Sie die `keyfile`-Variable nicht angeben, tritt ein Fehler auf. Wenn Sie die `keyfile`-Variable angeben und die Datei nicht vorhanden ist, erstellt der StarTeam-Client ein beliebiges Schlüsselpaar, erstellt die Datei und speichert die Schlüssel darin. Achten Sie darauf, die Datei zu sichern. Unter UNIX sollten Sie beispielsweise darauf achten, dass nur der Besitzer diese Datei lesen kann.

-eol [on|off]

Führt die EOL-Konvertierung für Textdateien durch.

-epwdfilename

Speichert das Passwort eines Benutzers als verschlüsselten Wert in einer lokalen Datei. Diese Funktion unterstützt automatisierte Build-Skripts, die unbeaufsichtigt ausgeführt werden müssen. Das Skript würde den Befehl `stcmd` mit einem bestimmten Benutzernamen und dem Dateinamen aufrufen, der das verschlüsselte Passwort für diesen Benutzernamen enthält. Das Passwort wird dann intern entschlüsselt und an den Server weitergegeben, ohne dass es als Klartext durch das Netzwerk übermittelt wird.

Das Schlüsselwort `-epwdfilename` gibt den Pfad zu der Datei an, die das verschlüsselte Passwort enthält. Genauso wie `-pwdfilename` hat auch `-epwdfilename` Vorrang vor dem mit der `-p`-Option verwendeten Passwort, was verhindert, dass andere Benutzer das Passwort des Benutzers in der Befehlszeile sehen können. Die vollständige Syntax lautet:

```
-epwdfilename "Dateipfad"
```

In der folgenden Tabelle ist die Syntax der Befehle angegeben, die zum Speichern eines verschlüsselten Passworts verwendet werden können.

Aktion	Syntax
Aufforderung zur Eingabe des Passworts, das verschlüsselt und in einer Datei gespeichert wird.	<pre>stcmd store-password -epwdfilename "Dateipfad"</pre>
Das verschlüsselte Passwort als Klartext zu dem Befehl hinzufügen. (Beachten Sie, dass diese Aktion nicht mit dem Wert im Klartext auf das Netzwerk zugreift.)	<pre>stcmd store-password -epwdfilename "Dateipfad" -password "Passwort"</pre>

Nachdem ein verschlüsseltes Passwort gespeichert wurde, können andere `stcmd`-Befehle `-epwdfilename "filePath"` als Parameter angeben. Beispiel:

```
stcmd delete-local -p "JMarsch:Passwort@Orion:1024/StarDraw/StarDraw/SourceCode" -epwdfilename "C:\Arbeitsordner\MeineDatei.txt" -filter "N" "*" 
```

-f NCI

Wendet den Befehl auf alle einzucheckenden Dateien an.

`-f NCI` wird ignoriert, wenn `-filter` verwendet wird.

Dateien...

Gibt die im Befehl zu verwendenden Dateien über den Namen oder über Dateinamensmuster an, z. B. `*.c`. Alle hier genannten Optionen entsprechen den semantischen Konventionen von UNIX, da diese genauer als die Windows-Konventionen sind. Dies bedeutet, dass `"*"` und nicht `"*.*"` "Alle Dateien" bedeutet. Das Muster `"*.*"` bedeutet "Alle Dateien mit Dateinamenserweiterungen". Wenn Sie beispielsweise `"star*.*"` eingeben, werden `starteam.doc` und `starteam.cpp`, aber nicht `starteam` gefunden. Wenn Sie alle Vorkommen mit "starteam" finden möchten, verwenden Sie stattdessen `"star*"`.

Wenn diese Option nicht verwendet wird, lautet die Vorgabe `"*"`. Wenn sie verwendet wird, muss sie immer die letzte Option sein. Alle folgenden Optionen werden nicht berücksichtigt.

Wenn Sie `*` statt `"*"` als Platzhalter für alle Dateien verwenden, erweitert die UNIX-Shell diesen Befehl auf mehrere Elemente und übergibt diese als eine Optionengruppe an den `stcmd`-Befehl. Dies kann zu Problemen führen, z. B. beim Auschecken fehlender Dateien. Verwenden Sie daher `"*"`, um Komplikationen zu vermeiden.

Wenn Sie eine Gruppe von Dateimustern verwenden, sollte jedes Muster separat zwischen Anführungszeichen gesetzt werden. So können Sie beispielsweise `"*.bat"` `"*.c"`, jedoch nicht `"*.bat *.c"` verwenden.

Hinweis: Sie sollten diese Option, unabhängig von der Plattform, jedoch aus unterschiedlichen Gründen in Anführungszeichen setzen. Unter Windows werden Datei- und Ordernamen, die Leerzeichen enthalten, nicht korrekt erkannt, es sei denn, Sie verwenden Anführungszeichen. Wenn Sie unter UNIX keine Anführungszeichen verwenden, wird die Option von der Shell erweitert und anschließend wird die aufgrund der Erweiterung erstellte Elementliste an den Client übergeben. Diese Vorgehensweise führt jedoch häufig zu unerwarteten Ergebnissen. Um entsprechende Fehler zu vermeiden, sollten Sie diese Option immer in Anführungszeichen setzen. Nur in den Fällen, in denen es unbedingt erforderlich ist, dass die Option von der UNIX-Shell erweitert wird, empfiehlt es sich, die Anführungszeichen wegzulassen. Wenn Sie die Anführungszeichen weglassen, kann es auf beiden Plattformen bei der Ausführung von Batch-Skripten zu Fehlern kommen.

In der Dateispezifikation können mehrere Sonderzeichen verwendet werden:

`*`

Entspricht einem beliebigen String, einschließlich einem leeren String. `*` entspricht beispielsweise einem beliebigen Dateinamen mit oder ohne Erweiterung. `"xyz*"` entspricht `"xyz"`, `"xyz.cpp"` und `"xyzutfj"`.

`?`

Entspricht einem beliebigen einzelnen Zeichen. Beispiel: `"a?c"` entspricht `"abc"`, aber NICHT `"ac"`.

`[...]`

Entspricht allen Zeichen, die von der linken und der rechten Klammer eingeschlossen werden.

Mehrere Zeichen, die durch Bindestriche (`-`) getrennt werden, geben die Zeichen an, für die eine Übereinstimmung gefunden werden muss.

Wenn das erste Zeichen nach einer öffnenden, eckigen Klammer (`[`) ein Ausrufezeichen (`!`) oder ein Caret-Zeichen (`^`) ist, dürfen die nachfolgenden Zeichen nicht übereinstimmen. Alle Zeichen außerhalb der Klammer müssen übereinstimmen. Beispiel: `"x[a-d]y"` entspricht `"xby"`, aber nicht `"xey"`. `"x[!a-d]y"` entspricht `"xey"`, aber nicht `"xby"`.

Ein Bindestrich (`-`) oder eine schließende eckige Klammer (`]`) werden gefunden, wenn er bzw. sie als erstes oder als letztes Zeichen der in Klammern gesetzten Zeichen steht.

Wenn Sie ein Sternchen (`*`), Fragezeichen (`?`) oder eine öffnende, eckige Klammer (`[`) in einem Muster verwenden möchten, muss vor diesen Zeichen ein Escape-Zeichen, d. h. ein umgekehrter Schrägstrich (`\`), stehen.

-filter

Dateistatusfilter. Mögliche Statuswerte sind: `C` = Current (Aktuell), `M` = Modified (Geändert), `O` = Out of Date (Veraltet), `N` = Not in View (Nicht in Ansicht), `I` = Missing (Fehlt), `G` = Merge (Zusammenführung) und `U` = Unknown (Unbekannt). Der Filter `CM` wendet beispielsweise einen Befehl nur auf Dateien mit dem Status "Aktuell" oder "Geändert" an.

`-filter` hat Vorrang vor `-f NCI`. Wenn Sie `G`, `O` oder `U` verwenden, müssen Sie auch `-i` oder `-o` angeben. Anderenfalls wird `G`, `O` oder `U` ignoriert.

`-filter` hat außerdem Vorrang vor `-f NCO`. Wenn Sie `G`, `M`, `O` oder `U` verwenden, müssen Sie auch `-merge` oder `-o` angeben, um das Auschecken zu erzwingen. Anderenfalls wird `G`, `M`, `O` oder `U` ignoriert.

-fp

Überschreibt den Arbeitsordner oder das Arbeitsverzeichnis des angegebenen StarTeam-Ordners. Dies entspricht dem Festlegen eines alternativen Arbeitspfads für den Ordner.

Mithilfe dieser Option können Sie einen anderen Arbeitsordner verwenden als den, den Sie in der StarTeam-Ansicht angegeben haben, und zudem die plattformübergreifende Kompatibilität sicherstellen. Bei der Angabe von Laufwerks- und Verzeichnispfadnamen sind UNIX- und Windows-Systeme beispielsweise nicht kompatibel.

Der Pfad `D:\MEINPRODUKT\ENTWICKLUNG\QUELLE` wird unter Windows, jedoch nicht unter UNIX korrekt interpretiert. Mithilfe dieser Option können Sie den Pfad des Arbeitsverzeichnisses angeben, wenn der im StarTeam-Projekt angegebene Pfad von Ihrer Plattform nicht erkannt wird.

Umgekehrte Schrägstriche (`\`) werden als Escape-Zeichen interpretiert, wenn sie Anführungszeichen vorangestellt werden. Aus diesem Grunde tritt im folgenden Beispiel ein Fehler auf:

```
stcmd ci -p "xxx" -fp "C:\ " "*"
```

wird interpretiert als:

```
stcmd ci -p "xxx" -fp "C:" "*"
```

Um einen solchen Fehler zu vermeiden, geben Sie `"C:\\"` wie folgt an:

```
stcmd ci -p "xxx" -fp "C:\\ " "*"
```

Oder geben Sie Folgendes ein, wenn der Pfad `-rp` nicht mit dem Stammordner endet, wie beispielsweise in `C:\orion\`:

```
stcmd ci -p "xxx" -fp "C:\orion" "*"
```

Die vollständige Syntax lautet:

```
-rp "Ordnername"
```

Ordner ist der Windows-Begriff, der auch in der StarTeam-Benutzeroberfläche verwendet wird. Verzeichnis ist der Begriff, der für die UNIX-Plattform verwendet wird.

-h

Verwenden Sie diese Option, um die Hilfe aufzurufen. `-help` kann mit einem beliebigen Befehl verwendet werden. Spezielle Syntaxbeispiele werden hierfür nicht genannt.

-help

Verwenden Sie diese Option, um die Hilfe aufzurufen. `-help` kann mit einem beliebigen Befehl verwendet werden. Spezielle Syntaxbeispiele werden hierfür nicht genannt.

-i

Fordert den Benutzer auf, das Einchecken zu bestätigen, wenn der Dateistatus "Zusammenführung", "Veraltet" oder "Unbekannt" ist.

-is

Übernimmt den Befehl für alle Unterordner. Ohne diese Option wird der Befehl nur für den angegebenen Ordner übernommen.

Wenn diese Option zusammen mit `add-folder` verwendet wird, können Sie einen vollständigen Ordnerzweig zur StarTeam-Ordnerhierarchie hinzufügen. Wenn sie zusammen mit `add-project` verwendet wird, können Sie bereits beim Erstellen eines Projekts mehr Ordnern als nur den Stammordner anlegen.

-l

Sperrt eine Datei.

-mark

Markiert eine Änderungsanforderung als repariert, eine Anforderung als abgeschlossen oder einen Task als fertig gestellt.

-nel

Sperrt eine Datei nicht-exklusiv

-nologo

Mithilfe dieser Option unterdrücken Sie die Anzeige des Copyright-Hinweises. `-nologo` kann mit einem beliebigen Befehl verwendet werden. Spezielle Syntaxbeispiele werden hierfür nicht genannt.

-nomove

Bereits angehängte Labels werden nicht verschoben.

-o

Erzwingt das Einchecken.

-p

Zeigt die zu verwendende Ansicht oder den zu verwendenden Ordner an und stellt den Benutzernamen und das Passwort für den Serverzugriff bereit. Die vollständige Syntax lautet:

```
-p "Benutzername:Passwort@Host-Name:Endpunkt/Projektname/[Ansichtsname/]  
[Ordnerhierarchie/]"
```

Beispiel:

```
-p "bsmith:rocketfive@orion:49201/StarDraw/StarDraw/Quelltext/"
```

- ◆ Wenn der Benutzername weggelassen wird, wird der aktuelle Benutzername verwendet.
- ◆ Wenn das Passwort weggelassen wird, wird der Benutzer aufgefordert, das Passwort einzugeben. Wenn der Benutzer ein Passwort eingibt, werden die Zeichen nicht auf dem Bildschirm angezeigt.

- ◆ Wenn der Host-Name weggelassen wird, wird als Vorgabe localhost verwendet.
- ◆ Die Eingabe eines Endpunkts (Port-Nummer) ist erforderlich. Der Standardwert ist 1024.
- ◆ Der Projektname muss immer angegeben werden.
- ◆ Die Ansicht sollte mithilfe der Ansichtshierarchie gekennzeichnet werden. Verwenden Sie einen Doppelpunkt (:) als Begrenzungszeichen zwischen Ansichtsnamen. Die Ansichtshierarchie sollte stets die Stammansicht einschließen. "StarDraw:Release 4:Service Packs" gibt beispielsweise an, dass die Ansicht "Service Packs" verwendet werden soll, die der Ansicht "Release 4" untergeordnet ist, die wiederum eine untergeordnete Ansicht der zweiten Ebene der Stammansicht "StarDraw" ist. Wenn der Ansichtsname weggelassen wird, wird die Stammansicht verwendet. Wenn die Ansicht die einzige Ansicht dieses Projekts mit diesem Namen ist, reicht es aus, nur den Ansichtsnamen zu verwenden. Diese Vorgehensweise empfiehlt sich jedoch nicht, da im Verlauf des Projekts eine weitere Ansicht mit demselben Namen erstellt werden könnte.
- ◆ Der Ordner sollte mithilfe der Ordnerhierarchie gekennzeichnet werden. Verwenden Sie den Schrägstrich (/) als Begrenzungszeichen zwischen Ordnernamen. Der Stammordner ist in der Ordnerhierarchie niemals enthalten. Lassen Sie die Ordnerhierarchie weg, wenn die Datei sich im Stammordner der Ansicht befindet. Wenn der Name des Stammordners der Ansicht beispielsweise `StarDraw` lautet und Ihre Dateien sich in der Hierarchie `StarDraw/SourceCode/Client` befinden, verwenden Sie nur "SourceCode/Client".

Wenn eine der mit dieser Option verwendeten Variablen Zeichen enthält, die als Begrenzer eingesetzt werden, verwenden Sie das Prozentzeichen (%) gefolgt vom Hex-Wert für die entsprechenden Zeichen. Wenn beispielsweise "@" als Zeichen in einem Passwort verwendet wird, müssen sie es durch "%40" ersetzen.

Ersetzen Sie ":" durch "%3a"

Ersetzen Sie "/" durch "%2f"

Ersetzen Sie "@" durch "%40"

Ersetzen Sie "%" durch "%25"

Unter UNIX und anderen Betriebssystemen muss einigen Sonderzeichen ein umgekehrter Schrägstrich ("\") oder ein anderes Escape-Zeichen vorangestellt werden. Wenn Sie die Option `-p` verwenden, können Sie diese Zeichen durch den entsprechenden Hexcode ersetzen. Beispielsweise kann unter UNIX "%3c" anstelle von "<" verwendet werden.

Ersetzen Sie Leerzeichen durch "%20"

Ersetzen Sie "<" durch "%3c"

Ersetzen Sie ">" durch "%3e"

-pwdfile

Gibt den Pfad zu einer Datei an, in der das Passwort des Benutzers gespeichert ist. Diese Option hat Vorrang vor dem mit der `-p`-Option verwendeten Passwort. Sie verhindert, dass andere Benutzer das Passwort des Benutzers auf der Befehlszeile sehen können. Die vollständige Syntax lautet:

```
-pwdfile "Dateipfad"
```

-q

Unterdrückt das Erstellen von Fortschrittsberichten. Wenn diese Option nicht aktiviert ist, werden beim Ausführen jeder Aktion Meldungen zur jeweiligen Aktion angezeigt.

-r

Grund für das Einchecken.

-rf

Ist dem Namen der Datei vorangestellt, die den Grund für das Einchecken enthält.

-ro

Markiert eine Datei nach der Operation als schreibgeschützt.

-rp

Gibt den Arbeitsordner oder das Arbeitsverzeichnis für den Stammordner der StarTeam-Ansicht an oder setzt diese außer Kraft. Im Befehl `stcmd add-project` wird diese Option verwendet, um den Arbeitsordner für den Stammordner der neuen Ansicht anzugeben. In anderen Befehlen wird diese Option verwendet, um den vorhandenen Arbeitsordner außer Kraft zu setzen.

Mithilfe dieser Option können Sie einen anderen Arbeitsordner verwenden als den, den Sie in der StarTeam-Ansicht angegeben haben, und zudem die plattformübergreifende Kompatibilität sicherstellen. Bei der Angabe von Laufwerks- und Verzeichnispfadnamen sind UNIX- und Windows-Systeme beispielsweise nicht kompatibel.

Der Pfad `D:\MEINPRODUKT\ENTWICKLUNG\QUELLE` wird unter Windows, jedoch nicht unter UNIX korrekt interpretiert. Mithilfe dieser Option können Sie den Pfad des Arbeitsverzeichnisses angeben, wenn der im StarTeam-Projekt angegebene Pfad von Ihrer Plattform nicht erkannt wird.

Die UNIX-Shell interpretiert einen umgekehrten Schrägstrich (`\`) als Escape-Zeichen, wenn er bestimmten Zeichen (z. B. Anführungszeichen) vorangestellt wird. Aus diesem Grunde tritt im folgenden Beispiel ein Fehler auf:

```
stcmd ci -p "xxx" -rp "C:\ " "*"
```

wird interpretiert als:

```
stcmd ci -p "xxx" -rp "C:" "*"
```

Um einen solchen Fehler zu vermeiden, geben Sie `"C:\\"` wie folgt an:

```
stcmd ci -p "xxx" -rp "C:\\ " "*"
```

Oder geben Sie Folgendes ein, wenn der Pfad `-rp` nicht mit dem Stammordner endet, wie beispielsweise in `"C:\orion\"`:

```
stcmd ci -p "xxx" -rp "C:\orion" "*"
```

Die vollständige Syntax lautet:

```
-rp "Ordnername"
```

Ordner ist der Windows-Begriff, der auch in der StarTeam-Benutzeroberfläche verwendet wird. Verzeichnis ist der Begriff, der für die UNIX-Plattform verwendet wird.

-rw

Vergibt Lese- und Schreibrechte für eine Datei.

-stop

Wird häufig in Verbindung mit `-x` verwendet. Stoppt die Ausführung des Befehls, wenn der erste Fehler erkannt wird. Ohne diese Option wird die Ausführung trotz Fehlern fortgeführt.

-u

Entsperrt eine Datei.

-v

Versions-Label.

-x

Wechselt zwischen dem interaktiven und dem Batch-Modus. Wird diese Option nicht verwendet, müssen Sie Fehlermeldungen interaktiv bestätigen und der Exit-Code ist möglicherweise nicht verfügbar. Wird diese Option verwendet, werden keine Fehlermeldungen angezeigt und der Exit-Code wird gesetzt. Die Exit-Codes sind `0`, erfolgreich, und `1`, fehlgeschlagen.

Verwandte Konzepte

[Sonderzeichen](#)

Verwandte Referenz

[Weitere Client-Befehlszeilenoperationen](#)

Sonderzeichen

* entspricht einem beliebigen String, einschließlich einem leeren String. Beispiel: "x*z" entspricht "xyz" und "xz". ? entspricht einem beliebigen einzelnen Zeichen. Beispiel: "a?c" entspricht "abc", aber NICHT "ac".

[...] Entspricht allen Zeichen, die von der linken und der rechten Klammer eingeschlossen werden.

Mehrere Zeichen, die durch Bindestriche (-) getrennt werden, geben die Zeichen an, für die eine Übereinstimmung gefunden werden muss. Wenn das erste Zeichen nach einer nach links zeigenden Klammer ([) ein Ausrufezeichen (!) oder ein Caret-Zeichen (^) ist, dürfen die nachfolgenden Zeichen nicht übereinstimmen. Alle Zeichen außerhalb der Klammer müssen übereinstimmen. Eine Übereinstimmung für einen Bindestrich (-) oder eine nach rechts zeigende Klammer (]) kann gefunden werden, wenn er bzw. sie als erstes oder als letztes Zeichen mehrere in Klammern gesetzter Zeichen steht. Beispiel: "x[a - d]y" entspricht "xby", jedoch nicht "xey". "x[!a - d]y" entspricht "xey", jedoch nicht "xby". Wenn Sie ein Sternchen (*), Fragezeichen (?) oder eine nach links zeigende Klammer ([) in einem Muster verwenden möchten, muss vor diesen Zeichen ein Escape-Zeichen (d. h. ein umgekehrter Schrägstrich \) stehen.

Wenn Sie * statt "*" zum Anzeigen aller Dateien verwenden, erweitert die UNIX-Shell diesen Befehl auf mehrere Elemente und übergibt diese als eine Optionengruppe an den `stcmd`-Befehl. Dies kann zu Problemen (z. B. beim Auschecken fehlender Dateien) führen. Verwenden Sie daher "*", um Komplikationen zu vermeiden. Wenn Sie eine Gruppe von Dateimustern verwenden, sollte jedes Muster separat zwischen Anführungszeichen gesetzt werden. So können Sie beispielsweise "*.bat" "*.c", jedoch nicht "*.bat *.c" verwenden.

Diese Sonderzeichen gelten auch für die Option `files...`, die in einigen Befehlen vorkommt.

Verwandte Referenz

[Weitere Client-Befehlszeilenoperationen](#)

Exit-Codes

Die `stcmd`-Befehle geben Exit-Codes zurück, wenn die Option `-x` im Befehl verwendet wird. Es werden folgende Codes zurückgegeben: Der Exit-Code 0 zeigt an, dass die Aktion erfolgreich war, der Exit-Code 1 zeigt an, dass ein Fehler aufgetreten ist. 101 wird zurückgegeben, wenn mindestens eins der angegebenen Dateimuster nicht übereinstimmte, und 102 wird zurückgegeben, wenn keines der angegebenen Dateimuster übereinstimmte. Der Befehl `stcmd diff` verfügt über die zusätzliche Option (`-e`), mit der Exit-Codes zurückgegeben werden können. Mit der Option `-e` werden drei Exit-Codes zurückgegeben (0, 1 und 2), die mit den oben aufgeführten Codes nicht übereinstimmen. Der Code 1 wird möglicherweise nicht ausgegeben, wenn die Option `-x` nicht verwendet wird.

Auf Windows-Plattformen

Verwenden Sie `ERRORLEVEL` in einer Batch-Datei, wenn Sie Operationen basierend auf dem Ergebnis eines Befehls ausführen möchten. Nach einem `stcmd`-Befehl in einer Batch-Datei können Sie beispielsweise folgende Anweisung verwenden:

```
IF ERRORLEVEL int Anweisung
```

wobei int 0 oder 1 sein kann.

Beispiel:

```
IF NOT ERRORLEVEL 1 GOTO OPOK
ECHO ERROR OCCURRED AT STEP5>LOGFILE.TXT.

:OPOK
```

Unter Windows NT können Sie auch die Pseudo-Umgebungsvariable `%ERRORLEVEL%` verwenden. Sie können beispielsweise Folgendes in einer Shell oder in der Befehlszeile verwenden (nach einem `stcmd`-Befehl):

```
SET /A STEPNUMBER=5
SET /A THISERROR=STEPNUMBER*ERRORLEVEL
SET /A ERRORMASK=+THISERROR
```

UNIX

Unter UNIX verfügt jede Shell über eine eigene Methode, um Exit-Codes auszuwerten. In der Bourne-Shell kann beispielsweise folgende Anweisung auf einen `stcmd`-Befehl folgen:

```
if [ return ]; then statement
```

Verwandte Referenz

[Weitere Client-Befehlszeilenoperationen](#)

Dateien hinzufügen: stcmd add

Verwenden Sie den Befehl `stcmd add`, um Dateien von der Befehlszeile aus zu einem Projekt hinzuzufügen.

Die hinzugefügten Dateien können gleichzeitig mit einem Prozesselement verknüpft werden. Alle durch diesen Befehl erfolgreich hinzugefügten Dateien werden mit der aktuellen Revision des Prozesselements verknüpft und an dieses angeheftet. Verwenden Sie die Option `-active`, um das gerade aktive Prozesselement anzugeben (das Sie zuvor mit einem Client auf Ihrer Workstation festgelegt haben).

Wenn kein Element aktiv ist oder Sie ein anderes Element verwenden möchten, nutzen Sie die Option, die den Typ des Prozesselements angibt (`-cr`, `-req` oder `-task`), gefolgt vom vollständigen Pfad des Stammordners aus der StarTeam-Projektansicht des Elements. Verwenden Sie dabei den Schrägstrich (`/`) als Trennzeichen zwischen Ordernamen. Geben Sie für Prozesselemente, die sich außerhalb der Ansicht befinden, den Projektnamen und den Ansichtsnamen vor dem vollständigen Ordnerpfad an. Verwenden Sie einen Doppelpunkt (`:`) als Trennzeichen für den Pfad der Ansicht. Mit `-cr MyProject/RootView:ChildView/SourceCode/37` geben Sie beispielsweise Änderungsanforderung 37 im Ordner `SourceCode` der Ansicht `ChildView` des Projekts `MyProject` an. Bei der Ausführung des Prozesses wird zuerst davon ausgegangen, dass sich das Prozesselement in der aktuellen Ansicht befindet. Daher wird geprüft, ob der vollständige Pfad einem Ordnerpfad in dieser Ansicht entspricht. Wird das Prozesselement in der aktuellen Ansicht nicht gefunden, wird es als Prozesselement behandelt, das sich außerhalb der Ansicht befindet, und die Suche nach dem Element beginnt auf Projekt- und Ansichtsebene.

Verwenden Sie die Option `-mark`, um das Prozesselement je nach Typ gleichzeitig als "repariert", "fertig gestellt" oder "abgeschlossen" zu markieren. Eine Änderungsanforderung kann beispielsweise als "repariert" markiert werden. Elemente werden so lange nicht als "repariert", "fertig gestellt" oder "abgeschlossen" markiert, bis alle Dateien erfolgreich hinzugefügt wurden.

Syntax

Die Syntax für diesen Befehl lautet wie folgt:

```
stcmd add -p "Projektspezifikation" [-pwdfile "Dateipfad"] [-cmp] [-csf] [-encrypt  
Verschlüsselungstyp]  
[-is] [-q] [-x] [-stop] [-rp "Ordnerpfad" | -fp "Ordnerpfad"] [- l | -u | -nel] [-ro | -rw]  
[-d "Beschreibung"]  
[-vl "Label"] [-eol [on | off]] [[ -active | [-cr | -req | -task ] Prozesselementpfad] [-  
mark]] [-short] [Dateien...]
```

Option	Beschreibung
<code>-active</code>	Das aktive Prozesselement.
<code>-cr</code>	Vollständiger Pfad vom Stammordner der Projektansicht zu der Änderungsanforderungs-, Anforderungs- oder Task-Nummer, die als Prozesselement verwendet werden soll. Verwenden Sie den Schrägstrich (<code>/</code>) als Begrenzungszeichen zwischen Ordernamen.
<code>-req</code>	
<code>-task</code>	Geben Sie für Prozesselemente, die sich außerhalb der Ansicht befinden, den Projektnamen und den Ansichtsnamen vor dem vollständigen Ordnerpfad an. Verwenden Sie einen Doppelpunkt (<code>:</code>) als Trennzeichen für den Pfad der Ansicht. Mit <code>-cr MyProject/RootView:ChildView/SourceCode/37</code> geben Sie beispielsweise Änderungsanforderung 37 im Ordner <code>SourceCode</code> der Ansicht <code>ChildView</code> des Projekts <code>MyProject</code> an. Bei der Ausführung des Prozesses wird zuerst davon ausgegangen, dass sich das Prozesselement in der aktuellen Ansicht befindet. Daher wird geprüft, ob der vollständige Pfad einem Ordnerpfad in dieser Ansicht entspricht. Wird das Prozesselement in der aktuellen Ansicht nicht gefunden, wird es als Prozesselement behandelt, das sich außerhalb der Ansicht befindet, und die Suche nach dem Element beginnt auf Projekt- und Ansichtsebene.
<code>-d</code>	Legt die Beschreibung der neu hinzugefügten Dateien zur angegebenen Beschreibung fest. Die Beschreibung wird in doppelte Anführungszeichen gesetzt.

<code>-eol</code>	<p>Konvertiert automatisch EOL-Markierungen in Carriage Return/Zeilenvorschub-Zeichen (CR/LF), wenn Arbeitsdateien an das Repository des Servers übertragen werden. Wenn diese Option aktiviert ist, wird die aktuelle EOL-Markierung der Arbeitsdatei in eine CR/LF-Kombination konvertiert. Wenn sie deaktiviert ist (Standard), wird keine Zeilenendekonvertierung ausgeführt.</p> <p>Sie aktivieren diese Option beispielsweise, wenn Sie eine Arbeitsdatei von einem UNIX-Computer aus zum Repository hinzufügen und möchten, dass das Repository Textdateien mit "CR/LF" am Zeilenende speichert.</p>
<code>-l</code>	<p>Sperrt jede Datei, nachdem sie zur Ansicht hinzugefügt wurde. Ohne <code>-l</code>, <code>-u</code> oder <code>-nel</code> sind die Dateien standardmäßig nicht gesperrt.</p>
<code>-mark</code>	<p>Wenn alle Dateien erfolgreich hinzugefügt wurden, gibt diese Option an, dass der Status des Prozesselements in "Repariert" (bei einer Änderungsanforderung), "Fertig gestellt" (bei einer Task) oder "Abgeschlossen" (bei einer Anforderung) geändert wird. Die Dateien werden mit dem neuen Status an die Revision angeheftet.</p>
<code>-nel</code>	<p>Sperrt jede Datei nicht exklusiv, nachdem sie hinzugefügt wurde.</p>
<code>-ro</code>	<p>Definiert nach dieser Operation die Arbeitsdatei als schreibgeschützt. Ohne diese Option bliebe der Status der Datei derselbe wie vor der Operation. In der Regel verwenden Sie <code>-ro</code>, um zu verhindern, dass Sie selbst eine Datei bearbeiten, die Sie nicht gesperrt haben.</p> <p><code>-ro</code> muss mit <code>-l</code>, <code>-u</code> oder <code>-nel</code> verwendet werden. Wenn Sie <code>-ro</code> verwenden, ist die Verwendung von <code>-rw</code> nicht möglich.</p>
<code>-rw</code>	<p>Gibt die Arbeitsdatei nach dieser Operation zum Lesen und Schreiben frei. Ohne diese Option bliebe der Status der Datei derselbe wie vor der Operation.</p> <p><code>-rw</code> muss mit <code>-l</code>, <code>-u</code> oder <code>-nel</code> verwendet werden. Wenn Sie <code>-rw</code> verwenden, ist die Verwendung von <code>-ro</code> nicht möglich.</p>
<code>-u</code>	<p>Lässt die neu hinzugefügten Dateien entsperrt.</p>
<code>-vl</code>	<p>Legt ein Label für die neuen Dateien fest. Das Label wird in doppelte Anführungszeichen gesetzt. Diese Option kann mehrmals im Befehl verwendet werden. Das Label kann entweder ein Ansichts- oder ein Revisions-Label sein, es muss aber bereits in der Anwendung vorhanden sein.</p>

Beispiel

Im folgenden Beispiel wird der Befehl `stcmd add` verwendet, um alle `.doc`-Dateien mit dem Status "Nicht in Ansicht" zum Ordner "User Manual" hinzuzufügen. "User Manual" ist ein Unterordner des Stammordners "StarDraw" (in der Ansicht "StarDraw" des Projekts "StarDraw"). Die Dateien werden gesperrt und erhalten die Beschreibung "Erster Entwurf des Kapitels".

```
stcmd add
-p "JMarsch:Passwort@Orion:1024/StarDraw/StarDraw/User Manual" -l-d
"Erster Entwurf des Kapitels" "*.doc"
```

Verwandte Referenz

[Weitere Client-Befehlszeilenoperationen](#)

Ordner hinzufügen: stcmd add-folder

Verwenden Sie den Befehl `stcmd add-folder`, um StarTeam-Ordner von der Befehlszeile aus zu einer Ansicht hinzuzufügen. Sie können Ordner zum Stammordner oder allen anderen Ordnern der Ansicht hinzufügen. Der Arbeitsordner des neuen StarTeam-Ordners wird standardmäßig in StarTeam und nicht auf Ihrer Workstation erstellt. Der Arbeitsordner erhält denselben Namen wie der StarTeam-Ordner. Er ist ein Unterordner des Arbeitsordners für den übergeordneten Ordner des StarTeam-Ordners.

Wenn Sie beispielsweise einen StarTeam-Ordner mit dem Namen "Experte" erstellen, ergeben sich folgende Beziehungen: "Experte" ist ein Unterordner des StarTeam-Ordners mit dem Arbeitsordner `C:\StarDraw`. Der Arbeitsordner für "Experte" ist daher `C:\StarDraw\Experte`.

Mit der Option `-is` können Sie einen ganzen Ordnerzweig zur Ordnerhierarchie der Projektansicht hinzufügen. Verwenden Sie mit der Option `-is` die Optionen `-rp` oder `-fp`, um einen Ordner auf Ihrer Workstation anzugeben, dessen Unterordner die Unterordner des neuen StarTeam-Ordners werden. Die Verwendung der Option `-fp` wird empfohlen, da sie den direkten Pfad zum übergeordneten Ordner der Unterordner angibt. Die Option `-rp` dagegen gibt den Pfad des Arbeitsordners für den Stammordner der Ansicht an und hängt die Namen der StarTeam-Ordner in der Hierarchie vom Stammordner bis zum neuen Ordner an den angegebenen Pfad an. Die Optionen `-rp` und `-fp` werden nur bei Verwendung der Option `-fp` aktiv.

Syntax

Die Syntax für diesen Befehl lautet wie folgt:

```
stcmd add-folder -p
  "Projektspezifikation" [-pwdfile
  "Dateipfad"] [-cmp] [-csf]
  [-encrypt Verschlüsselungstyp] [-is]
  [-q] [-x]
  [-stop] [-rp
  "Ordnerpfad" | -fp "Ordnerpfad"]
  -name "Ordnername" [-d
  "Beschreibung"] [-ex
  "Ausschlusstyp"] [-exlist
  "Dateimaske"
  | -exfile "Dateiname"]
```

Option	Beschreibung
<code>-d</code>	Gibt eine Beschreibung für den neuen Ordner an. Es können dazu maximal 254 Zeichen verwendet werden.
<code>-ex</code>	Gibt die von diesem neuen Ordner zu verwendenden Ausschlusslisten an. Ausschlusslisten schließen bestimmte Dateien oder Dateitypen von der Anzeige aus. Wenn eine Arbeitsdatei im Arbeitsordner dieses Ordners den Status "Nicht in Ansicht" hätte, aber einer Dateispezifikation in einer der Ausschlusslisten entspräche, würde die Anwendung diese Datei nicht anzeigen. Es wäre, als würde die Datei nicht existieren. Angenommen, Sie erstellen Dateien mit einer Anwendung, die bei jedem Speichervorgang automatisch Backup-Kopien (.bak) der entsprechenden Datei erstellt. Ihr Arbeitsordner kann mehrere .bak-Dateien enthalten, aber es besteht kein Grund, sie zur Projektansicht hinzuzufügen. Da es in der Anwendung überflüssig ist, diese .bak-Dateien als mögliche Kandidaten anzuzeigen, schließen Sie sie also aus. Das Ausschließen von Dateien erfolgt ordnerspezifisch. Ausschlusslisten von übergeordneten Ordnern können jedoch übernommen werden. Die vollständige Syntax lautet: <code>-ex excludeType</code> Es gibt folgende Typen: <code>inherit</code> bedeutet, dass dieser Ordner alle von seinem übergeordnetem Ordner verwendeten Ausschlusslisten übernimmt und die Ausschlussliste verwendet, die entweder mit <code>-exfile</code> oder <code>-exlist</code> angegeben ist (sofern eine erstellt wurde). Dies ist die Vorgabe.

`local` bedeutet, dass dieser Ordner nur die Ausschlussliste verwendet, die entweder mit `-exfile` oder `-exlist` angegeben ist.

`none` bedeutet, dass dieser Ordner keine Ausschlusslisten verwendet, unabhängig davon, was Sie mit `-exfile` oder `-exlist` angeben.

`-exfile` Der Pfad auf die Datei, die die lokalen Ausschlusslisten für diesen Ordner enthält. Eine Beschreibung des Inhalts von Ausschlusslisten finden Sie unter `-exlist`.

`-exlist` Die lokale Ausschlussliste für diesen Ordner. Es können dazu maximal 254 Zeichen verwendet werden. Geben Sie eine oder mehrere Dateispezifikationen an (unter Verwendung der Standardplatzhalter * und ?), die durch Kommas, Leerzeichen oder Semikolons getrennt sind. Um ein Komma, ein Leerzeichen oder ein Semikolon als Teil einer Spezifikation anzugeben, fassen Sie diese in doppelte Anführungszeichen ein.

Beispiel: `*.exe,*.dll p*z.doc;*.t?t "test *.* "`

Wenn Sie doppelte Anführungszeichen in Ihrer Ausschlussliste oder eine lange Ausschlussliste verwenden, sollten Sie die Option `-exfile` verwenden. Mit `-exlist` muss jedem Anführungszeichen in der Ausschlussliste das Escape-Zeichen für Ihr System oder Ihre Shell vorangestellt werden. Auf NT-Systemen ist beispielsweise das Caret (^) erforderlich. Bei Verwendung von `-exfile` sind keine Escape-Zeichen erforderlich.

`-name` Der Name des Ordners. Es können dazu maximal 254 Zeichen verwendet werden. Wenn die Ausschlussliste doppelte Anführungszeichen enthält, muss kein Escape-Zeichen verwendet werden.

Beispiel

Im folgenden Beispiel wird der Befehl `stcmd add-folder` verwendet, um einen Ordner namens "Experte" als Unterordner des Stammordners der StarDraw-Projektansicht, "StarDraw", zu erstellen. Zusätzlich wird eine lokale Ausschlussliste für "Experte" festgelegt. "Experte" verwendet standardmäßig die Ausschlussliste seines übergeordneten Ordners und die lokal festgelegte Liste.

```
stcmd add-folder -p "JMarsch:Passwort@Orion:1024/StarDraw/StarDraw/"
-name "Experte" -d "StarDraw-Setup-Experte" -exlist
"*.bak"
```

Im nächsten Beispiel wird derselbe Ordner wie im vorherigen Beispiel erstellt. Zusätzlich werden die Unterordner mit einbezogen. Der Ordner mit dem Pfad "C:\Experte" verfügt über Unterordner ("Quelle", "Spez" und "Doku"). Diese Unterordner werden zusätzlich zum Ordner "Experte" als StarTeam-Ordner hinzugefügt. Alle neuen Ordner ("Experte", "Quelle", "Spez" und "Doku") erhalten unabhängig von den Einstellungen für `-fp` Arbeitsordner, die standardmäßig durch den StarTeam-Server zugewiesen werden. "Experte" ist der übergeordnete Ordner für "Quelle", "Spez" und "Doku". "StarDraw" ist der übergeordnete Ordner für "Experte".

```
stcmd add p "JMarsch:Passwort@Orion:1024/StarDraw/ StarDraw/"
-name "Experte" -d "StarDraw-Setup-Experte" -is -fp
"C:\Experte" exlist "*.bak"
```

Verwandte Referenz

[Weitere Client-Befehlszeilenoperationen](#)

Projekte hinzufügen: stcmd add-project

Verwenden Sie den Befehl `stcmd add-project`, um Projekte von der Befehlszeile aus zu einer Serverkonfiguration hinzuzufügen. Beim Erstellen eines Projekts werden auch seine Stammsicht deren Stammordner erstellt. Bei der Ausführung dieses Befehls wird mit der Option `-rp` der Arbeitsordner für diesen Stammordner angegeben.

Mit `-is` werden die Unterordner des Arbeitsordners als Unterordner des Stammordners in der StarTeam-Ordnerhierarchie verwendet.

Syntax

Die Syntax für diesen Befehl lautet wie folgt:

```
stcmd add-project [-pwdfile "Dateipfad"
] [-cmp] [-encrypt
Verschlüsselungstyp] [-is]
[-q] [-x]
[-stop] -s
"Servername" -name "Projektname"
-rp "Ordnerpfad"
[-d "Beschreibung"]
[-kw "Dateimaske"
|-kwfile "Dateiname"]
[-ex "Ausschlusstyp" ] [-exlist
"Dateimaske"
|-exfile "Dateiname"]
```

Option	Beschreibung
--------	--------------

<code>-d</code>	Die Beschreibung für das Projekt. Es können dazu maximal 254 Zeichen verwendet werden.
-----------------	--

<code>-ex</code>	Die mit dem Stammordner des Projekts zu verwendenden Ausschlusslisten.
------------------	--

Ausschlusslisten schließen bestimmte Dateien oder Dateitypen von der Anzeige aus. Wenn eine Arbeitsdatei im Arbeitsordner dieses Ordners den Status "Nicht in Ansicht" hätte, aber einer Dateispezifikation in einer der Ausschlusslisten entspräche, würde die Anwendung diese Datei nicht anzeigen. Es wäre, als würde die Datei nicht existieren.

Angenommen, Sie erstellen Dateien mit einer Anwendung, die bei jedem Speichervorgang automatisch Backup-Kopien (.bak) der entsprechenden Datei erstellt. Ihr Arbeitsordner kann mehrere .bak-Dateien enthalten, aber es besteht kein Grund, sie zur Projektansicht hinzuzufügen. Da es in der Anwendung überflüssig ist, diese .bak-Dateien als mögliche Kandidaten anzuzeigen, schließen Sie sie also aus. Das Ausschließen von Dateien erfolgt ordnerspezifisch. Ausschlusslisten von übergeordneten Ordnern können jedoch übernommen werden. Die vollständige Syntax lautet:

```
-ex excludeType
```

Es gibt folgende Typen:

`inherit` bedeutet, dass der Stammordner alle von seinem übergeordnetem Ordner verwendeten Ausschlusslisten übernimmt und die Ausschlussliste verwendet, die entweder mit `-exfile` oder `-exlist` angegeben ist (sofern eine erstellt wurde). Dies ist die Vorgabe, obwohl der Stammordner nichts übernehmen kann.

`local` bedeutet, dass der Stammordner nur die Ausschlussliste verwendet, die entweder mit `-exfile` oder `-exlist` angegeben ist.

`none` bedeutet, dass der Stammordner keine Ausschlusslisten verwendet, unabhängig davon, was Sie mit `-exfile` oder `-exlist` angeben.

<code>-exfile</code>	Der Pfad auf die Datei, die die lokale Ausschlussliste für diesen Stammordner enthält. Eine Beschreibung des Inhalts von Ausschlusslisten finden Sie unter "-exlist".
----------------------	---

<code>-exlist</code>	Die lokale Ausschlussliste für den Stammordner. Es können dazu maximal 254 Zeichen verwendet werden. Geben Sie eine oder mehrere Dateispezifikationen an (unter Verwendung der Standardplatzhalter * und ?), die durch Kommas, Leerzeichen oder Semikolons getrennt sind. Um ein Komma, ein Leerzeichen oder ein Semikolon als Teil einer Spezifikation anzugeben, fassen Sie diese in doppelte Anführungszeichen ein. Beispiel: <pre>*.exe,*.dll p*z.doc;*.t?t "test *.*"</pre> <p>Wenn Sie doppelte Anführungszeichen in Ihrer Ausschlussliste oder eine lange Ausschlussliste verwenden, sollten Sie die Option -exfile verwenden. Mit -exlist muss jedem Anführungszeichen in der Ausschlussliste das Escape-Zeichen für Ihr System oder Ihre Shell vorangestellt werden. Auf NT-Systemen ist beispielsweise das Caret (^) erforderlich. Bei Verwendung von -exfile sind keine Escape-Zeichen erforderlich.</p>
----------------------	---

<code>-kw</code>	Die Erweiterungen der Dateien, für die Schlüsselwörter verwendet werden sollen. Es können dazu maximal 254 Zeichen verwendet werden. Geben Sie eine oder mehrere Dateispezifikationen an (unter Verwendung der Standardplatzhalter * und ?), die durch Kommas, Leerzeichen oder Semikolons getrennt sind. Um ein Komma, ein Leerzeichen oder ein Semikolon als Teil einer Spezifikation anzugeben, fassen Sie diese in doppelte Anführungszeichen ein. Beispiel: <pre>*.cpp,*.h p*z.doc;*.t?t "test *.*"</pre> <p>Wenn Sie doppelte Anführungszeichen in Ihrer Schlüsselwortliste oder eine lange Liste verwenden, sollten Sie die Option -kwfile verwenden. Mit -kwlist muss jedem Anführungszeichen in der Schlüsselwortliste das Escape-Zeichen für Ihr System oder Ihre Shell vorangestellt werden. Auf NT-Systemen ist beispielsweise das Caret (^) erforderlich. Bei Verwendung von -kwfile sind keine Escape-Zeichen erforderlich.</p>
------------------	---

<code>-kwfile</code>	Der Pfad für die Datei mit der entsprechenden Dateierweiterung, für die Schlüsselwörter verwendet werden sollen.
----------------------	--

<code>-name</code>	Der Name des Projekts. Es können dazu maximal 254 Zeichen verwendet werden.
--------------------	---

<code>-s</code>	Identifiziert den Server. Die vollständige Syntax lautet: <pre>-s "userName:password@host:portNumber"</pre> <p>Beispiel: <pre>-s "JMarsh:password@orion:49201"</pre> <p>Wenn der Benutzername weggelassen wird, wird der aktuelle Benutzername verwendet. Der Benutzername im Beispiel lautet "JMarsch". <p>Wenn das Passwort weggelassen wird, wird der Benutzer aufgefordert, das Passwort einzugeben. Das Passwort im Beispiel lautet "Passwort". Wenn der Host-Name weggelassen wird, wird als Vorgabe localhost verwendet. Der Host-Name im Beispiel lautet "orion". <p>Die Port-Nummer ist erforderlich. In diesem Beispiel wird die Standard-Port-Nummer 49201 verwendet.</p> </p></p></p>
-----------------	--

Beispiel

Im folgenden Beispiel wird der Befehl `stcmd add-project` verwendet, um ein Projekt namens "Integrationen" auf dem Computer "Orion" zu erstellen. (Auf Orion wird eine Instanz des StarTeam-Servers mit einer Serverkonfiguration ausgeführt, die Port 49201 verwendet.) Mit folgendem Befehl wird das Projekt erstellt. Es wird festgelegt, dass Daten, die von den Workstations zum Server und umgekehrt versendet werden, komprimiert und verschlüsselt werden und das Projekt erhält eine Beschreibung.

```
stcmd add-project --s "JMarsch:Passwort@Orion:1024"
-cmp -encrypt "RC4" -name "Integrationen" -rp "C:\integrationen"
-d "Integrationen unserer Produkte mit Partnerprodukten"
```

Verwandte Referenz

[Weitere Client-Befehlszeilenoperationen](#)

Ansichten hinzufügen: stcmd add-view

Verwenden Sie den Befehl `stcmd add-view`, um Ansichten von der Befehlszeile aus zu einer Serverkonfiguration hinzuzufügen. Beim Erstellen der Ansicht werden die übergeordnete Ansicht und der Stammordner mit der Option `-p` angegeben. Bei der Ausführung dieses Befehls wird mit der Option `-rp` der Arbeitsordner für den Stammordner angegeben. Verwenden Sie folgende Optionen, um die jeweiligen Ansichtstypen zu erstellen:

- ◆ Verwenden Sie `-dr`, um eine Referenzansicht mit Lese- und Schreibzugriff zu erstellen.
- ◆ Verwenden Sie `-dr -ro`, um eine Referenzansicht nur mit Lesezugriff zu erstellen.
- ◆ Verwenden Sie `-dr -ba`, um eine Ansicht mit Verzweigungen zu erstellen, in der für vorhandene Elemente die Option "Verzweigen bei Änderung" aktiviert ist.
- ◆ Verwenden Sie `-dr -bn`, um eine Verzweigungsansicht zu erstellen, in der für vorhandenen Elemente die Option "Verzweigen bei Änderung" nicht aktiviert ist.
- ◆ Wenn Sie `-dr` nicht verwenden, wird eine leere Ansicht erstellt.

Syntax

Die Syntax für diesen Befehl lautet wie folgt:

```
stcmd add-view -p "Projektspezifikation" [-pwdfile "Dateipfad"] [-cmp] [-encrypt  
Verschlüsselungstyp]  
[-q] [-x] [-stop] -name "Ansichtsname" [-rp "Dateipfad"] [-d "Beschreibung"] [-dr [-ro | -  
ba | -bn  
[-cfgl "Label-Name" | -cfgp "Statusname" | -cfgd "Datum"]]]
```

Option	Beschreibung
<code>-ba</code>	Verwenden Sie diese Option mit <code>-dr</code> , um eine Verzweigungsansicht zu erstellen, in der für vorhandene Elemente die Option "Verzweigen bei Änderung" aktiviert ist. Der Wert der Ansichtseigenschaft "Bei Änderung von gemeinsam genutzten Elementen in der Ansicht verzweigen" ist anfänglich aktiviert. Diese Option muss mit <code>-dr</code> verwendet werden.
<code>-bn</code>	Wenn diese Option zusammen mit <code>-dr</code> verwendet wird, gibt sie eine Verzweigungsansicht an, in der für vorhandene Elemente die Option "Verzweigen bei Änderung" nicht aktiviert ist. Der Wert der Ansichtseigenschaft "Bei Änderung von gemeinsam genutzten Elementen in der Ansicht verzweigen" ist anfänglich deaktiviert. Diese Option muss mit <code>-dr</code> verwendet werden.
<code>-cfgd</code>	Konfiguriert die Ansicht mit den angegebenen Datums-/Uhrzeitangaben. Folgende Beispiele sind u. a. möglich: "03.01.05 11:25" "3. Januar 2005 11:25:00 CET" "Montag, 3. Januar 2005 11:25 Uhr CET" Diese Option muss mit einer der folgenden Kombinationen verwendet werden: <code>-dr -ro</code> , <code>-dr -ba</code> oder <code>-dr -bn</code> .
<code>-cfgl</code>	Konfiguriert die Ansicht mit dem angegebenen Label. Ohne <code>-cfgl</code> , <code>-cfgp</code> oder <code>-cfgd</code> wird die aktuelle Konfiguration der Ansicht verwendet. Diese Option muss mit einer der folgenden Kombinationen verwendet werden: <code>-dr -ro</code> , <code>-dr -ba</code> oder <code>-dr -bn</code> .
<code>-cfgp</code>	Konfiguriert die Ansicht mit dem angegebenen Heraufstufungsstatus. Diese Option muss mit einer der folgenden Kombinationen verwendet werden: <code>-dr -ro</code> , <code>-dr -ba</code> oder <code>-dr -bn</code> .
<code>-d</code>	Die Beschreibung für die Ansicht. Es können dazu maximal 254 Zeichen verwendet werden.

<code>-dr</code>	<p>Eine abgeleitete Ansicht. Mit Ausnahme leerer Ansichten sind alle Ansichten abgeleitet. Siehe auch <code>-ba</code>, <code>-bn</code> und <code>-ro</code>.</p> <p>In Kombination mit <code>-ba</code>, <code>-bn</code> oder <code>-ro</code> wird eine Referenzansicht mit Lese-/Schreibzugriff erstellt. Die Konfiguration der Referenzansicht mit Lese-/Schreibzugriff ist mit der Konfiguration ihrer übergeordneten Ansicht identisch. Wenn Sie daher <code>-dr</code> ohne <code>-ba</code>, <code>-bn</code> oder <code>-ro</code>, aber mit <code>-cfdgl</code>, <code>-cfdgp</code> oder <code>-cfdgd</code> verwenden, wird eine Fehlermeldung angezeigt.</p> <p>Wenn Sie diese Option verwenden, wird eine leere Ansicht erstellt. Der Wert der Ansichtseigenschaft "Bei Änderung von gemeinsam genutzten Elementen in der Ansicht verzweigen" ist für leere Ansichten anfänglich deaktiviert.</p>
<code>-name</code>	Der Name der Ansicht. Es können dazu maximal 254 Zeichen verwendet werden.
<code>-ro</code>	In Kombination mit <code>-dr</code> wird eine schreibgeschützte Referenzansicht angegeben.

Beispiele

Im folgenden Beispiel wird der Befehl `stcmd add-view` verwendet, um eine Ansicht mit Verzweigungen namens "Wartung 5.1" auf dem Computer namens "Orion" zu erstellen. (Auf Orion wird eine Instanz des StarTeam-Servers mit einer Serverkonfiguration ausgeführt, die Port 1024 verwendet.)

Mit dem folgenden Befehl wird die Ansicht als Unteransicht der vorhandenen StarDraw-Ansicht erstellt. Der StarDraw-Ordner ist der Stammordner der Ansicht. Die neue Ansicht basiert auf dem Label des letzten Builds des Produkts 5.1 vor der Auslieferung (Build 403). Der Arbeitsordner der Ansicht unterscheidet sich vom Arbeitsordner des übergeordneten Ordners. Für alle vorhandenen Elemente der Ansicht ist die Option "Verzweigen bei Änderung" aktiviert.

```
stcmd add-view -p "JMarsch:Passwort@Orion:1024/StarDraw/StarDraw/"
-cmp
-encrypt "RC4" -name "Wartung 5.1" -rp "C:\StarDraw\Wartung
5.1" -d "Wartungsansicht für die Version 5.1 unseres Produktes"
-dr -ba -cfdgl "Build 403"
```

Im folgenden Beispiel wird der Befehl `stcmd add-view` verwendet, um eine Ansicht mit Lese- und Schreibzugriff namens "Stamm ist Ordner 'Source Code'" auf dem Computer "Orion" zu erstellen. Mit diesem Befehl wird die Ansicht als Unteransicht der vorhandenen StarDraw-Ansicht erstellt. Sie verwendet den Ordner "Source Code" als Stammordner. Sie verfügt über denselben Arbeitsordner wie die übergeordnete Ansicht. Da die Konfiguration einer Referenzansicht mit Lese- und Schreibzugriff mit der Konfiguration der übergeordneten Ansicht übereinstimmen muss, können die Optionen `-cfdgl`, `-cfdgp` und `-cfdgd` nicht verwendet werden.

```
stcmd add-view -p "JMarsch:Passwort@Orion:1024/ StarDraw/StarDraw/Source Code" -cmp -
encrypt "RC4" -name "Stamm ist Ordner Source Code"
-d "StarDraw-Hauptansicht - aber mit dem Ordner Source Code als Stamm der Hierarchie" -dr
```

Verwandte Referenz

[Weitere Client-Befehlszeilenoperationen](#)

Labels übernehmen: stcmd apply-label

Verwenden Sie den Befehl `stcmd apply-label`, um Ansichts- oder Revisions-Labels für Dateirevisionen zu übernehmen. Die Labels müssen in StarTeam bereits vorhanden sein. Sie können die Labels in StarTeam oder mit dem Befehl `stcmd label` erstellen.

Syntax

Die Syntax für diesen Befehl lautet wie folgt:

```
stcmd apply-label -p "Projektspezifikation" [-pwdfile
>Dateipfad"] [-cmp] [-csf]
[-encrypt Verschlüsselungstyp]
[-is] [-q]
[-x] [-stop]
[-rp "Ordnerpfad"
| -fp "Ordnerpfad"]
[-filter "Dateistatus"]
[-vl "Label-Name"
| -vd "Datum"
| -vn Revisionsnummer]
-lbl "Label-Name"
[Dateien...]
```

Option	Beschreibung
<code>-filter</code>	Gibt einen String mit einem oder mehreren Zeichen an, von denen jedes einen Dateistatus darstellt. Dieser String darf keine Leerzeichen oder anderen Whitespace-Zeichen enthalten. Das Label wird nur auf die Dateien angewendet, die aktuell den angegebenen Dateistatus aufweisen. Das Label kann nicht auf Dateien mit dem Status "Nicht in Ansicht" angewendet werden. Zur Darstellung der einzelnen Statuswerte werden die folgenden Buchstaben verwendet: C für "Current" (Aktuell), M für "Modified" (Geändert), G für "Merge" (Zusammenführen), O für "Out of Date" (Veraltet), I für "Missing" (Fehlt) und U für "Unknown" (Unbekannt).
<code>-lbl</code>	Der Label-Name, der zu den angegebenen Revisionen hinzugefügt werden soll. Diese Option kann mehrfach verwendet werden. Die Anwendung hängt alle Label an die angegebene Datei bzw. an die angegebenen Revisionen an.
<code>-vd</code>	Das "Ab"-Datum und die Zeit, anhand derer die Revisionen identifiziert werden, die das neue Label erhalten. Folgende Beispiele sind u. a. möglich: "03.01.2005 12:41" "3. Januar 2005 12:41:21" "Montag, 3. Januar 2005 12:41"
<code>-vl</code>	Das Revisions- oder Ansichtslabel, anhand dessen die Revisionen identifiziert werden, die das neue Label erhalten. Dieses Label muss in der Anwendung bereits vorhanden sein. Ohne die Option <code>-vn</code> , <code>-vd</code> oder <code>-vl</code> erhält die aktuelle Revision der jeweiligen Datei das Label.
<code>-vn</code>	Die Revisionsnummer, anhand der die Revisionen identifiziert werden, die das neue Label erhalten.

Beispiele

Im folgenden Beispiel wird der Befehl `stcmd apply-label` verwendet, um das Label "Beta" für die Dateien des Ordners "User Manual" zu übernehmen. "User Manual" ist ein Unterordner des Stammordners StarDraw (in der Ansicht "StarDraw" des Projekts "StarDraw"). StarTeam übernimmt das Label für die Revisionen der Dateien, die um 12 Uhr am 7. Juli 2003 aktuell waren.

```
stcmd apply-label -p "JMarsch:Password@Orion:1024/StarDraw/StarDraw/User Manual" -vd  
"07/07/03 12:00" -lbl "Beta" "*" 
```

Verwandte Referenz

[Weitere Client-Befehlszeilenoperationen](#)

Dateien einchecken: stcmd ci

Verwenden Sie den Befehl `stcmd ci` zum Einchecken von Dateien aus einem Arbeitsordner in ein Repository (oder einen Dateispeicher) von StarTeam über die Befehlszeile.

Die neuen Dateirevisionen können gleichzeitig mit einem Prozesselement verknüpft werden. Alle durch diesen Befehl erfolgreich hinzugefügten Dateien werden mit der aktuellen Revision des Prozesselements verknüpft und an dieses angeheftet. Verwenden Sie die Option `-active`, um das gerade aktive Prozesselement anzugeben (das Sie zuvor mit einem Client auf Ihrer Workstation festgelegt haben).

Wenn kein Element aktiv ist oder Sie ein anderes Element verwenden möchten, nutzen Sie die Option, die den Typ des Prozesselements angibt (`-cr`, `-req` oder `-task`), gefolgt vom vollständigen Pfad des Stammordners aus der StarTeam-Projektansicht des Elements. Verwenden Sie dabei den Schrägstrich (`/`) als Trennzeichen zwischen Ordernamen. Geben Sie für Prozesselemente, die sich außerhalb der Ansicht befinden, den Projektnamen und den Ansichtsnamen vor dem vollständigen Ordnerpfad an. Verwenden Sie einen Doppelpunkt (`:`) als Trennzeichen für den Pfad der Ansicht. Mit `-cr MyProject/RootView:ChildView/SourceCode/37` geben Sie beispielsweise Änderungsanforderung 37 im Ordner `SourceCode` der Ansicht `ChildView` des Projekts `MyProject` an. Bei der Ausführung des Prozesses wird zuerst davon ausgegangen, dass sich das Prozesselement in der aktuellen Ansicht befindet. Daher wird geprüft, ob der vollständige Pfad einem Ordnerpfad in dieser Ansicht entspricht. Wird das Prozesselement in der aktuellen Ansicht nicht gefunden, wird es als Prozesselement behandelt, das sich außerhalb der Ansicht befindet, und die Suche nach dem Element beginnt auf Projekt- und Ansichtsebene.

Verwenden Sie die Option `-mark`, um das Prozesselement je nach Typ gleichzeitig als "repariert", "fertig gestellt" oder "abgeschlossen" zu markieren. Eine Änderungsanforderung kann beispielsweise als "repariert" markiert werden. Elemente werden so lange nicht als "repariert", "fertig gestellt" oder "abgeschlossen" markiert, bis alle Dateien erfolgreich hinzugefügt wurden.

Syntax

Die Syntax für diesen Befehl lautet wie folgt:

```
stcmd ci -p "Projektspezifikation" [-pwdfile "Dateipfad"] [-cmp] [-csf] [-encrypt  
Verschlüsselungstyp]  
[-is] [-q] [-x] [-stop] [-rp "Ordnerpfad" | -fp "Ordnerpfad"] [-filter "Dateistatus"] [-l |  
-u | -nel]  
[-ro | -rw]] [-vl "Label-Name"] [-nomove] [-f NCI] [-o | -i ] [-r "Kommentar" | -rf "  
Dateiname "] [-eol  
[on | off]] [[ -active | [-cr | -req | -task ] processItemPath] [-mark]] [Dateien...]
```

Option	Beschreibung
--------	--------------

<code>-active</code>	Das aktive Prozesselement.
----------------------	----------------------------

<code>-cr</code>	Vollständiger Pfad vom Stammordner der Projektansicht zu der Änderungsanforderungs-, Anforderungs- oder Task-Nummer, die als Prozesselement verwendet werden soll. Verwenden Sie den Schrägstrich (<code>/</code>) als Begrenzungszeichen zwischen Ordernamen.
------------------	--

<code>-req</code>	
<code>-task</code>	Geben Sie für Prozesselemente, die sich außerhalb der Ansicht befinden, den Projektnamen und den Ansichtsnamen vor dem vollständigen Ordnerpfad an. Verwenden Sie einen Doppelpunkt (<code>:</code>) als Trennzeichen für den Pfad der Ansicht. Mit <code>-cr MyProject/RootView:ChildView/SourceCode/37</code> geben Sie beispielsweise Änderungsanforderung 37 im Ordner <code>SourceCode</code> der Ansicht <code>ChildView</code> des Projekts <code>MyProject</code> an.

Bei der Ausführung des Prozesses wird zuerst davon ausgegangen, dass sich das Prozesselement in der aktuellen Ansicht befindet. Daher wird geprüft, ob der vollständige Pfad einem Ordnerpfad in dieser Ansicht entspricht. Wird das Prozesselement in der aktuellen Ansicht nicht gefunden, wird es als Prozesselement behandelt, das sich außerhalb der Ansicht befindet, und die Suche nach dem Element beginnt auf Projekt- und Ansichtsebene.

<code>-eol</code>	<p>Konvertiert automatisch EOL-Markierungen in Carriage Return/Zeilenvorschub-Zeichen (CR/LF), wenn Arbeitsdateien an das Repository des Servers übertragen werden. Wenn diese Option verwendet wird, wird die aktuelle EOL-Markierung der Arbeitsdatei in eine CR/LF-Kombination konvertiert. Wenn sie deaktiviert ist (Standard), wird keine Zeilenendekonvertierung ausgeführt. Unter Windows wird die EOL-Markierung "CR/LF" (Carriage Return/Line Feed) verwendet; auf UNIX-Plattformen wird nur "LF" verwendet.</p> <p>Diese Option wird verwendet, wenn Sie eine Arbeitsdatei von einem Computer, auf dem UNIX ausgeführt wird, in das Repository einchecken und die Textdateien im Repository mit CR/LF-Zeichen gespeichert werden sollen.</p>
<code>-f NCI</code>	<p>Legt das Einchecken aller Dateien mit dem Status "Geändert" fest. NCI ist das Akronym für "needs check-in" (muss eingeecheckt werden). Für das Einchecken werden keine anderen Dateitypen ausgewählt.</p> <p><code>-f NCI</code> wird ignoriert, wenn <code>-filter</code> verwendet wird.</p>
<code>-filter</code>	<p>Gibt einen String mit einem oder mehreren Zeichen an, von denen jedes einen Dateistatus darstellt. In diesem String dürfen keine Leerzeichen oder andere Whitespace-Zeichen verwendet werden. Es werden nur Dateien eingeecheckt, die aktuell den angegebenen Status aufweisen. Dateien mit dem Status "Nicht in Ansicht" können nicht eingeecheckt werden.</p> <p>Zur Darstellung der einzelnen Statuswerte werden die folgenden Buchstaben verwendet:</p> <p>C für "Current" (Aktuell), M für "Modified" (Geändert), G für "Merge" (Zusammenführen), O für "Out of Date" (Veraltet), I für "Missing" (Fehlt) und U für "Unknown" (Unbekannt).</p> <p><code>-filter</code> hat Vorrang vor <code>-f NCI</code>. Wenn Sie G, O oder U verwenden, müssen Sie auch <code>-i</code> oder <code>-o</code> angeben. Anderenfalls wird G, O oder U ignoriert.</p>
<code>-i</code>	<p>Ermöglicht das interaktive Einchecken von Dateien, deren Status dies normalerweise verhindern würde. Der Benutzer wird aufgefordert, das Einchecken zu bestätigen, wenn der Dateistatus "Zusammenführung", "Veraltet" oder "Unbekannt" ist. Mit Ihrer Antwort können Sie das Einchecken der Datei erzwingen.</p> <p>Wenn Sie die Option <code>-i</code> verwenden, können Sie die Option <code>-o</code> nicht verwenden.</p>
<code>-l</code>	<p>Sperrt jede Datei nach dem Einchecken. Ohne <code>-l</code>, <code>-u</code> oder <code>-nel</code> bleibt der Sperrstatus der Dateien unverändert.</p>
<code>-mark</code>	<p>Wenn alle Dateien erfolgreich hinzugefügt wurden, gibt diese Option an, dass der Status des Prozesselements in "Repariert" (bei einer Änderungsanforderung), "Fertig gestellt" (bei einer Task) oder "Abgeschlossen" (bei einer Anforderung) geändert wird. Die Dateien werden mit dem neuen Status an die Revision angeheftet.</p>
<code>-nel</code>	<p>Sperrt jede Datei nicht-exklusiv, nachdem diese eingeecheckt wurde.</p>
<code>-nomove</code>	<p>Stoppt das Anwenden des durch die Option <code>-vl</code> angegebenen Labels, wenn die einzucheckende Datei bereits über eine Revision mit diesem Label verfügt. Anderenfalls wird das Label von der aktuellen Label-Revision in die neu eingeecheckte Revision verschoben.</p>
<code>-o</code>	<p>Erzwingt das Einchecken von Dateien, deren Status dies normalerweise verhindern würde. Diese Option erzwingt das Einchecken aller Dateien, deren Dateistatus "Zusammenführung", "Veraltet" oder "Unbekannt" ist.</p> <p>Wenn Sie die Option <code>-o</code> verwenden, können Sie die Option <code>-i</code> nicht verwenden.</p>
<code>-r</code>	<p>Ein Revisionskommentar, in der Regel der Grund für das Einchecken der Dateien. Wenn Sie die Option <code>-r</code> verwenden, können Sie die Option <code>-rf</code> nicht verwenden.</p>
<code>-rf</code>	<p>Der Pfad zu der Datei, die den Revisionskommentar enthält.</p> <p>Wenn Sie die Option <code>-rf</code> verwenden, können Sie die Option <code>-r</code> nicht verwenden.</p>
<code>-ro</code>	<p>Definiert nach dieser Operation die Arbeitsdatei als schreibgeschützt. Ohne diese Option bliebe der Status der Datei derselbe wie vor der Operation.</p> <p>In der Regel verwenden Sie <code>-ro</code>, um zu verhindern, dass Sie selbst eine Datei bearbeiten, die Sie nicht gesperrt haben.</p> <p><code>-ro</code> muss mit <code>-l</code>, <code>-u</code> oder <code>-nel</code> verwendet werden. Wenn Sie <code>-ro</code> verwenden, ist die Verwendung von <code>-rw</code> nicht möglich.</p>
<code>-rw</code>	<p>Gibt die Arbeitsdatei nach dieser Operation zum Lesen und Schreiben frei. Ohne diese Option bliebe der Status der Datei derselbe wie vor der Operation.</p>

`-rw` muss mit `-l`, `-u` oder `-nel` verwendet werden. Wenn Sie `-rw` verwenden, ist die Verwendung von `-ro` nicht möglich.

`-u` Entsperrt die neu eingeecheckten Dateien.

`-vl` Das mit dem Befehl `stcmd label` erstellte Label, das auf die eingeecheckten Dateien angewendet werden soll. Das Label wird in doppelte Anführungszeichen gesetzt. Diese Option kann mehrmals im Befehl verwendet werden. Das Label kann entweder ein Ansichts- oder ein Revisions-Label sein, es muss aber bereits in der Anwendung vorhanden sein.

Beispiel

Im folgenden Beispiel wird der Befehl `stcmd ci` verwendet, um `.bmp`-Dateien in den Ordner "Online-Hilfe" einzuchecken. "Online-Hilfe" ist ein Unterordner des Stammordners "StarDraw" (in der Ansicht "StarDraw" des Projekts "StarDraw"). Durch den Befehl werden die Dateien entsperrt, die Arbeitskopien werden schreibgeschützt und die Dateien erhalten einen Revisionskommentar (in der Regel der Grund für das Einchecken der Dateien).

```
stcmd ci -p "JMarsch:Passwort@Orion:1024/StarDraw/StarDraw/Source Code/Online-Hilfe" -u -ro  
-r "für Betaversion überarbeitet" "*.bmp"
```

Verwandte Referenz

[Weitere Client-Befehlszeilenoperationen](#)

Dateien auschecken: stcmd co

Verwenden Sie den Befehl `stcmd co`, um Dateien aus einem StarTeam-Repository (oder einem Datenspeicher) von der Befehlszeile aus in Ihren Arbeitsordner auszuchecken. Wenn die Option `-o` nicht verwendet wird, hält das Programm bei jeder Datei an, die den Status "Geändert", "Zusammenführung" oder "Unbekannt" hat, damit Sie wissen, dass die jeweilige Datei nicht ausgecheckt wird.

Mit der Option `-merge` können Dateien beim Auschecken zusammengeführt werden. Beim Einchecken werden Dateien nicht zusammengeführt.

Syntax

Die Syntax für diesen Befehl lautet wie folgt:

```
stcmd co -p "Projekt" [-pwdfile "Dateiname"] [-epwdfile "Dateiname"] [-cmp] [-encrypt RC4,
RC2_ECB, RC2_CBC, RC2_CFB]
[-cagl "Label" | -cagp "Heraufstufungsstatus" | -cagd "Datum"] [-is] [-csf] [-q] [-x] [-
stop]
[-rp "Verzeichnis" | -fp "Verzeichnis"] [-filter "Filter"][-?] [-h] [nologo] [-o | -i | -
merge] [-hook "ausführbare Datei"]
[-l | -u | -nel] [-ro | -rw]] [-vl "Name" | -vd "Datum" | -vn Nummer]
[-f NCO] [-ts] [-eol [on | off | cr| lf| crlf ]] [-fs] [-dryrun | -alwaysprompt | -
neverprompt | -conflictprompt] [-mpxCacheAgent Nummer]
[-useMPXCacheAgent "Host"Port" | autolocate]] [Dateien...]
```

Option	Beschreibung
<code>-cagd</code>	Konfiguriert die Ansicht mit den angegebenen Datums-/Uhrzeitangaben. Folgende Beispiele sind u. a. möglich: "03.01.05 11:25" "3. Januar 2005 11:25:00 CET" "Montag, 3. Januar 2005 11:25 Uhr CET"
<code>-cagl</code>	Konfiguriert die Ansicht mit dem angegebenen Label. Ohne <code>-cagl</code> , <code>-cagp</code> oder <code>-cagd</code> wird die aktuelle Konfiguration der Ansicht verwendet.
<code>-cagp</code>	Konfiguriert die Ansicht mit dem angegebenen Heraufstufungsstatus.
<code>-eol</code>	Führt die Zeilenendekodierung von Textdateien durch. Wenn "on", werden Textdateien aus dem Repository des Servers an den Arbeitsordner der Workstation übertragen, wobei die Zeilenendekodierung der Plattform verwendet wird, die den Befehl ausführt, wie von der JVM festgelegt. Wenn "off" (Standard), wird keine Zeilenendekodierung ausgeführt. Die Verwendung von "off" hat dieselbe Auswirkung wie das Nichtverwenden von <code>-eol</code> . Wenn Sie das Zeilenendezeichen (cr, lf oder crlf) angeben, werden die Textdateien aus dem Repository des Servers an den Arbeitsordner der Workstation übertragen, wobei die angegebene Zeilenendekodierung verwendet wird. Auf Windows-Clients wird die EOL-Markierung "Carriage Return / Line Feed" (crlf) verwendet; auf UNIX-Plattformen "Line Feed" (lf) und auf MAC-Systemen das "Carriage Return" (cr). Sie aktivieren diese Option beispielsweise, wenn Sie eine Datei aus dem Repository mit einer Arbeitsdatei auf einem UNIX-System vergleichen möchten (wenn das Repository Textdateien mit CRLF speichert).
<code>-f NCO</code>	Legt das Auschecken aller Dateien mit dem Status "Fehlt" oder "Veraltet" fest. NCO ist das Akronym für "needs check-out" (muss ausgecheckt werden). Für das Auschecken werden keine anderen Dateien ausgewählt. <code>-f NCO</code> wird ignoriert, wenn <code>-filter</code> verwendet wird.

-
- `-filter` Gibt einen String mit einem oder mehreren Zeichen an, von denen jedes einen Dateistatus darstellt. Dieser String darf keine Leerzeichen oder anderen Whitespace-Zeichen enthalten. Es werden nur Dateien mit dem angegebenen Status ausgecheckt. Dateien des Typs Nicht in Ansicht können nicht ausgecheckt werden. Zur Darstellung der einzelnen Statuswerte werden die folgenden Buchstaben verwendet: C für "Current" (Aktuell), M für "Modified" (Geändert), G für "Merge" (Zusammenführen), O für "Out of Date" (Veraltet), I für "Missing" (Fehlt) und U für "Unknown" (Unbekannt).
- `-filter` hat Vorrang vor `-f NCO`. Wenn Sie G, M, O oder U verwenden, müssen Sie auch `-merge` oder `-o` angeben, um das Auschecken zu erzwingen. Anderenfalls wird G, M, O oder U ignoriert.
-
- `-fs` Verhindert, dass der Dateistatus nach dem Auschecken beibehalten wird. Nachfolgende Statuswerte für diese Dateien sind falsch und unklar. Verwenden Sie diese Option, wenn der Status einer Datei irrelevant ist. Wenn Sie die Arbeitsordner beispielsweise routinemäßig vor dem Auschecken der Dateien für einen Build löschen, sind keine Dateien vorhanden und ihr Status spielt demnach keine Rolle.
- Bedenken Sie, dass dann der Status der Dateien niemals bekannt ist, selbst dann nicht, wenn Sie zu einem späteren Zeitpunkt den Befehl zum Aktualisieren des Status verwenden. Um die aktuellen Dateien mit ihren richtigen Statuswerten zu erhalten, können Sie das Auschecken ohne die Option `-fs` erzwingen.
-
- `-hook` Wird nur in Verbindung mit `-merge` verwendet. Wenn Sie diese Option verwenden, können Sie eine alternative Anwendung (ein anderes als das mit der Anwendung verfügbare Programm) für das Zusammenführen angeben. Der Wert der Option sollte der Name eines Programms sein, mit dem das Zusammenführen erfolgt. Beispiel: `-hook mymerge.sh`
- Die Anwendung für das Zusammenführen muss einen Exit-Code von 0 zurückgeben, was bedeutet, dass keine Konflikte erkannt wurden. Ein Exit-Code von 1 bedeutet, dass Konflikte erkannt wurden. Andere Werte stehen für einen Fehler. Die Anwendung für das Zusammenführen darf keine Batch-Datei auf Windows-Plattformen sein, da, wenn sie über Java ausgeführt wird, der Ergebnis-Code nicht ordnungsgemäß zurückgegeben wird.
- Für jede zusammenzuführende Datei stellt der Befehl "stcmd" der Anwendung für das Zusammenführen drei Argumente zur Verfügung. Die Argumente sind die voll qualifizierten Pfade der folgenden drei Dateien (in dieser Reihenfolge):
- localFile: Die lokale Arbeitsdatei
- commonFile: Der Inhalt der Dateirevision im Repository, bei der es sich um die letzte allgemeine Version zwischen der lokal geänderten Datei und der angegebenen Revision im Repository handelt.
- otherFile: Der Inhalt der auszucheckenden Dateirevision
- Das Programm "merge hook" muss die Zusammenführungsergebnisse an die Standardausgabe (stdout) senden.
-
- `-i` Ermöglicht das interaktive Auschecken von Dateien, deren Status dies normalerweise verhindern würde. Fordert den Benutzer auf, das Auschecken zu bestätigen, wenn der Dateistatus "Geändert", "Zusammenführung" oder "Unbekannt" ist. Mit Ihrer Antwort können Sie das Auschecken der Datei erzwingen. Wenn Sie die Option `-i` verwenden, können Sie nicht die Option `-o` oder `-merge` verwenden.
-
- `-l` Sperrt jede Datei nach dem Auschecken. Ohne `-l`, `-u` oder `-nel` bleibt der Sperrstatus der Dateien unverändert.
-
- `-merge` Mit dieser Option können Sie die Arbeitsdatei mit der auszucheckenden Revision und mit der Revision zusammenführen, auf der beide Dateien basieren. Die Arbeitsdatei muss den Status "Zusammenführung" haben. Sie können mit `-merge` eine der folgenden Berichtsoptionen angeben:
- `-dryrun`: Gibt an, ob die Zusammenführungsergebnisdatei Konflikte enthält; die lokale Arbeitsdatei ist unverändert, da nichts ausgecheckt wurde. Diese Option erstellt eine Vorschau.
- `-alwaysprompt`: Fordert den Benutzer immer auf, die Datei mit den Zusammenführungsergebnissen im Arbeitsordner zu speichern, unabhängig davon, ob es Konflikte beim Zusammenführen gab oder nicht.
- `-neverprompt`: Speichert die Datei mit den Zusammenführungsergebnissen immer im Arbeitsordner.
- `-conflictprompt`: Fordert den Benutzer nur dann zum Speichern der Datei mit den Zusammenführungsergebnissen auf, wenn Konflikte erkannt wurden.
- Die Optionen `-dryrun`, `-alwaysprompt`, `-neverprompt`, und `-conflictprompt` schließen sich gegenseitig aus. Wenn keine Option angegeben ist, wird standardmäßig `-conflictprompt` verwendet.

Wenn Sie diese Option verwenden, können Sie eine alternative Anwendung (ein anderes als das mit der Anwendung verfügbare Programm) für das Zusammenführen angeben.

Wenn Sie die Option `-merge` verwenden, können Sie nicht die Option `-i` oder `-o` verwenden.

Wenn Sie `-hook` nicht verwenden und eine zusammengeführte Datei mit Konflikten speichern, wird jeder Konflikt in der Arbeitsdatei wie folgt markiert:

```
<<<<<<< Dateiname (lokal)
```

```
Zeile, wie sie in der lokalen Datei angezeigt wird
```

```
=====
```

```
Zeile, wie sie in der auszucheckenden Revision angezeigt wird
```

```
>>>>>>> Dateiname (Version x), wobei "Dateiname" der Name der Datei und "x" die Nummer der auszucheckenden Revision ist.
```

Wenn Sie `-merge` nicht angeben, werden Dateien mit dem Status "Zusammenführung" ignoriert, es sei denn, Sie geben `-o` an und erzwingen das Auschecken.

<code>-nel</code>	Sperrt die Datei nicht-exklusiv, nachdem diese ausgecheckt wurde.
<code>-o</code>	Erzwingt das Auschecken von Dateien mit beliebigem Status. Wenn Sie die Option <code>-o</code> verwenden, können Sie die Option <code>-i</code> oder <code>-merge</code> nicht verwenden.
<code>-ro</code>	Definiert nach dieser Operation die Arbeitsdatei als schreibgeschützt. Ohne diese Option bliebe der Status der Datei derselbe wie vor der Operation. In der Regel verwenden Sie <code>-ro</code> , um zu verhindern, dass Sie selbst eine Datei bearbeiten, die Sie nicht gesperrt haben. <code>-ro</code> muss mit <code>-l</code> , <code>-u</code> oder <code>-nel</code> verwendet werden. Wenn Sie <code>-ro</code> verwenden, ist die Verwendung von <code>-rw</code> nicht möglich.
<code>-rw</code>	Gibt die Arbeitsdatei nach dieser Operation zum Lesen und Schreiben frei. Ohne diese Option bliebe der Status der Datei derselbe wie vor der Operation. <code>-rw</code> muss mit <code>-l</code> , <code>-u</code> oder <code>-nel</code> verwendet werden. Wenn Sie <code>-rw</code> verwenden, ist die Verwendung von <code>-ro</code> nicht möglich.
<code>-ts</code>	Legt den Zeitstempel der einzelnen Arbeitsdateien auf die Auscheckzeit fest. Ohne diese Option erhält die Datei denselben Zeitstempel wie die eingecheckte Revision der Datei.
<code>-u</code>	Entsperrt die ausgecheckten Dateien.
<code>-vd</code>	Das "Ab"-Datum und die Zeit, anhand derer die Revisionen identifiziert werden, die ausgecheckt werden sollen. Die letzte Revision vor den angegebenen Datums- und Zeitwerten ist diejenige, die für eine Datei ausgecheckt wird. Weitere Informationen hierzu finden Sie in den Datum/Zeit-Beispielen für <code>-cfgd</code> weiter oben.
<code>-vl</code>	Das Revisions- oder Ansichts-Label, anhand dessen die Revisionen identifiziert werden, die ausgecheckt werden sollen. Ohne die Option <code>-vn</code> , <code>-vd</code> oder <code>-vl</code> wird die aktuelle Revision der jeweiligen Datei ausgecheckt.
<code>-vn</code>	Die Revisionsnummer der auszucheckenden Dateien.

Beispiele

Im folgenden Beispiel wird der Befehl `stcmd co` verwendet, um `.doc`-Dateien aus dem Ordner "User Manual" zu sperren und auszuchecken. "User Manual" ist ein Unterordner des Stammordners "StarDraw" (in der Ansicht "StarDraw" des Projektes "StarDraw").

```
stcmd co -p "JMarsch:Passwort@Orion:1024/StarDraw/StarDraw/User Manual" -l "*.doc"
```

Im nächsten Beispiel wird der Befehl "stcmd co" verwendet, um Revisionen einer Readme-Datei zusammenzuführen.

```
stcmd co -p "NTesla:@10.50.5.179:49201/WebDev/WebDev"  
encrypt RC4 -fp "/export/home0/johnson/working" -merge "README"
```

Verwandte Referenz

[Weitere Client-Befehlszeilenoperationen](#)

Dateirevisionen vergleichen: stcmd diff

Verwenden Sie den Befehl `stcmd diff`, um die Unterschiede zwischen zwei Dateirevisionen anzuzeigen. Der Befehl kann auf mehrere Dateien angewendet werden. Falls Sie keine bestimmten Revisionen (mit `-vn`, `-vd` oder `-vl`) angeben, werden die jeweiligen Arbeitskopien der Dateien mit den aktuellen Dateirevisionen im Repository (oder Datenspeicher) verglichen. Wenn Sie eine bestimmte Revision angeben, werden die jeweiligen Arbeitskopien der angegebenen Dateien mit dieser Revision verglichen. Wenn Sie zwei Revisionen angeben, werden diese beiden Revisionen aller angegebenen Dateien jeweils miteinander verglichen.

Die Unterschiede beim Vergleichen von Textdateien können angezeigt werden. Bei Vergleichen von Binärdateien zeigen die Ausgabeergebnisse an, ob die Revisionen der Datei identisch oder unterschiedlich sind.

Syntax

Die Syntax für diesen Befehl lautet wie folgt:

```
stcmd diff -p "Projektspezifikation" [-pwdfile "Dateipfad"] [-cmp] [-csf]
[-encrypt Verschlüsselungstyp] [-cagl "Label" | -cagp "Status" | -cagd "Datum"]
[-is] [-q] [-x] [-stop] [-rp "Ordnerpfad" | -fp "Ordnerpfad"] [-filter "Dateistatus"]
[-eol [on | off | cr | lf | crlf]] [-w | -Bpvcs | -b] [-i] [-m "Maske"] [-t Nummer] [-c Nummer]
[-n]
[-nd] [-e] [-vl "Label" | -vd "Datum" | -vn Revisionsnummer] [Dateien...]
```

Option	Beschreibung
<code>-b</code>	Beim Vergleichen von zwei Textdateizeilen werden abschließende Whitespace-Zeichen ignoriert und alle anderen Strings mit Whitespace-Zeichen als gleich lang behandelt. Folgende Zeilen werden beispielsweise als äquivalent angesehen: " hi mom " " hi mom"
<code>-Bpvcs</code>	Ignoriert führende und abschließende Whitespace-Zeichen beim Vergleich von zwei Textdateizeilen. Die folgenden Zeilen sind beispielsweise äquivalent, weil es nur ein Leerzeichen zwischen "hi" und "mom" gibt: " hi mom " " hi mom" die nächste Zeile ist jedoch nicht äquivalent: "hi mom"
<code>-c</code>	Die Anzahl der unveränderten Zeilen, die vor und nach einem gefundenen Unterschied in Textdateien angezeigt werden sollen. Ohne diese Option werden alle Zeilen der Dateien angezeigt. Die Option <code>-c 2</code> fügt beispielsweise zwei unveränderte Zeilen vor und nach jeder Zeile oder nach Zeilen ein, die geändert wurden.
<code>-cagd</code>	Konfiguriert die Ansicht mit den angegebenen Datums-/Uhrzeitangaben. Folgende Beispiele sind u. a. möglich: "03.01.05 11:25" "3. Januar 2005 11:25:00 CET" "Montag, 3. Januar 2005 11:25 Uhr CET"
<code>-cagl</code>	Konfiguriert die Ansicht mit dem angegebenen Label. Ohne <code>-cagl</code> , <code>-cagp</code> oder <code>-cagd</code> wird die aktuelle Konfiguration der Ansicht verwendet.
<code>-cagp</code>	Konfiguriert die Ansicht mit dem angegebenen Heraufstufungsstatus.
<code>-e</code>	Führt dazu, dass der Befehl die folgenden Exit-Codes zurückgibt: 0, wenn alle verglichenen Dateien gleich sind

- 1, wenn eine Fehlerbedingung aufgetreten ist
- 2, wenn mindestens eine Datei unterschiedlich ist

Verwenden Sie `-e` sowohl mit Text- als auch mit Binärdateien.

<code>-eol</code>	Dieser Befehl ist derzeit irrelevant, da <code>diff</code> EOL-Markierungen ignoriert. Wenn sich zwei Zeilen nur darin unterscheiden, gelten sie als identisch.
<code>-filter</code>	<p>Gibt einen String mit einem oder mehreren Zeichen an, von denen jedes einen Dateistatus darstellt. In diesem String dürfen keine Leerzeichen oder andere Whitespace-Zeichen verwendet werden. Es werden nur Dateien verglichen, die aktuell den angegebenen Status aufweisen. Revisionen von Dateien mit dem Status "Nicht in Ansicht" können nicht verglichen werden.</p> <p>Zur Darstellung der einzelnen Statuswerte werden die folgenden Buchstaben verwendet: <code>C</code> für "Current" (Aktuell), <code>M</code> für "Modified" (Geändert), <code>G</code> für "Merge" (Zusammenführen), <code>O</code> für "Out of Date" (Veraltet), <code>I</code> für "Missing" (Fehlt) und <code>U</code> für "Unknown" (Unbekannt).</p>
<code>-i</code>	Ignoriert die Groß-/Kleinschreibung beim Vergleich zweier Textdateien. Beispiel: "A" entspricht "a".
<code>-m</code>	<p>Ignoriert beim Vergleichen zweier Textdateien die Zeichen in bestimmten Spalten, wie durch eine oder mehrere Masken angegeben. Masken haben die folgende Syntax:</p> <pre>" Spaltennummer-Spaltennummer[(numeric)] "</pre> <p>Beispiel: "1-6" ignoriert die Zeichen in den ersten sechs Spalten jeder Zeile und "1-6(numeric)" ignoriert die ersten sechs Spalten jeder Zeile, wenn das Zeichen in Spalte 1 in beiden Dateien eine Zahl ist.</p> <p>Sie können mehrere Masken verwenden, diese müssen aber durch Kommas getrennt werden. Die Syntax lautet:</p> <pre>" Maske[,Maske]..."</pre>
<code>-n</code>	Unterdrückt die Anzeige der Zeilennummern in den beiden Textdateien.
<code>-nd</code>	Unterdrückt die Anzeige der Unterschiede in den beiden Textdateien. Bei Vergleichen von Binärdateien werden keine Unterschiede angezeigt. Diese Option ist nützlich in Kombination mit der Option <code>-e</code> .
<code>-t</code>	Die Anzahl der Leerzeichen für die einzelnen Tabstopps bei der Anzeige der Unterschiede von Textdateien. Die Vorgabe ist vier. Verwenden Sie <code>-t0</code> , um die Tabulatorkonvertierung zu unterdrücken.
<code>-vd</code>	Das Datum und die Zeit, anhand derer die Revisionen identifiziert werden, die verglichen werden sollen. Es wird die letzte Revision vor den angegebenen Datums- und Zeitwerten verwendet. Weitere Informationen hierzu finden Sie in den Datum/Zeit-Beispielen für <code>-cfgd</code> weiter oben.
<code>-vl</code>	<p>Das Revisions- oder Ansichts-Label, anhand dessen die Revisionen identifiziert werden, die verglichen werden sollen. Sie können eine beliebige Kombination aus keiner, einer oder zwei der Optionen <code>-vn</code>, <code>-vd</code> oder <code>-vl</code> verwenden.</p> <p>Verwenden Sie keine Option, um die Arbeitsdatei mit der aktuellen Revision, eine Option, um die Arbeitsdatei mit der angegebenen Revision, und zwei Optionen, um zwei Revisionen zu vergleichen.</p>
<code>-vn</code>	Die Nummer der Revisionen, die verglichen werden sollen.
<code>-w</code>	<p>Ignoriert Whitespace-Zeichen (Tabulatoren und Leerzeichen) beim Vergleich zweier Zeilen in Textdateien. Folgende Zeilen werden beispielsweise als äquivalent angesehen:</p> <pre>" a = (b + 2);" "a=(b+2) ;"</pre> <p>Die Optionen <code>-w</code>, <code>-BpvcS</code> und <code>-b</code> schließen sich gegenseitig aus.</p>

Beispiel

Im folgenden Beispiel wird der Befehl `stcmd diff` verwendet, um die Revisionen "Beta1" und "Beta2" der `.cpp`-Dateien des Ordners `SourceCode`, eines Unterordners des Stammordners `StarDraw` (in der Ansicht "StarDraw" des Projekts "StarDraw"), zu vergleichen. Leer- und Tabulatorzeichen werden ignoriert.

```
stcmd diff -p "JMarsch:Passwort@Orion:1024/StarDraw/StarDraw/Quelltext"  
-w -vl "Beta1" -vl "Beta2" "*.cpp"
```

Verwandte Referenz

[Weitere Client-Befehlszeilenoperationen](#)

Dateibeschreibungen ändern: stcmd dsc

Mit `stcmd dsc` kann eine Dateibeschreibung über die Befehlszeile geändert werden. Durch diesen Befehl wird eine neue Dateirevision erstellt, die die neue Beschreibung als eine Eigenschaft enthält.

Syntax

Die Syntax für diesen Befehl lautet wie folgt:

```
stcmd dsc -p "Projektspezifikation" [-pwdfile
  "Dateipfad"] [-cmp] [-csf]
  [-encrypt Verschlüsselungstyp]
  [-is] [-q]
  [-x] [-stop]
  [-rp "Dateipfad"
  | -fp "Ordnerpfad"]
  [-filter "Dateistatus"]
  -d "Beschreibung"
  [Dateien...]
```

Option	Beschreibung
<code>-d</code>	Die Beschreibung für die Dateien. Die Beschreibung wird in doppelte Anführungszeichen gesetzt.
<code>-filter</code>	Gibt einen String mit einem oder mehreren Zeichen an, von denen jedes einen Dateistatus darstellt. Dieser String darf keine Leerzeichen oder anderen Whitespace-Zeichen enthalten. Es werden nur Dateien mit dem angegebenen Status beschrieben. Beschreibungen von Dateien mit dem Status "Nicht in Ansicht" können nicht geändert werden. Zur Darstellung der einzelnen Statuswerte werden die folgenden Buchstaben verwendet: C für "Current" (Aktuell), M für "Modified" (Geändert), G für "Merge" (Zusammenführen), O für "Out of Date" (Veraltet), I für "Missing" (Fehlt) und U für "Unknown" (Unbekannt).

Beispiel

Im folgenden Beispiel wird der Befehl `stcmd dsc` verwendet, um die Beschreibung der Datei "stdafx.cpp" im Ordner "User Manual" zu ändern. "User Manual" ist ein Unterordner des Stammordners "StarDraw" (in der Ansicht "StarDraw" des Projekts "StarDraw").

```
stcmd dsc -p "JMarsch:Passwort@Orion:1024/StarDraw/StarDraw/Source Code"
-d "Source Code für StarTeam" "stdafx.cpp"
```

Verwandte Referenz

[Weitere Client-Befehlszeilenoperationen](#)

Labels erstellen: stcmd label

Verwenden Sie den Befehl `stcmd label`, um Ansichts- oder Revisions-Labels zu erstellen. Ein Ansichts-Label kann als Build-Label vorgesehen werden. Standardmäßig werden Ansichts-Labels automatisch auf alle Ordner, Dateien, Änderungsanforderungen, Anforderungen, Themen und Tasks in der Ansicht angewendet. Revisions-Labels werden standardmäßig nicht auf Elemente angewendet.

Übernehmen Sie mit dem Befehl `stcmd apply-label` Labels für bestimmte Dateien, die mit `stcmd label` erstellt wurden. Sie können auch die Label-Option (`-v1`) in `stcmd ci` verwenden, um Dateien beim Einchecken neu erstellte Labels zuzuweisen.

Option	Beschreibung
<code>-b</code>	Bedeutet, dass das neue Label ein Build-Label ist. Ohne <code>-b</code> oder <code>-r</code> ist das Label ein Ansichts-Label. Ansichts-Labels (und ein Build-Label ist ein spezieller Typ eines Ansichts-Labels) werden sofort und automatisch allen Ordnern, Dateien, Änderungsanforderungen, Tasks und Themen in der Ansicht zugewiesen.
<code>-d</code>	Die Beschreibung für das Label.
<code>-f</code>	Erstellt das neue Label als eingefrorenes Label.
<code>-nl</code>	Der Name des neuen Labels.
<code>-r</code>	Bedeutet, dass das neue Label ein Revisions-Label ist. Sie können das neue Label verwenden, um die einzucheckenden Dateien zu kennzeichnen. Dieser Befehl hängt das neue Label nicht an Elemente an, es sei denn, sie erstellen das Label, indem Sie ein vorhandenes Revisions-Label kopieren, das an ein oder mehrere Elemente angehängt ist. Weitere Informationen hierzu finden Sie in der Beschreibung der Option <code>-v1</code> .
<code>-vd</code>	Gibt beim Erstellen eines Ansichts-Labels das Datum und die Zeit für das Label an. Beim Erstellen eines Revisions-Labels wird <code>-vd</code> ignoriert. Ohne die Option <code>-vn</code> , <code>-vd</code> oder <code>-v1</code> wird für Ansichts-Labels die aktuelle Zeit verwendet. Folgende Beispiele sind u. a. möglich: "03.01.2005 12:41" "3. Januar 2005 12:41:21" "Montag, 3. Januar 2005 12:41"
<code>-v1</code>	Der Name des zu kopierenden Labels. Der Name des Labels wird in doppelte Anführungszeichen gesetzt. Das angegebene Label muss in der Anwendung bereits vorhanden und vom selben Typ wie das zu erstellende Label sein. Wenn das angegebene Label ein an ein oder mehrere Elemente angehängtes Revisions-Label ist, wird das neue Label ebenfalls an diese Elemente angehängt. Ohne die Option <code>-vn</code> , <code>-vd</code> oder <code>-v1</code> wird für Ansichts-Labels die aktuelle Zeit verwendet. Wenn es sich um ein Revisions-Label handelt, stellen Sie sicher, dass die Option <code>-r</code> angegeben ist.
<code>-vp</code>	Ein Heraufstufungsstatus, dessen Label kopiert werden soll. Der Statusname wird in doppelte Anführungszeichen gesetzt. Das angegebene Label muss in der Anwendung bereits vorhanden sein und Sie müssen ein Ansichts-Label erstellen, damit diese Option gültig ist. Ohne die Option <code>-vn</code> , <code>-vd</code> oder <code>-v1</code> wird für Ansichts-Labels die aktuelle Zeit verwendet. Das Erstellen eines Ansichts-Labels basierend auf einem Heraufstufungsstatus ist nur dann möglich, wenn der Heraufstufungsstatus auf ein Label und nicht auf Aktuell gesetzt ist.

Syntax

Die Syntax für diesen Befehl lautet wie folgt:

```
stcmd label -p "Benutzername:Passwort@Host:Port/Projekt/Ansicht/Ordner" [-pwdfile  
"Dateipfad"] [-cmp] [-encrypt Verschlüsselungstyp]
```

```
[-q] [-x] [-stop] -nl "Label-Name" [-vl "Label-Name" | -vd "Datum" | -vp Status]
[-d "Beschreibung"] [-b | -r] [-f]
```

Beispiel

Im folgenden Beispiel wird der Befehl "stcmd label" verwendet, um ein neues Build-Label mit dem Namen "Beta" für die StarDraw-Ansicht des StarDraw-Projektes zu erstellen.

```
stcmd label -p "JMarsch:Passwort@Orion:1024/StarDraw/StarDraw"
-nl "Beta" -b
```

Verwandte Referenz

[Weitere Client-Befehlszeilenoperationen](#)

Arbeitsordner erstellen: stcmd local-mkdir

Verwenden Sie den Befehl `stcmd local-mkdir`, um auf Ihrer Workstation den Arbeitsordner oder das Arbeitsverzeichnis des angegebenen StarTeam-Ordners zu erstellen. Mit "-is" werden die Arbeitsordner (oder Arbeitsverzeichnisse) auch für die Unterordner des angegebenen StarTeam-Ordners erstellt.

Syntax

Die Syntax für diesen Befehl lautet wie folgt:

```
stcmd local-mkdir -p "Projektspezifikation" [-pwdfile
>Dateipfad"] [-cmp] [-csf]
[-encrypt Verschlüsselungstyp]
[-cfigl "Label-Name"
| -cfigp "Status"
| -cfigd "Datum"]
[-is] [-q]
[-x] [-stop]
[-rp "Ordnerpfad"
| -fp "Ordnerpfad"]
```

Option	Beschreibung
-cfigd	Konfiguriert die Ansicht mit den angegebenen Datums-/Uhrzeitangaben. Folgende Beispiele sind u. a. möglich: "03.01.05 11:25" "3. Januar 2005 11:25:00 CET" "Montag, 3. Januar 2005 11:25 Uhr CET"
-cfigl	Konfiguriert die Ansicht mit dem angegebenen Label. Ohne -cfigl, -cfigp oder -cfigd wird die aktuelle Konfiguration der Ansicht verwendet.
-cfigp	Konfiguriert die Ansicht mit dem angegebenen Heraufstufungsstatus.

Beispiel

Im folgenden Beispiel wird der Befehl `stcmd local-mkdir` verwendet, um die Arbeitsordner für den Ordner "Source Code" und seine Unterordner zu erstellen. "Source Code" ist ein Unterordner des Stammordners "StarDraw" (in der Ansicht "StarDraw" des Projekts "StarDraw").

```
stcmd local-mkdir -p "JMarsch:Passwort@Orion:1024/ StarDraw/StarDraw/Source Code"
-is
```

Verwandte Referenz

[Weitere Client-Befehlszeilenoperationen](#)

Lokale Dateien löschen: stcmd delete-local

Verwenden Sie den Befehl `stcmd delete-local`, um Dateien aus einem Arbeitsordner zu löschen. Sie können Dateien löschen, die der Versionskontrolle unterliegen oder sich nicht in StarTeam befinden. Durch diese Aktion werden keine Dateien aus der Versionskontrolle entfernt. Es wird lediglich die Datenmenge, die sich im Arbeitsordner auf Ihrer Workstation befindet, reduziert. Wenn Sie Dateien basierend auf ihrem StarTeam-Status löschen möchten, sollten Sie zunächst eine Statusaktualisierung mit `stcmd update-status` durchführen.

Syntax

Die Syntax für diesen Befehl lautet wie folgt:

```
stcmd delete-local -p
  "Projektspezifikation"
  [-pwdfile "Dateipfad"]
  [-cmp] [-csf]
  [-encrypt Verschlüsselungstyp]
  [-is] [-q]
  [-x] [-stop]
  [-rp "Ordnerpfad"
  | -fp "Ordnerpfad"]
  [-cfigl "Label-Name"
  | -cfgp "Status"
  | -cfgd "Datum"]
  [-filter "Dateistatus"]
  [Dateien...]
```

Option	Beschreibung
<code>-cfgd</code>	Konfiguriert die Ansicht mit den angegebenen Datums-/Uhrzeitangaben. Folgende Beispiele sind u. a. möglich: "03.01.05 11:25" "3. Januar 2005 11:25:00 CET" "Montag, 3. Januar 2005 11:25 Uhr CET"
<code>-cfigl</code>	Konfiguriert die Ansicht mit dem angegebenen Label. Ohne <code>-cfigl</code> , <code>-cfgp</code> oder <code>-cfgd</code> wird die aktuelle Konfiguration der Ansicht verwendet.
<code>-cfgp</code>	Konfiguriert die Ansicht mit dem angegebenen Heraufstufungsstatus.
<code>-filter</code>	Gibt einen String mit einem oder mehreren Zeichen an, von denen jedes einen Dateistatus darstellt. Dieser String darf keine Leerzeichen oder anderen Whitespace-Zeichen enthalten. Es werden nur Dateien gelöscht, die aktuell den angegebenen Status aufweisen. Zur Darstellung der einzelnen Statuswerte werden die folgenden Buchstaben verwendet: C für "Current" (Aktuell), M für "Modified" (Geändert), G für "Merge" (Zusammenführen), O für "Out of Date" (Veraltet), I für "Missing" (Fehlt) und U für "Unknown" (Unbekannt).

Beispiel

Im folgenden Beispiel wird der Befehl `stcmd delete-local` verwendet, um einige Dateien aus dem Arbeitsordner des StarTeam-Ordners `SourceCode` zu löschen. `Source Code` ist ein Unterordner des Stammordners `StarDraw` (in der StarDraw-Ansicht des StarDraw-Projekts). In diesem Beispiel werden alle Dateien gelöscht, die nicht der Versionskontrolle unterliegen. Diese Dateien haben den Status **Nicht in Ansicht**.

```
stcmd delete-local  
-p "JMarsch:Passwort@Orion:1024/StarDraw/StarDraw/Source Code"  
-filter "N" "**"
```

Verwandte Referenz

[Weitere Client-Befehlszeilenoperationen](#)

Dateihistorien anzeigen: stcmd hist

Verwenden Sie den Befehl `stcmd hist`, um die Revisionshistorie von Dateien anzuzeigen.

Syntax

Die Syntax für diesen Befehl lautet wie folgt:

```
stcmd hist -p "Projektspezifikation" [-pwdfile  
"Dateipfad"] [-cmp] [-csf]  
[-encrypt Verschlüsselungstyp]  
[-cfgl "Label-Name"  
| -cfgp "Status"  
| -cfgd "Datum"]  
[-is] [-q]  
[-x] [-stop]  
[-rp "Ordnerpfad"  
| -fp "Ordnerpfad"]  
[-filter "Dateistatus"]  
[Dateien...]
```

Option	Beschreibung
<code>-cfgd</code>	Konfiguriert die Ansicht mit den angegebenen Datums-/Uhrzeitangaben. Folgende Beispiele sind u. a. möglich: "03.01.05 11:25" "3. Januar 2005 11:25:00 CET" "Montag, 3. Januar 2005 11:25 Uhr CET"
<code>-cfgl</code>	Konfiguriert die Ansicht mit dem angegebenen Label. Ohne <code>-cfgl</code> , <code>-cfgp</code> oder <code>-cfgd</code> wird die aktuelle Konfiguration der Ansicht verwendet.
<code>-cfgp</code>	Konfiguriert die Ansicht mit dem angegebenen Heraufstufungsstatus.
<code>-filter</code>	Gibt einen String mit einem oder mehreren Zeichen an, von denen jedes einen Dateistatus darstellt. Dieser String darf keine Leerzeichen oder anderen Whitespace-Zeichen enthalten. Es werden nur Dateien aufgezeichnet, die aktuell den angegebenen Status aufweisen. Die Historie von Dateien mit dem Status "Nicht in Ansicht" kann nicht angezeigt werden. Zur Darstellung der einzelnen Statuswerte werden die folgenden Buchstaben verwendet: C für "Current" (Aktuell), M für "Modified" (Geändert), G für "Merge" (Zusammenführen), O für "Out of Date" (Veraltet), I für "Missing" (Fehlt) und U für "Unknown" (Unbekannt).

Beispiel

Im folgenden Beispiel wird der Befehl `stcmd hist` verwendet, um die Revisionshistorie der Datei `star.h` im Ordner `SourceCode` anzuzeigen, der ein Unterordner des Stammordners `StarDraw` (in der Ansicht "StarDraw" des Projekts "StarDraw") ist.

```
stcmd hist -p "JMarsch:Passwort@Orion:1024/StarDraw/StarDraw/Source Code"  
"star.h"
```

Verwandte Referenz

[Weitere Client-Befehlszeilenoperationen](#)

Dateien auflisten: stcmd list

Verwenden Sie den Befehl `stcmd list`, um alle Dateien eines Ordners, die mit der Option `-p` angegeben werden, aufzulisten. Es können Dateien angezeigt werden, die zu einem angegebenen Zeitpunkt vorhanden waren oder die ein bestimmtes Label tragen.

Syntax

Die Syntax für diesen Befehl lautet wie folgt:

```
stcmd list -p "Projektspezifikation" [-pwdfile
  "Dateipfad"] [-cmp] [-csf]
  [-encrypt Verschlüsselungstyp]
  [-cfgl "Label-Name"
  | -cfgp "Status"
  | -cfgd "Datum"]
  [-is] [-q]
  [-x] [-stop]
  [-rp "Ordnerpfad"
  | -fp "Ordnerpfad"]
  [-filter "Dateistatus"]
  [-cf] [Dateien...]
```

Option	Beschreibung
<code>-cf</code>	Fügt die Namen der untergeordneten Ordner innerhalb des Ordners zur Liste hinzu.
<code>-cfgd</code>	Konfiguriert die Ansicht mit den angegebenen Datums-/Uhrzeitangaben. Folgende Beispiele sind u. a. möglich: "03.01.05 11:25" "3. Januar 2005 11:25:00 CET" "Montag, 3. Januar 2005 11:25 Uhr CET"
<code>-cfgl</code>	Konfiguriert die Ansicht mit dem angegebenen Label. Ohne <code>-cfgl</code> , <code>-cfgp</code> oder <code>-cfgd</code> wird die aktuelle Konfiguration der Ansicht verwendet. <code>-cfgp</code> konfiguriert die Ansicht unter Verwendung des angegebenen Heraufstufungsstatus.
<code>-filter</code>	Gibt einen String mit einem oder mehreren Zeichen an, von denen jedes einen Dateistatus darstellt. In diesem String dürfen keine Leerzeichen oder andere Whitespace-Zeichen verwendet werden. Es werden nur Dateien aufgelistet, die aktuell den angegebenen Status aufweisen. Dateien des Typs "Nicht in Ansicht" können nicht aufgelistet werden. Zur Darstellung der einzelnen Statuswerte werden die folgenden Buchstaben verwendet: C für "Current" (Aktuell), M für "Modified" (Geändert), G für "Merge" (Zusammenführen), O für "Out of Date" (Veraltet), I für "Missing" (Fehlt) und U für "Unknown" (Unbekannt).
<code>-short</code>	Eine kurze und einfache Liste der lokalen Dateien mit deren Status. Dabei wird die Abkürzung für den Status sowie der relative Pfad zur Arbeitsdatei angegeben. Beispiel: M /starteam/Server.java N /starteam/LabelInfo.java Wenn diese Option nicht verwendet wird, enthält die Liste Folgendes: Eine Zeile für jeden Ordnernamen gefolgt vom Pfad des Arbeitsordners. Innerhalb eines Ordners eine Zeile für jede Datei, beginnend mit dem nicht abgekürzten Status sowie den Zugriffsrechten, dem Zeitstempel und dem Namen der Datei. Beispiel: Ordner: Quelle (Arbeitsverz: E:\Quelle)

Beispiel

Im folgenden Beispiel wird der Befehl `stcmd list` verwendet, um alle Dateien des Ordners "Source Code" und seiner Unterordner aufzulisten. "Source Code" ist ein Unterordner des Stammordners "StarDraw "(in der Ansicht "StarDraw" des Projekts "StarDraw").

```
stcmd list -p "JMarsch:Passwort@Orion:1024/StarDraw/StarDraw/Source Code"  
-is "*"
```

Die Option `-short` gibt eine kurze und einfache Liste der lokalen Dateien mit deren Status zurück. Dabei wird die Abkürzung für den Status sowie der relative Pfad zur Arbeitsdatei angegeben. Beispiel:

```
M /starteam/Server.java N /starteam/LabelInfo.java.
```

Ohne die Option `-short` besteht die Liste aus einer Zeile für jeden Ordnernamen gefolgt vom Pfad des Arbeitsordners. Innerhalb eines Ordners enthält die Liste eine Zeile für jede Datei, beginnend mit dem Status (nicht abgekürzt) sowie den Zugriffsrechten, dem Zeitstempel und dem Namen der Datei. Beispiel:

```
Ordner: Quelle (Arbeitsverz: E:\Quelle) Unknown rw 4/6/02 7:42:18 PM PST 230 req.bmp
```

Verwandte Referenz

[Weitere Client-Befehlszeilenoperationen](#)

Dateien sperren und entsperren: stcmd lck

Verwenden Sie den Befehl `stcmd lck`, um Dateien von der Befehlszeile aus zu sperren oder zu entsperren.

Option	Beschreibung
<code>-break</code>	Setzt die von einem anderen Benutzer gesetzte Sperre außer Kraft, sofern Sie die erforderlichen Zugriffsrechte haben.
<code>-filter</code>	Gibt einen String mit einem oder mehreren Zeichen an, von denen jedes einen Dateistatus darstellt. In diesem String dürfen keine Leerzeichen oder andere Whitespace-Zeichen verwendet werden. Es werden nur Dateien mit dem angegebenen Status gesperrt oder entsperrt. Dateien mit dem Status "Nicht in Ansicht" können weder gesperrt noch entsperrt werden. Zur Darstellung der einzelnen Statuswerte werden die folgenden Buchstaben verwendet: C für "Current" (Aktuell), M für "Modified" (Geändert), G für "Merge" (Zusammenführen), O für "Out of Date" (Veraltet), I für "Missing" (Fehlt) und U für "Unknown" (Unbekannt).
<code>-l</code>	Sperrt die Dateien. Dies ist die Vorgabe, wenn <code>-l</code> , <code>-nel</code> oder <code>-u</code> nicht verwendet werden.
<code>-nel</code>	Sperrt die Dateien nicht-exklusiv.
<code>-ro</code>	Definiert nach dieser Operation die Arbeitsdatei als schreibgeschützt. Ohne diese Option bliebe der Status der Datei derselbe wie vor der Operation. In der Regel verwenden Sie <code>-ro</code> , um zu verhindern, dass Sie selbst eine Datei bearbeiten, die Sie nicht gesperrt haben. <code>-ro</code> muss mit <code>-l</code> , <code>-u</code> oder <code>-nel</code> verwendet werden. Wenn Sie <code>-ro</code> verwenden, ist die Verwendung von <code>-rw</code> nicht möglich.
<code>-rw</code>	Gibt die Arbeitsdatei nach dieser Operation zum Lesen und Schreiben frei. Ohne diese Option bliebe der Status der Datei derselbe wie vor der Operation. <code>-rw</code> muss mit <code>-l</code> , <code>-u</code> oder <code>-nel</code> verwendet werden. Wenn Sie <code>-rw</code> verwenden, ist die Verwendung von <code>-ro</code> nicht möglich.
<code>-u</code>	Entsperrt die Dateien.

Syntax

Die Syntax für diesen Befehl lautet wie folgt:

```
stcmd lck -p "Projektspezifikation" [-pwdfile  
"Dateipfad"] [-cmp] [-csf]  
[-encrypt Verschlüsselungstyp]  
[-is] [-q]  
[-x] [-stop]  
[-rp "Ordnerpfad"  
| -fp "Ordnerpfad"]  
[-filter "Dateistatus"]  
[-break] [-l  
| -u | -nel]  
[-ro | -rw]  
[Dateien...]
```

Beispiel

Im folgenden Beispiel wird der Befehl `stcmd lck` verwendet, um alle Dateien des Ordners "Source Code" und seiner Unterordner zu entsperren. "Source Code" ist ein Unterordner des Stammordners "StarDraw" (in der Ansicht "StarDraw" des Projekts "StarDraw").

```
stcmd lck -p "JMarsch:Passwort@Orion:1024/StarDraw/StarDraw/Source Code"  
-is -u "*" 
```

Verwandte Referenz

[Weitere Client-Befehlszeilenoperationen](#)

Server sperren und entsperren: stcmd server-mode

Wenn Sie über die entsprechenden Zugriffsrechte verfügen, können Sie mit dem Befehl `stcmd server-mode` Serverkonfigurationen sperren und entsperren. Durch das Sperren einer Serverkonfiguration können Sie den Zugriff einschränken, während Sie eine Sicherungskopie erstellen oder andere Prozeduren durchführen. Ein gesperrter Server akzeptiert nur Serveradministrationsbefehle. Nach dem Entsperren nimmt die Serverkonfiguration ihren normalen Betrieb wieder auf.

Hinweis: Wenn der Benutzername "StarTeam" ist, erfolgt nach Eingabe des Befehls eine Passwortabfrage, auch wenn bereits ein Passwort eingegeben wurde oder wenn der Benutzer über ein leeres Passwort verfügt.

Syntax

Die Syntax für diesen Befehl lautet wie folgt:

```
stcmd server-mode [-pwdfile "Dateipfad"] [-cmp] [-encrypt Verschlüsselungstyp] [-q] [-x]
[-stop] -s "Server" -mode [lock | exlock | unlock]
```

Option	Beschreibung
<code>exlock</code>	Sperrt den Server exklusiv, sodass kein anderer Benutzer darauf zugreifen kann.
<code>lock</code>	Sperrt den Server nicht-exklusiv. Es können nur administrative Befehle ausgeführt werden.
<code>-mode</code>	Gibt an, ob der Server gesperrt, exklusiv gesperrt oder entsperrt werden soll. Wenn Sie <code>-mode lock</code> verwenden, werden bis zum Entsperren des Servers nur Befehle für die Serveradministration akzeptiert. Sie können diesen Befehl beispielsweise verwenden, wenn Sie ein Backup-Programm ausführen. Wenn Sie <code>-mode exlock</code> verwenden, können Sie erst wieder auf den Server zugreifen, wenn dieser entsperrt ist. Sie können diesen Befehl beispielsweise verwenden, wenn Sie ein benutzerdefiniertes Feld erstellen. Verwenden Sie <code>-mode unlock</code> , um den Server wieder für andere Benutzer verfügbar zu machen.
<code>unlock</code>	Entsperrt den Server, sodass alle Benutzer mit den entsprechenden Zugriffsrechten auf ihn zugreifen können.
<code>-s</code>	Der Server. Die vollständige Syntax lautet: <pre>-s "Benutzername:Passwort@Host-Name:Portnummer"</pre> Beispiel: <pre>-s "JMarsch:Passwort@orion:49201"</pre> Wenn der Benutzername weggelassen wird, wird der aktuelle Benutzername verwendet. Der Benutzername im Beispiel lautet "JMarsch". Wenn das Passwort weggelassen wird, wird der Benutzer aufgefordert, das Passwort einzugeben. Das Passwort im Beispiel lautet "Passwort". Wenn der Host-Name weggelassen wird, wird als Vorgabe localhost verwendet. Der Host-Name im Beispiel lautet "orion". Die Port-Nummer ist erforderlich. In diesem Beispiel wird die Standard-Port-Nummer 49201 verwendet.

Beispiel

Im folgenden Beispiel wird der Befehl `stcmd server-mode` verwendet, um den Server zu sperren, der Port 1024 auf Orion verwendet.

```
stcmd server-mode -s "JMarsch:Passwort@Orion:1024"  
-mode lock
```

Verwandte Referenz

[Weitere Client-Befehlszeilenoperationen](#)

Dateien entfernen: stcmd remove

Verwenden Sie den Befehl `stcmd remove`, um Dateien aus der Versionskontrolle zu entfernen. Die angegebenen Dateien und ihre Revisionshistorien werden nicht mehr in StarTeam angezeigt, wenn die Projektansicht nicht auf einen Zeitpunkt vor dem Entfernen zurückgesetzt wird.

Syntax

Die Syntax für diesen Befehl lautet wie folgt:

```
stcmd remove -p "Projektspezifikation" [-pwdfile
  "Dateipfad"] [-cmp] [-csf]
  [-encrypt Verschlüsselungstyp]
  [-is] [-q]
  [-x] [-stop]
  [-rp "Ordnerpfad"
  | -fp "Ordnerpfad"]
  [-filter "Dateistatus"]
  [-df] [Dateien...]
```

Option	Beschreibung
<code>-df</code>	Löscht die Arbeitsdatei des Benutzers. Wenn diese Option nicht verwendet wird, verbleibt die Arbeitsdatei im Arbeitsordner auf Ihrer Workstation.
<code>-filter</code>	Gibt einen String mit einem oder mehreren Zeichen an, von denen jedes einen Dateistatus darstellt. Dieser String darf keine Leerzeichen oder anderen Whitespace-Zeichen enthalten. Es werden nur Dateien entfernt, die aktuell den angegebenen Status aufweisen. Dateien mit dem Status "Nicht in Ansicht" können aus der Versionskontrolle nicht entfernt werden. Zur Darstellung der einzelnen Statuswerte werden die folgenden Buchstaben verwendet: C für "Current" (Aktuell), M für "Modified" (Geändert), G für "Merge" (Zusammenführen), O für "Out of Date" (Veraltet), I für "Missing" (Fehlt) und U für "Unknown" (Unbekannt).

Beispiel

Im folgenden Beispiel wird der Befehl `stcmd remove` verwendet, um alle `.hm`-Dateien aus dem Ordner "Source Code" und seinen Unterordnern zu entfernen. "Source Code" ist ein Unterordner des Stammordners "StarDraw" (in der Ansicht "StarDraw" des Projekts "StarDraw"). Die Arbeitsdateien werden ebenfalls gelöscht.

```
stcmd remove -p "JMarsch:Passwort@Orion:1024/StarDraw/StarDraw/Source Code"
-is -df "*.hm"
```

Verwandte Referenz

[Weitere Client-Befehlszeilenoperationen](#)

Persönliche Optionen anpassen: stcmd set-personal-options

Mit dem Befehl `stcmd set-personal-options` werden persönliche Optionen festgelegt und aufgelistet. Gegenwärtig können mit dem Befehl nur folgende persönliche Optionen festgelegt werden:

- ◆ Art und Weise, wie Dateistatusinformationen gespeichert werden
- ◆ Speicherort für Dateistatusinformationen, wenn diese an einem zentralen Ort gespeichert werden.

Diese Informationen werden in der Datei `starteam-client-options.xml` gespeichert. Wenn die Datei `starteam-client-options.xml` nicht vorhanden ist, wird sie durch diesen Befehl an einem Standard-Ablageort erstellt. Der Standard-Ablageort ist identisch mit dem Standard-Ablageort für die zentrale Speicherung von Statusinformationen (siehe unten).

Syntax

Die Syntax für diesen Befehl lautet wie folgt:

```
stcmd set-personal-options
[-q] [-x]
[-stop] [-central-status
| -per-folder-status] [-central-repository "Ordnerpfad"]
[-list]
```

Option	Beschreibung
<code>-central-repository</code>	Mit dieser Option können Sie einen Speicherort für das zentrale Repository angeben.
<code>-central-status</code>	<p>Bedeutet, dass die Dateistatusinformationen für diesen Benutzer auf dieser Workstation an einem zentralen Speicherort gespeichert werden. Dieser Speicherort kann mit dem Windows-Client, dem Cross-Platform-Client oder mit diesem Befehl festgelegt werden (siehe Option <code>-central-repository</code>).</p> <p>Wenn Sie den Speicherort nicht festlegen, wird standardmäßig folgender Speicherort verwendet:</p> <p>Windows NT: Der Ordner, in dem die Anwendung installiert wurde</p> <p>Windows XP: <code>C:\Dokumente und Einstellungen\ Benutzername\Lokale Einstellungen\Anwendungsdaten\Borland\StarTeam\</code></p> <p>Für Nicht-Windows-Plattformen: <code>/Benutzerstammverzeichnis/.starteam-client/</code></p>
<code>-per-folder-status</code>	Bedeutet, dass jeder Arbeitsordner Dateistatusinformationen für die eigenen Dateien enthält, die sich in ihm befinden. Die Informationen werden im <code>.sbas</code> -Ordner, einem untergeordneten Ordner des Arbeitsordners, gespeichert. Wenn Sie den Arbeitsordner verschieben, werden die Dateistatusinformationen ebenfalls verschoben.
<code>-list</code>	Zeigt eine Liste der persönlichen Optionen an, wie sie derzeit in der Datei <code>starteam-client-options.xml</code> gespeichert sind.

Beispiel

Im folgenden Beispiel wird `stcmd set-personal-options` verwendet, um festzulegen, dass Dateistatusinformationen an einem zentralen Ablageort gespeichert werden: `C:\JMarsch\statusinfo`.

```
stcmd set-personal-options -central-status -central-repository  
"C:\JMarsch\statusinfo"
```

Verwandte Referenz

[Weitere Client-Befehlszeilenoperationen](#)

Dateistatus aktualisieren: stcmd update-status

Wenn Sie den Status einer Datei aktualisieren, vergleicht StarTeam die Arbeitsdatei mit der von Ihnen ausgecheckten Revision und der aktuellen Revision. Wenn beispielsweise auf ihrer Dateiliste der Status "Aktuell" "vermerkt ist, aber jemand anders gerade eine Kopie der Datei eingchecked hat, ist der Status Ihrer Datei in Wirklichkeit "Veraltet".

Das Aktualisieren von Dateistatusinformationen ist nicht gleichbedeutend mit dem Aktualisieren von Dateien. Wenn sich eine Datei nicht in Ihrem Arbeitsordner befindet, erhält sie durch die Aktualisierung den Status "Fehlt". Sie wird jedoch nicht automatisch für Sie ausgecheckt. In der Regel aktualisieren Sie den Dateistatus, um zu entscheiden, ob eine Datei ein- oder ausgecheckt, hinzugefügt oder ignoriert werden soll.

Sie können beispielsweise Folgendes ausführen:

- ◆ Eine Datei einchecken, wenn ihr Status "Veraltet", "Fehlt" oder "Zusammenführung" ist.
- ◆ Eine Datei auschecken, wenn ihr Status "Geändert" oder "Zusammenführung" ist.
- ◆ Eine Datei zur Anwendung hinzufügen, wenn ihr Status "Nicht in Ansicht" ist. Der Befehl `update-status` listet jedoch niemals Dateien auf, deren Status "Nicht in Ansicht" ist, da sie nicht im Repository gespeichert werden.

Mit dem Befehl `stcmd update-status` werden der Dateiname und der Status vor und nach Ausführung des Befehls angezeigt. Es könnte beispielsweise Folgendes ausgegeben werden: `x.cpp: Status ist "Aktuell" (war "Unbekannt")`

Syntax

Die Syntax für diesen Befehl lautet wie folgt:

```
stcmd update-status -p
  "Projektspezifikation"
  [-pwdfile "Dateipfad"]
  [-cmp] [-csf]
  [-encrypt Verschlüsselungstyp]
  [-is] [-q]
  [-x] [-stop]
  [-rp "Ordnerpfad"
  | -fp "Ordnerpfad"]
  [-cfgl "Label-Name"
  | -cfgp "Status"
  | -cfgd "Datum"]
  [-filter "Dateistatus"]
  [-eol [on
  | off ]][-v]
  [Dateien...]
```

Option	Beschreibung
<code>-cfgd</code>	Konfiguriert die Ansicht mit den angegebenen Datums-/Uhrzeitangaben. Folgende Beispiele sind u. a. möglich: "03.01.2005 11:25" "3. Januar 2005 11:25:00 CET" "Montag, 3. Januar 2005 11:25 Uhr CET"
<code>-cfgl</code>	Konfiguriert die Ansicht mit dem angegebenen Label. Ohne <code>-cfgl</code> , <code>-cfgp</code> oder <code>-cfgd</code> wird die aktuelle Konfiguration der Ansicht verwendet.

<code>-cfgp</code>	Konfiguriert die Ansicht mit dem angegebenen Heraufstufungsstatus.
<code>-eol</code>	Wenn diese Option verwendet wird, ermittelt sie den Status von Textdateien, ohne die EOL-Markierungen zu berücksichtigen. Wird diese Option nicht verwendet (Vorgabe), wird der Aktualisierungsstatus für die Arbeitsdatei und die aktuellen Revisionen mit ihren EOL-Markierungen ermittelt. Wird diese Option verwendet, kann eine Arbeitsdatei mit der EOL-Markierung LF mit einer aktuellen Revision mit der EOL-Markierung CR/LF verglichen und als "Aktuell" betrachtet werden, wenn sie sich nur durch die EOL-Markierungen unterscheiden.
<code>-contents</code>	Sendet Dateiinhalte anstelle der MD5-Prüfsumme.
<code>-filter</code>	Gibt einen String mit einem oder mehreren Zeichen an, von denen jedes einen Dateistatus darstellt. Dieser String darf keine Leerzeichen oder anderen Whitespace-Zeichen enthalten. Es werden nur Dateien aktualisiert, die aktuell den angegebenen Status aufweisen. C für "Current" (Aktuell), M für "Modified" (Geändert), G für "Merge" (Zusammenführen), O für "Out of Date" (Veraltet), I für "Missing" (Fehlt) und U für "Unknown" (Unbekannt).
<code>-v</code>	Der Status aller Dateien im angegebenen Arbeitsordner des Ordners, es sei denn, ihr Status lautet "Nicht in Ansicht". Ohne diese Option wird der Status einer Datei nur angezeigt, wenn er sich geändert hat.

Beispiel

Im folgenden Beispiel wird der Befehl `stcmd update-status` verwendet, um sicherzustellen, dass alle Dateien des Arbeitsordners des StarTeam-Ordners `SourceCode` über den richtigen Status verfügen. `Source Code` ist ein Unterordner des Stammordners `StarDraw` (in der StarDraw-Ansicht des StarDraw-Projekts).

```
stcmd update-status -p "JMarsch:Passwort@Orion:1024/StarDraw/StarDraw/Source Code"
"*"
```

Verwandte Referenz

[Weitere Client-Befehlszeilenoperationen](#)

VCM-Befehlszeilenprogramm

Das View Compare/Merge-Befehlszeilenprogramm (`VCMUtility`) vergleicht eine Quellansicht von StarTeam mit einer Zielansicht und führt die Unterschiede optional in der Zielansicht zusammen.

In diesem Abschnitt sind alle `VCMUtility`-Hilfethemen enthalten.

View Compare/Merge steht nur im StarTeam Cross-Platform Client und dem `VCMUtility` zur Verfügung.

In diesem Abschnitt

[Übersicht über das VCM-Befehlszeilendienstprogramm \(VCMUtility\)](#)

Erläutert das Befehlszeilenprogramm zum Vergleichen und Zusammenführen von Ansichten namens `VCMUtility`.

[VCMUtility-Befehle](#)

Erläutert die Befehlstypen des Befehlszeilenprogramms zum Vergleichen und Zusammenführen von Ansichten (`VCMUtility`).

[VCMUtility-Verbindungsoptionen](#)

Erläutert die Verbindungsoptionen des Befehlszeilenprogramms zum Vergleichen und Zusammenführen von Ansichten (`VCMUtility`).

[Sitzungsoptionen von VCMUtility](#)

In diesem Abschnitt werden die `VCMUtility`-Optionen beschrieben, die in neuen Sitzungen verwendet werden können.

[Weitere Optionen des VCM-Dienstprogramms](#)

Erläutert die verschiedenen Optionen des Befehlszeilenprogramms zum Vergleichen und Zusammenführen von Ansichten (`VCMUtility`).

[Beispiele für das VCM-Dienstprogramm](#)

Beispiele zur Verwendung von (`VCMUtility`).

[Befehlsreferenz](#)

Liste der `VCMUtility`-Befehle und Optionen zusammen mit deren Syntax.

[Syntax der zusammengesetzten Optionen des VCM-Dienstprogramms](#)

Erläutert die Syntaxanforderungen für die einzelnen `VCMUtility`-Optionen.

Übersicht über das VCM-Befehlszeilendienstprogramm (VCMUtility)

VCMUtility ist ein Befehlszeilenprogramm, mit dem eine Quellansicht von StarTeam mit einer Zielansicht verglichen werden kann und die Unterschiede optional in der Zielansicht zusammengeführt werden können.

Zur Dokumentation von VCMUtility gehören:

- ◆ Befehle
- ◆ Verbindungsoptionen
- ◆ Sitzungsoptionen
- ◆ Verschiedene Optionen, die nicht in Sitzungen gespeichert werden

Hinweis: Sie können eine Sitzung zum Vergleichen/Zusammenführen von Ansichten über die Befehlszeile starten und sie in dem StarTeam-Cross-Platform-Client in der View Compare/Merge-Benutzeroberfläche beenden. Sie können beispielsweise das VCMUtility verwenden, um eine VCM-Sitzung zu erstellen, und dabei vielleicht die Option DefaultAction verwenden. Sie dürfen jedoch noch keinen Festschreibevorgang zulassen. Die VCM-Sitzung wird automatisch zusammen mit einem ggf. von Ihnen ausgewählten alternativen Namen gespeichert. Danach können Sie die VCM-Sitzung im StarTeam-Cross-Platform-Client öffnen, ggf. Anpassungen vornehmen und die Änderungen in das Repository festschreiben.

Syntaxkonventionen

Für die Befehlszeilensyntax werden folgende Konventionen verwendet:

Konvention	Beschreibung
Geschweifte Klammern { }	Umschließen erforderliche Syntaxelemente.
Eckige Klammern []	Umschließen optionale Elemente.
Spitze Klammern < >	Umschließen ein Wort oder einen Ausdruck, der durch einen oder mehrere geeignete Werte ersetzt werden muss. Beispielsweise würde <Dateiname> durch einen tatsächlichen Dateinamen oder Pfad ersetzt und <Benutzer-ID> durch eine tatsächliche Benutzer-ID. Viele der Wörter oder Ausdrücke in spitzen Klammern können jedoch eine komplexere Syntax haben. Beispielsweise kann <Änderungsanforderungen> durch CR, CRs, ChangeRequest, ChangeRequests, ChangeRequest *4277 usw. ersetzt werden. Wenn Sie nicht sicher sind, was Sie verwenden können, lesen Sie die VCMUtility-Befehlsreferenz im Abschnitt Referenz/CompareMerge der Dokumentation. In der Befehlsreferenz wird die vollständige Syntax für Ausdrücke wie <Änderungsanforderungen> angegeben.
Senkrechter Strich	Trennt alternative Elemente.
Voranstellung eines Sternchens *	Bedeutet, dass das nachfolgende Element wiederholt werden kann.

Hinweis: Bei allen Optionen kann die Groß-/Kleinschreibung ignoriert werden (z. B. ist Server dasselbe wie server).

VCMUtility-Befehl

VCMUtility [<Optionsdatei>] [Optionen]

Sie können die Optionen in Form einer `<Optionsdatei>` (als ersten Parameter), in Befehlszeilenargumenten oder in beiden angeben. Befehlszeilenargumente haben Vorrang vor den Optionen in der `<Optionsdatei>`. In der `<Optionsdatei>` muss der Optionsname am Anfang einer neuen Zeile stehen und ohne den vorangestellten – angegeben werden.

VCMUtility-Optionsdatei

Sie können die `VCMUtility`-Optionen in einer Optionsdatei angeben, deren Name der erste Parameter des `VCMUtility`-Befehls ist.

Beispiel:

```
VCMUtility c:\VCMconfig.txt
```

Jede Option muss in der Datei in einer neuen Zeile beginnen. Optionsnamen müssen in Spalte 1 beginnen und es muss mindestens ein Whitespace-Zeichen folgen. Der Wert einer Option kann sich über mehrere Zeilen erstrecken, wenn jede fortlaufende Zeile mit einem Leerzeichen oder einem Tabulatorzeichen beginnt. Leerzeilen werden ignoriert. Kommentare werden durch zwei vorangestellte Schrägstriche (`//`) gekennzeichnet.

Beispiel:

```
// Dies ist ein Kommentar
server jsmith:mypw@somehost:49201
type Rebase
include "/Cygnus/StarTeam/<StarTeam Core>/Server/Common/*.h" +ALL
*.cpp *.rc Makefile // langer Wert, der in einer zusätzlichen Zeile fortgesetzt
wird

// Die Zeile darüber ist leer
save
my-rebase-session // Wert in separater Zeile
```

Befehlszeilenparameter

`VCMUtility`-Optionen können als Befehlszeilenparameter übergeben werden, indem ein Bindestrich vor den Optionsnamen gesetzt wird. Beispielsweise kann die Option `Server` wie folgt als Befehlszeilenparameter verwendet werden: `-server`. Wenn eine Option weitere Werte hat, müssen sie dem Optionsnamen direkt folgen (ohne Bindestrich).

Input-Quellen mischen

`VCMUtility`-Optionen können in einer Optionsdatei, mit Befehlszeilenparametern oder durch eine Mischung aus beiden angegeben werden. Häufig verwendete oder "statische" Optionswerte können beispielsweise in der Konfigurationsdatei angegeben werden, während "dynamische" Werte als Befehlszeilenparameter angegeben werden können.

Mit einem Befehlszeilenparameter kann dieselbe Option angegeben werden, wie sie in der Konfigurationsdatei definiert wird. Wenn ein Befehlszeilenargument dieselbe Option enthält wie die Optionsdatei, hat der Wert der Befehlszeilenoption Vorrang vor dem Wert der Konfigurationsdateioption. Wenn beispielsweise in der Konfigurationsdatei `-Source View1` und in der Befehlszeile `-Source View2` angegeben wird, wird `View2` als Quellansicht verwendet.

Optionswerte und Unicode-Zeichen

Die Codierung der Optionswerte, die als Befehlszeilenargumente übergeben werden, wird durch die Umgebung (z. B. die Command-Shell) gesteuert, in der das Programm gestartet wird. Optionen, die an `VCMUtility` übergeben

werden müssen und die Zeichen enthalten, die in dieser speziellen Umgebung nicht dargestellt werden können, müssen anhand der Optionsdatei übergeben werden.

Wenn die Optionsdatei nicht mit einer BOM ("byte-order mark") beginnt, wird sie mit dem Standardzeichensatz des Systems (z. B. ANSI [Windows-1252] auf Windows, UTF-8 auf Linux) geöffnet. Hat die Optionsdatei am Dateianfang eine BOM, wird der passende Zeichensatz zugeordnet. Die Codierungen UTF-8 und UTF-16 ermöglichen die Verwendung des vollständigen Satzes an Unicode-Zeichen in der Optionsdatei.

Verfügbare BOM-Sequenzen:

BOM	Codierung
0xEFBBBF	UTF-8
0xFEFF	UTF-16 BE (big-endian)
0xFFFE	UTF-16 LE (little-endian)

Boolesche Optionen

Die Vorgabe für alle booleschen Optionen (die `True` oder `False` sein können) lautet `False`. Allerdings entspricht die Verwendung einer booleschen Option ohne Angabe eines Optionswerts der Angabe von `True`. Dies bedeutet, eine boolesche Option wird dadurch aktiviert, dass sie aufgeführt wird. Beispiel:

```
// Diese Optionen auf True setzen
AutoLogon
BreakLocks
```

Abkürzungen

Zusätzlich zu ihrer in diesem Dokument verwendeten "langen Form" haben die meisten Befehls- und Optionsnamen eine oder mehrere "Kurzformen" oder Abkürzungen. Mit diesen alternativen Formen können `VCMUtility`-Befehle mit vielen Befehlsoptionen verkürzt werden. Die vollständige Liste der Abkürzungen erhalten Sie durch Eingabe des Befehls `-Help abbreviations`. Nachfolgend sind einige Abkürzungen aufgeführt:

`Help`: `H` oder `?`.

`ActiveProcessItem`: `ActivePI` oder `API`.

`SourceLabel`: `SrcLabel` oder `SL`.

In den meisten Fällen kann ein syntaktisches Element, dessen einzelne Wortbestandteile mit Großbuchstaben beginnen, anhand dieser Großbuchstaben abgekürzt werden. Beispielsweise kann `ManualMergeFiles` durch `MMF` oder `AutoMergeProperties` durch `AMP` usw. abgekürzt werden.

Exit-Codes

Das VCM-Dienstprogramm gibt folgende Exit-Codes zurück, die für das Ergebnis der Ausführung stehen:

Exit-Code	Beschreibung
0	Keine Fehler aufgetreten.
1	Es ist ein schwerwiegender Fehler aufgetreten.

2	Teilweise erfolgreich. Dieses Ergebnis wird zurückgegeben, wenn die Vergleichsphase, aber aufgrund nicht lösbarer Konflikte die Festschreibungsphase nicht durchgeführt werden konnte.
---	--

VCMUtility-Protokolldateien

Während der Ausführung schreibt `VCMUtility` Status-, Warn- und Fehlermeldungen in das Konsolenfenster (Standardausgabe). Für die meisten Vorgänge erstellt das `VCMUtility` auch eine Protokolldatei, in der der Vorgang zusammengefasst ist. Wie im Falle der Konsolenfensterausgabe werden mehr Details in die Protokolldatei geschrieben, wenn die Option `Verbose` aktiviert ist. Die Protokolldatei wird für neue VCM-Sitzungen sowie für die Befehle `Import`, `Open`, `Replay` und `Resume` erstellt. Die Protokolldatei wird jedoch erst dann angelegt, wenn die Befehlszeilenparameter und die Optionsdatei, falls verwendet, ohne Fehler analysiert werden konnten. Eine Protokolldatei wird nicht für die Befehle `Help` und `Delete` erstellt.

Die VCMUtility-Protokolldatei wird mit dem folgenden Dateinamen im Stammverzeichnis des Benutzers (als `user.home` in Java bezeichnet) erstellt:

```
VCMUtility-JJJJ-MM-TT_hh-mm-ss.log
```

wobei `JJJJ-MM-TT` und `hh-mm-ss` das aktuelle Datum bzw. die aktuelle Uhrzeit der jeweiligen Zeitzone sind. Der vollständige Pfadname der Protokolldatei wird im Konsolenfenster angezeigt, wenn die Protokolldatei angelegt wird.

VCMUtility-Unterstützung für Änderungspakete

Ab StarTeam 2009 unterstützt das `VCMUtility` Änderungspakete für alle StarTeam-Konfigurationen, die auf die Version 2009 aktualisiert wurden. Da Änderungspakete beständige Objekte sind, die auf dem Server gespeichert werden, bieten sie viele Vorteile gegenüber VCM-Sitzungs- (`.vcms`) und VCM-Exportdateien (`.vcmx`). Demnach werden zum Speichern und zur Wiederaufnahme von Sitzungen bei aktualisierten StarTeam-Konfigurationen Änderungspakete gegenüber Sitzungs- und Exportdateien vorgezogen. Entsprechend werden die `Save`-Option ohne Parameter und der `Open`-Befehl gegenüber der `Save`-Option mit Parameter, dem `Resume`-, `Export`- und `Import`-Befehl vorgezogen. Aus Gründen der Abwärtskompatibilität unterstützt das `VCMUtility` 2009 jedoch nach wie vor Befehle, die auf VCM-Sitzungsdateien zurückgreifen. Weitere Informationen hierzu finden Sie unter dem `Open`-Befehl und der `Save`-Option.

Verwandte Referenz

[VCMUtility-Befehle](#)

[VCMUtility-Verbindungsoptionen](#)

[Sitzungsoptionen von VCMUtility](#)

[Weitere Optionen des VCM-Dienstprogramms](#)

[Beispiele für das VCM-Dienstprogramm](#)

[Befehlsreferenz](#)

VCMUtility-Befehle

In diesem Abschnitt wird die `VCMUtility`-Funktionalität erläutert. Bei jedem Aufruf von `VCMUtility` wird ein Befehl ausgeführt.

Eine Beschreibung der Syntaxkonventionen finden Sie in den Referenz-Links unter "Übersicht über das VCM-Befehlszeilenprogramm".

VCMUtility-Befehl

```
VCMUtility [<Optionsdatei>] [Optionen]
```

Sie können die Optionen in Form einer `<Optionsdatei>` (als ersten Parameter), in Befehlszeilenargumenten oder in beiden angeben. Befehlszeilenargumente haben Vorrang vor den Optionen in der `<Optionsdatei>`. Geben Sie in der `<Optionsdatei>` die Optionsnamen in Spalte 1 ohne Angabe des führenden Bindestrichs ("-") ein.

VCMUtility-Befehlstypen

In diesem Abschnitt werden die `VCMUtility`-Befehlstypen behandelt. Der Standard-Befehlstyp ist eine neue VCM-Sitzung.

Neue Sitzung (Befehl)

Standardmäßig startet jede Ausführung von `VCMUtility` eine neue VCM-Sitzung, es sei denn, es wird explizit der Befehl `Help`, `OPEN`, `Replay`, `Resume`, `Delete` oder `Import` angegeben.

Help (Befehl)

?

```
Help [<Option>]
```

Zeigt die `VCMUtility`-Hilfe an. Wenn Sie eine `<Option>`, angeben, wird ein themenspezifischer Hilfetext angezeigt. Bei der Angabe von `Help MMF` wird beispielsweise die Hilfe zur Option `ManualMergeFiles` angezeigt.

Delete (Befehl)

```
Delete <VCM-Sitzungsdatei>
```

Gibt an, dass die Sitzung, die in der angegebenen `<VCM-Sitzungsdatei>` gespeichert ist, gelöscht werden soll. Alle Zwischendateien (beispielsweise zusammengeführte Ergebnisdateien) und die Sitzungsdatei selbst werden gelöscht. Wenn die Sitzung jedoch bereits als nicht festgeschriebenes Änderungs paket in der Zielansicht gespeichert wurde, wird das Änderungs paketobjekt nicht gelöscht.

Import (Befehl)

```
Import <VCM-Austauschdatei>
```

Der `Import`-Befehl ist identisch mit dem `Resume`-Befehl mit der Ausnahme, dass die an ihn übergebene `<VCM-Austauschdatei>` eine VCM-Austauschdatei (`.vcmx`) sein muss, die zuvor durch einen `Export`-Befehl erstellt wurde. Die importierte VCM-Sitzung wird dort fortgesetzt, wo sie beendet wurde:

- ◆ Die Vergleichsphase wird durchgeführt, wenn sie noch nicht erfolgreich abgeschlossen wurde.
- ◆ Eine manuelle Zusammenführung wird durchgeführt, wenn `ManualMergeFiles` angegeben wird und es Konflikte bei der Dateizusammenführung gibt.

- ◆ Die Zielansicht "merge preview" wird ausgecheckt, wenn `CheckoutPreview` angegeben wird und die Festschreibung noch nicht erfolgt ist.
- ◆ Der Unterschiedsbericht wird erzeugt, wenn `ReportDiffs` angegeben wird und die Festschreibung noch nicht erfolgt ist.
- ◆ Die Festschreibungsphase wird durchgeführt, wenn `CommitMerge` den Wert `True` hat und die Festschreibung noch nicht erfolgt ist.
- ◆ Der Aktualisierungsbericht wird erzeugt, wenn `ReportUpdates` angegeben wird und die Festschreibung ausgeführt wurde.

`Export` und `Import` können gemeinsam verwendet werden, um eine VCM-Sitzung von einer auf eine andere Workstation zu übertragen. Beispielsweise könnte ein Benutzer eine neue VCM-Sitzung erstellen, alle Konflikte auflösen und anschließend die Sitzung `exportieren`. Die daraus resultierende Archivdatei kann dann auf einen Test-Computer übertragen werden. Der `Import`-Befehl kann dann in Verbindung mit der `CheckoutPreview`-Option (wobei `CommitMerge` auf `<keyword>False</keyword>` gesetzt ist) zum Auschecken, Erzeugen und Testen der Zielzusammenführungsvorschau verwendet werden. Verlaufen die Tests erfolgreich, kann der Test-Computer den `Resume`-Befehl ausführen und `CommitMerge` auf `True` setzen.

Hinweis: Sitzungen, die mit den Befehlen `Resume` oder `Import` fortgesetzt werden, werden nicht automatisch gespeichert, wenn sie nicht festgeschrieben werden. Bei Angabe der Option `Save` wird die Sitzung in der angegebenen `<VCM-Sitzungsdatei>` gespeichert. Anderenfalls wird die durch einen `Resume` -Befehl angegebene VCM-Sitzung verwendet; ein automatisch erzeugter Dateiname für die VCM-Sitzung wird für einen `Import`-Befehl verwendet.

Open (Befehl)

`Open <Name des Änderungspakets>`

Setzt eine VCM-Sitzung fort, die vorher mit dem angegebenen Namen als Änderungspaket gespeichert wurde. Diese Option steht nur auf Servern zur Verfügung, die Änderungspakete unterstützen. Bei dem angegebenen Namen muss es sich um den Standard- oder den vom Benutzer angegebenen Namen eines gespeicherten, nicht festgeschriebenen Änderungspakets handeln, das zum angegebenen `Project` und `TargetView` gehört (diese sind erforderlich). Die Sitzung darf auch nicht von einem anderen Benutzer gesperrt sein, was gewöhnlich erkennen lässt, dass sie bereits von diesem Benutzer geöffnet wurde.

Weitere Informationen hierzu finden Sie bei den Befehlen `Name`, `Save`, `Import` und `Resume`.

Replay (Befehl)

`Replay <Name des Änderungspakets>`

Erstellt eine neue VCM-Sitzung, indem ein bereits festgeschriebenes Änderungspaket in einer neuen Zielansicht "wiederholt" wird. Dieser Befehl steht nur dann zur Verfügung, wenn der Server Änderungspakete unterstützt. Das benannte Änderungspaket muss dem Projekt angehören, das durch die `Project`-Option und die über die Option `SourceView` definierte Ansicht angegeben ist. (Da festgeschriebene Änderungspakete zu der Zielansicht gehören, die sie aktualisieren, ist die Zielansicht des Änderungspakets, das wiederholt werden soll, immer die Quellansicht der neuen Sitzung.)

Die `TargetView` sollte angegeben werden, wenn der `Replay` -Befehl verwendet wird. Somit kann basierend auf der Beziehung zwischen den beiden Ansichten automatisch der `MergeType` der neuen Sitzung ausgewählt werden:

- ◆ Wenn die Zielansicht ein untergeordnetes Objekt der Quellansicht ist, wird eine Sitzung zur Neubestimmung der Basis durchgeführt.
- ◆ Wenn die Zielansicht ein übergeordnetes Objekt der Quellansicht ist, wird eine Heraufstufungssitzung durchgeführt.

- ◆ Anderenfalls wird eine Replizierungssitzung durchgeführt.

Alternativ können Sie einen `MergeType` von "Promote" angeben. In diesem Fall ist keine Zielansicht erforderlich.

In einer VCM-Wiederholungssitzung wird versucht, die gleichen Änderungen in der neuen Zielansicht vorzunehmen, die im angegebenen Änderungspaket vorgenommen wurden. Dies bedeutet, dass der Quellbereich der neuen VCM-Sitzung automatisch ausgewählt wird. Somit sind die Optionen `Include` und `Exclude` nicht erlaubt. In einer Wiederholungssitzung sind manche Änderungen, die im ursprünglichen Änderungspaket vorgenommen wurden, in der neuen Zielansicht nicht möglich (z. B. wenn eine neue Version bereits vorhanden ist). Einige Änderungen müssen möglicherweise auf andere Art und Weise vorgenommen werden (beispielsweise `Move-and-Merge` anstatt `Merge`) und neue Konflikte könnten auftreten (z. B. `Merge` anstatt `Repin`). Die Wiederholungssitzung kann nur dann festgeschrieben werden, wenn keine ungelösten Konflikte bestehen.

Resume (Befehl)

`Resume <VCM-Sitzungsdatei>`

Gibt an, dass die Sitzung, die in der angegebenen `<VCM-Sitzungsdatei>` gespeichert ist, fortgesetzt werden soll, anstatt dass eine neue Sitzung erstellt wird. Normalerweise wird damit die Festschreibungsphase einer vorherigen Sitzung gestartet, für die bislang nur die Vergleichsphase durchgeführt wurde. Eine Sitzung, die bereits festgeschrieben wurde, kann auch fortgesetzt werden, aber nur, um einen Unterschiedsbericht zu generieren. Weitere Informationen hierzu finden Sie unter der `Export`-Option und dem `Import`-Befehl.

Verwandte Referenz

- [Übersicht über das VCM-Befehlszeilendienstprogramm \(VCMUtility\)](#)
- [VCMUtility-Verbindungsoptionen](#)
- [Sitzungsoptionen von VCMUtility](#)
- [Weitere Optionen des VCM-Dienstprogramms](#)
- [Beispiele für das VCM-Dienstprogramm](#)
- [Befehlsreferenz](#)

VCMUtility-Verbindungsoptionen

In diesem Abschnitt wird die `VCMUtility`-Funktionalität hinsichtlich der Verbindungsoptionen erläutert.

AutoLogon

AL

`AutoLogon [True] | [False]`

Wenn ein `<Benutzer>` nicht mit der Option `Server` angegeben wird, fordert `AutoLogon` auf, ob unter Verwendung der Benutzer-ID/des Passworts des angegebenen StarTeam-Servers, die von StarTeamToolbar-Dienstprogramm gespeichert wird, eine Anmeldung versucht werden soll.

Encryption

Encrypt

En

`Encryption {NONE | RC4 | RC2_ECB | RC2_CBC | RC2_CFB}`

Gibt die Verschlüsselungsstufe der Serververbindung an. Die Vorgabe ist `NONE`. Allerdings aktualisiert `VCMUtility` aufgrund des SDK-Verhaltens bei Bedarf die Verschlüsselungsstufe automatisch auf den vom StarTeam-Server benötigten Mindestwert.

PwdFile

PF

`PwdFile <Dateiname>`

Gibt eine Datei an, die das Passwort für die Anmeldung enthält. `-PwdFile` hat Vorrang vor dem `<Passwort>`, wenn dieses im `Server`-Parameter angegeben wird.

Server

S

`Server [<Benutzer>[:<Passwort>]@]<Host>[:<Port>]`

Gibt den StarTeam-Server an, zu dem das VCM-Dienstprogramm eine Verbindung herstellt.

- ◆ Werden `<Benutzer>` und `AutoLogon` nicht angegeben, wird als `<Benutzer>` für die Anmeldung standardmäßig "Administrator" verwendet.
- ◆ Wird weder `<Passwort>` noch `PwdFile` angegeben, wird der Benutzer von `VCMUtility` aufgefordert, ein Passwort einzugeben.
- ◆ Wenn `<Benutzer>` oder `<Passwort>` die Zeichen ":" oder "@" bzw. ein Leerzeichen enthält, muss der Wert zwischen einfache oder doppelte Anführungszeichen gesetzt werden.
- ◆ Wird ein `<Benutzer>` oder ein `<Passwort>` von Anführungszeichen umschlossen, kann er bzw. es ein Anführungszeichen enthalten, wenn diesem ein umgekehrter Schrägstrich (\) als Escape-Zeichen vorangestellt wird.
- ◆ Enthält ein `<Benutzer>` oder ein `<Passwort>`, der bzw. das von Anführungszeichen umschlossen ist, einen umgekehrten Schrägstrich, muss diesem ein weiterer umgekehrter Schrägstrich als Escape-Zeichen

vorangestellt werden. Ein doppelter umgekehrter Schrägstrich in einem von Anführungszeichen umschlossenen Token wird als einzelner umgekehrter Schrägstrich interpretiert.

- ◆ Der Server-`<Host>` kann ein Host-Name oder eine IP-Adresse sein. `<Host>` ist erforderlich, wenn die `Server`-Option angegeben wird.
- ◆ Wird die `Server`-Option nicht angegeben, wird als `<Host>` standardmäßig `localhost` verwendet. Wird `<Port>` nicht angegeben, wird standardmäßig "49201" verwendet.

UseCA

UCA

```
UseCA {<Host>:<Port> | AutoLocate}
```

Gibt an, dass bei Auscheckvorgängen von Dateien ein MPX-Cache-Agent verwendet werden soll. Der Cache-Agent kann entweder explizit mit einem Host-Namen oder einer Adresse (`<Host>`) und Port-Nummer (`<Port>`) angegeben werden bzw. der im Netzwerk am nächsten befindliche Cache-Agent kann automatisch gesucht werden (`AutoLocate`).

UseServerProfile

USP

```
UseServerProfile [True | False]
```

Wenn "true", gibt diese Option an, dass der `<Host>`-Name, der in der `Server`-Option angegeben ist, als Server-Profilname zu interpretieren ist. Serverprofile werden in der Datei `starteam-servers.xml` des Benutzers gespeichert. Ein Serverprofil gibt den StarTeam-Server-Host-Namen, die Port-Nummer, die Verschlüsselungsstufe und die Komprimierungseinstellung an. Daher muss bei Angabe von `UseServerProfile` auch die `Server`-Option angegeben werden. Diese darf aber keine Port-Nummer enthalten. Außerdem darf die Option `Encryption` nicht angegeben werden.

Verwandte Referenz

- [Übersicht über das VCM-Befehlszeilendienstprogramm \(VCMUtility\)](#)
- [VCMUtility-Befehle](#)
- [Sitzungsoptionen von VCMUtility](#)
- [Weitere Optionen des VCM-Dienstprogramms](#)
- [Beispiele für das VCM-Dienstprogramm](#)
- [Befehlsreferenz](#)

Sitzungsoptionen von VCMUtility

In diesem Abschnitt werden die `VCMUtility`-Optionen beschrieben, die in neuen Sitzungen verwendet werden können.

Sitzungsoptionen können in zwei Abschnitte unterteilt werden: "Optionen für neue Sitzungen" und "Optionen für fortgesetzte Sitzungen".

Optionen für neue Sitzungen

In diesem Abschnitt werden die `VCMUtility`-Sitzungsoptionen behandelt.

AutoMergeFiles

AMF

`AutoMergeFiles [True | False]`

Wenn `True`, wird eine automatische Zusammenführung der Dateien angefordert, die in der Vergleichsphase den Zusammenführungsstatus haben. Verläuft eine automatische Zusammenführung erfolgreich, wird die resultierende Datei als Teil der VCM-Sitzung beibehalten. Anderenfalls wird die resultierende Datei verworfen und die betroffenen Dateien verbleiben im unaufgelösten Status der Zusammenführung. `AutoMergeFiles` wird bei Vergleichssitzungen ignoriert.

AutoMergeProperties

AMP

`AutoMergeProperties [True | False]`

Wenn `True`, wird eine automatische Zusammenführung der Eigenschaften für Elemente angefordert, die in der Vergleichsphase den Zusammenführungsstatus haben. Verläuft eine automatische Zusammenführung der Eigenschaften erfolgreich, wird das zusammengeführte Element als Teil der VCM-Sitzung beibehalten. Anderenfalls wird das zusammengeführte Element verworfen und die Elemente werden als in einem unaufgelösten Status der Eigenschaftszusammenführung befindlich gekennzeichnet. `AutoMergeProperties` wird bei Vergleichssitzungen ignoriert.

BreakLocks

BL

`BreakLocks [True | False]`

Wenn `True`, wird angefordert, dass ein Versuch unternommen wird, alle Sperren von Elementen zu beseitigen, die in der Vergleichsphase verwendet werden. Das Beseitigen einer Sperre von einem Quell- oder Zielelement ist nur dann erforderlich, wenn ein anderer Benutzer der Eigentümer der Sperre ist. Zur Beseitigung einer Sperre ist eine spezielle Berechtigung erforderlich. Der Vorgang kann möglicherweise nicht erfolgreich durchgeführt werden. `BreakLocks` wird bei Vergleichssitzungen ignoriert.

CaseSensitiveFileNames

CSF

`CaseSensitiveFileNames [True | False]`

Wenn `True`, werden Dateinamen, die sich nur durch Groß-/Kleinschreibung unterscheiden, bei der Auswertung der Option `PreventDuplicateFileNames` und beim Vergleich von Dateien in Quell- und Zielansichten als unterschiedlich angesehen.

CheckoutPreview

Auschecken

CP

CheckoutPreview <Dateien> [<Auscheckoptionen>]

Diese Option gibt an, dass Dateien in einer Zusammenführungsvorschau ("merge preview") in den Client-Arbeitsbereich ausgecheckt werden. Eine Zusammenführungsvorschau ist eine Simulation der Zielansicht, aktualisiert mit allen Änderungen, die in der VCM-Sitzung vorgenommen wurden. Mit der Angabe für <Dateien> können Dateinamen und/oder -muster aus angegebenen Ordnern in der Zusammenführungsvorschau ausgecheckt werden. Mit den optionalen <Auscheckoptionen> wird festgelegt, wohin Dateien ausgecheckt werden sollen. Zudem können die auszucheckenden Dateien anhand des Status festgelegt werden.

Wird `CheckoutPreview` angegeben, werden die Dateien nach der Vergleichsphase ausgecheckt, nachdem die automatische oder manuelle Zusammenführung stattgefunden hat, aber vor der Festschreibung. Der Auscheckvorgang findet nur statt, wenn es in der VCM-Sitzung keine Zusammenführungskonflikte bezüglich des Dateiinhalts gibt. Bestehen solche Zusammenführungskonflikte, wird ein Fehler angezeigt und unabhängig von der `CommitMerge`-Option keine Zusammenführung durchgeführt. Bestehen keine Zusammenführungskonflikte und `CommitMerge` ist `True`, wird die VCM-Sitzung nach dem Auscheckvorgang festgeschrieben.

Beispiel:

```
CheckoutPreview /src/com/acme/*.java +cwf +eol LF +filter CGMIOU +o +ro
+rp C:\BuildDir
```

CommitMerge

Festschreiben

CM

CommitMerge [True | False]

Gibt an, ob die Ergebnisse der VCM-Sitzung festgeschrieben werden sollen oder nicht. `False` gibt an, dass keine Festschreibung ausgeführt wird. Mithilfe dieser Option kann eine Sitzung nur zum Vergleichen und zur Berichtserstellung gestartet werden. `True` gibt an, dass die Daten nur festgeschrieben werden, wenn es keine ungelösten Konflikte mehr gibt. `CommitMerge` wird bei Vergleichssitzungen ignoriert.

DefaultAction

DA

DefaultAction [MergeType <Zusammenführungstyp>] [ItemType <Elementtyp>] <Bedingungen für Übereinstimmung> <Aktion>

Gibt eine Standard-<Aktion> für Elemente an, die verglichen werden und die die unter <Bedingungen für Übereinstimmung> spezifizierten Bedingungen erfüllen. Das VCM-Dienstprogramm verwendet eine auf Regeln basierende "Entscheidungstabelle", um zu ermitteln, welche Aktion bei Bedarf durchzuführen ist, wenn zwischen der Quell- und Zielansicht Unterschiede in den Elementen gefunden werden. Mit `DefaultAction` können die Standardregeln aufgehoben werden. Diese Option kann mehrfach angegeben werden, um für die einzelnen Unterschiede jeweils eigene Standardaktionen zu definieren. Dabei spielt die Reihenfolge der Definitionen eine gewichtige Rolle: Wenn zwei Aufhebungen auf einen Elementunterschied zutreffen, der in der Vergleichsphase gefunden wird, hat die letzte angegebene Aufhebung Vorrang vor der vorherigen Aufhebung.

- ◆ Wird `Zusammenführungstyp` angegeben, gilt `DefaultAction` nur für VCM-Sitzungen des angegebenen <Zusammenführungstyps>: Basis neu bestimmen (Rebase), Heraufstufen (Promote) oder Replizieren (Replicate).

- ◆ Wird kein `Zusammenführungstyp` angegeben, gilt `DefaultAction` für die aktuelle VCM-Sitzung.

Die Angabe von `DefaultAction` mit einem anderen `<Zusammenführungstyp>` als dem der aktuellen Sitzung ermöglicht, dass Regeln, die von unterschiedlichen VCM-Sitzungen verwendet werden, in einer einzigen Optionsdatei angegeben werden können.

Wird `Elementtyp` angegeben, gilt `DefaultAction` nur für die Elemente des angegebenen `<Elementtyps>`: Änderungsanforderungen, Dateien, Ordner, Anforderungen, Tasks oder Themen. Standardmäßig gilt `DefaultAction` für die Elemente aller Typen.

Mit `<Bedingungen für Übereinstimmung>` werden die Bedingungen festgelegt, die beim Vergleich von den Quell- und/oder Zielelementen erfüllt werden müssen. `<Bedingungen für Übereinstimmung>` besteht aus einer oder mehreren `<Elementbedingungen>` für Quelle und Ziel, von denen jede aus einem `<Bedingungsnamen>` (z. B.: `source.moved`) und einem `<Bedingungswert>` (`True`, `False` oder `Unspecified`) besteht. Der `<Bedingungswert>` ist optional und lautet standardmäßig `True`. `<Bedingungen für Übereinstimmung>` ist die Zusammenfassung aller definierten Bedingungen.

`<Aktion>` legt fest, wie Quelle/Ziel-Elementpaare behandelt werden, deren Unterschiede mit den `<Bedingungen für Übereinstimmung>` übereinstimmen. Die `<Aktion>` definiert lediglich die Standardaktion zum Vergleichen von Elementen; die tatsächliche Aktion kann nach dem Vergleich im StarTeam-Cross-Platform-Client geändert werden.

Nachfolgend sind einige `DefaultAction`-Definitionen als Beispiele aufgeführt:

```
//Wenn ein Quellelement verschoben wurde, aber das Zielelement nicht,
//soll die Verschiebung ignoriert werden.
DefaultAction source.moved target.moved false Ignore

//Wenn eine Datei bei "Rebase" binär ist und in Quelle und Ziel geändert
//wurde, soll die Ziel- durch die Quellversion überschrieben werden.
DefaultAction MergeType Rebase
    items.binaryfile
    source.modified
    target.modified
    Overwrite

//Wenn eine Änderungsanforderung (CR) bei "Promote" in der Quell- und Zielansicht verschoben
//wurde
//(in unterschiedliche Ordner), verschieben Sie das Zielelement als Quellelement in den
//entsprechenden
//Quellordner, aber nur, wenn sich die Änderungsanforderungen im selben Zweig befinden.
DefaultAction MergeType Promote ItemType CR
    source.moved
    target.moved
    items.branched false
    Move
```

`DefaultAction` wird bei Vergleichssitzungen ignoriert.

DefaultComment

DC

`DefaultComment <Kommentar>`

Der vorgegebene Revisionskommentar, der für neue Element-Revisionen verwendet wird, die in der Zielansicht erstellt werden. Der `<Kommentar>` ist ein Textstring in freier Form. Alle Whitespace-Zeichen, einschließlich Zeilenumbrüchen (CRs und LFs), Leerzeichen und Tabulatoren, werden pro Vorkommen in ein einzelnes Leerzeichen konvertiert. Standardmäßig wird ein automatisch generierter Kommentar als vorgegebener

Revisionskommentar für neue Element-Revisionen verwendet. Wenn Sie die Verwendung eines vorgegebenen Revisionskommentars deaktivieren möchten, geben Sie für `DefaultComment` keinen Wert an.

`DefaultComment` wird bei Vergleichssitzungen ignoriert.

Exclude

`Exc`

`Exclude <Ordner>`

Schließt die angegebenen Ordner aus dem Quellbereich aus. Nur explizit durch `Include <Dateien>` oder `Include <Ordner>` angegebene Ordner werden ausgeschlossen. Folglich kann `Exclude<Ordner>` dazu verwendet werden, unerwünschte Ordner oder Änderungsanforderungen aus dem Quellbereich zu entfernen.

Beispiel:

```
//Änderungsanforderungen und Dateien aus allen Ordnern unter /a/b/ berücksichtigen
Include /a/b/ +all CRs Files

//Aber Änderungsanforderungen im Ordner /a/b/c/ ausschließen
Exclude /a/b/c/ CRs

//Befindet sich diese Änderungsanforderung im Ordner /a/b/c/, wird sie dennoch
berücksichtigt
Include CR 12345
```

Unabhängig von der Deklarationsreihenfolge werden `Exclude`-Optionen nach `Include`-Optionen verarbeitet.

Include- und Exclude-Semantik

Wenn keine `Include`-Optionen angegeben werden, wird als vorgegebener Umfang der VCM-Sitzung implizit "alle Dateien der Quellansicht" angenommen. Dies entspricht `include /* +all`. Wird mindestens eine `Include`-Option angegeben, beschränkt sich der Umfang explizit auf die Elemente, die durch die `Include`-Anweisungen spezifiziert werden. Sowohl bei der expliziten wie bei der impliziten Angabe des Umfangs werden alle ausgewählten Quellelemente durch die `Exclude`-Optionen ausgeschlossen.

Bei allen `Include`- und `Exclude`-Optionen müssen Objekte (Labels, Dateien, Änderungsanforderungen usw.) in der Quellansicht angegeben werden. Die ausgewählten Typen können beliebig in Singular- oder Pluralform (`RevLabel`, `CR` usw.) angegeben werden, unabhängig davon, ob ein oder mehrere Werte vorhanden sind.

Hinweis: `Exclude`-Optionen werden unabhängig von der Deklarationsreihenfolge immer nach `Include`-Optionen verarbeitet. Dies führt beispielsweise dazu, dass bei Angabe von `Exclude /src/foo/bar/` gefolgt von `Include /src/foo/ +all` der Ordner `/src/foo/bar/` ausgeschlossen wird.

Export

`Exp`

`Export <VCM-Austauschdatei>`

Die `Export`-Option gibt an, dass alle Informationen zur VCM-Sitzung, einschließlich zusammengeführter Ergebnisdateien, kombiniert und in der angegebenen `<VCM-Austauschdatei>` gespeichert werden. Der Name der Austauschdatei hat immer die Dateierweiterung `.vcmx`. Eine VCM-Austauschdatei dient zur Übertragung der gesamten VCM-Sitzung auf einen anderen Computer, wobei auf diesem Computer ein `Import`-Befehl zur Fortsetzung der Sitzung durchgeführt werden kann. (Weitere Informationen hierzu finden Sie in den Ausführungen zum `Import`-Befehl.)

Wenn die `<VCM-Austauschdatei>` keine Pfadinformationen enthält, wird sie im Stammverzeichnis des Benutzers (in Java mit `user.home` festgelegt) gespeichert.

Hinweis: Die Ausführung der `Export`-Option führt immer dazu, dass die VCM-Austauschdatei erstellt wird, selbst wenn die Sitzung nicht gespeichert wird. Weitere Informationen hierzu finden Sie in den Erläuterungen zur `Save`-Option.

FixFloatingChildShares

FFCS

`FixFloatingChildShares [True | False]`

Gibt an, ob in "Rebase"- und "Replicate"-Zusammenführungsoperationen jedes Zielansichtselement, das eine unverankerte Freigabe eines Quellansichtselements ist, durch Anheften "repariert" werden soll. Wenn ein Zielansichtselement eine Freigabe eines unverankerten untergeordneten Elements eines Quellelements ist (was zudem bedeutet, dass das Zielelement nicht verzweigt wurde), werden Unterschiede zwischen dem Quell- und Zielelement bei VCM-Sitzungen nicht erkannt, da Änderungen am Quellelement sofort an das untergeordnete Element weitergegeben werden. Es gehört zu den Empfehlungen für VCM, dass Freigaben untergeordneter Elemente immer angeheftet werden sollen, damit Änderungen in kontrollierter Form von der Quell- an die Zielansicht weitergegeben werden können. Diese Option ermöglicht, dass von VCM gefundene unverankerte untergeordnete Elemente an die übergeordnete Element-Revision angeheftet werden. Diese Option führt aufgrund der zusätzlichen Befehle zum Überprüfen jedes Zielelements während der Vergleichsphase zu Performance-Verlusten.

IgnoreMergePoints

IMP

`IgnoreMergePoints [True] | [False]`

Gibt an, ob Zusammenführungspunkte während der Vergleichsphase ignoriert werden sollen. Wenn `True`, wird für Elemente mit Konflikten bei der Zusammenführung anstelle der Quellrevision des letzten Zusammenführungspunkts der Verzweigungspunkt als gemeinsamer Vorgänger verwendet.

Include

Inc

`Include {<Änderungsanforderungen> | <Dateien> | <Ordner> | <Prozesselemente> | <Anforderungen> | <Revisions-Labels | <Tasks> | <Themen> }`

Definiert die angegebenen Elemente als Quellbereich. Die `Include`-Option kann mehrfach angegeben werden. Alle aufgeführten Elemente werden berücksichtigt. Mit einer `Include`-Option kann nur ein Elementtyp (Revisions-Labels, Änderungsanforderungen usw.) angegeben werden. Das Auswahl Schlüsselwort, das für Dateien und Ordner optional ist, kann in Singular- oder Pluralform angegeben werden, beispielsweise `ProcessItem` oder `ProcessItems`.

Beispiele:

```
Include CRs ALL
Include /src/com/*.java +all *.jar +2 *.jpx Buildnumber.h
Include Folders /docs/api/ +all
Include ProcessItem CR 451
Include Reqs 4515 4516
Include RevLabel "Beta Fix 12.413"
Include Topic 14512
Include Task 413
```

LockMergeConflicts

LMC

```
LockMergeConflicts {None | Source | Target | Both}
```

Gibt an, dass Elemente mit ungelösten Konflikten in der Quellansicht, der Zielansicht oder in beiden Ansichten exklusiv gesperrt werden. Sperren werden in der Vergleichsphase angelegt. Die Vorgabe ist `None`. Dies bedeutet, dass Elemente mit ungelösten Konflikten nicht gesperrt werden. Beachten Sie, dass Sperren nur auf Quell- und/ oder Zielelemente angewendet werden, für die Unterschiede gefunden wurden. Sperren werden nicht auf Elemente angewendet, bei deren Vergleich keine Unterschiede gefunden wurden. Beachten Sie zudem, dass diese Option weder von der Projektoption **Exklusive Sperre beim Einchecken der Dateien verlangen** noch von der Client-Workstation-Option **Dateien beim Auschecken exklusiv sperren** betroffen ist. Diese Optionen werden ordnungsgemäß von der VCM-Engine behandelt. `LockMergeConflicts` wird bei Vergleichssitzungen ignoriert.

ManualMergeFiles

MMF

```
ManualMergeFiles [True | False]
```

Wenn `True`, wird für jedes gefundene Quelle/Ziel-Dateipaar im Zustand der Inhaltszusammenführung das für die Workstation konfigurierte Tool zur Dateizusammenführung gestartet.

Die Option `ManualMergeFiles` kann in Verbindung mit `AutoMergeFiles` verwendet werden:

- ◆ Wenn ein Zusammenführungskonflikt erkannt und `AutoMergeFiles` angefordert wird, wird zuerst eine automatische Zusammenführung versucht.
- ◆ Bei Lösung des Konflikts wird die zusammengeführte Ergebnisdatei gespeichert. Eine manuelle Zusammenführung ist nicht erforderlich.
- ◆ Ist die automatische Zusammenführung nicht erfolgreich oder wurde `AutoMergeFiles` nicht angefordert, wird die Datei manuell zusammengeführt, wenn `ManualMergeFiles` auf `True` gesetzt ist.

Hinweis: `ManualMergeFiles` wird ignoriert (und eine Warnung wird angezeigt), wenn auf der Workstation kein Tool zur manuellen Zusammenführung konfiguriert wurde. Wenn das Tool zur manuellen Zusammenführung nicht gestartet werden kann oder einen Fehler zurückgibt, bleibt für die betroffene Datei der Konflikt bestehen. `ManualMergeFiles` wird bei Vergleichssitzungen ignoriert.

Match

```
Match [Ordner] *{<Ordnerpfad> to <Ordnerpfad>}
```

Gibt an, dass zu Vergleichszwecken der im ersten `<Ordnerpfad>` angegebene Ordner, der sich in der Quellansicht befinden muss, mit dem im zweiten `<Ordnerpfad>` angegebenen Ordner übereinstimmen soll, der sich in der Zielansicht befinden muss. Die Option `Match` ist manchmal erforderlich, um nicht eindeutige Übereinstimmungen zu verhindern, die auftreten können, wenn es sich bei einer der Ansichten um eine nicht abgeleitete Ansicht handelt. In der Regel wird die Option `Match` nur dazu benötigt, um eine Übereinstimmung zwischen den Stammordnern der Quell- und Zielansicht herzustellen. Es können jedoch Übereinstimmungen zwischen anderen Ordnern hergestellt werden, um andere nicht eindeutige Übereinstimmungen aufzulösen, die von der Vergleichsphase ermittelt werden.

Der `<Ordnerpfad>` für den Ordner der Quell- und der Zielansicht muss mit einem Schrägstrich beginnen und enden ("/").

Gemäß der Konvention wird der Stammordner durch einen einfachen Schrägstrich ("/") dargestellt. Dies bedeutet, dass der Stammordnername in Ordnerpfaden nicht angegeben werden sollte. Wenn der Stammordner beispielsweise "StarDraw" heißt, lautet der Ordnerpfad für den unmittelbar untergeordneten Ordner "Source Code" einfach `"/Source Code/"`.

Beispiele:

```
// Eine Übereinstimmung zwischen den Stammansichtsordnern der Quell- und Zielansicht erzwingen.  
Match / to /
```

```
//Eine Übereinstimmung zwischen dem Quellansichtsordner "/Source Code" //und dem Zielansichtsordner  
"/Modules/Materials/src" erzwingen.  
Match "/Source Code/" to "/Modules/Materials/src/"
```

MergeType

Type

MT

```
MergeType {Compare | Rebase | Promote | Replicate}
```

Gibt an, ob eine Vergleichssitzung (Compare) oder eine Basis-Neubestimmungs-, Heraufstufungs- oder Replizierungs-Zusammenführungssitzung (Rebase, Promote oder Replicate) ausgeführt werden soll. Wenn nur eine Quellansicht ([SourceView](#)) angegeben ist, lautet der Wert für [MergeType](#) standardmäßig "Promote". Wenn nur eine Zielansicht ([TargetView](#)) angegeben ist, lautet der Wert für [MergeType](#) standardmäßig "Rebase". Wenn sowohl [SourceView](#) als auch [TargetView](#) angegeben sind, muss der Wert für [MergeType](#) festgelegt werden. Bei einer Vergleichssitzung können Quell- und Zielansicht gleich sein.

Name

-

```
Name <Name des Änderungspakets>
```

Gibt den Namen des Änderungspakets an, das der VCM-Sitzung zugeordnet ist. Bei Servern, die Änderungspakete unterstützen, wird automatisch ein Name gewählt, wenn durch Speichern oder Festschreiben der Sitzung ein Änderungspaket erstellt wird. Diese Option ermöglicht die Verwendung eines spezifischen Namens anstelle des Standardnamens. Der Name muss sich allerdings von den Namen aller anderen Änderungspakete unterscheiden, die bereits für die Zielansicht gespeichert oder festgeschrieben wurden, anderenfalls schlägt das Speichern bzw. Festschreiben fehl.

Wenn die Option [Name](#) in Verbindung mit dem [Open](#)-Befehl verwendet wird, wird das geöffnete Änderungspaket in den angegebenen Namen umbenannt.

Siehe auch die Optionen [Save](#) und [CommitMerge](#).

PostCommitLabel

PostCL

```
PostCommitLabel <Label>
```

Wenn die VCM-Sitzung festgeschrieben wurde, wird das angegebene *Ansichts-<Label>* in der Zielansicht erstellt, nachdem alle Aktualisierungen vorgenommen wurden. Ein Label gibt die Revisionen aller Zielansichtselemente wieder, die in der Vergleichsphase verwendet und die aufgrund der in der Festschreibungsphase vorgenommenen Modifikationen *geändert* wurden. Dies bedeutet, dass das Label neue Elemente, neue Element-Revisionen und Elementverschiebungen enthält, aber durch die Festschreibung gelöschte Elemente werden vom Label entfernt. Das Label nach der Festschreibung ("post-commit label") ist im Wesentlichen identisch mit der Ansicht vor der Zusammenführung ("pre-merge view"). [PostCommitLabel](#) wird bei Vergleichssitzungen ignoriert.

Standardmäßig wird ein Ansichts-Label nach Festschreibung mit einem vorgegebenen Namen erstellt. Wenn das Ansichts-Label nach Festschreibung deaktiviert werden soll, geben Sie `PostCommitLabel` mit einem leeren Wert an (d. h. " ").

PostCommitRevLabel

PostRL

`PostCommitRevLabel <Label>`

Wenn die VCM-Sitzung festgeschrieben wird, wird das angegebene *Revisions-`<Label>`* in der Zielansicht erstellt und mit Ausnahme gelöschter Elemente werden alle Elemente, die in der VCM-Sitzung geändert wurden, an das Label angehängt. Folglich enthält das Label Elemente, die in der VCM-Sitzung hinzugefügt, verschoben, neu angeheftet oder auf andere Art aktualisiert wurden (Löschvorgänge ausgenommen). `PostCommitRevLabel` wird bei Vergleichssitzungen ignoriert.

Standardmäßig wird kein Revisions-Label nach Festschreibung erstellt.

PreCommitLabel

PreCL

`PreCommitLabel <Label>`

Das angegebene Ansichts-`<Label>` wird in der Zielansicht erstellt und gibt den Snapshot wieder, der in der Vergleichsphase verwendet wird. Das Label reflektiert die Revisionen aller Zielelemente, die in der Vergleichsphase verwendet werden. `PreCommitLabel` wird bei Vergleichssitzungen ignoriert.

Standardmäßig wird kein Ansichts-Label vor Festschreibung erstellt.

PreCommitRevLabel

PreRL

`PreCommitRevLabel <Label>`

Wenn die VCM-Sitzung festgeschrieben wird, wird das angegebene *Revisions-`<Label>`* in der Zielansicht erstellt und alle Zielansichtselemente, die nicht ignoriert wurden, werden dem Label in ihrem "vorherigen" Zustand beigefügt. Dies bedeutet, dass Zielansichtselemente, die von der Sitzung geändert werden, dem Revisions-Label vor der Änderung beigefügt werden. Weiterhin bedeutet dies, dass zur Zielansicht hinzuzufügende Elemente (beispielsweise gemeinsam genutzte Elemente) **nicht beigefügt**, aber zu löschende Elemente **beigefügt** werden. `PreCommitRevLabel` wird bei Vergleichssitzungen ignoriert.

Standardmäßig wird kein Revisions-Label vor Festschreibung erstellt.

PreventDuplicateFileNames

PDF

`PreventDuplicateFileNames [True | False]`

`True` bedeutet, dass die Freigabe einer neuen Datei in der Zielansicht nicht zulässig ist, wenn dadurch zwei Dateien mit identischen Namen in demselben Ordner entstünden.

Project

Pro

`Project <Projekt>`

Gibt das Projekt an, das in der VCM-Sitzung verwendet werden soll. Diese Option ist erforderlich. Die Quell- und Zielansicht müssen zum selben `<Projekt>` gehören. Bei Projektnamen muss die Groß-/Kleinschreibung nicht beachtet werden.

ReportDiffs

RD

`ReportDiffs [True | False]`

Wenn `True`, wird ein Bericht generiert, der die in der Vergleichsphase gefundenen Elementunterschiede auflistet. Der Unterschiedsbericht wird im Stammverzeichnis des Benutzers (in Java mit `user.home` festgelegt) mit folgendem Titel generiert:

```
VCMDiffReport-JJJJ-MM-TT_hh-mm-ss.html
```

wobei `JJJJ-MM-TT` und `hh-mm-ss` das aktuelle Datum und die aktuelle Uhrzeit der jeweiligen Zeitzone sind.

ReportUpdates

RU

`ReportUpdates [True | False]`

Wenn `True`, wird ein Bericht generiert, der die in der Festschreibungsphase vorgenommenen Änderungen an der Zielansicht auflistet. Der Aktualisierungsbericht wird im Stammverzeichnis des Benutzers (in Java mit `user.home` festgelegt) mit folgendem Titel generiert:

```
VCUpdateReport-JJJJ-MM-TT_hh-mm-ss.html
```

wobei `JJJJ-MM-TT` und `hh-mm-ss` das aktuelle Datum und die aktuelle Uhrzeit der jeweiligen Zeitzone sind.

`ReportUpdates` wird bei Vergleichssitzungen ignoriert.

Save

`Save [<VCM-Sitzungsdatei>]`

Gibt an, dass die VCM-Sitzung gespeichert werden soll. Standardmäßig werden nicht festgeschriebene VCM-Sitzungen automatisch in einer VCM-Sitzungsdatei (`.vcms`) mit einem Standardnamen gespeichert. Der Name setzt sich wie folgt zusammen:

```
<Benutzerstammverzeichnis>/VCMSession-JJJJ-MM-TT_hh-mm-ss.vcms
```

wobei `JJJJ-MM-TT_hh-mm-ss` das Datum und die Uhrzeit sind, zu der die Sitzung gespeichert wurde. Der Ordner `<user home>` ist das Stammverzeichnis des Benutzers.

Wenn bei der Option `Save` der Name einer `<VCM-Sitzungsdatei>` angegeben wird, wird die nicht festgeschriebene Sitzung mit dem angegebenen anstelle des Standardnamens gespeichert. Falls erforderlich wird dem Namen die Dateierweiterung `.vcms` hinzugefügt. Falls der angegebene Dateiname keine Pfadinformationen enthält, wird die Sitzungsdatei im Ordner `<user.home>` gespeichert.

Eine `.vcms`-Datei enthält Metadaten zur VCM-Sitzung, aber nicht den Inhalt der zusammengeführten Dateien. (Der Inhalt zusammengeführter Dateien wird in einem benutzerbezogenen temporären Ordner gespeichert, der von Elementen der Sitzungsdatei referenziert werden kann.) Somit kann eine `.vcms`-Datei nur zum Fortsetzen der VCM-Sitzung auf derselben Workstation verwendet werden. (Weitere Informationen hierzu finden Sie beim `Resume`-Befehl.)

Wird die Option `Save` ohne Dateinamen angegeben, wird versucht, die nicht festgeschriebene VCM-Sitzung als aktives Änderungspaket in der Zielansicht zu speichern. Das Änderungspaket wird unter dem Standard- bzw. vom Benutzer angegebenen Namen gespeichert (siehe die Option `Name`). Eine als Änderungspaket gespeicherte VCM-Sitzung kann zu einem späteren Zeitpunkt mithilfe der Option `Open` auf jeder Workstation fortgesetzt werden. Wenn der Server jedoch keine Änderungspakete unterstützt oder das serverseitige Speichern fehlschlägt, wird die Sitzung stattdessen in einer `.vcms`-Datei unter einem Standarddateinamen wie oben beschrieben gespeichert.

Die Option `Save` wird bei einer erfolgreichen Festschreibevorgang ignoriert. Sofern der Server Änderungspakete unterstützt, erstellt die festgeschriebene Sitzung ein "Committed"-Änderungspaket unter Verwendung des Standard- bzw. vom Benutzer angegebenen Namens (siehe die Option `Name`). Falls eine `.vcms`-Datei bereits vorher erstellt wurde, wird sie zusammen mit allen von der VCM-Sitzung erstellten, zusammengeführten Ergebnisdateien gelöscht.

Beachten Sie auch die `Export`-Option.

SourceLabel

`SrcLabel`

SL

`SourceLabel <Label>`

Fordert an, dass die Quellansicht ab einem bestimmten Ansichts-Label verwendet werden soll. Bei Label-Namen muss die Groß-/Kleinschreibung nicht beachtet werden. Es kann nur eine der Optionen `SourceLabel`, `SourceState` und `SourceTime` angegeben werden. Wenn keine dieser Optionen angegeben wird, wird die Option `SourceTime Now` implizit verwendet.

SourceState

`SrcState`

SS

`SourceState <Status>`

Fordert an, dass die Quellansicht ab einem bestimmten Ansichts-Heraufstufungsstatus verwendet werden soll. Bei Heraufstufungsstatusnamen muss die Groß-/Kleinschreibung nicht berücksichtigt werden. Es kann nur eine der Optionen `SourceLabel`, `SourceState` und `SourceTime` angegeben werden. Wenn keine dieser Optionen angegeben wird, wird die Option `SourceTime Now` implizit verwendet.

SourceTime

`SrcTime`

ST

`SourceTime {<Zeitstempel> | Now}`

Fordert an, dass die Quellansicht ab einem bestimmten Zeitstempel verwendet werden soll. Das Schlüsselwort `Now` führt dazu, dass ein Snapshot der aktuellen Zeit als Zeitstempel der Konfiguration verwendet wird. Es kann nur eine der Optionen `SourceLabel`, `SourceState` und `SourceTime` angegeben werden. Wenn keine dieser Optionen angegeben wird, wird die Option `SourceTime Now` implizit verwendet.

SourceView

`Source`

SV

`SourceView <Ansicht>`

Gibt die Quellansicht an, die in der VCM-Sitzung verwendet werden soll. Wenn mehr als eine Ansicht innerhalb des Projekts denselben `<Ansichts>`-Namen hat, kann ein "Pfad zur Ansicht" angegeben werden (beispielsweise

`MainView/ChildView/GrandchildView`), wobei Schrägstriche als Trennzeichen verwendet werden. Enthält ein Ansichtsname Schrägstriche, muss er zwischen Anführungszeichen gesetzt werden.

`SourceView` ist optional für Rebase-Zusammenführungen. Wenn angegeben, muss sie das übergeordnete Objekt der Zielansicht sein.

Hinweis: Bei Ansichtsnamen muss die Groß-/Kleinschreibung nicht beachtet werden.

TargetLabel

`TgtLabel`

TL

`TargetLabel <Label>`

Fordert an, dass die Zielansicht ab einem bestimmten Ansichts-Label verwendet werden soll. `TargetLabel` kann nur für Vergleichssitzungen verwendet werden. Bei Label-Namen muss die Groß-/Kleinschreibung nicht beachtet werden. Es kann nur eine der Optionen `TargetLabel`, `TargetState` und `TargetTime` angegeben werden. Wenn keine dieser Optionen angegeben wird, wird die Option `TargetTime Now` implizit verwendet.

TargetState

`TgtState`

TS

`TargetState <Status>`

Fordert an, dass die Zielansicht ab einem bestimmten Ansichts-Heraufstufungsstatus verwendet werden soll. `TargetState` kann nur für Vergleichssitzungen verwendet werden. Bei Heraufstufungsstatusnamen muss die Groß-/Kleinschreibung nicht berücksichtigt werden. Es kann nur eine der Optionen `TargetLabel`, `TargetState` und `TargetTime` angegeben werden. Wenn keine dieser Optionen angegeben wird, wird die Option `TargetTime Now` implizit verwendet.

TargetTime

`TgtTime`

TT

`TargetTime {<Zeitstempel> | Now}`

Fordert an, dass die Zielansicht ab einem bestimmten Zeitstempel verwendet werden soll. `TargetTime` kann nur für Vergleichssitzungen verwendet werden. Das Schlüsselwort `Now` führt dazu, dass ein Snapshot der aktuellen Zeit als Zeitstempel der Konfiguration verwendet wird. Es kann nur eine der Optionen `TargetLabel`, `TargetState` und `TargetTime` angegeben werden. Wenn keine dieser Optionen angegeben wird, wird die Option `TargetTime Now` implizit verwendet.

TargetView

`Target`

TV

`TargetView <Ansicht>`

Gibt die Zielansicht an, die in der VCM-Sitzung verwendet werden soll. Wenn mehr als eine Ansicht innerhalb des Projekts denselben `<Ansichts>`-Namen hat, kann ein "Pfad zur Ansicht" angegeben werden (beispielsweise

`MainView/ChildView/GrandchildView`), wobei Schrägstriche als Trennzeichen verwendet werden. Enthält der Ansichtsname Schrägstriche, muss er zwischen Anführungszeichen gesetzt werden.

`TargetView` ist optional für Promote-Zusammenführungen. Wenn angegeben, muss sie das übergeordnete Objekt der Quellansicht sein. Bei Vergleichssitzungen können Quell- und Zielansicht gleich sein.

Hinweis: Bei Ansichtsnamen muss die Groß-/Kleinschreibung nicht beachtet werden.

Optionen für wieder aufgenommene Sitzungen

Damit dieselbe Optionsdatei für einen `Resume`-Befehl verwendet werden kann, können alle für neue Sitzungen zulässigen Optionen auch für wieder aufgenommene Sitzungen angegeben werden. Allerdings werden die meisten Optionen, falls angegeben, ignoriert, da sie nicht geändert werden können, sobald die Sitzung gestartet wurde. Davon ausgenommen sind die Optionen, die nachfolgend besonders hervorgehoben werden:

- ◆ **Verbindungsoptionen:** Da Verbindungsinformationen (Serveradresse und Port, Benutzer-ID und Passwort) in der VCM-Sitzungsdatei nicht persistent sind, müssen sie für wieder aufgenommene Sitzungen erneut angegeben werden. Allerdings schlägt eine wieder aufgenommene Sitzung fehl, wenn keine Verbindung zu demselben StarTeam-Server hergestellt wird oder wenn es sich um einen anderen Benutzer handelt, der nicht die erforderlichen Berechtigungen für die Ansichten oder Elemente besitzt, die in der VCM-Sitzung verwendet werden.
- ◆ **`CommitMerge`:** Diese Option ist in einer wieder aufgenommenen Sitzung üblicherweise `True`. Dadurch kann die ursprüngliche Ausführung des VCM-Dienstprogramms nur zum Vergleichen und eine zweite Ausführung des VCM-Dienstprogramms zur Festschreibung verwendet werden.
- ◆ **`ReportDiffs`:** Diese Option kann in einer wieder aufgenommenen Sitzung angegeben werden. Wenn "true", wird vor der Festschreibungsphase, falls vorhanden, ein Unterschiedsbericht erstellt.
- ◆ **`ReportUpdates`:** Diese Option kann in einer wieder aufgenommenen Sitzung angegeben werden. Wenn "true" und wenn die Festschreibungsphase erfolgreich verlaufen ist, werden alle Änderungen an der Zielansicht in einem Bericht zusammengefasst.
- ◆ **`CheckoutPreview`:** Wenn in der ursprünglichen VCM-Sitzung `CheckoutPreview` angegeben wurde, wird in der wieder aufgenommenen Sitzung der "merge preview"-Auscheckvorgang mit denselben Optionen wie zuvor ausgeführt. Wird jedoch in der wieder aufgenommenen Sitzung `CheckoutPreview` angegeben, hat sie Vorrang vor der ursprünglichen Option. Gegebenenfalls werden in der wieder aufgenommenen Sitzung Dateien gemäß der neuen Einstellungen ausgecheckt.
- ◆ **`Description`:** Wenn angegeben, hat diese Option Vorrang vor dem Standard- oder zuvor angegebenen Beschreibungstext des Änderungspakets. Der neue Beschreibungstext wird für die neue Änderungspaketrevision verwendet, die erstellt wird, wenn die VCM-Sitzung gespeichert oder festgeschrieben wird.
- ◆ **`ManualMergeFiles`:** Wenn in der ursprünglichen VCM-Sitzung `ManualMergeFiles` angegeben wurde und die Sitzung mit nicht aufgelösten Dateizusammenführungskonflikten gespeichert wurde, werden die Dateien üblicherweise erneut manuell zusammengeführt, wenn die Sitzung wieder aufgenommen wird. Wenn jedoch in der ursprünglichen VCM-Sitzung `ManualMergeFiles` nicht angegeben wurde, kann in der wieder aufgenommenen Sitzung die Option mit `True` angegeben werden, um eine manuelle Zusammenführung zu starten. Alternativ kann sie in der wieder aufgenommenen Sitzung mit `False` angegeben werden, um eine manuelle Zusammenführung zu verhindern.
- ◆ **`PostCommitLabel`, `PostCommitRevLabel`, `PreCommitLabel` und `PreCommitRevLabel`:** Wird eine dieser Label-Optionen in einer wieder aufgenommenen Sitzung angegeben, hat sie Vorrang vor dem vorherigen Wert des entsprechenden Labels. Wenn eine Label-Option auf ein Leerzeichen (" ") gesetzt wird, wird die entsprechende Label-Option deaktiviert und beim Festschreiben nicht erstellt.

Verwandte Referenz

[Übersicht über das VCM-Befehlszeilendienstprogramm \(VCMUtility\)](#)

[VCMUtility-Befehle](#)

[VCMUtility-Verbindungsoptionen](#)

[Weitere Optionen des VCM-Dienstprogramms](#)

[Beispiele für das VCM-Dienstprogramm](#)

[Befehlsreferenz](#)

Weitere Optionen des VCM-Dienstprogramms

In diesem Abschnitt werden die verschiedenen Optionen von `VCMUtility` definiert, die in Sitzungen zum Vergleichen und Zusammenführen von Ansichten nicht gespeichert werden.

NetMon

NM

`NetMon [True | False]`

Aktiviert die SDK-Funktion "Net Monitor". Jeder Befehl, den das `VCMUtility` an den StarTeam-Server absetzt, wird im Konsolenfenster protokolliert (jedoch nicht in der `VCMUtility`-Protokolldatei).

Zeit

T

`Time [True | False]`

Für jede Phase der ablaufenden VCM-Sitzung werden Zeitangaben angezeigt. Informationen zum zeitlichen Ablauf werden sowohl im Konsolenfenster als auch in der `VCMUtility`-Protokolldatei erfasst.

Verbose

vb

V

`Verbose [True | False]`

Während der Ausführung werden zusätzliche Informationen zur Diagnose und zum Fortschritt auf der Konsole (Standardausgabe) angezeigt und in die `VCMUtility`-Protokolldatei geschrieben.

Verwandte Referenz

- [Übersicht über das VCM-Befehlszeilendienstprogramm \(VCMUtility\)](#)
- [VCMUtility-Befehle](#)
- [VCMUtility-Verbindungsoptionen](#)
- [Sitzungsoptionen von VCMUtility](#)
- [Beispiele für das VCM-Dienstprogramm](#)
- [Befehlsreferenz](#)

Beispiele für das VCM-Dienstprogramm

Dieses Thema liefert Beispiele zur Verwendung von `VCMUtility` für verschiedene Zusammenführungstypen.

'Hello World'-Äquivalent einer Neubestimmung der Basis

Nachfolgend finden Sie die Optionen für das "Hello World"-Äquivalent einer Neubestimmung der Basis (Rebase) von `VCMUtility`:

```
Type      Rebase
Project    Hello
Target     World
```

Basis automatisch neu bestimmen (Automatic Rebase)

Mit der Optionsdatei wird dieselbe Neubestimmung der Basis ausgeführt wie im vorherigen Beispiel, aber die Daten werden, falls möglich, festgeschrieben und es wird ein ausführlicher Bericht zu den Ergebnissen erstellt:

```
Type              Rebase
Project            Hello
Target             World
CommitMerge        True
LockMergeConflicts Both

// Alle diese Optionen werden auf 'True' gesetzt:
AutoMergeFiles
BreakLocks
ReportDiffs
ReportUpdates
```

Sowohl der Inhalt als auch die Eigenschaften aller Dateien werden automatisch zusammengeführt. Dateien mit Unterschieden, die nicht aufgelöst werden können, werden in der Quell- und Zielansicht gesperrt. Bestehende Sperren werden, falls möglich, aufgehoben. Wenn keine ungelösten Konflikte gefunden werden, wird die Sitzung festgeschrieben. Details sowohl der Vergleichsphase (Unterschiede) als auch der Festschreibungsphase (Aktualisierungen) werden protokolliert. Verläuft die Festschreibung erfolgreich, werden alle temporären Dateien der VCM-Sitzung gelöscht.

Per Ansichts-Label heraufstufen: Nur vergleichen

Mit den nachfolgenden Optionen wird auf Vergleichsbasis eine Heraufstufung von Dateien und Änderungsanforderungen ab einem bestimmten Ansichts-Label durchgeführt, wobei die Sitzung in einer Sitzungsdatei gespeichert wird:

```
// Verbindungseinstellungen
Server      MyUserid@ProdServer:4000
PwdFile     MyPassword.txt

// Zusammenführungstyp und Ansichtskonfiguration
Type        Promote
Project     StarDraw
Source      "Beta Release"
SrcLabel    Build-4.0_142
```

```
// Alle Dateien und Änderungsanforderungen als Quellelemente auswählen
include          /* +all
include          / +all CRs

// Nur vergleichen, Bericht erstellen und in einer bestimmten Sitzungsdatei speichern
CommitMerge     False
save            Build-4.0_142-Promote
ReportDiffs

// Verschiedene Optionen
AutoMergeFiles      True
AutoMergeProperties False // als Konflikte belassen und manuell zusammenführen
LockMergeConflicts  Target
```

Per Ansichts-Label heraufstufen: Zusammenführung

Über die Befehlszeile des VCM-Dienstprogramms wird die im vorherigen Beispiel gespeicherte Sitzung fortgesetzt und festgeschrieben, sofern keine neuen Konflikte aufgetreten sind.

```
VCMUtility -resume Build-4.0_142-Promote -CommitMerge -ReportUpdates
```

Verwandte Referenz

[Übersicht über das VCM-Befehlszeilendienstprogramm \(VCMUtility\)](#)

[VCMUtility-Befehle](#)

[VCMUtility-Verbindungsoptionen](#)

[Sitzungsoptionen von VCMUtility](#)

[Weitere Optionen des VCM-Dienstprogramms](#)

[Befehlsreferenz](#)

Befehlsreferenz

VCMUtility-Befehlszeilensyntax: VCMUtility [<Optionsdatei>] [*<Option>]

In der <Optionsdatei> muss jede <Option> in Spalte 1 beginnen, kann aber auf nachfolgende Zeilen ausgedehnt werden, wenn diese mit einem Leer- oder Tabulatorzeichen beginnen. Bei der Eingabe von Optionen in die Befehlszeile muss jeder <Option> ein Bindestrich (-) vorangestellt werden.

Optionen

In der nachfolgenden Tabelle sind alle Befehlszeilenoptionen des VCMUtility und ihre Syntax aufgelistet.

Befehl/Option

```
<Option>
  <Befehl> | <Verbindungsoption> | <Sitzungsoption> | <Verschiedene Optionen>
```

```
<Befehl>
{{Help | H | ?} [<Hilfethema>]} |
{Delete <VCM-Sitzungsdatei>} |
{Import <VCM Archivdatei>} |
{Open <Name des Änderungspakets>} |
{Replay <Name des Änderungspakets>} |
{Resume <VCM-Sitzungsdatei>}
```

```
<Verbindungsoption>
{{AutoLogon | AL} [True | False]} |
{{Encryption | Encrypt | En} {None | RC4 | RC2_ECB | RC2_CBC | RC2_CFB}} |
{{PwdFile | PF} <Dateiname>} |
{{Server | S} [<Benutzer>[:<Passwort>]@<Host>[:<Port>]} |
{{UseCA | UCA} {<Host>:<Port> | AutoLocate}} |
{{UseServerProfile | USP} [True | False]}
```

```
<Sitzungsoption>
{{AutoMergeFiles | AMF} [True | False]} |
{{AutoMergeProperties | AMP} [True | False]} |
{{BreakLocks | BL} [True | False]} |
{{CaseSensitiveFileNames | CSF} [True | False]} |
{{CheckoutPreview | check-out | CP} <Dateien> [<Auscheckoptionen>]} |
{{CommitMerge | Commit | CM} [True | False]} |
{{DefaultAction | DA} [MergeType <Zusammenführungstyp>] [ItemType <Elementtyp>]
<Bedingungen für Übereinstimmung> <Aktion>} |
{{DefaultComment | DC} <Kommentar>} |
{{Description|} <Beschreibung>} |
{{Exclude | Exc} <Ordner>} |
{{Export | Exp} <VCM Archivdatei>} |
{{FixFloatingChildShares | True | False}} |
{{IgnoreMergePoints | IMP} [True | False]} |
{{Include | Inc} {<Änderungsanforderungen> | <Dateien> | <Ordner> |
<Prozesselemente> |
  <Anforderungen> | <Revisions-Labels> | <Tasks> | <Themen>}} |
{{LockMergeConflicts | LMC} {None | Source | Target | Both}} |
{{ManualMergeFiles | MMF} [True | False]} |
{{Match [Folder] *{<Ordnerpfad> to <Ordnerpfad>}} |
{{MergeType | Type | MT} {Compare | Rebase | Promote | Replicate}} |
{{Name | Na} <Name des Änderungspakets>} |
```

```

{{PostCommitLabel | PostCL} <Label>} |
{{PostCommitRevLabel | PostRL} <Label>} |
{{PreCommitLabel | PreCL} <Label>} |
{{PreCommitRevLabel | PreRL} <Label>} |
{{PreventDuplicateFileNames | PDF} [True | False]} |
{{Project | Pro} <Projekt>} |
{{ReportDiffs | RD} [True | False]} |
{{ReportUpdates | RU} [True | False]} |
{Save [<VCM-Sitzungsdatei>]} |
{{SourceLabel | SrcLabel | SL} <Label>} |
{{SourceState | SrcState | SS} <Status>} |
{{SourceTime | SrcTime | ST} {<Zeitstempel> | Now}} |
{{SourceView | Source | SV} <Ansicht>} |
{{TargetLabel | TgtLabel | TL} <Label>} |
{{TargetState | TgtState | TS} <Status>} |
{{TargetView | Target | TV} <Ansicht>} |

```

<Verschiedene Optionen>

```

{{NetMon | NM} [True | False]} |
{{Time | T} [True | False]} |
{{Verbose | Vb | V} [True | False]} |

```

Weitere Syntaxelemente

In der nachfolgenden Tabelle sind weitere Syntaxelemente aufgeführt:

Weitere Syntaxelemente

<Aktion>

```

Delete | DeleteAndReverseShare | Fail | Ignore | Merge | Move | MoveAndMerge |
MoveAndRepin | NeedsReview | Overwrite | Repin | RepinAndMove |
ReverseShare | Share

```

<Änderungsanforderungen>

```

{CR | CRs | ChangeRequests} {ALL | *<ÄÄ-Nr.>}

```

<Name des Änderungspakets>

```

{Ein Name, der aus einem oder mehreren Zeichen besteht}

```

<Auscheckoptionen>

```

[+cwf] [+eol {on | off | cr | lf}] [+filter {CGIMOU}] [+o] [+ro]
[+rp <Arbeitsordnerpfad>]

```

<Bedingungsname>

```

items.binaryfile | items.branched | items.samecontent | source.childshare |
source.deleted | source.floating | source.modified | source.moved |
source.present | source.rootbranch | target.childshare | target.deleted |
target.floating | target.modified | target.moved | target.present |
target.parentdeleted | target.rootbranch

```

```
<Bedingungswert>
  True | False | Unspecified
```

```
<Dateien>
  [File | Files] {ALL | *{<Dateinamensmuster>} [+<Ordnererebenen>]}
```

```
<Ordnerpfad>
{Ein Schrägstrich gefolgt von einer optionalen Abfolge von Ordnernamen (jeweils mit einem Schrägstrich am Ende)}
```

```
<Ordner>
  [Folder | Folders] {ALL | *{<Ordnerpfad>} [+<Ordnererebenen>] *{<Elementtyp>}}
```

```
<Elementbedingung>
  <Bedingungsname> [<Bedingungswert>]
```

```
<Elementtyp>
  {ChangeRequest | CR | ChangeRequests | CRs} |
  {File | Files} |
  {Folder | Folders} |
  {Requirement | Req | Requirements | Reqs}
  {Task | Tasks}
  {Topic | Topics}
```

```
<Bedingungen für Übereinstimmung>
  *{<Elementbedingung>}
```

```
<Prozesselemente>
  ProcessItems *{[View <Ansicht>] CR <ÄA-Nr.> |
                [View <Ansicht>] Req <Anf.-Nr.> |
                [View <Ansicht>] Task <Task-Nr.>}
```

```
<Anforderungen>
  {Requirement | Req | Requirements | Reqs} {ALL | *{<Anf.-Nr.>}}
```

```
<Revisions-Labels>
  RevLabels *{<Label>}
```

```
<Task>
  {Task | Tasks} {ALL | *{<Task-Nr.>}}
```

```
<Zeitstempel>
  Beispielformate:
  "3/11/06 1:32 PM"
  "Mar 11, 2006 1:32:38 PM"
  "March 11, 2006 1:32:38 PM PST"
  "Saturday, March 11, 2006 1:32:38 PM PST"
```

```
<Themen>  
{Topic | Topics} {ALL | *<Themenr.>}
```

```
<VCM-Austauschdatei>  
Eine .vcms-Dateiname}
```

```
<VCM-Sitzungsdatei>  
Eine .vcms-Dateiname}
```

Verwandte Referenz

[Übersicht über das VCM-Befehlszeilendienstprogramm \(VCMUtility\)](#)

[VCMUtility-Befehle](#)

[VCMUtility-Verbindungsoptionen](#)

[Sitzungsoptionen von VCMUtility](#)

[Weitere Optionen des VCM-Dienstprogramms](#)

[Beispiele für das VCM-Dienstprogramm](#)

[Syntax der zusammengesetzten Optionen des VCM-Dienstprogramms](#)

[<Aktion>](#)

[<Änderungsanforderungen>](#)

[<Auscheckoptionen>](#)

[<Dateien>](#)

[<Ordner>](#)

[<Prozesselement>](#)

[<Anforderungen>](#)

[<Revisions-Labels>](#)

[<Tasks>](#)

[<Zeitstempel>](#)

[<Themen>](#)

Syntax der zusammengesetzten Optionen des VCM-Dienstprogramms

Folgende zusammengesetzte Syntaxelemente werden in den `VCMUtility`-Optionen verwendet.

In diesem Abschnitt

[<Aktion>](#)

Beschreibt die Syntax der zusammengesetzten `VCMUtility`-Option `<Aktion>`.

[<Auscheckoptionen>](#)

Beschreibt die Syntax der zusammengesetzten `VCMUtility`-Option `<Auscheckoptionen>`.

[<Änderungsanforderungen>](#)

Beschreibt die Syntax der zusammengesetzten `VCMUtility`-Option `<Änderungsanforderungen>`.

[<Dateien>](#)

Beschreibt die Syntax der zusammengesetzten `VCMUtility`-Option `<Dateien>`.

[<Ordner>](#)

Beschreibt die Syntax der zusammengesetzten `VCMUtility`-Option `<Ordner>`.

[<Elementtyp>](#)

Beschreibt die Syntax der zusammengesetzten `VCMUtility`-Option `<Bedingungen für Übereinstimmung>`.

[<Bedingungen für Übereinstimmung>](#)

Beschreibt die Syntax der zusammengesetzten `VCMUtility`-Option `<Bedingungen für Übereinstimmung>`.

[<Prozesselement>](#)

Beschreibt die Syntax der zusammengesetzten `VCMUtility`-Option `<Prozesselement>`.

[<Anforderungen>](#)

Beschreibt die Syntax der zusammengesetzten `VCMUtility`-Option `<Anforderungen>`.

[<Revisions-Labels>](#)

Beschreibt die Syntax der zusammengesetzten `VCMUtility` -Option `<Revisions-Label>`.

[<Tasks>](#)

Beschreibt die Syntax der zusammengesetzten `VCMUtility`-Option `<Tasks>`.

[<Zeitstempel>](#)

Beschreibt die Syntax der zusammengesetzten `VCMUtility`-Option `<Zeitstempel>`.

[<Themen>](#)

Beschreibt die Syntax der zusammengesetzten `VCMUtility`-Option `<Themen>`.

<Aktion>

Gibt die Aktion an, die bei einem Unterschied zwischen einem Quell- und einem Zielelement durchzuführen ist. Folgende <Aktionen> stehen zur Verfügung:

Aktion	Beschreibung
Delete	Löscht das Zielelement.
DeleteAndReverseShare	Entspricht der Aktion <code>Delete</code> gefolgt von der Aktion <code>ReverseShare</code> .
Fail	Synonym für <code>NeedsReview</code> (siehe unten).
Ignore	Keine Aktion.
MarkResolved	Erstellt nur einen Zusammenführungspunkt, der die Quell- und Zielelemente als "resolved" (aufgelöst) kennzeichnet.
Merge	Führt die Quell- und Zielelemente zusammen.
Move	Verschiebt das Zielelement als Quellelement in den entsprechenden Ordner.
MoveAndMerge	Entspricht der Aktion <code>Move</code> gefolgt von der Aktion <code>Merge</code> .
MoveAndOverwrite	Entspricht der Aktion <code>Move</code> gefolgt von der Aktion <code>Overwrite</code> .
MoveAndRepin	Entspricht der Aktion <code>Move</code> gefolgt von der Aktion <code>Repin</code> .
NeedsReview	Erzwingt eine Überprüfung vor dem Festschreiben. D. h., das Festschreiben ist bei Auswahl dieser Aktion nicht zulässig. Elementunterschiede weisen bei dieser Aktion Konflikte auf. Daher muss ihre zugehörige Aktion geändert werden.
Overwrite	Überschreibt das Ziel mit dem Inhalt der Quelle.
Repin	Ändert die Revision, an die das Ziel angeheftet ist, sodass sie dem Quellelement entspricht.
ReverseShare	Verschiebt das Quellelement in die Zielansicht und führt eine umgekehrte Freigabe zur Quellansicht durch.
Share	Gibt das Quellelement in der Zielansicht frei.

Hinweis: Nicht jede <Aktion> ist für jeden Elementunterschied gültig. `Delete` ist beispielsweise nicht gültig, wenn das Zielelement bereits gelöscht wurde.

Verwandte Referenz

[Syntax der zusammengesetzten Optionen des VCM-Dienstprogramms](#)
[<Änderungsanforderungen>](#)
[<Auscheckoptionen>](#)
[<Dateien>](#)
[<Ordner>](#)
[<Bedingungen für Übereinstimmung>](#)
[<Prozesselement>](#)
[<Anforderungen>](#)
[<Revisions-Labels>](#)
[<Tasks>](#)
[<Zeitstempel>](#)
[<Themen>](#)
[Befehlsreferenz](#)

<Auscheckoptionen>

Im folgenden Abschnitt wird die Syntax der zusammengesetzten VCMUtility-Option <Auscheckoptionen> beschrieben.

```
[+cwf] [+eol {on | off | cr | lf | crlf}] [+filter {CGIMOU}] [+o] [+ro] [+rp  
<Arbeitsordnerpfad>]
```

Gibt die nicht standardmäßigen Auscheckoptionen an. Die verfügbaren Auscheckoptionen sind denen des StarTeam-Befehlszeilenprogramms "stcmd" und von "bulk check-out" (BCO) ähnlich mit der Ausnahme, dass Optionsnamen ein Plussymbol "+" vorangestellt werden muss. Die verfügbaren Optionen sind nachfolgend aufgeführt.

+cwf

Fordert die Erstellung von Arbeitsordnern für alle angegebenen Ordner an, selbst wenn bei dieser Ausführung keine Dateien ausgecheckt werden müssen. Es werden nur sichtbare Ordner erstellt.

+eol <EOL-Option>

Fordert die Konvertierung aller Zeilenendezeichen für Textdateien in das angegebene Format an. Die <EOL-Option> **on** verwendet das vom Client konfigurierte EOL-Format, während bei **off** keine EOL-Konvertierung durchgeführt wird; **cr**, **lf** und **crlf** konvertieren das Zeilenendezeichen entsprechend in Carriage-Return, Line-Feed oder Carriage-Return/Line-Feed. Beachten Sie, dass Textdateien mit "festem" EOL-Format immer in das angegebene Format konvertiert werden.

+filter

```
+filter {CGIMOU}
```

Gibt den Status der für das Auschecken zu berücksichtigenden Dateien an: **Current** (Aktuell), **merge** (Zusammenführung), **missing** (Fehlt), **Modified** (Geändert), **Out-of-date** (Veraltet) oder **Unknown** (Unbekannt). Es können mehrere Status-Flags miteinander kombiniert werden. Ist **+filter** nicht angegeben, wird der Standardfilter **IO** (**Missing** [Fehlt] und **Out-of-date** [Veraltet]) verwendet. Wenn Dateien mit dem Status **Merge**, **Merge** oder **Unknown** ohne die Option **+o** angegeben werden, wird für jede dieser Dateien eine Warnung ausgegeben und sie werden nicht ausgecheckt.

+o

Gibt an, dass zusätzlich zu den Dateien mit dem Status **Missing** und **Out-of-date** Dateien mit dem Status **Modified**, **Merge** oder **Unknown** berücksichtigt werden sollen. Zudem werden alle Dateien ohne Warnung überschrieben. Wenn auch **+filter** angegeben wird, werden nur die angegebenen Dateien ausgecheckt.

+ro

Setzt alle Dateien nach dem Auschecken auf schreibgeschützt. Standardmäßig können ausgecheckte Dateien gelesen und geschrieben werden.

+rp

Gibt den Stammarbeitsordner der Zusammenführungsvorschau ("merge preview") an. Dateien werden in untergeordnete Arbeitsordner relativ zum <Arbeitsordnerpfad> ausgecheckt.

Verwandte Referenz

[Syntax der zusammengesetzten Optionen des VCM-Dienstprogramms](#)

[<Aktion>](#)

[<Änderungsanforderungen>](#)

[<Dateien>](#)

[<Ordner>](#)

[<Bedingungen für Übereinstimmung>](#)

[<Prozesselement>](#)

[<Anforderungen>](#)

[<Revisions-Labels>](#)

[<Tasks>](#)

[<Zeitstempel>](#)

[<Themen>](#)

[Befehlsreferenz](#)

<Änderungsanforderungen>

<Änderungsanforderungen> {CR | CRs | ChangeRequests} {ALL | *<CR #>}

Gibt alle Änderungsanforderungen in der Ansicht oder einzelne Änderungsanforderungen nach Änderungsanforderungsnummer an. `CRs` und `ChangeRequests` sind Synonyme; auch die jeweilige Singularform wird angenommen.

Verwandte Referenz

[Syntax der zusammengesetzten Optionen des VCM-Dienstprogramms](#)

[<Aktion>](#)

[<Auscheckoptionen>](#)

[<Dateien>](#)

[<Ordner>](#)

[<Bedingungen für Übereinstimmung>](#)

[<Prozesselement>](#)

[<Anforderungen>](#)

[<Revisions-Labels>](#)

[<Tasks>](#)

[<Zeitstempel>](#)

[<Themen>](#)

[Befehlsreferenz](#)

<Dateien>

```
[File | Files] {ALL | *{<Dateinamensmuster> [+Ordner Ebenen]}}
```

Gibt alle Dateien in der Ansicht oder eine Gruppe von bestimmten Dateien in Form einer Liste und/oder von Mustern an, jeweils optional mit der Angabe der <Ordner Ebenen>. Das Schlüsselwort `File` (oder `Files`) ist optional, sofern nicht das Schlüsselwort `All` verwendet wurde. Ein <Dateinamensmuster> kann ein bestimmter Dateiname (z. B. `foo.java`), ein Dateinamensmuster (z. B. `*.java`) oder ein Dateiname oder Muster mit einem Ordnerpfad (z. B. `/src/com/acme/foo.java`) oder `/src/com/acme/*.java`) sein.

Verwendung

Ordnerpfade müssen mit Schrägstrichen angegeben werden. Ein einzelner Schrägstrich (`/`) ist ein Synonym für den Stammordner. (Konsistent mit anderen StarTeam-Dienstprogrammen; der Name des Stammordners, der in der Regel dem Ansichtsnamen entspricht, sollte in Pfadnamen **nicht** angegeben werden.)

- ◆ Wird ein Dateiname oder Muster ohne Ordnerpfad angegeben, wird als Ordner der im vorherigen <Dateinamensmuster>-Parameter angegebene Ordner verwendet.
- ◆ Wenn der erste <Dateinamensmuster>-Parameter keinen Ordnerpfad enthält, wird der Stammordner verwendet.
- ◆ Falls angegeben, gibt <Ordner Ebenen> die Anzahl der einzuschließenden untergeordneten Ordner Ebenen unterhalb des angegebenen Ordners an. Der Wert kann eine Zahl oder das Schlüsselwort **All** sein.
- ◆ Wenn Dateien oder Muster Leerzeichen enthalten, müssen sie zwischen Anführungszeichen gesetzt werden.

Beispiele

Nachfolgend finden Sie Beispiele für die Verwendung von <Dateien>:

```
// Alle Dateien in der Ansicht
include Files ALL

//foo.java und bar.java im Ordner /src/com/acme
include /src/com/acme/foo.java bar.java

// Alle .java-Dateien im Ordner /src/com/acme and below
include /src/com/acme/*.java +all

// Alle .txt-Dateien im Stammordner, alle .zip-Dateien eine // Ordner Ebene tiefer und eine
bestimmte Readme-Datei
include *.txt *.zip +1 /docs/acme/readme.txt
```

Verwandte Referenz

[Syntax der zusammengesetzten Optionen des VCM-Dienstprogramms](#)

[<Aktion>](#)

[<Änderungsanforderungen>](#)

[<Auscheckoptionen>](#)

[<Ordner>](#)

[<Bedingungen für Übereinstimmung>](#)

[<Prozesselement>](#)

[<Anforderungen>](#)

[<Revisions-Labels>](#)

[<Tasks>](#)

[<Zeitstempel>](#)

[<Themen>](#)

[Befehlsreferenz](#)

<Ordner>

```
[Folder | Folders] {ALL | *{<Ordnerpfad> [+<Ordnererebenen>] *{<Elementtyp>}}
```

Gibt alle Ordner in der Ansicht oder bestimmte Ordnerpfade an. Dabei können die Anzahl der Ordnererebenen und einzelne Elementtypen optional angegeben werden. Das Schlüsselwort `Folder` (oder `Folders`) ist optional, sofern nicht das Schlüsselwort `ALL` verwendet wurde.

Verwendung

Ein gültiger `<Ordnerpfad>` muss mit einem Schrägstrich beginnen und enden (`/src/com/`). Falls angegeben, gibt `<Ordnererebenen>` die Anzahl der einzubeziehenden untergeordneten Ordnererebenen unterhalb des angegebenen Ordners an. Der Wert kann eine Zahl oder das Schlüsselwort `All` sein.

- ◆ Wenn ein Ordnerpfad Leerzeichen enthält, muss er zwischen Anführungszeichen gesetzt werden.
- ◆ Wird kein `<Elementtyp>`-Parameter angegeben, werden in den genannten Ordnern nur Dateien berücksichtigt. Anderenfalls werden alle Elemente der angegebenen Elementtypen berücksichtigt.

Gültige Elementtypen sind `CRs`, `Files`, `Folders`, `Tasks`, `Topics` und `Requirements` (singular oder plural).

Beispiele

Nachfolgend finden Sie Beispiele für die Verwendung von `<Ordner>`:

```
// Alle Ordner in der Ansicht
include folders ALL

// Alle Dateien nur im Ordner /src/com/acme/
include /src/com/acme/

// Alle Dateien und Tasks in /src/ und darunter
include /src/ +all files tasks

// Alle Änderungsanforderungen im Ordner "/triage/" und alle Dateien in "/PR docs/"
// einschließlich zwei Ordnererebenen darunter
include /triage/ CRs "/PR docs/" +2
```

Üblicherweise wird der Stammordner durch einen einfachen Schrägstrich ("/") angegeben. Dies bedeutet, dass der Stammordnername in Ordnerpfaden nicht angegeben werden sollte. Wenn der Stammordner beispielsweise "StarDraw" heißt, lautet der Ordnerpfad für den unmittelbar untergeordneten Ordner "Source Code" einfach `/Source Code/`.

Verwandte Referenz

[Syntax der zusammengesetzten Optionen des VCM-Dienstprogramms](#)

[<Aktion>](#)

[<Änderungsanforderungen>](#)

[<Auscheckoptionen>](#)

[<Dateien>](#)

[<Bedingungen für Übereinstimmung>](#)

[<Prozesselement>](#)

[<Anforderungen>](#)

[<Revisions-Labels>](#)

[<Tasks>](#)

[<Zeitstempel>](#)

[<Themen>](#)

[Befehlsreferenz](#)

<Elementtyp>

<Elementtyp>

Definiert einen Elementtyp. Die zulässigen Werte lauten: `ChangeRequest` (oder `CR`), `File`, `Folder`, `Requirement` (oder `Req`), `Task` und `Topic`. Bei den Namen der Elementtypen muss die Groß-/Kleinschreibung nicht berücksichtigt werden. Zudem können sie im Plural verwendet werden.

Verwandte Referenz

[Syntax der zusammengesetzten Optionen des VCM-Dienstprogramms](#)

[<Aktion>](#)

[<Änderungsanforderungen>](#)

[<Auscheckoptionen>](#)

[<Dateien>](#)

[<Ordner>](#)

[<Prozesselement>](#)

[<Anforderungen>](#)

[<Revisions-Labels>](#)

[<Tasks>](#)

[<Zeitstempel>](#)

[<Themen>](#)

[Befehlsreferenz](#)

<Bedingungen für Übereinstimmung>

*<Elementbedingung>

Definiert Bedingungen für die Elementunterschiede von Quelle und Ziel. <Bedingungen für Übereinstimmung> ist die Zusammenfassung aller definierten <Elementbedingungen>.

<Elementbedingungen> haben die Form:

<Bedingungsname> [<Bedingungswert>]

<Bedingungsname>

Die gültigen <Bedingungsnamen> und ihre Bedeutung sind:

<Bedingungsname>	Bedeutung
<code>items.binaryfile</code>	Gibt an, ob eines der in Frage kommenden Elemente eine Binärdatei ist.
<code>items.branched</code>	Gibt an, ob sich die Quell- und Zielelemente in unterschiedlichen Zweigen des Versionsbaums des Objekts befinden.
<code>items.samecontent</code>	Gibt an, ob die Quell- und Zielelemente über dieselben benutzerdefinierten Eigenschaften und bei Dateien über denselben Dateninhalt verfügen.
<code>source.childshare</code>	Gibt an, ob das Quellelement eine untergeordnete Freigabe des Zielelements ist.
<code>source.deleted</code>	Gibt an, ob das in Frage kommende Element in der Quellansicht gelöscht wird.
<code>source.floating</code>	Gibt an, ob das Quellelement eine unverankerte Konfiguration hat.
<code>source.modified</code>	Gibt an, ob das in Frage kommende Element in der Quellansicht geändert wird.
<code>source.moved</code>	Gibt an, ob das in Frage kommende Element in der Quellansicht verschoben wird.
<code>source.present</code>	Gibt an, ob das in Frage kommende Element in der Quellansicht vorhanden ist.
<code>source.rootbranch</code>	Gibt an, ob das Quellelement die Stammverzweigung seines freigegebenen Baums ist.
<code>target.childshare</code>	Gibt an, ob das Zielelement eine untergeordnete Freigabe des Quellelements ist.
<code>target.deleted</code>	Gibt an, ob das in Frage kommende Element in der Zielansicht gelöscht wird.
<code>target.floating</code>	Gibt an, ob das Zielelement eine unverankerte Konfiguration hat.
<code>target.modified</code>	Gibt an, ob das in Frage kommende Element in der Zielansicht geändert wird.
<code>target.moved</code>	Gibt an, ob das in Frage kommende Element in der Zielansicht verschoben wird.
<code>target.parentdeleted</code>	Gibt an, ob der Ordner des Zielelements gelöscht wurde.
<code>target.present</code>	Gibt an, ob das in Frage kommende Element in der Zielansicht vorhanden ist.
<code>target.rootbranch</code>	Gibt an, ob das Zielelement die Stammverzweigung seines freigegebenen Baums ist.

<Bedingungswert>

Gültige <Bedingungswerte> sind:

<Bedingungswert>	Bedeutung
<code>True</code>	Die Bedingung ist für die entsprechenden Elemente wahr.
<code>False</code>	Die Bedingung ist für die entsprechenden Elemente falsch.

Unspecified

Die Bedingung ist unbekannt oder für die entsprechenden Elemente nicht relevant.

Der `<Bedingungswert>` ist optional und lautet standardmäßig `True`. Bei `<Bedingungen für Übereinstimmung>` sind alle nicht angegebenen Bedingungen anfänglich `Unspecified`.

Eine `<Elementbedingung>` kann als `True` oder `False` definiert werden, damit die entsprechende Bedingung für das Suchen nach Elementunterschieden "aktiviert" wird.

Eine Bedingung kann als `Unspecified` definiert werden, um beispielsweise die Bedingung zu Testzwecken aus den Abgleichkriterien zu entfernen, ohne sie aber aus der Optionsdatei zu löschen.

Hinweis: Einige Bedingungen schließen sich gegenseitig aus: Wenn sie gleichzeitig angegeben werden, können keine Elementunterschiede festgestellt werden. Beispielsweise kann ein Quellelement nicht gleichzeitig vorhanden (`source.present.true`) und gelöscht (`source.deleted.true`) sein.

Verwandte Referenz

[Syntax der zusammengesetzten Optionen des VCM-Dienstprogramms](#)

[<Aktion>](#)

[<Änderungsanforderungen>](#)

[<Auscheckoptionen>](#)

[<Dateien>](#)

[<Ordner>](#)

[<Prozesselement>](#)

[<Anforderungen>](#)

[<Revisions-Labels>](#)

[<Tasks>](#)

[<Zeitstempel>](#)

[<Themen>](#)

[Befehlsreferenz](#)

<Prozesselement>

```
ProcessItems *{[View <Ansicht>] CR <CR-Nr.> | [View <Ansicht>] Req <Anf.-Nr.> | [View <Ansicht>] Task <Task-Nr.>}
```

Gibt die zu berücksichtigenden Prozesselemente (Änderungsanforderungen, Tasks und/oder Anforderungen) an. Bei Angabe eines Prozesselements werden auch die in der Quellansicht mit ihm verbundenen Elemente berücksichtigt. Das Schlüsselwort `ProcessItems` kann auch in der Singularform verwendet werden. Die vollständigen Namen `ChangeRequest` und `Requirement` können anstelle von `CR` und `Req` entsprechend verwendet werden.

Standardmäßig muss sich ein angegebenes Prozesselement in der Quellansicht befinden. Sie können jedoch den optionalen Präfix `View <Ansicht>` zum Auswählen eines Prozesselements verwenden, das sich in einer anderen Ansicht als der Quellansicht befindet. Wenn ein Prozesselement angegeben wird, das sich nicht in der Quellansicht befindet, wird das Prozesselement **nicht** in den Quellbereich übernommen, aber die Elemente, die ihm in der Quellansicht beigelegt wurden, werden übernommen. Es wird die spezifische Revision jedes Quellansichtselements übernommen, das mit dem Prozesselement verknüpft ist.

Beispiele:

```
// CR 451 und ihre verknüpften Elemente in die Quellansicht aufnehmen
include ProcessItem CR 451

// Die Elemente in die Quellansicht aufnehmen, die mit Task 909 verknüpft sind
// Anforderung 518 aufnehmen, beide aus der Ansicht "Triage"
// include ProcessItem View Triage Task 909 View Triage Requirement 518
```

Hinweis: Wenn der Ansichtsname Leerzeichen enthält, muss er zwischen Anführungszeichen gesetzt werden ("`Release 4.3`"). Wenn mehr als eine Ansicht innerhalb des Projekts denselben Ansichtsnamen hat, kann ein "Pfad zur Ansicht" angegeben und dabei Schrägstriche als Trennzeichen verwendet werden ("`Apps/Releases/Release 4.3`").

Verwandte Referenz

[Syntax der zusammengesetzten Optionen des VCM-Dienstprogramms](#)

[<Aktion>](#)

[<Änderungsanforderungen>](#)

[<Auscheckoptionen>](#)

[<Dateien>](#)

[<Ordner>](#)

[<Bedingungen für Übereinstimmung>](#)

[<Anforderungen>](#)

[<Revisions-Labels>](#)

[<Tasks>](#)

[<Zeitstempel>](#)

[<Themen>](#)

[Befehlsreferenz](#)

<Anforderungen>

{Reqs | Requirements} {ALL | *{<Anf.-Nr.>}}

Gibt die einzelnen Anforderungen nach Anforderungsnummer an. `Reqs` und `Requirements` sind Synonyme; auch die jeweilige Singularform wird angenommen.

Verwandte Referenz

[Syntax der zusammengesetzten Optionen des VCM-Dienstprogramms](#)

[<Aktion>](#)

[<Änderungsanforderungen>](#)

[<Auscheckoptionen>](#)

[<Dateien>](#)

[<Ordner>](#)

[<Bedingungen für Übereinstimmung>](#)

[<Prozesselement>](#)

[<Revisions-Labels>](#)

[<Tasks>](#)

[<Zeitstempel>](#)

[<Themen>](#)

[Befehlsreferenz](#)

<Revisions-Labels>

RevLabels *<Label>

Gibt alle Elemente an, die jedem angegebenen Revisions-Label (<Label>) beigefügt sind. Das Schlüsselwort `RevLabels` kann auch in der Singularform verwendet werden. Bei Revisions-Labels muss die Groß-/Kleinschreibung nicht beachtet werden.

Verwandte Referenz

[Syntax der zusammengesetzten Optionen des VCM-Dienstprogramms](#)

[<Aktion>](#)

[<Änderungsanforderungen>](#)

[<Auscheckoptionen>](#)

[<Dateien>](#)

[<Ordner>](#)

[<Bedingungen für Übereinstimmung>](#)

[<Prozesselement>](#)

[<Anforderungen>](#)

[<Tasks>](#)

[<Zeitstempel>](#)

[<Themen>](#)

[Befehlsreferenz](#)

<Tasks>

Tasks {ALL | *{<Task-Nr.>}}

Enthält die einzelnen angegebenen Tasks nach Task-Nummer. Das Schlüsselwort `Tasks` kann auch in der Singularform verwendet werden.

Verwandte Referenz

[Syntax der zusammengesetzten Optionen des VCM-Dienstprogramms](#)

[<Aktion>](#)

[<Änderungsanforderungen>](#)

[<Auscheckoptionen>](#)

[<Dateien>](#)

[<Ordner>](#)

[<Bedingungen für Übereinstimmung>](#)

[<Prozesselement>](#)

[<Anforderungen>](#)

[<Revisions-Labels>](#)

[<Zeitstempel>](#)

[<Themen>](#)

[Befehlsreferenz](#)

<Zeitstempel>

Ein <Zeitstempel> muss einen der in Java gebräuchlichen String-Formate für Datum und Uhrzeit haben.

- ◆ Datumsformate werden anhand der lokalen Datumsformatkonventionen interpretiert (beispielsweise wird `11.03.06` als 11. März 2006 interpretiert).
- ◆ Die Angabe der Sekunden ist optional (`1:32` und `1:32:00` sind beispielsweise identisch).
- ◆ Der `AM/PM`-Indikator ist erforderlich.
- ◆ Der Zeitonenindikator ist optional. Wird er weggelassen, gilt die Uhrzeit der jeweiligen Zeitzone.
- ◆ Der Wochentag, falls angegeben, wird ignoriert.

Beispiele:

```
"3/11/06 1:32 PM"  
"Mar 11, 2006 1:32:38 PM"  
"March 11, 2006 1:32:38 PM PST"  
"Saturday, March 11, 2006 1:32:38 PM PST"
```

Verwandte Referenz

[Syntax der zusammengesetzten Optionen des VCM-Dienstprogramms](#)

[<Aktion>](#)

[<Änderungsanforderungen>](#)

[<Auscheckoptionen>](#)

[<Dateien>](#)

[<Ordner>](#)

[<Bedingungen für Übereinstimmung>](#)

[<Prozesselement>](#)

[<Anforderungen>](#)

[<Revisions-Labels>](#)

[<Tasks>](#)

[<Themen>](#)

[Befehlsreferenz](#)

<Themen>

Topics {ALL | *{<Themennr.>}}

Enthält die einzelnen angegebenen Themen nach Themennummer. Das Schlüsselwort `Topics` kann auch in der Singularform verwendet werden.

Verwandte Referenz

[Syntax der zusammengesetzten Optionen des VCM-Dienstprogramms](#)

[<Aktion>](#)

[<Änderungsanforderungen>](#)

[<Auscheckoptionen>](#)

[<Dateien>](#)

[<Ordner>](#)

[<Bedingungen für Übereinstimmung>](#)

[<Prozesselement>](#)

[<Anforderungen>](#)

[<Revisions-Labels>](#)

[<Tasks>](#)

[<Zeitstempel>](#)

[Befehlsreferenz](#)

Index

- All, 110
- Änderungspakete
 - VCMUtility, 79
- Bulk Check-out-Dienstprogramm
 - Befehlszeilenoptionen, 9
- Checkout Trace-Dienstprogramm
 - Befehlszeilenoptionen, 18
- I, 54
- Vault Verify
 - Befehlszeile, 20
- VCM Protokolldateien
 - VCMUtility, 79
- VCM-Dienstprogramm
 - <Tasks>, 120
- VCMUtility
 - Übersicht, 76
 - <Aktion>, 106
 - <Änderungsanforderungen>, 109
 - <Anforderungen>, 118
 - <Auscheckoptionen>, 107
 - <Bedingungen für Übereinstimmung>, 115
 - <Dateien>, 110
 - <Elementtyp>, 114
 - <Ordner>, 112
 - <Prozesselement>, 117
 - <Revisions-Labels>, 119
 - <Themen>, 122
 - <Zeitstempel>, 121
 - Abkürzungen, 78
 - AutoLogon, 83
 - AutoMergeFiles, 85
 - AutoMergeProperties, 85
 - Basis automatisch neu bestimmen, 99
 - Befehl, 76 80
 - Befehle, 80
 - Befehlsreferenz, 101
 - Befehlstypen, 80
 - Befehlszeilenparameter, 77
 - Beispiele, 99
 - Boolesche Optionen, 78
 - BreakLocks, 85
 - CaseSensitiveFileNames, 85
 - CheckoutPreview, 86
 - CommitMerge, 86
 - DefaultAction, 86
 - DefaultComment, 87
 - Delete-Befehl, 80
 - Encryption, 83
 - Exclude, 88
 - Exit-Codes, 78
 - Export, 88
 - FixFloatingChildShares, 89
 - Help-Befehlstyp, 80
 - IgnoreMergePoints, 89
 - Import-Befehl, 80
 - Include, 89
 - Input-Quellen, 77
 - LockMergeConflicts, 90
 - ManualMergeFiles, 90
 - Match, 90
 - MergeType, 91
 - Name des Änderungspakets, 91
 - NetMon, 98
 - Neue Sitzung (Befehlstyp), 80
 - Open-Befehlstyp, 81
 - Optionen für wieder aufgenommene Sitzungen, 96
 - Optionsdatei, 77
 - Per Views-Label heraufstufen, 99 100
 - PostCommitLabel, 91
 - PostCommitRevLabel, 92
 - PreCommitLabel, 92
 - PreCommitRevLabel, 92
 - PreventDuplicateFileNames, 92
 - Project, 92
 - PWDFile, 83
 - Replay-Befehlstyp, 81
 - ReportDiffs, 93
 - ReportUpdates, 93
 - Resume-Befehlstyp, 82
 - Save, 93
 - Server, 83
 - Sitzungsoptionen, 85
 - Sonstige Optionen, 98
 - SourceTime, 94
 - SourceView, 94
 - SourceState, 94
 - SrcLabel, 94
 - Syntaxkonventionen, 76
 - TargetLabel, 95
 - TargetState, 95
 - TargetTime, 95
 - TargetView, 95
 - Time, 98
 - Unicode-Optionswerte, 77
 - UseCA, 84 84
 - Verbindungsoptionen, 83
 - Verbose, 98