

Fedora Administrator

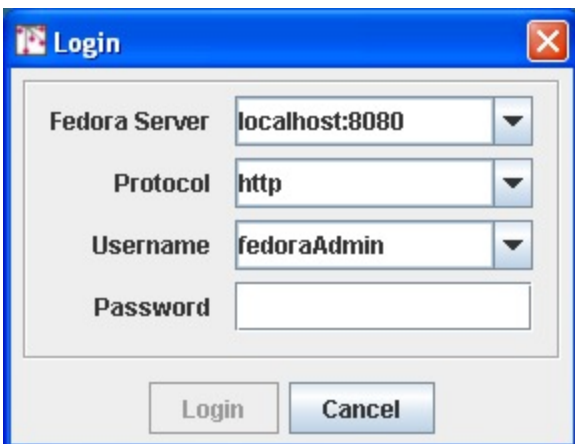
- [Introduction](#)
- [Starting Fedora Administrator](#)
- [File Menu](#)
 - [Creating a New Object](#)
 - [Create a New Data Object](#)
 - [View and Modify the Default Dublin Core Datastream](#)
 - [Create a New Datastream in the Object](#)
 - [Opening an Object for Viewing, Editing, Export, and Purge](#)
 - [The Object Properties Pane](#)
 - [The Datastreams Pane](#)
 - [Editing Datastream Content](#)
 - [Ingesting Objects](#)
 - [Ingest One Object](#)
 - [Ingest From File](#)
 - [Ingest From Repository](#)
 - [Ingest Multiple Objects](#)
 - [Ingest From Directory](#)
 - [Exporting Objects](#)
 - [Export One Object](#)
 - [Export Multiple Objects](#)
 - [Purging Objects](#)
 - [Viewing Object XML](#)
 - [Changing Repository](#)
 - [Exiting the Fedora Administrator](#)
- [Tools Menu](#)
 - [Searching and Browsing the Repository](#)
 - [Simple Search Tab](#)
 - [Advanced Search Tab](#)
 - [Search Results Window](#)
 - [Batch Processing](#)
- [Window Menu](#)
- [Help Menu](#)
- [Appendix A: Digital Object Construction](#)

Introduction

Fedora Administrator is the direct link to API-M functionality for repository administrators. Using this tool it is possible to ingest, search for and retrieve, modify and purge objects.

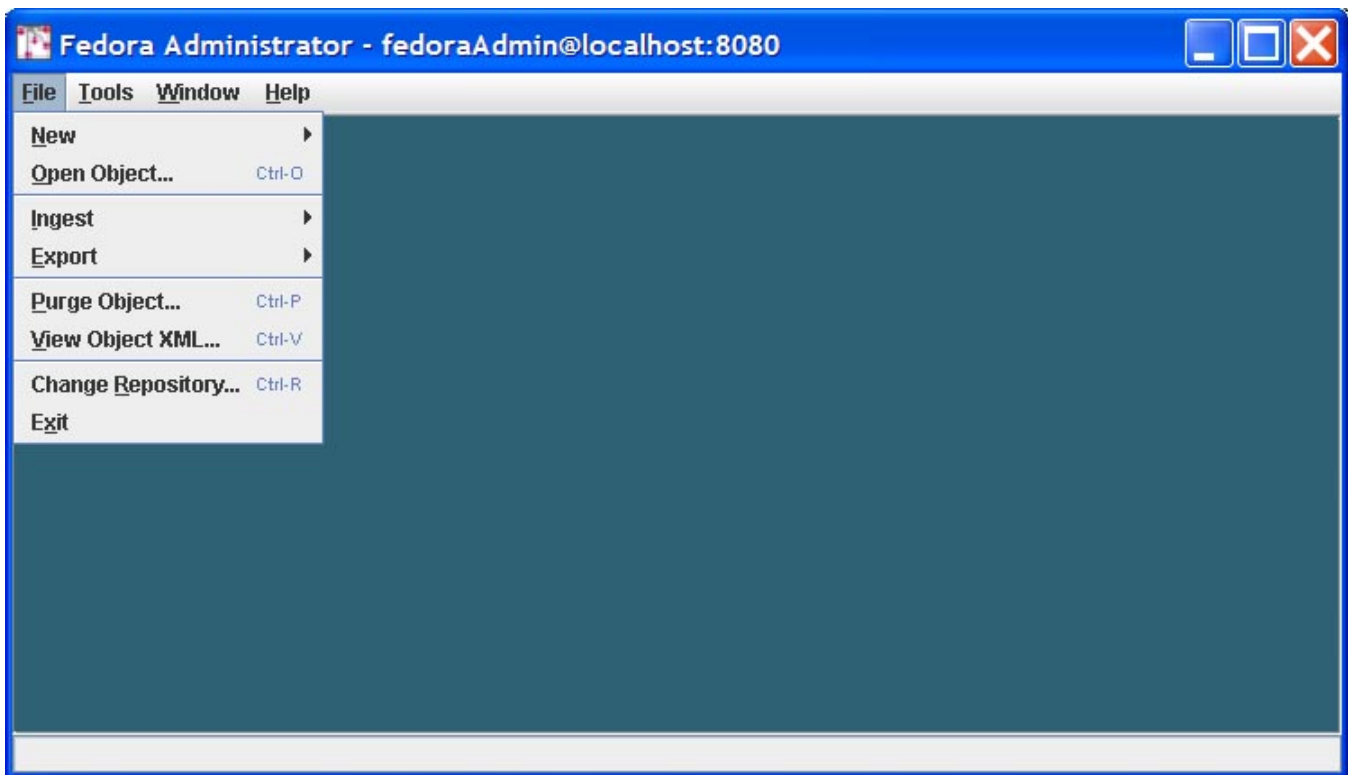
Starting Fedora Administrator

Navigate to the `$FEDORA_HOME/client/bin` directory and execute either the `fedora-admin.bat` batch file (for Windows) or the `fedora-admin.sh` script (for Unix/Linux).



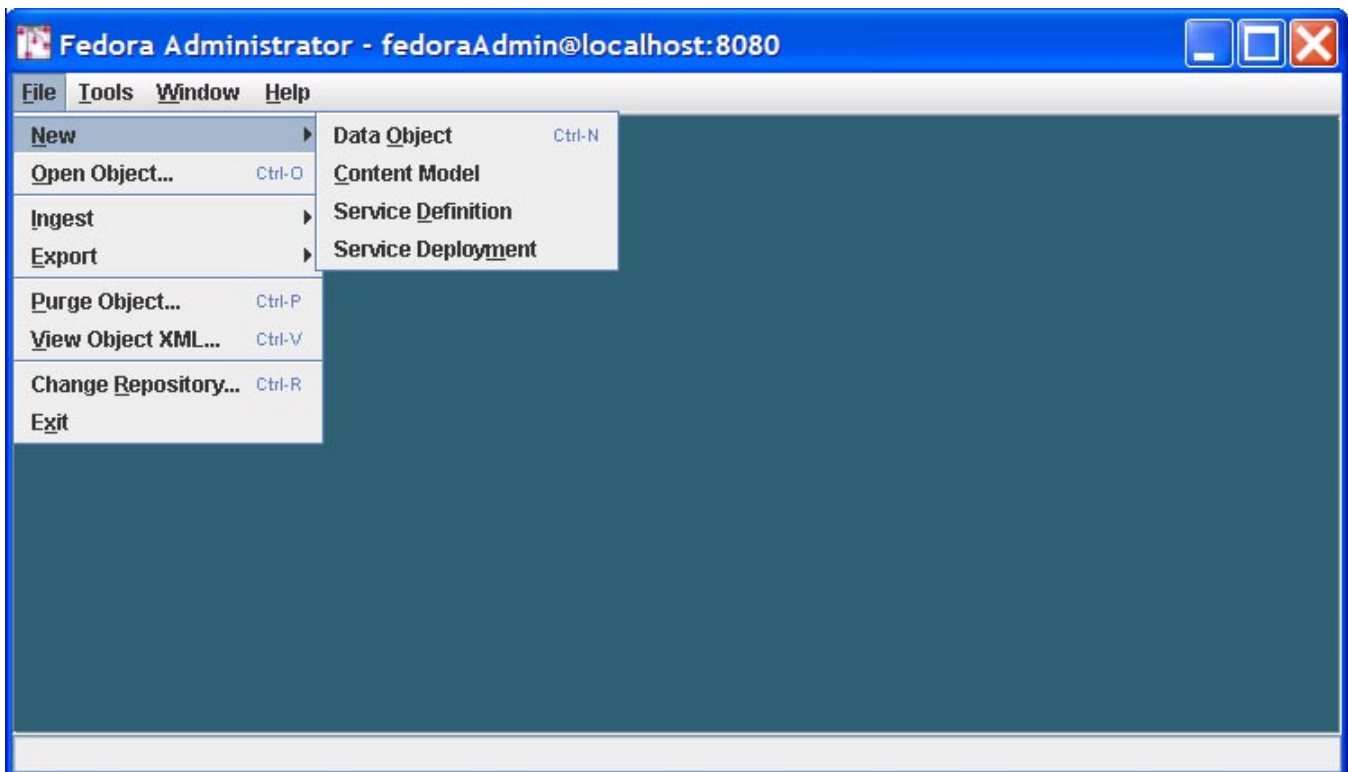
When you start the Fedora Administrator, you will be asked to choose the server to which you wish to connect, the protocol you wish to use to connect to your Fedora repository (http or https), your username and password. The server and username fields are pre-populated and the password is validated using values from `fedora.fcfg`.

File Menu



Commands on the file menu allow a repository administrator to perform operations on objects in the repository or to log in to a different repository.

Creating a New Object



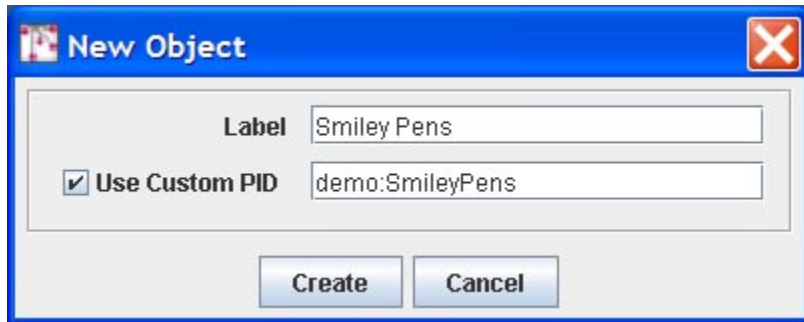
The *New* menu option allows users to build new Fedora objects from pre-existing component parts. When any of the object types (Data Object, Content Model, Service Definition, or Service Deployment) are created using the *New* option a skeletal Fedora object specific to that type is ingested into the repository. This allows the repository administrator or object owner to complete the object by defining metadata, asserting relationships, updating the provided datastreams, and adding new datastreams.

Create a New Data Object

On the File Menu, select *New*.

From the New submenu, select *Data Object*.

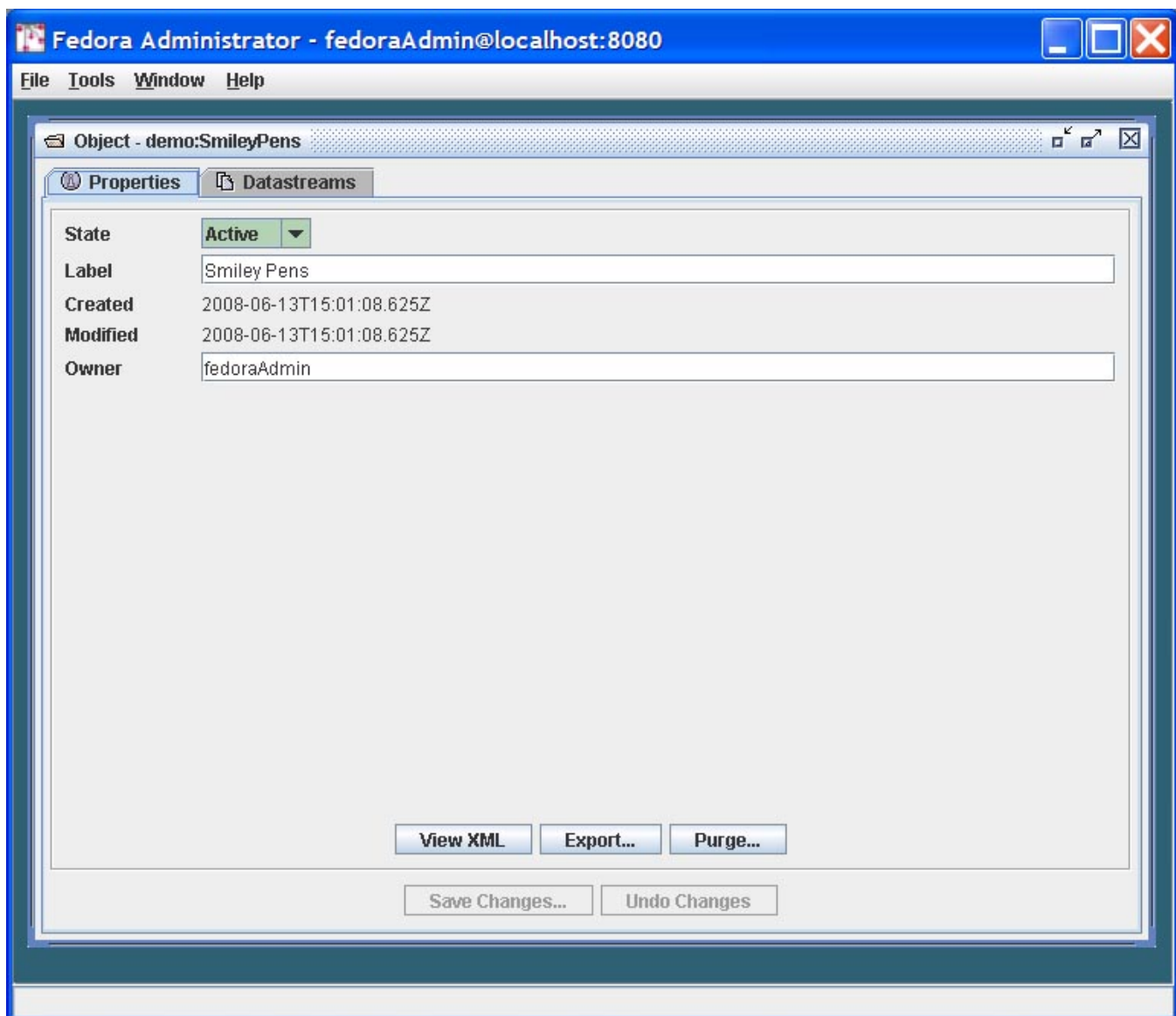
The New Object Dialog appears.

A screenshot of the 'New Object' dialog box. The dialog has a blue title bar with the text 'New Object' and a red 'X' button. Inside, there is a 'Label' text field containing 'Smiley Pens'. Below it, there is a checked checkbox labeled 'Use Custom PID' and a corresponding text field containing 'demo:SmileyPens'. At the bottom, there are two buttons: 'Create' and 'Cancel'.

Fill in the label for the object. If a custom PID is desired, click the check box and fill in the PID value.

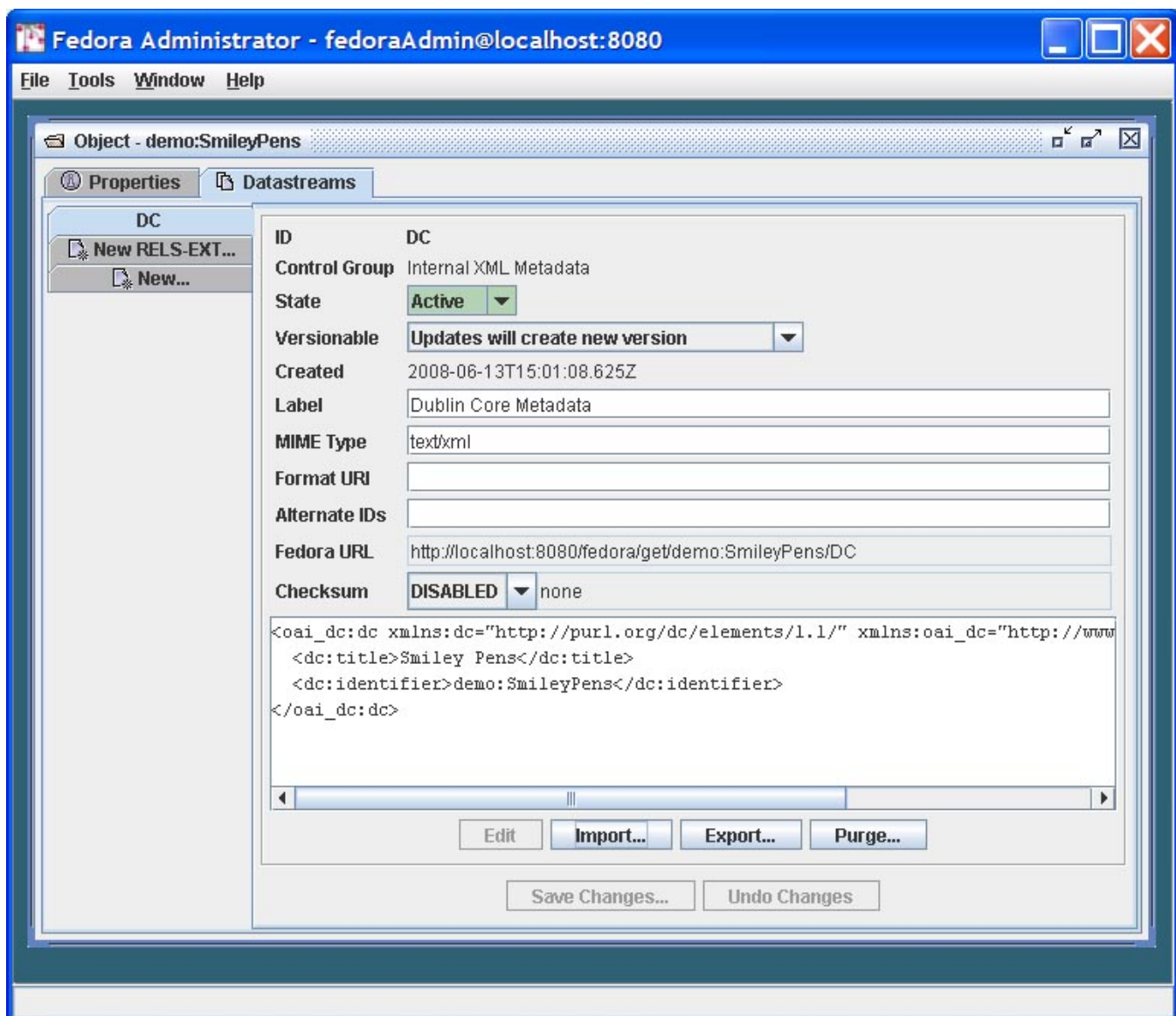
Clicking *Create* causes a Fedora data object to be created and ingested into the repository. The new object is displayed in a tabbed pane showing two tabs: *Properties* and *Datastreams*. You will first be presented with the Object Properties pane. On this pane you can add the following information:

- **State:** The state of the object, defined as follows:
 - *Active:* the object is available to users conditional upon any access control policy restrictions
 - *Inactive:* the object is only available to repository administrators.
 - *Deleted:* the object has been marked for permanent removal from the repository, pending review by repository administrators.
- **Label:** A description of the digital object
- **Owner:** An identifier for the owner of the object



View and Modify the Default Dublin Core Datastream

Click on the *Datastreams* tab labeled as "**DC**." The display will show a side tabbed pane displaying all datastreams currently in the object. When first created, all Fedora data objects will contain a default Dublin Core metadata datastream. You can edit this metadata datastream, adding fields as appropriate for the data object in question. To add new Dublin Core elements, edit the XML content for the datastream in the editing window. When done editing, click the Save Changes button on the bottom of the pane.



Create a New Datastream in the Object

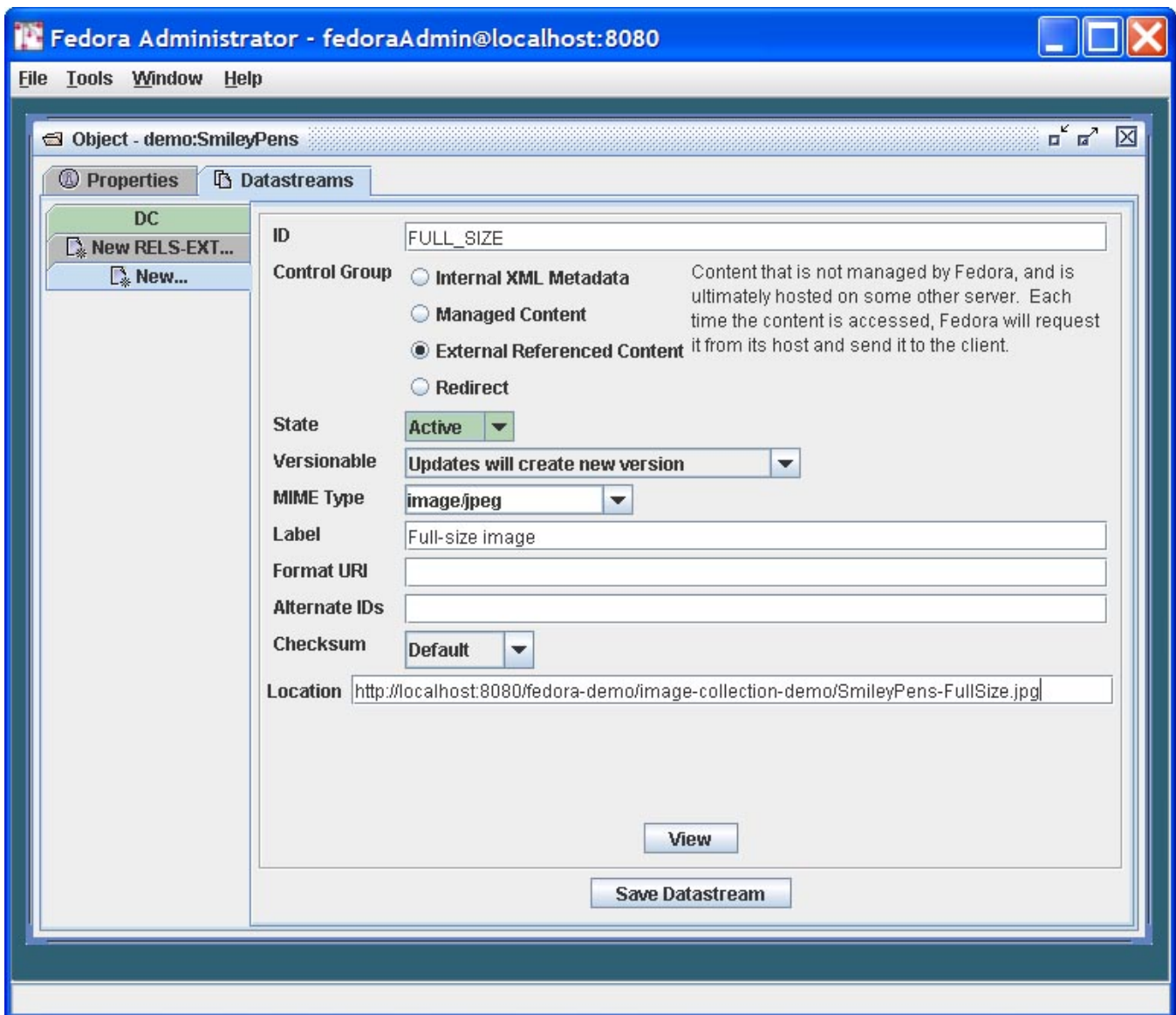
To create a new datastream in the digital object, click on the side tab labeled "New..." You are presented with a dialog window that provides data input fields to enter all attributes for a datastream. Enter the following information:

- **ID:** Enter a custom identifier for the datastream. If left blank, the system will automatically assign a unique datastream id.
- **Control Group:** Clicking through the control group options will show text defining each control group option. The Control Group lets you decide how you want the datastream content to be associated with the digital object. Datastream content can be in-lined within the object XML container file, it can be stored in the repository, or it can be stored external to the repository (and digital objects point to the external location).
 - **Internal XML Metadata:** In this case, the datastream will be stored as XML that is actually stored inline within the digital object XML file. The user may enter text directly into the editing window or data may be imported from a file by clicking *Import* and selecting or browsing to the location of the XML metadata file.
 - **Managed Content:** In this case, the datastream content will be stored in the Fedora repository and the digital object XML file will store an internal identifier to that datastream. To get content, click *Import* and select or browse to the file location of the import file. Once import is complete, you will see the imported file in a preview box on the screen.
 - **External Referenced Content:** In this case, the datastream content will be stored outside of the Fedora repository, and the digital object will store a URL to that datastream. The datastream is "by reference" since it is not actually stored inside the Fedora repository. While the datastream content is stored outside of the Fedora repository, at runtime, when an access request for this type of datastream is made, the Fedora repository will use this URL to get the content from its remote location, and the Fedora repository will mediate access to the content. This means that behind the scenes, Fedora will grab the content and stream it out to the client requesting the content as if it were served up directly by Fedora. This is a good way to create digital objects that point to distributed content, but still have the repository in charge of serving it up. To create this type of datastream, specify the URL for the datastream content in the Location URL text box.
 - **Redirect:** In this case, the datastream content is also stored outside the repository and the digital object points to its URL ("by-reference"). However, unlike the External Referenced Content scenario, the Redirect scenario signals the repository to redirect to the URL when access requests are made for this datastream. This means that the datastream will not be streamed through the Fedora repository when it is served up. This is beneficial when you want a digital object to have a datastream that is stored and served up by

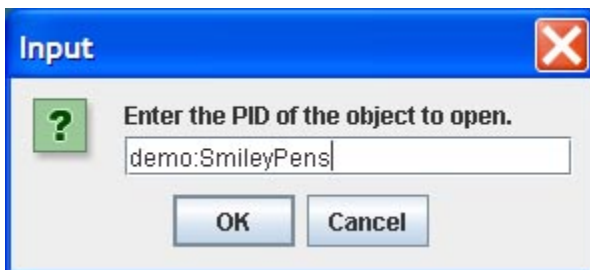
some external service, and you want the repository to get out of the way when it comes time to serve the content up. A good example is when you want a datastream to be content that is stored and served by a streaming media server. In such a case, you would want to pass control to the media server to actually stream the content to a client (e.g., video streaming), rather than have Fedora in the middle re-streaming the content out. To create a Redirect datastream, specify the URL for the content in the Location text box.

- **State:** Set the state for the digital object as Active, Inactive, or Deleted. The object states are defined as follows:
 - *Active:* the datastream is available to users conditional upon any access control policy restrictions
 - *Inactive:* the datastream is only available to repository administrators.
 - *Deleted:* the datastream has been marked for permanent removal from the repository, pending review by repository administrators.
- **Versionable:** You can decide on a datastream-by-datastream basis whether the Fedora repository will version the datastream when modification are made. By default all datastreams are versioned, but you can override this with the Versionable drop down settings. If you select "Updates will create new version," a new version of the datastream will be created when any change is made. All previous versions of the datastream will be maintained. If you select "Updates will replace most recent version" then any changes will overwrite the most recent version of the datastream. In the "replace" case, if there is already a previous version history of the datastream it will be maintained (it will not be lost if you change to the "replace" option at some point in the future after you have been versioning a datastream).
- **MIME Type:** Select the MIME type for the datastream content.
- **Label:** Give your datastream a descriptive label.
- **Format URI:** Optionally, you can provide a format identifier for the datastream. Examples of emerging format identifier schemes found in [PRONOM](#) and the [Global Digital Format Registry](#) (GDRF).
- **Alternate IDs:** Optionally, you can assign one or more alternate identifiers for you datastream. Such identifiers could be local identifiers or global identifiers such as [Handles](#) or [DOI](#).
- **Checksum:** The Fedora repository has a global setting for enabling datastream checksums to be calculated by the repository when a datastream is created or modified. See the Fedora system documentation of repository configuration for details. Also, refer to the system documentation on Checksum Features for a complete discussion of this functionality. A checksum is calculated on datastream content (it does not include the various attributes of a datastream such as its identifier, state, or mime type). In the datastream creation dialog pane, you will see a drop down list where you can select a checksum algorithm. This gives you some control for how you would like the checksum handled for a particular datastream. (Or you can let the default repository behavior take over.) The following can be accomplished using the checksum data entry box:
 - If you select "Default" in the checksum drop down list, the repository will handle everything based on how it is configured. This means that if the repository is configured to automatically checksum datastreams, it will use a default checksum algorithm configured for the repository, it will calculate a checksum for the datastream, and store it.
 - If you select a particular checksum algorithm from the drop down list box, you are essentially telling the repository that whatever it has configured for datastream checksum, you wish to override that behavior for this particular datastream. By specifying an algorithm you are telling the repository to calculate a checksum for that datastream using the algorithm that you specify.
 - When you select a checksum algorithm from the drop down box, you will notice that a text entry box appears next to it. You can leave the box blank, and the repository will use the specified algorithm and calculate a checksum values. Another option is to provide the repository with a checksum value that you have already calculated. In this case, the repository service will still calculate a new checksum when it creates the datastream, but if you provide a value from the client end, the repository will use it as an integrity check to ensure that the client and the server both agree on the checksum value for the datastream content. The repository will calculate a checksum value, using the algorithm you specify, and then it will compare its checksum calculation with the checksum value you provided in the text box. If they match, that means that the datastream content was not altered in the transmission from the client to the repository service. If they don't match, that indicates that something happened to the datastream during the transaction, and the repository will throw an exception.
- **What about the Datastream Content?** Depending on the datastream control group that you selected, the user interface will present you with a different way to get the actual content for your datastream.
 - If you selected "Internal XML Metadata" you will see an editing window at the bottom of the pane for creating XML content. Alternatively, you can use the **"Import"** button to insert XML content from a file.
 - If you selected "Managed Content" you will see an **"Import"** button on the bottom of the pane. Click this button to select a file whose content will be pulled into the repository and stored as managed datastream content. The digital object will have an internal pointer to the repository stored content.
 - If you selected "External Referenced Content" a new text entry box labeled **"Location"** will appear where you are prompted to enter a URL for content stored external to the repository. The digital object will store this URL, and the content will NOT be copied into the repository. It will remain "by-reference." You will notice a "View" button appear at the bottom of the pane that allows you to preview the content.
 - If you selected "Redirect" you will also enter a URL in the **"Location"** text entry box.

When all information is supplied in the add Datastream pane, click the **Save Datastream** button. This will send an API-M request to the Fedora repository service to add the new Datastream to the digital object. A completed Datastream pane is depicted below.



Opening an Object for Viewing, Editing, Export, and Purge



The *Open* menu requires input of an object PID for retrieval of the object. Upon retrieval, the object is displayed on two tabbed panes: Properties, Datastreams. You can edit any aspect of a digital object from these panes, in the same way information was added via the panes (as described earlier). Some additional functions are available on the Object Properties pane for an existing digital object:

The Object Properties Pane

- **Viewing XML:** From this window, the XML of the object can be viewed, but cannot be modified.
- **Export Object:** If Export is selected, the user will be prompted for a file name and location to which to write the XML file. The format for exported objects is fedora:mets.
- **Purge Object:** Purging an object completely removes it from the repository. Upon selecting the Purge option, the user will be prompted to enter a reason for the object's removal.

Warning

There is no "Are you sure" dialog with this option and the purge cannot be undone.

The Datastreams Pane

On the Datastreams Pane, the state of each datastream in the object can be modified, along with the datastream label and location. The MIME type of the datastream is shown, along with the control group, info type, create date, and the Fedora URL of the object.

- **Datastream States**
 - *Active*: the datastream is freely available to all users.
 - *Inactive*: the datastream is only available to repository administrators.
 - *Deleted*: the datastream has been marked for permanent removal from the repository, pending review by repository administrators.

From this pane, users may additionally request to view a datastream, add a new datastream to the object, export a datastream's contents, or purge the datastream from the object.

Editing Datastream Content

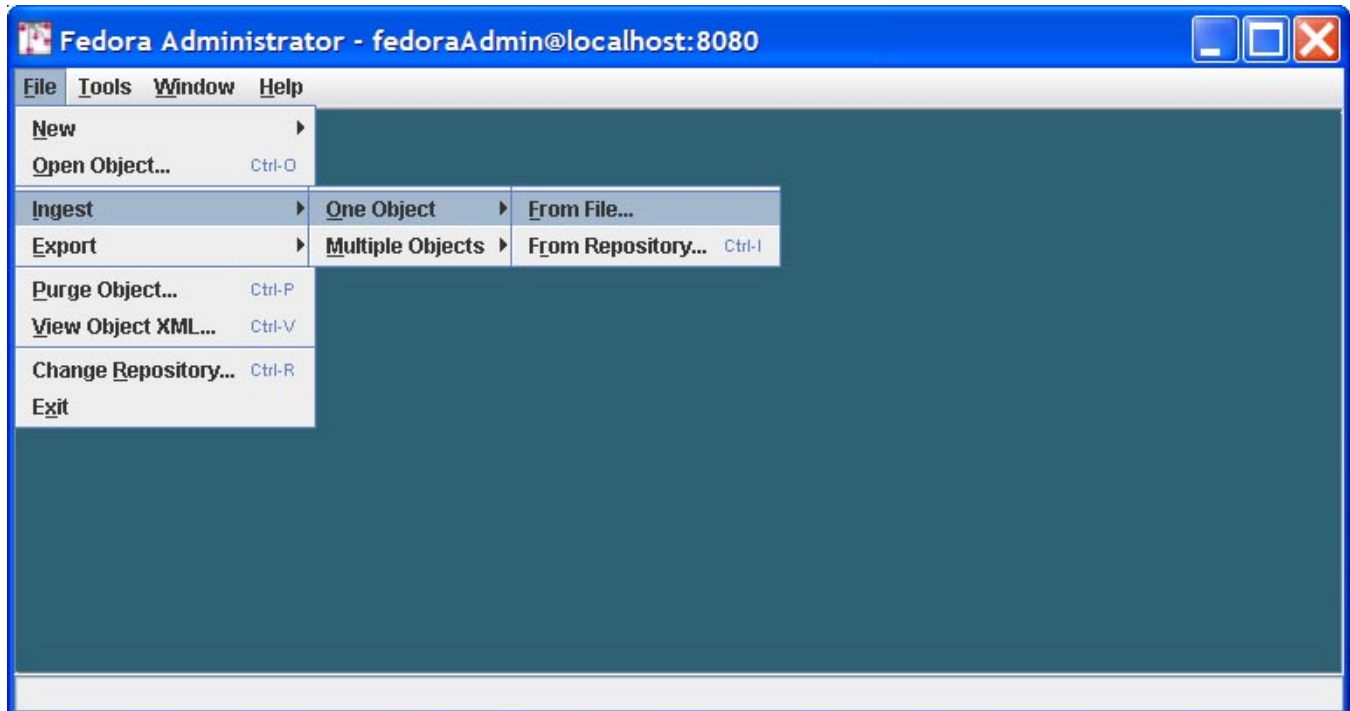
If a datastream has a *text* MIME type (e.g., *text/xml*), it may be edited in place by clicking the *Edit* button and making the desired changes in the editing window. Export the data contained in the datastream by clicking the *Export* button. Datastreams with non-text MIME types (e.g., *image* or *application*, e.g., *image/jpeg* or *application/pdf*) may only be viewed, exported, or purged from the object.

- **Import**: Choose this button to import new data by clicking the *Import* button. You will be prompted for a file name or url where the the import file is located.
- **Export**: Choose this button to export the content of a datastream. You will be prompted for a file name and location to which to write the XML file of the datastream content.
- **Purge**: Choose this button to remove a datastream from the object. You will be warned that the operation is permanent and must click "Yes" to continue. If "Yes" is selected, the datastream is immediately purged from the object. Purge cannot be performed on the DC metadata datastream.

Ingesting Objects

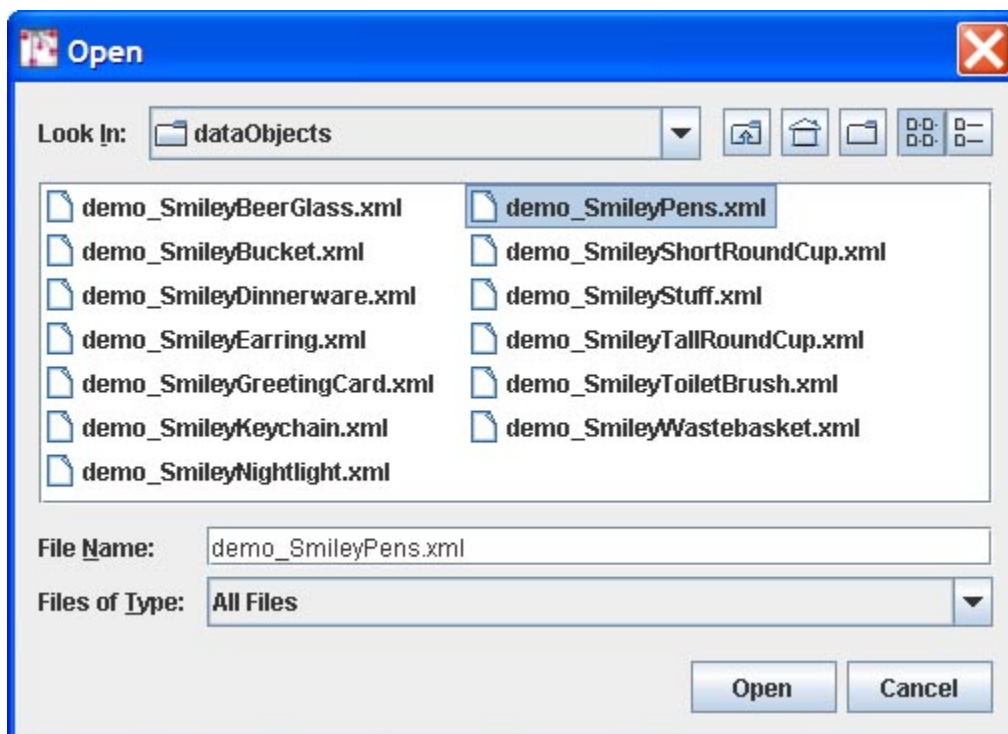
When selecting to ingest objects from the File menu, users have the option of ingesting a single object or multiple objects. Objects may be ingested from a file, directory, or from another repository.

Ingest One Object



Ingest From File

Choosing Ingest One Object from File, the user is prompted to select the file name from a dialog box or browse to the location of the file on the local drive (s) for the file to be ingested. Clicking *Open* will cause the file to be ingested. If the repository has been set to retain PIDs on ingest in *fedora.fcgi*, the PID in the object XML will be maintained. Otherwise, the PID will be overwritten.



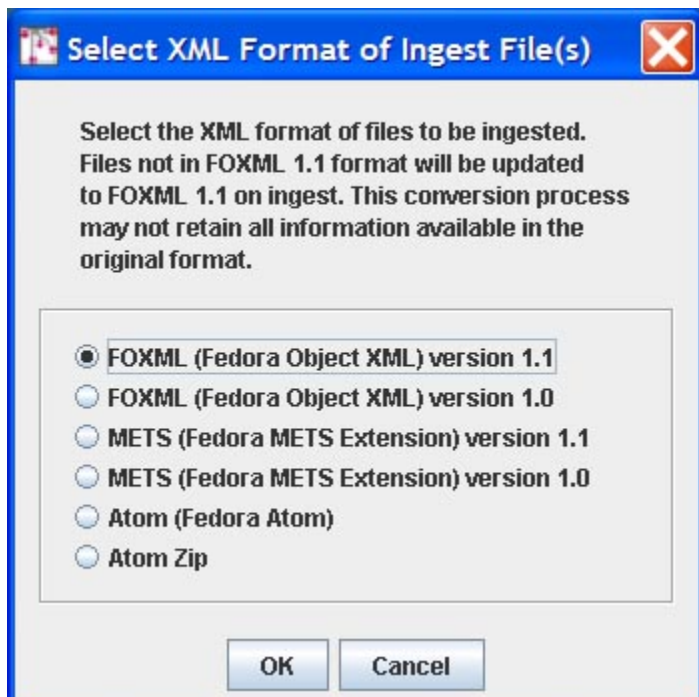
Ingest From Repository

Choosing Ingest One Object from Repository causes the Source Repository dialog box to appear. The user must fill in the hostname: port of the source repository, the protocol (http or https) and enter a username and password. Clicking OK initiates the Input dialog, where the user is prompted for a PID value. Clicking OK on the Input dialog completes the object ingestion.

Ingest Multiple Objects

Ingest From Directory

Upon selecting Ingest Multiple Objects From Directory, a dialog box prompts the user to select or browse to the directory containing the objects to be selected. Once the directory has been identified, clicking *Open* will activate a second dialog box which prompts the user to select the format of the objects to be ingested. All objects in the directory must be in the same format.

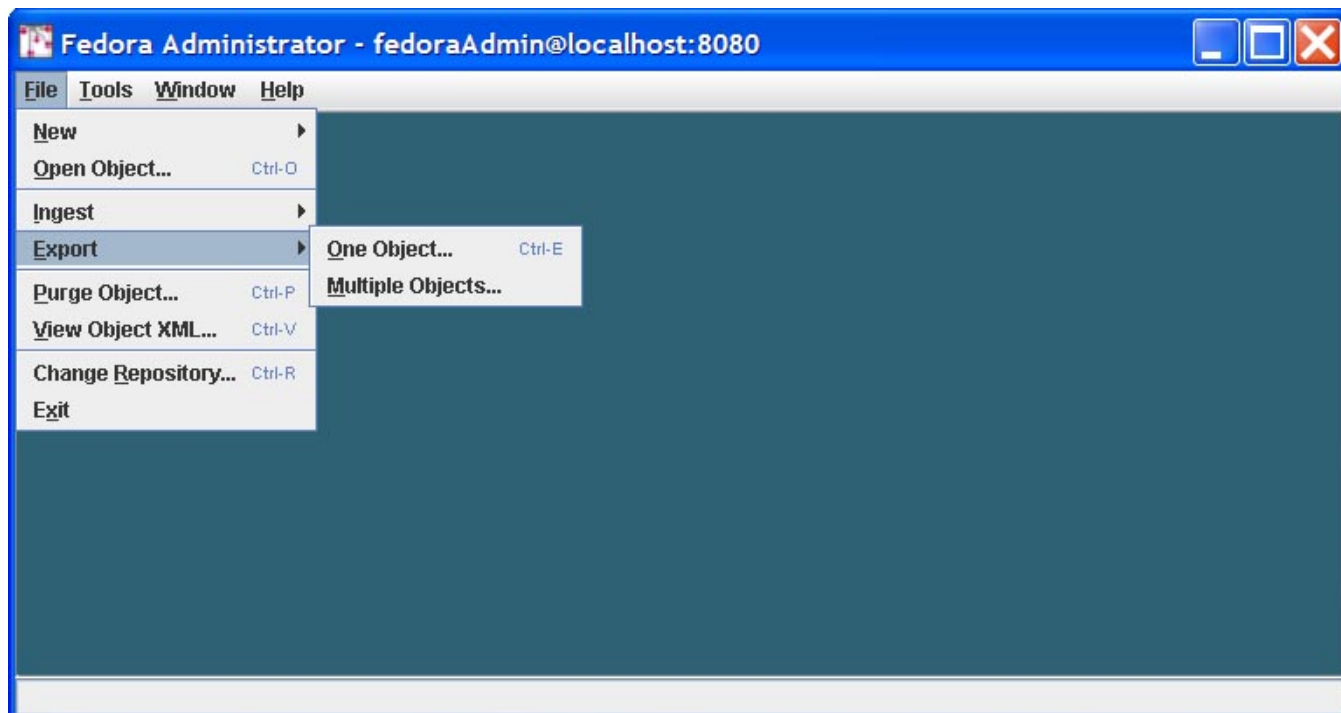


If the repository has been configured to "retain PIDs" on ingest (see the Fedora repository configuration documentation), then whatever PID is found in the ingest XML file (also known as the Submission Information Package or SIP) is what the repository accepts as the PID for the new ingested digital object. If the "retain PIDs" option is not enabled for the repository, the ingest function will automatically assign a new PID to the digital object, and ignore whatever PID was in the ingest XML file.

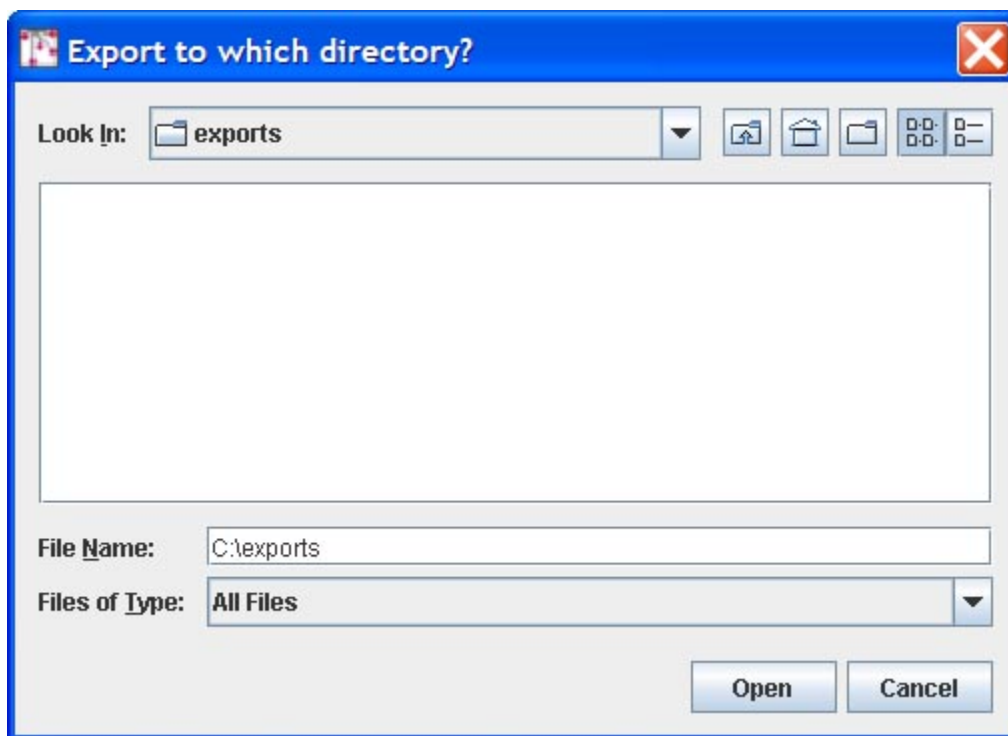
During ingest, the status bar at the bottom of the Fedora Administrator window shows the activities of the server. Once ingest is complete, a summary pane will appear giving counts of objects successfully ingested, objects failed, and time elapsed. Click *OK* to clear this message. The View Ingest Log dialog will then open. The user may click *Yes* and view the detailed log file or *No* to view the file at a later time. The log file is created in the `$FEDORA_HOME/client/logs/` directory.

Exporting Objects

Users have the option of exporting a single object or exporting multiple objects. This functionality can be accessed through the File/Export menu.



Export One Object

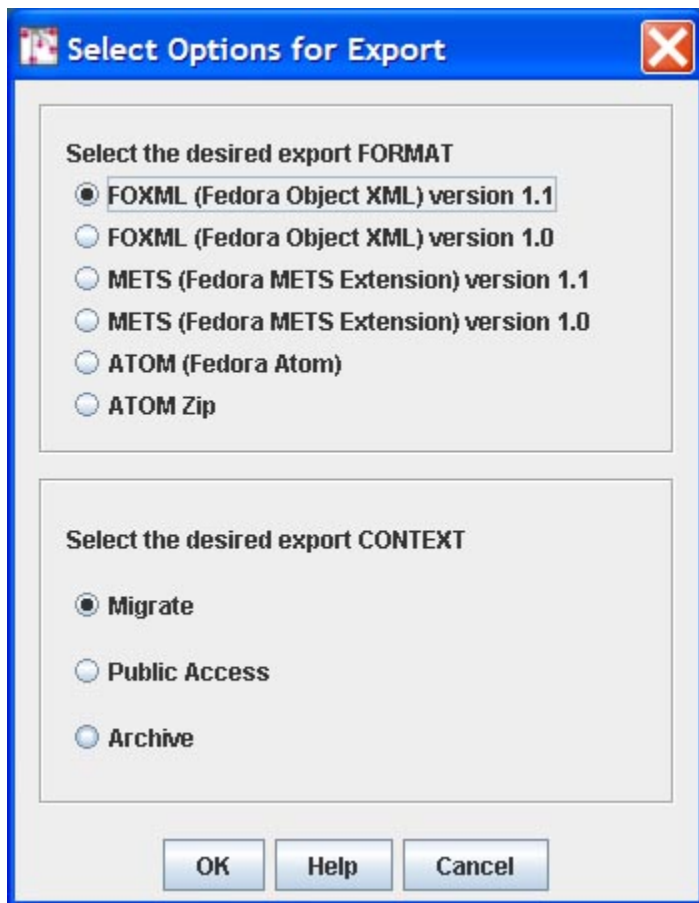


When the *Export One Object* option is selected, the user is prompted to select a directory to which the export file will be written. Clicking *Open* causes the user to be prompted for an object PID. The PID will be used as the basis for the export file (named like *pid.xml*). Before the export is done, you must first provide a few more pieces of information. First, you must choose the export serialization format:

- FOXML 1.1 (the most current FOXML format)
- FOXML 1.0 (the FOXML format used with pre-3.0 Fedora repositories)
- METS 1.1 (the most current Fedora extension of METS)
- METS 1.0 (the METS format used with pre-3.0 Fedora repositories)
- ATOM (the Fedora extension of Atom)
- ATOM ZIP (an ATOM based format which packages all datastreams along with the object XML in a ZIP file)

Next you must choose the "export context" which will create an appropriate export file for the context in which you plan to use it. There are three export context types to choose from:

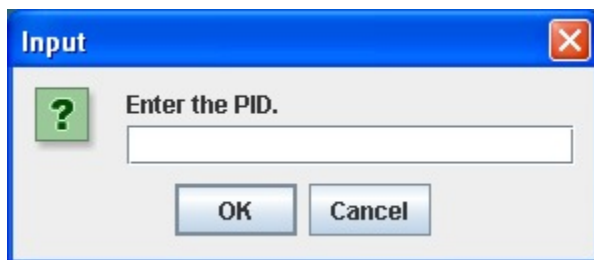
- **Migrate:** an export file will be create that is appropriate for migrating into another Fedora repository. This means that any references to the repository's base URL (with its host and port) will be virtualized, so that if the export object is subsequently ingested into another Fedora repository, the object will inherit the base URL of the new repository. The result is that repository-referential assertions in the exported digital object will naturally adapt to the new repository environment.
- **Public Access:** an export file will be created that is appropriate for facilitating public access and re-use of the digital object. All datastream locations will be resolvable URLs, most notably, the Managed Content datastreams will contain public callback URLs to the repository from which the object was exported. The assumption is that the repository will still exist, and that the original object will continue to be served up by the repository (so the callback URLs continue to work).
- **Archive:** an export file will be created with all managed content contained within it. This means that all Managed Content datastreams will have the datastream content Base64-encoded within the digital object export file. For Referenced and Redirected datastreams, their by-reference URLs will be in the export file (meaning that the repository will not pull the external content into the export file, but keep it by-reference). For datastreams that were of type "Inlined XML" the XML will remain in-lined as it normally is the the digital object XML file.



Export Multiple Objects


When the *Export Multiple Objects* option is selected, the user is prompted to select a directory to which the export files will be written. Clicking *Open* causes the user to be prompted for the export serialization format, as described above. When the format has been selected and *OK* is clicked, **all** objects are exported into the selected directory in the selected format type. The files are named based on their pid values.

Purging Objects

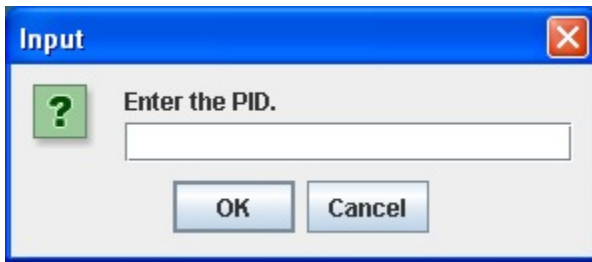


Purging an object completely and permanently removes it from the repository. Upon selecting the Purge option, the user will be prompted to enter an object PID and a reason for the object's removal.

Warning

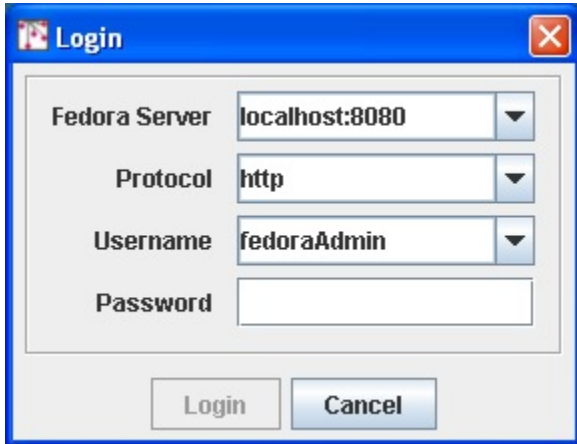
 There is no "Are you sure" dialog with this option.

Viewing Object XML



This menu option allows a user to view the xml, but not edit. XML may be cut and pasted into another application using standard keyboard commands of the host operating system, (e.g. CTL-C, CTL-V in Windows) if desired.

Changing Repository

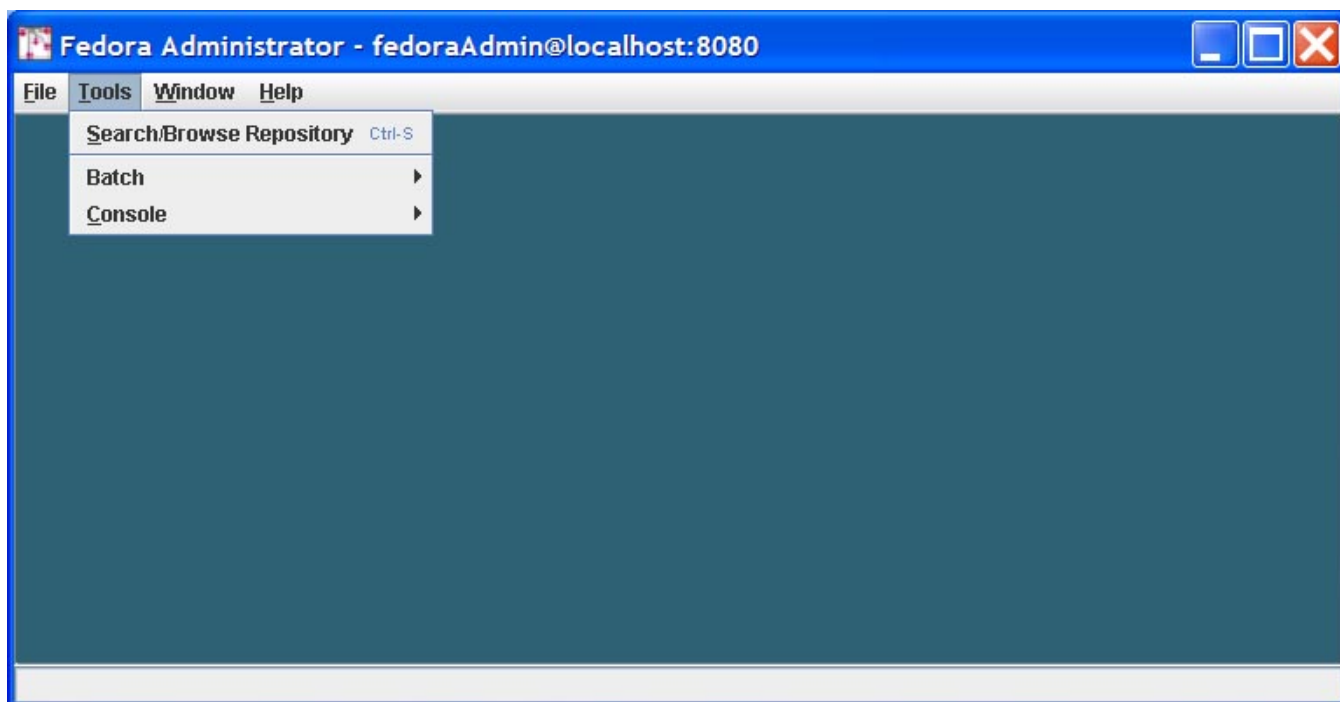


The Change Repository option allows a repository administrator to login to a different Fedora repository. When selected, this menu option causes the Login dialog to be displayed. The repository administrator may then select a different Fedora server with which to connect, entering the appropriate login name and password.

Exiting the Fedora Administrator

The Exit menu option closes all connections with the Fedora server instance and logs the user out of the repository.

Tools Menu

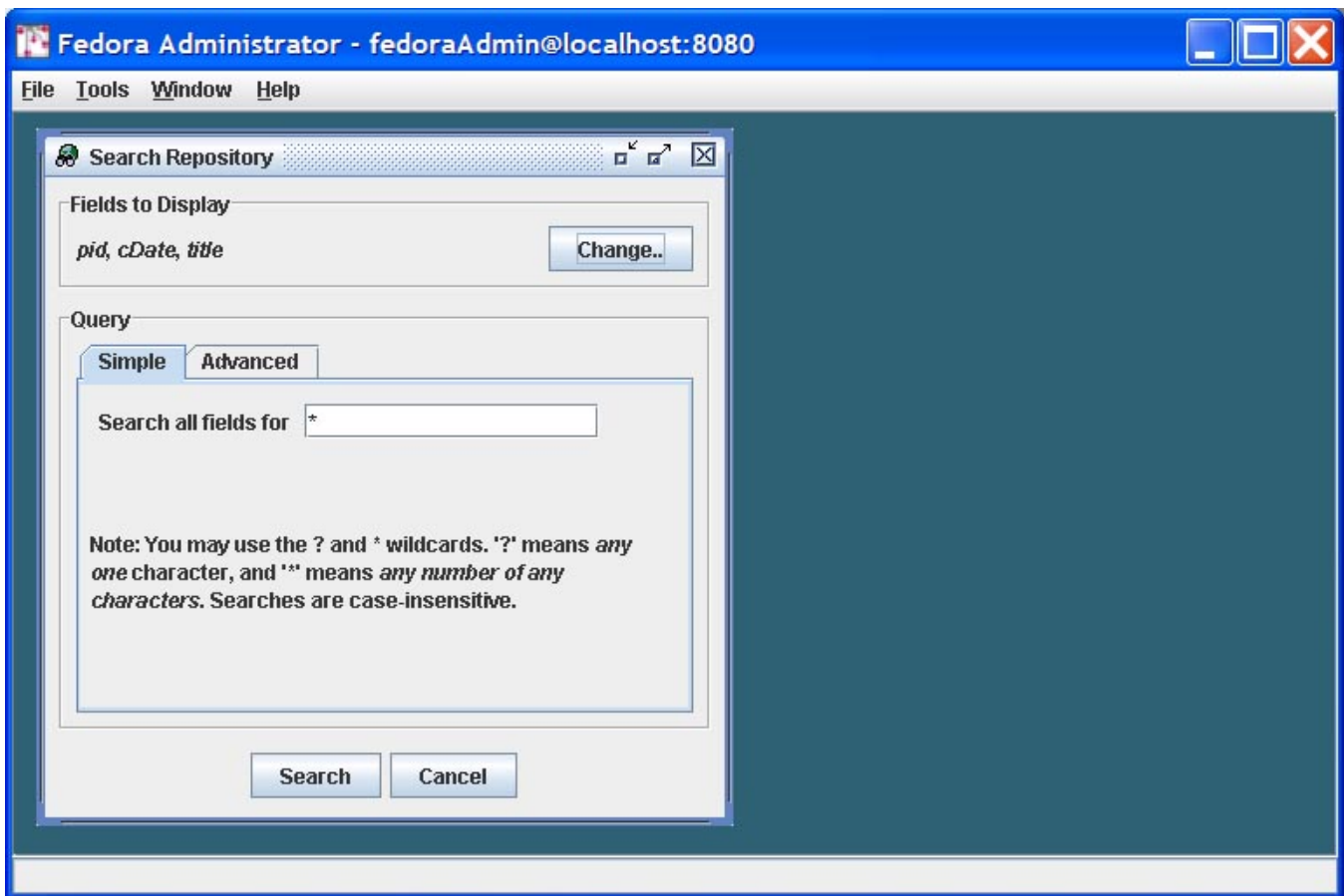


Commands on the Tools menu provide the user with the ability to search and retrieve objects from the repository, build and ingest batches of digital objects, and under the console submenu, gain access directly to API-M and API-A methods for testing purposes.

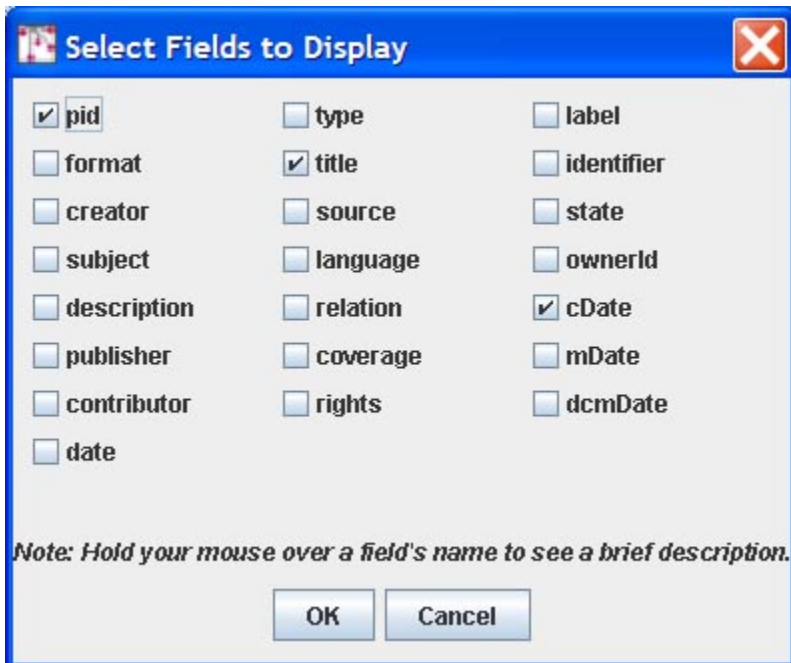
Searching and Browsing the Repository

The Search/Browse Repository menu option provides a mechanism for searching and retrieving objects from the Fedora repository. Upon ingestion, metadata from the Fedora System Metadata section and the Dublin Core (DC) Metadata section of the object are indexed in a relational database, and may be searched using this menu option. The DC Metadata section is an optional Implementer-Defined XML Metadata datastream in the object, where the *Datastream ID* is DC, and the XML conforms to the schema at http://www.openarchives.org/OAI/2.0/oai_dc.xsd. If a Dublin Core metadata datastream is not provided, Fedora will construct a minimal DC datastream consisting of the elements dc:title and dc:identifier. The value for dc:title will be obtained from the object's label (if present in the object) and the value for dc:identifier will be assigned to the object's persistent identifier or PID. The search interface provides both simple and advanced searching. All queries are case insensitive. Simple Search enables queries of words and phrases occurring anywhere in an object's indexed metadata fields. Advanced Search enables fielded searching across any combination of metadata elements using string comparison operators (= and ~) for string fields, and value comparison operators (=, >, >=, <, <=) for date fields (dc:date fields may be treated as both). The wildcards, * and ? may be used in any string-based query.

Simple Search Tab

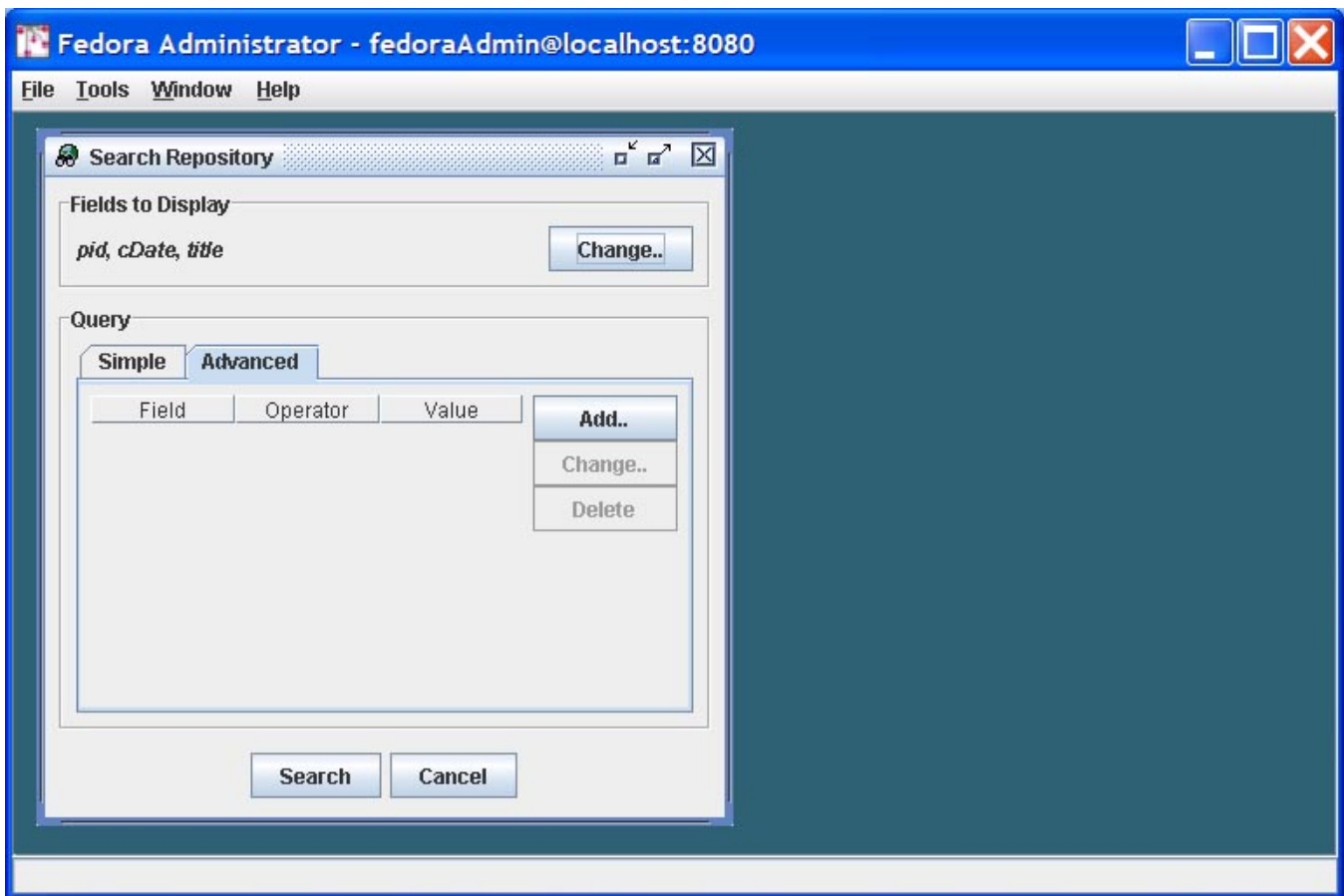


The Simple Search tab is the default selection in the Search Repository window. The Simple Search query searches both the Dublin Core metadata and the Fedora System Metadata fields. At the top of the Search window, the user may select fields to be displayed by clicking the *Change* button and selecting /deselecting field names from the dialog.

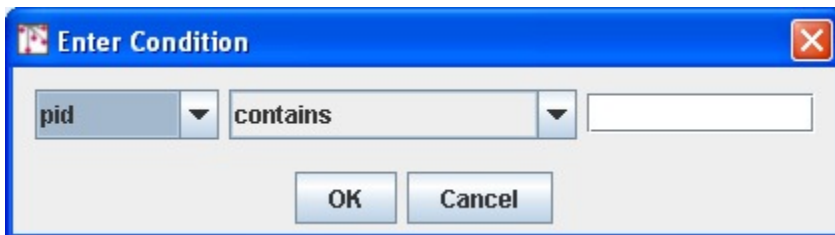


The Simple Search searches all indexed metadata fields for the text entered into the text box. All searches are case insensitive. The wildcard character '*'; can be substituted for any string of characters. The wildcard character '?'; can be substituted for any single character. Clicking *Search* will retrieve a list of objects where the entered text string appears in an indexed metadata field.

Advanced Search Tab



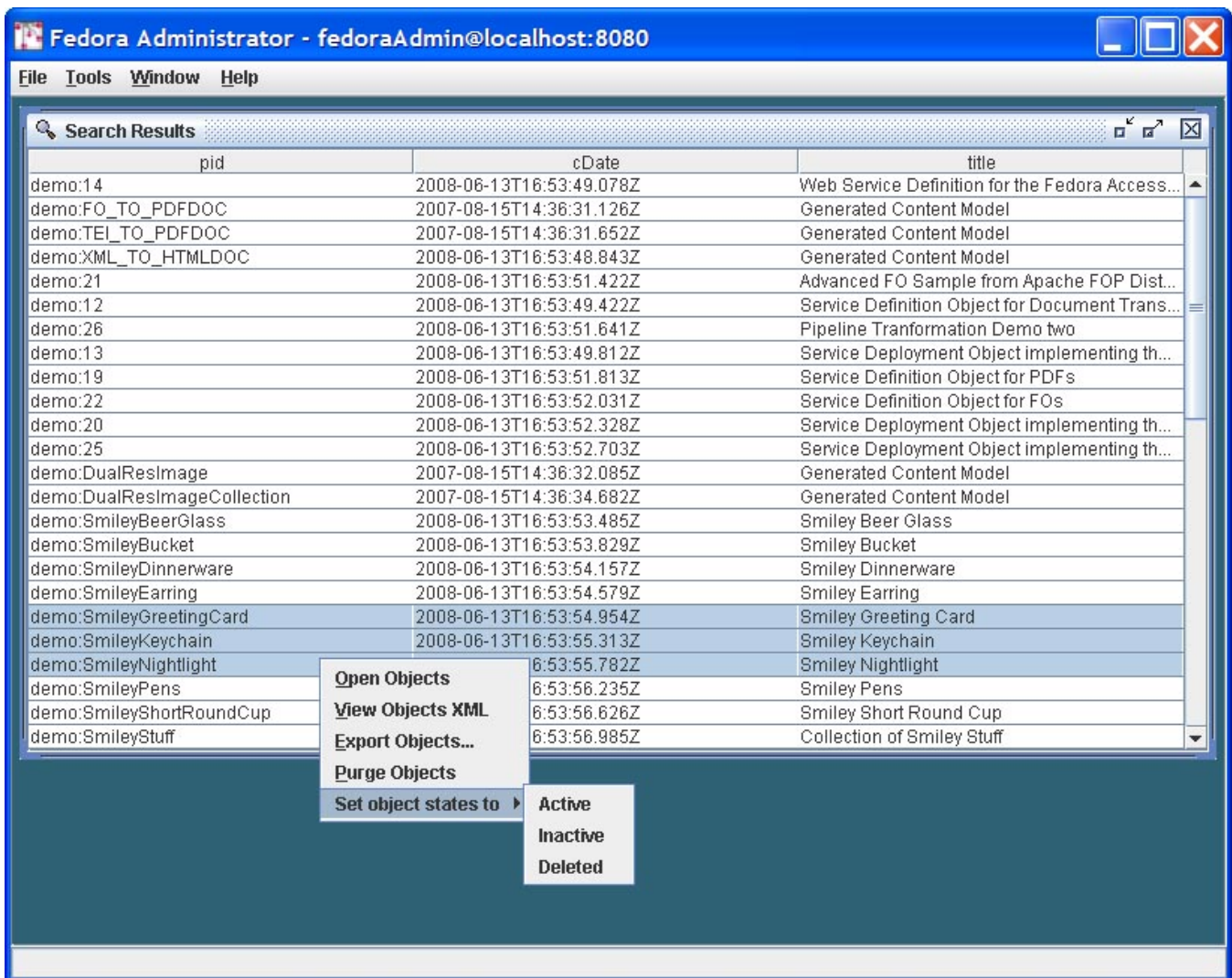
The Advanced Search query enables users to refine their repository search by searching specific fields for specific values provided in the query.



The search conditions can be modified by clicking the *Add* button, which opens the Enter Condition dialog. The user selects the field to be defined from the drop down menu, selects the condition to be matched, and enters the text to be matched, if appropriate. Clicking *OK* saves the condition. Once all conditions are entered, clicking *Search* will retrieve a list of objects in which all conditions are met.

Search Results Window

The Search Results Window displays the results of a successful search in a table format. Across the top of the table are a row of labels of the fields that have been returned from the objects meeting the search criteria. Double clicking anywhere on a row opens that object. Right clicking anywhere on a row opens a pop-up menu that contains object level tasks from which the user may select. These tasks include *Open Object*, *View Object XML*, *Export ...*, *Purge*, and *Set Object State To*. If *Set Object State To* is selected, a submenu will provide the user with valid states from which to select.

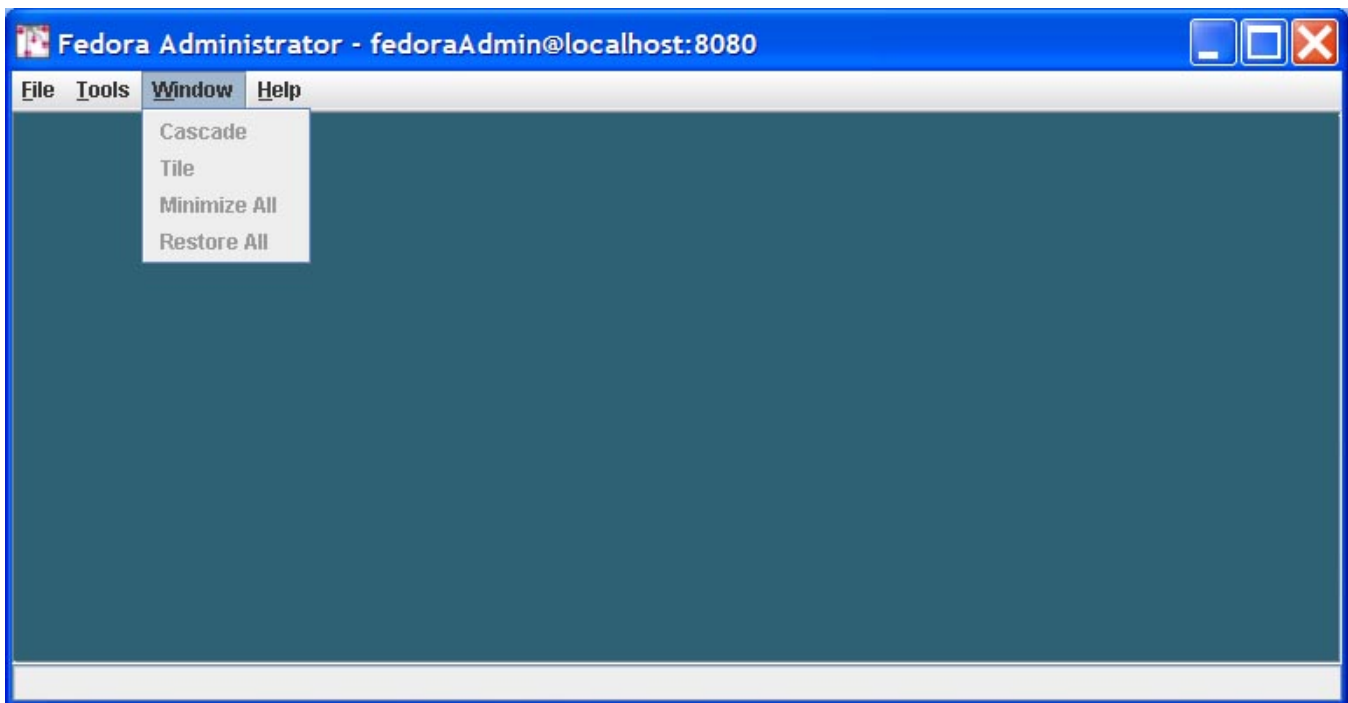


The *Purge* and *Set Object State To* submenu options can be used on multiple objects by using mouse clicks or the equivalent keyboard commands to multiselect rows in the Search Results Window. In this way, groups of objects can be purged from the system, or have their states changed by means of one search and retrieval operation.

Batch Processing

The Batch menu item includes tools to create and update multiple Fedora objects. For more information on batch processing, see [Batch Processing](#).

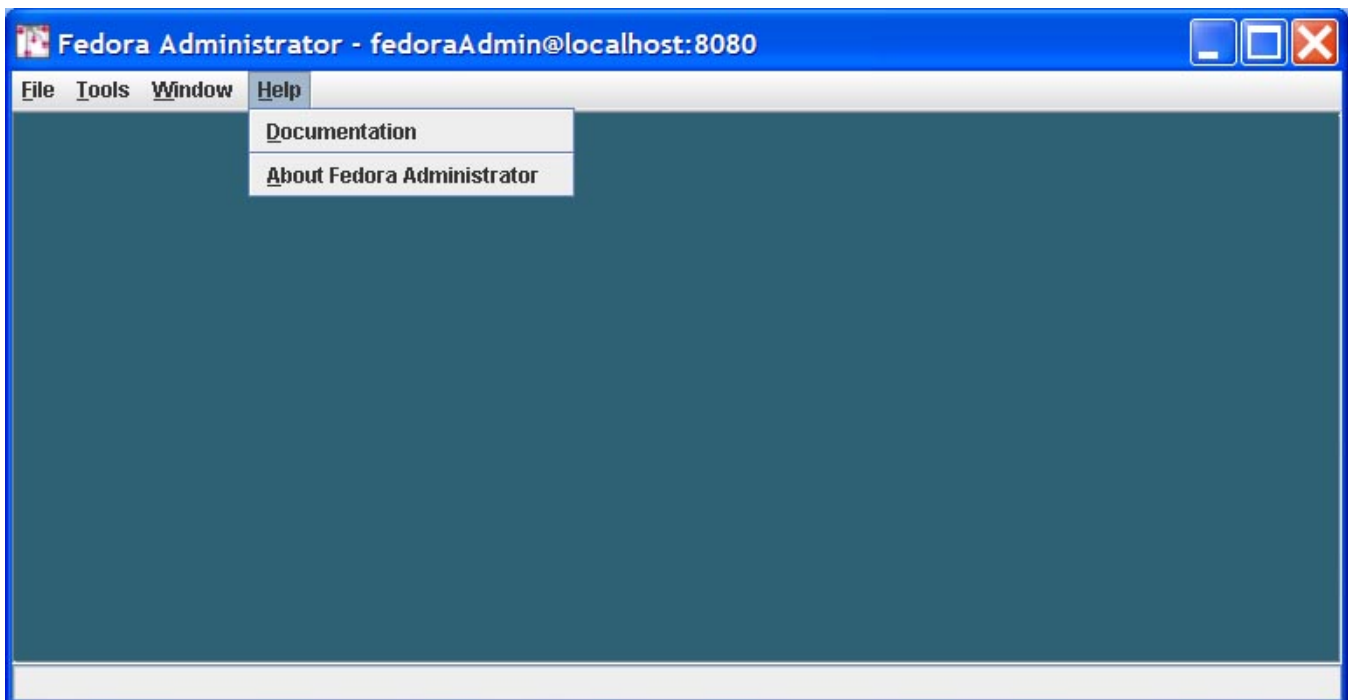
Window Menu



The Window menu contains standard commands for managing multiple panes open in the Fedora Administrator window. These include:

- Cascade
- Tile
- Minimize All
- Restore All

Help Menu



The help menu has two options:

- **Documentation** - which gives users the URL to the online documentation for Fedora.
- **About Fedora Administrator** - which provides version information, and copyright and licensing notices.

Appendix A: Digital Object Construction

This manual's focus is on practical use of the Fedora Administrator Tool. For further discussion on the theory behind digital object construction, please see the [Fedora Digital Objects](#).