

Rational. software

The IBM logo, consisting of the letters "IBM" in a stylized, striped font, is positioned in the top right corner of the page.

Customizing Administration Tools in IBM Rational ClearCase 4.0 NT

Abstract

This white paper describes how to customize the administration tools available in IBM Rational ClearCase 4.0 on Windows NT. These tools, implemented using Microsoft Management Console technology, can be used to build a variety of special-purpose administration interfaces tailored to the needs of individual administrators, tasks, and sites. A brief description of the Microsoft Management Console (MMC) is given, followed by tutorials of how to create custom administration applications.

NOTE: The examples shown in this paper assume that you're running ClearCase 4.0 on Windows 2000. Windows 2000 provides MMC version 1.2 by default, which contains a number of UI improvements over previous versions. ClearCase 4.0 installed on Windows NT 4.0 installs MMC 1.1 by default if MMC is not already present. MMC 1.2 for Windows NT 4.0 is available for free in the Microsoft Platform SDK, January 2000 and beyond. Where noted, tutorial examples in this paper can be implemented using MMC 1.1.

Introduction

The graphical administration tools available in ClearCase 4.0 for Windows NT are written using the Microsoft Management Console (MMC) administration framework.

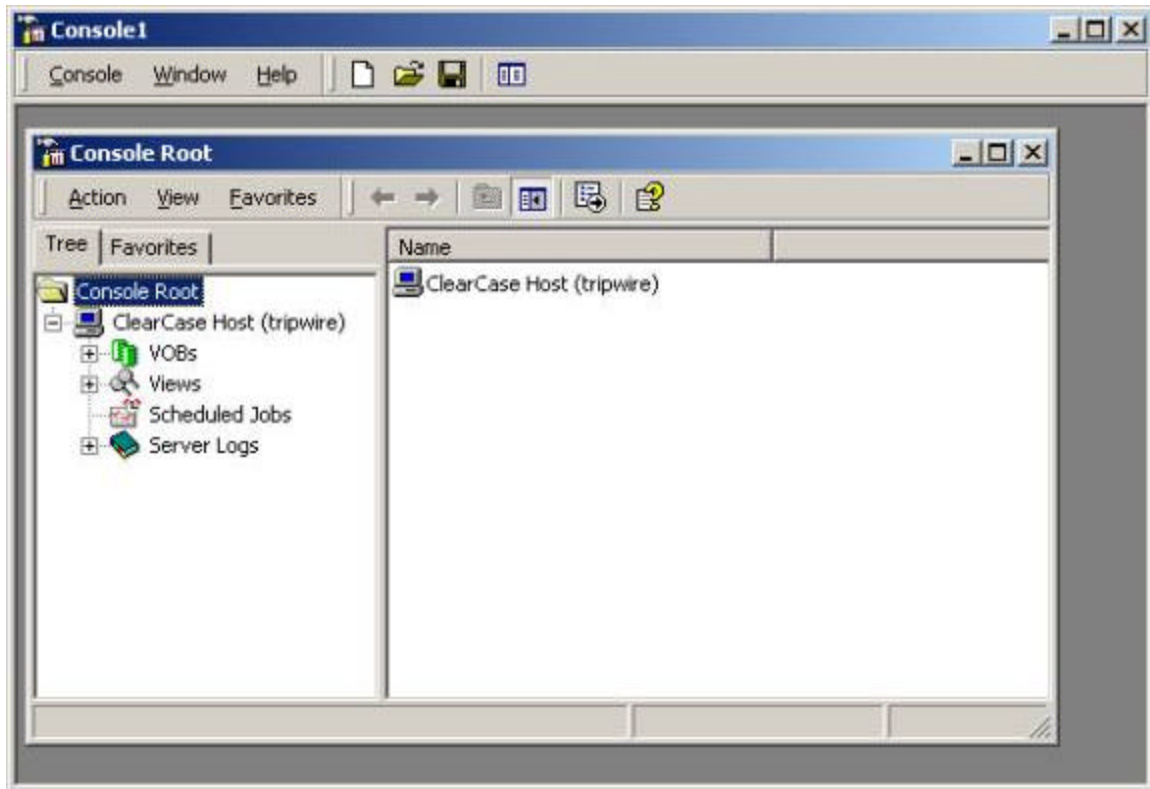
MMC is a multiple document framework that allows you to create and save *consoles*, which provide graphical interfaces to systems, services, and applications on your computer and the network of which it is a part. Each console is a window managed by the MMC that contains one or more administrative components called *snap-ins*. Vendors provide snap-ins (in-proc COM servers) that allow you to construct an administration *namespace* by creating hierarchies of nodes that each contain some manageable object, task, or view for that vendor's domain. The snap-in architecture is extensible. Using well-known snap-in identifiers and data transfer formats, snap-ins from the same or different vendors may extend one another's namespace, property sheets, menus, and toolbars. IBM Rational provides a number of ClearCase snap-ins that allow for the remote administration of ClearCase servers.

User Mode Consoles

Figure 1 shows the ClearCase Host Administration console as shipped with ClearCase 4.0. **cchostadmin.msc** is actually an MMC console file saved in *user* mode. It is an OLE-structured storage compound document, read and written by MMC. As such it can be transferred to (for example, by e-mail) and opened by any workstation on which ClearCase 4.0 and MMC are installed. Because it was saved in user mode, the user cannot change the console configuration in any way. - ClearCase Host Administration

Author Mode Consoles

Figure 2 shows an MMC session in *author* mode, with the ClearCase Host snap-in loaded. Because MMC was started in author mode, more user interface elements are available and are described here briefly to aid in understanding the examples that follow.



MMC in console authoring mode

At the top of the application is the console main menu bar and authoring toolbars. These UI components allow you to create console windows and to craft the namespace of those consoles by adding and removing primary and extension snap-ins.

Beneath the main menu bar is a console window that contains the *action* menu bar and toolbar. (Note that an MMC session can contain multiple console windows, although only one is shown here.) These UI features allow you to perform an action on a selected item or to alter the view of the console window. Beneath that action bar are the *console* and *details* panes. The *console pane* on the left contains the namespace for the snap-ins added to the console, shown in the tree tab by default. The *Favorites* tab (available in MMC 1.2 only) displays short cuts to items within the console tree. (We describe this feature in detail later in this document.) The *details pane* on the right displays details for the item selected in the console pane.

The remainder of this document describes how to create custom consoles to shape the ClearCase administration namespace in a way that meets your administrative needs.

Customization Tutorials

[Tutorial 1: Important VOBs](#)

If your network hosts a large number of ClearCase clients, the ClearCase Network namespace of the ClearCase Administration Console can become dense and difficult to navigate. An alternative approach to managing ClearCase servers and clients is to create a custom namespace of only those servers that require frequent monitoring. In this example, we'll create a custom namespace called "Important VOBs" that consists of those ClearCase VOB servers that we require more efficient access to. This tutorial can make use of either MMC 1.1 or MMC 1.2.

Step 1: Run MMC

Start MMC from the Run menu or a command shell. The window as shown in Figure 3 appears.

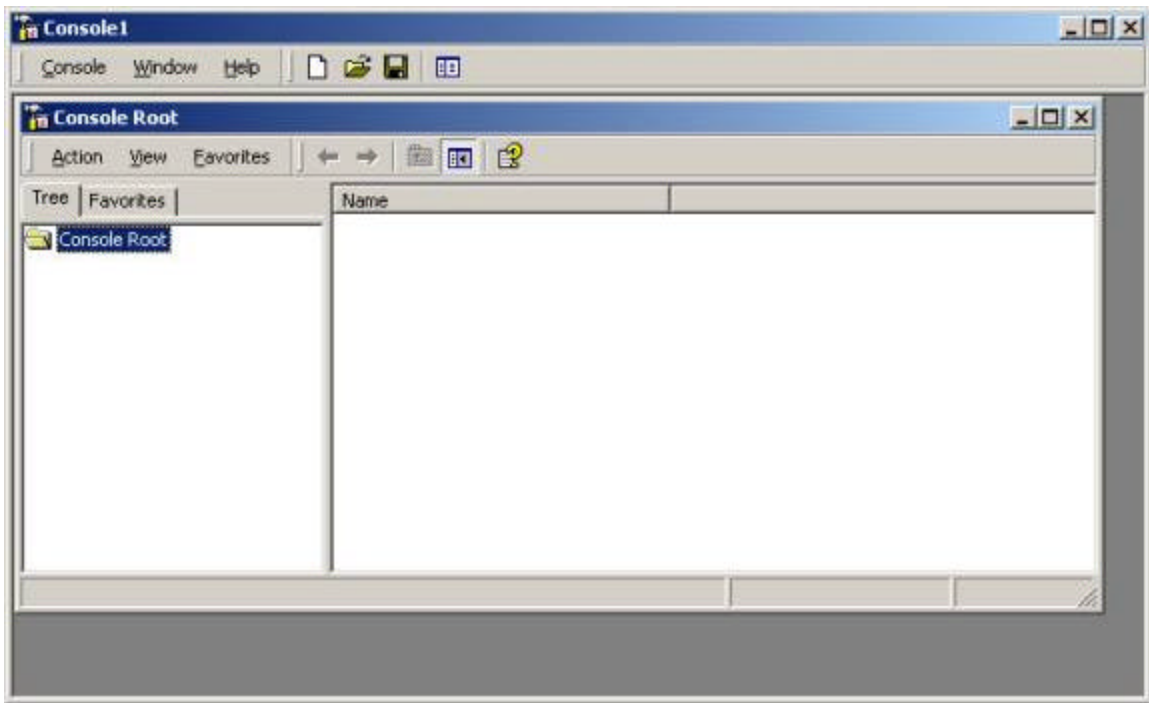


Figure 3 MMC application

Step 2: Add/Remove Snap-in

From the Console menu, select **Add/Remove Snap-in**. The Add/Remove Snap-in dialog box appears (Figure 4).

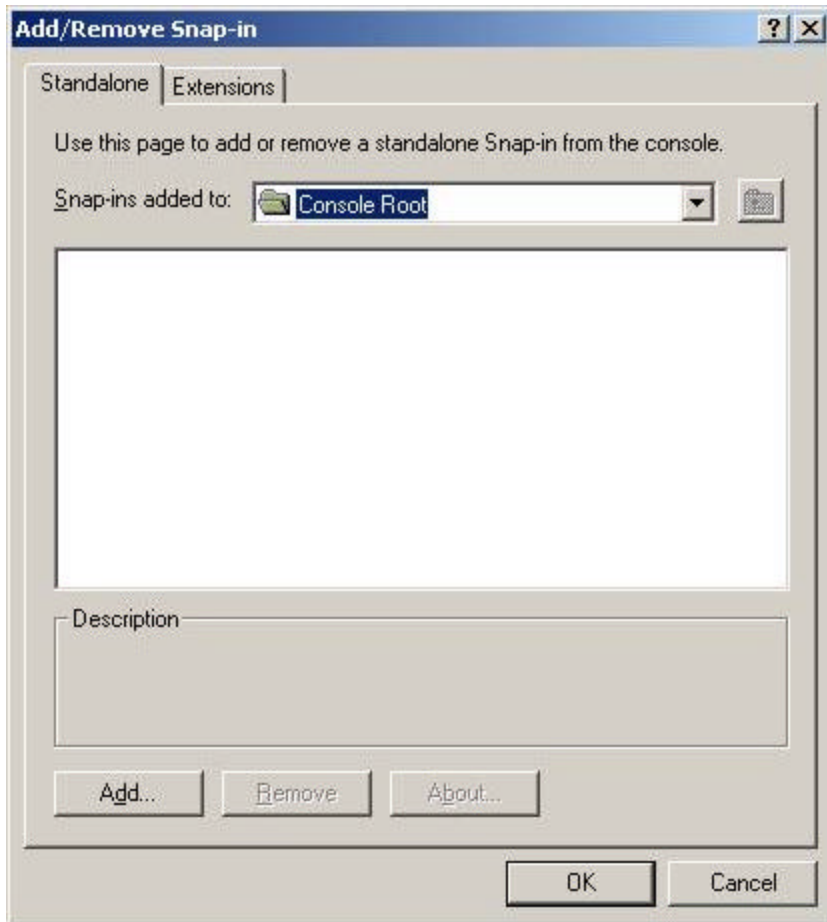


Figure 4 Add/Remove Snap-in dialog box

The Add/Remove Snap-in dialog box allows you to modify the namespace of the selected console. The dialog box contains two tabs, one for adding Standalone snap-ins, and the other for adding Extension snap-ins to selected snap-ins (Extension snap-ins, an advanced topic, is not discussed in this paper). On the **Standalone** tab, a combo box shows where in the namespace added snap-ins will appear. In this example, we will be adding snap-ins beneath the *Console Root* node, the default root node of any custom namespace.

Step 3: Add Standalone Snap-in

From the Add/Remove Snap-in dialog box, click **Add** to begin adding snap-ins beneath the Console Root. The Add Standalone Snap-in dialog box appears (Figure 5).



Figure 5 Add Standalone Snap-in dialog box

The list of snap-ins that appears here depends on the configuration of the computer you're using. Figure 5 shows snap-ins shipped with ClearCase 4.0 as well as snap-ins from Microsoft and other ISVs, including the standard snap-ins such as *Folder*, *Link to Web Address*, and *ActiveX Control*. Scroll down the list, select the *Folder* snap-in, click **Add**, and then click **Close**. The Add/Remove Snap-in dialog box now appears as in

Step 4: Review Add/Remove Snap-in Work Area

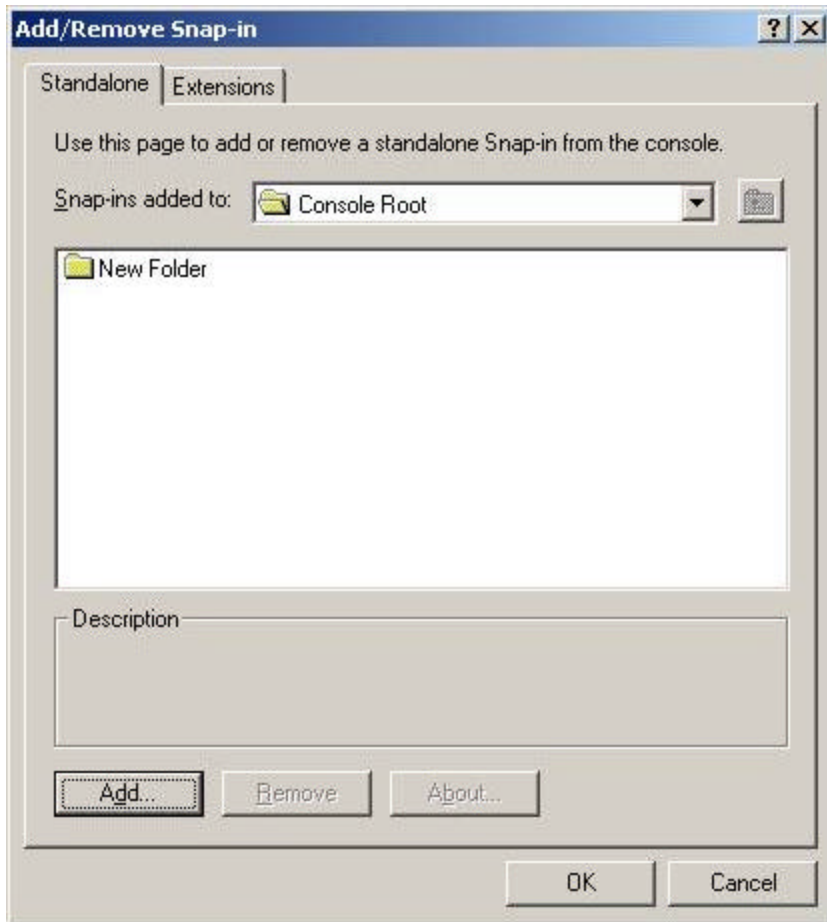


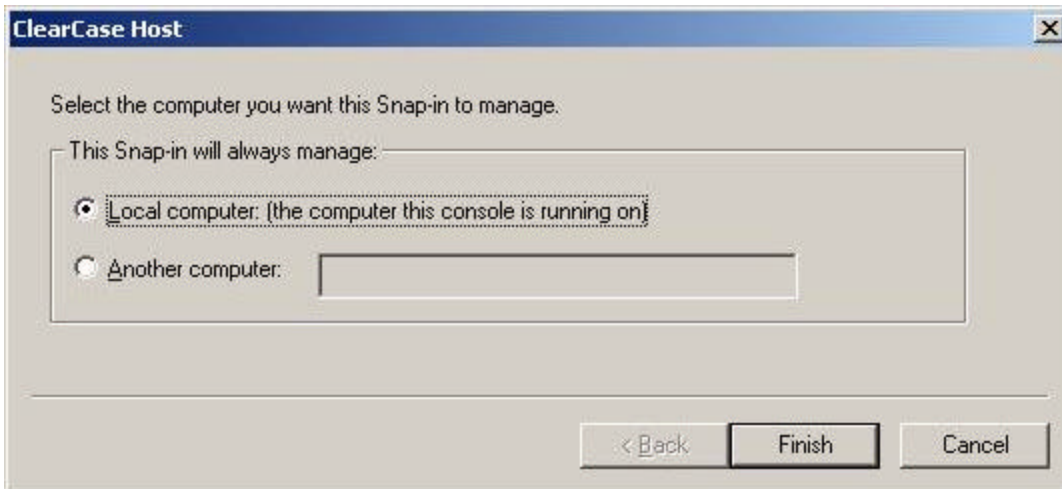
Figure 6 Add/Remove Snap-in work area

Think of the Add/Remove Snap-in dialog box as a work area in which we create our namespace. After we confirm it (by clicking **OK**), the namespace is added to the console window. Right now, all we have is the New Folder we created in the previous step. In the namespace we're creating, it will appear subordinate to the Console Root.

Step 5: Add ClearCase Host Snap-in

Next we'll be adding ClearCase snap-ins subordinate to New Folder:

1. Double-click the New Folder entry so that it becomes the current selection in the **Snap-ins added to** combo box, or scroll to select New Folder.
2. Click **Add** to open the Add Standalone Snap-in dialog box again. This time, click **ClearCase Host** and then click **Add**. The ClearCase Host dialog box appears (Figure 7).



< Snap-in Host ClearCase 7>The ClearCase Host Snap-in can be configured to allow administration of any computer running ClearCase. Since we're interested in other computers, click **Another Computer** and type in the name of a machine that is the VOB server for one of the "Important VOBs" (a machine named **saltydog** in this example). Click **Finish**. Note that the ClearCase Host Snap-in instantiated for **saltydog** has been added to the Add/Remove Snap-in dialog box. Continue to click **Add** in the Add Standalone Snap-in dialog box, adding snap-ins for the remaining Important VOB servers. When you're finished, click **Close** in the Add Standalone Snap-in dialog box.

Step 6: Confirm the Snap-in Configuration

In the Add/Remove Snap-in dialog box, click **OK** to confirm your edits and update the MMC console. That console appears as in Figure 8

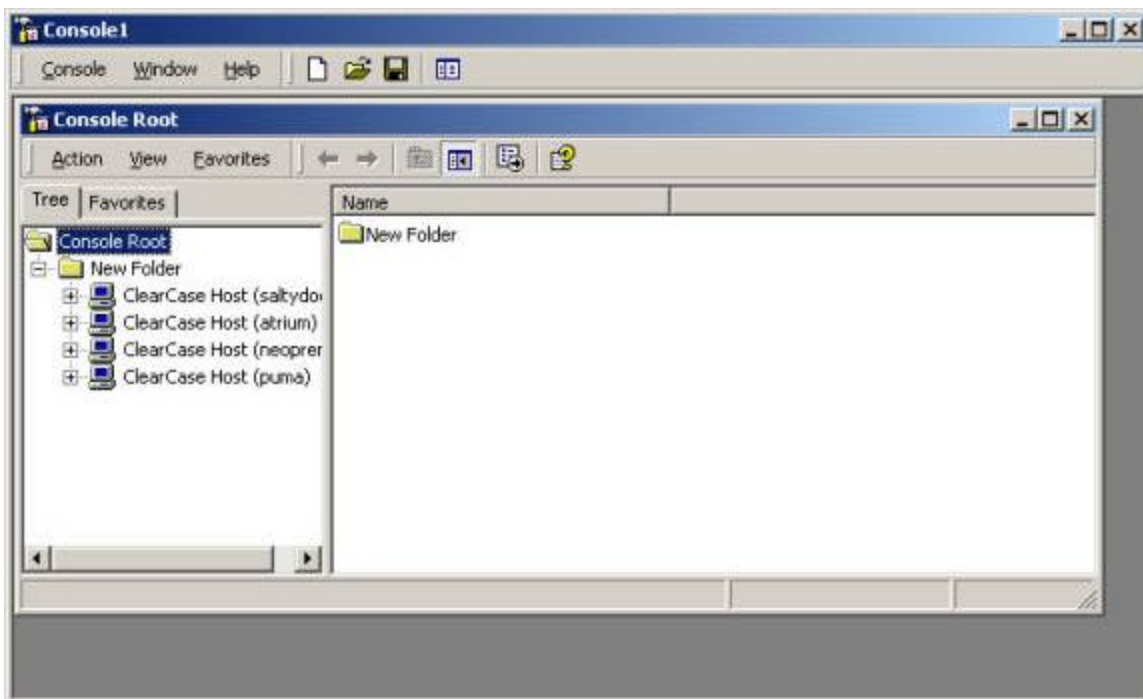


Figure 8 ClearCase Hosts added to New Folder

Step 7: Tidy the Namespace

Now that the basic console namespace has been created, we can clean it up for better usability. First, give the New Folder folder a more meaningful name. Select the New Folder node and rename it, by either clicking **Action > Rename**, or right-clicking and clicking **Rename** on the shortcut menu. After clicking **Rename**, the New Folder text item becomes editable. Type "Important VOBs" and press Enter to confirm your edits.

Next, we want to get rid of the Console Root node. Since we don't intend this console to host additional namespaces under the root, the root does not need to appear. You could rename it in the existing console, but to remove it entirely, you must create a new console window with the Important VOBs node as its root. Select Important VOBs, and then click **Action > New Window from Here**. A new console window appears, rooted on Important VOBs. Close the console window rooted on Console Root, and then maximize the window rooted on Important VOBs so that MMC looks like Figure 9

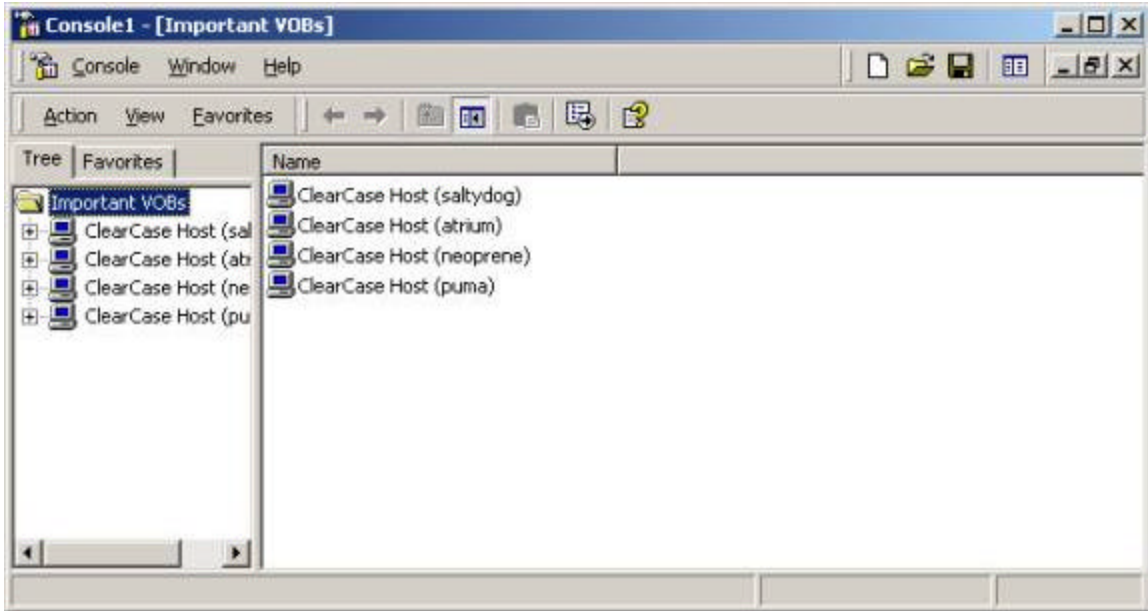


Figure 9 Important VOBs with Console Root removed

Step 8: Save the Console

We're almost finished. Because we don't expect to modify this console, and the mechanisms for modification are typically confusing for most users, we want to save this console in user mode, disallowing customization for this particular console.

Click **Console > Options**. The Options dialog box appears as in Figure 10

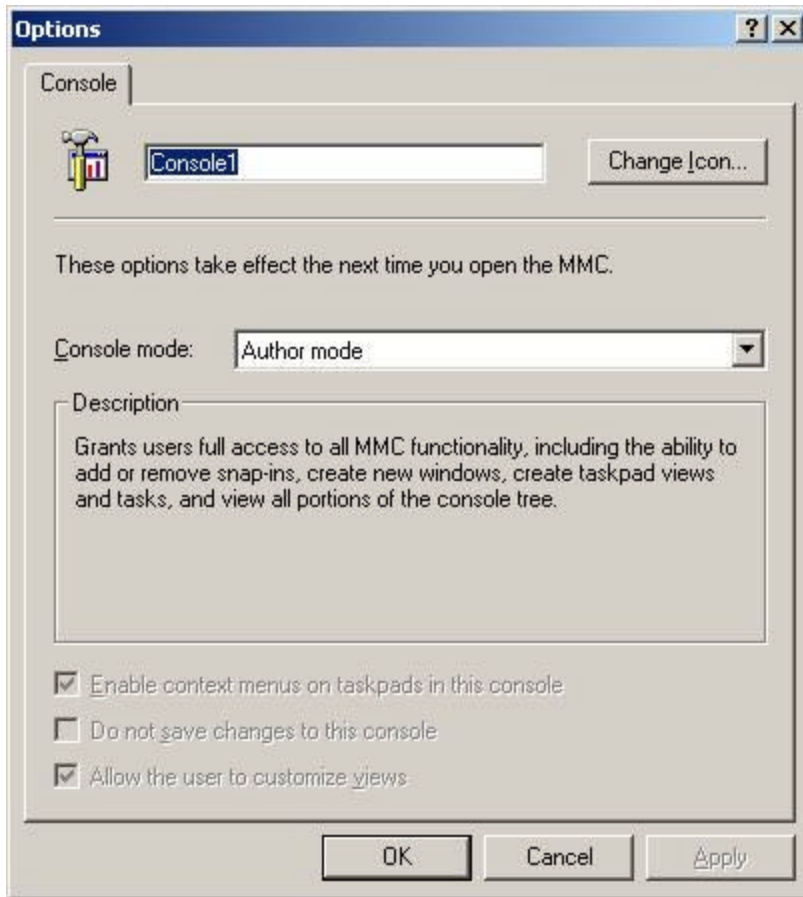


Figure 10 Saved console options

The top text edit allows you rename the main console window. Click this item and type Important VOBs. You can also change the iconic representation of this console as it appears in (for example) Windows Explorer or Taskbar, but we'll leave that for a later example.

Select the **Console mode** combo box. Scroll and select **User mode limited access, single window**. This mode is most restrictive, and will only show those parts of the namespace that were shown when the console was saved (recall that we created a window minus the console root). Also, because only one window will be available to the user, we reduce potential confusion created by multiple windows. Finally, clear the **Enable context menus on taskpads in this console** and the **Allow the user to customize views check boxes**. Click **OK** to apply these changes to the console.

Note that there will be no visible change to the console once you apply those options, as they apply only to the console once it has been saved. Save the console now by clicking **Console > Save**. A File Save dialog box appears. Type a name for the console that describes its purpose. Again, type Important VOBs in the **File Name** box, and click **Save**. Now click **Console > Exit** to close the MMC.

Step 9: Open the Console

To review the changes that we made to the console, we'll need to open it. Open Windows Explorer, and move to the place in the file system where you saved Important VOBs.msc. Double-click that file, or select **Open** from its shortcut menu. The console as shown in Figure 11 appears.

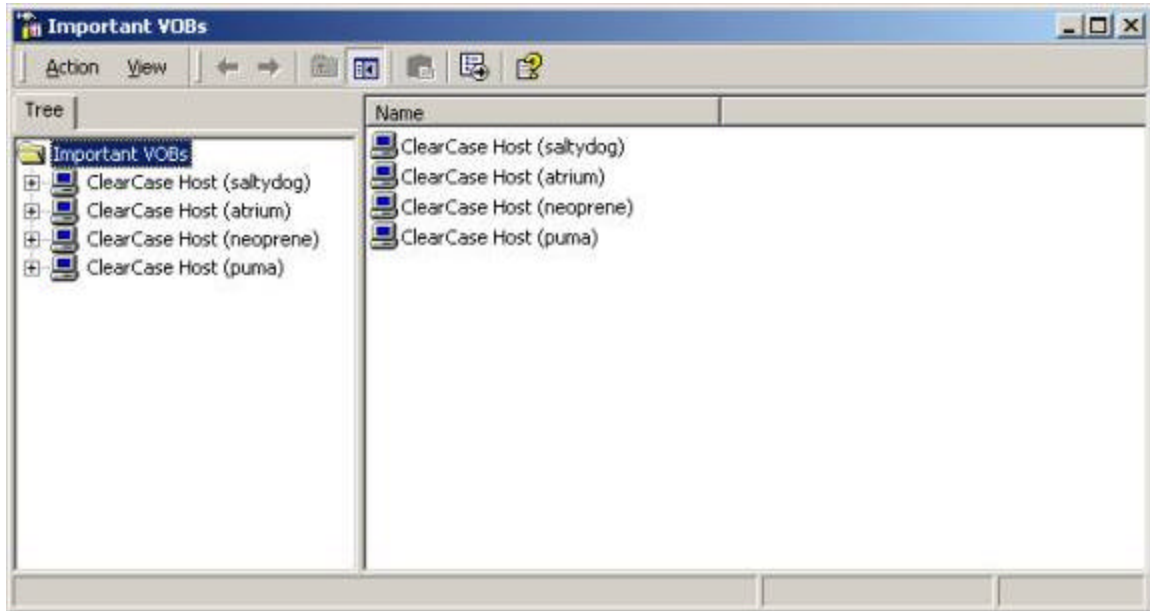


Figure 11 Important VOBs saved console

You're finished with this tutorial. You now have a console file that allows quick access to your most significant VOB servers, a namespace reduced from potentially hundreds of computers to a critical few. And because it's actually a saved document file, it can be easily installed on any machine running ClearCase 4.0 and used to remotely administer **Important VOBs**.

Tutorial 2 : Favorites

This example is similar to [the previous tutorial](#) in that we'll craft a namespace that focuses only on VOB servers of importance. However, we'll do this in a different way, by adding customization to the ClearCase Administration Snap-in, specifically by adding **Favorites** bookmarks that refer back to special locations in the ClearCase Administration namespace (note that the Favorites feature is available only in MMC 1.2 and beyond).

Step 1: Run MMC and Add the ClearCase Administration Snap-in

Start by running MMC as described in the [previous tutorial](#). Click **Console > Add/Remove Snap-in** as [described previously](#), and click **Add**. In the [Add Standalone Snap-in dialog box](#), select **ClearCase Administration**, and then click **Add**, and then **Close**. The Add/Remove Snap-in dialog box now shows **ClearCase** in its work area. Click **OK**. The console appears as in Figure 12.

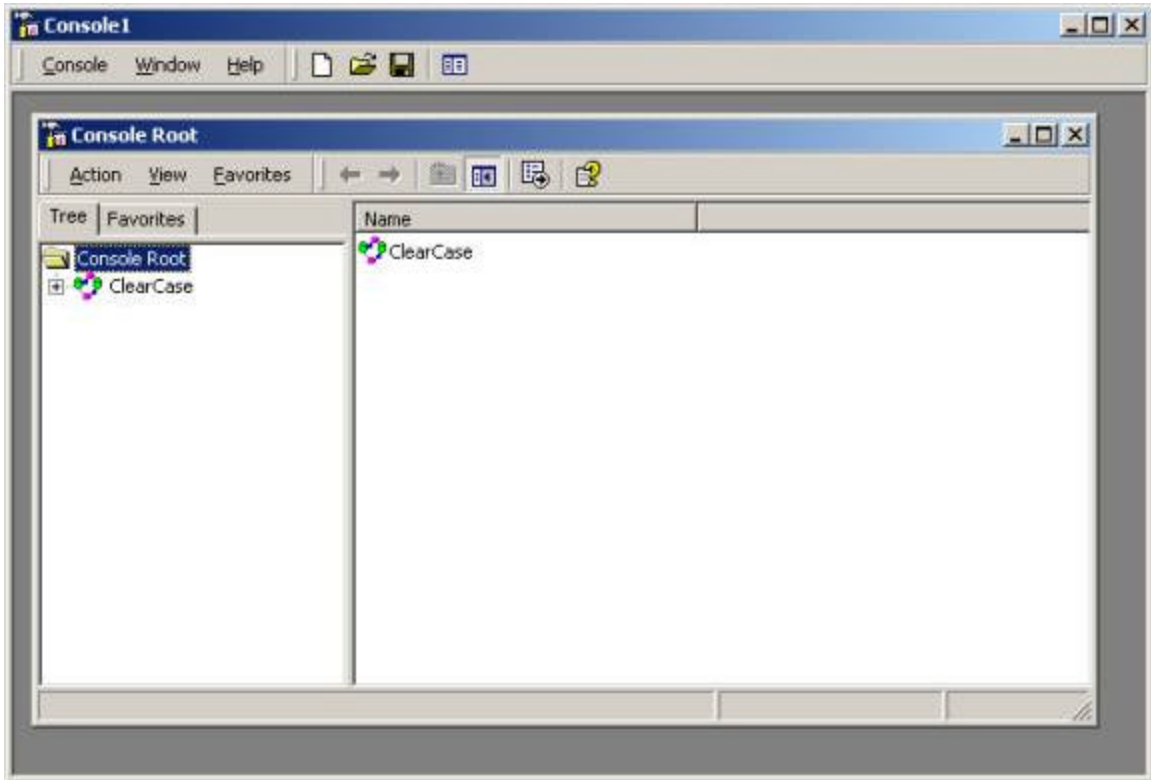


Figure 12 Adding the ClearCase Administration Snap-in

Step 2: Show ClearCase as the Console Root

As in the previous tutorial, remove the Console Root node by selecting the ClearCase node, and then clicking **Action > New Window from Here**. Close the original console window and maximize the new one. The console now appears as Figure 13.

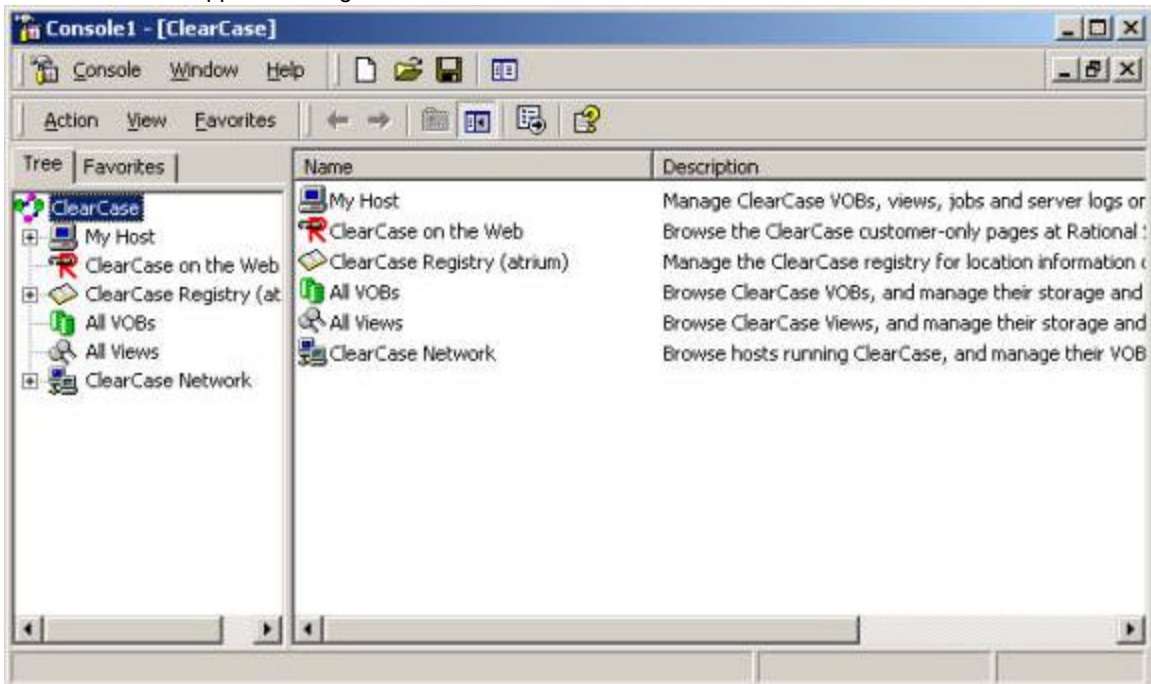


Figure 13 ClearCase Administration Snap-in with Console Root removed

Step 3: Select VOB space

Next you'll want to add shortcuts to console items that you frequently visit. Expand the ClearCase Network node and search for a VOB server of interest. In this example, we select the node for server named *atrium* in the scope pane, and then expand the node to show VOBs on that host. We select */usr1/vbs/atria_vob* to show its space display, as in Figure 14.

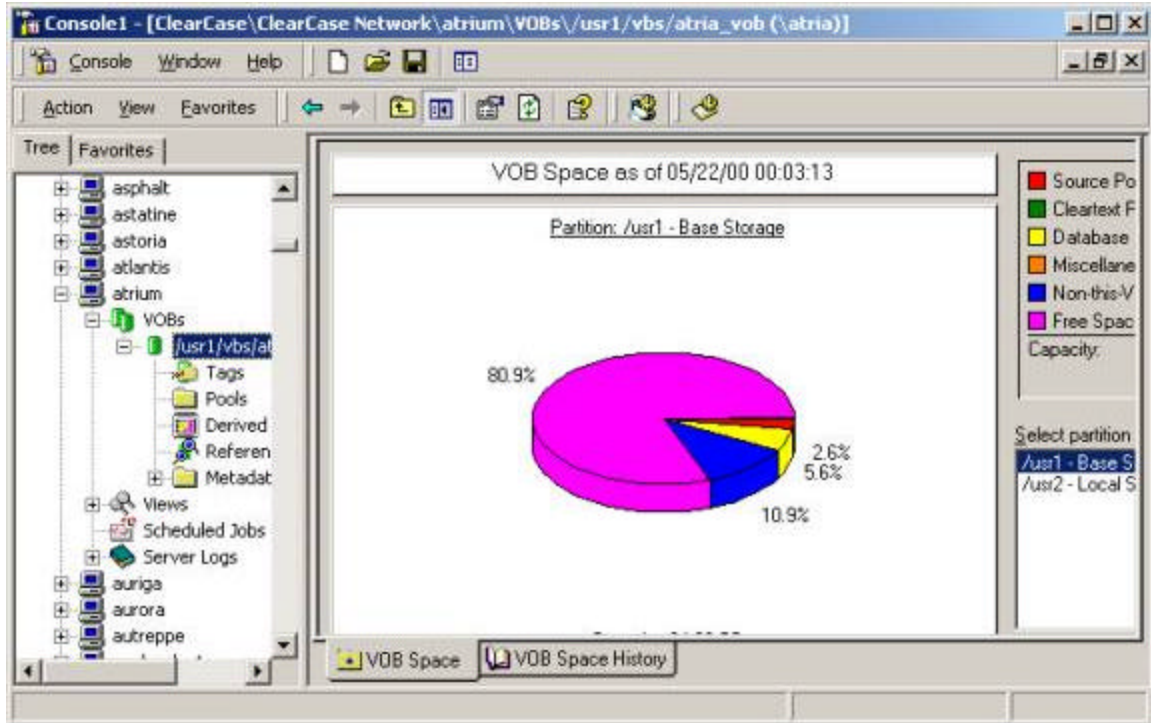


Figure 14 VOBs on host atrium

Step 4: Add /usr1/vbs/atria_vob to the Favorites Tab

With */usr1/vbs/atria_vob* selected, click **Favorites > Add to Favorites** from the Favorites action bar menu. The Add to Favorites dialog box is shown as in Figure 15.

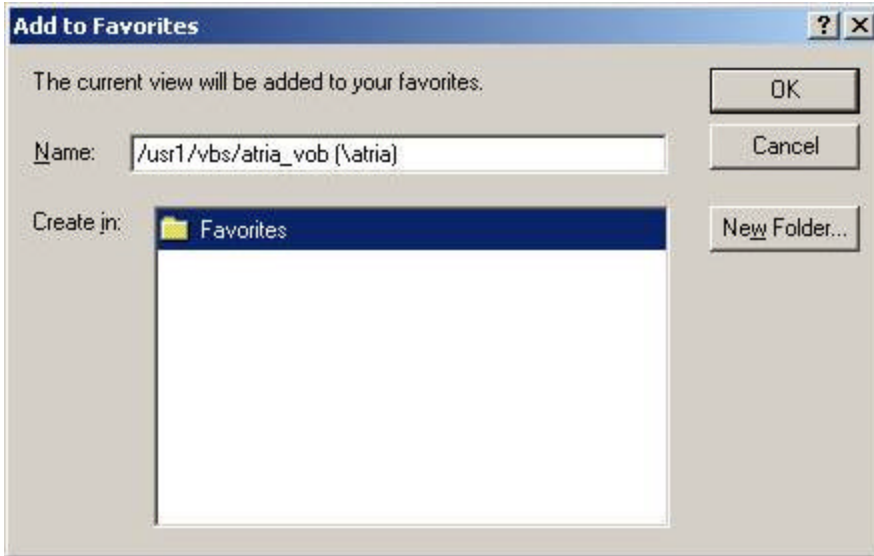


Figure 15 Add To Favorites

In this example, we'll add a folder to organize the VOBs, and then add shortcuts to the namespace to that. Click New Folder. A New Favorites dialog box is opened with a text item in which you'll type the name of your new folder. Type **Important VOBs**, and then click **OK**. The folder Important VOBs appears subordinate to the Favorites folder. Select the Important VOBs folder. Next, type the name of the VOB in the **Name** box of the Add to Favorites dialog box. By default the VOB storage path is shown; you may want to give it a name that's easier to remember. In this example, we'll call it **/vobs/atria**. After typing the name, click **OK** to confirm the edits and dismiss the dialog box. In the main console window, click the **Favorites** tab. Note the inclusion of the Important VOBs folder, with its single child **/vobs/atria**. Selecting **/vobs/atria** shows the space pie chart as in the Tree display, as in Figure 16.

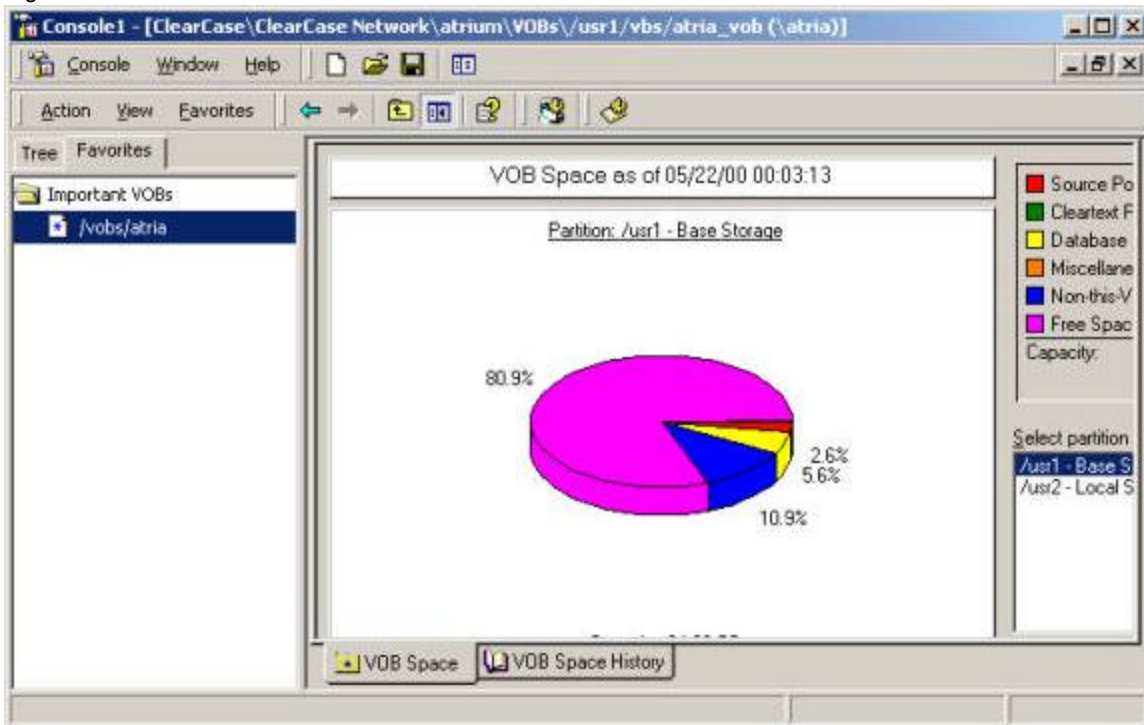


Figure 16 /vobs/atria as Favorites

Continue to add VOBs to your list of Favorites until you have added all the important ones. Note that *any* item in the console tree may be added to Favorites as a shortcut, not just the VOB space OCX.

Step 5: Save the Console

Finally, specify whether to save this console in author or user mode via Console/Options, and then save the console to your administrative tools folder as in the [previous tutorial](#).

Tutorial 3: Taskpads

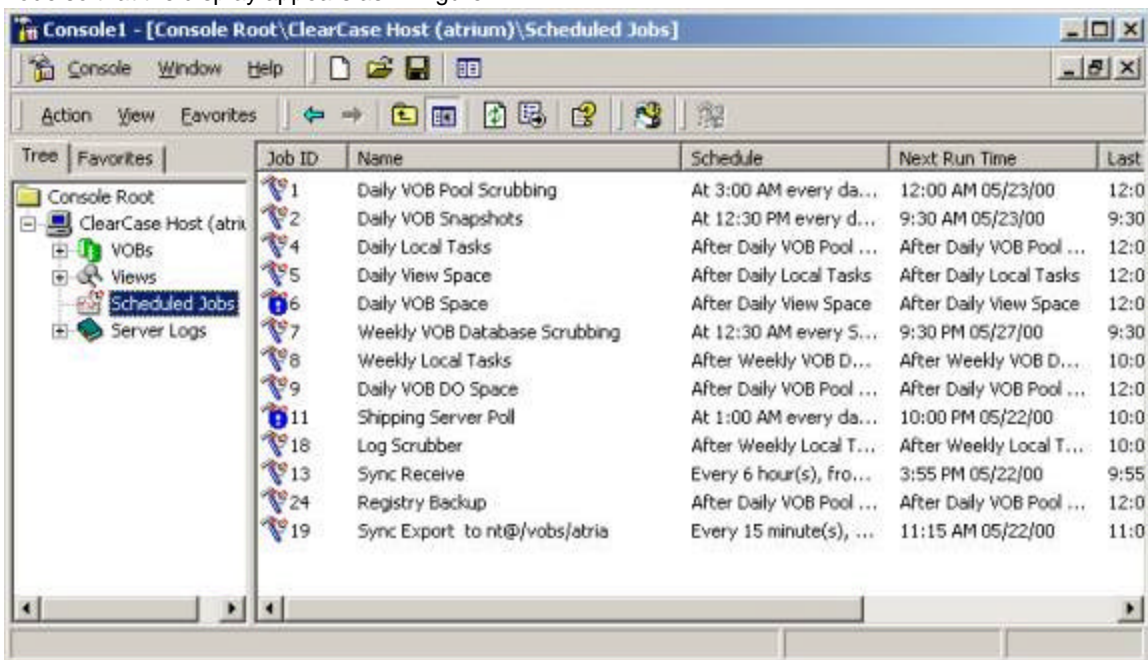
In this tutorial, we'll again take advantage of a feature that's available only in MMC 1.2, and demonstrate how to create a custom taskpad.

Taskpads in MMC are displays that embed commands and selection details on a single HTML page. The All VOBs and All Views console items in the ClearCase Administration Snap-in are examples of custom taskpads that were written for this application, to aid in navigating from a VOB or view-tag to its storage or registry entry. (More precisely, they are *extension* snap-ins, that extend only the ClearCase Administration Snap-in.)

MMC 1.2 allows you to write your own taskpads when creating a namespace, typically to create a more intuitive or easy-to-use administration tool. In this example, we'll show how to create a taskpad for ClearCase scheduled jobs. It is assumed that you have run the previous tutorials and are familiar with creating a custom namespace.

Step 1: Create ClearCase Host Namespace and Show Scheduled Jobs

Run MMC. Click **Console > Add/Remove Snap-in** and then click **Add**. In the Add Standalone Snap-in dialog box, select **ClearCase Host**, click **Add**, and then in the ClearCase Host dialog box ([as described in a previous tutorial](#)), type the name of a VOB server. In our example, we click **Another Computer**, type **atrium**, and then click **OK**. Click **Close** on the Add Standalone Snap-in dialog box, and then click **OK** in the Add/Remove Snap-in dialog box to create the namespace. The ClearCase Host namespace is displayed in the console. Expand the ClearCase Host item (by selecting its + icon), and then select the Scheduled Jobs node so that the display appears as in Figure 17.



Job ID	Name	Schedule	Next Run Time	Last
1	Daily VOB Pool Scrubbing	At 3:00 AM every da...	12:00 AM 05/23/00	12:0
2	Daily VOB Snapshots	At 12:30 PM every d...	9:30 AM 05/23/00	9:30
4	Daily Local Tasks	After Daily VOB Pool ...	After Daily VOB Pool ...	12:0
5	Daily View Space	After Daily Local Tasks	After Daily Local Tasks	12:0
6	Daily VOB Space	After Daily View Space	After Daily View Space	12:0
7	Weekly VOB Database Scrubbing	At 12:30 AM every S...	9:30 PM 05/27/00	9:30
8	Weekly Local Tasks	After Weekly VOB D...	After Weekly VOB D...	10:0
9	Daily VOB DO Space	After Daily VOB Pool ...	After Daily VOB Pool ...	12:0
11	Shipping Server Poll	At 1:00 AM every da...	10:00 PM 05/22/00	10:0
18	Log Scrubber	After Weekly Local T...	After Weekly Local T...	10:0
13	Sync Receive	Every 6 hour(s), fro...	3:55 PM 05/22/00	9:55
24	Registry Backup	After Daily VOB Pool ...	After Daily VOB Pool ...	12:0
19	Sync Export to nt@/vobs/atria	Every 15 minute(s), ...	11:15 AM 05/22/00	11:0

Figure 17 Scheduled Jobs Details

Step 2: Create Taskpad

With the Scheduled Jobs node selected, right-click to display its shortcut menu, and click **New Taskpad View**. (Alternatively, you can click **Action > New Taskpad View**.) The New Taskpad View wizard is displayed as in Figure 18.

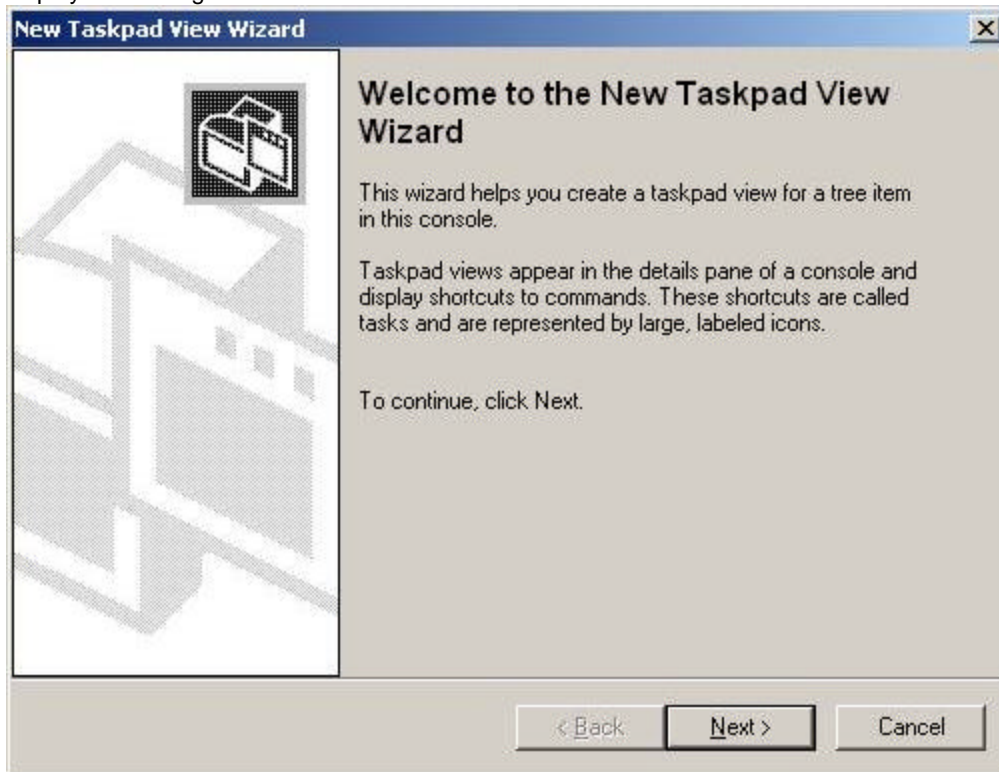


Figure 18 New Taskpad View Wizard

Step 3: Select Taskpad View and Target

Click **Next** to display the Taskpad display page. Click **Horizontal list** for its details pane, **InfoTip** for its task descriptions, and **Medium** for its List size. Click **Next** to display the Taskpad Target page, as in Figure 19.

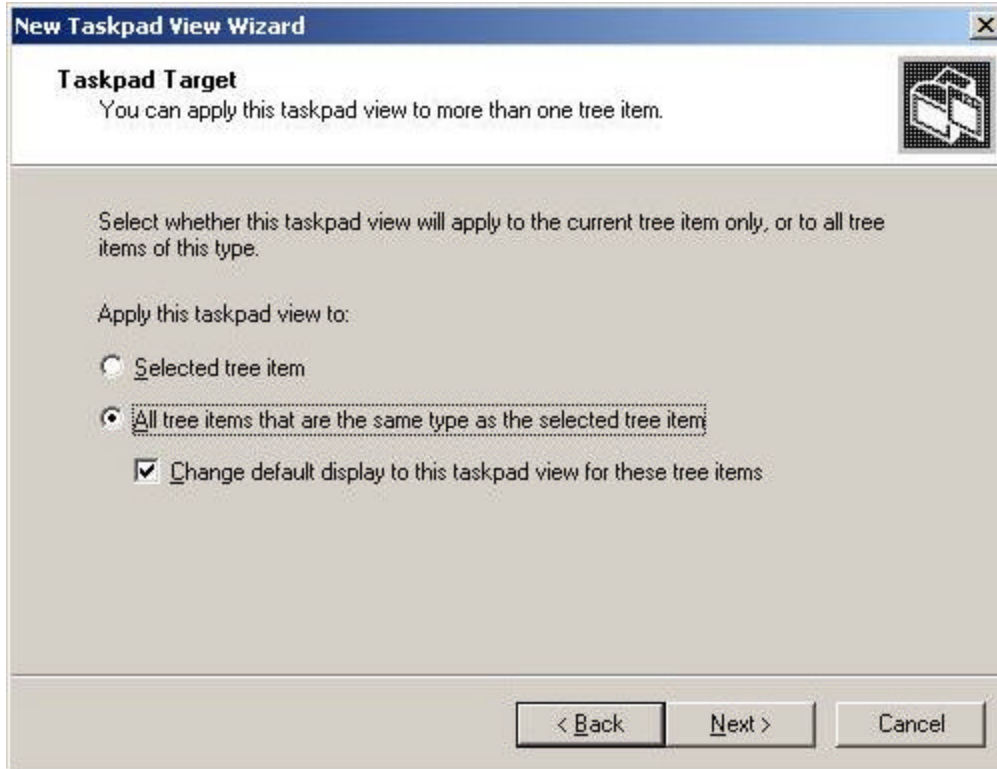


Figure 19 Taskpad Target page

In this particular example, it doesn't really matter which options you select, as we'll be showing only one specific Scheduled Jobs node. You can leave the defaults as they are and click **Next**.

Step 4: Specify Taskpad Name and Description

In the next page, type the name of this taskpad, as it will be displayed as the title of the view, along with a description. In this example, we type "ClearCase Jobs on atrium" for the name, and "All ClearCase jobs scheduled on server `atrium'", as shown in Figure 20. Click **Next** to show the Completion page.

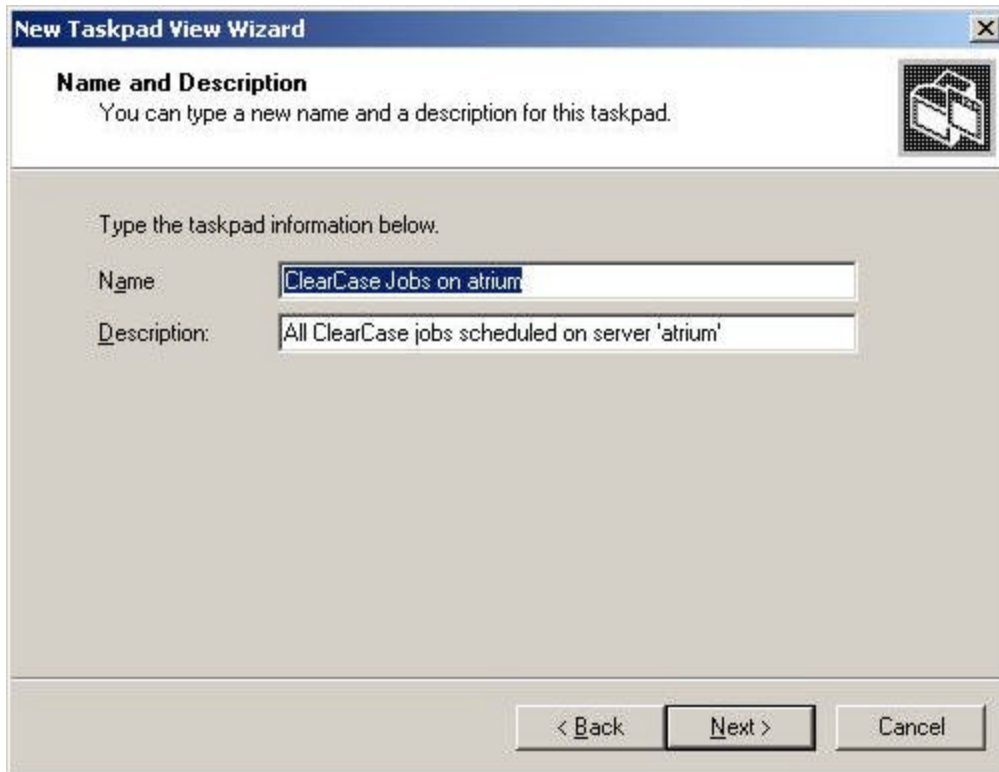


Figure 20 Name and Description

The Completion page allows us to start adding tasks to our task pad. Leave the Start New Task wizard check box selected, and click **Finish** to end this wizard and start the next.

Step 5: Add a Taskpad Task

The New Task wizard is shown as in Figure 21.



Figure 21 Welcome to the New Task Wizard

Click **Next** on the Welcome page to show the Command Type page as seen in Figure 22.

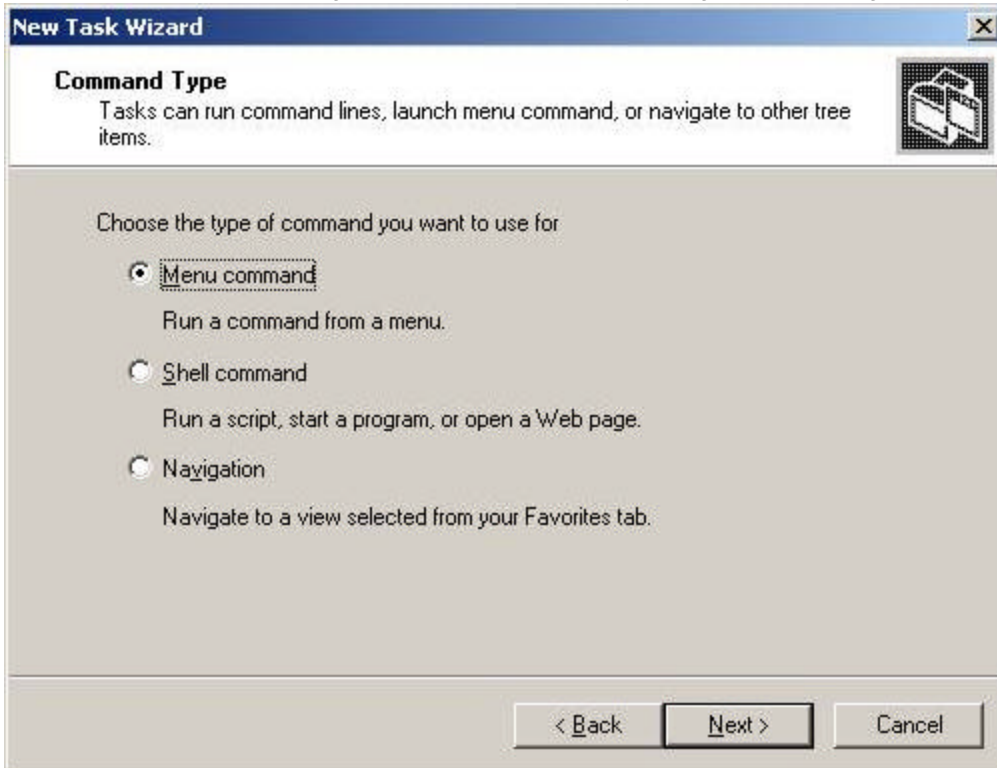


Figure 22 Command Type Page

On the Command Type page, you can add tasks that refer to scheduled jobs menu items, run shell scripts or commands parameterized with data specific to a selection in the taskpad, or you can navigate to another place in the namespace. In this example, we choose a menu command. Click **Menu command** and then click **Next**. This brings up the Shortcut Menu Command page, as shown in Figure 23.

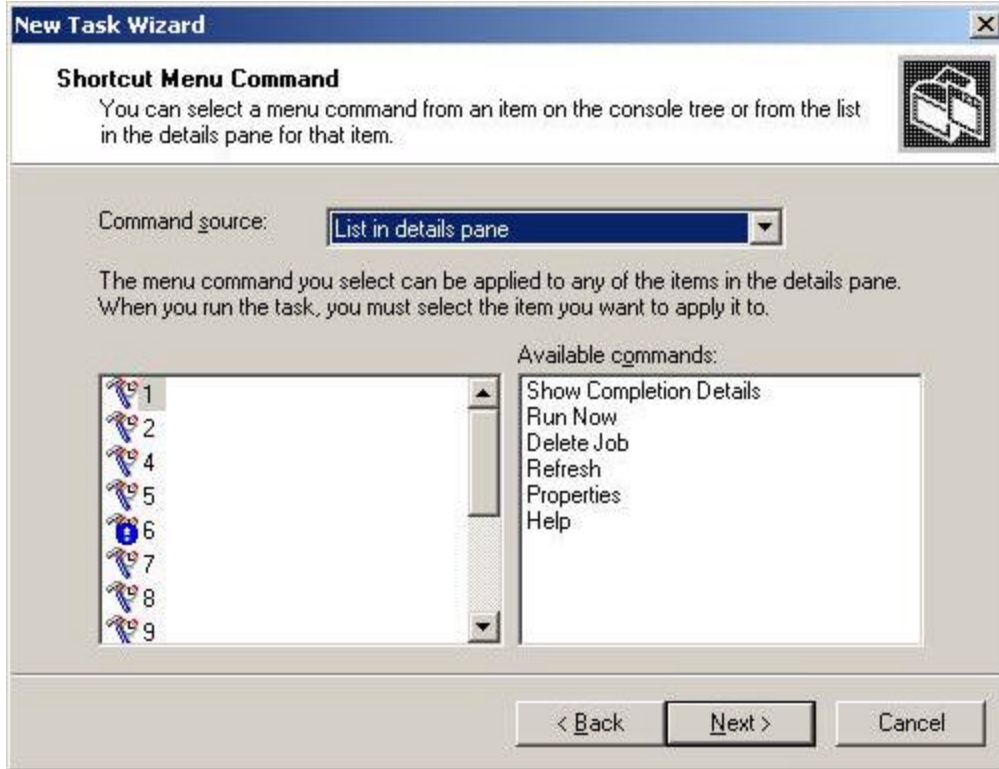


Figure 23 Shortcut Menu Command Page

The Shortcut Menu Command page allows you to associate commands that are available in the tree or details view with HTML buttons and descriptions. In this example, we'll add a task for editing the ACL (access control list) of the scheduler.

In the **Command source** box, select **Tree item task**. The Console tree list shows the Scheduled Jobs node selected; leave this selected. In the **Available commands** list, select **All Tasks->Edit Permissions** and then click **Next**.

The Name and Description page is shown in Figure 24.



Figure 24 Name and Description page

In the **Task name** box type "Schedule Permissions". In the **Description** box type "Modify the access control list (ACL) that allows users or groups to modify the ACL itself or the properties of scheduled jobs". >Click **Next**. The Task Icon page is shown as in Figure 25.

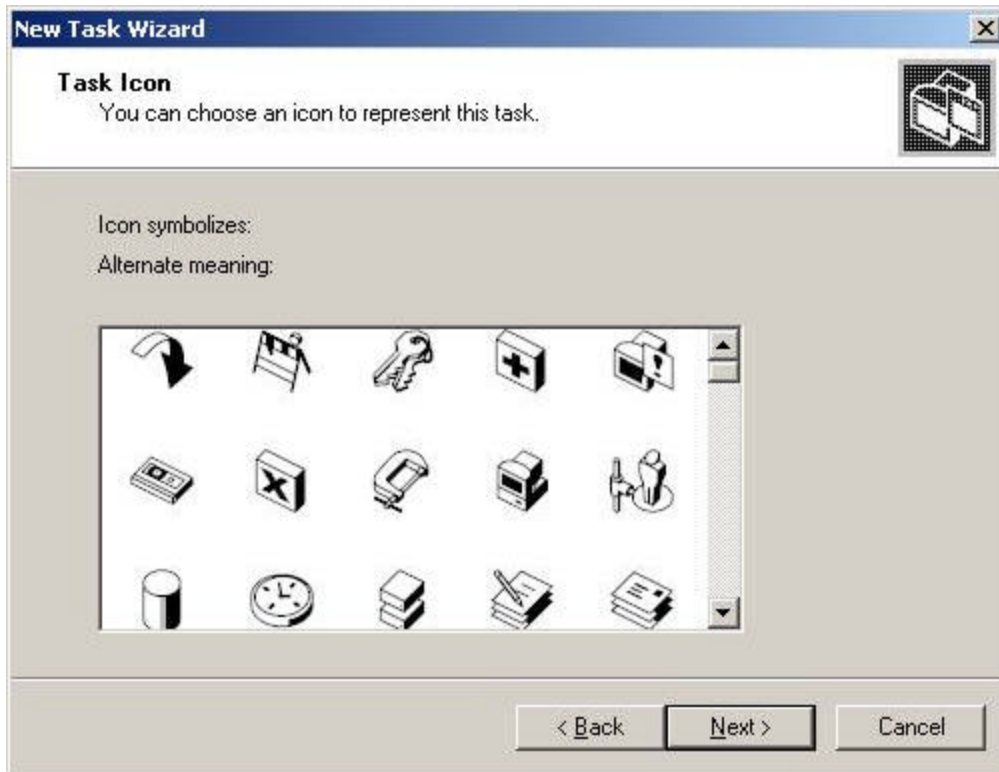


Figure 25 Task Icon page

Choose a symbol that best represents the associated task. In this example, we select the keys icon to represent security. Click **Next**.
The Completing the New Task Wizard page is shown as in Figure 26.>



Figure 26 Completing the New Task Wizard page

Because we don't want to add a new task for this example, we clear the **Run this wizard again** check box, and click **Finish**. The main console is modified to show the newly created taskpad, as shown in Figure 27.

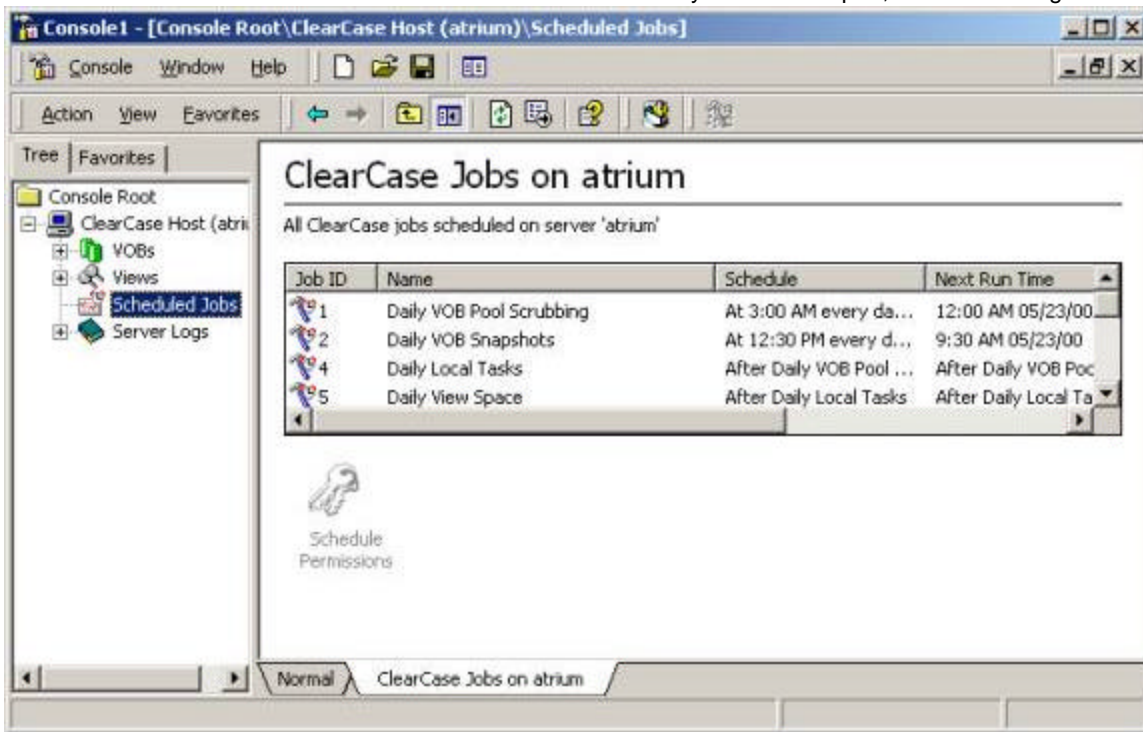


Figure 27 ClearCase Jobs on atrium Taskpad

Step 6: Remove the Console Tree

Now we're going to modify the namespace to show only scheduled jobs for this server and remove the (unnecessary) console tree. Select the Scheduled Jobs node, and then click **Action > New Window from Here**. Maximize the resulting window.

From the **View** menu, select **Customize**. The dialog box shown in Figure 28 is displayed.

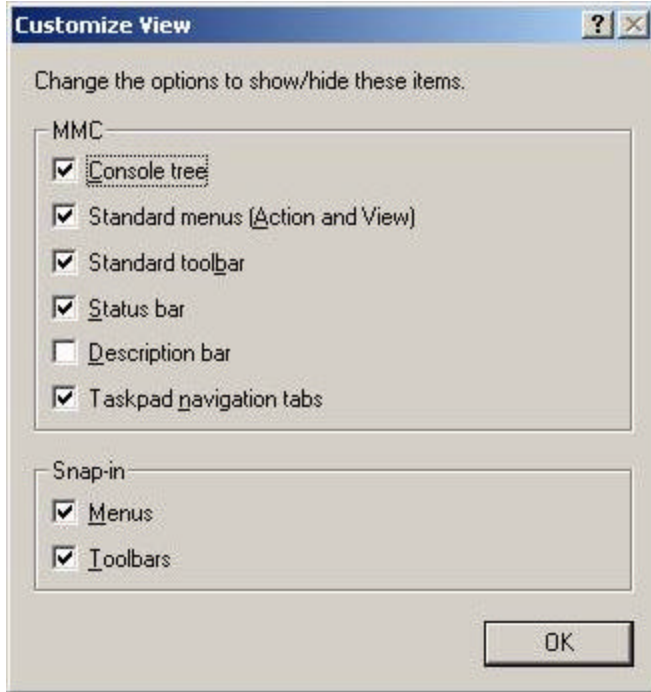


Figure 28 Customize View

Clear the **Console tree** check box and click **OK**. In the Action bar toolbar, click the toolbar button that shows and the console view to hide that view as well. (The button is to the right of the navigation arrows.) The display now appears as in Figure 29.

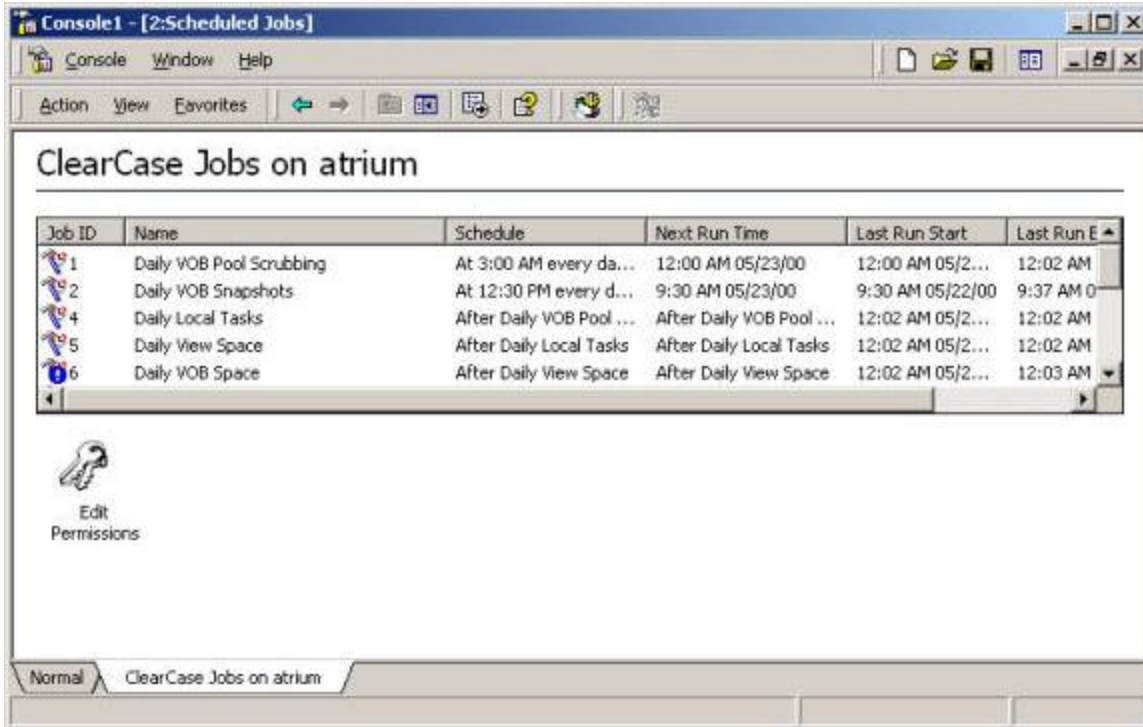


Figure 29 Scheduled Jobs without console tree

Step 7: Save the Console

Finally, modify the console for user mode operation only (Console/Options), and then save the console to your administrative tools folder. Starting the tool from your administrative folder results in the console as displayed in Figure 30.

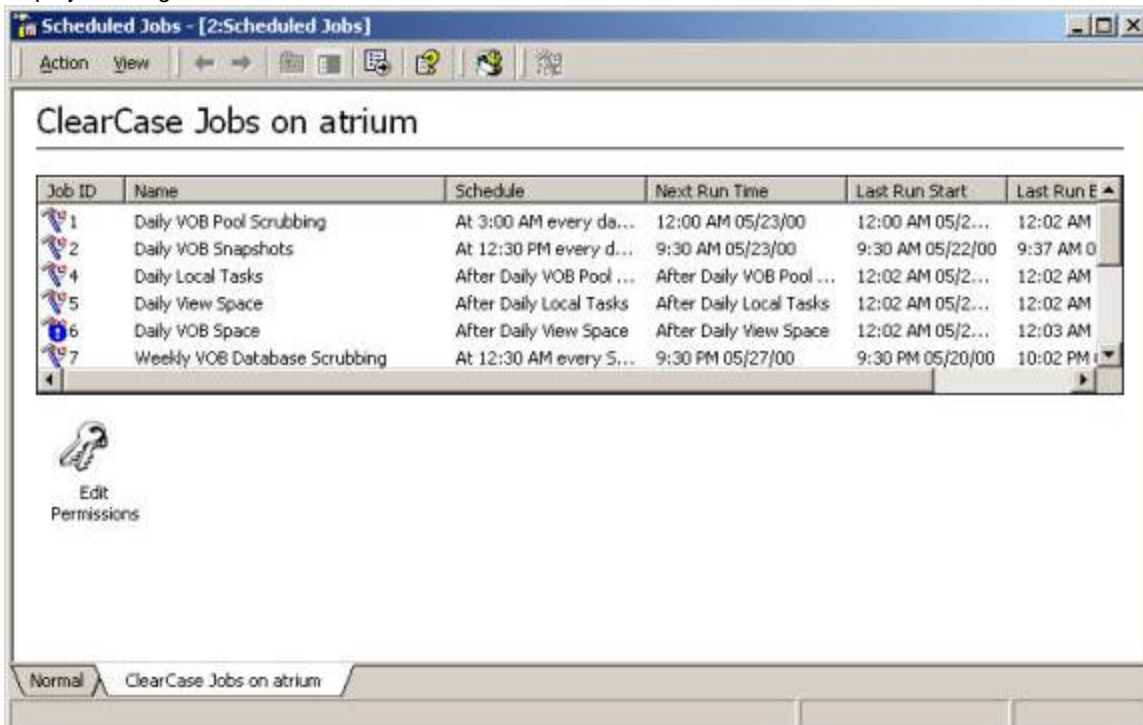


Figure 30 Scheduled Jobs saved console

Tutorial 4: Editing Multiple ClearCase Registries

Some ClearCase customers have installed multiple registry servers. Editing registry contents on multiple registry servers is facilitated with the ClearCase 4.0 administration tools. This tutorial can make use of either MMC 1.1 or MMC 1.2.

(Note: To edit the ClearCase registry, you must be a ClearCase privileged user, that is, the clearcase group must appear in your identity's group list. Otherwise, you will be limited to read-only access to the registry)

Step 1: Run MMC and Show the Add/Remove Snap-in Dialog

Run MMC. Click **Console > Add/Remove Snap-in**. From the resulting [Add/Remove Snap-in](#) dialog box, click **Add**. The [Add Standalone Snap-in](#) dialog box appears.

Step 2: Add Multiple ClearCase Registry Snap-ins

In the Add Standalone Snap-in dialog box, select **ClearCase Registry**, and click **Add**. The ClearCase Registry Snap-in manager dialog box appears, as in Figure 31.

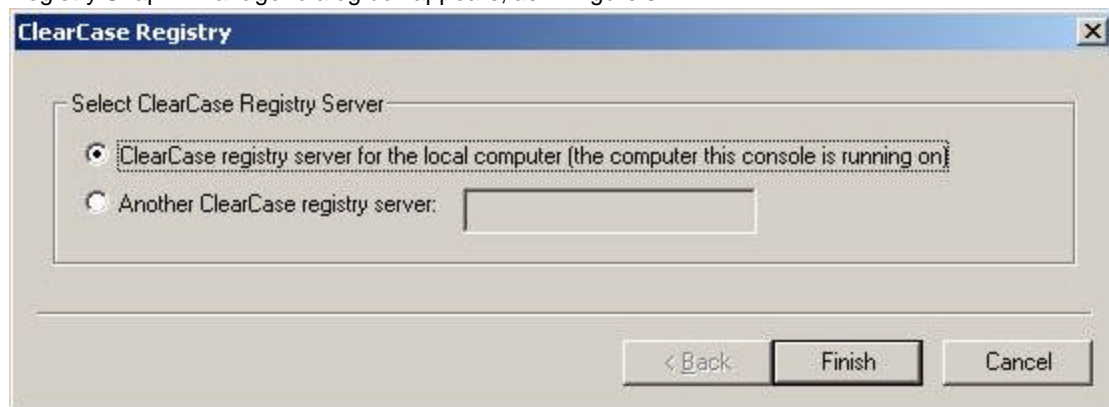


Figure 31 ClearCase Registry Snap-in Manager dialog box

To display the ClearCase Registry namespace for the default registry server, click the **ClearCase registry server for the local computer** button and click **Finish**. A ClearCase Registry namespace for the default registry server is added to the Add/Remove Snap-in work area.

Select **ClearCase Registry** from the Add Standalone Snap-in dialog box again, and click **Add**. This time, click **Another ClearCase registry server**, and type the name of a host that you know is another registry server. Click **Finish** and note that another ClearCase Registry namespace has been added to the Add/Remove Snap-in work area.

Finally, click **OK** on the Add/Remove Snap-in dialog box to apply the edits to the console window, and see a display similar to that shown in Figure 32.

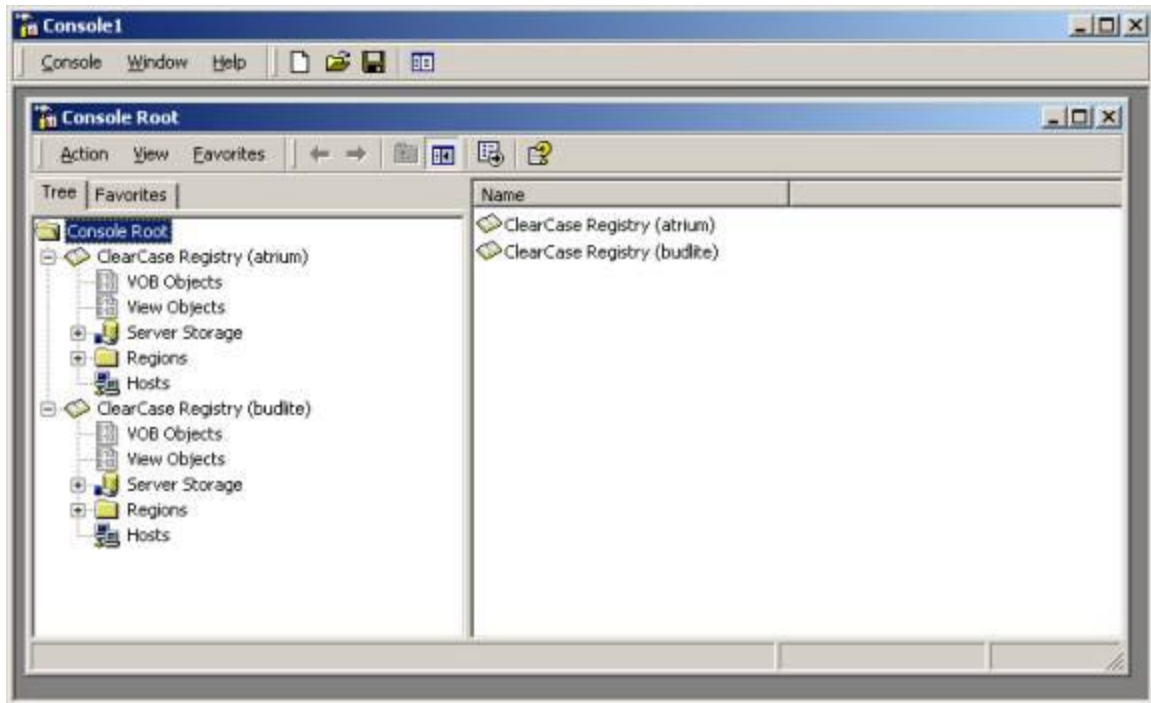


Figure 32 Console window with multiple registries

Step 3: Add Another Window

It's sometimes useful to have multiple windows open on the same display. In this example, we'll open two windows to display the registry namespaces, to facilitate editing between the registries.

First, rename the Console Root node to ClearCase Registries. Either edit the node name in place, or click **Action > Rename** while the Console Root node is selected. Next, click **Window > New Window**. A duplicate of the first console window is opened. Click **Window > Tile Horizontally** to arrange the two consoles horizontally in the MMC application window. The display should appear as in Figure 33.

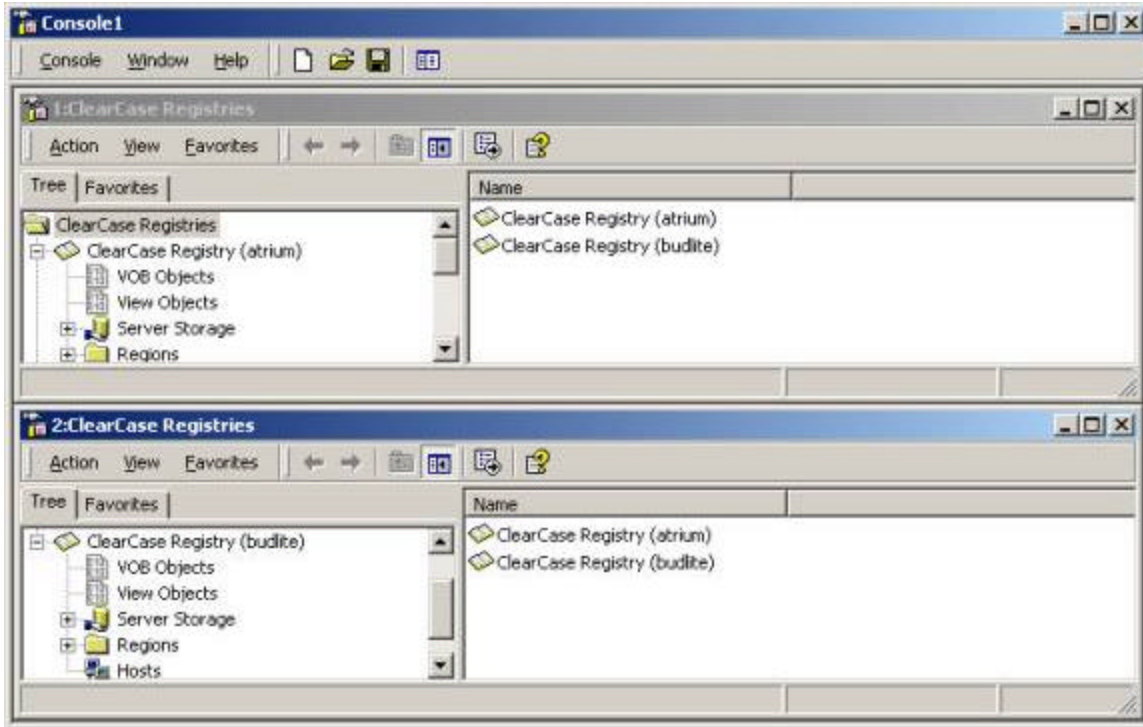


Figure 33 Multiple console windows viewing the ClearCase registry snap-ins

Step 4: Save the Console File

As in previous tutorials, modify the console author mode and then save the console to your administrative tools folder. However, in this case you'll save the console as User mode/limited access, multiple window. This allows the user to see the multiple windows that you just created (and create more), but not close the originals. When saving the console and the file, give it a descriptive name. In this case, we use Multiple Registries. The saved console file, when opened, appears as in Figure 34.

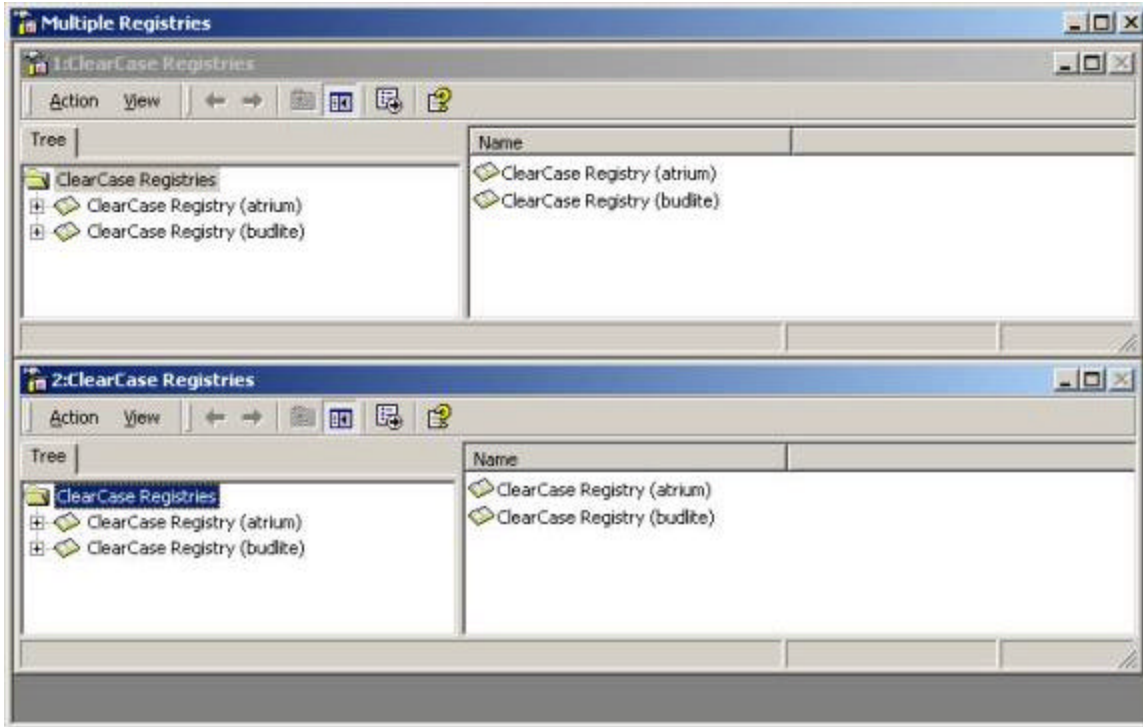


Figure 34 Multiple Registries saved console file ý user mode, limited access

Conclusion

These tutorials provide a glimpse of the many ways to customize the ClearCase 4.0 administration tools using the MMC. Other features implied, but not explored, are column customization, multi-window views, copying tags and objects from one registry snap-in to another, and creating custom commands from a taskpad view. Go ahead on your own and try new tasks. For example, create a taskpad command on the registry snap-in that will remove all references to a view from all VOBs known to that system. Finally, the concept of extension snap-ins has not been explored here. As previously mentioned, the MMC architecture allows for snap-ins to extend the functionality of one another. One example of this extensible architecture is shown via the Computer Management Snap-in (**My Computer**) in Windows 2000 (Figure 35). Selecting **My Computer** on the Windows desktop and then selecting **Manage** from its shortcut menu opens an MMC console on the current computer. Expand the Services and Applications console tree and note the inclusion of the ClearCase Host namespace. ClearCase extends the Computer Management Snap-in namespace including ClearCase-specific administrative functionality for that computer and any remote computer that is running ClearCase. Third parties and individuals, writing in either C++ or Visual BASIC, can create custom extensions to the ClearCase snap-ins using node IDs and custom clipboard formats that ClearCase exports. Refer to the Microsoft Platform SDK on writing MMC applications for more information on how to extend snap-ins.

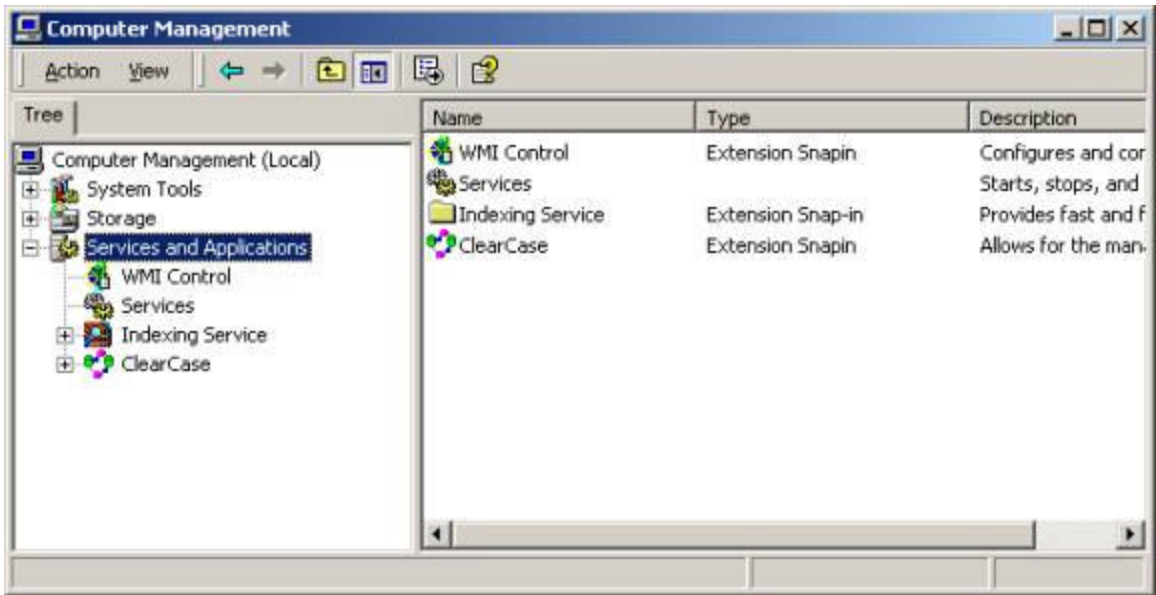


Figure 35 Computer Management with ClearCase Extension

[END]



IBM software integrated solutions

IBM Rational supports a wealth of other offerings from IBM software. IBM software solutions can give you the power to achieve your priority business and IT goals.

- *DB2[®] software helps you leverage information with solutions for data enablement, data management, and data distribution.*
- *Lotus[®] software helps your staff be productive with solutions for authoring, managing, communicating, and sharing knowledge.*
- *Tivoli[®] software helps you manage the technology that runs your e-business infrastructure.*
- *WebSphere[®] software helps you extend your existing business-critical processes to the Web.*
- *Rational[®] software helps you improve your software development capability with tools, services, and best practices.*

Rational software from IBM

Rational software from IBM helps organizations create business value by improving their software development capability. The Rational software development platform integrates software engineering best practices, tools, and services. With it, organizations thrive in an on demand world by being more responsive, resilient, and focused. Rational's standards-based, cross-platform solution helps software development teams create and extend business applications, embedded systems and software products. Ninety-eight of the Fortune 100 rely on Rational tools to build better software, faster. Additional information is available at www.rational.com and www.therationaledge.com, the monthly e-zine for the Rational community.

Rational is a wholly owned subsidiary of IBM Corp. (c) Copyright Rational Software Corporation, 2003. All rights reserved.

IBM Corporation
Software Group
Route 100
Somers, NY 10589
U.S.A.

Printed in the United States of America
01-03 All Rights Reserved.
Made in the U.S.A.

IBM the IBM logo, DB2, Lotus, Tivoli and WebSphere are trademarks of International Business Machines Corporation in the United States, other countries, or both.

Rational, and the Rational Logo are trademarks or registered trademarks of Rational Software Corporation in the United States, other countries or both.

Microsoft and Windows NT are registered trademarks of Microsoft Corporation in the United States, other countries, or both.

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

ActionMedia, LANDesk, MMX, Pentium and ProShare are trademarks of Intel Corporation in the United States, other countries, or both.

UNIX is a trademark of The Open Group in the United States, other countries or both.

Other company, product or service names may be trademarks or service marks of others.

The IBM home page on the Internet can be found at ibm.com