﻿== CMD\_DESCRIPTION\_ACL ==

객체에 권한을 설정합니다.

== CMD\_USAGE\_ACL ==

사용법:

cm ^acl (--^user=<사용자 이름> | --^group=<그룹 이름>)

(-^allowed|-^denied|-^overrideallowed|-^overridedenied=+|-<권한>[,...])[,...]

<객체 사양>

--^user 사용자 이름입니다.

--^group 그룹 이름입니다.

-^allowed 지정된 권한을 활성화합니다. 쉼표를

사용하여 권한을 구분하십시오. ('cm ^showpermissions'을 사용하여

사용 가능한 모든 권한을 표시할 수 있습니다.)

-^denied 지정된 권한을 거부합니다. 쉼표를

사용하여 권한을 구분하십시오. ('cm ^showpermissions'을 사용하여

사용 가능한 모든 권한을 표시할 수 있습니다.)

-^overrideallowed 허용된 권한을 재정의합니다. 쉼표를

사용하여 권한을 구분하십시오. ('cm ^showpermissions'을 사용하여

사용 가능한 모든 권한을 표시할 수 있습니다.)

-^overridedenied 거부된 권한을 재정의합니다. 쉼표를

사용하여 권한을 구분하십시오. ('cm ^showpermissions'을 사용하여

사용 가능한 모든 권한을 표시할 수 있습니다.)

객체 사양 권한이 설정될 객체입니다.

이 명령에 유효한 객체는 다음과 같습니다.

리포지토리 서버, 리포지토리, 브랜치, 체인지 세트, 레이블, 항목,

속성.

(사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을 사용하십시오.)

보안 설정된 경로의 특수 용도:

cm ^acl [(--^user=<사용자\_이름> | --^group=<그룹\_이름>)

(-^allowed|-^denied|-^overrideallowed|-^overridedenied=+|-<권한>[,...])[,...]]

[--^delete] [--^branches=[+ | -]<브랜치>[,...]]

<사양>

--^delete 보안 설정된 경로를 제거합니다.

자세한 내용은 비고를 확인하십시오.

--^branches 브랜치 그룹에 보안 설정된 경로 권한을 설정합니다.

쉼표를 사용하여 브랜치를 구분하십시오.

필요에 따라 각 브랜치 앞에 +나 - 기호를 붙여

편집할 때 해당 브랜치가 목록에 추가되어야 하는지 아니면

삭제되어야 하는지 지정할 수 있습니다.

자세한 내용은 비고를 확인하십시오.

사양 권한을 설정할 보안 설정된 경로입니다.

== CMD\_HELP\_ACL ==

권한을 구성하려면 Plastic SCM 보안 작동 방식을 알아야 합니다.

권한 작동 방식에 대해 알아보려면 보안 가이드를 확인하십시오.

https://www.plasticscm.com/download/help/securityguide

비고:

이 명령은 지정된 객체, 리포지토리, 브랜치, 레이블 또는 서버 경로에 대한

사용자 또는 그룹의 권한을 설정합니다.

객체 사양:

(객체를 지정하는 방법을 알아보려면 'cm ^help ^objectspec'을 사용하십시오.)

'^acl' 명령은 사양의 특수한 유형에 속하는 보안 설정된 경로를 사용합니다.

- 보안 설정된 경로 사양:

^path:server\_path[#tag]

예: ^path:/src/foo.c

^path:/doc/pdf

^path:/doc/pdf#documents

권한 작업:

-^allowed 및 -^denied를 사용해 설정할 권한을 지정합니다.

-^overrideallowed 및 -^overridedenied 인수를 사용해

재정의할 권한을 지정합니다.

각 작업에는 쉼표로 구분된 권한 목록이 필요합니다.

권한 이름:

각 권한 이름 앞에는 + 또는 - 기호가 있습니다.

+ 기호는 권한을 설정하고 - 기호는 권한을 초기화합니다.

객체의 권한을 확인하려면 'cm ^showacl' 명령을 사용하십시오.

재정의된 권한:

-^overrideallowed 및 -^overridedenied를

사용해 권한을 재정의하면 상속을 무시할 수 있습니다.

리포지토리나 서버 수준에서 설정된 권한을 무시해야 할 때 사용하면

좋습니다.

예:

cm ^acl --^user=vio -^allowed=+^ci -^overrideallowed=+^ci ^br:qa@test

(사용자 'vio'의 권한이 리포지토리 수준에서 거부된 경우에도

사용자가 'test' 리포지토리의 'qa' 브랜치에 체크인할 수 있습니다.)

서버 경로 권한(보안 설정된 경로):

- 주어진 서버 경로에 대한 권한을 지정할 수 있습니다.

- 이 권한은 체크인 작업 중에 확인됩니다.

- 해당 권한은 업데이트 작업 중에도 확인될 수 있으며,

특정 디렉터리 및 파일이 워크스페이스로 다운로드되는 것을 방지하는 데

사용할 수 있습니다.

- 체크인할 모든 항목에 대해 서버는 항목 경로를

보안 설정된 경로와 일치시키기 위해 시도합니다. 경로가 일치하는 경우, 체크인 작업은

해당 항목에 체크인될 수 있는 권한이 있는지 확인합니다.

보안 설정된 경로에 대해 정의할 수 있는 권한은

다음과 같습니다.

'^ci', '^change', '^add', '^move', '^rm', '^read'

관련 항목 중 하나라도 권한 확인이 되지 않으면

체크인 작업이 롤백됩니다.

브랜치 그룹에 보안 설정된 경로 권한을 설정하려면

--^branches 옵션을 사용하십시오.

예:

cm ^acl --^user=jo -^denied=+^ci ^path:/src#rule0 --^branches=main,main/rel0

보안 설정된 경로와 관련된 ACL을 편집하려면 태그를 사용하는 것이 좋습니다.

예:

cm ^acl --^user=jo -^denied=+^rm ^path:/src#rule0

(태그가 없는 경우, 브랜치 목록을

다시 지정해야 합니다.)

보안 설정된 경로의 브랜치 목록은 편집할 수 있습니다.

예:

cm ^acl ^path:/src#rule0 --^branches=-main,+main/rel1

(목록에서 'main'을 제거하고 'main/rel1'을 추가합니다.)

보안 설정된 경로를 제거하려면 --^delete 인수를 사용하십시오.

예:

cm ^acl --^user=jo --^delete ^path:/src#rule0

상속:

상속은 Plastic SCM 3.0 버전부터 도입된 옵션입니다.

고급 기능이지만 거의 사용되지 않습니다.

상속을 사용하면 객체가 기본 상속 관계를 재정의하여

다른 객체의 권한을 상속할 수 있습니다.

상속 체인을 잘라내려면 -^cut 옵션을 사용하십시오.

현재 상속된 권한을 잘라내고 복사하려면 -^cutncpy 옵션을

사용하십시오. (이는 상속을 잘라내지만 실제 권한은 유지할 수 있는

Windows의 파일 시스템 권한과 유사합니다.)

-^inherit 옵션을 사용하면 사용자가 객체 사양에서 권한을 상속할 수 있습니다.

예: '-^inherit=객체 사양'

예:

cm ^acl --^user=danipen -^denied=+^ci ^rep:core

('core' 리포지토리에서 사용자 'danipen'에 대한 체크인을 거부합니다.)

cm ^acl --^group=developers -^allowed=+^view,-^read -^denied=+^chgperm ^br:main

(이 명령은 'main' 브랜치의 'developers' 그룹에 보기 권한을 부여하고,

읽기 권한을 초기화하고, 권한 변경 권한을 거부합니다.)

보안 설정된 경로 예:

cm ^acl --^group=devs -^denied=+^ci ^path:/server#rel --^branches=main,main/2.0

(이 명령은 'main' 및 'main/2.0' 브랜치의 '/server'와 일치하는 경로에 대해

'개발자' 그룹의 체크인 권한을 거부합니다.

나중에 참조할 수 있도록 '#rel' 태그가 생성됩니다.)

cm ^acl ^path:/server#rel --^branches=-/main,+/main/Rel2.1

(태그가 'rel'인 보안 설정된 경로 '/server'를 업데이트하고 'main' 브랜치를 제거하고

보안 설정된 경로가 적용되는 브랜치 그룹에

'main/Rel2.1' 브랜치를 추가합니다. 앞선 예제를 고려하면,

이제 브랜치 목록에 'main/Rel2.1'과 'main/2.0'이 있게 됩니다.)

cm ^acl --^user=vsanchezm -^allowed=-^read -^overrideallowed=+^read ^path:/doc

('/doc' 경로에서 '^read' 권한을 재정의하는 사용자 'vsanchezm'에 대한 '^read' 권한을 제거합니다.)

== CMD\_DESCRIPTION\_ACTIVATEUSER ==

라이센스 사용자를 활성화합니다.

== CMD\_USAGE\_ACTIVATEUSER ==

사용법:

cm ^activateuser | ^au <사용자 이름>[ ...] [--^server=<리포지토리 서버 사양>]

사용자 이름 사용자 이름 또는 활성화할 사용자 이름입니다. 공백을 포함하여 사용자 이름을 지정하려면

큰따옴표(" ")를 사용하십시오. 사용자 이름을 구분하려면

공백을 사용하십시오.

옵션:

--^server=<리포지토리 서버 사양> 지정된 서버에서 사용자를 활성화합니다.

서버가 지정되지 않은 경우, client.conf 파일의

기본 서버에서 명령을 실행합니다.

(리포지토리 서버 사양에 관한 자세한 내용을 확인하려면

'cm ^help ^objectspec'을 사용하십시오.)

== CMD\_HELP\_ACTIVATEUSER ==

비고:

사용자를 활성화하려면 해당 사용자가 이전에 비활성화되었어야 합니다.

기본적으로 Plastic SCM에서 처음 쓰기 작업을 수행할 때

사용자가 활성화됩니다. 최대 사용자 수를 초과하지 않은 경우에만

사용자가 자동으로 활성화됩니다.

Plastic SCM 사용자 비활성화에 관한 자세한 내용은

'cm ^help ^deactivateuser' 명령을 확인하십시오.

예:

cm ^activateuser john

cm ^activateuser david "mary collins"

cm ^au peter --^server=localhost:8087

== CMD\_DESCRIPTION\_ADD ==

버전 제어에 항목을 추가합니다.

== CMD\_USAGE\_ADD ==

사용법:

cm ^add [-^R | -^r | --^recursive] [--^silent] [--^ignorefailed]

[--^skipcontentcheck] [--^coparent] [--^filetypes=<파일>] [--^noinfo]

[--^format=<문자열 형식>] [--^errorformat=<문자열 형식>]

<항목 경로>[ ...]

항목 경로 추가할 하나 또는 여러 항목입니다. 공백이 포함된 경로를 지정하려면

큰따옴표(" ")를 사용하십시오. 공백을 사용하여 항목을 구분하십시오.

현재 디렉터리의 모든 내용을 추가하려면 \*을 사용하십시오.

옵션:

-^R -^r --^recursive 항목을 재귀적으로 추가합니다.

--^silent 어떤 내용도 출력하지 않습니다.

--^ignorefailed 항목을 추가할 수 없는 경우, 추가 작업이 해당 항목을 제외하고

계속 진행됩니다. 참고: 디렉터리를 추가할 수 없는 경우

디렉터리의 내용이 추가되지 않습니다.

--^skipcontentcheck 확장자가 부족하여 파일을 텍스트나 바이너리로

설정할 수 없는 경우, 내용을 확인하는 대신

파일을 바이너리로 설정하여 유형을 감지합니다. 이는 대규모

체크인에서 성능을 높이기 위해 수행됩니다.

--^coparent 추가되는 항목의 부모에 대한 체크아웃을 실행합니다.

--^filetypes 사용할 파일 유형 파일입니다. 자세한 내용은 다음 링크를

참조하십시오.

http://blog.plasticscm.com/2008/03/custom-file-types.html

--^noinfo 진행 정보를 출력하지 않습니다.

--^format 특정 형식으로 출력 메시지를 가져옵니다. 자세한 내용은

예를 참조하시기 바랍니다.

--^errorformat 오류 메시지가 있는 경우 특정 형식으로

가져옵니다. 자세한 내용은 예를 참조하시기 바랍니다.

== CMD\_HELP\_ADD ==

비고:

항목 추가 요구 사항:

- 추가할 항목의 부모 디렉터리가 미리 추가되어 있어야 합니다.

표준 입력에서 입력 읽기:

'^add' 명령은 표준 입력에서 경로를 읽을 수 있습니다. 그러려면 단일 대시 "-" 기호를

사용하십시오.

예: cm ^add -

경로 읽기는 빈 행이 입력될 때까지 계속됩니다.

따라서 파이프를 사용해 추가할 파일을 지정할 수 있습니다.

예:

dir /S /B \*.c | cm ^add -

(Windows에서는 워크스페이스에 모든 .c 파일을 추가합니다.)

예:

cm ^add file1.txt file2.txt

('file1.txt' 및 'file2.txt' 항목을 추가합니다.)

cm ^add c:\workspace\file.txt

('file.txt' 항목을 경로 'c:\workspace'에 추가합니다.)

cm ^add -^R c:\workspace\src

('src'를 재귀적으로 추가합니다.)

cm ^add -^R \*

(현재 디렉터리의 모든 내용을 재귀적으로 추가합니다.)

cm ^add -^R \* --^filetypes=filetypes.conf

(현재 디렉터리의 내용을 확인하는 대신

'filetypes.conf'를 사용하여 확장자에 따라 파일의 유형을 각각 지정하여

현재 디렉터리의 모든 내용을 재귀적으로 추가합니다.)

cm ^add --^coparent c:\workspace\dir\file.txt

('file.txt'를 소스 제어에 추가하고 'dir' 체크아웃을 수행합니다.)

cm ^add -^R \* --^format="ADD {0}" --^errorformat="ERR {0}"

(현재 디렉터리의 모든 내용을 재귀적으로 추가하며

성공적으로 추가한 파일에는 '^ADD <항목>'을, 추가할 수 없는 항목에는 '^ERR <항목>'을

출력합니다.)

== CMD\_USAGE\_ADDIGNOREPATTERN ==

사용법:

cm ^addignorepattern <패턴>[ ...]

[--^workspace=<워크스페이스 경로> | --^allworkspaces] [--^remove]

== CMD\_DESCRIPTION\_ADMIN ==

서버에서 관리 명령을 실행합니다.

== CMD\_USAGE\_ADMIN ==

사용법:

cm ^admin <명령> [옵션]

사용 가능한 명령:

^readonly

각 명령에 대한 자세한 정보는 다음을 실행해 확인하십시오.

cm ^admin <명령> --^usage

cm ^admin <명령> --^help

== CMD\_HELP\_ADMIN ==

비고:

서버 관리자만 관리 명령을 실행할 수 있습니다.

예:

cm ^admin ^readonly ^enter

cm ^admin ^readonly ^status

== CMD\_DESCRIPTION\_ADMIN\_READONLY ==

서버 읽기 전용 모드를 활성화/비활성화합니다.

== CMD\_USAGE\_ADMIN\_READONLY ==

사용법:

cm ^admin ^readonly (^enter | ^leave | ^status) [<서버>]

작업:

^enter 서버가 읽기 전용 모드로 변경됩니다. 쓰기 작업이 거부됩니다.

^leave 서버가 읽기 전용 모드를 종료합니다.

^status 서버의 읽기 전용 모드 상태를 표시합니다.

옵션:

서버 지정된 서버(서버:포트)에서 명령을 실행합니다. (서버 사양에 관한

자세한 내용을 확인하려면 'cm ^help ^objectspec'을 사용하십시오.)

서버가 지정되지 않은 경우, 현재 워크스페이스의 서버에서

명령이 적용됩니다.

현재 경로가 워크스페이스에 없는 경우, client.conf 구성 파일에 정의된

기본 서버에서 명령이 적용됩니다.

== CMD\_HELP\_ADMIN\_READONLY ==

비고:

서버 관리자만 서버 읽기 전용 모드로 변경할 수 있습니다.

예:

cm ^admin ^readonly ^enter diana:8086

cm ^admin ^readonly ^leave

== CMD\_DESCRIPTION\_ANNOTATE ==

파일의 각 행이 마지막으로 수정된 체인지 세트와 작성자를 표시합니다.

== CMD\_USAGE\_ANNOTATE ==

사용법:

cm ^annotate | ^blame <사양>[ ...]

[--^format=<문자열 형식>]

[--^ignore=(^eol | ^whitespaces | ^"eol&whitespaces" | ^none)]

[--^dateformat=<문자열 날짜 형식>]

[--^encoding=<이름>]

[--^stats]

[--^repository=<리포지토리 사양>]

사양 주석을 지정할 파일의 사양입니다.

(사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을 사용하십시오.)

공백이 포함된 경로를 지정하려면 큰따옴표(" ")를 사용하십시오.

옵션:

--^format 특정 형식으로 출력 메시지를 가져옵니다. 자세한 내용은

비고를 확인하십시오.

--^ignore 지정된 비교 방법을 설정합니다.

자세한 내용은 비고를 확인하십시오.

--^dateformat 날짜를 출력할 출력 형식을 설정합니다.

--^encoding 출력 인코딩, 즉 UTF-8을 지정합니다.

지원되는 인코딩 및 형식에 대한 표를 확인하려면

http://msdn.microsoft.com/en-us/library/system.text.encoding.aspx에서

MSDN 문서를 확인하십시오.

페이지 마지막 "Name"열에 있습니다.

--^stats 통계 정보를 표시합니다.

--^repository 주석을 계산하는 데 사용되는 리포지토리 사양을

지정합니다. 기본적으로 이 명령은

워크스페이스에 로드된 개정 리포지토리가 저장된

리포지토리를 사용합니다. (리포지토리 사양에 관한 자세한 내용을 확인하려면

'cm ^help ^objectspec'을 사용하십시오.)

== CMD\_HELP\_ANNOTATE ==

비고:

바이너리 파일에는 주석을 달 수 없습니다.

--^ignore 옵션:

^none 행 끝 및 공백 차이점을 감지합니다.

^eol 행 끝 차이점을 무시합니다.

^whitespaces 공백 차이점을 무시합니다.

^"eol&whitespaces" 행 끝 및 공백 차이점을 무시합니다.

--^format 옵션:

이 명령의 출력 매개 변수는 다음과 같습니다.

{^owner} 마지막으로 행을 변경한 사용자.

{^rev} 행의 소스 개정 사양.

{^content} 행의 내용.

{^date} 행이 체크인된 날짜.

{^comment} 행의 소스 개정에 대한 코멘트.

{^changeset} 행의 소스 개정에 대한 체인지 세트.

{^line} 파일의 행 번호.

{^id} 항목 ID.

{^parentid} 항목의 부모 ID.

{^rep} 항목의 리포지토리.

{^branch} 행의 소스 개정 브랜치.

{^ismergerev} 병합 시 행 개정 생성 여부.

--^dateformat:

날짜를 출력할 출력 형식을 지정할 때 사용합니다.

다음 주소에서 지원되는 형식을 참조하십시오.

https://docs.microsoft.com/en-us/dotnet/standard/base-types/custom-date-and-time-format-strings

--^repository:

원격 리포지토리에서 데이터를 가져올 때 사용합니다. 분산형 시나리오에서

사용하기 좋습니다.

예:

cm ^blame c:\workspace\src --^ignore=^"eol&whitespaces" --^encoding=UTF-8

cm ^annotate c:\workspace\file.txt --^ignore=^eol

cm ^annotate c:\workspace\file.txt --^format="{^owner} {^date, 10} {^content}"

(소유자 필드, 공백, 날짜 필드(오른쪽 정렬), 공백, 내용을

차례로 작성합니다.)

cm ^blame c:\workspace\file.txt --^format="{^owner, -7} {^comment} {^date}" \

--^dateformat=yyyyMMdd

(소유자 필드(7글자, 왼쪽 정렬), 공백,

코멘트, 공백을 차례로 작성하고 마지막으로

형식에 맞는 날짜(예: 20170329)를 작성합니다.)

cm ^annotate c:\workspace\file.txt --^repository=centralRep@myserver:8084

cm ^blame ^serverpath:/src/client/checkin/Checkin.cs#^cs:73666

(서버 경로를 사용하여 체인지 세트 73666에서부터 파일에 주석을 답니다.)

== CMD\_DESCRIPTION\_APPLY\_LOCAL ==

로컬 변경사항(로컬에서 이동, 삭제, 변경)을 확인하고

이를 적용하여 Plastic SCM이 변경사항을 추적합니다.

== CMD\_USAGE\_APPLY\_LOCAL ==

사용법:

cm ^applylocal | ^al [--^dependencies] [<항목 경로>[ ...]]

[--^machinereadable [--^startlineseparator=<구분 기준>]

[--^endlineseparator=<구분 기준>] [--^fieldseparator=<구분 기준>]]

옵션:

--^dependencies 적용할 항목에 로컬 변경 종속성을

추가합니다.

항목 경로 적용될 항목. 공백을 사용하여 경로를

구분하십시오. 공백이 포함된 경로를 지정하려면 큰따옴표(" ")를

사용하십시오.

--^machinereadable 결과를 구문 분석이 쉬운 형식으로 출력합니다.

--^startlineseparator '--^machinereadable' 플래그와 함께 사용되며, 행이 시작되는 방법을

지정합니다.

--^endlineseparator '--^machinereadable' 플래그와 함께 사용되며, 행이 끝나는 방법을

지정합니다.

--^fieldseparator '--^machinereadable' 플래그와 함께 사용되며, 필드를 구분할 방법을

지정합니다.

== CMD\_HELP\_APPLY\_LOCAL ==

비고:

--^dependencies 및 <항목 경로>를 지정하지 않은 경우, 작업에

워크스페이스의 모든 로컬 변경사항이 포함됩니다.

항상 주어진 경로에서 재귀적으로 적용됩니다.

예:

cm ^applylocal foo.c bar.c

cm ^applylocal .

(현재 디렉터리의 모든 로컬 변경사항을 적용합니다.)

cm ^applylocal

(워크스페이스의 모든 로컬 변경사항을 적용합니다.)

cm ^applylocal --^machinereadable

(워크스페이스의 모든 로컬 변경사항을 적용하고, 그 결과를

구문 분석이 쉬운 간단한 형식으로 출력합니다.)

cm ^applylocal --^machinereadable --^startlineseparator=">" \

--^endlineseparator="<" --^fieldseparator=","

(워크스페이스의 모든 로컬 변경사항을 적용하고, 그 결과를

구문 분석이 쉬운 간단한 형식으로 출력하며, 지정된 문자열로 행을 시작 및 종료하고

필드를 구분합니다.)

== CMD\_DESCRIPTION\_ARCHIVE ==

데이터를 외부 스토리지에 보관합니다.

== CMD\_USAGE\_ARCHIVE ==

사용법:

cm ^archive | ^arch <개정 사양>[ ...] [-^c | --^comment=<문자열 코멘트>]

[--^file=<베이스 파일>]

(리포지토리에서 데이터를 추출하고 외부 스토리지에 저장합니다.)

cm ^archive | ^arch <개정 사양>[ ...] --^restore

(이전에 보관된 개정을 다시 리포지토리로 복원합니다.)

개정 사양 하나 이상의 개정 사양. "-" 한정어를 사용해

표준 입력에서 읽을 수 있습니다. (개정 사양에 관한 자세한 내용을 확인하려면

'cm ^help ^objectspec'을 사용하십시오.)

--^restore 생성된 보관 파일에서 이전에 보관된 데이터를

복원합니다.

옵션:

-^c | --^comment 생성할 보관 스토리지 파일에 코멘트를 설정합니다.

--^file 새로운 보관 데이터 파일의 이름 접두어 및

경로(선택사항).

== CMD\_HELP\_ARCHIVE ==

비고:

이 명령은 리포지토리 데이터베이스에서 데이터를 추출하고

외부 스토리지에 저장해 데이터베이스 공간을 절약합니다.

이 명령은 이전에 보관된 개정을 다시 리포지토리 데이터베이스로

복원(--^restore)할 수도 있습니다.

개정 사양을 지정하는 방법을 알아보려면 'cm ^help ^objectspec'을 사용하십시오.

이 명령을 실행하는 사용자는 반드시 작업을 완료할 권한이 있는

Plastic SCM 서버 관리자(리포지토리 서버 소유자)여야 합니다.

지정된 개정의 모든 데이터는 서로 다른 파일에 저장되며, 파일 이름은 --^file

인수로 정의된 값으로 시작하는 이름으로

저장됩니다. 이 인수는 향후 보관 파일의 접두어를 포함하는 전체 경로 값을 포함하거나,

접두어 값만 포함할 수 있습니다.

보관이 완료되면 지정된 개정의 데이터에 다음 두 가지 방법으로

액세스할 수 있습니다.

- 클라이언트에서: 클라이언트는 데이터의 보관 여부를 감지하고

사용자에게 파일의 위치를 입력하라는 메시지를 표시합니다.

사용자는 보관된 데이터가 위치한 경로를 포함하는

externaldata.conf라는 파일을 client.conf 파일에 적용되는 것과 같은 규칙을 사용해

표준 구성 파일 위치에 생성하여

외부 데이터 위치를 구성할 수 있습니다.

- 서버에서: 이 방법으로 액세스하는 경우, 요청이 서버에 의해

투명하게 해결되므로 사용자는 데이터의 보관 여부를

알 필요가 없습니다. 그렇게 하려면 관리자는 externaldata.conf라는 파일을

서버 디렉터리에 생성하고 해당 파일을 보관된 분량이 있는 경로로

채워야 합니다.

개정 또는 개정 세트의 보관을 취소(복원)하려면

보관된 파일을 클라이언트에서 액세스할 수 있어야 합니다. 따라서 서버에 의해 해결되는 데이터(방법 2)의 경우,

클라이언트가 이를 보관된 데이터로 인식할 수 없기 때문에

데이터 보관을 취소할 수 없습니다. 방법 2를 사용했을 때

성공적으로 보관을 취소하려면, 관리자는 먼저

externaldata.conf 서버 파일을 편집하여 보관을 취소해야 하는 보관된 파일에 대한 액세스를

제거해야 합니다.

코멘트를 입력할 편집기를 지정하려면 PLASTICEDITOR 환경 변수를

설정하십시오.

예:

cm ^archive bigfile.zip#^br:/main

('main' 브랜치에 있는 'bigfile.zip'의 마지막 개정을 보관합니다.)

cm ^archive ^rev:myfile.pdf#^cs:2 -^c="대용량 PDF 파일" --^file=c:\arch\_files\arch

(myfile.pdf의 체인지 세트 2가 있는 개정을 'c:\archived\_files' 폴더에

보관합니다. 보관된 파일의 이름은 'arch'로 시작됩니다. 예: arch\_11\_56)

cm ^find "^revs ^where ^size > 26214400" --^format="{^item}#{^branch}" \

--^nototal | cm ^archive --^comment="volume00" --^file="volume00" -

(크기가 25Mb보다 큰 모든 파일을 파일 이름이 'volume00'으로 시작하는 파일에

보관합니다.)

cm ^find "^revs ^where ^size > 26214400 ^and ^archived='true'" \

--^format="{^item}#{^branch}" --^nototal | cm ^archive --^restore

(크기가 25Mb보다 큰 모든 보관된 파일을 복원합니다.)

== CMD\_DESCRIPTION\_ATTRIBUTE ==

사용자가 속성을 관리하도록 허용합니다.

== CMD\_USAGE\_ATTRIBUTE ==

사용법:

cm ^attribute | ^att <명령> [옵션]

명령:

^create | ^mk

^delete | ^rm

^set

^unset

^rename

^edit

각 명령에 대한 자세한 정보는 다음을 실행해 확인하십시오.

cm ^attribute <명령> --^usage

cm ^attribute <명령> --^help

== CMD\_HELP\_ATTRIBUTE ==

예:

cm ^attribute ^create 상태

cm ^attribute ^set ^att:status ^br:/main/SCM105 공개

cm ^attribute ^unset ^att:status ^br:/main/SCM105

cm ^attribute ^delete ^att:status

cm ^attribute ^rename ^att:status "빌드 상태"

cm ^attribute ^edit ^att:status "CI 파이프라인의 작업 상태"

== CMD\_DESCRIPTION\_CHANGELIST ==

변경사항 목록에서 대기 중인 변경사항을 그룹화합니다.

== CMD\_USAGE\_CHANGELIST ==

사용법:

1) 변경 목록 객체 관리:

cm ^changelist | ^clist [--^symlink]

(모든 변경사항 목록을 표시합니다.)

cm ^changelist | ^clist ^add <변경사항 목록 이름>

[<변경사항 목록 설명>] [--^persistent | --^notpersistent] [--^symlink]

(변경사항 목록을 생성합니다.)

cm ^changelist | ^clist ^rm <변경사항 목록 이름> [--^symlink]

(선택된 변경사항 목록을 제거합니다. 변경사항 목록에 대기 중인 변경사항이 있으면

해당 변경사항은 ^default 변경사항 목록으로 이동됩니다.)

cm ^changelist | ^clist ^edit <변경사항 목록 이름> [<작업 이름> <작업 값>]

[--^persistent | --^notpersistent] [--^symlink]

(선택된 변경사항 목록을 편집합니다.)

2) 주어진 변경사항 목록의 내용 관리:

cm ^changelist | ^clist <변경사항 목록 이름> (^add | ^rm) <경로 이름>[ ...]

[--^symlink]

(주어진 경로 이름과 일치하는 변경사항을 추가('^add') 또는 제거('^rm')하여

선택한 변경사항 목록을 추가합니다. 경로 이름을 구분하려면

공백을 사용하십시오. 공백이 포함된 경로를 지정하려면 큰따옴표(" ")를

사용하십시오. 경로 상태는 '^Added' 또는 '^Checked-out'이어야 합니다.)

옵션:

변경사항 목록 이름 변경사항 목록의 이름.

변경사항 목록 설명 변경사항 목록의 설명.

작업 이름 변경사항 목록을 편집하려면 '^rename' 또는 '^description' 중에

선택하십시오.

작업 값 변경사항 목록을 편집할 때 새 이름이나 새 설명을

적용합니다.

--^persistent 변경사항 목록의 내용이 체크인 또는 되돌려진 경우에도

변경사항 목록은 워크스페이스에 남아 있습니다.

--^notpersistent (기본값) 변경사항 목록의 내용이 체크인 또는

되돌리진 경우에도 변경사항 목록이 워크스페이스에

남아 있지 않습니다.

--^symlink 대상이 아닌 기호 링크에 작업을

적용합니다.

== CMD\_HELP\_CHANGELIST ==

비고:

'^changelist' 명령은 워크스페이스의 대기 중인 변경사항 목록과

변경사항 목록에 포함된 변경사항을 모두 처리합니다.

예:

cm ^changelist

(현재 워크스페이스 변경사항 목록을 표시합니다.)

cm ^changelist ^add 구성 변경사항 "dotConf 파일" --^persistent

(대기 중인 변경사항 목록이 체크인되거나 되돌리기되면 현재 워크스페이스에 계속 유지될

파일 이름이 '구성 변경사항'이고 설명에 'dotConf 파일'이라고 기재된

새로운 변경사항 목록을 생성합니다.)

cm ^changelist ^edit 구성 변경사항 ^rename 구성 파일 --^notpersistent

('구성 변경사항'이라는 변경사항 목록을 편집하고 이름을 '구성 파일'로

변경합니다. 또한, 변경사항 목록을 "유지되지 않음"으로 변경합니다.)

cm ^changelist ^edit 구성 변경사항 --^notpersistent

('구성 변경사항'이라는 변경사항 목록을 편집하여 "유지되지 않음"으로 변경합니다.)

cm ^changelist ^rm 구성 파일

(대기 중인 변경사항 목록 '구성 파일'을 현재 워크스페이스에서 제거합니다.)

cm ^changelist 구성 파일 ^add foo.conf

('foo.conf' 파일을 '구성 파일' 변경사항 목록에 추가합니다.)

cm ^changelist 구성 파일 ^rm foo.conf readme.txt

('foo.conf' 파일과 'readme.txt' 파일을 '구성 파일' 변경사항 목록에서

제거하고 해당 파일을 시스템 기본 변경사항 목록으로 이동합니다.)

== CMD\_DESCRIPTION\_CHANGESET ==

체인지 세트에 대해 고급 작업을 실행합니다.

== CMD\_USAGE\_CHANGESET ==

사용법:

cm ^changeset <명령> [옵션]

명령:

^move | ^mv

^delete | ^rm

^editcomment | ^edit

각 명령에 대한 자세한 정보는 다음을 실행해 확인하십시오.

cm ^changeset <명령> --^usage

cm ^changeset <명령> --^help

== CMD\_HELP\_CHANGESET ==

예:

cm ^changeset ^move ^cs:15@myrepo ^br:/main/scm005@myrepo

cm ^changeset ^delete ^cs:2b55f8aa-0b29-410f-b99c-60e573a309ca@devData

== CMD\_DESCRIPTION\_CHANGESET\_EDIT\_COMMENT ==

체인지 세트의 코멘트를 수정합니다.

== CMD\_USAGE\_CHANGESET\_EDIT\_COMMENT ==

사용법:

cm ^changeset ^editcomment | ^edit <체인지 세트 사양> <새 코멘트>

옵션:

체인지 세트 사양 코멘트를 편집할 대상 체인지 세트.

(체인지 세트 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을

사용하십시오.)

새 코멘트 대상 체인지 세트에 추가될

새 코멘트.

== CMD\_HELP\_CHANGESET\_EDIT\_COMMENT ==

비고:

- 대상 체인지 세트 사양이 반드시 유효해야 합니다.

예:

cm ^changeset ^editcomment ^cs:15@myrepo "체크인 세부사항을 추가하지 않았습니다"

cm ^changeset ^edit ^cs:cb11ecdb-1aa9-4f11-8698-dcab14e5885a \

"이전 코멘트 텍스트는 이 코멘트 텍스트로 교체됩니다."

== CMD\_DESCRIPTION\_CHANGESET\_MOVE ==

체인지 세트와 그 모든 자손을 다른 브랜치로 이동합니다.

== CMD\_USAGE\_CHANGESET\_MOVE ==

사용법:

cm ^changeset ^move | ^mv <체인지 세트 사양> <브랜치 사양>

옵션:

체인지 세트 사양 다른 브랜치로 이동할 첫 번째 체인지 세트. 같은

브랜치에 있는 모든 자손 체인지 세트도

해당 명령의 실행 대상입니다.

(체인지 세트 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을

사용하십시오.)

브랜치 사양 대상 체인지 세트가 저장되는

대상 브랜치. 비어 있거나 존재하지 않는 브랜치여야 합니다.

대상 브랜치가 존재하지 않는 경우, 명령에 따라

생성됩니다.

(브랜치 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을

사용하십시오.)

== CMD\_HELP\_CHANGESET\_MOVE ==

비고:

- 대상 체인지 세트 사양이 반드시 유효해야 합니다.

- 대상 브랜치가 비어 있거나 존재하지 않아야 합니다.

- 대상 브랜치가 존재하지 않는 경우, 브랜치가 생성됩니다.

- 브랜치는 병합 링크에 영향을 주지 않기 때문에, 병합 링크가 변경되지 않은 상태로 유지됩니다.

예:

cm ^changeset ^move ^cs:15@myrepo ^br:/main/scm005@myrepo

cm ^changeset ^move ^cs:cb11ecdb-1aa9-4f11-8698-dcab14e5885a ^br:/hotfix/TL-352

== CMD\_DESCRIPTION\_CHANGESET\_DELETE ==

리포지토리에서 체인지 세트를 삭제합니다.

== CMD\_USAGE\_CHANGESET\_DELETE ==

사용법:

cm ^changeset ^delete | ^rm <체인지 세트 사양>

옵션:

체인지 세트 사양 제거될 대상 체인지 세트. 체인지 세트는

특정 조건을 충족해야 합니다. 자세한 내용은 비고를 확인하십시오.

(체인지 세트 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을

사용하십시오.)

== CMD\_HELP\_CHANGESET\_DELETE ==

비고:

- 대상 체인지 세트는 해당 브랜치의 마지막 체인지 세트여야 합니다.

- 대상 체인지 세트는 다른 체인지 세트의 부모일 수 없습니다.

- 대상 체인지 세트는 병합 링크의 소스일 수 없으며,

소스로서 간격 병합의 일부일 수도 없습니다.

- 대상 체인지 세트에 적용된 레이블이 없어야 합니다.

- 대상 체인지 세트가 루트 체인지 세트('^cs:0')여서는 안 됩니다

예:

cm ^changeset ^rm ^cs:4525@myrepo@myserver

cm ^changeset ^delete ^cs:cb11ecdb-1aa9-4f11-8698-dcab14e5885a

== CMD\_DESCRIPTION\_CHANGEUSERPASSWORD ==

사용자 비밀번호(UP)를 변경합니다.

== CMD\_USAGE\_CHANGEUSERPASSWORD ==

사용법:

cm ^changepassword | ^passwd

== CMD\_HELP\_CHANGEUSERPASSWORD ==

비고:

이 명령은 보안 구성이 UP(사용자/비밀번호)인 경우에만

사용할 수 있습니다. 자세한 내용은 관리 가이드를 참조하십시오.

기존 비밀번호와 새 비밀번호가 필요합니다.

예:

cm ^passwd

== CMD\_DESCRIPTION\_CHECKCONNECTION ==

서버 연결을 확인합니다.

== CMD\_USAGE\_CHECKCONNECTION ==

사용법:

cm ^checkconnection | ^cc

== CMD\_HELP\_CHECKCONNECTION ==

비고:

- 이 명령은 구성된 Plastic SCM 서버와의 연결이 유효한지 여부를 나타내는

메시지를 반환합니다.

- 이 명령은 구성된 사용자가 유효한지 여부를 확인합니다. 또한

서버와의 버전 호환성을 확인합니다.

== CMD\_DESCRIPTION\_CHECKDB ==

리포지토리 무결성을 검사합니다.

== CMD\_USAGE\_CHECKDB ==

사용법:

cm ^checkdatabase | ^chkdb [<리포지토리 서버 사양> | <리포지토리 사양>]

리포지토리 서버 및 리포지토리 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을 사용하십시오.

== CMD\_HELP\_CHECKDB ==

비고:

- 리포지토리 서버 사양이나 리포지토리 사양이 모두 지정되지 않은 경우,

client.conf 파일에 지정된 서버에서 검사가 진행됩니다.

예:

cm ^checkdatabase ^repserver:localhost:8084

cm ^chkdb ^rep:default@localhost:8084

== CMD\_DESCRIPTION\_CHECKIN ==

변경사항을 리포지토리에 저장합니다.

== CMD\_USAGE\_CHECKIN ==

사용법:

cm ^checkin | ^ci [<항목 경로>[ ...]]

[-^c=<문자열 코멘트> | -^commentsfile=<코멘트 파일>]

[--^all|-^a] [--^applychanged] [--^private] [--^update] [--^symlink]

[--^noshowchangeset]

[--^machinereadable [--^startlineseparator=<구분 기준>]

[--^endlineseparator=<구분 기준>] [--^fieldseparator=<구분 기준>]]

옵션:

항목 경로 체크인될 항목. 공백이 포함된 경로를 지정하려면

큰따옴표(" ")를 사용하십시오. 항목 경로를 구분하려면

공백을 사용하십시오.

현재 디렉터리에 체크인을 적용하려면 온점(.)을 사용하십시오.

-^c 체크인 작업에서 생성된 체인지 세트에 지정된 코멘트를

적용합니다.

-^commentsfile 체크인 작업에서 생성된 체인지 세트에

지정된 파일의 코멘트를 적용합니다.

--^all | -^a 주어진 경로에서 로컬로 변경, 이동, 삭제된 항목도

포함됩니다.

--^applychanged 체크아웃된 항목과 함께 워크스페이스에서 감지된

변경된 항목에 체크인 작업을

적용합니다.

--^private 워크스페이스에서 감지된 비공개 항목도

포함됩니다.

--^update 업데이트-병합이 발생하면 이를 자동으로

처리합니다.

--^symlink 대상이 아닌 기호 링크에 체크인 작업을

적용합니다.

--^noshowchangeset 결과 체인지 세트를 출력하지 않습니다.

--^machinereadable 결과를 구문 분석이 쉬운 형식으로 출력합니다.

--^startlineseparator '--^machinereadable' 플래그와 함께 사용되며, 행이 시작되는 방법을

지정합니다.

--^endlineseparator '--^machinereadable' 플래그와 함께 사용되며, 행이 끝나는 방법을

지정합니다.

--^fieldseparator '--^machinereadable' 플래그와 함께 사용되며, 필드를 구분할 방법을

지정합니다.

== CMD\_HELP\_CHECKIN ==

비고:

- <항목 경로>를 지정하지 않은 경우, 체크인에

워크스페이스의 모든 대기 중인 변경사항이 포함됩니다.

- 체크인 작업은 항상 주어진 경로에서 재귀적으로 적용됩니다.

- 항목을 체크인하려면 다음 조건을 충족해야 합니다.

- 항목이 소스 코드 제어 상태여야 합니다.

- 항목이 비공개(소스 코드 제어 대상이 아님)인 경우, --^private

플래그가 있어야 항목을 체크인할 수 있습니다.

- 항목이 체크아웃된 상태여야 합니다.

- 항목이 변경되었으나 체크아웃되지 않은 경우, <항목 경로>가 디렉터리이거나,

경로에 임의 문자 기호('\*')가 없는 한 --^applychanged 플래그는 필요하지

않습니다.

개정의 내용이 이전 개정과 달라야 체크인할 수

있습니다.

코멘트를 입력할 편집기를 지정하려면 PLASTICEDITOR 환경 변수를

설정하십시오.

표준 입력에서 입력 읽기:

'^checkin' 명령은 표준 입력에서 경로를 읽을 수 있습니다. 그러려면 단일 대시 "-"를

전달합니다.

예: cm ^checkin -

경로 읽기는 빈 행이 입력될 때까지 계속됩니다.

따라서 파이프를 사용해 체크인할 파일을 지정할 수 있습니다.

예:

dir /S /B \*.c | cm ^checkin --^all -

(Windows에서는 워크스페이스에 있는 모든 .c 파일을 체크인합니다.)

예:

cm ^checkin file1.txt file2.txt

(체크아웃된 'file1.txt' 파일과 'file2.txt' 파일을 체크인합니다.)

cm ^checkin . -^commentsfile=mycomment.txt

(현재 디렉터리를 체크인하고 'mycomment.txt' 파일에 코멘트를

설정합니다.)

cm ^checkin 링크 --^symlink

(대상이 아닌 '링크' 파일을 체크인합니다. UNIX 환경에서 사용할 수

있습니다.)

cm ^ci file1.txt -^c="내 코멘트"

('file1.txt' 파일을 체크인하고 코멘트를 포함합니다.)

cm ^status --^short --^compact --^changelist=대기 중인 검토 | cm ^checkin -

('대기 중인 검토'라는 변경사항 목록에 경로를 목록으로 나열하고

해당 목록을 체크인 명령의 입력으로 리디렉션합니다.)

cm ^ci . --^machinereadable

(현재 디렉터리를 체크인하고 그 결과를 구문 분석이 쉬운 간단한 형식으로

출력합니다.)

cm ^ci . --^machinereadable --^startlineseparator=">" --^endlineseparator="<" --^fieldseparator=","

(현재 디렉터리를 체크인하고, 그 결과를 구문 분석이 쉬운 간단한

형식으로 출력하며, 지정된 문자열로 행을 시작 및 종료하고

필드를 구분합니다.)

== CMD\_DESCRIPTION\_CHECKOUT ==

파일을 수정 준비 완료 상태로 표시합니다.

== CMD\_USAGE\_CHECKOUT ==

사용법:

cm ^checkout | ^co [<항목 경로>[ ...]] [-^R | -^r | --^recursive]

[--^format=<문자열 형식>]

[--^errorformat=<문자열 형식>] [--^resultformat=<문자열 형식>]

[--^silent] [--^symlink] [--^ignorefailed]

[--^machinereadable [--^startlineseparator=<구분 기준>]

[--^endlineseparator=<구분 기준>] [--^fieldseparator=<구분 기준>]]

옵션:

항목 경로 체크아웃될 항목. 공백이 포함된 경로를 지정하려면

큰따옴표(" ")를 사용하십시오. 항목 경로를 구분하려면

공백을 사용하십시오.

현재 디렉터리에 체크아웃을 적용하려면 온점(.)을 사용하십시오.

-^R 파일을 재귀적으로 체크아웃합니다.

--^format 특정 형식으로 출력 진행도 메시지를

가져옵니다. 자세한 내용은 예를 참조하시기 바랍니다.

--^errorformat 오류 메시지가 있는 경우 특정 형식으로

가져옵니다. 자세한 내용은 예를 참조하시기 바랍니다.

--^resultformat 특정 형식으로 출력 결과 메시지를

가져옵니다. 자세한 내용은 예를 참조하시기 바랍니다.

--^silent 어떤 출력도 표시하지 않습니다.

--^symlink 대상이 아닌 기호 링크에 체크아웃 작업을

적용합니다.

--^ignorefailed 항목을 잠글 수 없는 경우(배타적 체크아웃을

수행할 수 없는 경우), 체크아웃 작업이 해당 항목을 제외하고

계속 진행됩니다.

--^machinereadable 결과를 구문 분석이 쉬운 형식으로 출력합니다.

--^startlineseparator '--^machinereadable' 플래그와 함께 사용되며, 행이 시작되는 방법을

지정합니다.

--^endlineseparator '--^machinereadable' 플래그와 함께 사용되며, 행이 끝나는 방법을

지정합니다.

--^fieldseparator '--^machinereadable' 플래그와 함께 사용되며, 필드를 구분할 방법을

지정합니다.

== CMD\_HELP\_CHECKOUT ==

비고:

항목을 체크아웃하려면 다음 조건을 충족해야 합니다.

- 항목이 소스 코드 제어 상태여야 합니다.

- 항목이 체크인된 상태여야 합니다.

서버에서 잠금이 구성된 경우(lock.conf가 있는 경우), 경로에서 체크아웃이

발생할 때마다 Plastic은 체크아웃이 규칙을 충족하는지 확인하고,

규칙을 충족하는 경우, 다른 사용자가 동시에 체크아웃할 수 없도록 경로가

배타적 체크아웃(잠김) 상태가 됩니다.

'cm ^lock ^list'를 사용해 서버의 모든 잠금을 확인할 수 있습니다.

자세한 내용은 관리자 가이드를 참조하십시오.

https://www.plasticscm.com/download/help/adminguide

형식 문자열은 '{0}' 플레이스홀더를 체크아웃 중인 항목의 경로로

교체합니다. 사용법은 예를 참조하시기 바랍니다.

표준 입력에서 입력 읽기:

'^checkout' 명령은 표준 입력에서 경로를 읽을 수 있습니다. 그러려면 단일 대시 "-"를

전달합니다.

예: cm ^checkout -

경로 읽기는 빈 행이 입력될 때까지 계속됩니다.

따라서 파이프를 사용해 체크아웃할 파일을 지정할 수 있습니다.

예:

dir /S /B \*.c | cm ^checkout -

(Windows에서는 워크스페이스에 있는 모든 .c 파일을 체크아웃합니다.)

예:

cm ^checkout file1.txt file2.txt

('file1.txt' 파일 및 'file2.txt' 파일을 체크아웃합니다.)

cm ^co \*.txt

(모든 txt 파일을 체크아웃합니다.)

cm ^checkout .

(현재 디렉터리를 체크아웃합니다.)

cm ^checkout -^R c:\workspace\src

('src' 폴더를 재귀적으로 체크아웃합니다.)

cm ^co file.txt --^format="{0} 항목 체크아웃 중"

--^errorformat="{0} 체크아웃 중 오류가 발생했습니다" /

--^resultformat="{0} 항목 체크아웃됨"

('file.txt' 항목을 체크아웃하고 지정된 형식 문자열을 사용하여

작업의 진행도, 결과, 오류를 표시합니다.)

cm ^checkout 링크 --^symlink

(대상이 아닌 '링크' 파일을 체크아웃합니다. UNIX 환경에서 사용할 수

있습니다.)

cm ^checkout . -^R --^ignorefailed

(현재 폴더를 재귀적으로 체크아웃하고 체크아웃될 수 없는 파일은

무시합니다.)

cm ^co . --^machinereadable --^startlineseparator=">"

(현재 디렉터리를 체크아웃하고, 그 결과를 구문 분석이 쉬운 간단한

형식으로 출력하며, 지정된 문자열로 행을 시작합니다.)

== CMD\_DESCRIPTION\_CHECKSELECTORSYNTAX ==

선택기의 구문을 확인합니다.

== CMD\_USAGE\_CHECKSELECTORSYNTAX ==

사용법:

cm ^checkselectorsyntax | ^css --^file=<선택기 파일>

(선택기 파일 구문을 확인합니다.)

^cat <선택기 파일> | cm ^checkselectorsyntax | ^css -

(UNIX: 표준 입력에서 선택기 파일을 확인합니다.)

^type <선택기 파일> | cm ^checkselectorsyntax | ^css -

(Windows: 표준 입력에서 선택기 파일을 확인합니다.)

--^file 선택기를 읽을 파일.

== CMD\_HELP\_CHECKSELECTORSYNTAX ==

비고:

이 명령은 선택기를 파일이나 표준 입력에서 읽고

유효한 구문이 있는지 검사합니다. 구문 검사에 실패하는 경우, 그 이유가

표준 출력에 출력됩니다.

예:

cm ^checkselectorsyntax --^file=myselector.txt

('myselector.txt' 파일의 구문을 검사합니다.)

^cat myselector.txt | cm ^checkselectorsyntax

(표준 입력에서 'myselector.txt'의 구문을 검사합니다.)

== CMD\_DESCRIPTION\_CHGREVTYPE ==

항목 개정 유형(바이너리 또는 텍스트)을 변경합니다.

== CMD\_USAGE\_CHGREVTYPE ==

사용법:

cm ^changerevisiontype | ^chgrevtype | ^crt <항목 경로>[ ...] --^type=(^bin | ^txt)

항목 경로 개정 유형을 변경할 항목. 공백이 포함된 경로를

지정하려면 큰따옴표(" ")를 사용하십시오. 항목 경로를 구분하려면

공백을 사용하십시오.

--^type 대상 개정 유형. '^bin' 또는 '^txt' 중에 선택하십시오.

== CMD\_HELP\_CHGREVTYPE ==

비고:

이 명령은 디렉터리가 아닌 파일에만 적용할 수 있습니다.

지정된 유형은 시스템에서 지원하는 '^bin' 또는 '^txt'(바이너리 또는

텍스트)여야 합니다.

예:

cm ^changerevisiontype c:\workspace\file.txt --^type=^txt

('file.txt'의 개정 유형을 텍스트로 변경합니다.)

cm ^chgrevtype comp.zip "image file.jpg" --^type=^bin

('comp.zip' 및 "image file.jpg"의 개정 유형을 바이너리로 변경합니다.)

cm ^crt \*.\* --^type=^txt

(모든 파일의 개정 유형을 텍스트로 변경합니다.)

== CMD\_DESCRIPTION\_TRIGGER\_EDIT ==

트리거를 편집합니다.

== CMD\_USAGE\_TRIGGER\_EDIT ==

사용법:

cm ^trigger | ^tr ^edit <하위 유형 유형> <위치 번호>

[--^position=<새 위치>]

[--^name=<새 이름>] [--^script=<스크립트 경로>]

[--^filter=<문자열 필터>] [--^server=<리포지토리 서버 사양>]

하위 유형 유형 트리거 실행 및 트리거 작업.

트리거 유형을 확인하려면 'cm ^showtriggertypes'를

입력하십시오.

위치 번호 수정할 트리거가 점유하는 위치.

옵션:

--^position 지정된 트리거의 새 위치.

이 위치를 같은 유형의 다른 트리거가

사용 중이면 안 됩니다.

--^name 지정된 트리거의 새 이름.

--^script 지정된 트리거 스크립트의 새 실행 경로.

스크립트가 "^webtrigger "로 시작하는 경우,

웹 트리거로 간주됩니다. 추가 세부사항을 보려면

비고를 확인하십시오.

--^filter 지정된 필터와 일치하는 항목만 확인합니다.

--^server 지정된 서버의 트리거를 수정합니다.

서버가 지정되지 않은 경우, 클라이언트에 구성된 서버를

대상으로 명령을 실행합니다.

(서버 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을

사용하십시오.)

== CMD\_HELP\_TRIGGER\_EDIT ==

비고:

웹 트리거: "^webtrigger <대상 URI>"를 트리거 명령으로 입력하면

트리거가 생성됩니다. 이 경우, 트리거는 지정된 URI에 대해 POST 쿼리를

실행합니다. 이 URI의 요청 본문에는 트리거 환경 변수가 있는

JSON 사전과 문자열 배열을 가리키는

고정 INPUT 키가 포함됩니다.

예:

cm ^trigger ^edit ^after-setselector 6 --^name="Backup2 관리자" --^script="/new/path/al/script"

cm ^tr ^edit ^before-mklabel 7 --^position=4 --^server=myserver:8084

cm ^trigger ^edit ^after-add 2 --^script="^webtrigger http://myserver.org/api"

== CMD\_DESCRIPTION\_CODEREVIEW ==

코드 검토를 생성, 편집, 삭제합니다.

== CMD\_USAGE\_CODEREVIEW ==

사용법:

cm ^codereview <사양> <제목> [--^status=<상태 이름>]

[--^assignee=<사용자 이름>] [--^format=<문자열 형식>]

[--^repository=<리포지토리 사양>]

(코드 검토를 생성합니다.)

cm ^codereview -^e <id> [--^status=<상태 이름>] [--^assignee=<사용자 이름>]

[--^repository=<리포지토리 사양>]

(코드 검토를 편집합니다.)

cm ^codereview -^d <ID> [ ...] [--^repository=<리포지토리 사양>]

(하나 이상의 코드 검토를 삭제합니다.)

사양 체인지 세트 사양이거나 브랜치 사양일 수 있으며

새 코드 검토의 대상이 됩니다. (체인지 세트 사양

또는 브랜치 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을

사용하십시오.)

제목 새 코드 검토의 제목으로 사용될

텍스트 문자열.

ID 코드 검토 식별 번호. GUID를 사용할 수도

있습니다.

옵션:

-^e 기존 코드 검토의 매개 변수를 편집합니다.

-^d 하나 이상의 기존 코드 검토를 삭제합니다. 공백을

사용하여 코드 검토 ID를 구분하십시오.

--^status 코드 검토의 새 상태를 설정합니다. 자세한 내용은

비고를 참조하십시오.

--^assignee 코드 검토의 새 할당 대상을 설정합니다.

--^format 특정 형식으로 출력 메시지를 가져옵니다. 자세한 내용은

비고를 참조하십시오.

--^repository 기본값으로 사용할 리포지토리를 설정합니다. (리포지토리

사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을

사용하십시오.)

== CMD\_HELP\_CODEREVIEW ==

비고:

이 명령을 사용하면 사용자는 코드 검토를 관리할 수 있습니다. 즉, 체인지 세트나 브랜치에 대한

코드 검토를 생성, 편집, 삭제할 수 있습니다.

새 코드 검토를 생성하려면 체인지 세트/브랜치 사양과 제목이

필요합니다. 초기 상태와 할당 대상도 설정할 수 있습니다. 코드 검토 ID(또는

요청된 경우 GUID)가 결과로 반환됩니다.

기존 코드 검토를 편집하거나 삭제하려면 대상 코드 검토 ID(또는 GUID)가

필요합니다. 오류가 발생하지 않으면 메시지가 표시되지 않습니다.

상태 매개 변수는 ^"Under review"(기본값),

^"Reviewed", ^"Rework required" 중 하나여야 합니다.

리포지토리 매개 변수를 사용하여 기본 작업 리포지토리를 설정할 수

있습니다. 사용자가 현재 워크스페이스에 연결된 서버와 다른 서버에 대한

검토를 관리하려 하거나, 현재 워크스페이스가 전혀 없는 경우에

사용하면 좋습니다.

출력 형식 커스터마이즈:

이 명령은 출력을 표시할 형식 문자열을 허용합니다.

이 명령의 출력 매개 변수는 다음과 같습니다.

{0} ID

{1} GUID

참고로 --^format' 매개 변수는 새 코드 검토를 생성하는 경우에만

적용됩니다.

예:

cm ^codereview ^cs:1856@myrepo@myserver:8084 "내 코드 검토" --^assignee=더미

cm ^codereview ^br:/main/task001@myrepo@myserver:8084 "내 코드 검토" \

--^status=^"Rework required" --^assignee=신입 --^format="{^id} -> {^guid}"

cm ^codereview 1367 -^e --^assignee=새 할당 대상

cm ^codereview -^e 27658884-5dcc-49b7-b0ef-a5760ae740a3 --^status=검토됨

cm ^codereview -^d 1367 --^repository=myremoterepo@myremoteserver:18084

cm ^codereview 27658884-5dcc-49b7-b0ef-a5760ae740a3 -^d

== CMD\_DESCRIPTION\_CRYPT ==

비밀번호를 암호화합니다.

== CMD\_USAGE\_CRYPT ==

사용법:

cm ^crypt <내 비밀번호>

내 비밀번호 암호화할 비밀번호.

== CMD\_HELP\_CRYPT ==

비고:

이 명령은 인수로 전달된 주어진 비밀번호를 암호화합니다.

구성 파일에 비밀번호를 암호화하고 보안 수준을 향상시키기 위해

만들어졌습니다.

예:

cm ^crypt DB 구성 비밀번호 -> 암호화됨: 암호화된 DB 구성 비밀번호

(데이터베이스 구성 파일인 'db.conf'의 비밀번호를 암호화합니다.)

== CMD\_DESCRIPTION\_DEACTIVATEUSER ==

라이센스 사용자를 비활성화합니다.

== CMD\_USAGE\_DEACTIVATEUSER ==

사용법:

cm ^deactivateuser | ^du <사용자 이름>[ ...] [--^server=<이름:포트>]

[--^nosolveuser]

사용자 이름 비활성화할 사용자 이름. 사용자 이름을 구분하려면

공백을 사용하십시오.

SID인 경우, --^nosolveuser'가 필요합니다.

옵션:

--^server 지정된 서버의 사용자를 비활성화합니다.

서버가 지정되지 않은 경우, 클라이언트에 구성된 서버를

대상으로 명령을 실행합니다.

--^nosolveuser 이 옵션을 사용하면 명령은

해당 사용자 이름이 인증 시스템에 존재하는지 여부를 확인하지 않습니다. 이 경우,

<사용자 이름>은 반드시 사용자 SID여야 합니다.

== CMD\_HELP\_DEACTIVATEUSER ==

비고:

이 명령은 사용자를 비활성 상태로 설정하여 해당 사용자가 Plastic SCM을

사용할 수 없게 만듭니다.

Plastic SCM 사용자 활성화에 관한 자세한 내용은 'cm ^activateuser' 명령을

확인하십시오.

이 명령은 사용자가 기본 인증 시스템(예: ActiveDirectory, LDAP, 사용자/비밀번호 등)에

존재하는지 확인합니다.

더 이상 인증 시스템에 존재하지 않는 사용자를 강제로

비활성화하려면 --^nosolveuser' 옵션을 사용하면 됩니다.

예:

cm ^deactivateuser john

cm ^du peter "mary collins"

cm ^deactivateuser john --^server=myserver:8084

cm ^deactivateuser S-1-5-21-3631250224-3045023395-1892523819-1107 --^nosolveuser

== CMD\_DESCRIPTION\_DIFF ==

파일, 체인지 세트, 레이블 간의 차이점을 표시합니다.

== CMD\_USAGE\_DIFF ==

사용법:

cm ^diff <체인지 세트 사양> | <레이블 사양> | <임시 저장 사양> [<체인지 세트 사양> | <레이블 사양> | <임시 저장 사양>]

[<경로>]

[--^added] [--^changed] [--^moved] [--^deleted]

[--^repositorypaths] [--^download=<다운로드 경로>]

[--^encoding=<이름>]

[--^ignore=(^eol | ^whitespaces | ^"eol&whitespaces" | ^none)]

[--^clean]

[--^format=<문자열 형식>] [--^dateformat=<문자열 형식>]

'소스' 체인지 세트 또는 임시 저장 세트와 '대상' 체인지 세트 또는

임시 저장 세트의 차이점을 표시합니다. 체인지 세트 사양이나 레이블 사양을

사용해 체인지 세트를 지정할 수 있습니다.

두 가지 사양이 주어지면, 첫 번째 사양이 비교의 '소스'가 되고

두 번째 사양은 '대상'이 됩니다.

사양이 하나만 주어지면 '소스'는 지정된 '대상'의

부모 체인지 세트가 됩니다.

임의의 경로가 지정된 경우 비교 창이 실행되어

해당 파일의 두 개정 간의 차이점을 표시합니다.

cm ^diff <개정 사양 1> <개정 사양 2>

두 개정 간의 차이점을 표시합니다. 차이점은 비교 창에

표시됩니다. 지정된 첫 번째 개정이 왼쪽에

표시됩니다.

cm ^diff <브랜치 사양> [--^added] [--^changed] [--^moved] [--^deleted]

[--^repositorypaths] [--^download=<다운로드 경로>]

[--^encoding=<이름>]

[--^ignore=(^eol | ^whitespaces | ^"eol&whitespaces" | ^none)]

[--^clean]

[--^format=<문자열 형식>] [--^dateformat=<문자열 형식>]

[--^fullpaths | --^fp]

브랜치 차이점을 표시합니다.

(사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을 사용하십시오.)

옵션:

--^added 리포지토리에 추가된 항목으로 구성된 차이점만

출력합니다.

--^changed 변경된 항목으로 구성된 차이점만

출력합니다.

--^moved 이동됐거나 이름이 변경된 항목으로 구성된 차이점만

출력합니다.

--^deleted 삭제된 항목으로 구성된 차이점만

출력합니다.

'--^added', '--^changed', '--^moved' 또는 '--^deleted'가

지정되지 않은 경우, 해당 명령은 모든 차이점을 출력합니다.

'^A'는 추가된 항목을 의미합니다.

'^C'는 변경된 항목을 의미합니다.

'^D'는 삭제된 항목을 의미합니다.

'^M'은 이동된 항목을 의미합니다. 왼쪽 항목은 원본이며,

오른쪽 항목은 대상입니다.

--^repositorypaths 워크스페이스 경로 대신 리포지토리 경로를 출력합니다.

(이 옵션은 '--^fullpaths' 옵션을 재정의합니다.)

--^download 지정된 출력 경로에 차이점 내용을

저장합니다.

--^encoding 출력 인코딩, 즉 UTF-8을 지정합니다.

지원되는 인코딩 및 형식에 대한 표를 확인하려면

http://msdn.microsoft.com/en-us/library/system.text.encoding.aspx에서

MSDN 문서를 확인하십시오.

페이지 마지막 "Name"열에 있습니다.

--^ignore 지정된 비교 방법을 설정합니다.

자세한 내용은 비고를 확인하십시오.

--^clean 병합으로 인해 생겨난 차이점은 고려하지 않으며

단순 체크인으로 생성된 차이점만

고려합니다.

--^format 특정 형식으로 출력 메시지를 가져옵니다. 자세한 내용은

비고를 확인하십시오.

--^dateformat 날짜 출력에 사용되는 형식.

--^fullpaths, --^fp 가능한 경우 강제로 파일 및 디렉터리에 대한 전체

워크스페이스 경로를 출력합니다.

== CMD\_HELP\_DIFF ==

비고:

비교 방법:

^eol 행 끝 차이점을 무시합니다.

^whitespaces 공백 차이점을 무시합니다.

^"eol&whitespaces" 행 끝 및 공백 차이점을 무시합니다.

^none 행 끝 및 공백 차이점을 감지합니다.

이 명령은 출력을 표시할 형식 문자열을 허용합니다.

이 명령의 매개 변수는 다음과 같습니다.

{^path} 항목 경로.

{^date} 날짜/시간 변경.

{^owner} 작성자 변경.

{^revid} 비교에서 대상으로 간주되는

개정의 개정 ID.

{^parentrevid} 비교의 대상으로 간주되는 개정 부모의

개정 ID.

{^baserevid} 비교에서 소스로 간주되는

개정의 개정 ID.

{^srccmpath} 항목을 이동(이동 작업)하기 전 서버 경로.

{^dstcmpath} 항목을 이동(이동 작업)한 이후의 서버 경로.

{^type} 항목 유형:

^D 디렉터리,

^B 바이너리 파일,

^F 텍스트 파일,

^S 기호 링크,

^X Xlink.

{^repository} 항목의 리포지토리.

{^status} 항목 상태:

^A 추가됨,

^D 삭제됨,

^M 이동됨,

^C 변경됨.

{^fsprotection} 항목 권한을 표시합니다(Linux/Mac chmod).

{^srcfsprotection} 부모 개정 항목 권한을 표시합니다.

{^newline} 새 행을 삽입합니다.

'^revid'에 관한 참고 사항:

추가된 항목의 경우, 이전 개정이 없기 때문에 '^baserevid'와

'^parentrevid'가 -1이 됩니다.

삭제된 항목의 경우, 대상 개정이 없기 때문에 '^revid'가 소스 개정의 ID가 되고

'^baserevid'가 -1이 됩니다.

Xlink의 경우 '^baserevid'와 '^parentrevid'가 모두 항상 -1입니다.

예:

브랜치 비교:

cm ^diff ^br:/main/task001

cm ^diff ^br:/main/task001 \doc\readme.txt

체인지 세트 트리 비교:

cm ^diff 19

cm ^diff 19 25

cm ^diff ^cs:19 ^cs:25 --^format="{^path} {^parentrevid}"

cm ^diff ^cs:19 ^cs:23 --^format="{^date} {^path}" --^dateformat="yy/dd/MM HH:mm:ss"

cm ^diff ^cs:19 ^cs:23 --^changed

cm ^diff ^cs:19 ^cs:23 --^repositorypaths

cm ^diff ^cs:19 ^cs:23 --^download="D:\temp"

cm ^diff ^cs:19 ^cs:23 --^clean

cm ^diff ^cs:19 ^cs:23 \doc\readme.txt

레이블 트리 비교:

cm ^diff ^lb:FirstReleaseLabel ^lb:SecondReleaseLabel

cm ^diff ^lb:tag\_193.2 ^cs:34214

cm ^diff ^cs:31492 ^lb:tag\_193.2

임시 저장 트리 비교:

cm ^diff ^sh:2

cm ^diff ^sh:2 ^sh:4

개정 사양 비교:

cm ^diff ^rev:readme.txt#^cs:19 ^rev:readme.txt#^cs:20

cm ^diff ^serverpath:/doc/readme.txt#^cs:19@myrepo \

^serverpath:/doc/readme.txt#^br:/main@myrepo@localhost:8084

== CMD\_DESCRIPTION\_DIFFMETRICS ==

두 개정 간의 비교 지표를 표시합니다.

== CMD\_USAGE\_DIFFMETRICS ==

사용법:

cm ^diffmetrics | ^dm <개정 사양 1> <개정 사양 2> [--^format=<문자열 형식>]

[--^encoding=<이름>]

[--^ignore=(^eol | ^whitespaces | ^"eol&whitespaces" | ^none)]

개정 사양 비교에 사용되는 개정.

(개정 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을 사용하십시오.)

옵션:

--^format 특정 형식으로 출력 메시지를 가져옵니다. 자세한 내용은

비고를 확인하십시오.

--^encoding 출력 인코딩, 즉 UTF-8을 지정합니다.

지원되는 인코딩 및 형식에 대한 표를 확인하려면

http://msdn.microsoft.com/en-us/library/system.text.encoding.aspx에서

MSDN 문서를 확인하십시오.

페이지 마지막 "Name"열에 있습니다.

--^ignore 지정된 비교 방법을 설정합니다.

자세한 내용은 비고를 확인하십시오.

== CMD\_HELP\_DIFFMETRICS ==

비고:

지표는 변경, 추가, 삭제된 행의 수입니다.

이 명령은 출력을 표시할 형식 문자열을 허용합니다.

이 명령의 출력 매개 변수는 다음과 같습니다.

{0} 변경된 행의 수.

{1} 추가된 행의 수.

{2} 삭제된 행의 수.

예:

cm ^diffmetrics file.txt#^cs:2 file.txt#^br:/main/scm0211 \

--^format="변경된 행 {0}개, 추가된 행 {1}개, 삭제된 행 {2}개가 있습니다."

(형식화된 비교 지표 결과를 가져옵니다.)

cm ^dm file.txt#^cs:2 file.txt#^cs:3 --^encoding=utf-8 --^ignore=^whitespaces

== CMD\_DESCRIPTION\_FASTEXPORT ==

빠른 익스포트 형식으로 리포지토리를 익스포트합니다.

== CMD\_USAGE\_FASTEXPORT ==

사용법:

cm ^fast-export | ^fe <리포지토리 사양> <빠른 익스포트 파일>

[--^import-marks=<표시 파일>]

[--^export-marks=<표시 파일>]

[--^branchseparator=<문자 구분 기준>]

[--^nodata] [--^from=<체인지 세트 ID>] [--^to=<체인지 세트 ID>]

옵션:

리포지토리 사양 데이터를 익스포트할 리포지토리.

(리포지토리 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을 사용하십시오.)

빠른 익스포트 파일 Git 빠른 익스포트 형식의 리포지토리 데이터가 있는

파일.

--^import-marks 증분 임포트에 사용되는 표시 파일. 이 파일은

이전에 '--^export-marks'로 익스포트된 파일입니다. 이

파일에 설명된 체인지 세트는 이미 이전 임포트에 있었기 때문에

임포트되지 않습니다.

--^export-marks 임포트된 체인지 세트가 저장될 파일.

이 파일은 이미 임포트된 체인지 세트에 신호를 보내기 위해

다음 빠른 임포트에서 사용됩니다.

--^branchseparator Plastic SCM은 브랜치 계층에서 "/" 기호를 기본 구분 기준으로

사용합니다. 이 옵션을 사용하면 문자를 계층 구분 기준으로

사용할 수 있으므로, main-task-sub가 Plastic

SCM에서 /main/task/sub로 매핑됩니다.

--^nodata 데이터를 포함하지 않고 리포지토리를 익스포트합니다. 이는

익스포트가 올바르게 실행되는지 확인하는 데 유용합니다.

--^from 특정 체인지 세트에서 익스포트합니다.

--^to 특정 체인지 세트로 익스포트합니다.

== CMD\_HELP\_FASTEXPORT ==

비고:

- Plastic SCM 리포지토리를 Git으로 임포트하려면 다음과 같은 명령을 사용하십시오.

^cat repo.fe.00 | ^git ^fast-import --^export-marks=marks.git --^import-marks=marks.git

- 증분 익스포트는 이전에 임포트된 체인지 세트를

포함하는 표시 파일('--^import-marks' 및 '--^export-marks'

파일)을 사용해 지원됩니다.

즉, 이전 빠른 익스포트에서 익스포트되지 않은

새로운 체인지 세트만 익스포트됩니다.

예:

cm ^fast-export repo@localhost:8087 repo.fe.00 --^import-marks=marks.cm \

--^export-marks=marks.cm

(로컬 서버에 있는 리포지토리 'repo'를 Git 빠른 익스포트 형식으로 'repo.fe.00'으로

익스포트하고 다음에 증분 익스포트를 수행할 수 있도록

표시 파일을 생성합니다.)

cm ^fast-export repo@localhost:8087 repo.fe.00 --^from=20

(로컬 서버에 있는 리포지토리 'repo'를 Git 빠른 익스포트 형식으로

체인지 세트 '20'에서 'repo.fe.00'으로 익스포트합니다.)

== CMD\_DESCRIPTION\_FASTIMPORT ==

Git 빠른 익스포트 데이터를 리포지토리로 임포트합니다.

== CMD\_USAGE\_FASTIMPORT ==

사용법:

cm ^fast-import | ^fi <리포지토리 사양> <빠른 익스포트 파일>

[--^import-marks=<표시 파일>]

[--^export-marks=<표시 파일>]

[--^stats] [--^branchseparator=<문자 구분 기준>]

[--^nodata] [--^ignoremissingchangesets] [--^mastertomain]

옵션:

리포지토리 사양 데이터를 임포트할

리포지토리. 해당 리포지토리가 기존에 없었던 경우에

생성됩니다. (리포지토리 사양에 관한 자세한 내용을 확인하려면

'cm ^help ^objectspec'을 사용하십시오.)

빠른 익스포트 파일 Git 빠른 익스포트 형식의 리포지토리

데이터가 있는 파일.

--^import-marks 증분 임포트에 사용되는 표시 파일.

이 파일은 이전에 '--^export-marks'로

익스포트된 파일입니다. 이 파일에 설명된 체인지 세트는

이미 이전 임포트에 있었기 때문에

임포트되지 않습니다.

--^export-marks 임포트된 체인지 세트가 저장될

파일. 이 파일은 이미 임포트된

체인지 세트에 신호를 보내기 위해 다음 빠른 임포트에서

사용됩니다.

--^stats 임포트 프로세스에 관한 일부 통계를 출력합니다.

--^branchseparator Plastic SCM은 브랜치 계층에서 "/" 기호를 기본 구분 기준으로

사용합니다. 이 옵션을 사용하면 문자를

계층 구분 기준으로 사용할 수 있으므로, main-task-sub는

Plastic SCM에서 /main/task/sub로 매핑됩니다.

--^nodata 데이터를 포함하지 않고 Git 빠른 익스포트를

임포트합니다. 이는 임포트가 올바르게 실행되는지 확인하는 데

유용합니다.

--^ignoremissingchangesets 임포트할 수 없는 모든 체인지 세트는

무시되며 빠른 임포트 작업이 해당 체인지 세트 없이

계속 진행됩니다.

--^mastertomain "^master" 대신 "^main"을 사용하여 임포트합니다.

== CMD\_HELP\_FASTIMPORT ==

비고:

- git 리포지토리를 익스포트하려면 다음과 같은 명령을 사용하십시오.

^git ^fast-export --^all -^M --^signed-tags=^strip --^tag-of-filtered-object=^drop> ..\git-fast-export.dat

-^M은 이동된 항목을 감지할 때 중요합니다.

- 현재 없는 경우 지정된 리포지토리가 생성됩니다.

- 증분 임포트는 이전에 임포트된 체인지 세트를

포함하는 표시 파일('--^import-marks' 및 '--^export-marks'

파일)을 사용해 지원됩니다.

즉, 이전 빠른 임포트에서 임포트되지 않은

새로운 체인지 세트만 임포트됩니다.

예:

cm ^fast-import mynewrepo@atenea:8084 repo.fast-export

('repo.fast-export' 파일에서 익스포트된 내용을

서버 'atenea:8084'의 'mynewrepo' 리포지토리로 임포트합니다.)

cm ^fast-import repo@atenea:8084 repo.fast-export --^export-marks=rep.marks

('repo.fast-export' 파일에서 익스포트된 내용을

서버 'atenea:8084'의 'repo' 리포지토리로 임포트하고 다음에 증분 임포트를 수행할 수 있도록

표시 파일을 생성합니다.)

cm ^fast-import repo@server:8084 repo.fast-export --^import-marks=repo.marks \

--^export-marks=repo.marks

('repo.fast-export' 파일의 내용을 임포트합니다. 표시 파일에 없는

새로운 체인지 세트만 임포트됩니다. 다음 증분 임포트를 위한

체인지 세트 목록을 다시 저장할 때 같은 표시 파일이

사용됩니다.)

== CMD\_DESCRIPTION\_FILEINFO ==

워크스페이스의 항목에 관한 자세한 정보를 가져옵니다.

== CMD\_USAGE\_FILEINFO ==

사용법:

cm ^fileinfo <항목 경로>[ ...] [--^fields=<필드 값>[,...]]

[[--^xml | -^x [=<출력 파일>]] | [--^format=<문자열 형식>]]

[--^symlink] [--^encoding=<이름>]

항목 경로 표시할 항목. 공백을 사용하여 항목을

구분하십시오.

공백이 포함된 경로를 지정하려면 큰따옴표(" ")를

사용하십시오.

옵션:

--^fields 쉼표로 구분된 값의 문자열. 각 항목에 대해 출력할

필드를 선택합니다. 자세한 내용은

비고를 확인하십시오.

--^xml | -^x XML 형식의 출력을 표준 출력으로 출력합니다.

출력 파일을 지정할 수 있습니다. 이 옵션은

'--^format'과 함께 사용할 수 없습니다.

--^format 특정 형식으로 출력 메시지를 가져옵니다. 자세한 내용은

비고를 확인하십시오. 이 옵션은 '--^xml'과 함께 사용할 수

없습니다.

두 옵션이 지정된 경우 이 '--^format' 옵션이 '--^fields' 옵션에

우선합니다.

--^symlink 대상이 아닌 기호 링크에 파일 정보 작업을

적용합니다.

--^encoding 출력 인코딩, 즉 UTF-8을 지정합니다.

지원되는 인코딩 및 형식에 대한 표를 확인하려면

http://msdn.microsoft.com/en-us/library/system.text.encoding.aspx에서

MSDN 문서를 확인하십시오.

페이지 마지막 "Name"열에 있습니다.

== CMD\_HELP\_FILEINFO ==

비고:

이 명령은 선택한 각 항목에 대한 세부적인 속성 목록을 출력합니다.

각 속성은 기본적으로 새 행에 출력됩니다.

사용자가 필요한 속성만 표시하도록 속성 목록을 수정할 수

있습니다. '--^fields=<필드 목록>'을 사용하면 목록을 수정할 수 있으며, 이 경우 쉼표로 구분된

속성 이름 문자열을 사용할 수 있습니다. 이렇게 하면 자신의 이름이 명시된

인수만 표시됩니다.

개정 헤드 체인지 세트:

이 옵션은 기본적으로 비활성화되어 있습니다. 이 속성은 다른 속성에 비해

가져올 때 시간이 훨씬 많이 소요되므로

최대한 많은 항목을 그룹화하는 것이 좋습니다. 그러면 많은 수의 'cm ^fileinfo'가

별도로 실행되는 것을 방지할 수 있어 실행 시간을 단축할 수 있습니다.

또한 현재 이 기능은 제어되는 디렉터리에 대해 사용할 수 없습니다.

아래에서 사용할 수 있는 전체 속성 이름 목록을 확인할 수 있습니다.

별표('\*')로 표시된 이름은 기본적으로 표시되지 않습니다.

^ClientPath 항목의 디스크 로컬 경로.

^RelativePath 워크스페이스 관련 경로.

^ServerPath 항목의 리포지토리 경로.

(참고: 변환된 워크스페이스는

현재 이 옵션에서 지원되지 않습니다.)

^Size 항목 크기.

^Hash 항목 해시 합계.

^Owner 해당 항목을 소유한 사용자.

^RevisionHeadChangeset (\*) 브랜치의 헤드 체인지 세트에 로드된

개정 체인지 세트.

(위의 참고를 참조하십시오.)

^RevisionChangeset 현재 워크스페이스에 로드된

개정 체인지 세트.

^RepSpec 항목의 리포지토리 사양.

(리포지토리 사양에 관한 자세한 내용을 확인하려면

'cm ^help ^objectspec'을 사용하십시오.)

^Status 워크스페이스 항목 상태(추가됨, 체크아웃됨,

삭제됨 등).

^Type 개정 유형(텍스트, 바이너리, 디렉터리, 기호 링크,

알 수 없음).

^Changelist 해당 항목을 소유하는 변경사항 목록(해당하는 경우).

^IsLocked (\*) 배타적 체크아웃으로 인한 항목의

잠금 여부.

^LockedBy (\*) 해당 항목을 배타적으로 체크아웃한 사용자.

^LockedWhere (\*) 항목이 배타적으로 체크아웃된

위치.

^IsUnderXlink 항목이 Xlink에 있는지

여부.

^UnderXlinkTarget 항목이 있는 Xlink의 대상(해당하는

경우).

^UnderXlinkPath Xlink 리포지토리의 항목 서버 경로

(해당하는 경우).

^UnderXlinkWritable 항목이 속한 Xlink의 쓰기 가능

여부.

^UnderXlinkRelative 항목이 속한 Xlink가 상대 Xlink인지

여부.

^IsXlink 항목 자체가 Xlink인지 여부.

^XlinkTarget 항목이 Xlink인 경우, 항목이 가리키는

대상 리포지토리.

^XlinkName 항목이 실제로 한 개인 경우,

항목의 Xlink 이름.

^XlinkWritable Xlink 항목이 쓰기 가능한 Xlink인지

여부.

^XlinkRelative Xlink 항목이 상대 Xlink인지

여부.

출력 형식 커스터마이즈:

이 명령은 출력을 표시할 형식 문자열을 허용합니다.

이 명령의 출력 매개 변수는 다음과 같습니다.

{^ClientPath}

{^RelativePath}

{^ServerPath}

{^Size}

{^Hash}

{^Owner}

{^RevisionHeadChangeset}

{^RevisionChangeset}

{^Status}

{^Type}

{^Changelist}

{^IsLocked}

{^LockedBy}

{^LockedWhere}

{^IsUnderXlink}

{^UnderXlinkTarget}

{^UnderXlinkPath}

{^UnderXlinkWritable}

{^UnderXlinkRelative}

{^IsXlink}

{^XlinkTarget}

{^XlinkName}

{^XlinkWritable}

{^XlinkRelative}

{^RepSpec}

'--^format' 옵션과 '--^xml' 옵션은 상호 배타적이므로,

함께 사용할 수 없습니다.

예:

cm ^fileinfo file1.txt file2.txt dir/

cm ^fileinfo "New Project.csproj" --^xml

cm ^fileinfo assets.art --^fields=^ServerPath,^Size,^IsLocked,^LockedBy

cm ^fileinfo proj\_specs.docx --^fields=^ServerPath,^RevisionChangeset --^xml

cm ^fileinfo samples.ogg --^format="{^ServerPath}[{^Owner}] -> {^Size}"

== CMD\_DESCRIPTION\_FIND\_QUERY ==

SQL과 같은 쿼리를 실행하여 Plastic SCM 객체를 찾습니다.

== CMD\_USAGE\_FIND\_QUERY ==

사용법:

cm ^find <객체 유형>

[^where <문자열 조건>]

[^on ^repository '<리포지토리 사양>' | ^on ^repositories '<리포지토리 사양 1>','<리포지토리 사양 2>'[,...]]

[--^format=<문자열 형식>] [--^dateformat=<날짜 형식>]

[--^nototal] [--^file=<덤프 파일>] [--^xml]

[--^encoding=<이름>]

객체 유형 찾을 객체 유형.

이러한 객체를 지정하는 방법을 알아보려면

'cm ^help ^showfindobjects'를 사용하십시오.

'cm ^find' 가이드를 참조할 수도 있습니다.

https://www.plasticscm.com/download/help/cmfind

옵션:

문자열 조건 객체 속성에서 조건을 검색합니다.

리포지토리 사양 리포지토리 별칭이나 사양을 검색합니다.

'^on ^repositories'의 경우, 쉼표를 사용하여

리포지토리 사양 필드를 구분하십시오.

(리포지토리 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을

사용하십시오.)

--^format 특정 형식으로 출력 메시지를 가져옵니다.

출력 형식 문자열로 사용할 수 있는

모든 객체 속성을 확인하려면 'cm ^find' 가이드를 참조하십시오.

https://www.plasticscm.com/download/help/cmfind

--^dateformat 날짜 출력에 사용되는 형식.

--^nototal 끝에 레코드 수를 출력하지 않습니다.

--^file 결과를 덤프할 파일.

--^xml XML 형식의 출력을 표준 출력으로 출력합니다.

--^encoding 출력 인코딩, 즉 UTF-8을 지정합니다.

지원되는 인코딩 및 형식에 대한 표를 확인하려면

http://msdn.microsoft.com/en-us/library/system.text.encoding.aspx에서

MSDN 문서를 확인하십시오.

페이지 마지막 "Name"열에 있습니다.

== CMD\_HELP\_FIND\_QUERY ==

비고:

리포지토리가 지정되지 않으면 워크스페이스에 구성된

리포지토리에서 검색을 수행합니다.

명령줄에서 비교 연산자(>, <, >=, <=)를 사용하여 쿼리를 실행할 때

셸은 이러한 연산자를 IO 리디렉션으로

간주합니다. 따라서 쿼리를 큰 따옴표로 묶어야

합니다.

'cm ^find' 명령은 출력을 표시할 형식 문자열을 허용합니다.

각 출력 매개 변수는 문자열로 식별되며 사용자는 '{' 및 '}' 괄호 사이에

매개 변수 번호를 입력하여 참조할 수 있습니다.

출력 매개 변수는 일반적으로 객체의 속성에 해당합니다.

다음은 몇 가지 유효한 형식 문자열입니다.

--^format={^id}{^date}{^name}

--^format="{^item}#{^branch} ^with ^date {^date}"

XML 및 인코딩 고려 사항:

'--^xml' 옵션이 지정되면, 명령은 명령 결과를 표준 출력에

XML 텍스트로 표시합니다. 이 텍스트를 표시하는 데 운영 체제

기본 인코딩이 사용되므로, ANSI가 아닌 문자의 경우 콘솔에서

잘못된 형태로 표시될 수 있습니다. 명령 출력을 파일로 리디렉션하면

올바르게 표시됩니다. '--^xml' 옵션과 '--^file' 옵션이

모두 지정된 경우, 기본 인코딩으로 UTF-8을 사용합니다.

예:

cm ^find ^revision

cm ^find ^revision "^where ^changeset=23 ^and ^owner='maria'"

cm ^find ^branch "^on ^repository 'rep1'"

cm ^find ^label "^on ^repositories 'rep1', '^rep:default@localhost:8084'"

cm ^find ^branch "^where ^parent='^br:/main' ^on ^repository 'rep1'"

cm ^find ^revision "^where ^item='^item:.'" --^format="{^item}#{^branch}"

cm ^find ^revision "^where ^item='^item:.'" --^xml --^file=c:\queryresults\revs.xml

== CMD\_DESCRIPTION\_FINDCHANGED ==

변경된 파일 목록을 가져옵니다. 이 명령은 더 이상 사용되지 않으며, 이전 버전과의

호환성을 위해 유지되고 있습니다. 'cm ^status'를 대신 사용하십시오.

== CMD\_USAGE\_FINDCHANGED ==

사용법:

cm ^findchanged | ^fc [-^R | -^r | --^recursive] [--^checkcontent]

[--^onlychanged] [<경로>]

옵션:

-^R 디렉터리에서 재귀적으로 찾습니다.

--^checkcontent 내용별로 파일을 비교합니다.

--^onlychanged 변경된 파일만 찾습니다. 체크아웃은 검색되지

않습니다.

경로 (기본값: 현재 디렉터리.)

변경된 파일을 찾을 초기 경로.

== CMD\_HELP\_FINDCHANGED ==

비고:

'--^checkcontent' 옵션이 주어지지 않은 경우, Plastic은 파일 타임스탬프를

기반으로 변경사항을 찾습니다. '--^checkcontent' 옵션이 지정된 경우, 타임스탬프를

사용하는 대신 파일이나 폴더의 내용을 비교합니다.

이 명령은 Plastic SCM 서버에서 연결이 끊긴 동안 변경된 파일을

감지하는 데 유용합니다. 나중에 변경사항을 확인하기 위해 출력을 체크아웃 명령으로

파이핑할 수 있습니다(예 참고).

예:

cm ^findchanged .

(현재 디렉터리에서 변경된 파일을 찾습니다.)

cm ^findchanged -^R . | cm ^checkout -

(변경된 요소를 체크아웃합니다.)

== CMD\_DESCRIPTION\_FINDCHECKEDOUT ==

체크아웃된 항목 목록을 가져옵니다. 이 명령은 더 이상 사용되지 않으며, 이전 버전과의

호환성을 위해 유지되고 있습니다. 'cm ^status'를 대신 사용하십시오.

== CMD\_USAGE\_FINDCHECKEDOUT ==

사용법:

cm ^findcheckouts | ^fco [--^format=<문자열 형식>] [--^basepath]

옵션:

--^format 특정 형식으로 출력 메시지를 가져옵니다. 자세한 내용은

비고를 확인하십시오.

--^basepath 체크아웃 검색을 시작할 경로. 지정되지

않은 경우, 현재 경로를 사용합니다.

== CMD\_HELP\_FINDCHECKEDOUT ==

비고:

이 명령은 모든 체크아웃된 항목을 한 번에 체크인하거나, 체크아웃을 실행 취소하여

표준 출력을 다른 명령으로 리디렉션하는 데 유용합니다.

예를 참고하십시오.

이 명령은 출력을 표시할 형식 문자열을 허용합니다.

이 명령의 출력 매개 변수는 다음과 같습니다.

{0} 날짜.

{1} 소유자.

{2} 워크스페이스 정보.

{3} 클라이언트 기기 이름.

{4} 항목 경로.

{5} 브랜치 및 리포지토리 정보.

예:

cm ^findcheckouts --^format="브랜치 {5}에서 {4} 파일이 변경되었습니다."

(체크아웃된 항목을 찾고 출력을 파일 경로와 브랜치 및 리포지토리 정보로

형식화합니다.)

cm ^findcheckouts --^format={4} | cm ^checkin -

(체크아웃된 모든 항목을 체크인합니다.)

cm ^findcheckouts --^format={4} | cm ^undocheckout -

(체크아웃된 모든 항목의 체크아웃을 실행 취소합니다.)

== CMD\_DESCRIPTION\_FINDPRIVATE ==

비공개 항목 목록을 가져옵니다. 이 명령은 더 이상 사용되지 않으며, 이전 버전과의

호환성을 위해 유지되고 있습니다. 'cm ^status'를 대신 사용하십시오.

== CMD\_USAGE\_FINDPRIVATE ==

사용법:

cm ^findprivate | ^fp [-^R | -^r | --^recursive] [--^exclusions] [<경로>]

옵션:

-^R 디렉터리에서 재귀적으로 찾습니다.

--^exclusions 이 옵션을 사용하면 ignore.conf 파일에 정의된

무시된 경로 내에서 검색을 수행하지 않을 수 있습니다.

경로 (기본값: 현재 디렉터리.)

비공개 파일을 찾을 초기 경로.

== CMD\_HELP\_FINDPRIVATE ==

비고:

경로가 지정되지 않은 경우, Plastic SCM은 현재 디렉터리부터 검색을

시작합니다.

이 명령은 폴더에 비공개 항목을 추가하고 출력을 추가 명령으로 파이핑할 때

유용합니다. 예를 참고하십시오.

예:

cm ^findprivate .

cm ^findprivate -^R | cm ^add -

(비공개 항목을 재귀적으로 검색하고 추가합니다.)

== CMD\_DESCRIPTION\_GETCONFIG ==

구성 정보를 가져옵니다.

== CMD\_USAGE\_GETCONFIG ==

사용법:

cm ^getconfig [^setfileasreadonly] [^location] [^extensionworkingmode]

[^extensionprefix] [^defaultrepserver]

^setfileasreadonly 보호된 파일이 읽기 전용으로 남아있는지 여부를

반환합니다.

^location 클라이언트 구성 경로를 반환합니다.

^extensionworkingmode 확장자 작업 모드를 반환합니다.

^extensionprefix 구성된 확장자 접두어를 반환합니다.

^defaultrepserver 기본 리포지토리 서버의 위치를

반환합니다.

== CMD\_HELP\_GETCONFIG ==

예:

cm ^getconfig ^setfileasreadonly

== CMD\_DESCRIPTION\_GETFILE ==

주어진 개정의 내용을 다운로드합니다.

== CMD\_USAGE\_GETFILE ==

사용법:

cm ^getfile | ^cat <개정 사양> [--^file=<출력 파일>] [--^debug]

[--^symlink] [--^raw]

개정 사양 객체 사양. (사양에 관한 자세한 내용을 확인하려면

'cm ^help ^objectspec'을 사용하십시오.)

옵션:

--^file 출력을 저장할 파일. 기본적으로 표준 출력에

출력됩니다.

--^debug 디렉터리 사양을 사용하는 경우, 명령은

디렉터리의 모든 항목, 항목의 개정 ID,

파일 시스템 보호를 표시합니다.

--^symlink 대상이 아닌 기호 링크에 작업을

적용합니다.

--^raw 파일의 원시 데이터를 표시합니다.

== CMD\_HELP\_GETFILE ==

예:

cm ^cat myfile.txt#^br:/main

('myfile.txt'의 브랜치 '^br:/main'에서 마지막 개정을 가져옵니다.)

cm ^getfile myfile.txt#^cs:3 --^file=tmp.txt

('myfile.txt'의 체인지 세트 3을 가져오고 이를 'tmp.txt' 파일에 씁니다.)

cm ^cat ^serverpath:/src/foo.c#^br:/main/task003@myrepo

(리포지토리 'myrepo'에 있는 브랜치 '/main/task003'의 마지막 체인지 세트에서

'/src/foo.c'의 내용을 가져옵니다)

cm ^cat ^revid:1230@^rep:myrep@^repserver:myserver:8084

(ID가 1230인 개정을 가져옵니다.)

cm ^getfile ^rev:info\ --^debug

('info' 디렉터리의 모든 개정을 가져옵니다.)

== CMD\_DESCRIPTION\_GETREVISION ==

워크스페이스에 개정을 로드합니다.

== CMD\_USAGE\_GETREVISION ==

이 명령은 워크스페이스에 로드된 개정을 수정하므로, 향후 병합에

영향을 줄 수 있습니다.

이전 버전에서 상속된 고급 명령이므로 주의해서 사용하십시오.

사용법:

cm ^getrevision <개정 사양>

개정 사양 객체 사양. (개정 사양에 관한 자세한 내용을 확인하려면

'cm ^help ^objectspec'을 사용하십시오.)

== CMD\_HELP\_GETREVISION ==

예:

cm ^getrevision file.txt#^cs:3

('file.txt'의 체인지 세트 3 개정을 가져옵니다.)

== CMD\_DESCRIPTION\_GETSTATUS ==

항목의 상태를 가져옵니다.

== CMD\_USAGE\_GETSTATUS ==

이는 'cm'만 자동화할 때 사용하는 자동화 명령입니다.

주의해야 하는 만큼 사용자 친화적이지 않은 명령입니다.

사용법:

cm ^getstatus | ^gs <항목 경로>[ ...] [--^format=<문자열 형식>] [--^stats]

[-^R | -^r | --^recursive]

항목 경로 상태를 가져올 하나의 또는 여러 항목입니다. 공백이 포함된 경로를

지정하려면 큰따옴표(" ")를 사용하십시오. 공백을

사용하여 경로를 구분하십시오.

옵션:

--^format 특정 형식으로 출력 메시지를 가져옵니다. 자세한 내용은

비고를 확인하십시오.

--^stats 상태를 가져오는 프로세스에 관한 일부 통계를 출력합니다.

-^R 디렉터리에서 상태를 재귀적으로 표시합니다.

== CMD\_HELP\_GETSTATUS ==

비고:

이 명령은 출력을 표시할 형식 문자열을 허용합니다.

이 명령의 출력 매개 변수는 다음과 같습니다.

{0} 항목 경로.

{1} 항목 상태:

0 비공개,

1 체크인됨,

2 체크아웃됨.

표준 입력에서 입력 읽기:

'^getstatus' 명령은 표준 입력에서 경로를 읽을 수 있습니다. 그러려면 단일 대시

"-" 기호를 넣으십시오.

예: cm ^getstatus -

경로 읽기는 빈 행이 입력될 때까지 계속됩니다.

따라서 파이프를 사용해 상태를 가져올 경로를 지정할 수 있습니다.

예:

dir /S /B \*.c | cm ^getstatus --^format="경로 {0} 상태 {1}" -

(Windows에서는 워크스페이스에 있는 모든 .c 파일의 상태를 가져옵니다.)

예:

cm ^getstatus file1.txt file2.txt

(파일의 상태를 가져옵니다.)

cm ^gs info\ -^R --^format="{0} 항목은 {1} 상태입니다"

(디렉터리와 그 모든 항목의 상태를 가져오고 형식에 맞게 출력하여

표시합니다.)

== CMD\_DESCRIPTION\_GETTASKBRANCHES ==

작업과 연결된 브랜치를 가져옵니다.

== CMD\_USAGE\_GETTASKBRANCHES ==

이는 'cm'만 자동화할 때 사용하는 자동화 명령입니다.

주의해야 하는 만큼 사용자 친화적이지 않은 명령입니다.

사용법:

cm ^gettaskbranches | ^gtb <작업 이름> [--^format=<문자열 형식>]

[--^dateformat=<날짜 형식>]

작업 이름 작업 식별자.

옵션:

--^format 특정 형식으로 출력 메시지를 가져옵니다. 자세한 내용은

비고를 확인하십시오.

--^dateformat 날짜 출력에 사용되는 형식.

== CMD\_HELP\_GETTASKBRANCHES ==

비고:

이 명령은 출력을 표시할 형식 문자열을 허용합니다.

이 명령의 출력 매개 변수는 다음과 같습니다.

{^tab} 탭 공백을 삽입합니다.

{^newline} 새 행을 삽입합니다.

{^name} 브랜치 이름.

{^owner} 브랜치의 소유자.

{^date} 브랜치가 생성된 날짜.

{^type} 브랜치 유형(스마트 브랜치인 경우 'T', 아닌 경우 'F').

{^parent} 부모 브랜치.

{^comment} 브랜치 코멘트.

{^repname} 브랜치가 있는 리포지토리.

{^repserver} 서버 이름.

예:

cm ^gettaskbranches 4311

cm ^gtb 4311 --^format="^br:{^name}"

cm ^gtb 4311 --^format="^br:{^name} {^date}" --^dateformat="yyyy/MM/dd HH:mm:ss"

== CMD\_DESCRIPTION\_GETWWI ==

워크스페이스 선택기에 관한 정보를 표시합니다.

== CMD\_USAGE\_GETWWI ==

사용법:

cm ^wi [<워크스페이스 경로>]

옵션:

워크스페이스 경로 기기의 워크스페이스 경로.

== CMD\_HELP\_GETWWI ==

비고:

'^wi' 명령은 워크스페이스(리포지토리, 브랜치, 레이블)의 작업 구성을

표시합니다.

예:

cm ^wi c:\mywk

== CMD\_DESCRIPTION\_GWP ==

경로에서 워크스페이스 정보를 가져옵니다.

== CMD\_USAGE\_GWP ==

이는 'cm'만 자동화할 때 사용하는 자동화 명령입니다.

주의해야 하는 만큼 사용자 친화적이지 않은 명령입니다.

사용법:

cm ^getworkspacefrompath | ^gwp <항목 경로> [--^format=<문자열 형식>]

항목 경로 디스크의 파일이나 폴더.

옵션:

--^format 특정 형식으로 출력 메시지를 가져옵니다. 자세한 내용은

비고를 확인하십시오.

== CMD\_HELP\_GWP ==

비고:

이 명령은 경로에 위치한 워크스페이스에 관한 정보를 표시합니다.

이 명령은 출력을 표시할 형식 문자열을 허용합니다.

이 명령의 출력 매개 변수는 다음과 같습니다.

{0} | {^wkname} 워크스페이스 이름.

{1} | {^wkpath} 워크스페이스 경로.

{2} | {^machine} 클라이언트 기기 이름.

{3} | {^owner} 워크스페이스 소유자.

{4} | {^guid} 워크스페이스 GUID.

{^tab} 탭 공백을 삽입합니다.

{^newline} 새 행을 삽입합니다.

예:

cm ^gwp c:\myworkspace\code\file1.cpp --^format="워크스페이스 이름: {^wkname}"

== CMD\_DESCRIPTION\_HELP ==

Plastic SCM 명령에 관한 도움말을 가져옵니다.

== CMD\_USAGE\_HELP ==

사용법:

cm ^help <명령>

== CMD\_HELP\_HELP ==

== CMD\_DESCRIPTION\_IOSTATS ==

하드웨어에 관한 통계를 표시합니다.

== CMD\_USAGE\_IOSTATS ==

사용법:

cm ^iostats [<리포지토리 서버 사양>] [<테스트 목록>[ ...]]

[--^nettotalmb=<값 MB>] [--^networkiterations=<값 반복>]

[--^diskdatasize=<값 크기>] [--^disktestpath=<값 경로>]

[--^systemdisplaytime=<값 시간>]

[--^systemdisplaytimeinterval=<값 간격>]

옵션:

리포지토리 서버 사양 "serverUploadTest", "serverDownloadTest"와 같은

네트워크 테스트를 수행하기 위해 사용할 수 있는

Plastic SCM 서버.

서버가 제공되지 않은 경우, 명령은

기본값으로 구성된 서버와 연결을

시도합니다.

(서버 사양에 관한 자세한 내용을 확인하려면

'cm ^help ^objectspec'을 사용하십시오.)

테스트 목록 수행할 수 있는 테스트. 공백을 사용하여 테스트 필드를

구분하십시오.

자세한 내용은 비고를 확인하십시오.

--^nettotalmb "^serverDownloadTest"나 "^serverUploadTest".와 같은

네트워크 테스트에서 전송된 사용자 데이터의

양을 메가바이트로

나타냅니다.

값은 "4"에서 "512" 사이여야 합니다.

(기본값: 16)

--^networkiterations 실행될 "^serverDownloadTest",

"^serverUploadTest"의 반복 횟수를

나타냅니다.

값은 "1"에서 "100" 사이여야 합니다.

(기본값: 1)

--^diskdatasize "^diskTest"에서 쓰고 읽을 데이터의 양을

메가바이트 단위로

나타냅니다.

값은 "100"에서 "4096" 사이여야 합니다.

(기본값: 512)

--^disktestpath "^diskTest"가 테스트 파일을 쓸

경로. 매개 변수가 제공되지 않은 경우,

명령은 시스템 임시 경로를 사용하려고

시도합니다.

--^systemdisplaytime 시스템 리소스 사용량을 보여주는

시간 간격(초). 이 옵션은

다음 테스트에 사용할 수 있습니다. "^systemNetworkUsage"

및 "^systemDiskUsage".

값은 "1"에서 "3600" 사이여야 합니다.

(기본값: 5초).

--^systemdisplaytimeinterval 시스템 성능 샘플 간의

시간 간격(초). 이 옵션은

다음 테스트에 사용할 수 있습니다.

"^systemNetworkUsage" 및 "^systemDiskUsage".

값은 "1"에서 "60" 사이여야 합니다.

(기본값: 1초).

== CMD\_HELP\_IOSTATS ==

비고:

이 명령을 사용하려면 네트워크 속도 테스트("^serverUploadTest" 및/또는 "^serverDownloadTest")

진행 중에 사용할 수 있는 서버가 있어야 합니다.

'--^diskTestPath'는 테스트할 물리 디스크 드라이브에 속한

경로를 가리켜야 합니다. 경로가 지정되지 않은 경우, 명령은

시스템 기본 임시 경로를 사용하려고 시도합니다.

지정된 경로의 디스크 드라이브에 테스트를 실행할 만큼 충분한 여유 공간이

있어야 합니다.

명령 실행 중에 수행되는 테스트로 인해 시스템 성능이

저하될 수 있습니다.

사용 가능한 테스트:

--^serveruploadtest (기본값) Plastic SCM 클라이언트에서 서버로 향하는

데이터 업로드 속도를 측정합니다.

--^serverdownloadtest (기본값) Plastic SCM 서버에서 클라이언트로 향하는

데이터 다운로드 속도를 측정합니다.

--^disktest (기본값) 디스크 읽기 속도와 디스크 쓰기 속도를

측정합니다.

--^systemnetworkusage 시스템 네트워크 리소스의 현재 사용량을

표시합니다.

(Microsoft Windows에서 제공하는 네트워크

인터페이스 성능 카운터를 표시합니다).

Microsoft Windows에서만 사용할 수 있습니다.

--^systemdiskusage 시스템 물리 디스크의 현재 사용량을

표시합니다.

(Microsoft Windows에서 제공하는 네트워크

인터페이스 성능 카운터를 표시합니다).

Microsoft Windows에서만 사용할 수 있습니다.

예:

cm ^iostats MYSERVER:8087 --^serveruploadtest --^serverdownloadtest --^nettotalmb=32

== CMD\_DESCRIPTION\_ISSUETRACKER ==

지정된 이슈 추적기에서 이슈 상태를 가져오거나, 업데이트하거나, 찾습니다.

== CMD\_USAGE\_ISSUETRACKER ==

사용법:

cm ^issuetracker <이름> ^status ^get <작업 ID> <매개 변수>[ ...]

cm ^issuetracker <이름> ^status ^update <작업 ID> <상태> <매개 변수>[ ...]

cm ^issuetracker <이름> ^status ^find <상태> <매개 변수>[ ...]

cm ^issuetracker <이름> ^connection ^check <매개 변수>[ ...]

이름 연결할 이슈 추적기 이름.

현재 Jira만 지원됩니다.

작업 ID 쿼리하거나 업데이트할 이슈 번호.

^status 이슈 추적기의 이슈에 대한 유효한 상태.

Jira 매개 변수(모두 필수):

--^user=<사용자> 인증할 사용자.

--^password=<비밀번호> 인증할 비밀번호.

--^host=<URL> 이슈 추적기의 대상 URL.

--^projectkey=<키> Jira 프로젝트의 프로젝트 키.

== CMD\_HELP\_ISSUETRACKER ==

예:

cm ^issuetracker jira ^status ^get 11 --^user=user@mail.es --^password=pwd \

--^host=https://user.atlassian.net --^projectkey=PRJ

(프로젝트 'PRJ'에 대한 이슈 11의 상태를 가져옵니다.)

cm ^issuetracker jira ^status ^update 11 "완료" --^user=user@mail.es \

--^password=pwd --^host=https://user.atlassian.net --^projectkey=PRJ

(프로젝트 'PRJ'에 대한 이슈 11의 상태를 '완료'로 업데이트합니다.)

cm ^issuetracker jira ^status ^find "완료" --^user=user@mail.es --^password=pwd \

--^host=https://user.atlassian.net --^projectkey=PRJ

(프로젝트 'PRJ'에 대한 상태가 '완료'로 설정된 작업 ID를 가져옵니다.)

cm ^issuetracker jira ^connection ^check --^user=user@mail.es --^password=pwd \

--^host=https://user.atlassian.net --^projectkey=PRJ

(구성 매개 변수가 유효한지 확인합니다.)

== CMD\_DESCRIPTION\_LICENSEINFO ==

라이센스 정보와 라이센스 사용 상태를 표시합니다.

== CMD\_USAGE\_LICENSEINFO ==

사용법:

cm ^licenseinfo | ^li [--^server=<리포지토리 서버 사양>] [--^inactive] [--^active]

[--^sort=(^name|^status)]

옵션:

--^server 지정된 서버에서 라이센스 정보를 가져옵니다.

서버가 지정되지 않은 경우, 클라이언트에 구성된 서버를

대상으로 명령을 실행합니다.

(리포지토리 서버 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을

사용하십시오.)

--^inactive "라이센스 사용" 섹션에 비활성 사용자만 표시합니다.

--^active "라이센스 사용" 섹션에 활성 사용자만 표시합니다.

--^sort 지정된 정렬 옵션으로 사용자를 정렬합니다.

'^name' 또는 '^status'를 사용할 수 있습니다.

== CMD\_HELP\_LICENSEINFO ==

비고:

표시되는 정보는 만료일, 활성화된 사용자, 비활성화된 사용자 등으로

구성됩니다.

예:

cm ^licenseinfo

cm ^licenseinfo --^server=myserver:8084

cm ^licenseinfo --^sort=^name

== CMD\_DESCRIPTION\_LINKTASK ==

체인지 세트에 작업을 연결합니다.

== CMD\_USAGE\_LINKTASK ==

이는 'cm'만 자동화할 때 사용하는 자동화 명령입니다.

주의해야 하는 만큼 사용자 친화적이지 않은 명령입니다.

사용법:

cm ^linktask | ^lt <체인지 세트 사양> <확장자 접두어> <작업 이름>

체인지 세트 사양 작업에 연결할 전체 체인지 세트 사양.

(체인지 세트 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을

사용하십시오.)

확장자 접두어 사용할 구성된 이슈 추적 시스템의 확장자

접두어.

작업 이름 이슈 추적 시스템의 작업 식별자.

== CMD\_HELP\_LINKTASK ==

예:

cm ^lt ^cs:8@^rep:default@^repserver:localhost:8084 jira PRJ-1

== CMD\_DESCRIPTION\_LOCK\_LIST ==

서버에 잠금을 표시합니다.

== CMD\_USAGE\_LOCK\_LIST ==

사용법:

cm ^lock ^list | ^ls [<개정 사양> [ ...]] [--^server=<서버>]

[--^onlycurrentuser] [--^onlycurrentworkspace]

[--^ignorecase]

개정 사양 하나 이상의 잠금이 있고 개정과 관련된 항목이

서버에서 잠겨 있는 경우, 이 명령은 지정된 개정별로

하나의 잠금 행을 표시합니다. 그렇지 않은 경우,

이 명령은 기본 서버(또는 '--^server' 옵션으로 설정된 서버)의

모든 잠긴 항목을 나열합니다.

개정 사양을 두 개 이상 사용하는 경우

공백을 사용해 구분하십시오.

(개정 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을 사용하십시오.)

옵션:

--^server 리포지토리 서버 사양.

이 옵션은 현재 워크스페이스 또는

client.conf 파일에서 가져온 기본 서버를

재정의합니다.

(서버 사양에 관한 자세한 내용을 확인하려면

'cm ^help ^objectspec'을 사용하십시오.)

--^onlycurrentuser 현재 사용자가 수행한 잠금만 표시하도록

결과를 필터링합니다.

--^onlycurrentworkspace 현재 워크스페이스에서 수행된 잠금만 표시하도록

결과를 필터링합니다(이름으로 일치).

--^ignorecase 서버 경로 사양이 사용되는 경우 경로의 대소문자를

무시합니다. 이 플래그를 사용하면, 사용자가 경로를

"/sRc/fOO.c"라 써도 명령이 "/src/foo.c"에서 실행됩니다.

== CMD\_HELP\_LOCK\_LIST ==

비고:

이 명령은 기본 서버에서 현재 잠긴 항목의 목록을

표시합니다. 개정 사양 목록도 허용합니다. 이 경우,

선택된 항목에 속한 잠금만 표시됩니다.

'--^server=<서버>'를 사용해 쿼리할 기본 서버를 설정할 수 있습니다.

이 명령은 지정된 서버의 모든 잠금에 대한 행을 표시합니다.

- 잠겨 있는 항목의 GUID.

- 잠금을 수행한 사용자 이름.

- 잠금이 수행된 워크스페이스 이름.

- 잠겨 있는 항목의 경로(서버 경로 형식).

예:

cm ^lock ^list

cm ^lock ^list --^server=myserver:8084

cm ^lock ^ls ^serverpath:/src/foo.c#^cs:99@default@localhost:8084

cm ^lock ^list ^revid:3521@default ^itemid:2381@secondary --^onlycurrentuser

cm ^lock ^ls --^onlycurrentuser

cm ^lock ^ls --^onlycurrentuser --^onlycurrentworkspace

== CMD\_DESCRIPTION\_LISTUSERS ==

사용자와 그룹을 나열합니다.

== CMD\_USAGE\_LISTUSERS ==

사용법:

cm ^listusers | ^lu <개정 서버 사양> [--^onlyusers] [--^onlygroups]

[--^filter= <문자열 필터>]

리포지토리 서버 사양 리포지토리 서버의 사양.

(사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을 사용하십시오.)

옵션:

--^onlyusers 사용자만 나열합니다.

--^onlygroups 그룹만 나열합니다.

--^filter 지정된 필터와 일치하는 사용자 또는 그룹만

나열합니다.

== CMD\_HELP\_LISTUSERS ==

예:

cm ^lu localhost:8084

(서버의 모든 사용자를 나열합니다.)

cm ^listusers localhost:8084 --^onlyusers --^filter=m

("m"이 포함된 서버의 사용자만 나열합니다.)

== CMD\_DESCRIPTION\_LOCATION ==

'cm'의 경로를 반환합니다.

== CMD\_USAGE\_LOCATION ==

사용법:

cm ^location

== CMD\_HELP\_LOCATION ==

== CMD\_DESCRIPTION\_LOCK ==

이 명령을 사용하면 사용자가 잠금을 관리할 수 있습니다.

== CMD\_USAGE\_LOCK ==

사용법:

cm ^lock <명령> [옵션]

명령:

^list | ^ls

^unlock

각 명령에 대한 자세한 정보는 다음을 실행해 확인하십시오.

cm ^lock <명령> --^usage

cm ^lock <명령> --^help

== CMD\_HELP\_LOCK ==

예:

cm ^lock ^list

cm ^lock

(인수가 없는 경우 '^list'는 선택사항입니다.)

cm ^lock ^ls ^serverpath:/src/foo.c#^cs:99@default@localhost:8084

cm ^lock ^unlock 91961b14-3dfe-4062-8c4c-f33a81d201f5

== CMD\_DESCRIPTION\_LOG ==

체인지 세트의 개정에 관한 정보를 가져옵니다.

== CMD\_USAGE\_LOG ==

사용법:

cm ^log [<체인지 세트 사양> | <리포지토리 사양>] [--^from=<가져올 체인지 세트 사양>] [--^allbranches]

[--^ancestors] [--^csformat=<문자열 형식>] [--^itemformat=<문자열 형식>]

[--^xml[=<출력 파일>]] [--^encoding=<이름>]

[--^repositorypaths | --^fullpaths | --^fp]

옵션:

체인지 세트 사양 사양이 제공된 체인지 세트의 모든 변경사항을

명령이 반환합니다.

(체인지 세트 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을

사용하십시오.)

리포지토리 사양 명령이 지정된 리포지토리의 모든 변경사항을

나열합니다.

(리포지토리 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을

사용하십시오.)

--^from 체인지 세트 사양 [가져올 체인지 세트 사양]부터

체인지 세트 사양 [체인지 세트 사양]까지의 모든 체인지 세트에서 변경된

모든 사항을 나열합니다.

[가져올 체인지 세트 사양] 체인지 세트는 출력에 포함되지

않습니다.

리포지토리 사양이 제공되면 무시됩니다.

--^allbranches 체인지 세트가 생성된 모든 브랜치에 대해

지정된 간격으로 생성된 체인지 세트에 관한 정보를

표시합니다.

--^ancestors 주어진 체인지 세트([체인지 세트 사양])에 대한 부모 및

병합 링크에서 연결 가능한 체인지 세트에 관한 정보를

표시합니다. 가져올 체인지 세트([가져올 체인지 세트 사양])도

제공된 경우, 모든 경로에 대해 낮은 한도로

사용됩니다. 비고: 이 옵션을 사용하면

체인지 세트 변경사항이 표시되지 않습니다.

--^csformat 특정 형식으로 체인지 세트 정보를 가져옵니다. 자세한 내용은

비고를 확인하십시오.

--^itemformat 특정 형식으로 항목 정보를 가져옵니다. 자세한 내용은

비고를 확인하십시오.

--^xml XML 형식의 출력을 표준 출력으로 출력합니다.

출력 파일을 지정할 수 있습니다.

--^encoding '--^xml' 옵션과 함께 사용되며 XML 출력에 사용할

인코딩, 즉 UTF-8을 지정합니다.

지원되는 인코딩 및 형식에 대한 표를 확인하려면

http://msdn.microsoft.com/en-us/library/system.text.encoding.aspx에서

MSDN 문서를 확인하십시오.

페이지 마지막 "Name"열에 있습니다.

--^fullpaths, --^fp 가능한 경우 강제로 파일 및 디렉터리에 대한 전체

워크스페이스 경로를 출력합니다.

--^repositorypaths 워크스페이스 경로 대신 리포지토리 경로(서버 경로)를

출력합니다. (이 옵션은 '--^fullpaths' 옵션을

재정의합니다).

== CMD\_HELP\_LOG ==

비고:

- '체인지 세트 사양'이나 옵션이 지정되지 않은 경우, 명령은

모든 브랜치에서 지난달에 생성된 모든 체인지 세트에 관한 정보를

표시합니다.

- '--^from' 옵션만 포함된 경우, 명령은 체인지 세트가 생성된 브랜치에서

지정된 체인지 세트부터 마지막 체인지 세트까지

모든 체인지 세트에 관한 정보를 표시합니다.

- '--^allbranches' 옵션이 간격 없이 표시되면, 명령은

'체인지 세트 사양'만 지정된 경우와 같은 정보를

가져옵니다.

- '--^from'을 사용한 경우, 출력에는 '가져올 체인지 세트 사양'+1의

정보가 포함됩니다.

- 체인지 세트 정보를 표시하는 데 사용되는 리포지토리는 명령이 실행되는

경로에 로드된 리포지토리입니다.

이 명령은 항목에 대한 형식 문자열('--^itemformat') 및

체인지 세트에 대한 형식 문자열('--^csformat')을 허용합니다.

'--^csformat'의 출력 매개 변수는 다음과 같습니다.

{^tab} 탭 공백을 삽입합니다.

{^newline} 새 행을 삽입합니다.

{^changesetid} 체인지 세트 번호.

{^branch} 체인지 세트가 생성된 브랜치.

{^date} 체인지 세트 날짜.

{^owner} 체인지 세트 소유자.

{^comment} 체인지 세트 코멘트.

{^items} 체인지 세트에 포함된 항목.

{^repository} 체인지 세트가 있는 리포지토리.

{^repserver} 서버 이름.

'--^itemformat'의 출력 매개 변수는 다음과 같습니다.

{^tab} 탭 공백을 삽입합니다.

{^newline} 새 행을 삽입합니다.

{^path} 항목 경로.

{^branch} 체인지 세트가 생성된 브랜치.

{^date} 체인지 세트 날짜.

{^owner} 체인지 세트 소유자.

{^shortstatus} 짧은 형식을 출력합니다. 아래를 참고하십시오.

{^fullstatus} 긴 형식을 출력합니다. 아래를 참고하십시오.

짧은 형식과 그에 해당하는 긴 형식:

'^A' ^Added

'^D' ^Deleted

'^M' ^Moved

'^C' ^Changed

다음은 유효한 출력 문자열입니다.

--^csformat="{^newline}체인지 세트 {^changesetid}이(가) {^date}에 생성됨,{^tab} 변경된 항목: {^items}개."

--^itemformat="{^newline}{^path} 항목이 {^branch} 브랜치에서 변경되었습니다."

예:

cm ^log

(모든 브랜치에서 지난달에 생성된 모든 체인지 세트에 관한 정보를

표시합니다.)

cm ^log ^cs:16

(체인지 세트가 생성된 브랜치의 체인지 세트 16에서 변경된 사항에 관한

정보를 표시합니다.)

cm ^log ^cs:16 --^csformat="{^newline}체인지 세트 {^changesetid}이(가) \

{^date}에 생성됨,{^tab} 변경된 항목: {^items}개."

(지정된 형식으로 정보를 표시합니다.)

cm ^log --^from=^cs:20 ^cs:50

(체인지 세트 21부터 체인지 세트 50까지의 모든 체인지 세트에 포함된

모든 개정에 관한 정보를 표시합니다.)

cm ^log --^from=^cs:20 ^cs:50 --^allbranches

(리포지토리의 모든 브랜치에 있는 체인지 세트 21부터 체인지 세트 50까지의

모든 체인지 세트에 포함된 모든 개정에 관한 정보를

표시합니다.)

cm ^log ^rep:myrep@localhost:8084

(지정된 리포지토리에서 변경된 사항에 관한 정보를 표시합니다.

워크스페이스를 지정하지 않아도 명령을 실행할 수 있습니다.)

cm ^log --^from=^cs:20@^rep:mainRep@localhost:8084

(체인지 세트 21부터 모든 체인지 세트에 포함된 모든 개정에 관한 정보를

표시합니다. 전체 체인지 세트 사양이 지정되었으므로, 워크스페이스를

지정하지 않아도 명령을 실행할 수 있습니다.)

== CMD\_DESCRIPTION\_LS ==

트리의 내용을 나열합니다.

== CMD\_USAGE\_LS ==

사용법:

cm ^ls | ^dir [<경로>[ ...]] [--^format=<문자열 형식>] [--^symlink]

[--^selector[=<선택기 형식>]] [--^tree=<객체 사양>]

[-^R | -^r | --^recursive]

[--^xml[=<출력 파일>]] [--^encoding=<이름>]

옵션:

경로 표시할 경로 목록. 공백을 사용하여 경로를

구분하십시오.

공백이 포함된 경로를 지정하려면 큰따옴표(" ")를

사용하십시오.

--^format 특정 형식으로 출력 메시지를 가져옵니다. 자세한 내용은

비고를 확인하십시오.

--^symlink 대상이 아닌 기호 링크에 작업을

적용합니다.

--^selector 활성 워크스페이스 선택기에서 콘텐츠를 가져옵니다.

선택기 형식이 지정된 경우 지정된 선택기를

나열합니다.

선택기는 4.x 버전부터 Plastic SCM의

핵심 부분이 아니기 때문에 거의 사용되지 않습니다.

--^tree 지정된 체인지 세트나 브랜치의 트리를 나열합니다.

(사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을 사용하십시오.)

-^R 재귀적으로 나열합니다.

--^xml XML 형식의 출력을 표준 출력으로 출력합니다.

출력 파일을 지정할 수 있습니다.

--^encoding '--^xml' 옵션과 함께 사용되며 XML 출력에 사용할

인코딩, 즉 UTF-8을 지정합니다.

지원되는 인코딩 및 형식에 대한 표를 확인하려면

http://msdn.microsoft.com/en-us/library/system.text.encoding.aspx에서

MSDN 문서를 확인하십시오.

페이지 마지막 "Name"열에 있습니다.

== CMD\_HELP\_LS ==

비고:

- 경로를 메타 문자로 입력할 수 있습니다.

- 목록은 워크스페이스 선택기에 따라 다릅니다.

- 명령의 출력을 형식 문자열을 지정하여 지정된 형식으로 출력할 수 있습니다.

- '--^tree' 또는 '--^selector' 옵션이 지정된 경우, 주어진

경로는 워크스페이스 경로(C:\Users\myuser\mywk\dir\file.txt)가 아닌

/dir/file.txt와 같은 서버 경로('cm path')여야 합니다.

- 경로가 제공되지 않으면 현재 디렉터리를 워크스페이스 경로로

간주합니다. '--^tree' 또는 '--^selector' 옵션을 사용하는 경우,

루트 경로("/")로 간주합니다.

기본 형식 문자열은 다음과 같습니다.

"{^size,10} {^date:dd/MM/yyyy} {^date:HH:mm}\

{^type,-6} {^location,-12} {^checkout,-5} {^name}\

{^symlinktarget}"

이 명령은 출력을 표시할 형식 문자열을 허용합니다.

이 명령의 출력 매개 변수는 다음과 같습니다.

{^size}

{^formattedsize}

{^date}

{^type}

^dir 디렉터리,

^txt 텍스트 파일,

^File 파일.

{^location} 예: ^br:branch#cset

{^checkout}

{^name}

{^changeset}

{^path}

{^repspec}

{^owner}

{^revid}

{^parentrevid}

{^itemid}

{^brid}

{^repid}

{^server}

{^symlinktarget}

{^hash}

{^chmod}

{^wkpath} 워크스페이스 루트 관련 경로

{^branch}

{^newlocation} cset@branch

{^guid} (해결에 더 긴 시간이 걸릴 수 있습니다)

{^itemguid}

{^transformed} 변환된 항목에 적용된 규칙 표시

PLASTIC\_LS\_FORMAT 환경 변수를 설정하여 '^ls' 형식을 커스터마이즈할 수

있습니다.

예:

cm ^ls

cm ^ls c:\workspace\src

cm ^ls --^format={^name}

(파일 이름만 표시합니다.)

cm ^ls --^symlink

('기호 링크 처리된' 파일이나 디렉터리 대신 기호 링크에 관한 정보를

표시합니다. UNIX 환경에서 사용할 수 있습니다.)

cm ^ls code --^selector

(현재 워크스페이스 선택기에서 'code' 하위 디렉터리의 내용을

표시합니다.)

cm ^ls /code --^selector="^rep 'myrep' ^path '/' ^branch '/^main'"

(지정된 선택기에 있는 '/code' 하위 디렉터리의 내용을 표시합니다.

경로는 서버 형식으로 지정해야 합니다.)

cm ^ls /code --^tree=44@myrep@denver:7070

(서버 'denver:7070'의 리포지토리 'myrep'에 있는 체인지 세트 44의

'/code' 하위 디렉터리를 나열합니다.)

cm ^ls /code --^tree=^br:/main/scm13596@myrep@denver:7070

(서버 'denver:7070'의 리포지토리 'myrep'에 있는 브랜치 '/main/scm13596'의

마지막 체인지 세트에 있는 '/code' 하위 디렉터리를 나열합니다.)

cm ^ls /code --^tree=ae1390ed-7ce9-4ec3-a155-e5a61de0dc77@myrep@denver:7070

(서버 'denver:7070'의 리포지토리 'myrep'에 있는

체인지 세트 ae1390ed-7ce9-4ec3-a155-e5a61de0dc77의 '/code' 하위 디렉터리를

나열합니다.)

== CMD\_DESCRIPTION\_TRIGGER\_LIST ==

서버에서 지정된 유형의 트리거를 나열합니다.

== CMD\_USAGE\_TRIGGER\_LIST ==

사용법:

cm ^trigger | ^tr ^list | ^ls [<하위 유형 유형>] [--^server=<리포지토리 서버 사양>]

[--^format=<문자열 형식>]

옵션:

하위 유형 유형 트리거 실행 및 트리거 작업.

트리거 유형을 확인하려면 'cm ^showtriggertypes'를

입력하십시오.

--^server 지정된 서버에서 트리거를 나열합니다.

서버가 지정되지 않은 경우, 클라이언트에 구성된 서버를

대상으로 명령을 실행합니다.

(서버 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을

사용하십시오.)

--^format 특정 형식으로 출력 메시지를 가져옵니다. 자세한 내용은

비고를 확인하십시오.

== CMD\_HELP\_TRIGGER\_LIST ==

비고:

유형이 지정되지 않은 경우, 서버에 있는 모든 트리거를 나열합니다.

이 명령은 출력을 표시할 형식 문자열을 허용합니다.

이 명령의 출력 매개 변수는 다음과 같습니다.

{0} 트리거 위치.

{1} 트리거 이름.

{2} 트리거 경로.

{3} 트리거 소유자.

{4} 트리거 유형.

{5} 트리거 필터.

예:

cm ^trigger mk 레이블 뒤에 나열

cm ^tr ^ls ^before-mkbranch --^server=myserver:8084

== CMD\_DESCRIPTION\_MANIPULATESELECTOR ==

선택기를 날짜로 변경합니다.

== CMD\_USAGE\_MANIPULATESELECTOR ==

이는 'cm'만 자동화할 때 사용하는 자동화 명령입니다.

주의해야 하는 만큼 사용자 친화적이지 않은 명령입니다.

사용법:

cm ^manipulateselector | ^ms [<워크스페이스 경로> | <워크스페이스 사양>] --^atdate=<선택기 날짜>

워크스페이스 경로 워크스페이스의 경로입니다.

워크스페이스 사양 워크스페이스 사양입니다. (사양에 관한 자세한 내용을 확인하려면

'cm ^help ^objectspec'을 사용하십시오.)

옵션:

--^atdate 지정된 날짜에 워크스페이스를 다시 생성할 선택기를

반환합니다.

== CMD\_HELP\_MANIPULATESELECTOR ==

비고:

경로나 워크스페이스 사양이 모두 지정되지 않은 경우, 명령은 현재 디렉터리를

워크스페이스 경로로 사용합니다.

예:

cm ^manipulateselector c:\workspace --^atdate=yyyy-MM-ddTHH:mm:ss

cm ^manipulateselector --^atdate=yyyy-MM-ddTHH:mm:ss

cm ^manipulateselector > mySelector.txt --^atdate=yyyy-MM-ddTHH:mm:ss

cm ^manipulateselector ^wk:build\_wk@BUILDER --^atdate=yyyy-MM-ddTHH:mm:ss

== CMD\_DESCRIPTION\_MERGE ==

브랜치를 다른 브랜치와 병합합니다.

== CMD\_USAGE\_MERGE ==

사용법:

cm ^merge <소스 사양> [--^merge] [--^cherrypicking] [--^forced]

[--^mergetype=(^onlyone|^onlysrc|^onlydst|^try|^forced)]

[--^interval-origin=<체인지 세트 사양> | --^ancestor=<체인지 세트 사양>]

[--^keepsource | --^ks] [--^keepdestination | --^kd]

[--^automaticresolution=<충돌 유형>[;...]]

[--^subtractive] [--^mount] [--^printcontributors]

[--^noprintoperations] [--^silent]

[(--^to=<브랜치 사양> | --^destination=<브랜치 사양>)[--^shelve]]

[--^no-dst-changes]

[-^c=<문자열 코멘트> | --^commentsfile=<코멘트 파일>]

[--^resolveconflict --^conflict=<색인>

--^resolutionoption=(^src|^dst|(^rename --^resolutioninfo=<문자열 이름>))

--^mergeresultfile=<경로> --^solvedconflictsfile=<경로>]

[--^nointeractiveresolution]

[--^machinereadable [--^startlineseparator=<구분 기준>]

[--^endlineseparator=<구분 기준>] [--^fieldseparator=<구분 기준>]]

소스 사양 병합할 소스 객체의 사양.

- 브랜치 사양: '[^br:/]br\_name'

- 레이블 사양: '^lb:lb\_name'

- 체인지 세트 사양: '^cs:cs\_number'

- 임시 저장 사양: '^sh:shelve\_number'

(사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을 사용하십시오.)

옵션:

--^merge 병합을 수행합니다. 수행할 수 없는 경우,

발견된 충돌을 출력합니다.

--^cherrypicking 소스 체인지 세트에 포함된 변경사항을

병합합니다. 병합 소스 사양이 레이블인 경우에는

이 옵션을 사용하지 않습니다.

--^forced 소스와 대상이 이미 연결되어 있는 지 여부를

확인하지 않습니다.

이 옵션은 간격 병합과 체리픽에만

사용할 수 있습니다.

--^mergetype 자세한 내용은 비고를 확인하십시오.

--^interval-origin 병합이 소스 체인지 세트와 지정된 간격 원본 간의

차이점만 가져올 수 있도록 어떤 체인지 세트가

간격 원본으로 선택되었는지

지정합니다.

--^ancestor --^interval-origin의 별칭입니다.

--^keepsource 충돌이 있는 항목에 대해 소스 기여자의 모든

변경사항을 허용합니다.

--^keepdestination 충돌이 있는 항목에 대해 대상 기여자의 변경사항을

보존합니다.

--^automaticresolution 디렉터리 충돌을 해결하는 데 사용됩니다. 이 옵션을

사용하면 충돌을 해결하기 위해 소스 기여자나

대상 기여자 중 누구를 자동으로 선택할지

선택할 수 있습니다.

세미콜론(;)을 사용하여 충돌 유형을 구분하십시오.

자세한 내용은 비고를 확인하십시오.

--^subtractive 병합으로 인한 변경사항을 삭제합니다. 명령에

전달된 이 매개 변수(소스 사양)는

변경사항을 삭제할 소스를 지정하는 데

사용됩니다. 이는 반드시 체인지 세트여야 합니다. 체인지 세트 간격의

경우, '--^interval-origin'을 사용해 간격 원본을

정의해야 합니다. 변경사항을

제거하기 위해 시스템은 삭제된 변경사항을 제외한

이전 내용을 포함하는 새로운 체크아웃된

개정을 생성합니다.

--^mount 지정된 리포지토리의 마운트 지점입니다.

--^printcontributors 기여자(베이스, 소스, 대상)를

출력합니다.

--^noprintoperations 해결 정보를 표시하지 않고 자동으로

병합을 해결합니다.

--^silent 어떤 내용도 출력하지 않습니다.

--^to | --^destination 전체 충돌 해결을 통해 지정된

브랜치(브랜치 사양 입력)를 대상으로 병합 작업을

수행합니다.

"대상에 병합"(또는 워크스페이스 없는 병합)은

서버 측에서 수행되는 병합입니다. 일반적인 병합은

브랜치, 레이블, 체인지 세트"에서" 병합되는 워크스페이스에서

발생하는 반면, 대상에 병합 작업은 온전히

서버에서 발생합니다. 일반 병합에서는

"대상"이 워크스페이스이지만, "대상에 병합"에서

대상이 항상 지정되어야 합니다(그런 이유로

"대상에" 병합이라 합니다).

"대상에 병합" 기능에 관한 자세한 내용은

다음 링크를 참조하십시오.

https://www.plasticscm.com/download/help/mergeto

--^shelve 새 체인지 세트를 생성하는 대신 병합 결과의

변경사항과 병합 추적 가능성 정보로

임시 저장을 생성합니다. 병합 소스가 임시 저장인 경우에는

이 옵션을 사용할 수 없습니다. 이 옵션은

server-side-merge("대상에 병합")에만

사용할 수 있습니다. 따라서, '--^to' 옵션 및

'--^merge' 옵션을 사용해야 합니다.

--^no-dst-changes 대상 기여자에 변경사항이 없는지

확인합니다(대상 체인지 세트 또한 공통

조상입니다). 대상에 변경사항이 있는 경우에는

병합을 할 수 없습니다.

-^c 병합 작업에서 생성된 체인지 세트에 지정된 코멘트를

적용합니다.

--^commentsfile 병합 작업에서 생성된 체인지 세트에

지정된 파일의 코멘트를 적용합니다.

--^resolveconflict (주로 플러그인에서 사용됩니다. 자세한 내용은 비고를 확인하십시오.)

디렉터리 충돌을 해결하는 데 사용됩니다.

--^conflict '--^resolveconflict' 플래그와 함께 사용되며,

1부터 시작해 해결할 충돌 색인을 지정합니다.

--^resolutionoption '--^resolveconflict' 플래그와 함께 사용되며,

충돌 해결 유형을 나타냅니다. '^src', '^dst', '^rename'

중 하나의 옵션을 사용하십시오.

자세한 내용은 비고를 확인하십시오.

--^resolutioninfo '--^resolveconflict' 플래그와 함께 사용되며,

'--^resolutionoption' 옵션이 'rename'인 경우 사용할

이름을 제공합니다.

--^mergeresultfile '--^resolveconflict' 플래그와 함께 사용되며,

서로 다른 호출 간의 병합 결과 정보를 파일로

출력합니다. 지정된 경로는 첫 번째 호출 중에

생성되며 다음 호출마다 경로가

업데이트됩니다.

--^solvedconflictsfile '--^resolveconflict' 플래그와 함께 사용되며,

서로 다른 호출 간에 해결된 충돌 정보를

파일로 출력합니다. 지정된 경로는

첫 번째 호출 중에 생성되며 다음 호출마다

업데이트됩니다.

--^nointeractiveresolution (주로 플러그인에서 사용됩니다. 자세한 내용은 비고를 확인하십시오.)

사용자에게 수동 충돌 메시지를 표시하지 않습니다.

이렇게 하면 디렉터리 충돌이 해결되지 않습니다.

--^machinereadable (주로 플러그인에서 사용됩니다. 자세한 내용은 비고를 확인하십시오.)

결과를 구문 분석이 쉬운 형식으로 출력합니다.

--^startlineseparator '--^machinereadable' 플래그와 함께 사용되며,

행이 시작되는 방법을 지정합니다. (기본값: 빈 문자열.)

--^endlineseparator '--^machinereadable' 플래그와 함께 사용되며,

행이 끝나는 방법을 지정합니다. (기본값: 빈 문자열.)

--^fieldseparator '--^machinereadable' 플래그와 함께 사용되며,

필드를 구분할 방법을 지정합니다. (기본값:

공백.)

== CMD\_HELP\_MERGE ==

비고:

이 명령은 두 브랜치 또는 레이블과 브랜치 간의 변경사항을 병합하는 데

사용됩니다. 병합 대상은 항상 브랜치여야 합니다.

병합 소스는 인수로 지정됩니다.

대상은 워크스페이스의 현재 콘텐츠입니다.

예를 들어, 브랜지 task001에서 주 브랜치로 병합될 요소를 표시하려면

선택기가 주 브랜치를 가리켜야 하며, 워크스페이스를 업데이트해야 합니다.

이후 다음을 수행합니다.

cm ^merge ^br:/task001

실제로 병합을 수행하려면 '--^merge' 옵션을 추가합니다.

cm ^merge ^br:/task001 --^merge

병합 소스를 정의하려면 다음 사양을 사용할 수 있습니다.

- 브랜치 사양:

[^br:/]br\_name

예: ^br:/main/task001

(이 경우, 이 브랜치의 마지막 체인지 세트에서 병합합니다.)

- 레이블 사양:

^lb:lb\_name

예: ^lb:BL001

(레이블이 적용된 변경세트에서 병합합니다.)

- 체인지 세트 사양:

^cs:cs\_number

예: ^cs:25

(지정된 변경세트 내용에서 병합합니다.)

- 임시 저장 사양:

^sh:shelve\_number

예: ^sh:2

(지정된 임시 저장 내용에서 병합합니다.)

자동으로 디렉터리 충돌을 해결하려면 '--^automaticresolution' 옵션을

사용하고 충돌의 유형과 병합 작업 중에 선택해야 하는

기여자(소스 또는 대상)를 차례로 지정하십시오.

(각 "충돌 유형"-"기여자" 쌍은 세미콜론(;)으로 구분하십시오.)

예:

cm ^merge ^cs:2634 --^merge --^automaticresolution=^eviltwin-src;^changedelete-src

(체인지 세트 2634의 병합 작업은 "^eviltwin" 및

"^changedelete" 충돌을 모두 소스("-^src") 기여자를 유지하여

해결합니다.)

- 충돌 유형 뒤의 "-^src" 접미어는 소스 기여자 변경사항을 유지하도록

병합 명령에 전달합니다.

- "-^dst" 접미어는 대상 기여자 변경사항을 유지합니다.

다음은 병합 명령이 지원하는 충돌 유형 목록입니다.

"^movedeviltwin", "^eviltwin", "^changedelete", "^deletechange", "^movedelete",

"^deletemove", "^loadedtwice", "^addmove", "^moveadd", "^divergentmove",

"^cyclemove", "^all".

"^all" 값은 다른 옵션을 재정의합니다. 다음 예에서는

"^eviltwin-dst"가 무시됩니다.

cm ^merge ^br:/main/task062 --^merge --^automaticresolution=^all-src;^eviltwin-dst

병합 충돌에 관한 자세한 내용은 다음 링크를 참조하십시오.

https://www.plasticscm.com/download/help/directorymerges

다음은 '--^mergetype'에 사용할 수 있는 옵션입니다.

^onlyone 한 명의 기여자만 항목을 수정한 경우 자동으로

병합합니다.

^onlysrc 소스 기여자만 항목을 수정한 경우 자동으로

병합합니다.

^onlydst 대상 기여자만 항목을 수정한 경우

자동으로 병합합니다.

^try 충돌하는 코드 조각(충돌별)을 한 명의 기여자만 수정한

경우 자동으로 병합합니다.

^forced 항상 모든 비자동 충돌을 해결합니다.

다음은 주로 플러그인 및 통합에서 사용하는 옵션입니다.

- '--^resolveconflict'를 사용해 디렉터리 충돌을 해결합니다. 다음 옵션도

사용해야 합니다.

- '--^conflict'는 1부터 시작해 해결할 충돌

색인입니다.

- '--^resolutionoption'은 사용할 충돌 해결을

나타냅니다. 다음을 사용합니다.

- '^src': 소스 변경사항을 유지하고 대상 변경사항을

취소합니다

- '^dst': 대상 변경사항을 유지하고 소스 변경사항을

취소합니다

- '^rename': (충돌 유형이 이 해결을 지원하는

경우에만) 대상의 이름을 '--^resolutioninfo' 옵션에 제공된

이름으로 변경합니다.

- '--^resolutioninfo'를 사용해 '^rename' 해결에 사용할

이름을 지정합니다

- '--^mergeresultfile' 및 '--^solvedconflictsfile'은 모두 서로 다른

호출 간의 병합 정보를 저장하는 데 사용됩니다.

- '--^nointeractiveresolution'은 사용자에게 수동 충돌 해결을 요청하지 않는

병합을 나타냅니다.

- '--^machinereadable', '--^startlineseparator', '--^endlineseparator',

'--^fieldseparator' 옵션을 사용하면 출력을 기기가 읽을 수 있는 형식(구문 분석이 쉬운)으로

출력합니다.

예:

cm ^merge --^machinereadable --^startlineseparator=start@\_@line \

--^endlineseparator=new@\_@line --^fieldseparator=def#\_#sep \

--^mergeresultfile=C:\Users\Borja\AppData\Local\Temp\2tmp4D6C.tmp \

--^solvedconflictsfile=C:\Users\Borja\AppData\Local\Temp\2tmp4D6D.tmp \

--^resolveconflict --^conflict=1 --^resolutionoption=이름 변경 \

--^resolutioninfo=바이너리 대상 ^br:/main/task --^merge

예:

cm ^merge ^br:/task001

(병합하지 않고 병합할 항목만 출력합니다.)

cm ^merge ^br:/task001 --^merge

('task001' 브랜치에서 병합합니다.)

cm ^merge ^cs:5 --^merge --^cherrypicking --^interval-origin=^cs:2

(체인지 세트 간격(2,5]에서 체리픽합니다.)

cm ^merge ^cs:8 --^merge --^subtractive --^keepdestination

(체인지 세트 8에서 차감식 병합을 수행하여 충돌이 있는 요소의

대상 변경사항을 유지합니다.)

cm ^merge ^br:/main/task001 --^to=^br:/main --^merge -^c="새 UI 통합됨"

('task001' 브랜치에서 'main' 브랜치로 서버 측 병합 즉, 대상에 병합을 수행하고

코멘트를 설정합니다.)

cm ^merge ^br:/main/task001 --^to=^br:/main --^merge --^shelve

('task001' 브랜치에서 'main' 브랜치로 서버 측 병합을 수행하고

결과를 임시 저장에 남깁니다.)

cm ^merge ^sh:2 --^to=^br:/main --^merge --^no-dst-changes

(현재 'main' 헤드에서 생성된 경우에만 임시 저장 2를 'main'에

적용합니다)

== CMD\_DESCRIPTION\_ATTRIBUTE\_CREATE ==

새 속성을 생성합니다.

== CMD\_USAGE\_ATTRIBUTE\_CREATE ==

사용법:

cm ^attribute | ^att ^create | ^mk <속성 이름>

속성 이름 속성 이름

== CMD\_HELP\_ATTRIBUTE\_CREATE ==

예:

cm ^attribute ^create 상태

('상태' 속성을 생성합니다.)

cm ^att ^mk 통합

('통합' 속성을 생성합니다.)

== CMD\_DESCRIPTION\_BRANCH ==

사용자가 브랜치를 관리하도록 허용합니다.

== CMD\_USAGE\_BRANCH ==

사용법:

cm ^branch | ^br <명령> [옵션]

명령:

^create | ^mk

^delete | ^rm

^rename

^history

^showmain

^showmerges

각 명령에 대한 자세한 정보는 다음을 실행해 확인하십시오.

cm ^branch <명령> --^usage

cm ^branch <명령> --^help

== CMD\_HELP\_BRANCH ==

예:

cm ^branch /main/scm21345

cm ^branch ^create /main/scm21345

cm ^branch ^delete /main/scm21345

cm ^branch ^rename /main/scm21345 scm21346

cm ^branch ^history /main/scm21345

cm ^branch ^showmain

cm ^branch ^showmerges file.txt

== CMD\_DESCRIPTION\_BRANCH\_CREATE ==

새 브랜치를 생성합니다.

== CMD\_USAGE\_BRANCH\_CREATE ==

사용법:

cm ^branch | ^br [^create | ^mk] <브랜치 사양>

[--^changeset=<체인지 세트 사양> | --^label=<레이블 사양>]

[-^c=<문자열 코멘트> | -^commentsfile=<코멘트 파일>]

브랜치 사양 새 브랜치의 이름 또는 사양.

(브랜치 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을 사용하십시오.)

옵션:

--^changeset 새 브랜치의 시작 지점으로 사용되는 체인지 세트.

(체인지 세트 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을 사용하십시오.)

--^label 새 브랜치의 시작 지점으로 사용되는 레이블.

(레이블 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을 사용하십시오.)

-^c 새 브랜치의 코멘트 필드에 지정된 텍스트를

작성합니다.

-^commentsfile 새 브랜치의 코멘트 필드에 지정된 파일의 내용을

작성합니다.

== CMD\_HELP\_BRANCH\_CREATE ==

비고:

상위 수준 브랜치를 생성하려면 계층 없이 이름을 지정하십시오.

예:

cm ^br /dev

선택 매개 변수 '--^changeset'가 지정되지 않은 경우, 새 브랜치의 베이스는

부모 브랜치의 마지막 체인지 세트가 됩니다. 새 브랜치가

상위 수준 브랜치인 경우, 베이스 체인지 세트는 체인지 세트 0이 됩니다.

'-^c' 또는 '-^m' 전환을 사용하여 코멘트를 지정할 수 있습니다.

cm ^branch /main/task001 -^c="코멘트입니다"

cm ^branch /main/task001 -^m "코멘트입니다"

코멘트를 입력할 편집기를 지정하려면 PLASTICEDITOR 환경 변수를

설정하십시오. PLASTICEDITOR 환경 변수가 설정되었고

코멘트가 비어 있는 경우, 편집기가 자동으로 실행되어 사용자가

코멘트를 지정할 수 있습니다.

예:

cm ^branch task001

cm ^branch ^create task001

cm ^branch ^mk task001

cm ^br ^mk task001

(현재 워크스페이스의 리포지토리에 상위 수준 브랜치인 'task001'을

생성합니다.)

cm ^branch ^br:/task001/task002@

('task002' 브랜치를 'task001'의 자식 브랜치로 생성합니다.)

cm ^br /main/task001@myrep@myserver:8084 -^c="내 코멘트"

('task001' 브랜치를 리포지토리 'myrep@myserver:8084'에

'내 코멘트'라는 코멘트와 함께 'main'의 자식 브랜치로 생성합니다.)

cm ^branch ^br:/main/task001 --^changeset=2837 -^commentsfile=commenttask001.txt

('task001' 브랜치를 베이스 'changeset=2837'과 함께 'main'의 자식 브랜치로 생성하고

'commenttask001.txt' 파일의 코멘트를 적용합니다.)

== CMD\_DESCRIPTION\_BRANCH\_DELETE ==

하나 이상의 브랜치를 삭제합니다.

== CMD\_USAGE\_BRANCH\_DELETE ==

사용법:

cm ^branch | ^br ^delete | ^rm <브랜치 사양>[ ...]

브랜치 사양 삭제할 브랜치. 공백을 사용하여 브랜치를 구분하십시오.

(브랜치 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을

사용하십시오.)

== CMD\_HELP\_BRANCH\_DELETE ==

비고:

이 명령은 하나 이상의 브랜치를 삭제합니다.

예:

cm ^branch ^delete /main/task001

(현재 워크스페이스의 리포지토리에서 'main'의 자식 브랜치인 'task001'

브랜치를 삭제합니다.)

cm ^br ^rm main/task002 /main/task012@reptest@myserver:8084

(현재 워크스페이스의 리포지토리에서 '/main/task002' 브랜치를 삭제하고

리포지토리 'reptest@myserver:8084'에서 '/main/task012' 브랜치를 삭제합니다.)

== CMD\_DESCRIPTION\_BRANCH\_RENAME ==

브랜치의 이름을 변경합니다.

== CMD\_USAGE\_BRANCH\_RENAME ==

사용법:

cm ^branch | ^br ^rename <브랜치 사양> <새 이름>

브랜치 사양 이름을 변경할 브랜치.

(브랜치 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을 사용하십시오.)

새 이름 브랜치의 새 이름.

== CMD\_HELP\_BRANCH\_RENAME ==

비고:

이 명령은 브랜치 이름을 변경합니다.

예:

cm ^branch ^rename /main/task0 task1

('/main/task0' 브랜치의 이름을 '/main/task1'로 변경합니다.)

cm ^br ^rename ^br:/main@reptest@server2:8084 secondary

(리포지토리 'reptest'에 있는 'main' 브랜치의 이름을 'secondary'로 변경합니다.)

== CMD\_DESCRIPTION\_BRANCH\_HISTORY ==

브랜치 내역을 표시합니다.

== CMD\_USAGE\_BRANCH\_HISTORY ==

사용법:

cm ^branch | ^br ^history <브랜치 사양> [--^dateformat=<날짜 형식>]

[--^machinereadable]

브랜치 사양 내역을 가져올 브랜치 사양.

(브랜치 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을 사용하십시오.)

옵션:

--^dateformat 날짜 출력에 사용되는 형식.

--^machinereadable 결과를 구문 분석이 쉬운 형식으로 출력합니다.

== CMD\_HELP\_BRANCH\_HISTORY ==

예:

cm ^branch ^history ^br:/main/scm001@myrepository@myserver:8084

('myserver' 서버에 있는 리포지토리 'myrepository'의 브랜치인 '/main/scm001'의

내역을 표시합니다.)

cm ^br ^history main --^dateformat="yyyy, dd MMMM" --^machinereadable

(현재 리포지토리의 'main' 브랜치 내역을

지정된 날짜 형식과 구문 분석이 쉬운 형식으로 표시합니다.)

== CMD\_DESCRIPTION\_BRANCH\_SHOWMAIN ==

리포지토리의 주 브랜치를 표시합니다.

이는 'cm'만 자동화할 때 사용하는 자동화 명령입니다.

대부분 리포지토리의 주 브랜치는 '/main'입니다.

== CMD\_USAGE\_BRANCH\_SHOWMAIN ==

사용법:

cm ^branch | ^br ^showmain [<리포지토리 사양>] [--^encoding=<이름>]

[--^format=<형식 문자열>] [--^dateformat=<날짜 형식>]

리포지토리 사양 주 브랜치를 표시할 리포지토리

사양.

(리포지토리 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을 사용하십시오.)

옵션:

--^encoding 출력에 사용할 인코딩, 즉 UTF-8을

지정합니다.

지원되는 인코딩 및 형식에 대한 표를 확인하려면

http://msdn.microsoft.com/en-us/library/system.text.encoding.aspx에서

MSDN 문서를 확인하십시오.

페이지 마지막 "Name"열에 있습니다.

--^format 특정 형식으로 출력 메시지를 가져옵니다. 자세한 내용은

비고를 확인하십시오.

--^dateformat 날짜 출력에 사용되는 형식.

== CMD\_HELP\_BRANCH\_SHOWMAIN ==

비고:

이 명령은 리포지토리의 주 브랜치를 표시합니다.

이 명령은 출력을 표시할 형식 문자열을 허용합니다.

이 명령의 출력 매개 변수는 다음과 같습니다.

{^id} 브랜치 ID.

{^comment} 코멘트.

{^date} 날짜.

{^name} 이름.

{^owner} 소유자.

{^parent} 부모 브랜치 이름.

{^repository} 리포지토리.

{^repname} 리포지토리 이름.

{^repserver} 서버.

{^changeset} 브랜치의 헤드 체인지 세트.

예:

cm ^branch ^showmain

(현재 워크스페이스의 리포지토리에 대한 주 브랜치를 표시합니다.)

cm ^branch ^showmain repo@server:8084

('server:8084' 서버에 있는 리포지토리 'repo'에 대한 주 브랜치를

표시합니다.)

cm ^br ^showmain --^dateformat="yyyy, dd MMMM" --^encoding=utf8

(리포지토리의 주 브랜치를 지정된 날짜 형식으로 표시하고

출력 형식을 UTF-8로 지정합니다.)

cm ^br ^showmain --^format="{^id} - {^name}"

(리포지토리의 주 브랜치를 표시하고 브랜치의 ID와 이름만 출력합니다.)

== CMD\_DESCRIPTION\_BRANCH\_SHOWMERGES ==

병합 대기 중인 브랜치를 표시합니다.

== CMD\_USAGE\_BRANCH\_SHOWMERGES ==

이는 'cm'만 자동화할 때 사용하는 자동화 명령입니다.

주의해야 하는 만큼 사용자 친화적이지 않은 명령입니다.

사용법:

cm ^branch | ^br ^showmerges <항목 경로>[ ...]

[--^format=<형식 문자열>]

[--^dateformat=<날짜 형식>]

옵션:

--^format 특정 형식으로 출력 메시지를 가져옵니다. 자세한 내용은

비고를 확인하십시오.

--^dateformat 날짜 출력에 사용되는 형식.

== CMD\_HELP\_BRANCH\_SHOWMERGES ==

비고:

이 명령은 출력을 표시할 형식 문자열을 허용합니다.

이 명령의 출력 매개 변수는 다음과 같습니다.

{^id} 브랜치 ID.

{^comment} 코멘트.

{^date} 날짜.

{^name} 이름.

{^owner} 소유자.

{^parent} 부모 브랜치 이름.

{^parentid} 부모 브랜치 ID.

{^repid} 리포지토리 ID.

{^repository} 리포지토리.

{^repname} 리포지토리 이름.

{^repserver} 리포지토리 서버.

예:

cm ^branch ^showmerges file.txt

(보류 중인 'file.txt' 병합과 관련된 브랜치를 표시합니다.)

cm ^branch ^showmerges file.txt --^format="{^date} {^name}" --^dateformat="yyMMdd"

(병합과 관련된 브랜치를 표시하고 지정된 날짜 형식으로 날짜와 이름만

출력합니다.)

== CMD\_DESCRIPTION\_REPOSITORY ==

사용자가 리포지토리를 관리하도록 허용합니다.

== CMD\_USAGE\_REPOSITORY ==

사용법:

cm ^repository | ^repo <명령> [옵션]

명령:

^create | ^mk

^delete | ^rm

^list | ^ls

^rename

^add

각 명령에 대한 자세한 정보는 다음을 실행해 확인하십시오.

cm ^repository <명령> --^usage

cm ^repository <명령> --^help

== CMD\_HELP\_REPOSITORY ==

예:

cm ^repository

cm ^repository ^list

cm ^repository 새 리포지토리

cm ^repository ^create newrepo

cm ^repository ^rename 기존 이름 새 이름

cm ^repository ^add C:\repo\

== CMD\_DESCRIPTION\_REPOSITORY\_CREATE ==

서버에 리포지토리를 생성합니다.

== CMD\_USAGE\_REPOSITORY\_CREATE ==

사용법:

cm ^repository | ^repo <리포지토리 이름>

cm ^repository | ^repo <리포지토리 서버 사양> <리포지토리 이름>[ ...]

cm ^repository | ^repo [^create | ^mk] <리포지토리 이름>

리포지토리 서버 사양 리포지토리 서버의 사양.

(리포지토리 서버 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을

사용하십시오.)

리포지토리 이름 새 리포지토리의 이름.

공백을 사용하여 리포지토리 이름을 구분하십시오.

== CMD\_HELP\_REPOSITORY\_CREATE ==

예:

cm ^repository MyRep

cm ^repo 192.168.1.140:8087 Rep01 Rep01/ModuleA Rep01/ModuleB

cm ^repo ^create Rep01

cm ^repo ^mk 목록

== CMD\_DESCRIPTION\_REPOSITORY\_DELETE ==

서버에서 리포지토리를 삭제합니다.

== CMD\_USAGE\_REPOSITORY\_DELETE ==

사용법:

cm ^repository | ^repo ^delete | ^rm <리포지토리 사양>

옵션:

리포지토리 사양 리포지토리 사양입니다.

(리포지토리 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을 사용하십시오.)

== CMD\_HELP\_REPOSITORY\_DELETE ==

비고:

리포지토리 서버에서 리포지토리를 삭제합니다.

해당 데이터는 데이터베이스 백 엔드에서 제거되지 않지만, 더 이상

액세스할 수 없도록 연결이 해제됩니다.

(나중에 데이터를 다시 연결할 수 있습니다. 'cm ^repository ^add'를 참조하십시오.)

예:

cm ^repository ^delete myrepository@^repserver:myserver:8084

cm ^repository ^rm myrepository@myserver:8084

cm ^repo ^rm myrepository

== CMD\_DESCRIPTION\_REPOSITORY\_LIST ==

서버의 리포지토리를 나열합니다.

== CMD\_USAGE\_REPOSITORY\_LIST ==

사용법:

cm ^repository | ^repo [^list | ^ls] [<리포지토리 서버 사양>] [--^format=<문자열 형식>]

옵션:

리포지토리 서버 사양 리포지토리 서버의 사양.

(리포지토리 서버 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을

사용하십시오.)

--^format 특정 형식으로 출력 메시지를 가져옵니다. 자세한 내용은

비고를 확인하십시오.

== CMD\_HELP\_REPOSITORY\_LIST ==

비고:

이 명령은 출력을 표시할 형식 문자열을 허용합니다.

이 명령의 출력 매개 변수는 다음과 같습니다.

{^repid} | {0} 리포지토리 식별자.

{^repname} | {1} 리포지토리 이름.

{^repserver} | {2} 서버 이름.

{^repowner} | {3} 리포지토리 소유자.

{^repguid} | {4} 리포지토리의 고유 식별자.

{^tab} 탭 공백을 삽입합니다.

{^newline} 새 행을 삽입합니다.

형식 매개 변수 값이 '^TABLE'인 경우, 출력은 {^repid}, {^repname},

{^repserver} 필드가 있는 테이블 형식을 사용해 출력됩니다.

예:

cm ^repository

(모든 리포지토리를 나열합니다.)

cm ^repository ^list localhost:8084 --^format="{1, -20} {3}"

(20자 내의 리포지토리 이름, 공백, 리포지토리 소유자 순으로

왼쪽 정렬로 작성합니다.)

cm ^repository ^ls localhost:8084 --^format="{^repname, -20} {^repowner}"

(이전 예와 동일하게 작성합니다.)

cm ^repo ^ls localhost:8084 --^format=^TABLE

(리포지토리 목록을 리포지토리 ID, 리포지토리 이름, 리포지토리 서버 이름 필드가 있는

테이블 형식으로 작성합니다.)

== CMD\_DESCRIPTION\_REPOSITORY\_RENAME ==

리포지토리 이름을 변경합니다.

== CMD\_USAGE\_REPOSITORY\_RENAME ==

사용법:

cm ^repository | ^repo ^rename [<리포지토리 사양>] <새 이름>

리포지토리 사양 이름을 변경할 리포지토리.

(리포지토리 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을

사용하십시오.)

새 이름 새 리포지토리 이름.

== CMD\_HELP\_REPOSITORY\_RENAME ==

비고:

이 명령은 리포지토리 이름을 변경합니다.

리포지토리 사양이 지정되지 않으면 현재 리포지토리로 간주합니다.

예:

cm ^repository ^rename development

(현재 리포지토리의 이름이 'development'로 변경됩니다.)

cm ^repo ^rename ^rep:default@SERVER:8084 development

('SERVER'의 'default' 리포지토리의 이름이 'development'로 변경됩니다.)

== CMD\_DESCRIPTION\_REPOSITORY\_ADD ==

기존 리포지토리의 데이터베이스를 추가하여 기존 리포지토리를 연결합니다.

== CMD\_USAGE\_REPOSITORY\_ADD ==

사용법:

cm ^repository | ^repo ^add <DB 파일> <리포지토리 이름> <리포지토리 서버 사양>

DB 파일 데이터베이스 백 엔드에 있는 데이터베이스 파일 이름.

리포지토리 이름 리포지토리 이름.

리포지토리 서버 사양 리포지토리 서버 사양.

(서버 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을

사용하십시오.)

== CMD\_HELP\_REPOSITORY\_ADD ==

비고:

기존 리포지토리 데이터베이스를 서버에 다시 연결합니다.

예: 'cm ^repository ^delete' 명령을 사용한 후, '^add' 명령을 사용해

리포지토리를 한 서버에서 다른 서버로 이동하거나, 보관된 리포지토리를

복원합니다.

예:

cm ^repository ^add rep\_27 myrepository myserver:8084

== CMD\_DESCRIPTION\_TRIGGER\_CREATE ==

서버에 새 트리거를 생성합니다.

== CMD\_USAGE\_TRIGGER\_CREATE ==

사용법:

cm ^trigger | ^tr ^create | ^mk <하위 유형 유형> <새 이름> <스크립트 경로>

[--^position=<새 위치>]

[--^filter=<문자열 필터>]

[--^server=<리포지토리 서버 사양>]

하위 유형 유형 트리거 실행 및 트리거 작업.

트리거 유형을 확인하려면 'cm ^showtriggertypes'를

입력하십시오.

새 이름 새 트리거의 이름.

스크립트 경로 실행할 스크립트가 위치한 서버의 디스크

경로. 명령줄이 "^webtrigger "로 시작하는 경우,

트리거는 웹 트리거로 간주됩니다. 자세한 내용은

비고를 확인하십시오.

옵션:

--^position 지정된 트리거의 새 위치.

이 위치를 같은 유형의 다른 트리거가 사용 중이면

안 됩니다.

--^filter 지정된 필터와 일치하는 항목만 확인합니다.

--^server 지정된 서버에 트리거를 생성합니다.

서버가 지정되지 않은 경우, 클라이언트에 구성된 서버를

대상으로 명령을 실행합니다.

(서버 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을

사용하십시오.)

== CMD\_HELP\_TRIGGER\_CREATE ==

비고:

웹 트리거: "^webtrigger <대상 URI>"를 트리거 명령으로 입력하면

트리거가 생성됩니다. 이 경우, 트리거는 지정된 URI에 대해 POST 쿼리를

실행합니다. 이 URI의 요청 본문에는 트리거 환경 변수가 있는

JSON 사전과 문자열 배열을 가리키는

고정 INPUT 키가 포함됩니다.

예:

cm ^trigger ^create ^after-setselector "BackupMgr" "/path/to/script" --^position=4

cm ^tr ^mk ^before-mklabel new "/path/to/script" --^server=myserver:8084

cm ^tr ^mk ^after-mklabel Log "/path/to/script" --^filter="^rep:myRep,LB\*"

(이 트리거는 레이블 이름이 'LB'로 시작하고 레이블이 'myRep'이라는

리포지토리에서 생성되는 경우에만 실행됩니다.)

cm ^tr ^mk ^after-checkin NotifyTeam "^webtrigger http://myserver.org/api"

== CMD\_DESCRIPTION\_MOVE ==

파일 또는 디렉터리를 이동하거나 이름을 변경합니다.

== CMD\_USAGE\_MOVE ==

사용법:

cm ^move | ^mv <소스 경로> <대상 경로> [--^format=<문자열 형식>]

[--^errorformat=<문자열 형식>]

소스 경로 소스 항목 경로.

대상 경로 대상 항목 경로.

옵션:

--^format 특정 형식으로 출력 메시지를 가져옵니다. 자세한 내용은

비고를 확인하십시오.

--^errorformat 특정 형식으로 오류 메시지를 가져옵니다. 자세한 내용은

비고를 확인하십시오.

== CMD\_HELP\_MOVE ==

비고:

이 명령은 리포지토리에 있는 항목을 이동하거나 이름을 변경합니다.

변경사항은 로컬 파일 시스템에도 적용됩니다.

소스 경로가 파일인 경우, 대상 경로는 파일이나 디렉터리일 수

있습니다. 대상 경로가 파일인 경우, 파일의 이름이 변경되고, 디렉터리인 경우,

항목이 이동합니다.

소스 경로가 디렉터리인 경우, 대상 경로는 디렉터리여야 합니다.

이동하거나 이름을 변경할 항목이 반드시 존재해야 합니다.

형식:

{0} 소스 경로('--^format' 및 '--^errorformat' 모두)

{1} 대상 경로('--^format' 및 '--^errorformat' 모두)

예:

cm ^move file.txt file.old

(항목의 이름을 변경합니다.)

cm ^mv .\file.old .\oldFiles

('file.old'를 'oldFiles'로 이동합니다.)

cm ^move .\src .\src2

(디렉터리의 이름을 변경합니다.)

== CMD\_DESCRIPTION\_LABEL ==

사용자가 레이블을 관리하도록 허용합니다.

== CMD\_USAGE\_LABEL ==

사용법:

cm ^label | ^lb <명령> [옵션]

명령:

^create | ^mk

^delete | ^rm

^rename

각 명령에 대한 자세한 정보는 다음을 실행해 확인하십시오.

cm ^label <명령> --^usage

cm ^label <명령> --^help

== CMD\_HELP\_LABEL ==

예:

cm ^label myNewLabel ^cs:42

('^create' 명령은 선택사항입니다.)

cm ^label ^rename myNewLabel newLabelName

cm ^label ^delete newLabelName

== CMD\_DESCRIPTION\_LABEL\_CREATE ==

체인지 세트에 레이블을 적용하고 필요한 경우 레이블을 생성합니다.

== CMD\_USAGE\_LABEL\_CREATE ==

사용법:

cm ^label [^create] <레이블 사양> [<체인지 세트 사양> | <워크스페이스 경로>]

[--^allxlinkedrepositories]

[-^c=<문자열 코멘트> | -^commentsfile=<코멘트 파일>]

레이블 사양 새 레이블 이름.

(레이블 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을

사용하십시오.)

체인지 세트 사양 레이블을 지정할 체인지 세트의 이름이나 전체 사양.

(체인지 세트 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을

사용하십시오.)

워크스페이스 경로 레이블을 지정할 워크스페이스 경로. (워크스페이스가 가리키는

체인지 세트에 레이블이 지정됩니다.)

옵션:

--^allxlinkedrepositories 모든 Xlink 리포지토리에 새 레이블을 생성합니다.

-^c 새 레이블에 지정된 코멘트를 적용합니다.

-^commentsfile 새 레이블에 지정된 파일의 코멘트를

적용합니다.

== CMD\_HELP\_LABEL\_CREATE ==

비고:

코멘트를 입력할 편집기를 지정하려면 PLASTICEDITOR 환경 변수를

설정하십시오.

예:

cm ^label ^create ^lb:BL001 ^cs:1203 -^commentsfile=commentlb001.txt

(체인지 세트 1203에 연결된 레이블 'BL001'을 생성하고, 'commentlb001.txt' 파일의

코멘트를 적용합니다.)

cm ^label BL002 ^cs:1203 -^c="첫 번째 릴리스"

(체인지 세트 1203에 연결된 레이블 'BL002'를 코멘트와 함께 생성합니다.)

== CMD\_DESCRIPTION\_LABEL\_DELETE ==

하나 이상의 레이블을 삭제합니다.

== CMD\_USAGE\_LABEL\_DELETE ==

사용법:

cm ^label ^delete <레이블 사양>[ ...]

레이블 사양 삭제할 레이블. 공백을 사용하여 레이블을 구분하십시오.

(레이블 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을

사용하십시오.)

== CMD\_HELP\_LABEL\_DELETE ==

비고:

이 명령은 하나 이상의 레이블을 삭제합니다.

예:

cm ^label ^delete ^lb:BL001

(레이블 'BL001'을 삭제합니다.)

cm ^label ^delete ^lb:BL001 ^lb:BL002@reptest@server2:8084

(레이블 'BL001'과 'BL002'를 삭제합니다.)

== CMD\_DESCRIPTION\_LABEL\_RENAME ==

레이블의 이름을 변경합니다.

== CMD\_USAGE\_LABEL\_RENAME ==

사용법:

cm ^label ^rename <레이블 사양> <새 이름>

레이블 사양 이름을 변경할 레이블.

(레이블 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을 사용하십시오.)

새 이름 레이블의 새 이름.

== CMD\_HELP\_LABEL\_RENAME ==

비고:

이 명령은 레이블 이름을 변경합니다.

예:

cm ^label ^rename ^lb:BL001 BL002

(레이블 'BL001'의 이름을 'BL002'로 변경합니다.)

== CMD\_DESCRIPTION\_OBLITERATE ==

사용되지 않습니다

== CMD\_USAGE\_OBLITERATE ==

사용되지 않습니다.

== CMD\_HELP\_OBLITERATE ==

사용되지 않습니다.

== CMD\_DESCRIPTION\_OBJECTSPEC ==

객체 사양을 작성하는 방법을 설명합니다.

== CMD\_USAGE\_OBJECTSPEC ==

사용법:

cm ^objectspec

객체 사양을 작성하는 방법에 관한 모든 내용을 확인할 때 사용하십시오.

== CMD\_HELP\_OBJECTSPEC ==

일부 Plastic SCM 명령은 지정된 객체(주로 브랜치, 체인지 세트, 리포지토리 등)를

참조하기 위한 입력으로 '객체 사양'을 사용합니다.

이 문서는 사용할 수 있는 여러 "사양" 및 사양을 작성하는 방법을

설명합니다.

각 사양은 "^rep:" 또는 "^cs:". 등의 고유 태그로 시작합니다. 일반적인

객체 사양(예: "cm ^setowner 객체 사양")을 사용하는 명령에는 태그를 반드시

지정해야 하지만, 한 가지 유형의 사양만 사용하는 명령(예:"cm ^getfile 개정 사양")에는

태그를 생략할 수 있습니다.

-- 리포지토리 서버 사양 --

^repserver:name:port

예:

cm ^repo ^list ^repserver:skull:8084

cm ^repo ^list skull:8084

참고사항:

전례상으로 Plastic SCM에서는 '서버 사양' 대신 '리포지토리 서버 사양'이라는

명칭을 사용합니다. 예전에 워크스페이스와 리포지토리 서버를 구분한 이후

해당 명칭을 계속 사용하고 있습니다.

-- 리포지토리 사양 --

^rep:rep\_name@[리포지토리 서버 사양]

예:

cm ^showowner ^rep:codice@localhost:6060

(^showowner는 리포지토리 뿐만 아니라 다른 유형의 객체도 허용하므로

여기서는 "^rep:"를 사용해야 합니다. 따라서 사용자는 객체 유형을

명시해야 합니다.)

-- 브랜치 사양 --

^br:[/]br\_name[@repspec]

예:

cm ^switch ^br:/main@^rep:plastic@^repserver:skull:9095

(이 경우 "^br:", "^rep", "^repserver"를 사용할 필요가 없으므로

명령은 다음과 같이 더 짧은 형식을 허용합니다.

"cm ^switch main@plastic@skull:9095".)

cm ^find ^revisions "^where ^branch='^br:/main/task001'"

비고:

브랜치 맨 앞의 '/'는 생략할 수 있습니다. 이전에는 모든

브랜치를 /main, /main/task001과 같이 지정했습니다. 그러나 현재는 명령을

더 간략하게 할 수 있는 짧은 형식인 main, main/task001을 권장합니다.

-- 체인지 세트 사양 --

^cs:cs\_number|cs\_GUID[@repspec]

체인지 세트의 번호 또는 GUID를 지정할 수 있습니다.

예:

cm ^ls /code --^tree=ae1390ed-7ce9-4ec3-a155-e5a61de0dc77@code@skull:7070

-- 레이블 사양 --

^lb:lb\_name[@repspec]

예:

cm ^switch ^lb:RELEASE2.0

cm ^switch ^lb:RELEASE1.4@myrep@MYSERVER:8084

-- 개정 사양 --

개정 사양에는 다음과 같은 여러 유형이 있습니다.

^rev:item\_path[#(브랜치 사양|체인지 세트 사양|레이블 사양)]

^rev:^serverpath:item\_path#(브랜치 사양|체인지 세트 사양|레이블 사양)

^rev:^revid:rev\_id[@rep\_spec]

^rev:^itemid:item\_id#(브랜치 사양|체인지 세트 사양|레이블 사양)

예:

cm ^diff ^rev:readme.txt#^cs:19 ^rev:readme.txt#^cs:20

cm ^diff ^serverpath:/doc/readme.txt#^cs:19@myrepo \

^serverpath:/doc/readme.txt#^br:/main@myrepo@localhost:8084

cm ^cat ^revid:1230@^rep:myrep@^repserver:myserver:8084

-- 항목 사양 --

^item:path

거의 사용되지 않습니다.

예:

cm ^find ^revision "^where ^item='^item:.'"

-- 속성 사양 --

^att:att\_name[@repspec]

예:

cm ^attribute ^set ^att:merged@code@doe:8084 ^cs:25@code@doe:8084 완료

-- 임시 저장 사양 --

^sh:sh\_number[@repspec]

예:

cm ^diff ^sh:2 ^sh:4

-- 워크스페이스 사양 --

^wk:name@clientmachine

워크스페이스 관련 명령에만 적용되므로 거의 사용되지 않습니다. 경로 대신

이름 및 기기로 워크스페이스를 지정하려는 경우에 유용합니다.

예:

cm ^showselector ^wk:codebase@modok

참고사항:

이 사양들은 중앙 집중형 방식으로 워크스페이스 메타데이터를 저장하기 위해

'워크스페이스 서버'를 사용했던 Plastic SCM 2.x 버전에서 사용되던

것입니다. 성능 문제로 인해 더 이상 사용되지 않습니다.

== CMD\_DESCRIPTION\_PARTIAL ==

부분 워크스페이스에서 명령을 실행합니다.

== CMD\_USAGE\_PARTIAL ==

사용법:

cm ^partial <명령> [옵션]

명령:

^configure

^add

^undo

^co | ^checkout

^unco | ^undocheckout

^ci | ^checkin

^mv | ^move

^rm | ^remove

^stb | ^switch

^upd | ^update

각 명령에 대한 자세한 정보는 다음을 실행해 확인하십시오.

cm ^partial <명령> --^usage

cm ^partial <명령> --^help

== CMD\_HELP\_PARTIAL ==

예:

cm ^partial ^configure +/background-blue.png

cm ^partial ^update landscape-1024.png

cm ^partial ^checkin eyes-green.png eyes-black.png

== CMD\_DESCRIPTION\_PARTIAL\_ADD ==

버전 제어에 항목을 추가합니다.

== CMD\_USAGE\_PARTIAL\_ADD ==

사용법:

cm ^partial ^add [-^R | -^r | --^recursive] [--^silent] [--^parents]

[--^ignorefailed] [--^skipcontentcheck] <항목 경로>[ ...]

항목 경로 추가할 항목. 공백이 포함된 경로를 지정하려면 큰따옴표(" ")를

사용하십시오. 공백을 사용하여 경로를 구분하십시오.

현재 디렉터리의 모든 내용을 추가하려면 \*을 사용하십시오.

옵션:

-^R 항목을 재귀적으로 추가합니다.

--^silent 어떤 내용도 출력하지 않습니다.

--^parents 작업에 지정된 항목의 부모 디렉터리를

포함합니다.

--^ignorefailed 항목을 추가할 수 없는 경우, 추가 작업이 해당 항목을 제외하고

계속 진행됩니다. 참고: 디렉터리를 추가할 수 없는 경우

디렉터리의 내용이 추가되지 않습니다.

--^skipcontentcheck 확장자가 부족하여 파일을 텍스트나 바이너리로

설정할 수 없는 경우, 내용을 확인하는 대신

기본적으로 파일을 바이너리로 설정하여 유형을 감지합니다.

== CMD\_HELP\_PARTIAL\_ADD ==

비고:

항목 추가 요구 사항:

- 추가할 항목의 부모 디렉터리가 미리 추가되어 있어야 합니다.

예:

cm ^partial ^add pic1.png pic2.png

('pic1.png' 및 'pic2.png' 항목을 추가합니다.)

cm ^partial ^add c:\workspace\picture.png

('picture.png' 항목을 경로 'c:\workspace'에 추가합니다.)

cm ^partial ^add -^R c:\workspace\src

('src'를 재귀적으로 추가합니다.)

cm ^partial ^add --^parents samples\design01.png

('design01.png' 파일과 'samples' 부모 폴더를 추가합니다.)

cm ^partial ^add -^R \*

(현재 디렉터리의 모든 내용을 재귀적으로 추가합니다.)

== CMD\_DESCRIPTION\_PARTIAL\_CHECKIN ==

변경사항을 리포지토리에 저장합니다.

== CMD\_USAGE\_PARTIAL\_CHECKIN ==

사용법:

cm ^partial ^checkin | ^ci [<항목 경로>[ ...]]

[-^c=<문자열 코멘트> | -^commentsfile=<코멘트 파일>]

[--^all | -^a] [--^applychanged] [--^keeplock]

[--^symlink] [--^ignorefailed]

옵션:

항목 경로 체크인할 항목. 공백이 포함된 경로를 지정하려면

큰따옴표(" ")를 사용하십시오. 공백을 사용하여 경로를

구분하십시오.

현재 디렉터리에 체크인을 적용하려면 온점(.)을 사용하십시오.

-^c 체크인 작업에서 생성된 체인지 세트에 코멘트를

지정합니다.

-^commentsfile 체크인 작업에서 생성된 체인지 세트에

지정된 파일의 코멘트를 적용합니다.

--^all | -^a 지정된 경로에서 로컬로 변경, 이동, 삭제된 항목도

포함합니다.

--^applychanged 체크아웃된 항목과 함께 워크스페이스에서 감지된

변경된 항목에 체크인 작업을

적용합니다.

--^keeplock 체크인 작업 후에 잠긴 항목의 잠금을

유지합니다.

--^symlink 대상이 아닌 기호 링크에 체크인 작업을

적용합니다.

--^ignorefailed 적용할 수 없는 모든 변경사항(잠금

즉, 배타적 체크아웃을 가져올 수 없거나, 로컬 변경이

서버 변경사항과 충돌이 일어나는 경우)은

변경이 취소되고 해당 변경사항을 제외하고 체크인 작업이

계속 진행됩니다.

== CMD\_HELP\_PARTIAL\_CHECKIN ==

비고:

- <항목 경로>를 지정하지 않은 경우, 체크인에

워크스페이스의 모든 대기 중인 변경사항이 포함됩니다.

- 체크인 작업은 항상 지정된 경로에서 재귀적으로 적용됩니다.

- 항목을 체크인하려면 다음 조건을 충족해야 합니다.

- 항목이 소스 코드 제어 상태여야 합니다.

- 항목이 체크아웃된 상태여야 합니다.

- 항목이 변경되었으나 체크아웃되지 않은 경우, <항목 경로>가 디렉터리이거나,

경로에 임의 문자 기호('\*')가 없는 한 '--^applychanged' 플래그는 필요하지

않습니다.

개정의 내용이 이전 개정과 달라야 체크인할 수

있습니다.

코멘트를 입력할 편집기를 지정하려면 PLASTICEDITOR 환경 변수를

설정하십시오.

표준 입력에서 입력 읽기:

'^partial ^checkin' 명령은 표준 입력에서 경로를 읽을 수 있습니다. 그러려면 단일 대시

"-" 기호를 넣으십시오.

예: cm ^partial ^checkin -

경로 읽기는 빈 행이 입력될 때까지 계속됩니다.

따라서 파이프를 사용해 체크인할 파일을 지정할 수 있습니다.

예:

dir /S /B \*.c | cm ^partial ^checkin --^all -

(Windows에서는 워크스페이스에 있는 모든 .c 파일을 체크인합니다.)

예:

cm ^partial ^checkin figure.png landscape.png

(체크아웃된 'figure.png' 및 'landscape.png' 파일에 체크인을 적용합니다.)

cm ^partial ^checkin . -^commentsfile=mycomment.txt

(현재 디렉터리에 체크인을 적용하고 'mycomment.txt' 파일의 코멘트를

설정합니다.)

cm ^partial ^ci background.png -^c="내 코멘트" --^keeplock

('background.png'에 체크인을 적용하고, 코멘트를 포함하고, 잠금을

유지합니다.)

cm ^partial ^checkin --^applychanged

(워크스페이스의 모든 대기 중인 변경사항에 체크인을 적용합니다.)

cm ^partial ^checkin 링크 --^symlink

(대상이 아닌 링크 파일에 체크인을 적용합니다.

UNIX 환경에서 사용할 수 있습니다.)

cm ^partial ^checkin . --^ignorefailed

(현재 디렉터리에 체크인을 적용하고 적용할 수 없는 변경사항은

무시합니다.)

== CMD\_DESCRIPTION\_PARTIAL\_CHECKOUT ==

파일을 수정 준비 완료 상태로 표시합니다.

== CMD\_USAGE\_PARTIAL\_CHECKOUT ==

사용법:

cm ^partial ^checkout | ^co [<항목 경로>[ ...]] [--^resultformat=<문자열 형식>]

[--^silent] [--^ignorefailed]

옵션:

항목 경로 체크아웃할 항목. 공백이 포함된 경로를 지정하려면

큰따옴표(" ")를 사용하십시오. 공백을 사용하여 경로를

구분하십시오.

현재 디렉터리에 체크아웃을 적용하려면 온점(.)을 사용하십시오.

--^resultformat 특정 형식으로 출력 결과 메시지를

가져옵니다.

--^silent 어떤 내용도 출력하지 않습니다.

--^ignorefailed 항목을 잠글 수 없는 경우(배타적 체크아웃을

수행할 수 없는 경우), 체크아웃 작업이 해당 항목을 제외하고

계속 진행됩니다.

== CMD\_HELP\_PARTIAL\_CHECKOUT ==

비고:

항목을 체크아웃하려면 다음 조건을 충족해야 합니다.

- 항목이 소스 코드 제어 상태여야 합니다.

- 항목이 체크인된 상태여야 합니다.

서버에서 잠금이 구성된 경우(lock.conf가 있는 경우), 경로에서 체크아웃이

발생할 때마다 Plastic은 체크아웃이 규칙을 충족하는지 확인하고,

규칙을 충족하는 경우, 다른 사용자가 동시에 체크아웃할 수 없도록 경로가

배타적 체크아웃(잠김) 상태가 됩니다.

'cm ^lock ^list'를 사용해 서버의 모든 잠금을 확인할 수 있습니다.

잠금 작동 방식에 대해 알아보려면 관리자 가이드를 확인하십시오.

https://www.plasticscm.com/download/help/locking

예:

cm ^partial ^checkout pic1.png pic2.png

('pic1.png' 및 'pic2.png' 파일을 체크아웃합니다.)

cm ^partial ^co \*.png

(모든 png 파일을 체크아웃합니다.)

cm ^partial ^checkout .

(현재 디렉터리를 체크아웃합니다.)

cm ^partial ^checkout -^R c:\workspace\src

('src' 폴더를 재귀적으로 체크아웃합니다.)

== CMD\_DESCRIPTION\_PARTIAL\_CONFIGURE ==

워크스페이스에 항목을 로드하거나 로드를 취소하여 워크스페이스를 구성하도록 허용합니다.

== CMD\_USAGE\_PARTIAL\_CONFIGURE ==

사용법:

cm ^partial ^configure <+|-경로>[ ...] [--^silent] [--^ignorefailed]

[--^ignorecase] [--^restorefulldirs]

경로 로드하거나 로드를 취소할 경로. 공백이 포함된 경로를 지정하려면

큰따옴표(" ")를 사용하십시오. 공백을 사용하여 경로를

구분하십시오.

경로는 "/"로 시작해야 합니다.

옵션:

--^silent 어떤 내용도 출력하지 않습니다.

--^ignorefailed 프로세스 중에 모든 오류를 건너뜁니다. 잘못된 경로를

입력해도 명령이 중단되지 않습니다.

--^ignorecase 경로의 대소문자를 무시합니다. 이 플래그를 사용하면,

사용자가 경로를 "/Data/Textures"라 써도 '^configure'가

"/data/teXtures"에서 실행됩니다.

--^restorefulldirs 부분 워크스페이스에서 전체 작업이 실행되는 경우 발생하는

잘못된 디렉터리 구성을 재설정합니다.

이 목록의 디렉터리는 완전히 구성(모두 선택)되므로

업데이트 중에 새로운 내용을 자동으로

다운로드합니다.

이 작업은 파일을 다운로드하지 않으며, 부분

워크스페이스의 디렉터리 구성을

복원합니다.

== CMD\_HELP\_PARTIAL\_CONFIGURE ==

비고:

해당 명령은 재귀 작업으로 간주합니다.

예:

cm ^partial ^configure +/landscape\_grey.png

('landscape\_grey.png' 항목을 로드합니다.)

cm ^partial ^configure -/landscape\_black.png

('landscape\_black.png' 항목의 로드를 취소합니다.)

cm ^partial ^configure +/soft -/soft/soft-black.png

('soft-black.png'를 제외한 모든 'soft' 디렉터리 자식 항목을 로드합니다.)

cm ^partial ^configure -/

(전체 워크스페이스 로드를 취소합니다.)

cm ^partial ^configure -/ +/

(전체 워크스페이스를 로드합니다.)

cm ^partial ^configure -/figure-64.png --^ignorefailed

('figure-64.png' 항목의 로드가 이미 취소되었어도 로드를 취소합니다.)

cm ^partial ^configure +/ --^restorefulldirs

(새로운 내용을 자동으로 다운로드하도록 모든 디렉터리를 설정합니다.)

cm ^partial ^configure +/src/lib --^restorefulldirs

('/src/lib' 및 그 하위 디렉터리만 새로운 내용을 자동으로 다운로드하도록

설정합니다.)

== CMD\_DESCRIPTION\_PARTIAL\_MOVE ==

파일 또는 디렉터리를 이동하거나 이름을 변경합니다.

== CMD\_USAGE\_PARTIAL\_MOVE ==

사용법:

cm ^partial ^move | ^mv <소스 경로> <대상 경로> [--^format=<문자열 형식>]

소스 경로 소스 항목 경로.

대상 경로 대상 항목 경로.

옵션:

--^format 특정 형식으로 출력 메시지를 가져옵니다. 자세한 내용은

비고를 확인하십시오.

== CMD\_HELP\_PARTIAL\_MOVE ==

비고:

이 명령은 리포지토리에 있는 항목을 이동하거나 이름을 변경합니다.

변경사항은 로컬 파일 시스템에도 적용됩니다.

소스 경로가 파일인 경우, 대상 경로는 파일이나 디렉터리일 수

있습니다. 대상 경로가 파일인 경우, 파일의 이름이 변경되고, 디렉터리인 경우,

항목이 이동합니다.

소스 경로가 디렉터리인 경우, 대상 경로는 디렉터리여야 합니다.

이동하거나 이름을 변경할 항목이 반드시 존재해야 합니다.

형식:

{0} 소스 경로.

{1} 대상 경로.

예:

cm ^partial ^move file.png file-blue.png

(항목의 이름을 변경합니다.)

cm ^partial ^mv .\file-blue.png .\blueFiles

('file-blue.png'를 'blueFiles'로 이동합니다.)

cm ^partial ^move .\design .\marketing

(디렉터리의 이름을 변경합니다.)

== CMD\_DESCRIPTION\_PARTIAL\_RM ==

버전 제어에서 파일 또는 디렉터리를 삭제합니다.

== CMD\_USAGE\_PARTIAL\_RM ==

사용법:

cm ^partial ^remove | ^rm <항목 경로>[ ...] [--^nodisk]

항목 경로 제거할 항목 경로입니다. 공백이 포함된 경로를 지정하려면

큰따옴표(" ")를 사용하십시오. 공백을 사용하여 경로를

구분하십시오.

옵션:

--^nodisk 항목이 버전 제어에서는 제거되지만, 디스크에서는 유지됩니다.

== CMD\_HELP\_PARTIAL\_RM ==

비고:

디스크에서 항목이 삭제됩니다. 삭제된 항목이 소스 코드 제어의 부모 디렉터리에서

삭제됩니다.

요구사항:

- 항목이 소스 코드 제어 상태여야 합니다.

예:

cm ^partial ^remove src

('src'를 제거합니다. 'src'가 디렉터리인 경우 다음과 같습니다.

cm ^partial ^remove -^R src.)

cm ^partial ^remove c:\workspace\pic01.png --^nodisk

('pic01.png'가 버전 제어에서는 제거되지만, 디스크에서는 유지됩니다.)

== CMD\_DESCRIPTION\_PARTIAL\_SWITCH ==

브랜치를 작업 브랜치로 설정합니다.

== CMD\_USAGE\_PARTIAL\_SWITCH ==

사용법:

cm ^switch <브랜치 사양> [--^report | --^silent] [--^workspace=<경로>]

(작업 브랜치를 설정하고 워크스페이스를 업데이트합니다.)

cm ^switch <브랜치 사양> --^configure <+|-경로>[ ...] [--^silent]

[--^ignorefailed] [--^ignorecase] [--^workspace=<경로>]

(작업 브랜치를 설정하고 'cm

^partial ^configure' 명령과 같이 워크스페이스 구성을 실행합니다.)

브랜치 사양 브랜치 사양입니다. (브랜치 사양에 관한 자세한 내용을 확인하려면

'cm ^help ^objectspec'을 사용하십시오.)

경로 로드하거나 로드를 취소할 경로. 공백이 포함된 경로를

지정하려면 큰따옴표(" ")를 사용하십시오. 공백을 사용하여 경로를

구분하십시오. 경로는 반드시 "/"로 시작해야 합니다.

옵션:

--^silent 어떤 내용도 출력하지 않습니다.

--^report 명령이 완료되면 적용된 변경사항 목록을

출력합니다. '--^silent'를 사용하면 이 설정을 재정의합니다.

이 옵션은 '--^configure' 옵션이 지정되지 않은 경우에만

작동합니다.

--^configure 작업 브랜치를 업데이트한 후 워크스페이스를

구성(항목 로드/로드 취소)합니다. 'cm ^partial

^configure --^help'를 사용해 구성할 경로를 지정하는

방법을 확인할 수 있습니다.

--^ignorefailed 구성 프로세스 중에 모든 오류를 건너뜁니다.

잘못된 경로를 입력해도 명령이 중단되지 않습니다.

--^ignorecase 경로의 대소문자를 무시합니다. 이 플래그를 사용하면,

사용자가 경로를 "/Data/Textures"라 써도 '--^configure' 옵션이

"/data/teXtures"에서 작동됩니다.

--^workspace=경로 워크스페이스가 위치한 경로.

== CMD\_HELP\_PARTIAL\_SWITCH ==

비고:

이 명령을 사용하면 사용자는 작업 브랜치를 업데이트할 수 있습니다. 브랜치를 업데이트한 후,

명령은 'cm ^partial ^update' 명령과 같이 워크스페이스를

새 브랜치로 업데이트합니다. 그러나, '--^configure' 옵션이

지정된 경우, 해당 명령을 사용하면 'cm ^partial ^configure' 명령과 같이 새 브랜치 구성을

사용하여 워크스페이스를 구성할 수 있습니다.

예:

cm ^switch ^br:/main/task

(/main/task를 작업 브랜치로 설정하고 워크스페이스를 업데이트합니다.)

cm ^switch ^br:/main/task --^configure +/art/images

(/main/task를 작업 브랜치로 설정하고 /art/images 폴더를 로드하도록

워크스페이스를 구성합니다.)

== CMD\_DESCRIPTION\_PARTIAL\_UNCO ==

항목의 체크아웃을 취소합니다.

== CMD\_USAGE\_PARTIAL\_UNCO ==

사용법:

cm ^partial ^undocheckout | ^unco <항목 경로>[ ...] [--^silent]

항목 경로 작업을 적용할 항목입니다. 공백이 포함된 경로를

지정하려면 큰따옴표(" ")를 사용하십시오. 공백을 사용하여 경로를

구분하십시오.

현재 디렉터리에 작업을 적용하려면 온점(.)을 사용하십시오.

옵션:

--^silent 어떤 내용도 출력하지 않습니다.

== CMD\_HELP\_PARTIAL\_UNCO ==

비고:

항목이 체크아웃되고 이를 체크인하지 않으려는 경우, 이 명령을 사용해 체크아웃을

취소할 수 있습니다. 파일과 폴더 모두 체크아웃 취소할 수

있습니다. 항목이 체크아웃하기 전 상태로 업데이트됩니다.

요구사항:

- 항목이 소스 코드 제어 상태여야 합니다.

- 항목이 체크아웃된 상태여야 합니다.

예:

cm ^partial ^undocheckout .

(현재 디렉터리의 체크아웃을 취소합니다.)

cm ^partial ^undocheckout pic1.png pic2.png

cm ^unco c:\workspace\design01.png

(선택한 파일의 체크아웃을 취소합니다.)

== CMD\_DESCRIPTION\_PARTIAL\_UNDO ==

워크스페이스의 변경사항을 실행 취소합니다.

== CMD\_USAGE\_PARTIAL\_UNDO ==

사용법:

cm ^partial ^undo [<경로>[ ...]] [--^symlink] [-^r | --^recursive]

[<필터>[ ...]]

[--^silent | --^machinereadable [--^startlineseparator=<구분 기준>]

[--^endlineseparator=<구분 기준>] [--^fieldseparator=<구분 기준>]]

경로 작업을 적용할 파일 또는 디렉터리의

경로. 공백이 포함된 경로를 지정하려면

큰따옴표(" ")를 사용하십시오. 공백을 사용하여 경로를

구분하십시오.

경로가 지정되지 않은 경우, 기본적으로 실행 취소

작업은 현재 디렉터리의 모든 파일을 대상으로

합니다.

필터 지정된 필터를 주어진 경로에

적용합니다. 공백을 사용하여 필터를 구분하십시오. 자세한 내용은

필터 섹션을 확인하십시오.

옵션:

--^symlink 대상이 아닌 기호 링크에 실행 취소 작업을

적용합니다.

-^r 실행 취소를 재귀적으로 실행합니다.

--^silent 어떤 내용도 출력하지 않습니다.

--^machinereadable 결과를 구문 분석이 쉬운 형식으로 출력합니다.

--^startlineseparator '--^machinereadable' 플래그와 함께 사용되며,

행이 시작되는 방법을 지정합니다.

--^endlineseparator '--^machinereadable' 플래그와 함께 사용되며,

행이 끝나는 방법을 지정합니다.

--^fieldseparator '--^machinereadable' 플래그와 함께 사용되며,

필드를 구분할 방법을 지정합니다.

필터:

플래그가 지정되지 않으면 기본적으로 모든 변경사항이 실행 취소되지만,

아래의 플래그를 하나 이상 사용하여 경로를 필터링할 수 있습니다.

파일 또는 디렉터리가 하나 이상의 지정된 변경사항 유형과 일치하는 경우

해당 파일 또는 디렉터리의 변경사항이 모두 실행 취소됩니다.

예를 들어 '--^checkedout' 및 '--^moved'를 지정하여 파일을

모두 체크아웃 및 이동한 경우에는 두 변경사항 모두 실행 취소됩니다.

--^checkedout 체크아웃된 파일 및 디렉터리를 선택합니다.

--^unchanged 내용이 변경되지 않은 파일을 선택합니다.

--^changed 로컬에서 변경되거나 체크아웃된 파일 및 디렉터리를

선택합니다.

--^deleted 삭제된 파일 및 디렉터리를 선택합니다.

--^moved 이동한 파일 및 디렉터리를 선택합니다.

--^added 추가된 파일 및 디렉터리를 선택합니다.

== CMD\_HELP\_PARTIAL\_UNDO ==

비고:

^undo 명령은 되돌릴 수 없는 방식으로 작업을 실행 취소하므로 위험합니다.

^undo가 완료되면 작업의 영향을 받은 파일 및 디렉터리를

이전 상태로 복원할 수 없습니다. 인수로 경로가 지정되지

않은 경우 실행 취소 작업은 기본적으로 현재 디렉터리의 모든 변경사항을 실행 취소하지만

작업이 재귀적으로 이루어지지는 않습니다.

/src 디렉터리에서 실행하는 경우에도 마찬가지입니다.

/src

|- file.txt

|- code.cs

\- /test

|- test\_a.py

\- test\_b.py

cm ^partial ^undo

cm ^partial ^undo \*

cm ^partial ^undo file.txt code.cs /test

cm ^partial ^undo .

cm ^partial ^undo /src file.txt code.cs

작업을 재귀적으로 수행하려면 '-^r' 플래그를 지정해야 합니다.

디렉터리 아래의 모든 변경사항을 실행 취소하려는 경우(디렉터리 자체에

영향을 주는 변경사항 포함):

cm ^partial ^undo 디렉터리 경로 -^r

디렉터리 경로가 워크스페이스 경로인 경우 워크스페이스의 모든 변경사항이

실행 취소됩니다.

예:

cm ^partial ^undo . -^r

(현재 디렉터리의 모든 변경사항을 재귀적으로 실행 취소합니다. 워크스페이스의

루트에서 실행되면 전체 워크스페이스의 모든 변경사항을 실행 취소합니다.)

cm ^partial ^co file.txt

cm ^partial ^undo file.txt

(file.txt의 체크아웃을 취소합니다.)

^echo ^content >> file.txt

cm ^partial ^undo file.txt

(file.txt의 로컬 변경사항을 취소합니다.)

cm ^partial ^undo src

(src 디렉터리 및 포함된 파일의 변경사항을 취소합니다.)

cm ^partial ^undo src/\*

(src에 포함된 모든 파일 및 디렉터리의 변경사항을 취소합니다.

src에는 영향이 없습니다.)

cm ^partial ^undo \*.cs

(현재 디렉터리에서 \*.cs와 일치하는 모든 파일 또는 디렉터리의 변경사항을

취소합니다.)

cm ^partial ^undo \*.cs -^r

(현재 디렉터리와 모든 하위 디렉터리에서 \*.cs와 일치하는 모든 파일 또는

디렉터리의 변경사항을 취소합니다.)

cm ^partial ^co file1.txt file2.txt

^echo ^content >> file1.txt

cm ^partial ^undo --^unchanged

(변경되지 않은 file2.txt의 체크아웃을 실행 취소하고 로컬에서 변경된

file1.txt를무시합니다.)

^echo ^content >> file1.txt

^echo ^content >> file2.txt

cm ^partial ^co file1.txt

cm ^partial ^undo --^checkedout

(체크아웃된 file1.txt 파일의 변경사항을 취소하고 file2.txt 파일은 체크아웃되지 않았으므로

무시합니다.)

cm ^partial ^add file.txt

cm ^partial ^undo file.txt

(file.txt 추가를 실행 취소하여 다시 비공개 파일로 만듭니다.)

^rm file1.txt

^echo ^content >> file2.txt

cm ^partial ^add file3.txt

cm ^partial ^undo --^deleted --^added \*

(file1.txt 삭제 및 file3.txt 추가를 실행 취소하고 file2.txt의 변경사항을

무시합니다.)

== CMD\_DESCRIPTION\_PARTIAL\_UPDATE ==

부분 워크스페이스를 업데이트하고 최근 변경사항을 다운로드합니다.

== CMD\_USAGE\_PARTIAL\_UPDATE ==

사용법:

cm ^partial ^update [<항목 경로>[ ...]] [--^changeset=<번호>]

[--^silent | --^report] [--^dontmerge]

항목 경로 업데이트할 항목. 공백이 포함된 경로를 지정하려면

큰따옴표(" ")를 사용하십시오. 공백을 사용하여 경로를

구분하십시오.

현재 디렉터리에 업데이트를 적용하려면 온점(.)을 사용하십시오.

경로가 지정되지 않은 경우, 현재 부분 워크스페이스

전체가 업데이트됩니다.

옵션:

--^changeset 부분 워크스페이스를 특정 체인지 세트로 업데이트합니다.

--^silent 어떤 내용도 출력하지 않습니다.

--^report 명령이 완료되면 적용된 변경사항 목록을

출력합니다. '--^silent'를 사용하면 이 설정을 재정의합니다.

--^dontmerge 파일 충돌을 병합하지 않고 건너뜁니다.

다른 변경사항은 올바르게 적용됩니다. 이 옵션은 사용자

상호작용을 방지하기 위한 자동화에 유용합니다.

== CMD\_HELP\_PARTIAL\_UPDATE ==

비고:

'^partial ^update' 명령은 이전 파일을 업데이트합니다.

해당 명령은 재귀 작업으로 간주합니다.

'--^changeset' 옵션을 사용할 때 지정된 모든 경로가 같은 Xlink 내의

파일인 경우, 다운로드할 버전이 Xlink 리포지토리의 지정된 체인지 세트에서

검색됩니다.

예:

cm ^partial ^update

(현재 부분 워크스페이스의 모든 항목을 업데이트합니다.)

cm ^partial ^update .

(현재 디렉터리의 모든 자식 항목을 업데이트합니다.)

cm ^partial ^update backgroud-blue.png

('backgroud-blue.png' 항목을 업데이트합니다.)

cm ^partial ^update soft\_black.png soft-grey.png

('soft\_black.png' 및 'soft-grey.png' 항목을 업데이트합니다.)

cm ^partial ^update src --^report

('src' 디렉터리의 모든 자식 항목을 업데이트하고 적용된 변경사항 목록을

마지막에 출력합니다.)

cm ^partial ^update src --^changeset=4

('src' 디렉터리의 모든 자식 항목을 체인지 세트 4에 로드된 내용으로

업데이트합니다.)

cm ^partial ^update xlink/first.png --^changeset=4

(''xlink/first.png' 항목을 Xlink 리포지토리의 체인지 세트 4에 로드된 내용으로

업데이트합니다.)

== CMD\_DESCRIPTION\_PATCH ==

사양에서 패치 파일을 생성하거나 생성된 패치를 현재 워크스페이스에

적용합니다.

== CMD\_USAGE\_PATCH ==

사용법:

cm ^patch <소스 사양> [<소스 사양>] [--^output=<출력 파일>]

[--^tool=<비교 경로>]

브랜치, 체인지 세트의 차이점 또는 체인지 세트 간의 차이점을 포함한

패치 파일을 생성합니다. 텍스트 및

바이너리 파일의 차이점도 추적합니다.

cm ^patch --^apply <패치 파일> [--^tool=<패치 경로>]

현재 워크스페이스에서 생성된 패치 파일의 내용을 적용하도록

허용합니다.

소스 사양 체인지 세트 또는 브랜치의 전체 사양. (사양에 관한

자세한 내용을 확인하려면 'cm ^help ^objectspec'을 사용하십시오.)

출력 파일 패치 내용을 저장할 파일. 파일이 지정되지 않은

경우, 패치 내용은 표준 출력에 출력됩니다.

패치 파일 현재 워크스페이스에 적용할 패치 파일.

옵션:

--^output 패치 명령의 출력 파일을 설정합니다.

--^tool 사용할 애플리케이션(비교 또는 패치)을 설정합니다.

== CMD\_HELP\_PATCH ==

제한사항:

출력 패치 파일이 이미 있는 경우, 명령이 기존 출력 패치 파일을 덮어쓰지 않습니다.

패치를 적용할 때 디스크에 수정된 파일이 없는 경우, 명령은 변경사항을 수정된 파일에

적용하지 않습니다.

중요:

이 명령을 사용하려면 비교 및 패치 툴이 필요합니다.

http://gnuwin32.sourceforge.net/packages/patch.htm과

http://gnuwin32.sourceforge.net/packages/diffutils.htm에서 다운로드할 수 있습니다.

설치가 완료되면 PATH 환경 변수에 해당 위치를 추가하는 것이

좋습니다.

예:

cm ^patch ^cs:4@default@localhost:8084

(통합된 형식으로 체인지 세트 4의 차이점을 콘솔에 출력합니다.)

cm ^patch ^br:/main --^output=file.patch

("main" 브랜치의 차이점이 있는 패치 파일을 생성합니다.)

cm ^patch ^br:/main --^output=file.patch --^tool=C:\gnu\diff.exe

(위와 같지만, 커스텀 실행 파일(exe)을 사용합니다.)

cm ^patch ^cs:2@default ^cs:4@default

(통합된 형식으로 체인지 세트 2와 4의 차이점을 콘솔에 출력합니다.)

cm ^patch --^apply file.patch --^tool=C:\gnu\patch.exe

(커스텀 실행 파일(exe)을 사용하여 'file.patch'의 패치를 로컬 워크스페이스에 적용합니다.)

== CMD\_DESCRIPTION\_QUERY ==

SQL 쿼리를 실행합니다. SQL 스토리지가 필요합니다.

== CMD\_USAGE\_QUERY ==

사용법:

cm ^query <SQL 명령> [--^outputfile=<출력 파일>]

[--^solveuser=<열 이름>[,...]]

[--^solvepath=<열 이름>[,...]]

[--^columnwidth=<값>] [--^nocolumnname]

[--^columnseparator=<구분 기준>] [--^repository=<이름>]

SQL 명령 실행할 SQL 쿼리.

옵션:

--^outputfile 출력 파일에 결과를 작성합니다.

--^solveuser 지정된 열을 사용자 이름 열로 설정합니다. 쿼리

인터프리터는 해당 열의 데이터가 사용자라 가정하고

해결을 시도합니다.

--^solvepath 지정된 열을 항목 ID 열로 설정합니다. 쿼리

인터프리터는 항목 ID를 파일 시스템 경로로 해결하려고

시도합니다.

--^columnwidth 출력 형식에 맞게 각 열의 너비를 지정합니다.

--^nocolumnname 열 이름을 출력하지 않습니다.

--^columnseparator 탭(\t) 대신 문자를 열 구분 기준으로 사용합니다.

--^repository 쿼리할 리포지토리입니다.

== CMD\_HELP\_QUERY ==

비고:

이 명령을 통해 사용자는 서버 데이터베이스에서 SQL 쿼리를 실행할 수 있습니다.

SQL 쿼리를 작성하려면 사전 정의된 다음 두 가지 함수를 사용하여 사용자 및 경로를

관리하십시오.

- '^SolveUser(<사용자 이름>)': 사용자 이름을 Plastic SCM 형식으로 바꿉니다.

- '^SolvePath(<경로>)': 디스크 경로를 항목 ID로 바꿉니다.

사람이 판독 가능한 형식으로 쿼리 결과를 표시하는 옵션을 사용할 수도 있습니다.

'--^solveuser=<열 이름>' 및

'--^solvepath=<열 이름>' 옵션을 사용하면 쿼리 인터프리터가 읽을 수 있는 텍스트로

변환해야 하는 열을 지정할 수 있습니다. 쉼표로 구분하여 2개 이상의 열 이름을

지정할 수 있습니다.

예:

cm ^query "^SELECT \* ^FROM ^revision" --^columnwidth=25 --^repository=reptest

('reptest' 리포지토리의 '개정' 테이블에서 데이터를 가져옵니다.)

cm ^query "^SELECT b.^sname ^as 브랜치 이름, o.^dtimestamp ^as 날짜 ^from ^branch b, \

^object o, ^seid s ^where b.^iobjid=o.^iobjid ^and o.^fidowner=s.^iseidid ^and \

s.^scode='^SolveUser(john)'" --^outputfile=query.txt

(소유자 'john'의 브랜치로 파일을 출력합니다.)

cm ^query "^select r.^iobjid, r.^fiditem ^as 경로, s.^scode ^as 사용자 이름 ^FROM \

^revision r, ^object o, ^seid s ^WHERE r.^iobjid=o.^iobjid ^and \

o.^fidowner=s.^iseidid ^and o.^dtimestamp>2014/04/25" \

--^solveuser=사용자 이름 --^solvepath=경로 --^repository=reptest@server2:9095

(선택한 리포지토리에서 선택한 데이터를 가져옵니다.)

cm ^query "^SELECT \* ^FROM ^revision ^WHERE ^fiditem=^SolvePath(c:\mywkpath\info)"

('info' 경로의 모든 개정 데이터를 가져옵니다.)

== CMD\_DESCRIPTION\_ATTRIBUTE\_DELETE ==

하나 이상의 속성을 삭제합니다.

== CMD\_USAGE\_ATTRIBUTE\_DELETE ==

사용법:

cm ^attribute | ^att ^delete | ^rm <속성 사양>[ ...]

속성 사양 삭제할 속성입니다. 공백을 사용하여 속성을

구분하십시오.

(속성 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을

사용하십시오.)

== CMD\_HELP\_ATTRIBUTE\_DELETE ==

비고:

이 명령은 하나 이상의 속성을 제거합니다.

예:

cm ^attribute ^delete ^att:status

('상태' 속성을 삭제합니다.)

cm ^att ^rm 상태 ^att:integrated@reptest@server2:8084

('상태' 및 '통합' 속성을 삭제합니다.)

== CMD\_DESCRIPTION\_ATTRIBUTE\_UNSET ==

객체의 속성을 설정 해제합니다.

== CMD\_USAGE\_ATTRIBUTE\_UNSET ==

사용법:

cm ^attribute | ^att ^unset <속성 사양> <객체 사양>

속성 사양 속성 사양입니다. (속성 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을

사용하십시오.)

객체 사양 속성을 제거할 객체의

사양입니다. 속성을 설정할 수 있는 대상은 브랜치, 체인지 세트,

임시 저장 세트, 레이블, 항목 및 개정입니다.

(사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을 사용하십시오.)

== CMD\_HELP\_ATTRIBUTE\_UNSET ==

비고:

이 명령은 이전에 객체에 설정된 속성을 설정 해제합니다.

객체 자체를 삭제하지는 않습니다.

예:

cm ^attribute ^unset ^att:status ^br:/main/SCM105

('main/SCM105' 브랜치에서 '상태' 속성 실체화를 제거합니다.)

cm ^att ^unset ^att:integrated@reptest@localhost:8084 ^cs:25@reptest@localhost:8084

('reptest' 리포지토리의 25번 체인지 세트에서 '통합' 속성 실체화를

제거합니다.)

== CMD\_DESCRIPTION\_ATTRIBUTE\_RENAME ==

속성 이름을 변경합니다.

== CMD\_USAGE\_ATTRIBUTE\_RENAME ==

사용법:

cm ^attribute | ^att ^rename <속성 사양> <새 이름>

속성 사양 이름을 변경할 속성입니다. (속성 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을

사용하십시오.)

새 이름 새 속성 이름입니다.

== CMD\_HELP\_ATTRIBUTE\_RENAME ==

비고:

이 명령은 속성 이름을 변경합니다.

예:

cm ^attribute ^rename ^att:status state

('상태' 속성 이름을 'state'로 변경합니다.)

== CMD\_DESCRIPTION\_ATTRIBUTE\_EDIT ==

속성 코멘트를 편집합니다.

== CMD\_USAGE\_ATTRIBUTE\_EDIT ==

사용법:

cm ^attribute | ^att ^edit <속성 사양> <새 코멘트>

속성 사양 코멘트를 변경할 속성입니다. (속성 사양에 관한 자세한 내용을

확인하려면 'cm ^help ^objectspec'을 사용하십시오.)

새 코멘트 새 속성 코멘트입니다. 속성 값의 기본 목록을

지정할 수도 있습니다.

== CMD\_HELP\_ATTRIBUTE\_EDIT ==

비고:

이 명령은 속성 코멘트를 변경합니다.

속성 값의 기본 목록을 지정하려면 속성 코멘트에 다음과 같은 행을

포함시키면 됩니다.

'default: value\_one, "value two", value3, "Final value"'.

예:

cm ^attribute ^edit ^att:status "CI 파이프라인의 브랜치 상태입니다."

('상태' 속성 코멘트를 편집합니다.)

cm ^attribute ^edit ^att:status "브랜치 상태입니다. 기본값: 공개, 해결됨, 검토됨"

('상태' 속성 코멘트를 편집합니다. 값 목록도 지정할 수

있습니다. 따라서 객체에 '상태' 속성을 설정하면 "공개", "해결됨", "검토됨" 값 중 하나를

선택할 수 있습니다.)

== CMD\_DESCRIPTION\_REPLICATE ==

경고: 이 명령은 사용되지 않습니다.

'cm ^pull'('^replicate'와 동등한 명령) 및 'cm ^push'(

'^replicate --^push'와 동등한 명령)를 사용하십시오.

== CMD\_USAGE\_REPLICATE ==

== CMD\_HELP\_REPLICATE ==

== CMD\_DESCRIPTION\_PULL ==

다른 리포지토리에서 브랜치를 가져옵니다.

== CMD\_USAGE\_PULL ==

사용법:

cm ^pull <소스 브랜치 사양> <대상 리포지토리 사양>

[--^preview] [--^nodata] [변환 옵션]

[--^user=<사용자 이름> [--^password=<비밀번호>] | 인증 옵션]

(server-to-server 직접 복제 방식입니다. 리포지토리에서 브랜치를 가져옵니다.)

cm ^pull <대상 리포지토리 사양> --^package=<패키지 파일> [인증 옵션]

(패키지 기반 복제 방식으로, 대상 리포지토리의 패키지를 임포트합니다.)

cm ^pull ^hydrate <대상 브랜치 사양> [<소스 리포지토리 사양>]

[--^user=<사용자 이름> [--^password=<비밀번호>] | 인증 옵션]

('--^nodata'로 이전에 복제된 브랜치의 모든 체인지 세트에서 누락된 데이터를

표시할 수 있습니다. 데이터를 가져올 리포지토리가 지정되어 있지 않으면

Plastic에서는 "복제 소스"(복제된 브랜치의 원본)를 사용하려고

시도합니다).

cm ^pull ^hydrate <대상 체인지 세트 사양> [<소스 리포지토리 사양>]

[--^user=<사용자 이름> [--^password=<비밀번호>] | 인증 옵션]

('--^nodata'로 이전에 복제된 체인지 세트에서 누락된 데이터를

표시할 수 있습니다. 데이터를 가져올 리포지토리가 지정되어 있지 않으면 Plastic에서는

"복제 소스"를 사용하려고 시도합니다).

소스 브랜치 사양 원격 리포지토리에서 가져올 브랜치입니다.

(브랜치 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을 사용하십시오.)

대상 브랜치 사양 하이드레이트할 브랜치입니다.

(브랜치 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을 사용하십시오.)

대상 체인지 세트 사양 하이드레이트할 체인지 세트입니다.

(체인지 세트 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을

사용하십시오.)

대상 리포지토리 사양 대상 리포지토리입니다.

(리포지토리 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을

사용하십시오.)

--^package 패키지 기반 복제를 위해 임포트할 이전에 생성된 패키지

파일을 지정합니다.

서버 간 데이터 이동 시 직접 네트워크 연결이 없는 경우에

유용합니다.

패키지 파일을 생성하려면 'cm ^push'를 사용하십시오.

옵션:

--^preview 가져올 변경사항에 대한 정보를 제공하지만

실제로 변경사항이 실행되지는 않습니다. 이 옵션은 변경사항을

복제하기 전에 전송될 데이터를 확인할 때

유용합니다.

--^nodata 데이터를 복제하지 않으면서 브랜치 변경사항을

복제합니다. 패키지 복제 시에는 이 옵션을 사용할 수

없습니다.

변환 옵션 자세한 내용은 변환 옵션 섹션을 참조하십시오.

--^user, --^password 소스와 대상의 인증 모드가 다르며 대상에 대해

인증할 프로필이 없는 경우에 사용할

자격 증명입니다.

인증 옵션 자세한 내용은 인증 옵션 섹션을

참조하십시오.

변환 옵션:

--^trmode=(^copy|^name|^table --^trtable=<변환 테이블 파일>)

소스와 대상 리포지토리는 서로 다른 인증 모드를 사용할 수

있습니다. '--^trmode' 옵션은 소스에서 대상으로 사용자 이름을 변환할 방식을

지정합니다. '--^trmode'는 다음 값 중 하나여야

합니다.

^copy (기본값). 사용자 식별자가 그대로 복사됨을 의미합니다.

^name 이름을 기준으로 사용자 식별자를 일치시킵니다.

^table '--^trtable' 옵션에서 지정된 변환 테이블을 사용합니다

(아래 참조).

--^trtable=<변환 테이블 파일>

변환 모드가 'table'인 경우 변환 테이블은 <이전 이름;새 이름>(행별로 1개)

형식의 행이 포함된 파일이 됩니다. 브랜치가

대상 리포지토리에 작성되면 소스 리포지토리에서 "이전 이름"으로 식별된

사용자에 의해 생성된 객체가 대상에서는 사용자 이름이 "새 이름"인 객체로

설정됩니다.

인증 옵션:

다음 두 가지 모드 중 하나를 사용하여 인증 데이터를 지정할 수 있습니다.

1) 인증 매개 변수 사용: --^authmode=<모드> --^authdata=<데이터>

--^authmode=(^NameWorkingMode|^LDAPWorkingMode|^ADWorkingMode|^UPWorkingMode)

예:

(^LDAPWorkingMode) --^authdata=::0:dave:fPBea2rPsQaagEW3pKNveA

(^UPWorkingMode) --^authdata=dave:fPBea2rPsQaagEW3pKNveA==

'--^authdata' 행은 client.conf 파일과 profiles.conf 파일에 있는

<^SecurityConfig> 항목의 내용입니다. profiles.conf

파일은 Plastic GUI의 기본 설정 아래에 있는 복제 프로필 탭에서 쉽게

생성할 수 있습니다.

^UPWorkingMode를 사용하는 경우 다음과 같이 간단히 지정할 수 있습니다.

--^authmode=^UPWorkingMode --^user=<사용자> --^password=<비밀번호>

2) 연결할 각 서버에 서로 다른 파일이 있는 경우 해당 서버의 자격 증명이

포함된 인증 파일입니다.

--^authfile=<인증 파일>

파일에는 다음 두 행이 포함됩니다.

행 1) '--^authmode'에 명시된 모드

행 2) '--^authdata'에 명시된 모드 인증 데이터

== CMD\_HELP\_PULL ==

비고:

'^pull' 명령을 통해 소스 리포지토리와 대상 리포지토리 간에 브랜치를

복제(체인지 세트와 함께)할 수 있습니다.

리포지토리가 위치한 서버는 각각 다를 수 있습니다.

'^push' 및 '^pull'의 두 가지 복제 작업이 있습니다.

'^pull' 작업은 복제 작업에 소스 리포지토리에서 대상 리포지토리로 저장할

데이터가 필요하다는 것을 의미합니다.

클라이언트가 해당 호스트에서 대상 리포지토리로 연결되고 소스 리포지토리에 대한

연결을 설정하여 대상 데이터를

가져옵니다. 가져오는 동안에는 클라이언트가 소스로 연결될 대상 서버가

됩니다.

개발자가 데이터를 자신의 로컬 서버에서 주 서버로 내보내는 일반적인

분산형 시나리오에서도 개발자가 주 서버에서 최신 리포지토리 업데이트를

가져오려 할 수 있습니다.

복제된 리포지토리 2개의 동일한 브랜치에서 동시에 변경된 사항이 있는 경우

복제를 통해 이러한 상황을 해결할 수 있습니다.

- 내보내기: 전송 대상으로 지정한 리포지토리가 아니라 최신 변경사항이 있는

리포지토리로 데이터를 내보내려는 경우 시스템에서 최신 변경사항을

가져올지 여부를 물어본 다음 병합 작업을 해결하고 다시 내보내려고 시도합니다.

- 가져오기: 원격 브랜치에서 체인지 세트를 가져올 때마다 체인지 세트는

부모 체인지 세트와 바르게 연결됩니다. 가져온 체인지 세트가

브랜치에 있는 최근 체인지 세트의 자식이 아닌 경우 여러 헤드가 존재하는 시나리오가

표시됩니다. 브랜치는 둘 이상의 '헤드' 또는 브랜치의 최근 체인지 세트를

가집니다. 먼저 두 '헤드'를 병합해야 다시 내보낼 수

있습니다.

가져오기는 두 가지 모드에서 이루어질 수 있습니다.

1) 서버 간 직접 통신: 대상 서버에서 소스 서버 데이터를 가져와

지정된 브랜치에 맞게 자동으로 데이터를

동기화합니다.

2) 이전에 내보내기와 '--^package' 옵션으로 생성된 패키지를 임포트합니다.

모드 1) 명령을 실행하는 사용자가 원격 서버에 의한 인증을 받아야 합니다.

이를 위해 client.conf 파일의 기본 인증을 사용하거나 '--^authmode' 및 '--^authdata' 한정어를

지정할 수 있으며, 아니면 '--^authmode' 및

'--^user'/'--^password'를 사용할 수 있습니다(인증 모드가

^UPWorkingMode인 경우.)

모드 2) 이전에 push 명령으로 생성된 패키지 파일을 사용해야

합니다.

pull 복제는 간접적인 방식으로 이루어진다는 점을 염두에 두십시오. 해당 명령을 실행하면

소스에 연결할 대상 리포지토리를 물어본 다음 선택한 브랜치를

가져옵니다.

하지만 push 명령을 사용하여 직접 이 작업을 수행할 수도 있습니다.

이렇게 하면 명령을 통해 소스에서 대상으로 선택한 브랜치를

복제할 수 있습니다.

예:

cm ^pull ^br:/main@project1@remoteserver:8084 projectx@myserver:8084

('remoteserver'에서 'myserver'로 'main' 브랜치를 가져옵니다. 이러한 경우

두 서버가 동일한 인증 모드로 구성됩니다.)

cm ^pull ^br:/main@project1@remoteserver:8084 projectx@myserver:8084 \

--^authmode=^LDAPWorkingMode --^authdata=::0:dave:fPBea2rPsQaagEW3pKNveA

(이전과 동일한 브랜치를 가져오지만 이제 원격 서버가 구성되어

Active Directory로 사용자를 인증할 수 있습니다. 예를 들어 Linux 시스템에서

Active Directory 통합 모드를 사용하도록 구성된 Windows 서버로

연결하는 경우입니다. 자신의 Active Directory 사용자 및 암호화된 비밀번호를

지정하고 LDAP로 이를 서버에 전달합니다.)

cm ^pull ^br:/main@project1@remoteserver:8084 projectx@myserver:8084 \

--^authmode=^UPWorkingMode --^user=dave --^password=mysecret

(동일한 브랜치를 가져오지만 이제 사용자가 원격 서버에서 인증되므로

Plastic SCM에 포함된 사용자/비밀번호 데이터베이스를

활용할 수 있습니다.)

cm ^pull ^br:/main@project1@remoteserver:8084 projectx@myserver:8084 --^nodata

('remoteserver'에서 'myserver'로 데이터를 제외하고 'main' 브랜치를 복제합니다.)

cm ^pull ^hydrate ^br:/main@projectx@myserver:8084 projectx@remoteserver:8084

(원격 서버에서 데이터를 가져오는 'main' 브랜치의 모든 체인지 세트를

하이드레이트합니다.)

cm ^pull ^hydrate ^cs:122169@projectx@myserver:8084 projectx@remoteserver:8084

(원격 서버에서 데이터를 가져오는 'myserver'의 122169번 체인지 세트를

하이드레이트합니다.)

== CMD\_DESCRIPTION\_PUSH ==

다른 리포지토리로 브랜치를 내보냅니다.

== CMD\_USAGE\_PUSH ==

사용법:

cm ^push <소스 브랜치 사양> <대상 리포지토리 사양>

[--^preview] [변환 옵션]

[--^user=<사용자 이름> [--^password=<비밀번호>] | 인증 옵션]

(server-to-server 직접 복제 방식입니다. 리포지토리에서 브랜치를 내보냅니다.)

cm ^push <소스 브랜치 사양> --^package=<패키지 파일> [인증 옵션]

(패키지 기반 복제 방식으로, 소스 서버에서 선택한 브랜치를 포함한 복제 패키지를

생성합니다.)

소스 브랜치 사양 원격 리포지토리로 내보낼 브랜치입니다.

(브랜치 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을 사용하십시오.)

대상 리포지토리 사양 대상 리포지토리입니다.

(리포지토리 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을

사용하십시오.)

--^package 패키지 기반 복제를 위해 복제 패키지를 익스포트할

경로를 지정합니다.

서버 간 데이터 이동 시 직접 네트워크 연결이 없는 경우에

유용합니다.

옵션:

--^preview 내보낼 변경사항에 대한 정보를 제공하지만

실제로 변경사항이 실행되지는 않습니다. 이 옵션은 변경사항을

복제하기 전에 전송될 데이터를 확인할 때

유용합니다.

변환 옵션 자세한 내용은 변환 옵션 섹션을 참조하십시오.

--^user, --^password 소스와 대상의 인증 모드가 다르며 대상에 대해

인증할 프로필이 없는 경우에 사용할

자격 증명입니다.

인증 옵션 자세한 내용은 인증 옵션 섹션을

참조하십시오.

변환 옵션:

--^trmode=(^copy|^name|^table --^trtable=<변환 테이블 파일>)

소스와 대상 리포지토리는 서로 다른 인증 모드를 사용할 수

있습니다. '--^trmode' 옵션은 소스에서 대상으로 사용자 이름을 변환할 방식을

지정합니다. '--^trmode'는 다음 값 중 하나여야

합니다.

^copy (기본값). 사용자 식별자가 그대로 복사됨을 의미합니다.

^name 이름을 기준으로 사용자 식별자를 일치시킵니다.

^table '--^trtable' 옵션에서 지정된 변환 테이블을 사용합니다

(아래 참조).

--^trtable=<변환 테이블 파일>

변환 모드가 'table'인 경우 변환 테이블은 <이전 이름;새 이름>(행별로 1개)

형식의 행이 포함된 파일이 됩니다. 브랜치가

대상 리포지토리에 작성되면 소스 리포지토리에서 "이전 이름"으로 식별된

사용자에 의해 생성된 객체가 대상에서는 사용자 이름이 "새 이름"인 객체로

설정됩니다.

인증 옵션:

다음 두 가지 모드 중 하나를 사용하여 인증 데이터를 지정할 수 있습니다.

1) 인증 매개 변수 사용: --^authmode=<모드> --^authdata=<데이터>

--^authmode=(^NameWorkingMode|^LDAPWorkingMode|^ADWorkingMode|^UPWorkingMode)

예:

(^LDAPWorkingMode) --^authdata=::0:dave:fPBea2rPsQaagEW3pKNveA

(^UPWorkingMode) --^authdata=dave:fPBea2rPsQaagEW3pKNveA==

'--^authdata' 행은 client.conf 파일과 profiles.conf 파일에 있는

<^SecurityConfig> 항목의 내용입니다. profiles.conf

파일은 Plastic GUI의 기본 설정 아래에 있는 복제 프로필 탭에서 쉽게

생성할 수 있습니다.

^UPWorkingMode를 사용하는 경우 다음과 같이 간단히 지정할 수 있습니다.

--^authmode=^UPWorkingMode --^user=<사용자> --^password=<비밀번호>

2) 연결할 각 서버에 서로 다른 파일이 있는 경우 해당 서버의 자격 증명이

포함된 인증 파일입니다.

--^authfile=<인증 파일>

파일에는 다음 두 행이 포함됩니다.

행 1) '--^authmode'에 명시된 모드

행 2) '--^authdata'에 명시된 모드 인증 데이터

== CMD\_HELP\_PUSH ==

비고:

'^push' 명령을 통해 소스 리포지토리와 대상 리포지토리 간에 브랜치를

복제(체인지 세트와 함께)할 수 있습니다.

리포지토리가 위치한 서버는 각각 다를 수 있습니다.

'^push' 및 '^pull'의 두 가지 복제 작업이 있습니다.

'^push' 작업은 복제 작업에 소스 리포지토리에서 대상 리포지토리로

데이터를 전송한다는 것을 의미합니다. 이 경우

클라이언트가 소스 리포지토리로 연결되어 복제할 데이터를 가져온 다음

대상 리포지토리로 전송합니다. 전자(소스)는

대상에 연결되어 있어야 하지만 후자(대상)는 자체적으로 소스에

연결되지 않습니다.

일반적인 분산형 시나리오에서 개발자는 자신의 로컬 서버에서 주 서버로

데이터를 내보냅니다. 또한 개발자가 최신 리포지토리 업데이트를 주 서버에서

가져오려 할 수 있습니다.

복제된 리포지토리 2개의 동일한 브랜치에서 동시에 변경된 사항이 있는 경우

복제를 통해 이러한 상황을 해결할 수 있습니다.

- 내보내기: 전송 대상으로 지정한 리포지토리가 아니라 최신 변경사항이 있는

리포지토리로 데이터를 내보내려는 경우 시스템에서 최신 변경사항을

가져올지 여부를 물어본 다음 병합 작업을 해결하고 다시 내보내려고 시도합니다.

- 가져오기: 원격 브랜치에서 체인지 세트를 가져올 때마다 체인지 세트는

부모 체인지 세트와 바르게 연결됩니다. 가져온 체인지 세트가

브랜치에 있는 최근 체인지 세트의 자식이 아닌 경우 여러 헤드가 존재하는 시나리오가

표시됩니다. 브랜치는 둘 이상의 '헤드' 또는 브랜치의 최근 체인지 세트를

가집니다. 먼저 두 '헤드'를 병합해야 다시 내보낼 수

있습니다.

내보내기는 두 가지 모드에서 이루어질 수 있습니다.

1) 서버 간 직접 통신: 원래 서버에서 대상 서버로 데이터를 전송하여

지정된 브랜치에 맞게 자동으로 데이터를

동기화합니다.

2) 패키지 익스포트 모드: 클라이언트는 소스에만 연결하고

지정된 브랜치의 데이터와 메타데이터를 모두 가져오는 복제 패키지를

생성합니다. 이때 '--^package' 한정어가 사용됩니다.

두 가지 모드에서는 명령을 실행하는 사용자가 원격 서버에 의한 인증을 받아야 합니다.

이를 위해 client.conf 파일의 기본 인증을 사용하거나

'--^authmode' 및 '--^authdata' 한정어를 지정할 수 있습니다.

^push 복제는 직접적인 방식으로 이루어집니다. 해당 명령이 실행되면

소스에 연결할 대상 리포지토리를 묻지 않고 소스에서 대상으로 선택한 브랜치를

복제한 다음 선택한 브랜치를 가져옵니다

(pull과 같은 방식).

예:

cm ^push ^br:/main@project1@myserver:8084 projectx@remoteserver:8084

('myserver'에서 'remoteserver'로 'main' 브랜치를 복제합니다. 이러한 경우

두 서버가 동일한 인증 모드로 구성됩니다.)

cm ^push ^br:/main@project1@myserver:8084 projectx@remoteserver:8084 \

--^authmode=^LDAPWorkingMode --^authdata=::0:dave:fPBea2rPsQaagEW3pKNveA

(이전과 동일한 브랜치를 복제하지만 이제 원격 서버가 구성되어

Active Directory로 사용자를 인증할 수 있습니다. 예를 들어 Linux 시스템에서

Active Directory 통합 모드를 사용하도록 구성된 Windows 서버로

연결하는 경우입니다. 자신의 Active Directory 사용자 및 암호화된 비밀번호를

지정하고 LDAP로 이를 서버에 전달합니다.)

cm ^push ^br:/main@project1@myserver:8084 projectx@remoteserver:8084 \

--^authmode=^UPWorkingMode --^user=dave --^password=mysecret

(동일한 브랜치를 복제하지만 이제 사용자가 원격 서버에서 인증되므로

Plastic SCM에 포함된 사용자/비밀번호 데이터베이스를

활용할 수 있습니다.)

== CMD\_DESCRIPTION\_CLONE ==

원격 리포지토리의 클론을 생성합니다.

== CMD\_USAGE\_CLONE ==

사용법:

cm ^clone <소스 리포지토리 사양> [<대상 리포지토리 사양> | <대상 리포지토리 서버 사양>]

[--^user=<사용자 이름> [--^password=<비밀번호>] | 인증 옵션]

[변환 옵션]

(repository-to-repository 직접 클론 방식입니다.)

cm ^clone <소스 리포지토리 사양> --^package=<패키지 파일>

[--^user=<사용자 이름> [--^password=<비밀번호>] | 인증 옵션]

(중간 패키지로 클론을 생성합니다. 이 패키지는 나중에 pull을 사용하여

대상 리포지토리로 임포트할 수 있습니다.)

소스 리포지토리 사양 클론 작업의 소스 리포지토리입니다.

(리포지토리 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을

사용하십시오.)

대상 리포지토리 사양 클론 작업의 대상 리포지토리입니다. 대상

리포지토리가 있는 경우에는 해당 리포지토리가 비어 있어야 합니다. 없는 경우에는

대상 리포지토리가 생성됩니다.

지정되어 있지 않은 경우에는 해당 명령이 사용자의

기본 리포지토리 서버를 사용합니다.

(리포지토리 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을

사용하십시오.)

대상 리포지토리 서버 사양 클론 작업의 대상 리포지토리 서버입니다.

대상 리포지토리 서버에

<소스 리포지토리 사양>과 이름이 동일한 리포지토리가 있는 경우에는

해당 리포지토리가 비어 있어야 합니다. 없는 경우에는 대상 리포지토리가 생성됩니다.

지정되어 있지 않은 경우에는 해당 명령이 사용자의

기본 리포지토리 서버를 사용합니다.

(서버 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을

사용하십시오.)

옵션:

--^user, --^password 소스와 대상의 인증 모드가 다르며 대상에 대해

인증할 프로필이 없는 경우에 사용할

자격 증명입니다.

--^package 지정된 리포지토리를 리포지토리가 아니라

패키지 파일로 익스포트합니다.

서버 간 데이터 이동 시 직접 네트워크 연결이 없는 경우에

유용합니다.

결과 패키지는 pull 명령을 사용하여 임포트해야

합니다.

변환 옵션 자세한 내용은 변환 옵션 섹션을 참조하십시오.

인증 옵션 자세한 내용은 인증 옵션 섹션을

참조하십시오.

변환 옵션:

--^trmode=(^copy|^name|^table --^trtable=<변환 테이블 파일>)

소스와 대상 리포지토리는 서로 다른 인증 모드를 사용할 수

있습니다. '--^trmode' 옵션은 소스에서 대상으로 사용자 이름을 변환할 방식을

지정합니다. '--^trmode'는 다음 값 중 하나여야

합니다.

^copy (기본값). 사용자 식별자가 그대로 복사됨을 의미합니다.

^name 이름을 기준으로 사용자 식별자를 일치시킵니다.

^table '--^trtable' 옵션에서 지정된 변환 테이블을 사용합니다

(아래 참조).

--^trtable=<변환 테이블 파일>

변환 모드가 'table'인 경우 변환 테이블은 <이전 이름;새 이름>(행별로 1개)

형식의 행이 포함된 파일이 됩니다. 브랜치가

대상 리포지토리에 작성되면 소스 리포지토리에서 "이전 이름"으로 식별된

사용자에 의해 생성된 객체가 대상에서는 사용자 이름이 "새 이름"인 객체로

설정됩니다.

인증 옵션:

다음 두 가지 모드 중 하나를 사용하여 인증 데이터를 지정할 수 있습니다.

1) 인증 매개 변수 사용: --^authmode=<모드> --^authdata=<데이터>

--^authmode=(^NameWorkingMode|^LDAPWorkingMode|^ADWorkingMode|^UPWorkingMode)

예:

(^LDAPWorkingMode) --^authdata=::0:dave:fPBea2rPsQaagEW3pKNveA

(^UPWorkingMode) --^authdata=dave:fPBea2rPsQaagEW3pKNveA==

'--^authdata' 행은 client.conf 파일과 profiles.conf 파일에 있는

<^SecurityConfig> 항목의 내용입니다. profiles.conf

파일은 Plastic GUI의 기본 설정 아래에 있는 복제 프로필 탭에서 쉽게

생성할 수 있습니다.

^UPWorkingMode를 사용하는 경우 다음과 같이 간단히 지정할 수 있습니다.

--^authmode=^UPWorkingMode --^user=<사용자> --^password=<비밀번호>

2) 연결할 각 서버에 서로 다른 파일이 있는 경우 해당 서버의 자격 증명이

포함된 인증 파일입니다.

--^authfile=<인증 파일>

파일에는 다음 두 행이 포함됩니다.

행 1) '--^authmode'에 명시된 모드

행 2) '--^authdata'에 명시된 모드 인증 데이터

== CMD\_HELP\_CLONE ==

비고:

clone 명령을 통해 소스 리포지토리에서 대상 리포지토리로

브랜치를 복제(체인지 세트, 레이블, 속성, 검토 등과 함께)할 수

있습니다. 리포지토리가 위치한 서버는 각각 다를 수 있습니다.

대상 리포지토리를 사전에 생성할 수 있지만 대상 리포지토리에 이전 데이터가

있는 경우에는 클론 작업이 실패합니다.

클론 작업은 리포지토리의 하위 모듈이나 Xlink 아래에 있는 리포지토리의 클론을

생성하지 않습니다.

예:

cm ^clone awesomeProject@tardis@cloud

('tardis@cloud' 조직에서 'awesomeProject' 리포지토리의 클론을

동일한 이름의 로컬 리포지토리에 생성합니다.)

cm ^clone repo@server.home:9095 repo-local

('server.home:9095'에서 'repo'의 클론을 사용자의 기본 리포지토리 서버의

'repo-local'에 생성합니다.)

cm ^clone project@192.168.111.130:8084 ^repserver:192.168.111.200:9095

('192.168.111.130:8084'에서 'project' 리포지토리의 클론을

'project@192.168.111.200:9095'에 생성합니다.)

cm ^clone project@ldapserver:8084 --authfile=credentials.txt \

--^trmode=table --^trtable=table.txt

(원격 리포지토리에 대한 인증 파일을 사용하고 지정된 변환 테이블에 따라

사용자를 변환하여 'ldapserver:8084'에서 'project' 리포지토리의

클론을 생성합니다.)

cm ^clone project@server.home:9095 --^package=project.plasticpkg

cm ^repository ^create project@mordor.home:8084

cm ^pull --^package=project.plasticpkg project@mordor.home:8084

('server.home:9095'에서 'project' 리포지토리의 클론을 'project.plasticpkg'

패키지로 생성합니다. 이 패키지는 나중에 pull을 통해 'mordor.home:8084'에 있는

'project' 리포지토리로 가져옵니다.)

== CMD\_DESCRIPTION\_REVERT ==

항목을 이전 개정으로 되돌립니다.

== CMD\_USAGE\_REVERT ==

사용법:

cm ^revert <개정 사양>

개정 사양 개정이 포함된 체인지 세트의 사양입니다. 개정 내용은

워크스페이스에 로드됩니다.

(개정 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을

사용하십시오.)

== CMD\_HELP\_REVERT ==

비고:

항목이 체크인된 상태여야 합니다.

예:

cm ^revert dir#^cs:0

cm ^revert C:\mywks\dir\file1.txt#23456

== CMD\_DESCRIPTION\_REVISION\_HISTORY ==

파일 또는 디렉터리 내역을 표시합니다.

== CMD\_USAGE\_REVISION\_HISTORY ==

사용법:

cm ^history | ^hist <항목 경로>[ ...] [--^long | --^format=<문자열 형식>]

[--^symlink] [--^xml[=<출력 파일>]] [--^encoding=<이름>]

항목 경로 항목 경로입니다. 공백을 사용하여 경로를 구분하십시오. 공백이

포함된 경로를 지정하려면 큰따옴표(" ")를 사용하십시오.

경로는 서버 경로 개정일 수도 있습니다.

(사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을 사용하십시오.)

옵션:

--^long 추가 정보를 표시합니다.

--^format 특정 형식으로 출력 메시지를 가져옵니다. 자세한 내용은

비고를 확인하십시오.

--^symlink 대상이 아닌 기호 링크에 내역 작업을

적용합니다.

--^xml XML 형식의 출력을 표준 출력으로 출력합니다.

출력 파일을 지정할 수 있습니다.

--^encoding '--^xml' 옵션과 함께 사용되며 XML 출력에 사용할

인코딩, 즉 UTF-8을 지정합니다.

지원되는 인코딩 및 형식에 대한 표를 확인하려면

http://msdn.microsoft.com/en-us/library/system.text.encoding.aspx에서

MSDN 문서를 확인하십시오.

페이지 마지막 "Name"열에 있습니다.

== CMD\_HELP\_REVISION\_HISTORY ==

비고:

이 명령은 특정 항목의 개정 목록과 각 개정의 레이블, 브랜치 및 코멘트 정보를

보여줍니다.

이 명령은 출력을 표시할 형식 문자열을 허용합니다.

이 명령의 출력 매개 변수는 다음과 같습니다.

{0} | {^date} 날짜.

{1} | {^changesetid} 체인지 세트 번호.

{2} | {^branch} 브랜치.

{4} | {^comment} 코멘트.

{5} | {^owner} 소유자.

{6} | {^id} 개정 ID.

{7} | {^repository} 리포지토리.

{8} | {^server} 서버.

{9} | {^repspec} 리포지토리 사양.

{^tab} 탭 공백을 삽입합니다.

{^newline} 새 행을 삽입합니다.

예:

cm ^history file1.txt "file 2.txt"

cm ^hist c:\workspace --^long

(모든 정보를 표시합니다.)

cm ^history 링크 --^symlink

(대상이 아닌 '링크' 파일에 내역 작업을 적용합니다.

UNIX 환경에서 사용할 수 있습니다.)

cm ^history ^serverpath:/src/foo/bar.c#^br:/main/task001@myserver

(특정 브랜치의 서버 경로에서 개정 내역을 가져옵니다.)

== CMD\_DESCRIPTION\_REVISION\_TREE ==

항목의 개정 트리를 표시합니다.

== CMD\_USAGE\_REVISION\_TREE ==

사용법:

cm ^tree <경로> [--^symlink]

경로 항목 경로입니다.

옵션:

--^symlink 대상이 아닌 링크 파일에 작업을 적용합니다.

== CMD\_HELP\_REVISION\_TREE ==

예:

cm ^tree fichero1.txt

cm ^tree c:\workspace

cm ^tree 링크 --^symlink

(대상이 아닌 링크 파일에 내역 작업을 적용합니다.

UNIX 환경에서 사용할 수 있습니다.)

== CMD\_DESCRIPTION\_RM ==

사용자의 파일 및 디렉터리 삭제를 허용합니다.

== CMD\_USAGE\_RM ==

사용법:

cm ^remove | ^rm <명령> [옵션]

명령:

^controlled(선택사항)

^private

각 명령에 대한 자세한 정보는 다음을 실행해 확인하십시오.

cm ^remove <명령> --^usage

cm ^remove <명령> --^help

== CMD\_HELP\_RM ==

예:

cm ^remove \path\controlled\_file.txt

cm ^remove ^private \path\private\_file.txt

== CMD\_DESCRIPTION\_RM\_CONTROLLED ==

버전 제어에서 파일 또는 디렉터리를 삭제합니다.

== CMD\_USAGE\_RM\_CONTROLLED ==

사용법:

cm ^remove | ^rm <항목 경로>[ ...] [--^format=<문자열 형식>]

[--^errorformat=<문자열 형식>] [--^nodisk]

항목 경로 제거할 항목 경로입니다. 공백이 포함된 경로를 지정하려면

큰따옴표(" ")를 사용하십시오. 공백을 사용하여 경로를

구분하십시오.

옵션:

--^format 특정 형식으로 출력 진행도 메시지를

가져옵니다. 자세한 내용은 예를 참조하시기 바랍니다.

--^errorformat 오류 메시지가 있는 경우 특정 형식으로

가져옵니다. 자세한 내용은 예를 참조하시기 바랍니다.

--^nodisk 버전 제어에서는 제거하지만 항목이 디스크에서

유지됩니다.

== CMD\_HELP\_RM\_CONTROLLED ==

비고:

디스크에서 항목이 삭제됩니다. 삭제된 항목이 소스 코드 제어의 부모 디렉터리에서

삭제됩니다.

요구사항:

- 항목이 소스 코드 제어 상태여야 합니다.

표준 입력에서 입력 읽기:

'^remove' 명령은 표준 입력에서 경로를 읽을 수 있습니다. 그러려면 단일 대시 "-"를

전달합니다.

예: cm ^remove -

경로 읽기는 빈 행이 입력될 때까지 계속됩니다.

따라서 파이프를 사용해 제거할 파일을 지정할 수 있습니다.

예:

dir /S /B \*.c | cm ^remove -

(Windows에서는 워크스페이스에 있는 모든 .c 파일을 제거합니다.)

예:

cm ^remove src

('src'를 제거합니다. src가 디렉터리인 경우에는

'cm ^remove -^R src'와 동일합니다.)

cm ^remove c:\workspace\file.txt --^format="{0} - REMOVED" \

--^errorformat="{0} - ERROR REMOVING"

(작업 성공 시 버전 제어에서 'file.txt'를 제거하고

"c:\workspace\file.txt - ^REMOVED"를 작성합니다. 그렇지 않은 경우

"c:\workspace\file.txt - ^ERROR ^REMOVING"을 작성합니다.)

cm ^remove c:\workspace\file.txt --^nodisk

(버전 제어에서 'file.txt'를 제거하지만 디스크에서는 유지됩니다.)

== CMD\_DESCRIPTION\_RM\_PRIVATE ==

비공개 파일 또는 디렉터리를 삭제합니다.

경고: 명령을 사용하여 삭제된 파일은 영구히 삭제되며

복원할 수 없습니다. '--^dry-run' 옵션을 사용하여

해당 명령의 영향을 받는 파일을 확인하는 것이 좋습니다.

== CMD\_USAGE\_RM\_PRIVATE ==

사용법:

cm ^remove | ^rm ^private <경로>[ ...] [-^R | -^r | --^recursive] [--^ignored]

[--^verbose] [--^dry-run]

경로 제거할 파일 또는 디렉터리의 경로입니다. 공백이 포함된

경로를 지정하려면 큰따옴표(" ")를 사용하십시오. 공백을

사용하여 경로를 구분하십시오.

옵션:

--^r 제어 대상 디렉터리 내에 있는 비공개 파일을 재귀적으로

삭제합니다.

--^ignored 무시되고 숨겨진 파일 및 디렉터리도 삭제합니다.

--^verbose 영향을 받는 경로를 모두 출력합니다.

--^dry-run 디스크 내용을 변경하지 않고 명령을 실행합니다.

== CMD\_HELP\_RM\_PRIVATE ==

비고:

경로가 비공개 파일 또는 디렉터리인 경우 디스크에서 삭제됩니다.

경로가 제어 대상 파일인 경우 명령이 실패합니다.

경로가 제어 대상 디렉터리인 경우 '-^r' 옵션을 지정하지 않으면

명령이 실패합니다. 해당 옵션을 사용하면 지정된 디렉터리 아래의 모든 비공개 파일과

디렉터리가 삭제됩니다.

예:

cm ^remove ^private 비공개 디렉터리

('비공개 디렉터리'를 삭제합니다.)

cm ^remove ^private c:\workspace\controlled\_directory

('controlled\_directory'가 비공개가 아니므로 실패합니다.)

cm ^remove ^private -^r c:\workspace\controlled\_directory

('controlled\_directory' 아래의 모든 비공개 파일과 디렉터리를 삭제합니다.)

cm ^rm ^private --^dry-run --^verbose c:\workspace\controlled\_directory -^r

('controlled\_directory' 아래에 있는 비공개 파일 삭제의 영향을 받는 경로를

모두 표시합니다. 실제로 삭제가 이루어지지는 않습니다.)

cm ^rm ^private --^verbose c:\workspace\controlled\_directory -^r

('controlled\_directory' 아래에 있는 비공개 파일 삭제의 영향을 받는 경로를

모두 표시하고 삭제를 수행합니다.)

== CMD\_DESCRIPTION\_TRIGGER\_DELETE ==

트리거를 삭제합니다.

== CMD\_USAGE\_TRIGGER\_DELETE ==

사용법:

cm ^trigger | ^tr ^delete | ^rm <하위 유형 유형> <위치 번호>

[--^server=<리포지토리 서버 사양>]

하위 유형 유형 트리거 실행 및 트리거 작업.

트리거 유형을 확인하려면 'cm ^showtriggertypes'를

입력하십시오.

위치 번호 트리거가 생성될 때 트리거에 할당되는 위치입니다.

옵션:

--^server 지정된 서버에서 트리거를 삭제합니다.

서버가 지정되지 않은 경우, 클라이언트에 구성된 서버를

대상으로 명령을 실행합니다.

== CMD\_HELP\_TRIGGER\_DELETE ==

예:

cm ^trigger ^delete ^after-setselector 4

cm ^tr ^rm ^after-setselector 4

== CMD\_DESCRIPTION\_ATTRIBUTE\_SET ==

특정 객체에 속성을 설정합니다.

== CMD\_USAGE\_ATTRIBUTE\_SET ==

사용법:

cm ^attribute | ^att ^set <속성 사양> <객체 사양> <속성 값>

속성 사양 속성 사양입니다. (속성 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을

사용하십시오.)

객체 사양 속성을 설정할 대상이 되는 객체의 사양입니다.

속성을 설정할 수 있는 대상은 브랜치, 체인지 세트,

임시 저장 세트, 레이블, 항목 및 개정입니다.

(사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을 사용하십시오.)

속성 값 객체에 설정할 속성 값입니다.

== CMD\_HELP\_ATTRIBUTE\_SET ==

비고:

이 객체에 대한 추가 정보를 저장하도록 속성을 객체에

설정할 수 있습니다.

속성을 설정할 수 있는 대상은 브랜치, 체인지 세트,

임시 저장 세트, 레이블, 항목 및 개정입니다.

예:

cm ^attribute ^set ^att:status ^br:/main/SCM105 공개

('공개' 값을 사용하여 '상태' 속성을 'SCM105' 브랜치에 설정합니다.)

cm ^att ^set ^att:integrated@reptest@server2:8084 ^lb:LB008@reptest@server2:8084 예

('예' 값을 사용하여 'reptest' 리포지토리의 'LB008' 레이블에

'통합' 속성을 설정합니다.)

== CMD\_DESCRIPTION\_SETOWNER ==

객체 소유자를 설정합니다.

== CMD\_USAGE\_SETOWNER ==

사용법:

cm ^setowner | ^sto --^user=<사용자 이름> | --^group=<그룹> <객체 사양>

--^user 사용자 이름입니다. 객체의 새로운 소유자입니다.

--^group 그룹 이름입니다. 객체의 새로운 소유자입니다.

객체 사양 새 소유자를 설정할 대상이 되는 객체의 사양입니다.

소유자는 다음 객체에 설정할 수 있습니다.

리포지토리 서버, 리포지토리, 브랜치, 체인지 세트,

레이블, 항목, 개정 및 속성.

(사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을 사용하십시오.)

== CMD\_HELP\_SETOWNER ==

비고:

객체 소유자는 사용자나 그룹일 수 있습니다.

소유자를 설정할 수 있는 객체는 리포지토리 서버, 리포지토리, 브랜치, 체인지 세트,

레이블, 항목, 개정 및 속성입니다.

예:

cm ^setowner --^user=john ^repserver:localhost:8084

('john'을 리포지토리 서버 소유자로 설정합니다.)

cm ^sto --^group=development ^rep:mainRep@PlasticServer:8084

('development' 그룹을 'mainRep' 리포지토리의 소유자로 설정합니다.)

== CMD\_DESCRIPTION\_SETSELECTOR ==

워크스페이스에 선택기를 설정합니다.

== CMD\_USAGE\_SETSELECTOR ==

이 명령은 사용되지 않습니다. 단, 이전 버전과의 호환성을 위해 남아 있습니다.

Plastic SCM 4.0에서는 대부분의 선택기가 사용되지 않습니다. 선택기는

작업 브랜치 또는 체인지 세트를 지정할 용도로 남아 있지만 경로를 필터링하는 기존 규칙은

더 이상 지원되지 않습니다.

사용법:

cm ^setselector | ^sts [--^file=<선택기 파일>] [--^ignorechanges]

[--^forcedetailedprogress] [<워크스페이스 경로> | <워크스페이스 사양>]

옵션:

--^file 선택기를 로드할 파일입니다.

--^ignorechanges 워크스페이스 업데이트 시 대기 중인 변경사항이 감지되는 경우

이와 관련하여 표시된 경고 메시지를

무시합니다.

--^forcedetailedprogress 표준 출력이 리디렉션되더라도 진행도를 강제로

표시합니다.

워크스페이스 경로 선택기를 설정할 워크스페이스의 경로입니다.

워크스페이스 사양 워크스페이스 사양입니다. (워크스페이스 사양에 관한 자세한 내용을

확인하려면 'cm ^help ^objectspec'을 사용하십시오.)

== CMD\_HELP\_SETSELECTOR ==

비고:

이 명령은 워크스페이스의 선택기를 설정합니다.

워크스페이스는 리포지토리에서 개정을 로드하기 위해 정보를 필요로 합니다.

이 정보를 얻기 위해 Plastic SCM에서는 선택기를 사용합니다.

선택기를 사용하면 특정 브랜치에서 개정, 레이블 또는 체인지 세트를

로드할 수 있습니다.

선택기를 로드할 파일이 지정되어 있지 않은 경우, 기본 운영

체제 편집기가 실행됩니다.

선택기 예시:

^repository "^default" // 작업 리포지토리

^path "/" // 규칙이 루트 디렉터리에 적용됨

^branch "/^main" // ^br:/^main에서 최신 개정을 가져옴

^checkout "/^main" // ^br:/^main 브랜치에서 체크아웃

예:

cm ^sts

(적용할 현재 선택기 파일을 엽니다.)

cm ^sts ^wk:workspace\_projA@reptest

(적용할 특정 선택기 파일을 엽니다.)

cm ^setselector --^file=c:\selectors\sel.xml

(현재 워크스페이스에서 특정 선택기 파일을 설정합니다.)

cm ^setselector --^file=c:\selectors\sel.xml ^wk:MyWorkspace

(선택한 워크스페이스에서 특정 선택기 파일을 설정합니다.)

== CMD\_DESCRIPTION\_SHELVE ==

체크아웃된 항목의 내용을 임시 저장합니다.

== CMD\_USAGE\_SHELVE ==

이 명령은 사용되지 않습니다. 'cm ^shelveset'를 대신 사용하십시오.

사용법:

cm ^shelve [<항목 경로>+] [--^all] [--^dependencies]

[-^c=문자열 코멘트 | -^commentsfile=<코멘트 파일>]

[--^encoding=이름] [--^comparisonmethod=비교 방법]

(내용을 임시 저장합니다.)

cm ^shelve --^apply=<임시 저장 사양> [--^mount]

(저장된 임시 저장 세트를 적용합니다.)

--^apply 지정된 임시 저장 세트에서 임시 저장된 내용을 복원합니다.

임시 저장 사양은 'cm ^help ^objectspec'을 확인하십시오.

cm ^shelve --^delete=<임시 저장 사양>

(저장된 임시 저장 세트를 제거합니다.)

--^delete 지정된 임시 저장 세트를 제거합니다.

임시 저장 세트 사양은 'cm ^help ^objectspec'을 확인하십시오.

옵션:

항목 경로 임시 저장할 항목이며, 공백으로 구분합니다. 큰따옴표(")를

사용하여 공백이 포함된 경로를 지정할 수 있습니다.

--^all 특정 경로에서 변경, 이동 및 삭제가 로컬로 이루어진

항목도 포함됩니다.

--^dependencies 임시 저장할 항목에 로컬 변경 종속성을

포함시킵니다.

-^c 생성된 임시 저장 세트에 지정된 코멘트를 적용합니다.

-^commentsfile 생성된 임시 저장 세트에 지정된 파일의 코멘트를

적용합니다.

--^encoding 출력 인코딩, 즉 UTF-8을 지정합니다.

지원되는 인코딩 및 형식에 대한 표를 확인하려면

http://msdn.microsoft.com/en-us/library/system.text.encoding.aspx에서

MSDN 문서를 확인하십시오.

페이지 마지막 "Name"열에 있습니다.

--^comparisonmethod 비교 방법을 설정합니다. 자세한 내용은 비고를 확인하십시오.

--^mount 지정된 리포지토리의 마운트 지점입니다.

== CMD\_HELP\_SHELVE ==

비고:

<항목 경로>가 지정되어 있지 않고 옵션도 지정되어 있지 않은 경우 임시 저장에

워크스페이스의 대기 중인 변경사항이 모두 포함됩니다.

임시 저장 작업은 항상 주어진 경로에서 재귀적으로 적용됩니다.

항목 임시 저장 요구사항:

- 항목이 소스 코드 제어 상태여야 합니다.

- 항목이 체크아웃되거나 변경되어야 합니다(--^all 옵션을 사용해야 함).

비교 방법:

^ignoreeol 행 끝 차이를 무시합니다.

^ignorewhitespaces 공백 차이를 무시합니다.

^ignoreeolwhitespaces 행 끝과 공백 차이를 무시합니다.

^notignore 행 끝과 공백 차이를 감지합니다.

코멘트를 입력할 편집기를 지정하려면 PLASTICEDITOR 환경 변수를

설정하십시오.

예:

cm ^shelve -^c="내 코멘트"

(코멘트를 포함하여 현재 워크스페이스의 대기 중인 변경사항을 모두

임시 저장합니다.)

cm ^shelve file1.txt "file 2.txt" -^commentsfile=commentshelve.txt

(선택한 대기 중인 변경사항을 임시 저장하고 commentshelve.txt 파일의

코멘트를 적용합니다.)

cm ^shelve --^apply=^sh:3

(저장된 임시 저장 세트를 적용합니다.)

cm ^shelve --^delete=^sh:3

(저장된 임시 저장 세트를 제거합니다.)

cm ^status --^short --^changelist=pending\_to\_review | cm ^shelve -

(클라이언트 변경사항 목록을 임시 저장합니다.

위 명령은 'pending\_to\_review'라는 변경사항 목록에

경로를 나열하며, 경로 목록은 shelve 명령 입력으로

리디렉션됩니다.)

== CMD\_DESCRIPTION\_SHELVESET ==

사용자가 임시 저장 세트를 관리하도록 허용합니다.

== CMD\_USAGE\_SHELVESET ==

사용법:

cm ^shelveset <명령> [옵션]

명령:

^create | ^mk

^delete | ^rm

^apply

각 명령에 대한 자세한 정보는 다음을 실행해 확인하십시오.

cm ^shelveset <명령> --^usage

cm ^shelveset <명령> --^help

== CMD\_HELP\_SHELVESET ==

예:

cm ^shelveset ^create -^c="내 코멘트"

cm ^shelveset ^delete ^sh:3

cm ^shelve ^apply ^sh:3

== CMD\_DESCRIPTION\_SHELVESET\_CREATE ==

대기 중인 변경사항을 임시 저장합니다.

== CMD\_USAGE\_SHELVESET\_CREATE ==

사용법:

cm ^shelveset ^create | ^mk [<항목 경로>[ ...]] [--^all] [--^dependencies]

[-^c=<문자열 코멘트> | -^commentsfile=<코멘트 파일>]

옵션:

항목 경로 임시 저장할 항목입니다. 사용자 이름을 구분하려면 공백을 사용하십시오.

공백이 포함된 경로를 지정하려면 큰따옴표(" ")를

사용하십시오.

--^all 특정 경로에서 변경, 이동 및 삭제가 로컬로 이루어진

항목도 포함됩니다.

--^dependencies 임시 저장할 항목에 로컬 변경 종속성을

포함시킵니다.

-^c 생성된 임시 저장에 지정된 코멘트를 적용합니다.

-^commentsfile 생성된 임시 저장에 지정된 파일의 코멘트를

적용합니다.

== CMD\_HELP\_SHELVESET\_CREATE ==

'^shelveset ^create' 명령은 리포지토리 내부에서 체크아웃된 항목의 내용을

저장합니다. 이렇게 하면 파일을 체크인할 필요 없이 내용을 보호할 수

있습니다.

비고:

<항목 경로>가 지정되어 있지 않고 옵션도 지정되어 있지 않은 경우 임시

저장 세트에 워크스페이스의 대기 중인 변경사항이 모두 포함됩니다.

'^shelveset ^create' 작업은 항상 주어진 경로에서 재귀적으로

적용됩니다.

항목 임시 저장 요구사항:

- 항목이 소스 코드 제어 상태여야 합니다.

- 항목이 체크아웃되거나 변경되어야 합니다('--^all' 옵션을 사용해야 함).

코멘트를 입력할 편집기를 지정하려면 PLASTICEDITOR 환경 변수를

설정하십시오.

예:

cm ^shelveset ^create -^c="내 코멘트"

(코멘트를 포함하여 현재 워크스페이스의 대기 중인 변경사항을 모두

임시 저장합니다.)

cm ^shelveset file1.txt "file 2.txt" -^commentsfile=commentshelve.txt

(선택한 대기 중인 변경사항을 임시 저장하고 'commentshelve.txt' 파일의

코멘트를 적용합니다. 참고로 '^create'가 기본 하위 명령입니다.)

cm ^status --^short --^changelist=pending\_to\_review | cm ^shelveset -

(클라이언트 변경사항 목록을 임시 저장합니다.

위 명령은 'pending\_to\_review'라는 변경사항 목록에

경로를 나열하며 경로 목록은

'^shelveset' 명령 입력으로 리디렉션됩니다.)

== CMD\_DESCRIPTION\_SHELVESET\_DELETE ==

임시 저장 세트를 삭제합니다.

== CMD\_USAGE\_SHELVESET\_DELETE ==

사용법:

cm ^shelveset ^delete | ^rm <임시 저장 사양>

임시 저장 사양 임시 저장 세트 사양입니다. (임시 저장 세트 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을

사용하십시오.)

== CMD\_HELP\_SHELVESET\_DELETE ==

'^shelveset ^delete' 명령은 임시 저장 세트를 삭제합니다.

예:

cm ^shelveset ^delete ^sh:3

(저장된 임시 저장 세트를 제거합니다.)

== CMD\_DESCRIPTION\_SHELVESET\_APPLY ==

저장된 임시 저장 세트를 적용합니다.

== CMD\_USAGE\_SHELVESET\_APPLY ==

사용법:

cm ^shelveset ^apply <임시 저장 사양> [--^mount] [--^encoding=<이름>]

[--^comparisonmethod=(^ignoreeol | ^ignorewhitespaces| \

^ignoreeolwhitespaces | ^notignore)]

임시 저장 사양 임시 저장 세트 사양입니다. (임시 저장 세트 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을

사용하십시오.)

옵션:

--^mount 지정된 리포지토리의 마운트 지점입니다.

--^encoding 출력 인코딩, 즉 UTF-8을 지정합니다.

지원되는 인코딩 및 형식에 대한 표를 확인하려면

http://msdn.microsoft.com/en-us/library/system.text.encoding.aspx에서

MSDN 문서를 확인하십시오.

페이지 마지막 "Name"열에 있습니다.

--^comparisonmethod 비교 방법을 설정합니다. 자세한 내용은 비고를 확인하십시오.

== CMD\_HELP\_SHELVESET\_APPLY ==

'^shelveset ^apply' 명령은 저장된 임시 저장 세트의 내용을 복원합니다.

비고:

비교 방법:

^ignoreeol 행 끝 차이를 무시합니다.

^ignorewhitespaces 공백 차이를 무시합니다.

^ignoreeolwhitespaces 행 끝과 공백 차이를 무시합니다.

^notignore 행 끝과 공백 차이를 감지합니다.

예:

cm ^shelveset ^apply ^sh:3

(저장된 임시 저장을 적용합니다.)

== CMD\_DESCRIPTION\_SHOW\_FIND\_OBJECTS ==

객체 및 속성을 나열합니다.

== CMD\_USAGE\_SHOW\_FIND\_OBJECTS ==

사용법:

cm ^showfindobjects

== CMD\_HELP\_SHOW\_FIND\_OBJECTS ==

사용 가능한 객체 및 속성:

^attribute:

다음 필드를 사용해 필터링하여 속성을 찾을 수 있습니다.

^type : 문자열입니다.

예:

cm ^find ^attribute "^where ^type = '상태'"

('상태' 유형의 모든 속성을 찾습니다.)

^value : 문자열입니다.

^date : 날짜입니다.

자세한 내용은 이 가이드의 "날짜 상수"를 확인하십시오.

예:

cm ^find ^attribute "^where ^date > '^this ^week'"

(금주에 적용된 모든 속성을 찾습니다.)

^owner : 사용자입니다.

특수 사용자 '^me'를 허용합니다.

예:

cm ^find ^attribute "^where ^value = '해결됨' ^and ^owner = '^me'"

(내가 적용한 '해결됨' 값이 있는 모든 속성을 찾습니다.)

^GUID : 전역 고유 식별자입니다.

xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx 형식의 16진수 ID입니다.

^comment : 문자열입니다.

^srcobj : 객체 사양: 항목 경로, 브랜치, 체인지 세트, 개정 또는 레이블입니다.

이러한 객체를 지정하는 방법을 알아보려면 'cm ^help ^objectspec'을 사용하십시오.

예:

cm ^find ^attribute "^where ^srcobj = '^item:readme.txt'"

('readme.txt' 항목에 적용된 속성을 찾습니다.)

cm ^find ^attribute "^where ^srcobj = '^br:/main/scm23343'"

(scm23343 브랜치에 적용된 속성을 찾습니다.)

cm ^find ^attribute "^where ^srcobj = '^rev:readme.txt#^br:/main/task002'"

(지정된 개정에 적용된 속성을 찾습니다.)

cm ^find ^attribute "^where ^srcobj = '^rev:^revid:1126'"

(지정된 개정 ID에 적용된 속성을 찾습니다.)

^ID : 정수입니다.

^attributetype:

다음 필드를 사용해 필터링하여 속성 유형을 찾을 수 있습니다.

^name : 문자열입니다.

예:

cm ^find ^attributetype "^where ^name ^like 'st%'"

(이름이 'st'로 시작하는 모든 속성을 찾습니다.)

^value : 문자열입니다.

^date : 날짜입니다.

자세한 내용은 이 가이드의 "날짜 상수"를 확인하십시오.

예:

cm ^find ^attribute "^where ^date > '^today'"

(오늘 적용된 모든 속성을 찾습니다.)

^owner : 사용자입니다.

특수 사용자 '^me'를 허용합니다.

^GUID : 전역 고유 식별자입니다.

xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx 형식의 16진수 ID입니다.

^comment : 문자열입니다.

예:

cm ^find ^attributetype "^where ^comment != ''" --^xml

(코멘트가 있는 모든 속성 유형을 찾고 XML 형식의 출력을

표준 출력으로 출력합니다.)

^source : 객체 사양: 항목 경로, 브랜치, 체인지 세트 또는 레이블입니다.

이러한 객체를 지정하는 방법을 알아보려면 'cm ^help ^objectspec'을 사용하십시오.

예:

cm ^find ^attributetype "^where ^source = '^item:readme.txt'"

('readme.txt' 항목의 모든 속성 유형을 찾습니다.)

cm ^find ^attributetype "^where ^source = '^cs:30'"

('30'번 체인지 세트의 모든 속성 유형을 찾습니다.)

cm ^find ^attributetype "^where ^source = '^lb:v0.14.1'"

('v0.14.1' 레이블의 모든 속성 유형을 찾습니다.)

^ID : 정수입니다.

복제 필드입니다. 아래의 "복제 관련 필드"를 확인하십시오.

^ReplLogId

^ReplSrcDate

^ReplSrcId

^ReplSrcRepository

^ReplSrcServer

^branch:

다음 필드를 사용해 필터링하여 브랜치를 찾을 수 있습니다.

^name : 문자열입니다.

예:

cm ^find ^branch "^where ^name ^like 'scm23%'"

(이름이 'scm23'으로 시작하는 모든 브랜치를 찾습니다.)

^date : 날짜입니다.

자세한 내용은 이 가이드의 "날짜 상수"를 확인하십시오.

예:

cm ^find ^branch "^where ^date > '^one ^week ^ago'"

(지난 주에 생성된 브랜치를 찾습니다.)

^changesets : 브랜치의 체인지 세트 날짜입니다.

자세한 내용은 이 가이드의 "날짜 상수"를 확인하십시오.

예:

cm ^find ^branch "^where ^changesets >= '^today'"

(오늘 생성된 체인지 세트가 있는 브랜치를 찾습니다.)

^attribute : 문자열입니다.

^attrvalue : 문자열입니다.

예:

cm ^find ^branch "^where ^attribute = '상태' ^and ^attrvalue = '실패'"

('상태' 속성이 있고 그 값이 '실패'인

브랜치를 찾습니다.)

^owner : 사용자입니다.

특수 사용자 '^me'를 허용합니다.

^parent : 브랜치 사양입니다.

이 객체를 지정하는 방법을 알아보려면 'cm ^help ^objectspec'을 사용하십시오.

예:

cm ^find ^branch "^where ^owner != '^me' ^and ^parent != '^br:/main'"

(내가 아닌 다른 사용자가 생성했으며 부모 브랜치가 '/main'이 아닌

브랜치를 찾습니다.)

^comment : 문자열입니다.

^GUID : 전역 고유 식별자입니다.

xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx 형식의 16진수 ID입니다.

^ID : 정수입니다.

예:

cm ^find ^branch "^where ^id = 2029607"

(ID가 2029607인 브랜치를 찾습니다.)

복제 필드입니다. 아래의 "복제 관련 필드"를 확인하십시오.

^ReplLogId

^ReplSrcDate

^ReplSrcId

^ReplSrcRepository

^ReplSrcServer

^changeset:

다음 필드를 사용해 필터링하여 체인지 세트를 찾을 수 있습니다.

^branch : 브랜치 사양입니다.

이 객체를 지정하는 방법을 알아보려면 'cm ^help ^objectspec'을

사용하십시오.

예:

cm ^find ^changeset "^where ^branch = '/main/scm23119'"

('scm23119' 브랜치의 모든 체인지 세트를 찾습니다.)

^changesetid : 정수입니다.

^attribute : 문자열입니다.

예:

cm ^find ^changeset "^where ^attribute = '상태'"

('상태' 속성이 있는 체인지 세트를 찾습니다.)

^attrvalue : 문자열입니다.

^date : 날짜입니다.

자세한 내용은 이 가이드의 "날짜 상수"를 확인하십시오.

^owner : 사용자입니다.

특수 사용자 '^me'를 허용합니다.

예:

cm ^find ^changeset "^where ^date >= '2018/8/6' ^and ^owner != '^me'"

(생성 날짜가 2018/8/6 이후이며 내가 아닌

다른 사용자가 생성한 모든 체인지 세트를 찾습니다.)

^GUID : 전역 고유 식별자입니다.

xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx 형식의 16진수 ID입니다.

예:

cm ^find ^changeset "^where ^guid = '1b30674f-14cc-4fd7-962b-676c8a6f5cb6'"

(지정된 GUID가 있는 체인지 세트를 찾습니다.)

^comment : 문자열입니다.

예:

cm ^find ^changeset "^where ^comment = ''"

(코멘트가 없는 체인지 세트를 찾습니다.)

^onlywithrevisions : 부울입니다.

체인지 세트에 개정이 있는지 여부를 필터링할 수 있습니다.

예:

cm ^find ^changeset "^where ^onlywithrevisions = 'false'"

(개정이 없는 체인지 세트를 찾습니다.)

^returnparent : 부울입니다.

부모 체인지 세트로 돌아가는 방법입니다. 스크립팅에 유용합니다.

예:

cm ^find ^changeset "^where ^changesetid = 29 ^and ^returnparent = 'true'"

(29번 체인지 세트의 부모를 찾습니다.)

^parent : 체인지 세트 ID입니다(정수).

예:

cm ^find ^changeset "^where ^parent = 548"

(부모가 체인지 세트 548번인 모든 체인지 세트를 찾습니다.)

^ID : 정수입니다.

복제 필드입니다. 아래의 "복제 관련 필드"를 확인하십시오.

^ReplLogId

^ReplSrcDate

^ReplSrcId

^ReplSrcRepository

^ReplSrcServer

^label:

다음 필드를 사용해 필터링하여 레이블을 찾을 수 있습니다.

^name : 문자열입니다.

예:

cm ^find ^label "^where ^name ^like '7.0.16.%'"

(이름이 '7.0.16.'으로 시작하는 레이블을 찾습니다.)

^attribute : 문자열입니다.

^attrvalue : 문자열입니다.

^date : 날짜입니다.

자세한 내용은 이 가이드의 "날짜 상수"를 확인하십시오.

예:

cm ^find ^label "^where ^date >= '^this ^month' ^and \

^attribute = '게시 상태' ^and ^attrvalue != '게시됨'"

(이번 달에 생성되었으며 설정된 '게시 상태' 속성 값이 '게시됨'이

아닌 레이블을 찾습니다.)

^owner : 사용자입니다.

특수 사용자 '^me'를 허용합니다.

^GUID : 전역 고유 식별자입니다.

xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx 형식의 16진수 ID입니다.

^branch : 브랜치 사양입니다.

이 객체를 지정하는 방법을 알아보려면 'cm ^help ^objectspec'을 사용하십시오.

예:

cm ^find ^label "^where ^branch = '/main'"

(main 브랜치에 적용된 모든 레이블을 찾습니다.)

^branchid : 정수입니다.

^changeset : 체인지 세트 ID입니다(정수).

예:

cm ^find ^label "^where ^changeset = 111733"

(111733번 체인지 세트에 적용된 레이블을 찾습니다.)

^comment : 문자열입니다.

^ID : 정수입니다.

복제 필드입니다. 아래의 "복제 관련 필드"를 확인하십시오.

^ReplLogId

^ReplSrcDate

^ReplSrcId

^ReplSrcRepository

^ReplSrcServer

^merge:

다음 필드를 사용해 필터링하여 병합을 찾을 수 있습니다.

^srcbranch : 브랜치 사양입니다.

이 객체를 지정하는 방법을 알아보려면 'cm ^help ^objectspec'을 사용하십시오.

예:

cm ^find ^merge "^where ^srcbranch = '^br:/main'"

(main 브랜치에서 병합을 찾습니다.)

^srcchangeset : 체인지 세트 ID입니다(정수).

^dstbranch : 브랜치 사양입니다.

이 객체를 지정하는 방법을 알아보려면 'cm ^help ^objectspec'을 사용하십시오.

^dstchangeset : 체인지 세트 ID입니다(정수).

예:

cm ^find ^merge "^where ^dstchangeset = 108261" \

--^format="{^srcbranch} {^srcchangeset} {^dstbranch} {^dstchangeset} {^owner}"

(108261번 체인지 세트에 대한 병합을 찾고 형식에 맞게 출력하여

소스(브랜치 및 체인지 세트 ID), 대상(브랜치 및 체인지 세트 ID),

병합 소유자를 표시합니다.)

^date : 날짜입니다.

자세한 내용은 이 가이드의 "날짜 상수"를 확인하십시오.

^owner : 사용자입니다.

특수 사용자 '^me'를 허용합니다.

^GUID : 전역 고유 식별자입니다.

xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx 형식의 16진수 ID입니다.

^type : 문자열입니다.

가능한 값은 '^merge', '^cherrypick',

'^cherrypicksubstractive', '^interval', '^intervalcherrypick'

및 '^intervalcherrypicksubstractive'입니다.

예:

cm ^find ^merge "^where ^type = '^cherrypick' ^and ^owner = '^me'"

(내 cherrypick을 모두 찾습니다.)

^ID : 정수입니다.

^replicationlog:

다음 필드를 사용해 필터링하여 복제 로그를 찾을 수 있습니다.

^branch : 브랜치 사양입니다.

이 객체를 지정하는 방법을 알아보려면 'cm ^help ^objectspec'을 사용하십시오.

예:

cm ^find ^replicationlog "^where ^branch = '/main/gm22358'"

('gm22358' 브랜치의 복제 로그를 찾습니다.)

^repositoryname : 문자열입니다.

^owner : 사용자입니다.

특수 사용자 '^me'를 허용합니다.

^date : 날짜입니다.

자세한 내용은 이 가이드의 "날짜 상수"를 확인하십시오.

^server : 문자열입니다.

^package : 부울입니다.

예:

cm ^find ^replicationlog "^where ^package = 'T' ^and ^server ^like '%cloud%'"

(서버 이름에 'cloud'가 포함된 패키지에서 생성된

복제 로그를 찾습니다.)

^ID : 정수입니다.

^review:

다음 필드를 사용해 필터링하여 코드 검토를 찾을 수 있습니다.

^status : 문자열입니다.

^assignee : 문자열입니다.

예:

cm ^find ^review "^where ^status = '대기 중' ^and ^assignee = '^me'"

(내 대기 중인 검토를 모두 찾습니다.)

^title : 문자열입니다.

^target : 객체 사양: 브랜치 또는 체인지 세트입니다.

이 객체를 지정하는 방법을 알아보려면 'cm ^help ^objectspec'을 사용하십시오.

예:

cm ^find ^review "^where ^target = '^br:/main/scm17932'"

('scm17932' 브랜치와 관련된 검토를 찾습니다.)

^targetid : 정수입니다.

^targettype : 문자열입니다.

가능한 값은 '^branch' 및 '^changeset'입니다.

예:

cm ^find ^review "^where ^targettype = '^changeset'"

(대상 유형이 체인지 세트인 검토를 찾습니다.)

^date : 날짜입니다.

자세한 내용은 이 가이드의 "날짜 상수"를 확인하십시오.

^owner : 사용자입니다.

특수 사용자 '^me'를 허용합니다.

^GUID : 전역 고유 식별자입니다.

xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx 형식의 16진수 ID입니다.

^ID : 정수입니다.

^revision:

다음 필드를 사용해 필터링하여 개정을 찾을 수 있습니다.

^branch : 브랜치 사양입니다.

이 객체를 지정하는 방법을 알아보려면 'cm ^help ^objectspec'을 사용하십시오.

^changeset : 체인지 세트 ID입니다(정수).

예:

cm ^find ^revision "^where ^changeset >= 111756"

(111756번 이상의 체인지 세트에서 생성된 개정을

찾습니다.)

^item : 문자열 또는 정수입니다.

^itemid : 정수입니다.

예:

cm ^find ^revision "^where ^item = 'readme.txt' ^or ^itemid = 2250"

('readme.txt' 항목이나 항목 ID 2250의 개정을

찾습니다.)

cm ^find ^revision "^where ^item = 'readme.txt' ^or ^item = 2250"

(이전 예와 동일한 개정을 찾습니다.)

^attribute : 문자열입니다.

^attrvalue : 문자열입니다.

예:

cm ^find ^revision "^where ^attribute = '상태' ^and ^attrvalue != '공개'"

('상태' 속성 값이 '공개'가 아닌

개정을 찾습니다.)

^archived : 부울입니다.

예:

cm ^find ^revision "^where ^archived = 'true'"

(외부 스토리지에 보관된 개정을

찾습니다.)

^comment : 문자열입니다.

^date : 날짜입니다.

자세한 내용은 이 가이드의 "날짜 상수"를 확인하십시오.

^GUID : 전역 고유 식별자입니다.

xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx 형식의 16진수 ID입니다.

^owner : 사용자입니다.

특수 사용자 '^me'를 허용합니다.

^parent : 개정 ID입니다(정수).

^returnparent : 부울입니다.

^shelve : 임시 저장 ID입니다(정수).

^size : 정수입니다(단위: 바이트).

^type : 문자열입니다.

가능한 값은 '^dir', '^bin' 및 '^txt'입니다.

예:

cm ^find ^revision "^where ^type = '^txt' and \

^size > 300000 ^and ^owner = '^me' and ^date >= '2 ^months ^ago'"

(크기가 약 3MB보다 크며 2개월 전에 내가 생성한

텍스트 개정을 찾습니다.)

^workspacecheckoutid : 정수입니다.

^ID : 정수입니다.

복제 필드입니다. 아래의 "복제 관련 필드"를 확인하십시오.

^ReplLogId

^ReplSrcDate

^ReplSrcId

^ReplSrcRepository

^ReplSrcServer

^shelve:

다음 필드를 사용해 필터링하여 임시 저장을 찾을 수 있습니다.

^owner : 사용자입니다.

특수 사용자 '^me'를 허용합니다.

^date : 날짜입니다.

자세한 내용은 이 가이드의 "날짜 상수"를 확인하십시오.

예:

cm ^find ^shelve "^where ^owner != '^me' ^and ^date >= '^1 ^years ^ago'"

(작년에 내가 아닌 다른 사용자가 생성한 임시 저장을

찾습니다.)

^attribute : 문자열입니다.

^attrvalue : 문자열입니다.

^comment : 문자열입니다.

^GUID : 전역 고유 식별자입니다.

xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx 형식의 16진수 ID입니다.

^parent : 정수입니다.

^shelveid : 정수입니다.

예:

cm ^find ^shelve "^where ^shelveid = 2"

(이름이 2인 임시 저장을 찾습니다.)

^ID : 정수입니다.

예:

cm ^find ^shelve "^where ^id >= 3848"

(객체 ID가 3848보다 큰 임시 저장을 찾습니다.)

복제 필드입니다. 아래의 "복제 관련 필드"를 확인하십시오.

^ReplLogId

^ReplSrcDate

^ReplSrcId

^ReplSrcRepository

^ReplSrcServer

복제 관련 필드:

여러 객체가 복제 데이터를 추적합니다. 즉, Plastic에서 원래 생성 위치를

추적합니다.

사용할 수 있는 필드는 다음과 같습니다.

^ReplSrcServer : 리포지토리 사양입니다, "복제 소스 서버"를 의미합니다.

객체가 복제된 서버입니다.

예:

cm ^find ^branch "^where ^replsrcserver='skull.codicefactory.com:9095'"

('skull' 서버에서 복제된 브랜치를 찾습니다.)

^ReplSrcRepository : 문자열입니다. "복제 소스 리포지토리"를 의미합니다. 객체가

복제된 리포지토리입니다.

예:

cm ^find ^branch "^where ^replsrcserver = 'skull.codicefactory.com:9095' \

^and ^replsrcrepository = 'codice'"

('skull' 서버와 'codice' 리포지토리에서 복제된 브랜치를

찾습니다.)

^ReplLogId : 정수입니다. 복제 작업 ID입니다. Plastic에서는

복제로 인해 새 객체가 생성될 때마다

새 'replicationlog'가 생성됩니다.

예:

cm ^find ^revision "^where ^repllogid = 2019974"

(복제본으로부터 복제된 개정을 찾습니다

2019974.)

^ReplSrcDate : 날짜입니다. 복제가 실제로 이루어진

날짜입니다.

복제된 객체의 원래 생성 날짜는 유지되며,

이 필드는 특정 기간 내에 복제된

객체를 찾으려는 경우에

유용합니다.

예:

cm ^find ^label "^where ^replsrcdate >= '^one ^month ^ago' \

^and ^date >= '15 ^days ^ago'"

(15일 전에 생성되었으며 1개월 전에 복제된 레이블을

찾습니다.)

cm ^find ^replicationlog "^where ^date > '^one ^week ^ago'"

8780433 27/09/2018 8:49:38 codice@BACKYARD:8087 F mbarriosc

(1주일 전에 생성된 복제 로그를 찾습니다.)

이제 복제된 브랜치가 생성 전에 복제되었는지

확인할 수 있습니다.

cm ^find ^branch "^where ^repllogid = 8780433"

8780443 26/09/2018 12:20:55 /main/scm23078 maria codice T

^ReplSrcId : 정수입니다. 복제 소스 서버 ID입니다.

'cm ^find' 명령을 통해 '^replicationsource'

객체를 검색하여 이 ID를 찾을 수 있습니다.

예:

cm ^find ^replicationsource

7860739 codice@AFRODITA:8087 d9c4372a-dc55-4fdc-ad3d-baeb2e975f27

8175854 codice@BACKYARD:8087 66700d3a-036b-4b9a-a26f-adfc336b14f9

이제 codice@AFRODITA:8087에서 복제된 체인지 세트를

찾을 수 있습니다.

cm ^find ^changesets "^where ^replsrcid = 7860739"

날짜 상수:

시스템의 현지화 설정을 준수하는 날짜 형식을 사용할 수 있습니다.

예를 들어 컴퓨터에서 날짜를 'MM-dd-yyyy' 형식으로 표시하는 경우,

쿼리 시 '12-31-2019'와 같은 날짜를 사용할 수 있습니다.

다음 상수를 사용하여 쿼리를 간소화할 수도 있습니다.

'^today' : 오늘 날짜.

'^yesterday' : 어제 날짜.

'^this ^week' : 금주 월요일 날짜.

'^this ^month' : 이번 달 1일.

'^this ^year' : 이번 연도 1월 1일.

'^one ^day ^ago' : 현재 날짜의 하루 전.

'^one ^week ^ago' : 현재 날짜의 7일 전.

'^one ^month ^ago' : 현재 날짜의 1개월 전.

'n ^days ^ago' : 현재 날짜의 'n'일 전.

'n ^months ^ago' : 현재 날짜의 'n'개월 전.

'n ^years ^ago' : 현재 날짜의 'n'년 전.

다음 '^where' 절은 '^date' 필드 유형에서 유효합니다.

'(...) ^where ^date > '^today' (...)'

'(...) ^where ^date < '^yesterday' (...)'

'(...) ^where ^date > '^this ^week' (...)'

'(...) ^where ^date > '^this ^month' (...)'

'(...) ^where ^date < '^one ^day ^ago' ^and ^date > '3 ^days ^ago' (...)'

'(...) ^where ^date < '^one ^week ^ago' ^and ^date > '3 ^weeks ^ago' (...)'

'(...) ^where ^date < '^one ^month ^ago' ^and ^date > '3 ^months ^ago' (...)'

'(...) ^where ^date > '1 ^year ^ago' (...)'

'cm ^find' 명령에

--^dateformat 플래그를 사용하여 특정 날짜 형식을 강제로 사용할 수도 있습니다. 자세한 내용은 'cm ^find --^help'를 확인하십시오.

== CMD\_DESCRIPTION\_TRIGGER\_SHOWTYPES ==

사용 가능한 트리거 유형을 표시합니다.

== CMD\_USAGE\_TRIGGER\_SHOWTYPES ==

사용법:

cm ^trigger ^showtypes

== CMD\_DESCRIPTION\_SHOWACL ==

객체의 ACL을 표시합니다.

== CMD\_USAGE\_SHOWACL ==

사용법:

cm ^showacl | ^sa <객체 사양> [--^extended] [--^xml[=<출력 파일>]]

[--^encoding=<이름>]

객체 사양 ACL을 표시할 객체의 사양입니다.

이 명령에 유효한 객체는 다음과 같습니다.

리포지토리 서버, 리포지토리, 브랜치, 체인지 세트, 레이블, 항목,

속성.

(사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을 사용하십시오.)

옵션:

--^extended ACL 계층 트리를 표시합니다.

--^xml XML 형식의 출력을 표준 출력으로 출력합니다.

출력 파일을 지정할 수 있습니다.

--^encoding '--^xml' 옵션과 함께 사용되며 XML 출력에 사용할

인코딩, 즉 UTF-8을 지정합니다.

지원되는 인코딩 및 형식에 대한 표를 확인하려면

http://msdn.microsoft.com/en-us/library/system.text.encoding.aspx에서

MSDN 문서를 확인하십시오.

페이지 마지막 "Name"열에 있습니다.

== CMD\_HELP\_SHOWACL ==

예:

cm ^showacl ^repserver:PlasticServer:8084

(선택한 서버의 ACL을 표시합니다.)

cm ^sa ^br:/main --^extended

(선택한 브랜치 사양의 ACL 계층 트리를 표시합니다.)

== CMD\_DESCRIPTION\_SHOWCOMMANDS ==

사용 가능한 명령을 모두 표시합니다.

== CMD\_USAGE\_SHOWCOMMANDS ==

사용법:

cm ^showcommands

== CMD\_HELP\_SHOWCOMMANDS ==

== CMD\_DESCRIPTION\_SHOWOWNER ==

객체 소유자를 표시합니다.

== CMD\_USAGE\_SHOWOWNER ==

사용법:

cm ^showowner | ^so <객체 사양>

객체 사양 소유자를 표시할 객체의 사양입니다.

객체는 다음 중 하나여야 합니다.

리포지토리 서버, 리포지토리, 브랜치, 체인지 세트,

레이블, 속성, 개정 및 항목.

(사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을 사용하십시오.)

== CMD\_HELP\_SHOWOWNER ==

비고:

이 명령은 객체 소유자를 표시합니다. 소유자는 사용자나

그룹일 수 있습니다. 사용자는 'cm ^setowner' 명령으로 수정 가능합니다.

예:

cm ^showowner ^repserver:PlasticServer:8084

(선택한 서버의 소유자를 표시합니다.)

cm ^so ^item:samples\

(선택한 항목 사양의 소유자를 표시합니다.)

== CMD\_DESCRIPTION\_SHOWPERMISSIONS ==

사용 가능한 권한을 나열합니다.

== CMD\_USAGE\_SHOWPERMISSIONS ==

사용법:

cm ^showpermissions | ^sp

== CMD\_HELP\_SHOWPERMISSIONS ==

예:

cm ^showpermissions

== CMD\_DESCRIPTION\_SHOWSELECTOR ==

워크스페이스 선택기를 표시합니다.

== CMD\_USAGE\_SHOWSELECTOR ==

이 명령은 사용되지 않습니다. 단, 이전 버전과의 호환성을 위해 남아 있습니다.

Plastic SCM 4.0에서는 대부분의 선택기가 사용되지 않습니다. 선택기는

작업 브랜치 또는 체인지 세트를 지정할 용도로 남아 있지만 경로를 필터링하는 기존 규칙은

더 이상 지원되지 않습니다.

사용법:

cm ^showselector | ^ss [<워크스페이스 경로> | <워크스페이스 사양>]

워크스페이스 경로 선택기를 표시할 워크스페이스의 경로입니다.

워크스페이스 사양 워크스페이스 사양입니다. (워크스페이스 사양에 관한 자세한 내용을

확인하려면 'cm ^help ^objectspec'을 사용하십시오.)

== CMD\_HELP\_SHOWSELECTOR ==

비고:

경로와 워크스페이스 사양이 지정되지 않은 경우 해당 명령은 현재 디렉터리를

워크스페이스 경로로 사용합니다.

예:

cm ^showselector c:\workspace

(선택한 워크스페이스 경로의 선택기를 표시합니다.)

cm ^ss

(현재 워크스페이스의 선택기를 표시합니다.)

cm ^showselector > mySelector.txt

(파일에 현재 워크스페이스의 선택기를 작성합니다.)

cm ^showselector ^wk:mywk@reptest

('reptest' 리포지토리에 있는 'mywk' 워크스페이스의 선택기를 표시합니다.)

== CMD\_DESCRIPTION\_SUPPORT ==

사용자가 지원 관련 작업을 수행할 수 있도록 허용합니다.

== CMD\_USAGE\_SUPPORT ==

사용법:

cm ^support <명령> [옵션]

명령:

^bundle

각 명령에 대한 자세한 정보는 다음을 실행해 확인하십시오.

cm ^support <명령> --^usage

cm ^support <명령> --^help

== CMD\_HELP\_SUPPORT ==

예:

cm ^support

cm ^support ^bundle

cm ^support ^bundle c:\outputfile.zip

== CMD\_DESCRIPTION\_SUPPORT\_BUNDLE ==

관련 로그가 포함된 "지원 번들" 패키지를 생성합니다.

도움을 구하거나, 추가 정보를 요청하거나, 버그를 제출하면서

파일을 첨부할 수 있습니다.

== CMD\_USAGE\_SUPPORT\_BUNDLE ==

사용법:

cm ^support ^bundle [<outputfile>]

옵션:

outputfile 지정된 위치에 "지원 번들" 패키지를

생성합니다.

== CMD\_HELP\_SUPPORT\_BUNDLE ==

비고:

이 명령을 통해 사용자는 도움을 구하거나, 추가 정보를 요청하거나, 버그를 제출할 때 첨부할 수 있는

"지원 번들" 패키지를 생성할 수 있습니다.

사용자는 출력 파일을 지정할 수도 있으며, 아니면

출력 파일이 임시 디렉터리에 작성됩니다.

예:

cm ^support ^bundle

(임시 디렉터리에 "지원 번들"을 생성합니다.)

cm ^support ^bundle c:\outputfile.zip

(지정된 위치에 "지원 번들"을 생성합니다.)

== CMD\_DESCRIPTION\_SWITCH ==

워크스페이스를 브랜치, 체인지 세트, 레이블 또는 임시 저장 세트로 전환합니다.

== CMD\_USAGE\_SWITCH ==

사용법:

cm ^switch (<브랜치 사양> | <체인지 세트 사양> | <레이블 사양> | <임시 저장 사양>)

[--^workspace=<경로>] [--^repository=<이름>]

[--^forcedetailedprogress]

(브랜치, 체인지 세트, 레이블 및 임시 저장 세트 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을

사용하십시오.)

옵션:

--^workspace 워크스페이스가 있는 경로입니다.

--^repository 지정된 리포지토리로 전환합니다.

--^forcedetailedprogress 표준 출력이 리디렉션되더라도 진행도를 강제로

표시합니다.

== CMD\_HELP\_SWITCH ==

비고:

이 명령을 통해 사용자는 지정된 객체(브랜치, 레이블, 임시 저장 세트 또는 체인지 세트)의 내용에

워크스페이스 트리를 업데이트할 수 있습니다.

예:

cm ^switch ^br:/main

cm ^switch ^lb:Rel1.1

cm ^switch ^br:/main/scm002 --^repository=rep2

cm ^switch ^cs:4375

cm ^switch ^sh:2

== CMD\_DESCRIPTION\_SWITCH\_TO\_BRANCH ==

브랜치를 작업 브랜치로 설정합니다.

== CMD\_USAGE\_SWITCH\_TO\_BRANCH ==

이 명령은 사용되지 않습니다. cm switch를 대신 사용하십시오.

사용법:

cm ^switchtobranch [옵션] [브랜치 사양]

브랜치 사양: 브랜치 사양입니다.

옵션:

--^label=name | --^changeset=number: 지정된 레이블 또는 체인지 세트에서 개정을

로드합니다. 브랜치 사양이 지정되지 않은 경우 다음 옵션 중 하나가

필요합니다.

--^changeset=cset: 지정된 체인지 세트로 전환합니다.

--^repository=rep: 지정된 리포지토리로 전환합니다.

--^workspace | -wk=path: 워크스페이스가 있는 경로입니다.

== CMD\_HELP\_SWITCH\_TO\_BRANCH ==

비고:

이 명령을 통해 사용자는 브랜치에서 작업할 수 있습니다.

브랜치 사양이 지정되지 않은 경우 레이블 또는 체인지 세트를 지정해야 합니다.

리포지토리가 지정되지 않으면 현재 리포지토리에 브랜치가 지정됩니다.

예:

cm ^switchtobranch ^br:/main

cm ^switchtobranch ^br:/main/task001

cm ^switchtobranch --^label=BL050

(읽기 전용 구성입니다. 이 명령은 레이블이 적용된 체인지 세트의 내용을

로드합니다.)

== CMD\_DESCRIPTION\_SYNC ==

Git와 동기화합니다.

== CMD\_USAGE\_SYNC ==

사용법:

cm ^synchronize | ^sync <리포지토리 사양> ^git [<url> [--^user=<사용자 이름> --^pwd=<비밀번호>]]

[(--^txtsimilaritypercent | --^binsimilaritypercent | \

--^dirsimilaritypercent)=<값>]

[--^author] [--^skipgitlfs]

리포지토리 사양 리포지토리 사양입니다. (리포지토리 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을

사용하십시오.)

git (기본값).

옵션:

url 원격 리포지토리 URL입니다(http(s):// 또는 git:// 또는

SSH URL).

--^user 지정된 URL의 사용자 이름입니다.

--^pwd 지정된 URL의 비밀번호입니다.

--^txtsimilaritypercent \

--^binsimilaritypercent \

--^dirsimilaritypercent

이동한 항목을 감지하려면 Plastic SCM GUI에서와

동일한 방법을 사용하십시오.

--^author git 작성자의 이름과 타임스탬프 값을 사용합니다.

(기본값: git 커미터)

--^skipgitlfs .gitattributes 파일의 Git LFS 구성을

무시합니다. Git LFS 지원이 없는 것처럼

작동합니다.

== CMD\_HELP\_SYNC ==

비고:

git 리포지토리에 사용자와 비밀번호가 필요한 경우 '^url', '--^user' 및 '--^pwd'

옵션을 사용하십시오.

git 리포지토리에 사용자와 비밀번호가 필요하지 않은 경우 첫 번째 동기화 작업 시 '^url'

옵션을 사용하십시오. 다음 동기화 작업 시 '^url'

옵션은 선택사항입니다.

SSH 프로토콜을 사용해 동기화를 수행하려면 PATH 환경 변수에 추가되고

원격 호스트에 연결할 수 있도록 올바르게 구성된(즉, 비공개/공개 키가 구성된)

'ssh' 클라이언트가 있어야 합니다.

예:

cm ^sync default@localhost:8087 ^git git://localhost/repository

== CMD\_DESCRIPTION\_TRIGGER ==

사용자가 트리거를 관리하도록 허용합니다.

== CMD\_USAGE\_TRIGGER ==

사용법:

cm ^trigger | ^tr <명령> [옵션]

명령:

^create | ^mk

^delete | ^rm

^edit

^list | ^ls

^showtypes

각 명령에 대한 자세한 정보는 다음을 실행해 확인하십시오.

cm ^trigger <명령> --^usage

cm ^trigger <명령> --^help

== CMD\_HELP\_TRIGGER ==

예:

cm ^tr ^mk ^before-mklabel new "/path/to/script" --^server=myserver:8084

cm ^tr ^edit ^before-mklabel 7 --^position=4 --^server=myserver:8084

cm ^tr ^ls ^before-mkbranch --^server=myserver:8084

cm ^tr ^rm ^after-setselector 4

cm ^tr ^showtypes

== CMD\_DESCRIPTION\_TUBE ==

Plastic Tube와 관련된 명령을 실행합니다.

== CMD\_USAGE\_TUBE ==

사용법:

cm ^tube ^config -^u=<사용자> -^p=<비밀번호>

(Plastic SCM에서 지정된 사용자 및 비밀번호로 Plastic Tube를 사용하도록

구성합니다.)

cm ^tube ^create <원격 사용자>

("원격 사용자 -> 내 사용자" Tube를 생성합니다.

"내 사용자" 사용자는 "원격 사용자"가 "내 사용자" 서버에 연결할 수 있도록 허용합니다.

"원격 사용자"에서 "내 사용자"로 연결할 수 있습니다.

다른 사용자로부터 현재 Tube 사용자로 연결된 Tube만 생성할 수 있습니다.)

cm ^tube ^remove <원격 사용자>

("원격 사용자 -> 내 사용자" Tube를 제거합니다.)

cm ^tube ^local

(로컬 서버에 공유된 로컬 리포지토리와 해당 로컬 서버를 공유하는

사용자를 나열합니다.)

cm ^tube ^remote

(현재 Tube 사용자에게 공유되는 공유 원격 리포지토리를

나열합니다.)

cm ^tube ^share <리포지토리 사양>[ ...] -^u=<원격 사용자> -^a=(^pull | ^push | ^pull,^push)

(로컬 리포지토리를 원격 사용자와 공유하고 지정된 액세스 모드를

설정합니다. 리포지토리 사양을 구분하려면 공백을 사용하십시오.)

cm ^tube ^unshare <리포지토리 사양>[ ...] -^u=<원격 사용자>

(로컬 리포지토리와 원격 사용자의 공유를 해제합니다. 리포지토리 사양을 구분하려면

공백을 사용하십시오.)

cm ^tube ^connect

(Plastic Tube에 Plastic SCM 서버를 연결합니다.)

cm ^tube ^disconnect

(Plastic Tube로부터 Plastic SCM 서버의 연결을 해제합니다.)

cm ^tube ^status

(Plastic Tube에 Plastic SCM 서버가 연결되었는지 여부를 표시합니다.)

옵션

-^u Tube 사용자(plasticscm.com 사용자)입니다.

-^p 사용자 비밀번호입니다.

-^a 액세스 모드를 설정합니다.

리포지토리 사양 리포지토리 사양입니다. (리포지토리 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을

사용하십시오.)

== CMD\_HELP\_TUBE ==

비고:

Plastic Tube를 관리하려면 'cm ^tube' 명령을 사용하십시오.

예:

cm ^tube ^config -^u=ruben@codicesoftware.com -^p=rubenpassword

cm ^tube ^create pablo@codicesoftware.com

('pablo@codicesoftware.com'에서 현재 Plastic Tube 사용자로 연결할 수 있음)

cm ^tube ^remove pablo@codicesoftware.com

cm ^tube ^local

cm ^tube ^remote

cm ^tube ^share repo@server:8087 -^u=pablo@codicesoftware.com -^a=^pull,^push

cm ^tube ^share repo@server:8087 doc@server:8087 -^u=pablo@codicesoftware.com -^a=^push

cm ^tube ^unshare repo@server:8087 -^u=pablo@codicesoftware.com

cm ^tube ^connect

cm ^tube ^disconnect

cm ^tube ^status

== CMD\_DESCRIPTION\_UNCO ==

항목의 체크아웃을 취소합니다.

== CMD\_USAGE\_UNCO ==

사용법:

cm ^undocheckout | ^unco <항목 경로>[ ...] [-^a | --^all] [--^symlink] [--^silent]

[--^machinereadable [--^startlineseparator=<구분 기준>]

[--^endlineseparator=<구분 기준>] [--^fieldseparator=<구분 기준>]]

항목 경로 작업을 적용할 항목입니다. 공백을 사용하여 경로를

구분하십시오. 공백이 포함된 경로를 지정하려면 큰따옴표(" ")를

사용하십시오.

현재 디렉터리에 작업을 적용하려면 온점(.)을 사용하십시오.

옵션:

-^a | --^all 지정된 항목의 모든 변경사항을 취소합니다. 항목이

체크아웃된 경우 체크아웃이

되돌려집니다. 항목이 로컬에서 수정된 경우

수정사항이 되돌려집니다.

--^symlink 대상이 아닌 링크에 체크아웃 취소 작업을

적용합니다.

--^silent 어떤 내용도 출력하지 않습니다.

--^machinereadable 결과를 구문 분석이 쉬운 형식으로 출력합니다.

--^startlineseparator '--^machinereadable' 플래그와 함께 사용되며, 행이 시작되는 방법을

지정합니다.

--^endlineseparator '--^machinereadable' 플래그와 함께 사용되며,

행이 끝나는 방법을 지정합니다.

--^fieldseparator '--^machinereadable' 플래그와 함께 사용되며,

필드를 구분할 방법을 지정합니다.

== CMD\_HELP\_UNCO ==

비고:

항목이 체크아웃되고 이를 체크인하지 않으려는 경우 이 명령을 사용해 체크아웃을

취소할 수 있습니다. 파일과 폴더 모두 체크아웃 취소할 수

있습니다. 항목이 체크아웃하기 전 상태로 업데이트됩니다.

요구사항:

- 항목이 소스 코드 제어 상태여야 합니다.

- 항목이 체크아웃된 상태여야 합니다.

표준 입력에서 입력 읽기:

'^undocheckout' 명령은 표준 입력에서 경로를 읽을 수 있습니다. 그러려면 단일 대시

"-" 기호를 넣으십시오.

예: cm ^undocheckout ^checkin -

경로 읽기는 빈 행이 입력될 때까지 계속됩니다.

따라서 파이프를 사용해 체크아웃을 취소할 파일을 지정할 수 있습니다.

예:

dir /S /B \*.c | cm ^undocheckout --^all -

(Windows에서는 워크스페이스에 있는 모든 .c 파일의 체크아웃을 취소합니다.)

예:

cm ^undocheckout .

(현재 디렉터리의 체크아웃을 취소합니다.)

cm ^undocheckout file1.txt file2.txt

cm unco c:\workspace\file.txt

(선택한 파일의 체크아웃을 취소합니다.)

cm ^unco -^a file1.txt

('file1.txt'의 체크아웃 또는 로컬 수정 취소)

cm ^unco 링크 --^symlink

(대상이 아닌 링크 파일에 체크아웃 취소 작업을 적용합니다.

UNIX 환경에서 사용할 수 있습니다.)

cm ^status --^short --^changelist=pending\_to\_review | cm ^undocheckout -

(클라이언트 변경사항 목록을 되돌립니다.

위 명령은 'pending\_to\_review'라는 변경사항 목록에 경로를 나열하며

경로 목록은 undocheckout 명령 입력으로

리디렉션됩니다).

cm ^unco . --^machinereadable

(현재 디렉터리의 체크아웃을 취소하고 그 결과를 구문 분석이 쉬운 간단한 형식으로

출력합니다.)

cm ^unco . --^machinereadable --^startlineseparator=">" --^endlineseparator="<" \

--^fieldseparator=","

(현재 디렉터리의 체크아웃을 취소하고, 그 결과를 구문 분석이 쉬운 간단한

형식으로 출력하며, 지정된 문자열로 행을 시작 및 종료하고

필드를 구분합니다.)

== CMD\_DESCRIPTION\_UNCOUNCHANGED ==

변경 없이 체크아웃된 항목의 체크아웃을 취소합니다.

== CMD\_USAGE\_UNCOUNCHANGED ==

사용법:

cm ^uncounchanged | ^unuc <항목 경로>[ ...] [-^R | -^r | --^recursive]

[--^symlink] [--^silent]

항목 경로 작업을 적용할 항목입니다. 공백을 사용하여 경로를

구분하십시오. 공백이 포함된 경로를 지정하려면 큰따옴표(" ")를

사용하십시오.

현재 디렉터리에 작업을 적용하려면 온점(.)을 사용하십시오.

옵션:

-^R 지정된 경로의 미변경 항목을 재귀적으로 되돌립니다.

--^symlink 대상이 아닌 링크에 미변경 항목에 대한 uncounchanged 작업을

적용합니다.

--^silent 어떤 내용도 출력하지 않습니다.

== CMD\_HELP\_UNCOUNCHANGED ==

비고:

이 명령은 루트 워크스페이스에서 재귀적으로 적용됩니다.

표준 입력에서 입력 읽기:

'^uncounchanged' 명령은 표준 입력에서 경로를 읽을 수 있습니다. 그러려면 단일 대시

"-" 기호를 넣으십시오.

예: cm ^uncounchanged -

경로 읽기는 빈 행이 입력될 때까지 계속됩니다.

따라서 파이프를 사용해 체크아웃을 취소할 미변경 파일을

지정할 수 있습니다.

예:

dir /S /B \*.c | cm ^uncounchanged -

(Windows에서는 워크스페이스에 있는 모든 미변경 .c 파일의 체크아웃을

취소합니다.)

예:

cm ^uncounchanged . -^R

(현재 디렉터리에 있는 미변경 파일의 체크아웃을 재귀적으로 취소합니다.)

cm ^unuc /home/myuser/mywk/project/templates -^R

(선택한 디렉터리에 있는 미변경 파일의 체크아웃을 재귀적으로 취소합니다.)

== CMD\_DESCRIPTION\_UNDELETE ==

특정 개정을 사용하여 항목의 삭제를 취소합니다.

== CMD\_USAGE\_UNDELETE ==

사용법:

cm ^undelete <개정 사양> <경로>

개정 사양 개정 사양입니다. 개정 내용은 워크스페이스에

로드됩니다. (개정 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을

사용하십시오.)

경로 복원 경로입니다.

== CMD\_HELP\_UNDELETE ==

비고:

삭제 취소할 항목이 워크스페이스에 이미 로드되어 있으면 안 됩니다.

Xlink에는 '^undelete' 작업이 지원되지 않습니다.

예:

cm ^undelete ^revid:756 C:\mywks\src\foo.c

cm ^undelete ^itemid:68#^cs:2 C:\mywks\dir\myfile.pdf

cm ^undelete ^serverpath:/src#^br:/main C:\mywks\Dir

== CMD\_DESCRIPTION\_UNDOCHANGE ==

경로에서 이루어진 변경사항을 취소합니다.

== CMD\_USAGE\_UNDOCHANGE ==

사용법:

cm ^undochange | ^unc <항목 경로>[ ...] [-^R | -^r | --^recursive]

항목 경로 작업을 적용할 항목입니다. 공백을 사용하여 경로를

구분하십시오. 공백이 포함된 경로를 지정하려면 큰따옴표(" ")를

사용하십시오.

현재 디렉터리에 작업을 적용하려면 온점(.)을 사용하십시오.

옵션:

-^R 작업을 재귀적으로 적용합니다.

== CMD\_HELP\_UNDOCHANGE ==

비고:

항목이 체크아웃되었거나, 수정되었으나 체크인되지 않아 이를 체크인하려는 경우

이 명령을 사용하여 변경사항을 취소할 수 있습니다. 항목이

이전 내용으로 업데이트됩니다.

표준 입력에서 입력 읽기:

'^undochange' 명령은 표준 입력에서 경로를 읽을 수 있습니다. 그러려면 단일 대시

"-" 기호를 넣으십시오.

예: cm ^undochange -

경로 읽기는 빈 행이 입력될 때까지 계속됩니다.

따라서 파이프를 사용해 변경사항을 실행 취소할 파일을 지정할 수 있습니다.

예:

dir /S /B \*.c | cm ^undochange -

(Windows에서는 워크스페이스에 있는 모든 .c 파일의 변경사항을 실행 취소합니다.)

예:

cm ^unc .

(현재 디렉터리에 있는 파일의 변경사항을 실행 취소합니다.)

cm ^undochange . -^R

(현재 디렉터리에 있는 파일의 변경사항을 실행 취소합니다.)

cm ^unc file1.txt "file 2.txt"

(선택한 파일의 변경사항을 실행 취소합니다.)

cm ^unc c:\workspace\file.txt

(선택한 파일의 변경사항을 실행 취소합니다.)

== CMD\_DESCRIPTION\_UNDO ==

워크스페이스의 변경사항을 실행 취소합니다.

== CMD\_USAGE\_UNDO ==

사용법:

cm ^undo [<경로>[ ...]] [--^symlink] [-^r | --^recursive] [<필터>[ ...]]

[--^silent | --^machinereadable [--^startlineseparator=<구분 기준>]

[--^endlineseparator=<구분 기준>] [--^fieldseparator=<구분 기준>]]

경로 작업을 적용할 파일 또는 디렉터리의

경로입니다. 공백이 포함된 경로를 지정하려면 큰따옴표(" ")를

사용하십시오. 공백을 사용하여 경로를 구분하십시오.

경로가 지정되지 않은 경우, 기본적으로 실행 취소

작업은 현재 디렉터리의 모든 파일을 대상으로

합니다.

필터 지정된 필터를 주어진 경로에

적용합니다. 공백을 사용하여 필터를 구분하십시오. 자세한 내용은

필터 섹션을 확인하십시오.

옵션:

--^symlink 대상이 아닌 기호 링크에 실행 취소 작업을

적용합니다.

-^r 실행 취소를 재귀적으로 실행합니다.

--^silent 어떤 내용도 출력하지 않습니다.

--^machinereadable 결과를 구문 분석이 쉬운 형식으로 출력합니다.

--^startlineseparator '--^machinereadable' 플래그와 함께 사용되며,

행이 시작되는 방법을 지정합니다.

--^endlineseparator '--^machinereadable' 플래그와 함께 사용되며,

행이 끝나는 방법을 지정합니다.

--^fieldseparator '--^machinereadable' 플래그와 함께 사용되며,

필드를 구분할 방법을 지정합니다.

필터:

플래그가 지정되지 않으면 기본적으로 모든 변경사항이 실행 취소되지만

아래의 플래그를 하나 이상 사용하여 경로를 필터링할 수 있습니다.

파일 또는 디렉터리가 하나 이상의 지정된 변경사항 유형과 일치하는 경우

해당 파일 또는 디렉터리의 변경사항이 모두 실행 취소됩니다.

예를 들어 '--^checkedout' 및 '--^moved'를 지정하여 파일을

모두 체크아웃 및 이동한 경우에는 두 변경사항 모두 실행 취소됩니다.

--^checkedout 체크아웃된 파일 및 디렉터리를 선택합니다.

--^unchanged 내용이 변경되지 않은 파일을 선택합니다.

--^changed 로컬에서 변경되거나 체크아웃된 파일 및 디렉터리를

선택합니다.

--^deleted 삭제된 파일 및 디렉터리를 선택합니다.

--^moved 이동한 파일 및 디렉터리를 선택합니다.

--^added 추가된 파일 및 디렉터리를 선택합니다.

== CMD\_HELP\_UNDO ==

비고:

'^undo' 명령은 되돌릴 수 없는 방식으로 작업을 실행 취소하므로 위험합니다.

^undo가 완료되면 작업의 영향을 받은 파일 및 디렉터리를

이전 상태로 복원할 수 없습니다. 인수로 경로가 지정되지

않은 경우 실행 취소 작업은 기본적으로 현재 디렉터리의 모든 변경사항을 실행 취소하지만

작업이 재귀적으로 이루어지지는 않습니다.

/src 디렉터리에서 실행하는 경우에도 마찬가지입니다.

/src

|- file.txt

|- code.cs

\- /test

|- test\_a.py

\- test\_b.py

cm ^undo

cm ^undo \*

cm ^undo file.txt code.cs /test

cm ^undo .

cm ^undo /src file.txt code.cs

작업을 재귀적으로 수행하려면 '-^r' 플래그를 지정해야 합니다.

디렉터리 아래의 모든 변경사항을 실행 취소하려는 경우(디렉터리 자체에

영향을 주는 변경사항 포함):

cm ^undo 디렉터리 경로 -^r

디렉터리 경로가 워크스페이스 경로인 경우 워크스페이스의 모든 변경사항이

실행 취소됩니다.

삭제된 항목:

파일 및 디렉터리 삭제를 실행 취소하려면 항목의 전체 경로를 지정하거나

항목이 포함하고 있는 디렉터리를 지정하고 재귀('-^r') 플래그를

사용합니다.

예:

cm ^undo .

(현재 디렉터리의 삭제를 실행 취소하지 않습니다.)

cm ^undo . -^r

(현재 디렉터리의 모든 삭제(및 다른 변경사항)를 재귀적으로 실행 취소합니다.)

cm ^undo src/file.txt

(src/file.txt의 삭제(또는 다른 변경사항)을 실행 취소합니다.)

예:

cm ^undo . -^r

(현재 디렉터리의 모든 변경사항을 재귀적으로 실행 취소합니다. 워크스페이스의

루트에서 실행되면 전체 워크스페이스의 모든 변경사항을 실행 취소합니다.)

cm ^co file.txt

cm ^undo file.txt

('file.txt'의 체크아웃을 실행 취소합니다.)

^echo ^content >> file.txt

cm ^undo file.txt

('file.txt'의 로컬 변경을 실행 취소합니다.)

cm ^undo src

(src 디렉터리 및 포함된 파일의 변경사항을 실행 취소합니다.)

cm ^undo src/\*

(src에 포함된 모든 파일 및 디렉터리의 변경사항을 실행 취소합니다.

src에는 영향이 없습니다.)

cm ^undo \*.cs

(현재 디렉터리에서 \*.cs와 일치하는 모든 파일 또는 디렉터리의 변경사항을

실행 취소합니다.)

cm ^undo \*.cs -^r

(현재 디렉터리와 모든 하위 디렉터리에서 \*.cs와 일치하는 모든 파일 또는 디렉터리의 변경사항을

실행 취소합니다.)

cm ^co file1.txt file2.txt

^echo ^content >> file1.txt

cm ^undo --^unchanged

(변경되지 않은 'file2.txt'의 체크아웃을 실행 취소하고 로컬에서 변경된 'file1.txt'를

무시합니다.)

^echo ^content >> file1.txt

^echo ^content >> file2.txt

cm ^co file1.txt

cm ^undo --^checkedout

(체크아웃된 'file1.txt' 파일의 변경사항을 실행 취소하고 'file2.txt'는

체크아웃되지 않았으므로 무시합니다.)

cm ^add file.txt

cm ^undo file.txt

('file.txt' 추가를 실행 취소하여 다시 비공개 파일로 만듭니다.)

^rm file1.txt

^echo ^content >> file2.txt

cm ^add file3.txt

cm ^undo --^deleted --^added \*

('file1.txt' 삭제 및 'file3.txt' 추가를 실행 취소하고 'file2.txt'의 변경사항을

무시합니다.)

== CMD\_DESCRIPTION\_LOCK\_UNLOCK ==

잠금 서버에서 이루어진 항목 잠금을 실행 취소합니다.

== CMD\_USAGE\_LOCK\_UNLOCK ==

사용법:

cm ^lock ^unlock [<리포지토리 서버 사양>] <GUID>[ ...]

리포지토리 서버 사양 리포지토리 서버의 사양. (리포지토리 서버 사양에 관한 자세한 내용을 확인하려면

'cm ^help ^objectspec'을 사용하십시오.)

GUID 잠금을 해제할 항목 GUID의 목록입니다. 공백을 사용하여 GUID를

구분하십시오.

== CMD\_HELP\_LOCK\_UNLOCK ==

비고:

- 이 명령은 지정된 서버를 사용하여 항목의 잠금을 해제합니다.

- 서버가 지정되지 않은 경우, 해당 명령은 현재 워크스페이스에서 서버를

가져오려고 시도합니다.

- 이전 단계에서 서버가 산출되지 않은 경우 서버를 현재 Plastic SCM

클라이언트 구성에서 가져옵니다.

- 서버 관리자만 'cm ^unlock' 명령을 실행할 수 있습니다.

- GUID를 지정하려면 형식이 하이픈으로 구분된 32자릿수 형식이어야

합니다(중괄호를 씌우는 것도 가능):

{00000000-0000-0000-0000-000000000000}

또는 00000000-0000-0000-0000-000000000000

예:

cm ^lock ^unlock 91961b14-3dfe-4062-8c4c-f33a81d201f5

(선택한 항목의 잠금을 실행 취소합니다.)

cm ^lock ^unlock DIGITALIS:8084 2340b4fa-47aa-4d0e-bb00-0311af847865 \

bcb98a61-2f62-4309-9a26-e21a2685e075

('DIGITALIS'라는 잠금 서버에서 선택한 항목의 잠금을 실행 취소합니다.)

cm ^lock ^unlock tardis@cloud 4740c4fa-56af-3dfe-de10-8711fa248635 \

71263c17-5eaf-5271-4d2c-a25f72e101d4

('tardis'라는 클라우드 잠금 서버에서 선택한 항목의 잠금을 실행 취소합니다.)

== CMD\_DESCRIPTION\_UPDATE ==

워크스페이스를 업데이트하고 최근 변경사항을 다운로드합니다.

== CMD\_USAGE\_UPDATE ==

사용법:

cm ^update [<항목 경로> | --^last]

[--^changeset=<체인지 세트 사양>] [--^cloaked] [--^dontmerge] [--^forced]

[--^ignorechanges] [--^override] [--^recursewk] [--^skipchangedcheck]

[--^silent] [--^verbose] [--^xml[=<출력 파일>]] [--^encoding=<이름>]

[--^machinereadable [--^startlineseparator=<구분 기준>]

[--^endlineseparator=<구분 기준>] [--^fieldseparator=<구분 기준>]]

[--^forcedetailedprogress]

항목 경로 업데이트할 경로입니다.

현재 디렉터리에 업데이트를 적용하려면 온점(.)을 사용하십시오.

경로가 지정되지 않은 경우, 현재 워크스페이스 전체가

업데이트됩니다.

--^last 체인지 세트 구성 또는 레이블 구성에서

브랜치 구성으로 워크스페이스 선택기를 변경한 다음

업데이트합니다.

선택기는 체인지 세트 또는 레이블이 속한 브랜치로

변경됩니다.

옵션:

--^changeset 특정 체인지 세트의 워크스페이스를 업데이트합니다.

(체인지 세트 사양에 관한 자세한 내용을 확인하려면

'cm ^help ^objectspec'을 사용하십시오.)

--^cloaked 업데이트 작업에 클로킹된 항목을 포함합니다.

이 옵션이 지정되지 않은 경우 클로킹된 해당 항목은

작업에서 무시됩니다.

--^dontmerge 업데이트 작업 중에 업데이트 병합이 필요한 경우라도

병합을 수행하지 않습니다.

--^forced 선택기에서 지정된 개정에 강제로 항목을

업데이트합니다.

--^ignorechanges 워크스페이스 업데이트 시 대기 중인 변경사항이 감지되는 경우

이와 관련하여 표시된 경고 메시지를

무시합니다.

--^override Plastic SCM 제어를 벗어나 변경된 파일을 재정의합니다.

해당 콘텐츠는 서버 콘텐츠로

덮어쓰게 됩니다.

--^recursewk 현재 경로 내에 있는 모든 워크스페이스를

업데이트합니다. 특정 경로에 포함된 워크스페이스를 모두 업데이트할 때

유용합니다.

--^skipchangedcheck 워크스페이스에 로컬 변경이 있는 경우 업데이트를

검사한 다음 시작합니다. 항상 파일을 수정하기 전에

체크아웃하는 경우 이 검사를 사용하여 작업 속도를

향상할 수 있습니다.

--^silent 어떤 내용도 출력하지 않습니다.

--^verbose 추가 정보를 표시합니다.

--^xml XML 형식의 출력을 표준 출력으로 출력합니다.

출력 파일을 지정할 수 있습니다.

--^encoding '--^xml 옵션과 함께 사용되며 XML 출력에 사용할

인코딩, 즉 UTF-8을 지정합니다.

지원되는 인코딩 및 형식에 대한 표를 확인하려면

http://msdn.microsoft.com/en-us/library/system.text.encoding.aspx에서

MSDN 문서를 확인하십시오.

페이지 마지막 "Name"열에 있습니다.

--^machinereadable 결과를 구문 분석이 쉬운 형식으로 출력합니다.

--^startlineseparator '--^machinereadable' 플래그와 함께 사용되며, 행이 시작되는 방법을

지정합니다.

--^endlineseparator '--^machinereadable' 플래그와 함께 사용되며,

행이 끝나는 방법을 지정합니다.

--^fieldseparator '--^machinereadable' 플래그와 함께 사용되며,

필드를 구분할 방법을 지정합니다.

--^forcedetailedprogress 표준 출력이 리디렉션되더라도 진행도를 강제로

표시합니다.

== CMD\_HELP\_UPDATE ==

비고:

'^update' 명령은 필요한 파일만 다운로드합니다.

해당 명령은 재귀 작업으로 간주합니다.

--^last' 옵션을 사용하면 경로를 지정할 필요가 없습니다.

이 경우에는 현재 작업 디렉터리가 속한 워크스페이스가

업데이트됩니다.

(이 플래그를 지정하면 이전에 선택기가 체인지 세트나 레이블을 가리켰던 경우

워크스페이스 선택기가 브랜치 구성으로 변경될 수 있습니다.

)

예:

cm ^update

(현재 워크스페이스의 모든 항목을 업데이트합니다.)

cm ^update .

(현재 디렉터리와 모든 자식 항목을 업데이트합니다.)

cm ^update . --^forced --^verbose

(모든 개정을 강제로 가져옵니다.)

cm ^update --^last

cm ^update . --^machinereadable --^startlineseparator=">"

(현재 디렉터리를 업데이트하고, 그 결과를 구문 분석이 쉬운 간단한

형식으로 출력하며, 지정된 문자열로 행을

시작합니다.)

== CMD\_DESCRIPTION\_VERSION ==

현재 클라이언트 버전 번호를 표시합니다.

== CMD\_USAGE\_VERSION ==

사용법:

cm ^version

== CMD\_HELP\_VERSION ==

== CMD\_DESCRIPTION\_WHOAMI ==

현재 Plastic SCM 사용자를 표시합니다.

== CMD\_USAGE\_WHOAMI ==

사용법:

cm ^whoami

== CMD\_HELP\_WHOAMI ==

== CMD\_USAGE\_WKTREENODESTATUS ==

사용법:

cm ^wktreenodestatus 경로 1, 경로 2, ...

== CMD\_DESCRIPTION\_WORKSPACE ==

사용자가 워크스페이스를 관리하도록 허용합니다.

== CMD\_USAGE\_WORKSPACE ==

사용법:

cm ^workspace | ^wk <명령> [옵션]

명령:

^list | ^ls

^create | ^mk

^delete | ^rm

^move | ^mv

^rename

각 명령에 대한 자세한 정보는 다음을 실행해 확인하십시오.

cm ^workspace <명령> --^usage

cm ^workspace <명령> --^help

== CMD\_HELP\_WORKSPACE ==

예:

cm ^workspace ^create myWorkspace 워크스페이스 경로

cm ^workspace ^list

cm ^workspace ^delete myWorkspace

== CMD\_DESCRIPTION\_WORKSPACE\_CREATE ==

새 워크스페이스를 생성합니다.

== CMD\_USAGE\_WORKSPACE\_CREATE ==

사용법:

cm ^workspace | ^wk [^create | ^mk] <워크스페이스 이름> <워크스페이스 경로> [<리포지토리 사양>]

[--^selector[=<선택기 파일>]

(새 워크스페이스를 생성합니다.)

cm ^workspace | ^wk [^create | ^mk] <워크스페이스 이름> <워크스페이스 경로> --^dynamic --^tree=[<트리>]

(동적 워크스페이스를 생성합니다. 이 기능은 아직 실험 기능이며 Windows에서만

사용 가능합니다.)

워크스페이스 이름 새 워크스페이스 이름입니다.

워크스페이스 경로 새 워크스페이스의 경로입니다.

리포지토리 사양 지정된 리포지토리가 포함된 새 워크스페이스를 만듭니다.

리포지토리 사양은 'cm ^help ^objectspec'을 확인하십시오.

옵션:

--^selector 새 워크스페이스의 선택기를 편집합니다.

선택기 파일이 지정된 경우 지정된 파일로부터

새 워크스페이스 선택기를 선택합니다.

--^dynamic 동적 워크스페이스를 생성합니다. 이 기능은 아직

실험 기능이며 Windows에서만 사용 가능합니다.

이 플래그를 지정하려면 --^tree 매개 변수를 사용해야 합니다.

--^tree '--^dynamic' 플래그와 함께 사용되며, 동적 워크스페이스를

로드할 최초 지점을 지정합니다. 브랜치,

체인지 세트 또는 레이블 사양 등이 지점이 될 수 있습니다.

워크스페이스는 나중에 해당 사양의 리포지토리를

사용합니다(사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을 사용하십시오.).

== CMD\_HELP\_WORKSPACE\_CREATE ==

비고:

- 워크스페이스는 로컬 파일 시스템에 매핑된 리포지토리의 뷰입니다.

워크스페이스 선택기는 워크스페이스 내용을 지정하는 규칙을 정의합니다.

'cm ^showselector'를 사용하여 워크스페이스 선택기를 표시하거나 'cm ^setselector'를

사용하여 이를 수정하십시오.

- 리포지토리 사양과 '--^selector'가 모두 지정되지 않은 경우 워크스페이스는

client.conf 파일에서 구성된 서버의 첫 번째 리포지토리(알파벳 기준)를

사용하도록 자동으로 구성됩니다.

- 동적 워크스페이스는 실험 기능(Windows만 해당)이며

사용하려면 plasticfs.exe 프로그램을 실행해야 합니다.

예:

cm ^workspace ^create myworkspace c:\workspace

cm ^wk ^mk myworkspace /home/john/plastic\_view

(Windows와 Linux 각각에 'myworkspace' 워크스페이스를 생성합니다.)

cm ^wk mywktest c:\wks\wktest --^selector=myselector.txt

('myselector.txt' 파일의 선택기를 사용하여 'mywktest' 워크스페이스를 생성합니다.)

cm ^wk mywkprj c:\wks\wkprj myrep@^repserver:localhost:8084

(선택한 리포지토리가 포함된 'mywkprj' 워크스페이스를 만듭니다.)

cm ^wk mywkprj c:\dynwks\mywkprj --^dynamic --^tree=^br:/main@myrep@localhost:8084

('myrep@localhost:8084' 리포지토리가 포함된

'mywkprj' 동적 워크스페이스를 생성합니다. 해당 워크스페이스가 처음으로 마운트되면 '^br:/main'을 가리킵니다.)

== CMD\_DESCRIPTION\_WORKSPACE\_DELETE ==

워크스페이스를 삭제합니다.

== CMD\_USAGE\_WORKSPACE\_DELETE ==

사용법:

cm ^workspace | ^wk ^delete | ^rm [<워크스페이스 경로> | <워크스페이스 사양>] [--^keepmetadata]

워크스페이스 경로 삭제할 워크스페이스의 경로입니다.

워크스페이스 사양 삭제할 워크스페이스의 사양입니다. (사양에 관한

자세한 내용을 확인하려면 'cm ^help ^objectspec'을 사용하십시오.)

옵션:

--^keepmetadata .plastic 폴더에 있는 메타데이터 파일을 삭제하지

않습니다.

== CMD\_HELP\_WORKSPACE\_DELETE ==

비고:

이 명령은 경로 또는 사양으로 지정된 워크스페이스를 삭제합니다.

인수가 지정되지 않으면 현재 워크스페이스로 간주합니다.

예:

cm ^workspace ^delete

(현재 워크스페이스를 제거합니다.)

cm ^wk ^delete c:\workspace

cm ^workspace rm /home/danipen/wks

cm ^wk ^rm ^wk:MiWorkspace

cm ^wk ^rm ^wk:MiWorkspace@DIGITALIS

== CMD\_DESCRIPTION\_WORKSPACE\_LIST ==

워크스페이스를 나열합니다.

== CMD\_USAGE\_WORKSPACE\_LIST ==

사용법:

cm ^workspace | ^wk [^list | ^ls] [--^format=<문자열 형식>]

옵션:

--^format 특정 형식으로 출력 메시지를 가져옵니다. 자세한 내용은

비고를 확인하십시오.

== CMD\_HELP\_WORKSPACE\_LIST ==

비고:

이 명령은 출력을 표시할 형식 문자열을 허용합니다.

이 명령의 출력 매개 변수는 다음과 같습니다.

{0} | {^wkname} 워크스페이스 이름.

{1} | {^machine} 클라이언트 기기 이름.

{2} | {^path} 워크스페이스 경로.

{3} | {^wkid} 워크스페이스 고유 식별자.

{4} | {^wkspec} 다음 형식을 사용하는 고유 식별자:

'wkname@machine'.

{^tab} 탭 공백을 삽입합니다.

{^newline} 새 행을 삽입합니다.

예:

cm ^wk

(모든 워크스페이스를 나열합니다.)

cm ^workspace ^list --^format={0}#{3,40}

cm ^workspace ^list --^format={^wkname}#{^wkid,40}

(모든 워크스페이스를 나열하고, 각각 40자 한도 내에서 워크스페이스 이름, # 기호,

워크스페이스 GUID 필드 순으로 왼쪽 정렬하여 표시합니다.)

cm ^wk --^format="경로 {2}의 워크스페이스 {0}"

cm ^wk --^format="경로 {^path}의 워크스페이스 {^wkname}"

(모든 워크스페이스를 나열하고 일정 형식의 문자열로 결과를 표시합니다.)

== CMD\_DESCRIPTION\_WORKSPACE\_MOVE ==

워크스페이스를 이동합니다.

== CMD\_USAGE\_WORKSPACE\_MOVE ==

사용법:

cm ^workspace | ^wk ^move | ^mv [<워크스페이스 사양>] <새 경로>

옵션:

워크스페이스 사양 이동할 워크스페이스의 사양입니다. (사양에 관한

자세한 내용을 확인하려면 'cm ^help ^objectspec'을 사용하십시오.)

새 경로 워크스페이스가 여기로 이동합니다.

== CMD\_HELP\_WORKSPACE\_MOVE ==

비고:

이 명령은 사용자가 워크스페이스를 디스크의 다른 위치로 이동하도록 허용합니다.

예:

cm ^workspace ^move myWorkspace \new\workspaceDirectory

('myWorkspace'를 지정된 위치로 이동합니다.)

cm ^wk ^mv c:\users\maria\wkspaces\newlocation

(현재 워크스페이스를 새 위치로 이동합니다.)

== CMD\_DESCRIPTION\_WORKSPACE\_RENAME ==

워크스페이스 이름을 변경합니다.

== CMD\_USAGE\_WORKSPACE\_RENAME ==

사용법:

cm ^workspace | ^wk ^rename [<워크스페이스 이름>] <새 이름>

워크스페이스 이름 변경할 워크스페이스의 이름입니다.

새 이름 새 워크스페이스 이름입니다.

== CMD\_HELP\_WORKSPACE\_RENAME ==

비고:

이 명령은 워크스페이스 이름을 변경합니다.

워크스페이스 이름이 지정되지 않은 경우, 현재 워크스페이스가 사용됩니다.

예:

cm ^workspace ^rename mywk1 wk2

(워크스페이스 이름을 'mywk1'에서 'wk2'로 변경합니다.)

cm ^wk ^rename newname

(현재 워크스페이스 이름을 'newname'으로 변경합니다.)

== CMD\_DESCRIPTION\_WORKSPACESTATUS ==

워크스페이스의 변경사항을 표시합니다.

== CMD\_USAGE\_WORKSPACESTATUS ==

사용법:

cm ^status [<워크스페이스 경로>] [--^changelist[=<이름>] | --^changelists] [--^cutignored]

[ --^header] [ --^noheader] [ --^nomergesinfo] [ --^head]

[--^short] [--^symlink] [ --^dirwithchanges] [--^xml[=<출력 파일>]]

[--^encoding=<이름>] [ --^wrp | --^wkrootrelativepaths]

[--^fullpaths | --^fp] [<레거시 옵션>] [<검색 유형>[ ...]]

[--^machinereadable [--^startlineseparator=구분 기준]

[--^endlineseparator=구분 기준] [--^fieldseparator=구분 기준]]

옵션:

워크스페이스 경로 검색을 수행할 워크스페이스

경로입니다.

--^changelist 선택한 변경사항 목록의 변경사항을 표시합니다.

--^changelists 클라이언트 변경사항 목록의 변경사항을 그룹으로 표시합니다.

--^cutignored 무시된 디렉터리의 내용을 건너뜁니다.

'--^ignored' 검색 유형이 필요합니다. 자세한 내용은

검색 유형 섹션을 참조하십시오.

--^header 워크스페이스 상태만 출력합니다.

--^noheader 수정된 항목에 대한 검색 결과만 출력합니다.

--^nomergesinfo 변경사항에 대한 병합 정보를 출력하지 않습니다.

--^head 브랜치의 마지막 체인지 세트 상태를 출력합니다.

--^short 변경사항이 있는 경로만 나열합니다.

--^symlink 대상이 아닌 기호 링크에 작업을

적용합니다.

--^dirwithchanges 내부에 변경사항이 있는 디렉터리를 표시합니다

(내부에 있는 추가, 이동, 제거된 항목).

--^xml XML 형식의 출력을 표준 출력으로 출력합니다.

출력 파일을 지정할 수 있습니다.

--^encoding --^xml 옵션과 함께 사용되며 XML 출력에 사용할

인코딩, 즉 UTF-8을 지정합니다.

지원되는 인코딩 및 형식에 대한 표를 확인하려면

http://msdn.microsoft.com/en-us/library/system.text.encoding.aspx에서

MSDN 문서를 확인하십시오.

페이지 마지막 "Name"열에 있습니다.

--^wrp 현재 디렉터리의 상대 경로 대신 워크스페이스의

루트 상대 경로를 출력합니다.

--^fullpaths, --^fp 절대 경로를 강제로 출력하고 다른 모든 경로 출력 설정을

재정의합니다.

--^machinereadable 결과를 구문 분석이 쉬운 형식으로 출력합니다.

--^startlineseparator '--^machinereadable' 플래그와 함께 사용되며, 행이 시작되는 방법을

지정합니다.

--^endlineseparator '--^machinereadable' 플래그와 함께 사용되며,

행이 끝나는 방법을 지정합니다.

--^fieldseparator '--^machinereadable' 플래그와 함께 사용되며,

필드를 구분할 방법을 지정합니다.

레거시 옵션:

--^cset 워크스페이스 상태를 레거시 형식으로 출력합니다.

--^compact 워크스페이스 상태 및 변경사항 목록을 레거시 형식으로

출력합니다.

--^noheaders '--^compact' 플래그와 함께 사용하면 변경사항 목록 헤더가

출력되지 않습니다. (새 변경사항 목록 형식에는

적용되지 않습니다.)

검색 유형:

--^added 추가된 항목을 출력합니다.

--^checkout 체크아웃된 항목을 출력합니다.

--^changed 변경된 항목을 출력합니다.

--^copied 복사된 항목을 출력합니다.

--^replaced 교체된 항목을 출력합니다.

--^deleted 삭제된 항목을 출력합니다.

--^localdeleted 로컬에서 삭제된 항목을 출력합니다.

--^moved 이동된 항목을 출력합니다.

--^localmoved 로컬에서 이동한 항목을 출력합니다.

--^percentofsimilarity=<value> 두 파일을 동일 항목으로 간주하는 데 필요한

해당 파일 간의 유사도입니다. 로컬

이동 항목 검색에 사용됩니다. 기본값은

20%입니다.

--^txtsameext 이동한 항목을 검색하는 중에 확장자가

동일한 텍스트 파일만 유사도 콘텐츠 판정

프로세스에

포함됩니다. 기본적으로 모든

텍스트 파일이 처리됩니다.

--^binanyext 이동한 항목을 검색하는 중에 모든 바이너리 파일이

유사도 콘텐츠 판정 프로세스에

포함됩니다. 기본적으로

확장자가 동일한 바이너리 파일만

처리됩니다.

--^private 제어되지 않는 항목을 출력합니다.

--^ignored 무시된 항목을 출력합니다.

--^hiddenchanged 숨겨진 변경된 파일을 출력합니다. (

'--^changed' 포함)

--^cloaked 클로킹된 항목을 출력합니다.

--^controlledchanged 이 플래그는 다음 옵션을 대체합니다.

'--^added', '--^checkout', '--^copied',

'--^replaced', '--^deleted', '--^moved'.

--^all 이 플래그는 다음 매개 변수를 대체합니다.

'--^controlledchanged', '--^changed',

'--^localdeleted', '--^localmoved', '--^private'.

== CMD\_HELP\_WORKSPACESTATUS ==

비고:

'^status' 명령은 워크스페이스에 로드된 체인지 세트를 출력하며

워크스페이스 내부에서 변경된 요소를 가져옵니다.

이 명령을 사용하여 워크스페이스에서 대기 중인 변경사항을 표시할 수 있습니다.

수정할 수 있는 변경 유형은 해당 명령 매개 변수를 사용하여 수정할 수

있습니다. 기본적으로 제어 및 로컬 여부에 상관없이 모든 변경사항이

표시됩니다.

두 요소가 동일 항목인지 판정하기 위해 로컬 이동 항목 검색 시

유사도 매개 변수 '--^percentofsimilarity'(-^p)가 사용됩니다.

기본값은 20%지만 조정 가능합니다.

클라이언트 변경사항 목록을 기준으로 워크스페이스 변경사항을 그룹으로 표시할 수 있습니다.

'^default' 변경사항 목록에는 다른 변경사항 목록에 포함되지 않은 변경사항이

포함됩니다. 하지만 기본 변경사항 목록에서 표시하는 변경사항은

지정된 검색 유형 플래그에 따라 달라집니다.

변경사항 목록을 기준으로 변경사항을 그룹으로 표시하려면 제어되는 변경사항도

표시해야 합니다.('^added', '^checkout', '^copied',

'^replaced', '^deleted' 또는 '^moved'와 동등한 상태인 항목). 따라서 변경 사항이 표시되면

'--^controlledchanged' 옵션이 자동으로 활성화됩니다.

XML 출력의 기본 인코딩은 UTF-8입니다.

기본적으로 이 명령은 '--^machinereadable' 또는 '--^short' 플래그가

지정되지 않은 경우 현재 디렉터리의 상대 경로를 출력합니다. 이 중

하나라도 지정되면 해당 명령은 절대 경로를 출력합니다.

'--^xml' 플래그가 지정된 경우 워크스페이스의 루트 상대 경로가

출력됩니다('--^fp' 플래그도 지정되지 않는 한

절대 경로가 대신 출력됨).

예:

cm ^status

(무시된 항목을 제외하고, 워크스페이스의 작업 중인 체인지 세트와 변경된

모든 항목 유형도 출력합니다.)

cm ^status --^controlledchanged

(작업 중인 체인지 세트와 체크아웃, 추가, 복사, 교체, 삭제 및 이동한 항목도

출력합니다.)

cm ^status --^added

(작업 중인 체인지 세트와 워크스페이스 내부에 추가된 항목만 출력합니다.)

cm ^status c:\workspaceLocation\code\client --^added

(지정된 경로 아래에 있는 작업 중인 체인지 세트와 추가된 항목을 재귀적으로

출력합니다.)

cm ^status --^changelists

cm ^status --^changelist

(클라이언트 변경사항 목록을 기준으로 모든 워크스페이스 변경사항을 그룹으로 표시합니다.)

cm ^status --^changelist=pending\_to\_review

('pending\_to\_review'라는 변경사항 목록의 변경사항을 표시합니다.)

cm ^status --^changelist=default --^private

('default' 변경사항 목록의 변경사항을 표시하고, 해당하는 경우 제어되는 변경사항과 함께

비공개 항목을 표시합니다.)

cm ^status --^short --^changelist=pending\_to\_review | cm ^checkin -

('pending\_to\_review' 변경사항 목록의 변경사항을 체크인합니다.)

cm ^status C:\workspaceLocation --^xml=output.xml

(output.xml에서 UTF-8을 사용하여 XML 형식의 상태 정보를

가져옵니다.)

cm ^status --^ignored

(무시된 항목을 모두 표시합니다.)

출력:

/main@myrepo@local (^cs:2 - ^head)

^Added

상태 크기 마지막 수정 경로

^Ignored 0바이트 19초 전 out\app.exe

^Ignored 48초 전 src

^Ignored 0바이트 48초 전 src\version.c

cm ^status --^ignored --^cutignored

(부모 디렉터리가 무시되지 않은 무시된 파일과 내용이 무시되지 않은 무시된

디렉터리를 표시합니다.)

출력:

/main@myrepo@local (^cs:2 - ^head)

^Added

상태 크기 마지막 수정 경로

^Ignored 0바이트 19초 전 out\app.exe

^Ignored 48초 전 src

== CMD\_DESCRIPTION\_XLINK ==

Xlink 세부사항을 생성하거나, 편집하거나, 표시합니다.

== CMD\_USAGE\_XLINK ==

사용법:

cm ^xlink [-^w] [-^rs] <Xlink 경로> / (<체인지 세트 사양> | <레이블 사양> | <브랜치 사양)>

[<확장 규칙>[ ...]]

(Xlink를 생성합니다.)

cm ^xlink [-^rs] <Xlink 경로> /<상대 경로> (<체인지 세트 사양> | <레이블 사양> | <브랜치 사양>)

[<확장 규칙>[ ...]]

(기본 루트 / 대신에 /<상대 경로>를 가리키는 읽기 전용 부분 Xliink를

생성합니다.)

cm ^xlink -^e <Xlink 경로> (<체인지 세트 사양> | <레이블 사양> | <브랜치 사양>)

(Xlink를 편집하여 대상 사양을 변경합니다.)

cm ^xlink -^s|--^show <Xlink 경로>

(확장 규칙을 포함한 Xlink 정보를 표시합니다.)

cm ^xlink -^ar|--^addrules <Xlink 경로> <확장 규칙>[ ...]

(주어진 확장 규칙을 Xlink에 추가합니다).

cm ^xlink -^dr|--^deleterules <Xlink 경로> <확장 규칙>[ ...]

(Xlink에서 주어진 확장 규칙을 제거합니다).

Xlink 경로 연결된 리포지토리를 마운트할 예정이거나(Xlink 생성 시),

마운트되어 있는(Xlink 편집 시)

현재 워크스페이스의 디렉터리입니다.

체인지 세트 사양 원격 리포지토리의 전체 대상 체인지 세트

사양입니다.

이는 연결된 리포지토리의 워크스페이스에 어떤 버전과

브랜치를 로드할지 결정합니다.

(체인지 세트 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을

사용하십시오.)

레이블 사양 원격 리포지토리의 전체 레이블 사양입니다.

(레이블 사양에 관한 자세한 내용을 확인하려면 'cm ^help ^objectspec'을

사용하십시오.)

브랜치 사양 원격 리포지토리의 전체 브랜치 사양입니다.

이는 지정된 브랜치가 가리키는 현재 체인지 세트를

사용합니다. (브랜치 사양에 관한 자세한 내용을 확인하려면

'cm ^help ^objectspec'을 사용하십시오.)

-^e 기존 Xlink를 편집하여 대상 체인지 세트 사양을

변경합니다.

-^s | --^show 선택한 Xlink에 관한 정보를 표시합니다.

-^ar | --^addrules 하나 이상의 확장 규칙을 선택한 Xlink에 추가합니다.

-^dr | --^deleterules 선택한 Xlink에서 하나 이상의 확장 규칙을

제거합니다.

확장 규칙 하나 이상의 확장 규칙을 지정합니다. 각 확장 규칙은

브랜치와 대상 브랜치의 쌍입니다.

^br:/main/fix-^br:/main/develop/fix

옵션:

-^w Xlink가 쓰기 가능하다는 것을 나타냅니다. 즉 브랜치 자동 확장을 통해

대상 리포지토리의 내용을

수정할 수 있습니다.

-^rs 관련 서버입니다. 이 옵션을 통해 리포지토리 서버로부터

독립적인 관련 Xlink를 생성할 수 있습니다. 이런 식으로

여러 서버에 있는 복제된 리포지토리에서 생성된

Xlink가 자동으로 식별됩니다.

== CMD\_HELP\_XLINK ==

비고:

이 명령은 주어진 체인지 세트에 대한 Xlink를 생성합니다. 기본적으로 읽기 전용

Xlink가 생성됩니다. 즉, Xlink 내부의 워크스페이스에 로드된 내용은

수정할 수 없습니다. Xlink 내용을 변경하려면

쓰기 가능한 Xlink를 대신 생성하십시오('-^w' 옵션 사용).

Xlink의 대상 체인지 세트를 편집할 때 간단한 명령 구문을

사용할 수 있습니다. 이 방법에서 필요한 매개 변수는 새 대상

체인지 세트뿐입니다. Xlink의 나머지 매개 변수는 수정되지

않습니다.

브랜치 자동 확장:

쓰기 가능한 Xlink 리포지토리('-^w' 옵션)가 변경되면 대상 리포지토리에

새 브랜치를 생성해야 합니다. 새 브랜치의 이름은

최상위 리포지토리에서 정의된 체크아웃 브랜치에 따라

달라집니다. 사용할 브랜치 이름을 결정하기 위해 다음 규칙이 적용됩니다.

1) 전체 이름이 동일한 브랜치가 대상 리포지토리에 존재하는지 확인하기 위해

검사가 이루어집니다.

- 존재하는 경우 해당 브랜치가 체크아웃 브랜치로 사용됩니다.

- 존재하지 않는 경우 브랜치 이름이 다음과 같은 방법으로 구성됩니다.

- 대상 Xlink 체인지 세트의 브랜치 이름 + 체크아웃 브랜치의

짧은 이름(마지막 부분).

- 이 브랜치가 존재하는 경우 해당 브랜치가 체크아웃 브랜치로 사용됩니다.

- 존재하지 않으면 브랜치가 생성되고 브랜치 베이스가 Xlink 체인지 세트에

설정됩니다.

2) Xlink 리포지토리의 새 체인지 세트를 가리키는 부모 리포지토리의 브랜치에

Xlink의 새 버전이 생성됩니다.

마지막으로 완전한 Xlink 구조에 적절한 버전의

최근 변경사항이 적용됩니다.

예:

cm ^xlink code\firstrepo / 1@first@localhost:8084

('first' 리포지토리의 '1' 체인지 세트를 마운트할 현재 워크스페이스의

'firstrepo' 폴더에 Xlink를 생성합니다.)

cm ^xlink opengl\include /includes/opengl 1627@includes@localhost:8087

('includes' 리포지토리에 있는 '1627' 체인지 세트의 '/includes/opengl' 경로를

루트로 마운트할 현재 워크스페이스의 'opengl\include' 디렉터리에

읽기 전용 부분 Xlink를 생성합니다.) 즉, '/includes/opengl' 내부의

어떤 항목이든 'opengl\include'에 마운트되지만

리포지토리의 나머지 항목은 무시됩니다.)

cm ^xlink -^w -^rs code\secondrepo / ^lb:LB001@second@localhost:8084

('second' 리포지토리의 'LB001' 레이블을 마운트할 현재 워크스페이스의

'secondrepo' 폴더에 쓰기 가능한 관련 Xlink를

생성합니다.)

cm ^xlink code\thirdrepo / 3@third@localhost:8087 ^br:/main-^br:/main/scm003

('third' 리포지토리의 '3' 체인지 세트를 마운트할 현재 워크스페이스의

'thirdrepo' 폴더에 Xlink를 생성합니다.)

cm ^xlink -^e code\secondrepo ^br:/main/task1234@second@localhost:8084

('code\secondrepo' Xlink를 편집하여 'second' 리포지토리의

'main/task1234' 브랜치를 연결해 대상 리포지토리를 변경합니다.)

cm ^xlink --^show code\thirdrepo

(존재하는 경우, 확장 규칙을 포함하여 'code\thirdrepo' Xlink의 정보를

표시합니다).

cm ^xlink -^ar code\secondrepo ^br:/main-^br:/main/develop ^br:/main/fix-^br:/main/develop/fix

(두 가지 확장 규칙을 'code\secondrepo' Xlink에 추가합니다.)

cm ^xlink -^dr code\secondrepo ^br:/main/fix-^br:/main/develop/fix

('code\secondrepo' Xlink에서 확장 규칙을 삭제합니다).

== CMD\_USAGE\_AUTOCOMPLETE ==

사용법:

cm ^autocomplete ^install

(셸에 'cm' 명령 완성을 설치합니다.)

cm ^autocomplete ^uninstall

(셸에서 'cm' 명령 완성을 제거합니다.)

cm ^autocomplete --^line <셸 행> --^position <커서 위치>

('커서 위치'에 삽입할 자동 완성 추천사항을

'셸 행'에 반환합니다. 이 명령은 최종 사용자가 사용하도록 고안되지 않았지만,

원하는 셸로 자동 완성 지원을 확장하려는 경우를 위해

문서화되어 있습니다.)

셸 행 자동 완성을 요청한 시점에

사용자가 셸에 작성한 행입니다.

Bash에서는 셸 행이 COMP\_LINE 환경 변수에 있습니다.

PowerShell에서는 셸 행이 $wordToComplete 변수에 있습니다.

커서 위치 자동 완성을 요청한 시점의 커서

위치입니다.

Bash에서는 커서 위치가 COMP\_LINE 환경 변수에 있습니다.

PowerShell에서는 커서 위치가 $cursorPosition 변수에 있습니다.