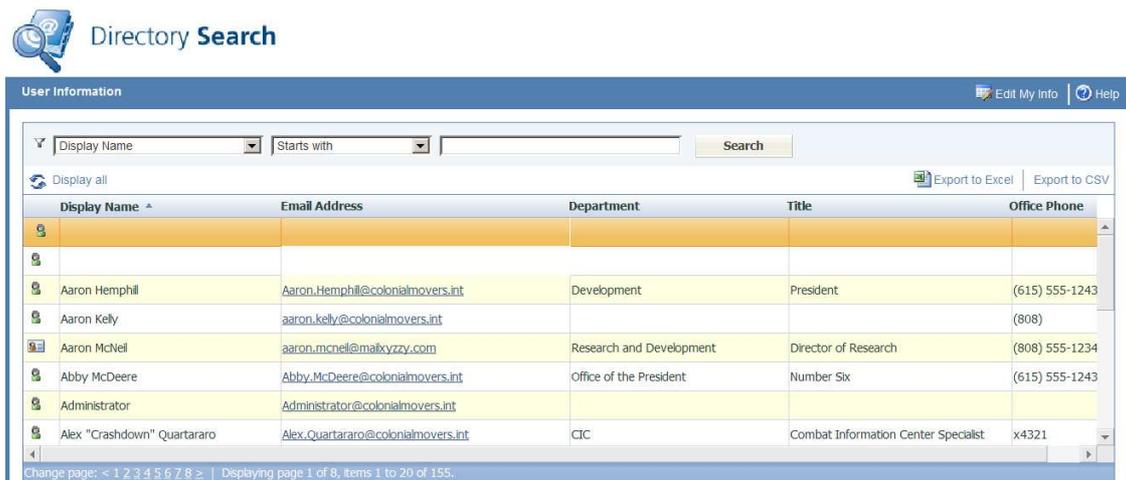# Applying Filters

By default, Directory Search will display all users and contacts in the Active Directory domain in which you have configured a domain controller for Directory Search to use. The search results will include all users accounts and contacts for that particular domain. Figure 1 shows the default Directory Search listing; notice that there are some blank accounts and the Administrator account that are included in the default listing.

Figure 1: Default search results

This may not produce the results you want for the users of Directory Search. We provide you ways to either increase or decrease the scope of users and contacts that are returned. Options include:

- List only user accounts (hiding contacts)
- Show only user accounts that have an Exchange mailbox
- List all users in the entire forest
- Increase the number of users/contacts listed per page and the maximum number of users
- Show or hide disabled user accounts
- Exclude certain users (such as service accounts and administrators) from the user listing
- Display all users in a specific OU or under a specific parent OU

Directory Search can be customized with all of the above features with different options in the AppSettings.XML file. We recommend that you make a backup copy of the AppSettings.XML file prior to editing it.

List Only User Accounts

By default, Directory Search lists all Contacts and all User accounts, but you can change that behavior. Look through the AppSettings.XML file for the *<userList…>* tag (shown in Figure 2.) This tag has a *showContacts="yes"* tag; set this tag to *showContacts="No"* so that you do not show Active Directory contact objects.

```
50    <!-- Declare maximum search results - Larger results will slow performance - Page size is the items per page on search listing -->
51    <!-- Microsoft Active Directory limits the maximum LDAP seach results returned to 1000. See Microsoft KB article 315071 for more information. -->
52    <!-- If you increase the search results returned to greater than 1000, consider the potentially negative peformance impact on the domain controller -->
53    <!-- showOnlyExchangeEnabledUsers will only display users with an Exchange mailbox. Set to "No" if your organization does not use Exchange Server -->
54    <!-- Single domain installations do not require the use of a global catalog. -->
55    <userList useGlobalCatalog="no" maxResults="200" pageSize="20" sortBy="displayName"
      showOnlyExchangeEnabledUsers="no" showContacts="yes" showDisabledUsers="yes">
56        <columns>
```

Figure 2: The userList tag from the AppSettings.XML file

Show Only Users with an Exchange Mailbox

In an environment with Exchange Server 2000/2003/2007, you may want to restrict the search listing so that you only see Exchange mail-enabled objects. Look for the *<userList…>* tag in the AppSettings.XML file; this is shown in Figure 2. Within this tag is the *showOnlyExchangeEnabledUsers="no"* option; set this tag to "yes" so that the default filter will only show user accounts that are mailbox or mail-enabled.

Show All Users and Contacts in the Forest

By default, Directory Search will show all users and contacts in the domain to which Directory Search was configured when it was initially installed. The domain name that is used is determined by the domain name and the domain controller name; when Directory Search does a query it uses a standard LDAP query and will return ONLY the users and contacts from that domain.

If you have a multi-domain forest, this query will only show you the users in that one specific domain. However, you can widen the scope to show all users by telling Directory Search to use a Global Catalog server search instead; this will show all users and contacts in the entire forest not just a specific domain.

To enable queries to all domains in a forest, locate the *<userList…>* tag in the AppSettings.XML file (shown in Figure 2). In this tag there is a *useGlobalCatalog="no"* option, set the option to "yes". This will widen the scope of your searches.

Please note that for organizations that do not use Exchange server, your Active Directory forest's "schema" was not prepped to replicate many of the common attributes from each domain to the Global Catalog. So, for example, if your forest was not prepped for Exchange and you are using a Global Catalog, you may see users but not see information in their state or postal code attributes.

You have a few options; one of these is to use the Active Directory Schema Management tool and configure each attribute you want to see to replicate to the Global Catalog. However, the simplest option is probably to get a copy of Exchange Server 2003 or Exchange Server 2007 and run the *setup /forestprep* (Exchange 2003) or *setup /schemaonly* (Exchange 2007). This will prep your forest to support Exchange Server even if you do not ever plan on implementing it.

Increase the Maximum Search Results and Search Results per Page

If you have more than 200 users in your Active Directory, you will notice that Directory Search only queries a maximum of 200 users and displays those in scroll pages of 20 users per page. An example of the number of items queried and the page size is shown in Figure 3.

Figure 3: Viewing the maximum entries per page and maximum search results

The maximum results per page and the maximum number of search results returned are both configurable. Locate the *<userList…>* tag in the AppSettings.XML file (shown in Figure 2.) The *maxResults="200"* sets the maximum number of results an LDAP query will return from Active Directory while the *pageSize="20"* shows the number of search results per scroll-page.

Directory Search was designed with the intent of actually searching for a small number of users rather than returning hundreds or thousands of search results. We recommend you keep the maximum number of search results at 200 or less in order to prevent your domain controllers from being overloaded. However, you can increase this value to 1,000 without any problems.

> Microsoft hard-codes in to Active Directory the maximum number of LDAP results that will be returned from a domain controller by default. Even if you set the *maxResults="5000"* option, Active Directory will still only return 1,000 search results. However, this can also be increased; see Microsoft Knowledge Base article 315071 for information on how to use the NTDSUTIL.EXE command to increase the maximum LDAP results returned by a domain controller. Exercise caution if your domain controllers are already overburdened or slow as this may put a large additional load on them if Directory Search is heavily used.

Hide Disabled User Accounts

By default, Directory Search will display all user accounts within the specific search criteria that you specify whether the account is enabled or disabled. In some organizations, this is a requirement because resource accounts (conference rooms, equipment resources, etc…) may need to be displayed in the address book, but their account is disabled.

You can change this behavior by locating the <userList…> tag in the AppSettings.XML file (shown previously in Figure 2) and changing the *showDisabledUsers="yes"* option to "no".

Exclude Some Users from the Search Results

Directory Search will enumerate and display ALL user accounts in your Active Directory by default. This includes service or system accounts, resource accounts, and even trust accounts.

Directory Search allows you to exclude specific accounts from the search listing. To configure this, you must first enable the account filer option and specify which attribute you will use and what value on which you will exclude. This is done using the *<accountFilter…>* tag (shown in Figure 4) in the AppSettings.XML file. The *enabled="no"* option should be changed to "yes".

```
46    <!-- If you have users (such as administrators or service accounts) that you do not want to show up in the lookup fields (Manager, Secretary, Assistant) -->
47    <!-- or in the search listings you can specify an attribute and a value for that attribute. The default is extensionAttribute12 (Exchange installations only); -->
48    <!-- if you put the text excluded in this attribute, it will be exluded from the listings. -->
49    <accountFilter enabled="no" attribute="extensionAttribute12" value="excluded" />
```
Figure 4: Specifying how to exclude certain user accounts

By default, we use the *extensionAttribute12* attribute (also known as Custom Attribute 11) in Active Directory; however this attribute will ONLY exist if you have prepped your forest to support Exchange 2000/2003/2007. You can change the attribute to any valid Active Directory attribute such as *description, st, givenname, sn*, or *l*. The final option in the *<accountFilter..>* tag is *value="excluded"*. This specifies the text that you will put in to the specific attribute (*extensionAttribute12* by default.) We use the text **"excluded"** by default, but you can change this to anything that you want to use.

> The exclusion text, **"excluded"** by default, is not case sensitive.

Once you have configured Directory Search to use the account filter, you can then populate this information in Active Directory. For Exchange "mail-enabled" accounts, you can simply use Active Directory Users and Computers, locate the user account, and edit extensionAttribute12 (found on the Exchange Advanced property page.) This property page is shown in Figure 5.
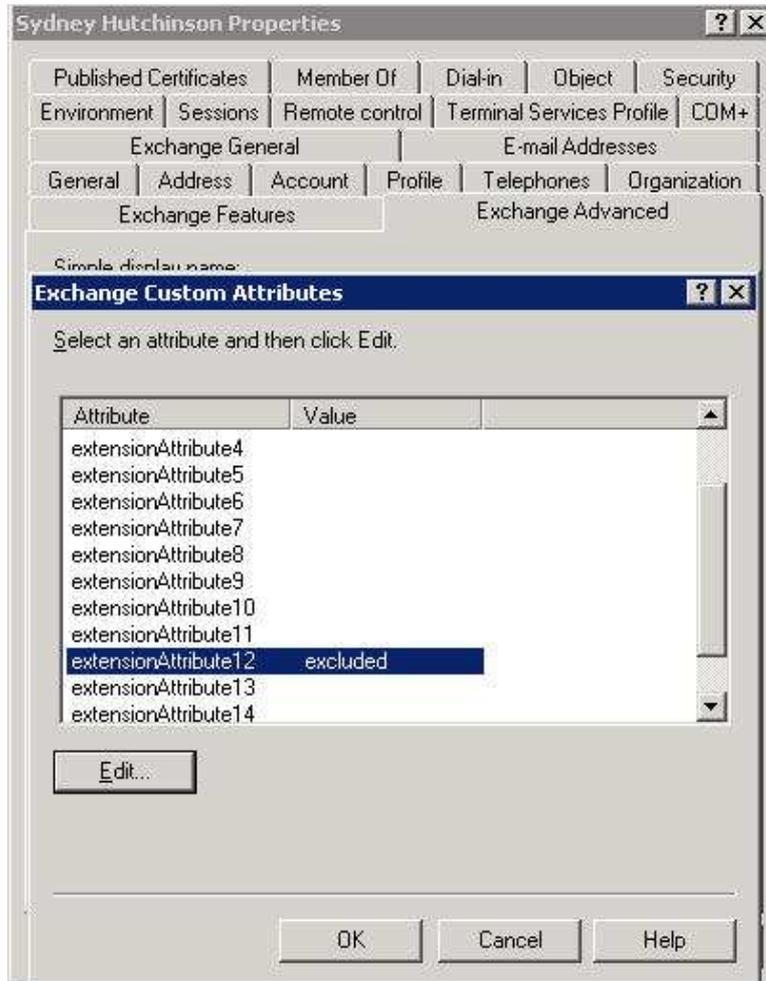
Figure 5: Editing a custom attribute using Active Directory Users and Computers

If the user account that you need to exclude is not mail-enabled, you can either change the attribute to some other attribute in Active Directory, or you can use the ADSIEDIT.MSC console (included with the Windows 2003 Support Tools). ADSIEDIT is a bit more difficult to use, but it allows you to edit the exact same information (and more), but just in a more "raw" format.

If you look in Figure 6, you can see the editing interface for ADSIEDIT. This is one other option for editing user account information that you cannot edit in Active Directory Users and Computers.
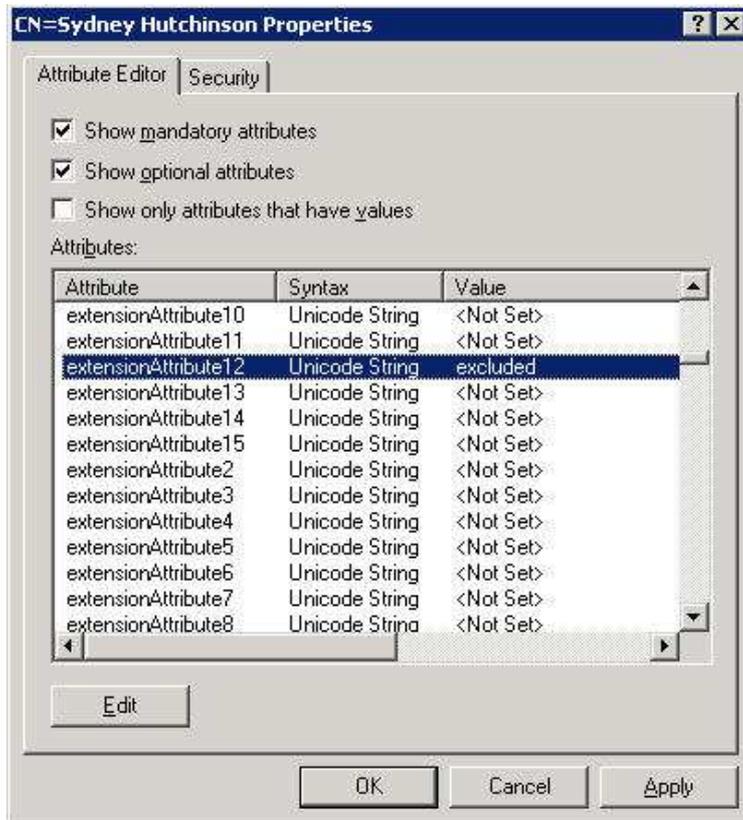
Figure 6: Editing a user using ADSIEDIT

OU Filtering

Directory Search has the ability to set a base organizational unit (*searchBaseOU* or just *baseOU*) from which to start searching. However, you can only set ONE baseOU; you cannot combine different search baseOUs together. This works best if you have all of your users and contacts under a single OU in Active Directory. The restriction on setting a single baseOU is a restriction placed on us by LDAP.

You can also set the search filter so that you can list all accounts or contacts in a specific OU.

> By default, all user accounts in Active Directory are created in the Users container; however this is not a true OU and we cannot filter on the Users container.

The best way to illustrate this feature is to use an example. Look at the OU structure seen in Figure 7. The DNS Active Directory name is *colonialmovers.int*. All users and contacts are found under the CorporateUsers root OU.

Figure 7: OU structure for an Active Directory

Display Only a Specific Parent OU

The first feature is how to tell Directory Search to display only the user accounts and contacts found under a specific root-level OU. This is the *searchBaseOU* feature. To enable this, locate the *<ouFilter..>* tag in the AppSettings.XML file; this tag actually has an open tag *(<ouFilter..>)* and a close tag *(</ouFilter...>)* and other tags and options within those tags.

```
<!-- OUs allows user to search an entire OU. OU name should be listed like this: name="All Users" or name="All Users/
<!-- The domain name must be the DNS domain name of the Active Directory, such as domain="acme.local" -->
<!-- If baseSearchOU is used, all searches will start from that base OU. If you want specify OU filters do not include
<!-- searchBaseOU specifies the starting point for the search listing.  Leave blank for no search base, all OUs queried.
<!-- Specify OU name without OU=, i.e. searchBaseOU="All Users" - You cannot specify Users container. -->
<ouFilter text="Organizational Unit" enabled="yes">
  <OUs domain="colonialmovers.int" searchBaseOU="CorporateUsers">
    <OU name="" displayText="" />
    <OU name="" displayText="" />
  </OUs>
</ouFilter>
```

Figure 8: Setting a searchBaseOU

The example in Figure 8 enables the *ouFilter* feature, sets the DNS domain name of the Active Directory to *colonialmovers.int* and sets the *searchBaseOU* to "CorporateUsers". This means that all users and accounts below the CorporateUsers OU will be displayed. To merely set the searchBaseOU, you do NOT need to enable any of the <OU…> options found below that tag.

Searching for All Users in a Specific OU

Directory Search allows you to search and display all users in a specific OU. However, we do NOT read the OU structure from Active Directory; you must specify the OU names AND a friendly name for each OU. These will then appear in the search filter drop-down list. Let's look at another example. In this example, we want the user to see ALL users and contacts by default, but be able to list just the users in a specific OU using the search filter option on the main Directory Search page.

Notice in this example, we took the OU structure that is seen in Figure 7 and we are providing an OU name (notice that they are in the format of CorporateUsers/Battlestar and that the OU= and DC= options are not necessary.) Further, notice that we provided a "friendly name" for each OU name.

```
<ouFilter text="Organizational Unit" enabled="yes">
  <OUs domain="colonialmovers.int" searchBaseOU="">
    <OU name="CorporateUsers/Battlestar" displayText="BSG-76" />
    <OU name="CorporateUsers/Firefly" displayText="Browncoats" />
    <OU name="CorporateUsers/LAPD" displayText="The Closer" />
    <OU name="CorporateUsers/Red October" displayText="Red October" />
  </OUs>
```

Figure 9: Creating filters by OU name

The resulting search function in Directory Search will allow you to select Organizational Unit as the search criteria and then search for users and contacts in one of your specified OUs. However, if the OU search is not specified, all users and contacts in the entire directory; this is because we did not specify a *searchBaseOU*.
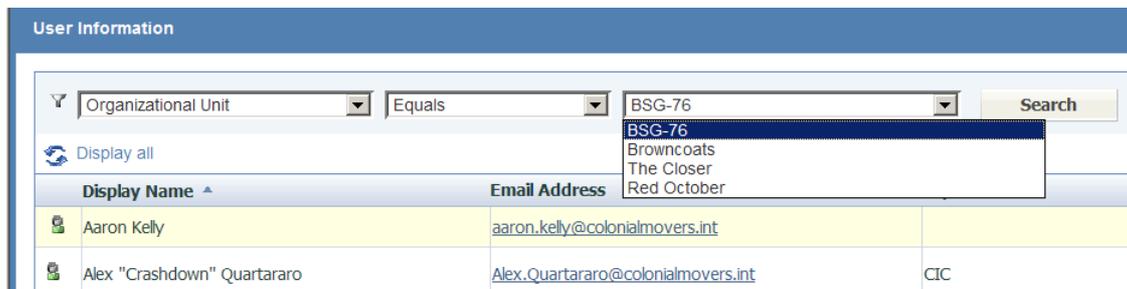

Figure 10: Searching for all users in a specific OU

If you want a *searchBaseOU* AND the ability to then further search by a specific OU, then the format of the AppSettings.XML file is a bit different. The *searchBaseOU* option sets the starting point for the search, so all OUs must be under the *searchBaseOU* starting point.

The AppSettings.XML example shown in Figure 11 is configured so that the search base is the root-level OU CorporateUsers AND the user can also enumerate all of the users in one of the sub-OUs, Battlestar, Firefly, LAPD, or Red October.

```xml
<ouFilter text="Organizational Unit" enabled="yes">
  <OUs domain="colonialmovers.int" searchBaseOU="CorporateUsers">
    <OU name="Battlestar" displayText="BSG-76" />
    <OU name="Firefly" displayText="Browncoats" />
    <OU name="LAPD" displayText="The Closer" />
    <OU name="Red October" displayText="Red October" />
  </OUs>
</ouFilter>
```

Figure 11: Setting a searchBaseOU and OU search filters