



Version 7.0 Trigger Writers Guide



Audience	3
Introduction.....	4
Customer Support	5
Supported Platforms.....	5
ClearWeb Client-Side Terminal Output	6
ClearWeb Client-Side Interactive Dialogs.....	7
CLEARWEB - QUESTION DIALOGS	7
CLEARWEB - PROMPT DIALOGS	8
CLEARWEB – SINGLE-SELECTION DIALOGS.....	9
CLEARWEB – MULTI-SELECTION DIALOGS.....	9
CLEARWEB – ADVISE DIALOGS	10
Associated ClearWeb Tools.....	11
CLEARWEB_WRITE	11
CLEARWEB_OUT.....	13
CLEARWEB_DIALOG	15
Select Dialog Example	16
Prompt Dialog Examples	16
Question Dialog Examples	17
Advise Dialog Example	18
ClearCase Remote Client (CRC) support	18



Audience

The **ClearWeb Trigger Writers Guide** is intended as a reference for those responsible for writing web-enabled triggers and policy for ClearCase Installations. Oft times these individuals are different than those responsible for day-to-day ClearWeb Administration. It is expected that this document is not made highly available and that copies might reside primarily with CM personnel responsible for creating and maintaining policy/triggers for ClearCase Web users.

For installation information, refer to the **ClearWeb Installation Guide**. For day-to-day Client User information, refer to the **ClearWeb User Guide**. For ClearWeb Administration issues, refer to the **ClearWeb Administration Guide**.

Introduction

ClearWeb is a custom add-on to Rational Software's ClearCase product. **ClearWeb allows for triggers written for ClearCase policy to extend to users of the ClearCase web interface.** The native ClearCase for web users interface does not support interactive triggers. Interactive triggers are those that write to stdout or have any dialogs displaying to or requesting information from the ClearCase user. When Native ClearCase encounters a write to stdout or stderr from a users trigger script, the ClearCase web user is shown a non-descript error message indicating the following:

- the associated trigger failed for reasons unknown
- the trigger is possibly “interactive” and unsupported.

ClearWeb allows for any writes to stdout or stderr from your trigger scripts to be displayed to the ClearCase web-client while allowing for interactive dialogs (dialogs that request information from the user). Due to the fact that ClearWeb is user sensitive, scripts are triggered using clearprompt or similarly functioning web dialogs, depending on whether the user is a standard (non-web) user *or* a ClearCase web user. Upon request, ClearWeb further enables organizations to add triggers for specialized purposes, while allowing for unique interactive policy based on the type of user (web user or non-web user). Interactive dialogs allow the client to answer questions or make selections to the server.

The Native ClearCase Web interface provides no per/user security in that all VOBs that are within the default region of the ClearCase Web Server are accessible for all users. In order to provide different VOB access for different users multiple ClearCase Web Servers must be purchased.

ClearWeb allows for a user to VOB matrix to be created so that the ClearWeb Administrator can limit the VOBs that their web users have access to from the web (this also extends to directories within VOBs). Now a single ClearCase Web Server can serve many user/user groups with different VOB/directory access rights.

The Native ClearCase Web interface provides ability to limit when a user can access the ClearCase Web Interface; therefore there is no ability to limit access to users during scheduled VOB/Web server maintenance, backups or to just the times when the support organization can support the web users. **ClearWeb allows a user/time matrix to be created so that the ClearWeb Administrator can limit when a user or group of users can access the ClearCase Web Interface.**

The Native ClearCase Web interface only supports a limited set of cleartool commands (checkout, checkin, uncheckout, update) **ClearWeb extends this by allowing for over 120 more cleartool commands than the native ClearCase Web interface and an unlimited number of non-ClearCase commands.**

ClearWeb facilitates communication between geographically disjoint development groups by providing a common communications platform between these groups (the ClearCase Web Interface). ClearWeb users can communicate with their ClearWeb peers (within the company – or other companies) to ask or answer questions (ClearWeb related or not) right from the ClearCase Web interface.



Additionally, ClearWeb allows for groups to make other company applications or data (i.e. company FAQs, company timesheet applications, etc.) available from the ClearCase Web Interface. ClearWeb also allows the popular local ClearCase/ClearQuest integration to work over the web just as it does for local ClearCase users.

ClearWeb requires no installation on the client side. If a user currently accesses ClearCase data from the web interface, they will continue to do so in the same way. The primary difference is the way in which the server communicates information to the client via stdout, stderr and dialogs windows. Additionally, the client will have the ability to answer questions or make selections through interactive dialogs as well as the ability to perform many more ClearCase and non-ClearCase commands.

Due to the complete integration of ClearWeb with ClearTrigger, all triggers are automatically web-ready for both stdout or stderr and ClearTrigger created dialogs. Simply install and configure ClearWeb on the server to get started!

Customer Support

To obtain additional information on ClearWeb or services offered by A Better Solution, Inc., visit our web site at www.abs-consulting.com. To report problems with the ClearWeb software or documentation, please submit a problem report via email to clearweb_team@abs-consulting.com.

If you purchased ClearWeb from ClearVision (the only licensed reseller of ClearWeb in the European regions) then you should contact support@clearvision-cm.com for support.

Supported Platforms

ClearWeb integrate with ClearCase web installations of version 5.0 and higher. A single installation of ClearWeb typically serves an entire organization. The system requirements for ClearWeb are consistent with those used to run ClearCase versions 5.0 and later. The following operating systems are supported for ClearWeb:

❖ WINDOWS:

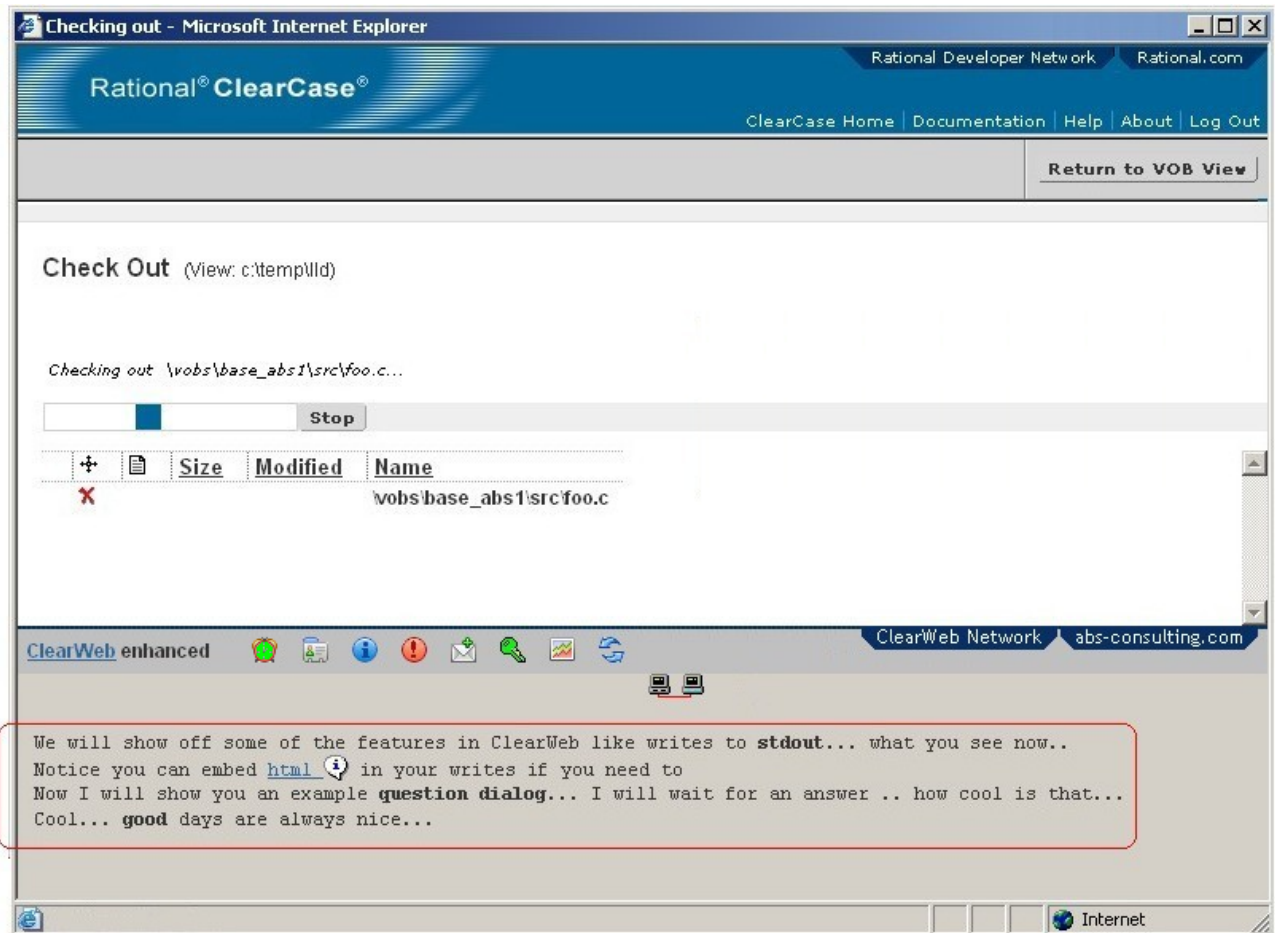
- **Windows 9x**
- **Windows NT**
- **Windows 2000 and higher**



❖ UNIX:

- **Solaris**
- **SunOS**
- **HP-UX**
- **RedHat Linux**
- **True64**
- **AIX**
- **IRIX**

ClearWeb Client-Side Terminal Output

When using ClearWeb, stdout and stderr messages are sent uninterrupted to the terminal for standard (non-web) ClearCase users; for web-users this output is displayed in the user's web interface in a scrolling (if needed) frame as depicted below. To enable this for your triggers, you will make use of the **clearweb_write.exe** or **clearweb_out.exe** programs distributed with ClearWeb.



ClearCase is still responsible for informing you when the command complete, but stdout and stderr are redirected here as well and no longer cause the command to fail. The Icon above the output will either be  indicating that ClearWeb is polling the server for input and dialogs to display or a this icon  which indicates that you have elected to stop polling. You should stop the polling when for example you have to leave you desk for a bit, but wish to stay in this screen. (i.e. to find someone to explain or to show a screen to). To toggle polling just select the icon.

ClearWeb Client-Side Interactive Dialogs

When using ClearWeb, interactive dialogs are available for both standard ClearCase users and ClearCase web-users. To enable this for scripts, use the **clearweb_dialog.exe** program distributed with ClearWeb. Users may be prompted for information from within your triggers that will in turn be available to your triggers. Examples of standard dialogs (displayed for standard users) and web-dialogs (displayed for web users) are displayed in the following sections

ClearWeb - Question Dialogs

A sample **Question Dialog** for a ClearCase non-web user is shown in Fig. 1. The same dialog for a web user is shown in Fig. 2.

Fig. 1

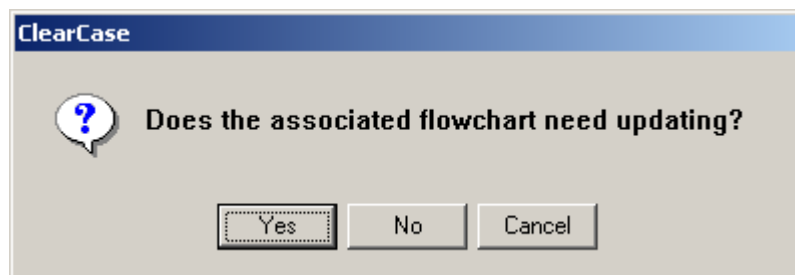


Fig. 2



ClearWeb - Prompt Dialogs

A sample **Prompt Dialog** for a ClearCase non-web user is shown in Fig. 3. The same dialog for a web user is shown in Fig. 4.

Fig. 3

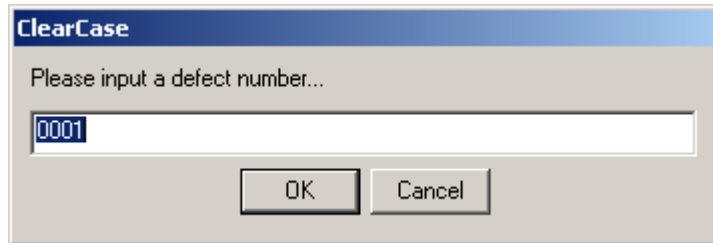
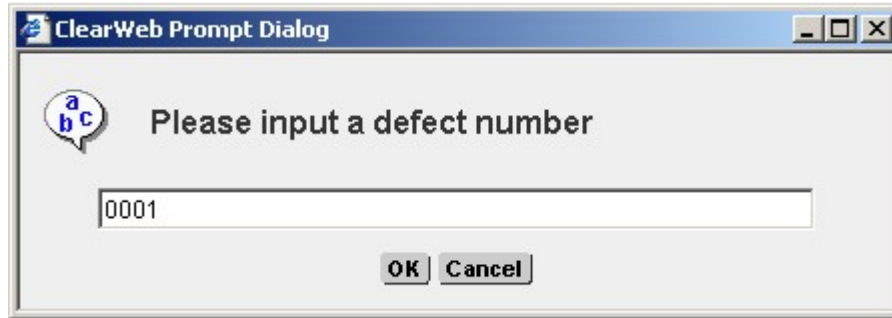


Fig. 4



ClearWeb – Single-Selection Dialogs

A sample **Single-Selection Dialog** for a ClearCase non-web user is shown in Fig. 5. The same dialog for a web user is shown in Fig. 6.

Fig. 5

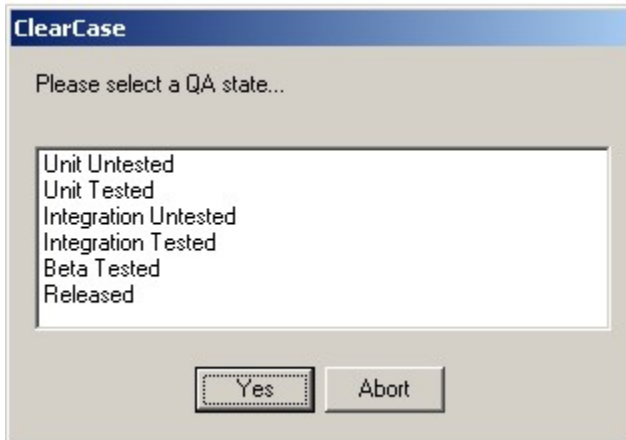


Fig. 6



ClearWeb – Multi-Selection Dialogs

A sample **Multi-Selection Dialog** for a ClearCase non-web user is shown in Fig. 7. The same dialog for a web user is shown in Fig. 8.

Fig. 7

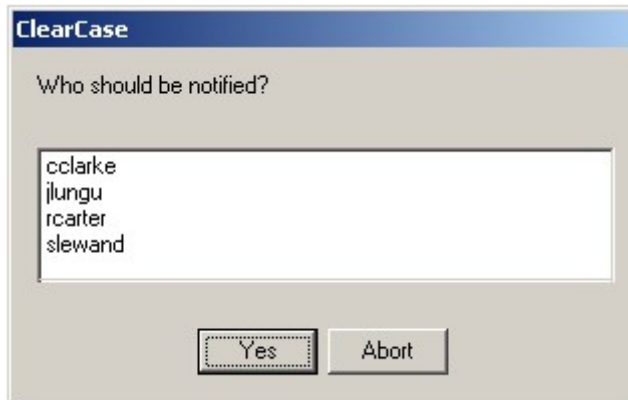
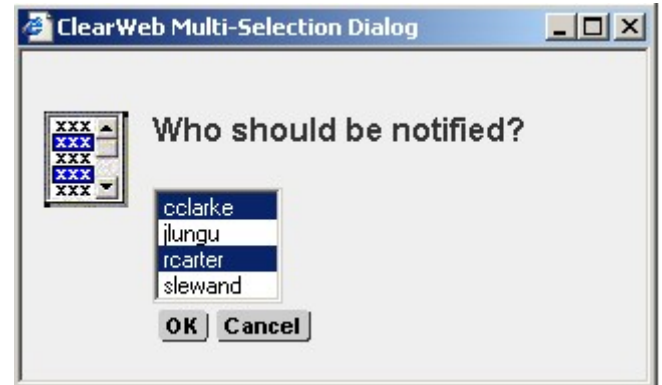


Fig. 8



ClearWeb – Advise Dialogs

A sample **Advise Dialog** for a ClearCase non-web user is shown in Fig. 9. The same dialog for a web user is shown in Fig. 10.

Fig. 9



Fig. 10





Associated ClearWeb Tools

ClearWeb is distributed with three executables to allow you to take advantage of the ClearWeb features from within your own previously written triggers. Each is described below:

- [clearweb_write.exe](#)
- [clearweb_out.exe](#)
- [clearweb_dialog.exe](#)

If you are using ClearTrigger for trigger management, only **clearweb_dialog.exe** is required in limited form and function.

clearweb_write

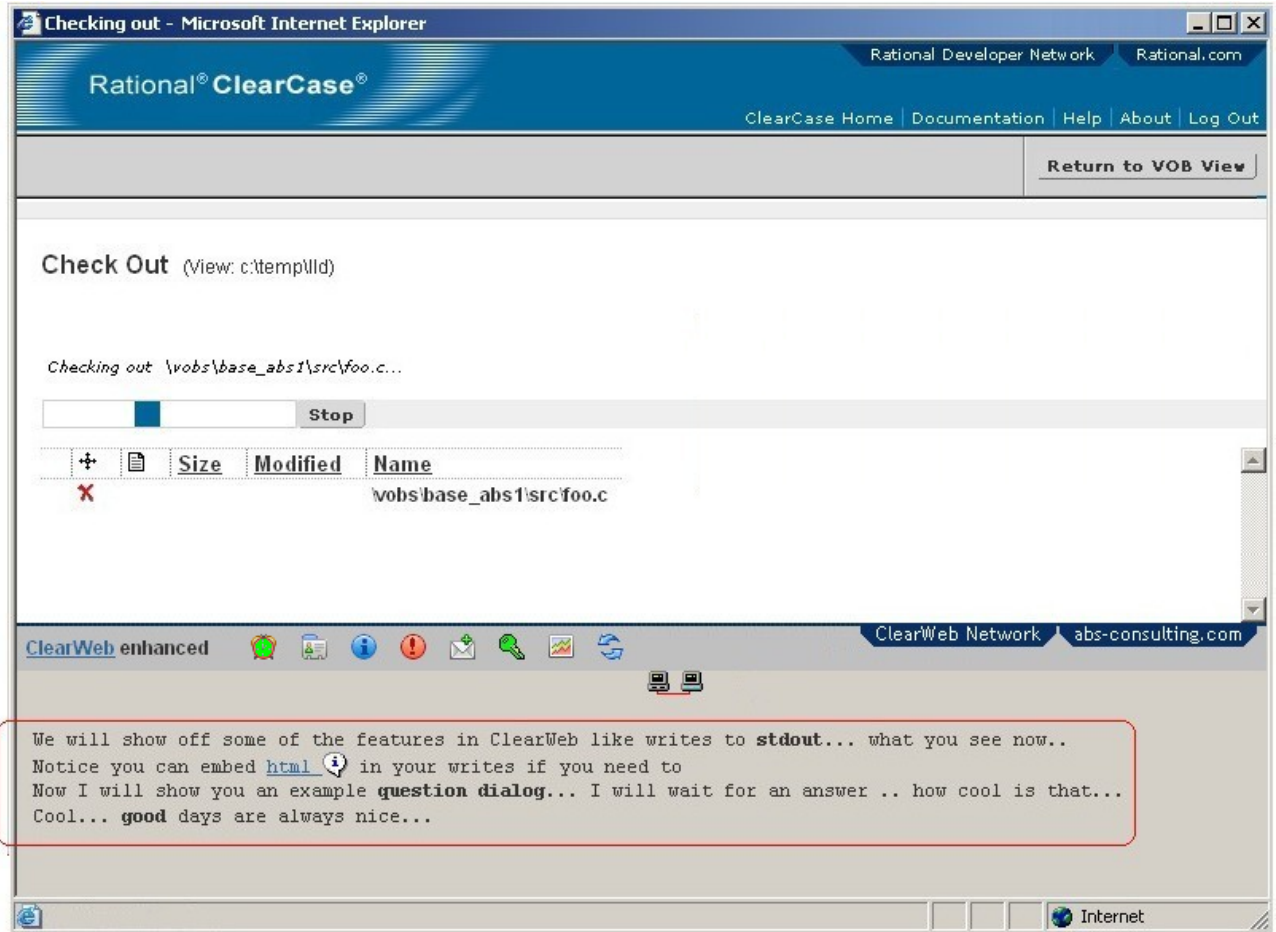
The **clearweb_write** executable is useful for making trigger scripts “web enabled” for ClearCase web users by writing to either stdout for non-web users or to the web interface for ClearCase web users. (In cases where your script did not previously call clearprompt, it is advisable to use **clearweb_out** . This performs the same function for the entire trigger.)



The **clearweb_write** executable writes to the ClearCase web-client if the user is a ClearCase web-user. If the current user is not a web user, the output is instead written to stdout. One or more arguments are accepted. That which is passed is printed to either stdout or the web-client window, which displays stdout/stderr for the ClearWeb user. The clearweb_write executable can be called from either scripting language or compiled code.

Replace any “writes” in your scripts with calls to clearweb_write. See below for examples:

Script language	Native use	Clearweb_write use
UNIX shell	echo your code is approved	clearweb_write.exe your code is approved
UNIX shell	echo \$string	clearweb_write.exe \$string
perl	printf (\$info);	system ('clearweb_write.exe ', \$info) ;
Windows batch	echo your code is approved	clearweb_write.exe your code is approved

When all “writes” are replaced with ClearWeb writes, writes to stdout will appear in normal locations for non-web users. Writes will appear in an appropriate window near the bottom of the web client for web users, as shown on the next page:



ClearCase is still responsible for informing you when the command complete, but stdout and stderr are redirected here as well and no longer cause the command to fail. The icon above the output will either be  indicating that ClearWeb is polling the server for input and dialogs to display or a this icon  which indicates that you have elected to stop polling. You should stop server polling when for example you have to leave your desk for a bit, but wish to stay in this screen. (i.e. to find someone to explain or show the screen to or to make a phone call). To toggle polling just select the icon.



clearweb_out

The **clearweb_out** executable is useful in making any trigger scripts or “web enabled” for ClearCase web users. If a script previously called clearprompt, it is advisable to call **clearweb_write** that performs the same function for specific “writes” and **clearweb_dialog** that performs clearprompt functions for the web-client.

The **clearweb_out** executable performs the task of writing to the ClearCase web-client if the user is a ClearCase web-user. If the current user is not a web user, any writes to stdout or stderr are passed through unaltered. If there are no calls to clearprompt in your script, prepend the call (or trigger definition) with clearweb_out as shown below.

Replace any “calls” in your scripts with calls to clearweb_out with your script as a parameter:

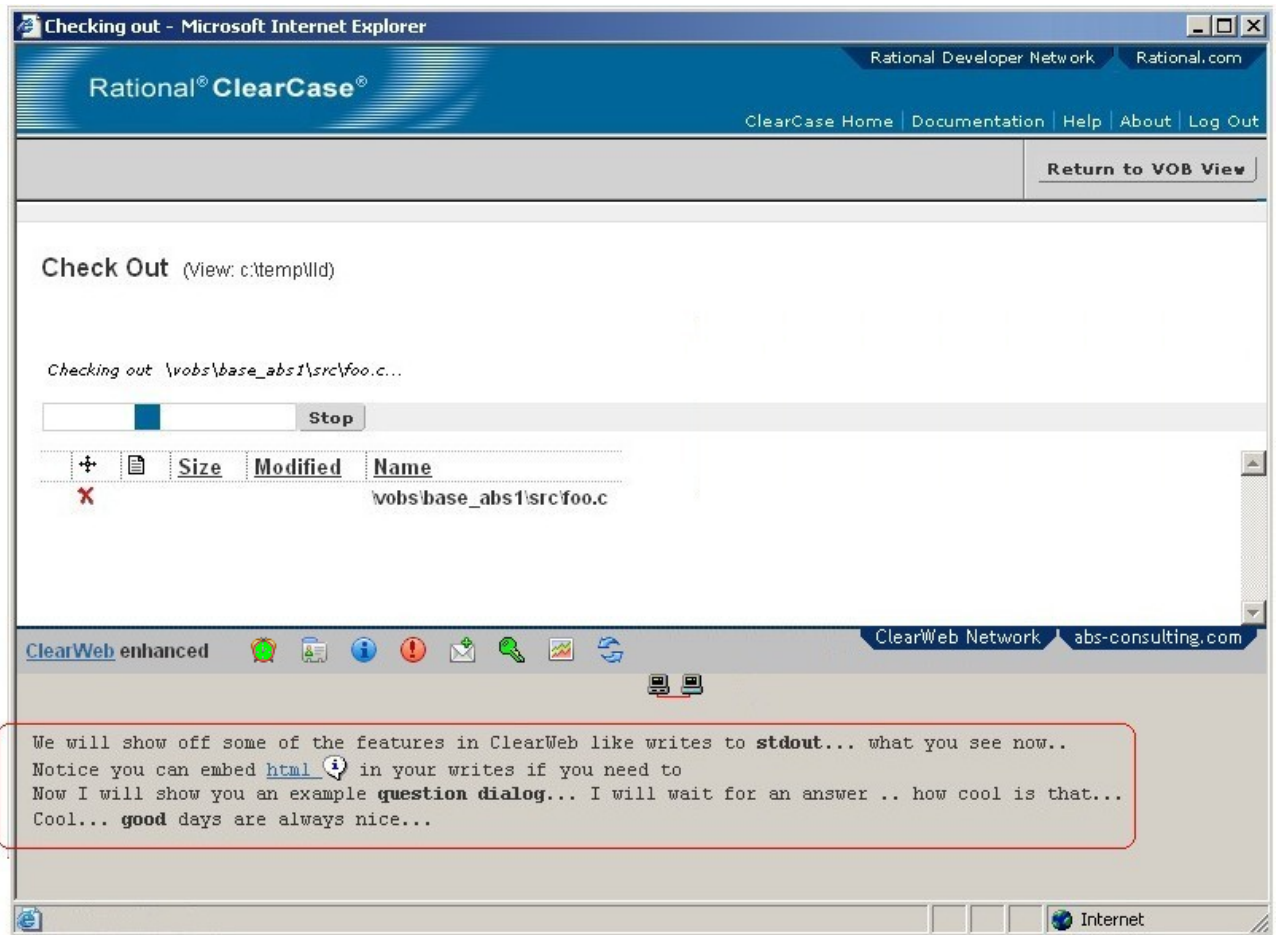
Script language	Native use	clearweb_write use
UNIX shell	script	clearweb_out.exe script
perl	system ('script ') ;	system ('clearweb_out.exe', 'script') ;
perl	system ('script ', \$parm) ;	system ('clearweb_out.exe', 'script', \$parm) ;
Windows batch	script	clearweb_out.exe script



Prepend any “trigger definitions” with clearweb_out.exe:

Native use	clearweb_out use
-exe script.pl	-exe clearweb_out.exe script.pl”
-exe “ perl script.pl arg1”	-exe “ clearweb_out.exe script.pl perl script.pl arg1”

NOTE: When a call to “script” is prepended with **clearweb_out** and the code in “script” calls other scripts or programs, those programs call in the “script” do not need to be prepended. Writes to stdout and stderr will likewise go to the web-client for ClearCase web-users.

When all appropriate “calls” are prepended with **clearweb_out**, writes to stdout will appear in normal locations for non-web users. Writes will appear in an appropriate window near the bottom of the web client for web users as shown on the next page:



ClearCase is still responsible for informing you when the command complete, but stdout and stderr are redirected here as well and no longer cause the command to fail. The Icon above the output will either be  indicating that ClearWeb is polling the server for input and dialogs to display or a this icon  which indicates that you have elected to stop polling. You should stop the polling when for example you have to leave your desk for a bit, but wish to stay in this screen. (i.e. to find someone to explain or to show a screen to). To toggle polling just select the icon.

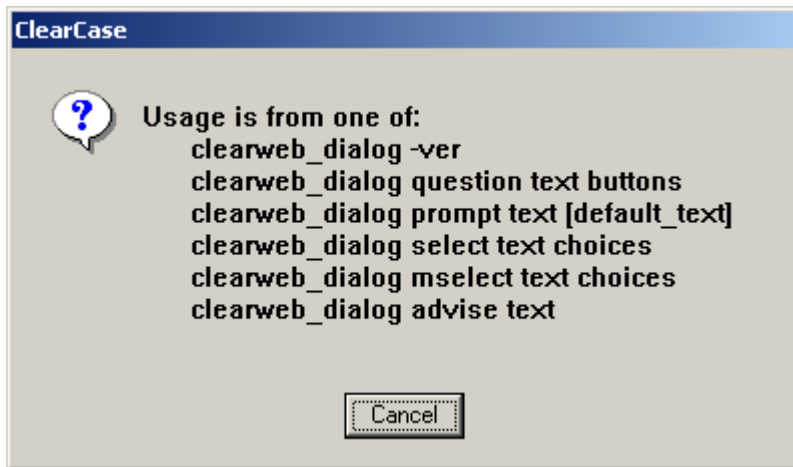
clearweb_dialog

The **clearweb_dialog** executable is useful for making trigger scripts or “web enabled” for ClearCase web users. If the script previously called clearprompt, call **clearweb_dialog**; this performs clearprompt functions for the web-client.

The **clearweb_dialog** executable performs the task of calling clearprompt when the current user is not a ClearCase web-client. If the client is a web-user, it displays a clearprompt-like dialog to the web-client. If there are no calls to clearprompt in script, it is advisable to prepend the trigger script with a call to **clearweb_out** or to use **clearweb_write** calls within your trigger script.

Replace any “calls” to clearprompt with calls to clearweb_dialog in trigger scripts.

See below for usage statement for clearweb_dialog.



Where:

argument	explanation
-ver	Displays the clearweb_dialog version information
dialog_type	question prompt select advise mselect
text	Text/prompt to display - quoted if multiple words
default_text	Pre-populated and selected text for prompt dialogs
choices	Quoted and comma-separated text for selection dialogs (i.e. “this,that,the other”)
buttons	Comma separated button text from “yes”, “ok”, “no”, “abort” or “cancel”. The string “yes” and “ok” are treated exactly the same and the strings “abort” and “cancel” are treated exactly the same. The first button listed is the “default” button (highlighted button) for non-web users; duplicated are ignored. Example: “yes,no,cancel” products three (3) buttons of which the “yes” button is the default. Example: “cancel,yes” products two (2) buttons of which the “cancel” button is the default.

The input text (if a “prompt” dialog or “select” dialog) is the output of the program

When invoked, **clearweb_dialog.exe** displays the appropriate dialog and waits for a user response (**Note Exception:** “advise” dialogs). The return code is useful for determining which button the user selected:

Clearweb_dialog return code	...means the user selected
0	“Yes” or “OK” button
1	“No” button
2	“Cancel”, “Abort” or window manager close
99	Invalid call to clearweb_dialog.exe

The input text (if a “prompt” dialog or “select” dialog) is the output of the program. Therefore, it is needless to worry about temporary file creation or deletion. Simply redirect the output directly in a variable before testing the return code to determine if the value will be used. While the same applies to any programming language, some shell examples are shown below:

Select Dialog Example

QA_State=`clearweb_dialog.exe **select** “Please select a QA_State” “Unit Untested,Unit Tested,...”`

```
if [ $? != 0 ]
then
    ### Other than “OK” ###
    echo no value was selected, user canceled
else
    ### “OK” ###
    echo the value selected is $QA_State
fi
```

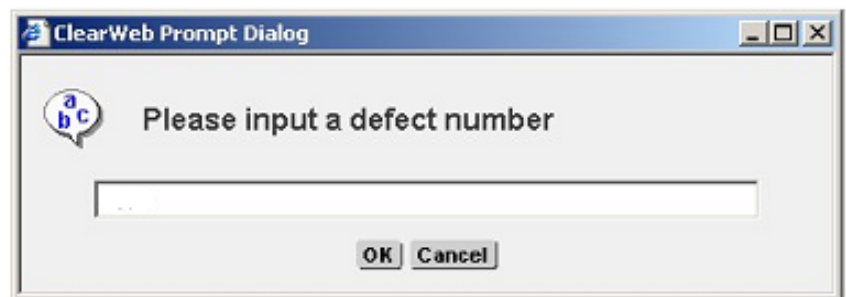
You can also test **if [“\$QA_State” == “”]**
Or use a **case/switch** statement on QA_State.



Prompt Dialog Examples

bug_num=`clearweb_dialog.exe **prompt** “Please input a defect number”`

```
if [ $? != 0 ]
then
    ### Other than “OK” ###
    echo no value was entered
    echo user canceled
else
    ### “OK” ###
    echo the value entered is $bug_num
fi
```



Same as above, but provide a default value to user

bug_num=`clearweb_dialog.exe **prompt** "Please input a defect number" 0001`

```
if [ $? != 0 ]
then
    ### Other than "OK" ###
    echo no value was entered
    echo user canceled
else
    ### "OK" ###
    echo the value entered is $bug_num
fi
```



Question Dialog Examples

clearweb_dialog.exe **question** "Does the associated flowchart need updating?" yes,no,cancel

```
case $? in
    0) echo yes selected ;;
    1) echo no selected ;;
    2) echo cancel selected ;;
    *) echo invalid call ;;
esac
```



Same as above but with different buttons

clearweb_dialog.exe **question** "Does the associated flowchart need updating?" yes,cancel

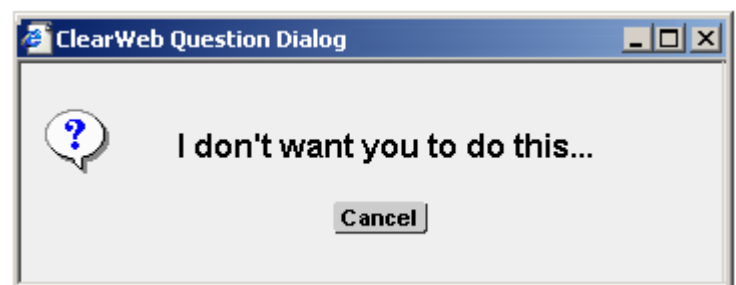
```
case $? in
    0) echo yes selected ;;
    2) echo cancel selected ;;
    *) echo invalid call ;;
esac
```



Same as above but only give user the cancel option

clearweb_dialog.exe **question** "I don't want you to do this..." cancel

user is only given the "cancel" option
 ### so this is useful for when you want
 ### them to see a denial message, but
 ### acknowledge the message first before
 ### proceeding



Advise Dialog Example

clearweb_dialog.exe **advise** "I don't want you to do this..."

user is only given the "ok" option
in a modeless dialog so it displays and
does not wait.



ClearCase Remote Client (CRC) support

The ClearCase Remote Client will natively will handle “*some*” interactive triggers. In previous releases you could not get interactive triggers to work for both ClearCase Web Clients and ClearCase Remote Client simultaneously (as they interfered with each other). ClearCase itself does not distinguish between the two. ClearWeb 7.0 fully supports both ClearCase Web Clients and ClearCaser Remote Clients even both at the same time from a single server instance.

When ClearWeb was installed it made an additional copy (or link on UNIX installations) of **ccweb.exe** called **ccweb_remote.exe** (or ccweb_remote for UNIX). Have your ClearCase Web users use the ccweb address (i.e. **http://server/ccweb/bin/ccweb**) and your ClearCase Remote Client users use the ccweb_remote address (i.e. **http://server/ccweb/bin/ccweb_remote**). Doing so will enable **clearweb_dialog**, **clearweb_out** and **clearweb_write** to know which is being use so now dialogs will work for both interfaces.