# LDAPQueryAnalyzer – Manual

[LDAPQueryAnalyzer – Manual 1](#_Toc433362234)

[LDAPQueryAnalyzer 4](#_Toc433362235)

[Fields 4](#_Toc433362236)

[Domains 4](#_Toc433362237)

[DCs 4](#_Toc433362238)

[NCs 4](#_Toc433362239)

[Base 5](#_Toc433362240)

[Filter 5](#_Toc433362241)

[History 5](#_Toc433362242)

[Attributes 5](#_Toc433362243)

[ASQ Attributes 5](#_Toc433362244)

[Results 6](#_Toc433362245)

[Controls 6](#_Toc433362246)

[Port 6](#_Toc433362247)

[LDAP 6](#_Toc433362248)

[GC 6](#_Toc433362249)

[Custom 6](#_Toc433362250)

[BaseType 6](#_Toc433362251)

[DN 6](#_Toc433362252)

[PhantomRoot 6](#_Toc433362253)

[rootDSE 6](#_Toc433362254)

[operational attributes 7](#_Toc433362255)

[Scope 8](#_Toc433362256)

[Base 8](#_Toc433362257)

[OneLevel 8](#_Toc433362258)

[SubTree 8](#_Toc433362259)

[Referrals 8](#_Toc433362260)

[None 8](#_Toc433362261)

[External 8](#_Toc433362262)

[Subordinate 8](#_Toc433362263)

[Query type 8](#_Toc433362264)

[Paged 8](#_Toc433362265)

[Non-Paged 8](#_Toc433362266)

[Attributes 8](#_Toc433362267)

[AttributeScoped 8](#_Toc433362268)

[Return 9](#_Toc433362269)

[Results 9](#_Toc433362270)

[Statistics 9](#_Toc433362271)

[Search 9](#_Toc433362272)

[Restore 9](#_Toc433362273)

[Menus 10](#_Toc433362274)

[File 10](#_Toc433362275)

[Close 10](#_Toc433362276)

[User 10](#_Toc433362277)

[Restart elevated 10](#_Toc433362278)

[Run elevated (2nd instance) 10](#_Toc433362279)

[Restart as 10](#_Toc433362280)

[Restart as elevated 10](#_Toc433362281)

[Run as (2nd instance) 10](#_Toc433362282)

[Run as elevated (2nd instance) 10](#_Toc433362283)

[Connection 10](#_Toc433362284)

[Refresh Forest 10](#_Toc433362285)

[Set Ports 10](#_Toc433362286)

[LDAP 11](#_Toc433362287)

[GC 11](#_Toc433362288)

[Set Timeout (s) 11](#_Toc433362289)

[Connect Forest 11](#_Toc433362290)

[Pass Credentials 11](#_Toc433362291)

[Query 11](#_Toc433362292)

[Browse (Base) 11](#_Toc433362293)

[Execute Query 11](#_Toc433362294)

[Provider 12](#_Toc433362295)

[LDAP 12](#_Toc433362296)

[GC 12](#_Toc433362297)

[Base type 12](#_Toc433362298)

[DN 12](#_Toc433362299)

[PhantomRoot 12](#_Toc433362300)

[rootDSE 12](#_Toc433362301)

[Query type 12](#_Toc433362302)

[Paged 12](#_Toc433362303)

[Non-Paged 12](#_Toc433362304)

[SearchScope 12](#_Toc433362305)

[Base 12](#_Toc433362306)

[OneLevel 12](#_Toc433362307)

[SubTree 12](#_Toc433362308)

[Return 13](#_Toc433362309)

[Results 13](#_Toc433362310)

[Statistics 13](#_Toc433362311)

[operational rootDSE attributes 13](#_Toc433362312)

[Autoswitch to PagedQuery 13](#_Toc433362313)

[Value Range Retreival 13](#_Toc433362314)

[Attribute Scoped Query 13](#_Toc433362315)

[Show deleted 14](#_Toc433362316)

[Show recycled 14](#_Toc433362317)

[Sort results ascending 14](#_Toc433362318)

[Sort results descending 14](#_Toc433362319)

[Filter 14](#_Toc433362320)

[Filter Wizard 14](#_Toc433362321)

[Clear Filter History 14](#_Toc433362322)

[Attributes 14](#_Toc433362323)

[Load Attribute List from Schema 14](#_Toc433362324)

[Hide Attribute List 14](#_Toc433362325)

[Decode 14](#_Toc433362326)

[GUIDs 14](#_Toc433362327)

[SIDs 14](#_Toc433362328)

[userParameters 14](#_Toc433362329)

[Resolve Sids 14](#_Toc433362330)

[SecurityDescriptors 14](#_Toc433362331)

[OctetStrings 14](#_Toc433362332)

[ReplicaLinks 14](#_Toc433362333)

[memberOf includes primaryGroupID 14](#_Toc433362334)

[To local time 15](#_Toc433362335)

[Results 15](#_Toc433362336)

[Copy selected 15](#_Toc433362337)

[Copy all 15](#_Toc433362338)

[Remembered results 15](#_Toc433362339)

[Show first 15](#_Toc433362340)

[Show previous 15](#_Toc433362341)

[Show next 15](#_Toc433362342)

[Show current 15](#_Toc433362343)

[Clear list 15](#_Toc433362344)

[Clear Output Clear current result pane. 15](#_Toc433362345)

[Change Font 15](#_Toc433362346)

[Tools 15](#_Toc433362347)

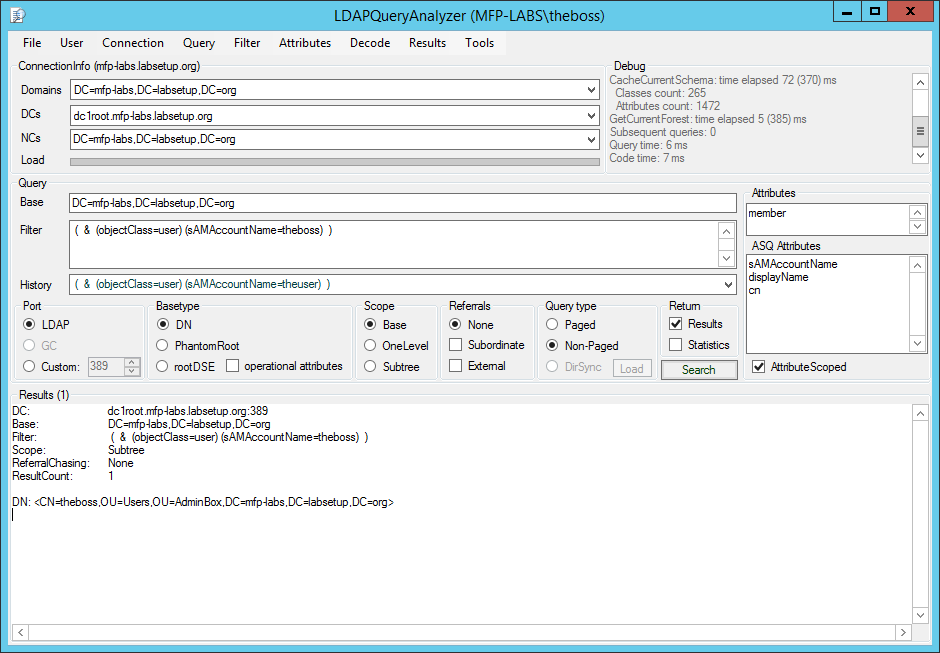
[DynamicTypeBuilder 15](#_Toc433362348)

[FilterWizard 16](#_Toc433362349)

[DynamicTypeBuilder 21](#_Toc433362350)

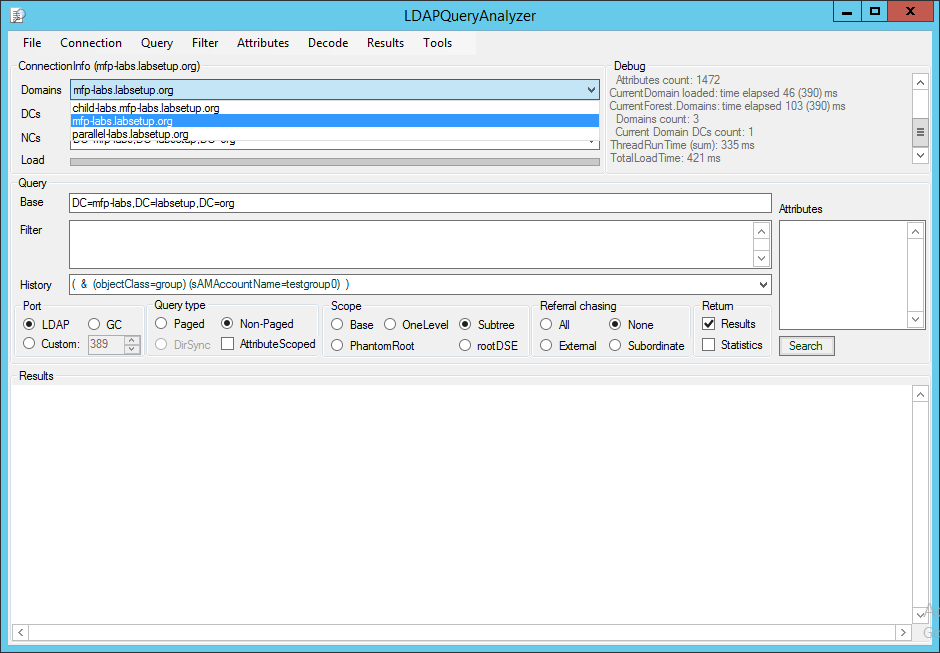
[Disclaimer 24](#_Toc433362351)

### LDAPQueryAnalyzer



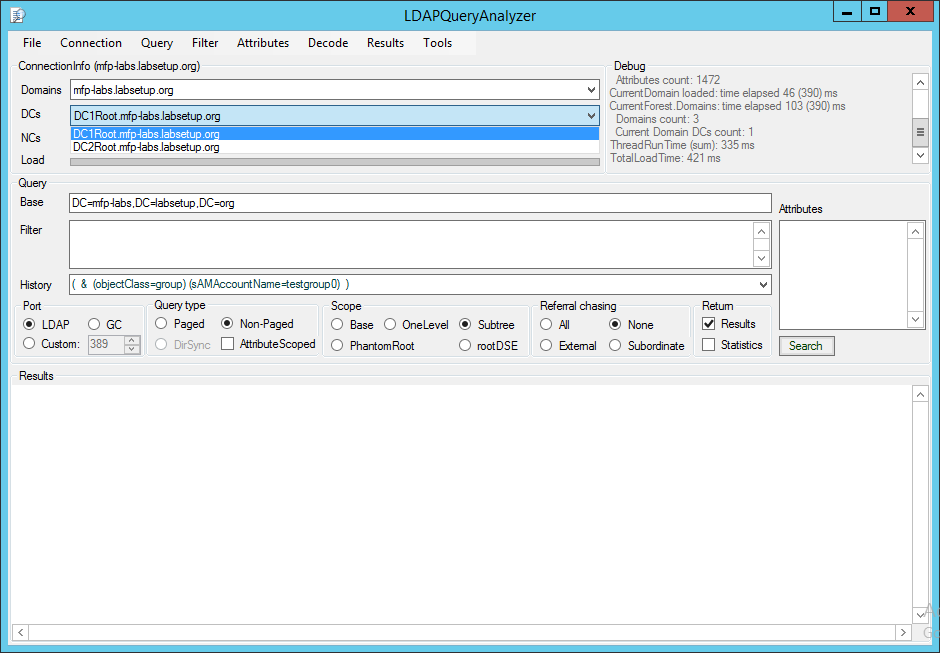
### Fields

Domains  
A list of all domain naming contexts (NC) in the current forest. The domain NC of the executing user is displayed after start.



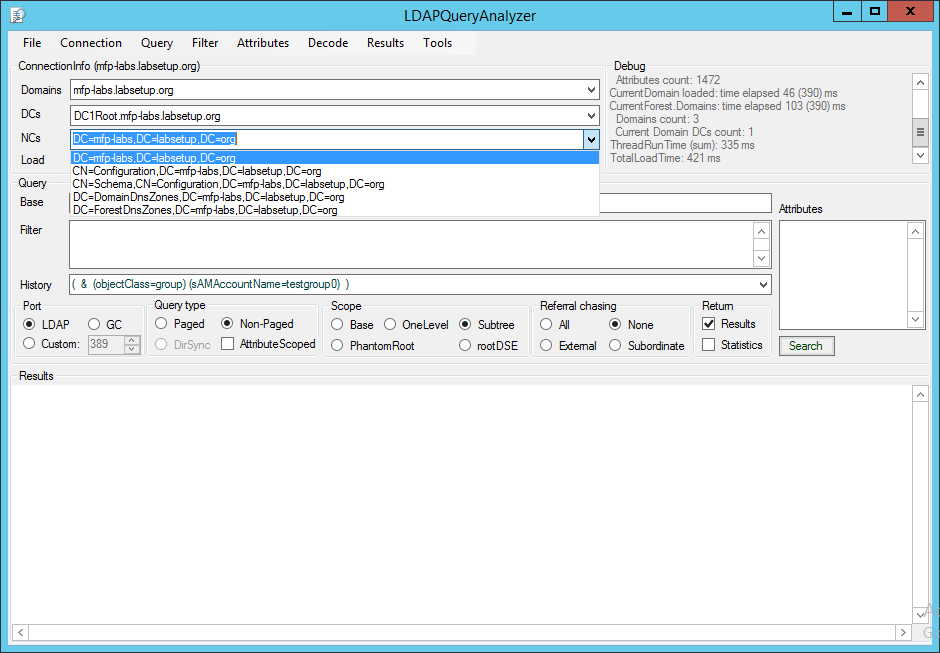
#### DCs

A list of all domain controllers (DC) in the currently displayed domain.

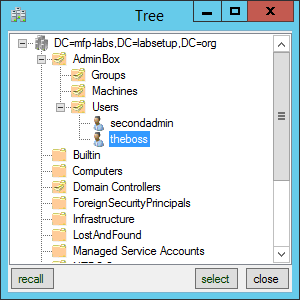


#### NCs

A list of all NCs held by the selected DC.

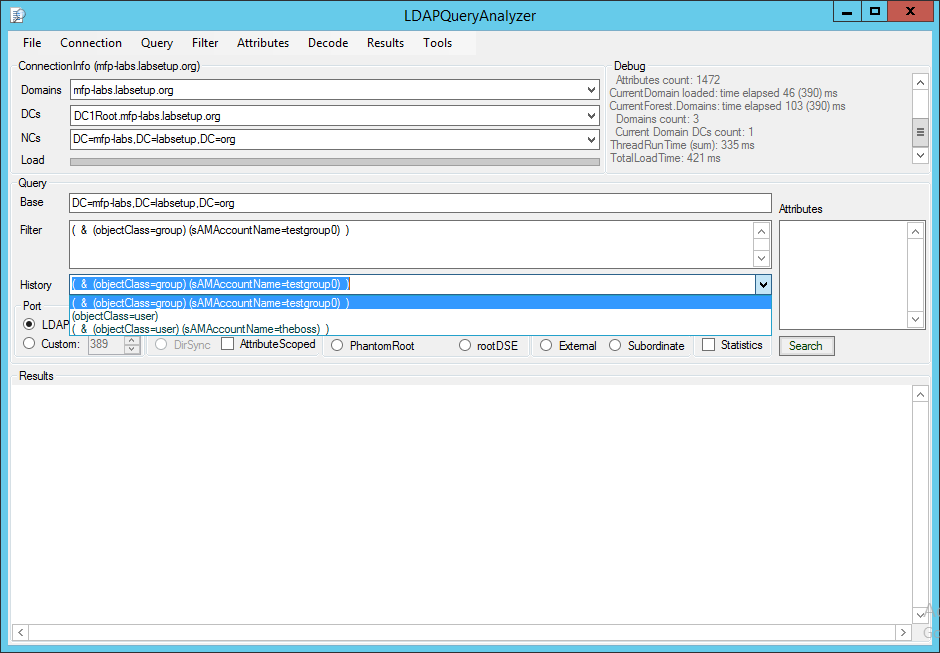


Base  
The base path to search in – selecting a NC in field NCs updates field Base.  
Additionally you may use the context menu <Browse> to load a browsing GUI to find object paths in your AD.

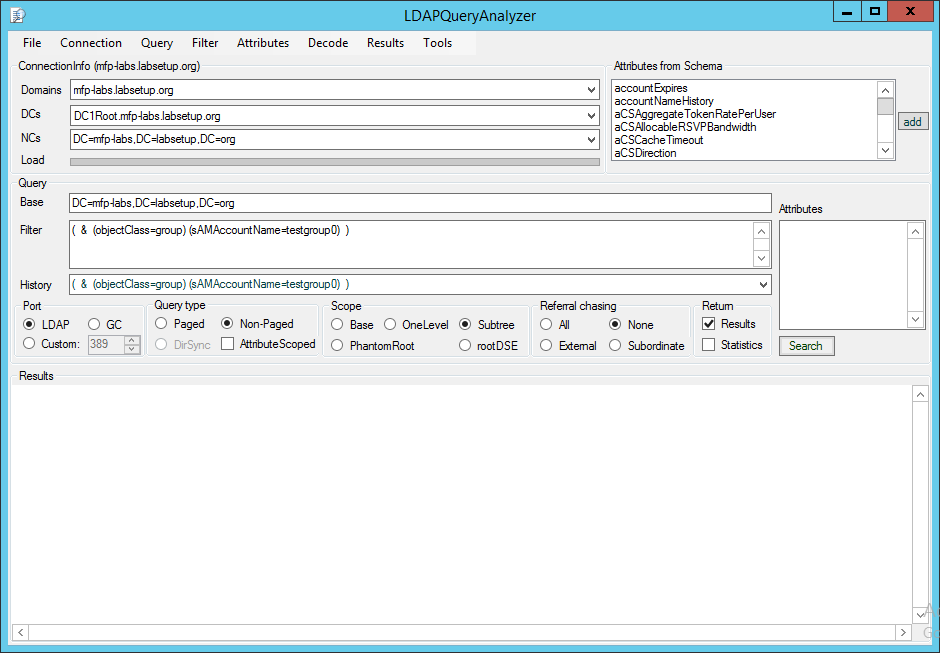


Filter  
Here you define the query filter in LDAP syntax. For the ease of use a FilterWizard is implemented -> context menu Filter Wizard. The FilterWizard will be described later.

History  
A list of already successfully executed queries. List can be cleared by selecting context menu <Clear History>. You may also delete single entries by selecting context menu <Delete Current>



Attributes  
The (list of) attributes you want to be returned by the query. You may use the context menu <Load Attributes List> to get a list of all available attributes from the schema (including custom attributes).

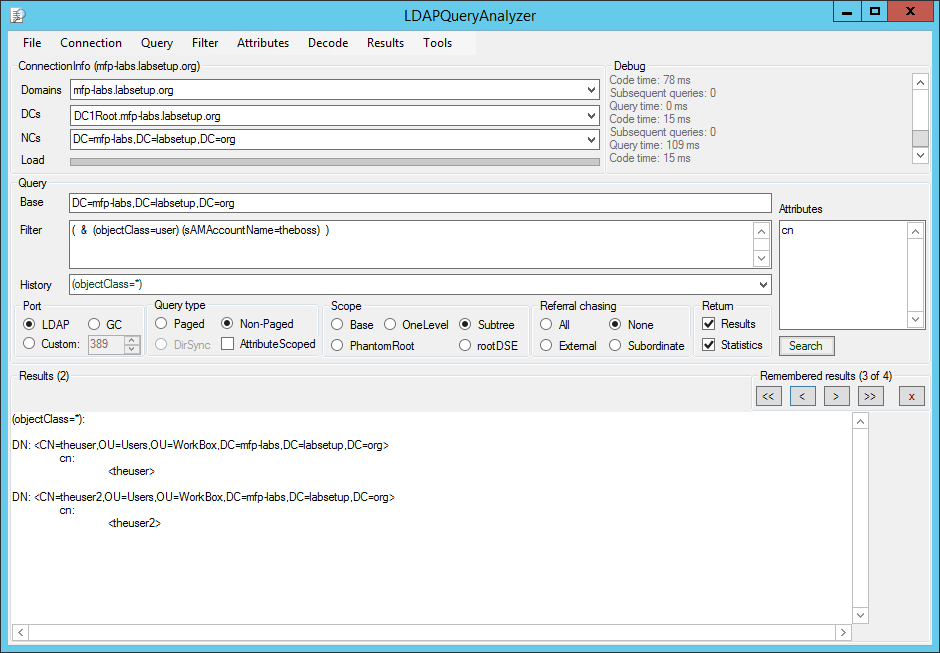


If the attribute list contains constructed attributes, LDAPQuerzAnalzyer switches to single object path base queries to retreive the constructed attributes from each object matching the given filter.

You may use 0 or null as attribute to get only the paths of the matching objects returned (no attributes will be requested from the DC).

ASQ Attributes  
The attributes you want to get returned from the objects who are actually stored via their distinguishedName (DN) in the queried attribute via an Attribute Scoped Query query.

Results  
The result output pane. Every query result during runtime is saved in a result cache. If the result cache count execeeds 20 the first result is deleted and the new result is added at the end of the cache.  
You may walk or clear the result cache via the <Remembered results> buttons.



Every query result will display query info.   
Example:

DC: dc1root.mfp-labs.labsetup.org:389  
Base: DC=mfp-labs,DC=labsetup,DC=org  
Filter: ( & (objectClass=user) (sAMAccountName=theboss) )  
Scope: Subtree  
ReferralChasing: None  
ResultCount: 1

You may search the query results by pressing Ctrl & F when the result output pane has the focus -> this will show the find dialogue:



Clicking the find button <?> will try to find the first match, pressing F3 will try to find the next match in the query results.

The <x> button will close the find dialogue.

### Controls

#### Port

LDAP  
The LDAP port (default 389) is used for DC connection. Default value can be defined in context menu Define LDAP Port:.

GC  
The Global Catalog (GC) port (default 3268) is used for DC connection. Default value can be defined in context menu Define GC Port. This option is only available when the selected DC is also a GC.

Custom  
You may define a custom port to connect to the DC – for example to connect to an AD snapshot instance mounted via dsamain.exe.

#### BaseType

DN  
Standard base type -> a distinguishedName (defined by Base field) is used as search base.

PhantomRoot  
Performing a subtree search against the phantom root of the forest (null-DN) with search scope SubTree. Thus you will be able to find any object in the whole forest – even if you have multiple parallel trees in your forest.

rootDSE  
Send an UDP ping (rootDSE call) against the selected DC with an empty search base and search scope Base.

operational attributes  
If selected you get back the operational attributes when sending an UDP ping (rootDSE call). You may customize the operational attributes to be returned by editing the the RootDSE.xml in the folder .\Cache\Settings. To add a new attribute to the operational attribuites list just insert a new line like <string>*newAttribute*</string> to the <OperationalAttributes> node.

#### Scope

Base  
Performing a base search against the object path defined in Base.

OneLevel  
Performing an one level search against the object path defined in Base.

SubTree  
Performing an subtree search against the object path defined in Base.

#### Referrals

None  
Never chase the referred-to server. Setting this option prevents the app from contacting other servers in a referral process.

External  
Chase external referrals.

Subordinate  
Chase only subordinate referrals which are a subordinate naming context in a directory tree.

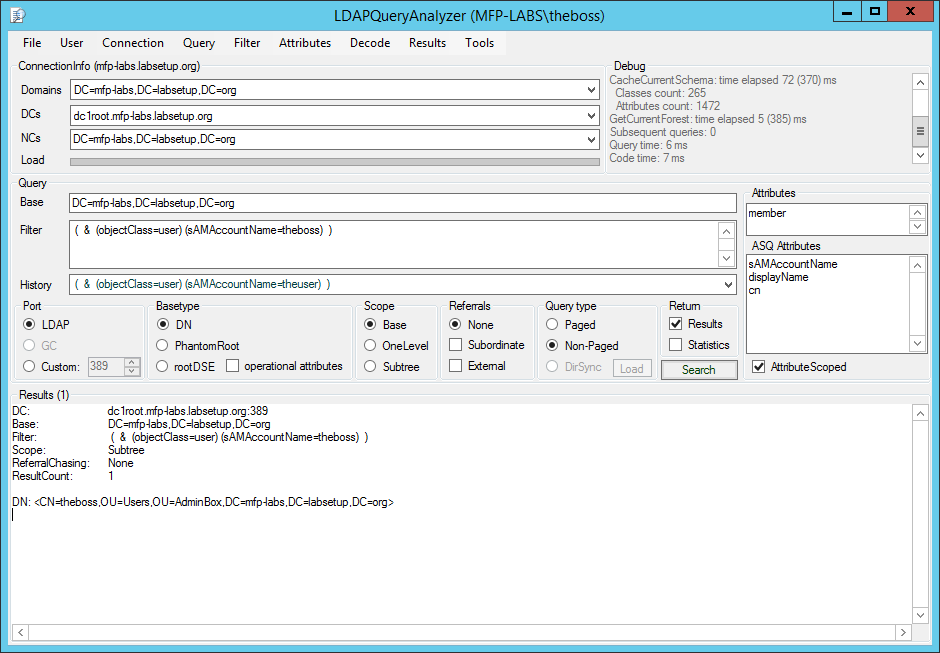
#### Query type

Paged  
The requested query will be send as paged query. Usefull if you expect more results to be returned as the MaxPageSize value defined in lDAPAdminLimits. Caveat – query index intersection in this case results in the usage of only one indexed attribute in the query even though there would be more indexed attributes used in the query - possible performance impact.   
Therefore the switch Autoswitch to PagedQuery was implemented in the Menu Query - see description in option Non-Paged below.

Non-Paged  
We first try to send a non paged query. If the result count exceeds the MaxPageSize value defined in lDAPAdminLimits we only return the results up to MaxPageSize.  
Per default the switch <Autoswitch to PagedQuery> from the Menu <Query> ist activated. This will resend the query as a paged query and we get all results returned.

#### Attributes

AttributeScoped  
Attribue scoped queries (ASQ) can be send against any linked attribute pair member, like for example:

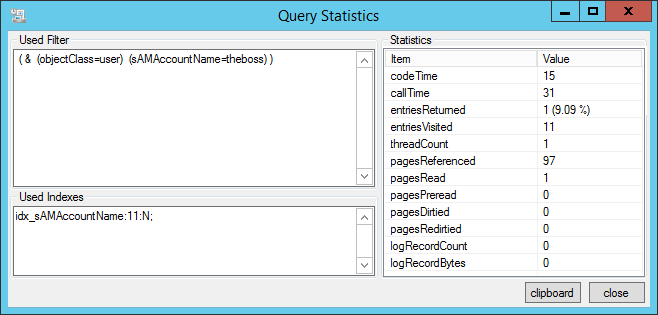
* member / memberOf
* manager / directReports

An ASQ must be send against the path of an object that holds this attribute as a base search. In the Attributes field we need the attribute we want to do the ASQ against as only entry in the list. The attributes in the field ASQ Attributes define the attributes you want to get returned from the objects who are actually stored via their distinguishedName (DN) in the queried attribute.

#### Return

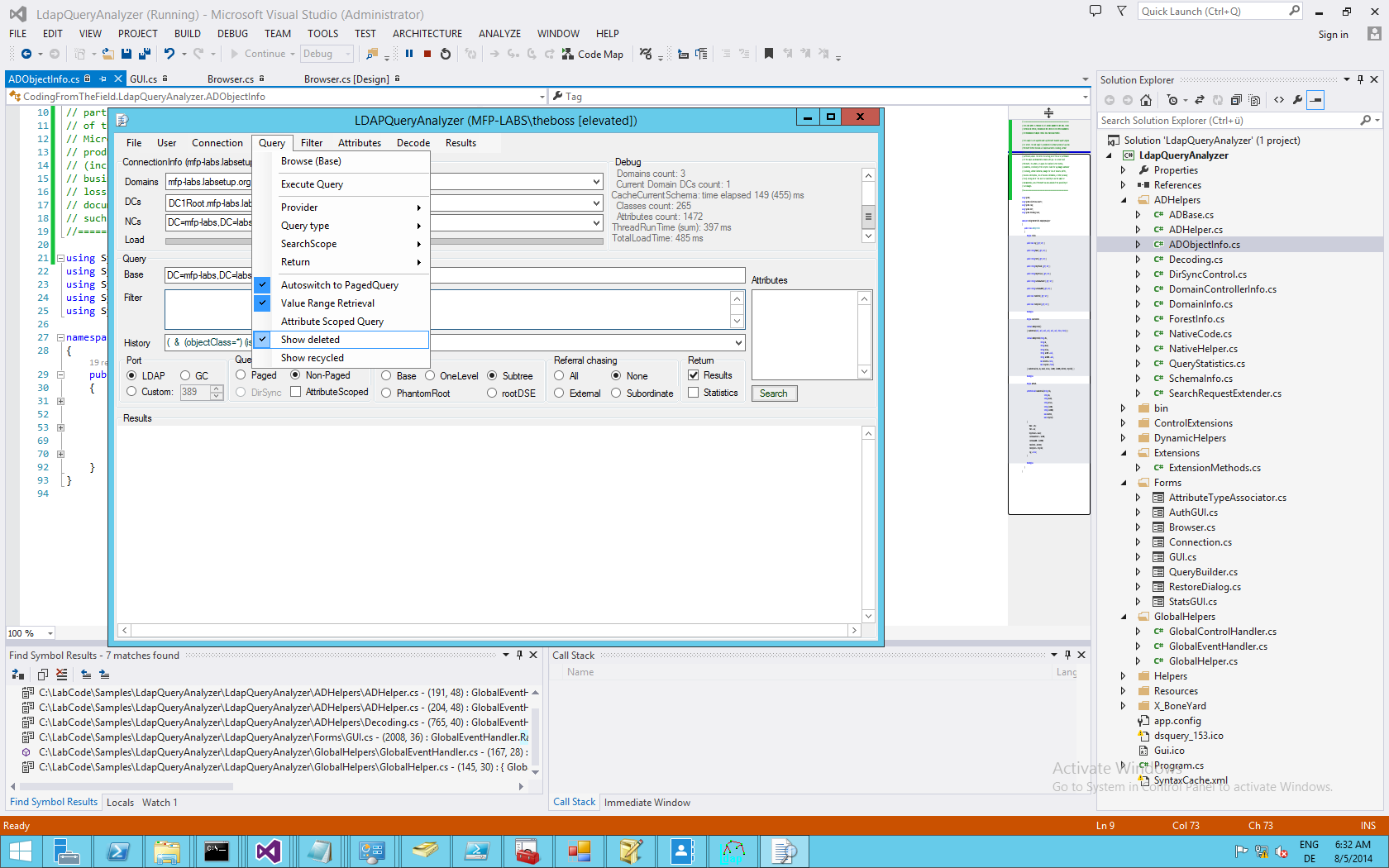
Results  
Select to display the query results in the results pane.

Statistics  
Select to retreive query statistics for your query from the DC. To be able to retreive the information from a DC you need to have the DebugPrivilege assigned to the calling user on the DC.

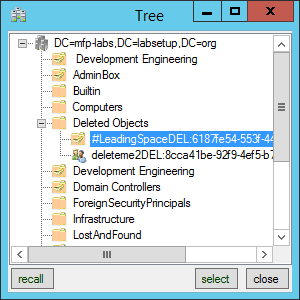


Search  
Send the query to the selected DC.

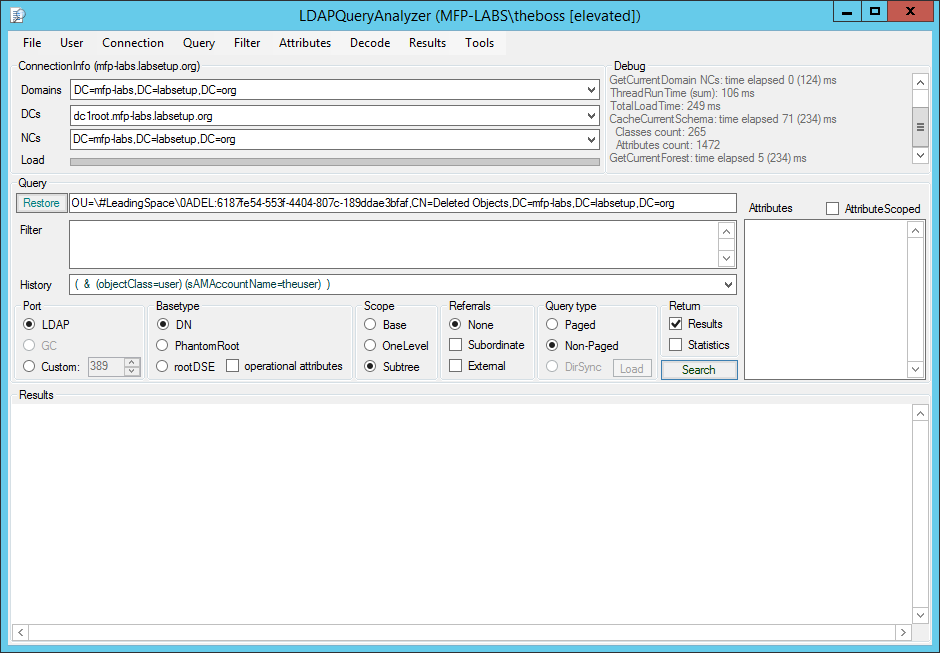
Restore  
Restore a deleted, non-recycled object. Note – you must run the app elevated.  
To do so you must first enable ShowDeletedControl in the Query menu:



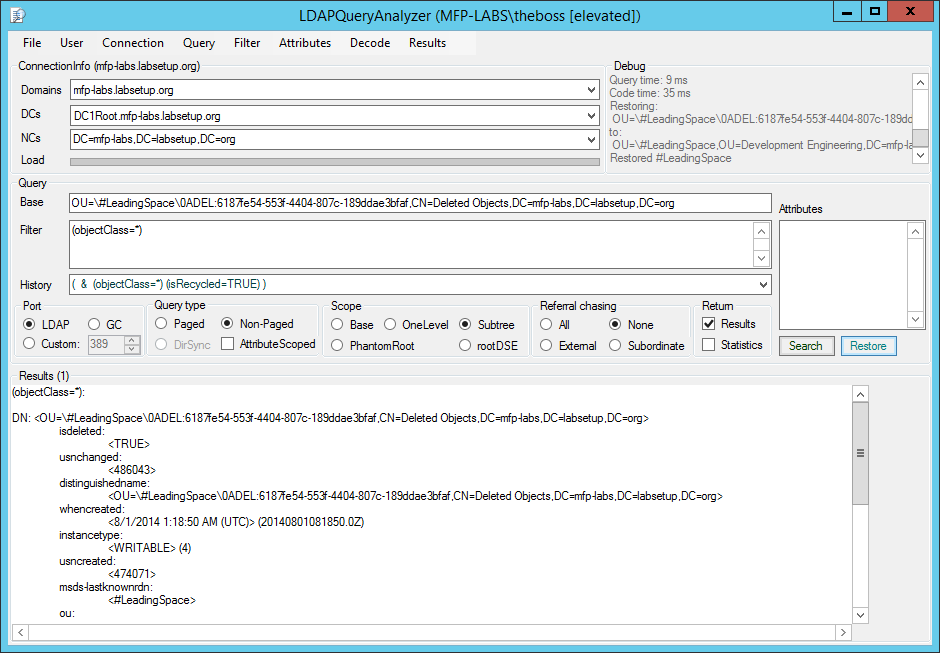
Then click <Browse> menu item from Base field context menu and select the object to be restored in the tree view:



Now you will find the <Restore> button besides the Base field (and the Base field context menu item <Restore> as enabled):



Clicking the button restores the object if possible.



If you enabled ShowRecycledControl as well and select a recycled object, the option to restore will not be available.

### Menus

#### File

Close  
Close app.

#### User

Restart elevated  
Restart app elevetad (UAC). Only available if the executing user has the Administrators group disabled in his token.

Run elevated (2nd instance)  
Run app elevetad (UAC) in a new instance. Only available if the executing user has the Administrators group in his token.

Restart as  
Restart app as a different user.

Restart as elevated  
Restart app elevated (UAC) as a different user.

Run as (2nd instance)  
Restart app as a different user in a new instance.

Run as elevated (2nd instance)  
Restart app elevated (UAC) as a different user in a new instance.

#### Connection

Refresh Forest  
Recall forest info collection for current forest.

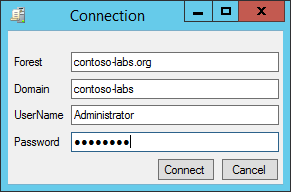
##### Set Ports

LDAP  
Define the LDAP port (default 389) used for DC connection.

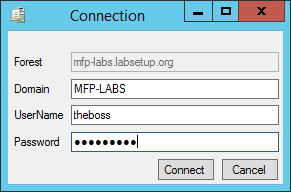
GC  
Define the GC port (default 3268) used for DC connection.

Set Timeout (s)  
Set timeout for DC connections to establish and queries to return results.

Connect Forest  
Connect to a different forest with the possibility to pass different credentials than the exeuting user. If Password field is empty the executing user’s credentials are used to connect.

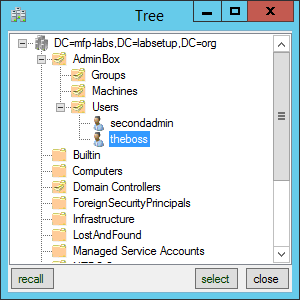


Pass Credentials  
Reconnect current forest with given credentials.



#### Query

Browse (Base)  
Load a browsing GUI to find object paths in your AD and set path in Base field.



Execute Query  
Send the query to the selected DC.

##### Provider

LDAP  
The LDAP port (default 389) is used for DC connection. Default value can be defined in context menu Define LDAP Port:.

GC  
The Global Catalog (GC) port (default 3268) is used for DC connection. Default value can be defined in context menu Define GC Port. This option is only available when the selected DC is also a GC.

##### Base type

DN  
Standard base type -> a distinguishedName (defined by Base field) is used as search base.

PhantomRoot  
Performing a subtree search against the phantom root of the forest (null-DN) with search scope SubTree. Thus you will be able to find any object in the whole forest – even if you have multiple parallel trees in your forest.

rootDSE  
Send an UDP ping (rootDSE call) against the selected DC with an empty search base and search scope Base.

##### Query type

Paged  
The requested query will be send as paged query. Usefull if you expect more results to be returned as the MaxPageSize value defined in lDAPAdminLimits. Caveat – query index intersection ini this case results in the usage of only one indexed attribute in the query even though there would be more indexed attributes used in the query - possible performance impact.   
Therefore the switch Autoswitch to PagedQuery was implemented in the Menu Query - see description in option Non-Paged below.

Non-Paged  
We first try send a non paged query. If the result count exceeds the MaxPageSize value defined in lDAPAdminLimits we only return the results up to MaxPageSize.  
Per default the switch Autoswitch to PagedQuery from the Menu Query ist activated. This will resend the query as a paged query and get all results returned.

##### SearchScope

Base  
Performing a base search against the object path defined in Base.

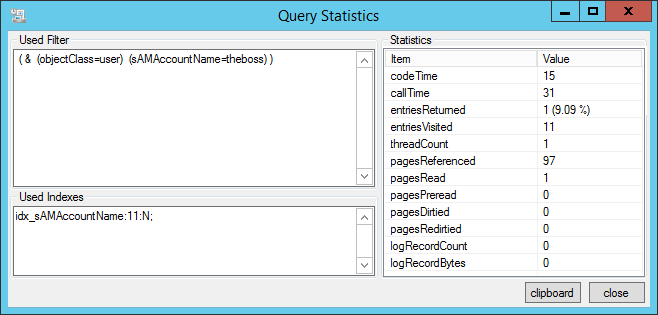
OneLevel  
Performing an one level search against the object path defined in Base.

SubTree  
Performing an subtree search against the object path defined in Base.

##### Return

Results  
Select to display the query results in the results pane.

Statistics  
Select to retreive query statistics for your query from the DC. To be able to retreive the information from a DC you need to have the DebugPrivilege assigned to the calling user on the DC.



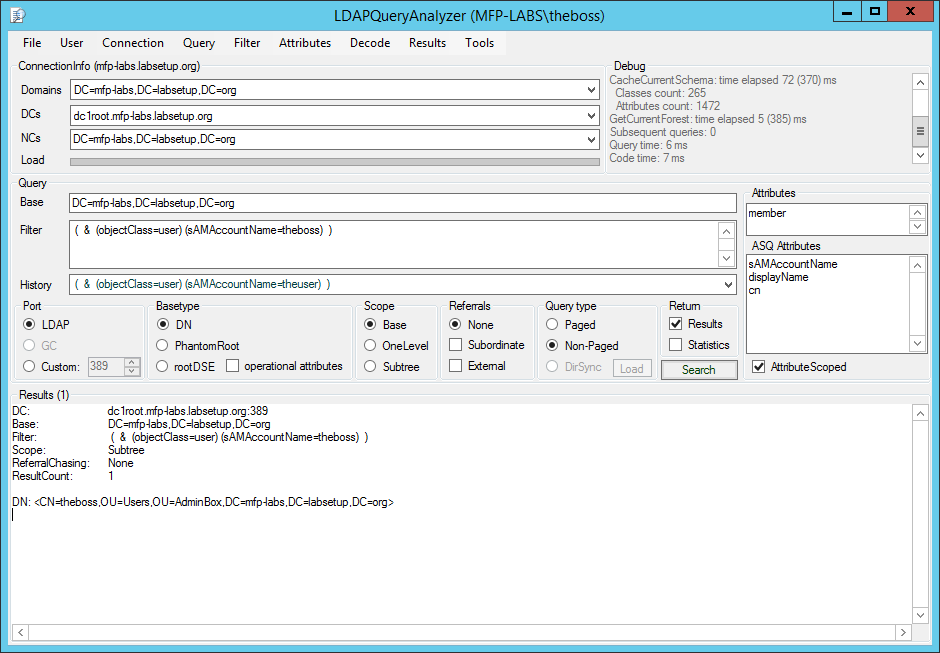
###### operational rootDSE attributes

If selected you get back the operational attributes when sending an UDP ping (rootDSE call). You may customize the operational attributes to be returned by editing the the RootDSE.xml in the folder .\Cache\Settings. To add a new attribute to the operational attribuites list just insert a new line like <string>*newAttribute*</string> to the <OperationalAttributes> node.

Autoswitch to PagedQuery  
When selected - if the result count exceeds the MaxPageSize value defined in lDAPAdminLimits we do not only return the results up to MaxPageSize - we will resend the query as a paged query and get all results returned.

Value Range Retreival  
If you ask for a multivalued attribute to be returned and the value count exceeds MaxValRange value defined in lDAPAdminlimits we return the values up to MaxValRange. If selected we do a value range retreival and get back all values.

Attribute Scoped Query  
Attribue scoped queries (ASQ) can be send against any linked attribute pair member, like for example:

* member / memberOf
* manager / directReports

An ASQ must be send against the path of an object that holds this attribute as a base search. In the Attributes field we need the attribute we want to do the ASQ against as only entry in the list. The attributes in the field below define the attributes you want to get returned from the objects who are actually stored via their distinguishedName (DN) in the queried attribute.

Show deleted  
If selected we also return deleted objects in the query results. Additionally you will find deleted objects in the browser GUI for base field as well.

Show recycled  
If selected we also return recycled objects in the query results. Additionally you will find recycled objects in the browser GUI for base field as well.

Sort results ascending  
Returned result set is sorted ascending (in code, not on DC) by distinguishedName

Sort results descending  
Returned result set is sorted descending (in code, not on DC) by distinguishedName

#### Filter

Filter Wizard  
Call the FilterWizards wich helps you to create LDAP filters.

Clear Filter History  
Clear the remembered filters in the filter history.

#### Attributes

Load Attribute List from Schema  
Show a list of all available attributes from the schema (including custom attributes).

Hide Attribute List  
Hide the list of all available attributes from the schema.

#### Decode

GUIDs  
Decode GUID attributes from byte arrays to String-GUIDs

SIDs  
Decode SID attributes from byte arrays to String-SIDs

userParameters  
Decode RAS- and TerminalServices-Settings from userParameters attribute.

Resolve Sids  
Reolve SIDs to names.

SecurityDescriptors  
Display decoded SecurityDescriptor (SD) instead of displaying SDDL-String of SD. If Resolve Sids is seletced, the trustees in the ACLs are resolved to names.

OctetStrings  
Not implemented yet.

ReplicaLinks  
Not implemented yet.

memberOf includes primaryGroupID  
Every SecurityPrincipal which is not a group in AD has a group RID (Relative identifier – last sequenze of the SID) set in primaryGroupID.   
This group is added to the memberOf list when logging on with a SecurityPrincipal.   
When selected the primaryGroupID is translated to the name of the group and added to the memberOf list displayed.

To local time  
Not implemented yet.

#### Results

Copy selected  
Copy selected result pane content into clipboard.

Copy all  
Copy result pane content into clipboard.

##### Remembered results

Show first  
Jump to the first result in the remembered results cache.

Show previous  
Move to the previous result in the remembered results cache.

Show next  
Move to the next result in the remembered results cache.

Show current  
Move to the last result in the remembered results cache.

Clear list  
Clear results cache list.

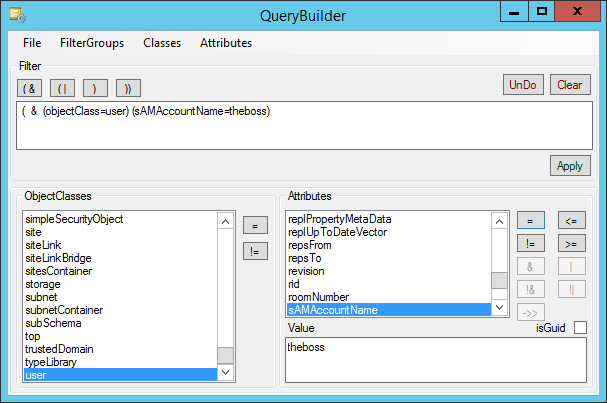
##### Clear Output Clear current result pane.

Change Font  
Change display font in result pane.

#### Tools

DynamicTypeBuilder  
The tool has no hardcoded translations for attribute syntax. We collect information from the Schema and cache this in a XML. Every time the modifiedTimeStamp of the Schema in a forest is updated we refresh the Schema cache.  
Since many attributes are enums, OID strings or blobs we implemented a way to dynamically decode those attributes. See detailed description of DynamicTypeBuilder.

## FilterWizard



### Fields

#### Filter

Auto-filled textbox showing the currently configured filter. Trailing goup closing bracktes from auto generated groups are closed automatically when applying filter to main window.

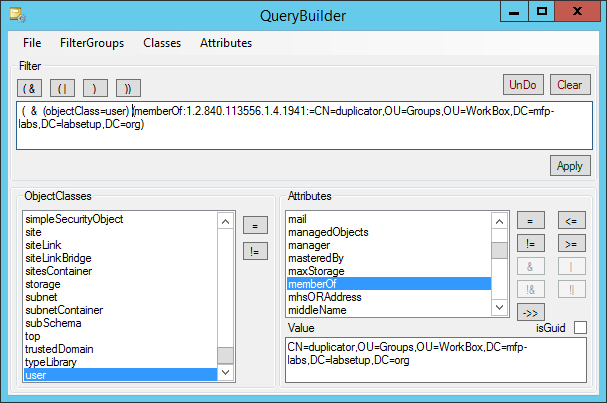
#### ObjectClasses

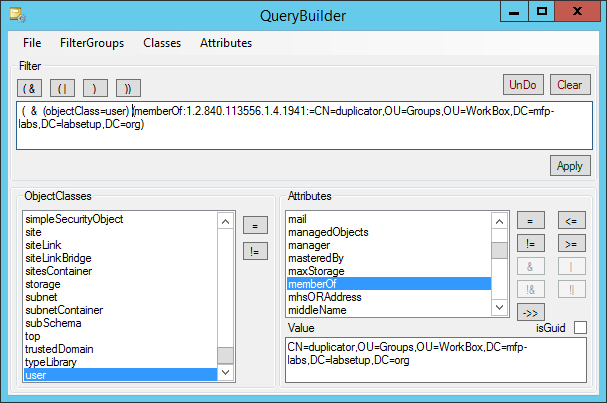
A list of all available classSchema objects in the current forest’s schema. Read from schedma cache xml.

#### Attributes

A list of attributes assigned to the selected classSchema object and it’s sub classes.

If the selected attribute is defined as part of a linked attribute pair the ->> control (Match In Chain) will be enabled.   
Synatx of <Value> has to be a distinguishedName.  
This adds the :1.2.840.113556.1.4.1941: operator to the filter for this attribute.



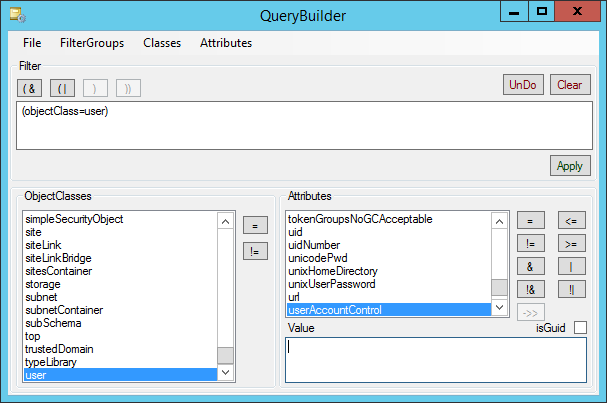


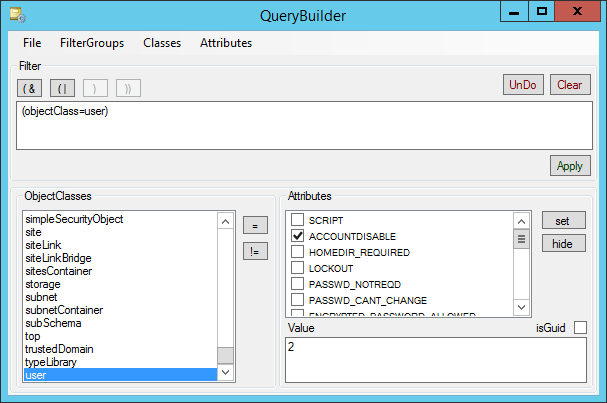
If the selected attribute is defined as attribute syntax integer or enumeration the controls

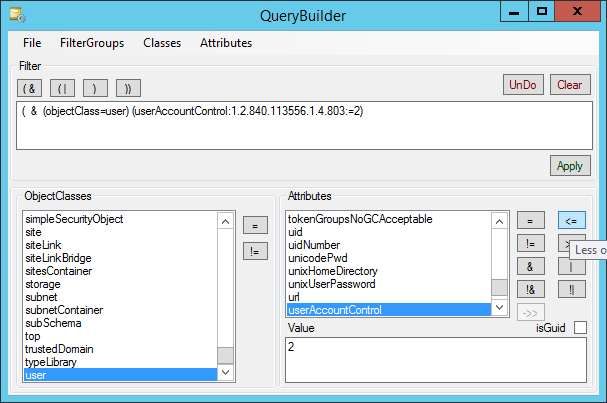
* BitWise AND &
* BitWise OR |
* Not BitWise AND !&
* Not BitWise OR !|

will be enabled.

By clicking one of these buttons a list of the exisiting enum values will be displayed (if enum is defined in the DynamicTypeBuilder). Multi-select in the list is alowed.







Otherwise you may just type the numeric value in the Value field.

This adds the operator :1.2.840.113556.1.4.803 for bitwise AND and :1.2.840.113556.1.4.804: for bitwise OR.

#### Value

The value defining the filter for the currently selected attribute.

### Controls

#### Open AND group (&

Opens a new AND group to combine filters where all filter clauses grouped here must match for a hit.

If you do not open any group before selecting filter clauses we automatically open an AND group if more than one filter clause is selected.

#### Open OR group (|

Opens a new Or group to combine filter clauses where only one or more of the filter clauses grouped here must match for a hit.

#### Close current group )

If you opened a group manually you can close it here.  
All automatically opened groups are closed automatically when applying the filter to the main window.

#### Close all groups ))

If you opened groups manually you can close them here automatically.  
All automatically opened groups are closed automatically when applying the filter to the main window.

#### UnDo

Remove the last selected filter clause.

#### Clear

Remove all selected filter clauses and groups.

#### Apply

Apply the constructed filter to the main window.

#### ObjectClasses

Equals =  
Adds a filter clause (objectClass=*selected schemaClass*).

Not Equals !=  
Adds a filter clause (! (objectClass=*selected schemaClass*) ).

#### Attributes

Equals =  
Adds a filter clause (*selected attribute*=*Value*).

Not Equals !=  
Adds a filter clause ( ! (*selected attribute*=*Value*) ).

Less or Equals <=  
Adds a filter clause (*selected attribute*<=*Value*) .  
Note – we do not offer ‘Not Less or Equals’ -> this is no recommended filter clause – use ‘Greater or Equals’ instead.

Greater or Equals >=  
Adds a filter clause (*selected attribute*>=*Value*).  
Note – we do not offer ‘Not Greater or Equals’ -> this is no recommended filter clause – use ‘Less or Equals’ instead.  
  
  
Bitwise AND &  
Adds a filter clause (*selected attribute*:1.2.840.113556.1.4.803:=*Value*).  
When this matching rule is used as a clause in a query filter, the clause is satisfied only if all the bits set to '1' in the value included in the clause correspond to bits set to '1' in the value stored in the directory.

Bitwise OR |  
Adds a filter clause (*selected attribute*:1.2.840.113556.1.4.804:=*Value*).  
When this matching rule is used as a clause in a query filter, the clause is satisfied only if at least one of the bits set to '1' in the value included in the clause corresponds to a bit set to '1' in the value stored in the directory.

Not Bitwise AND !&  
Adds a filter clause ( ! (*selected attribute*:1.2.840.113556.1.4.803:=*Value*) ).  
When this matching rule is used as a clause in a query filter, the clause is satisfied only if none the bits set to '1' in the value included in the clause correspond to bits set to '1' in the value stored in the directory.

Not Bitwise Or !|  
Adds a filter clause ( ! (*selected attribute*:1.2.840.113556.1.4.804:=*Value*) ).  
When this matching rule is used as a clause in a query filter, the clause is satisfied only if at none of the bits set to '1' in the value included in the clause corresponds to a bit set to '1' in the value stored in the directory.

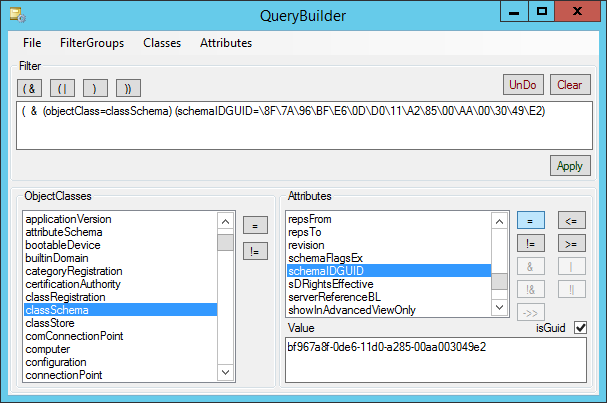
Match in Chain ->>  
Adds a filter clause (*selected attribute*:1.2.840.113556.1.4.1941: =*distinguishedName in* *Value field*).  
Allows you to retrieve for example:

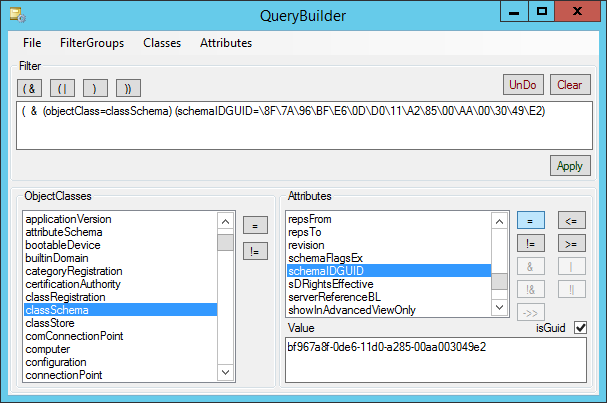
* + all direct and nested group memberships of a user
  + all direct and nested members of a group
  + following the directReports or manager chain
  + …

This is a very expensive call since we must walk the link table of the Active Directory database to find all objects in the link chain.

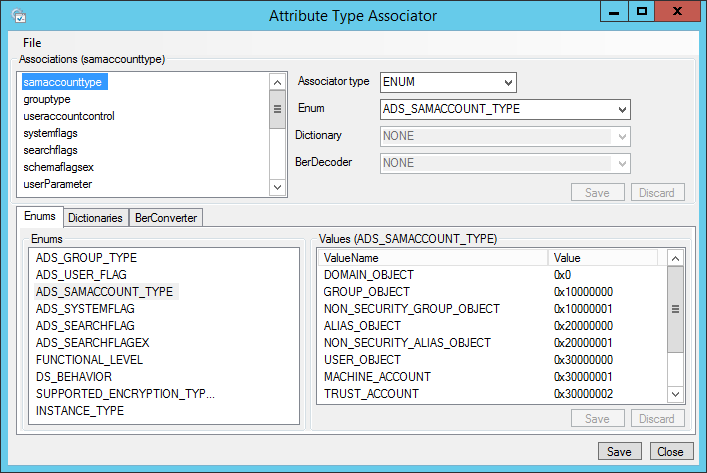
isGuid  
If the selected attribute is of attribute syntax GUID you can type a string Guid (like bf967a8f-0de6-11d0-a285-00aa003049e2. The GUID will be translated to the hex expression of the GUID to build the proper syntax.

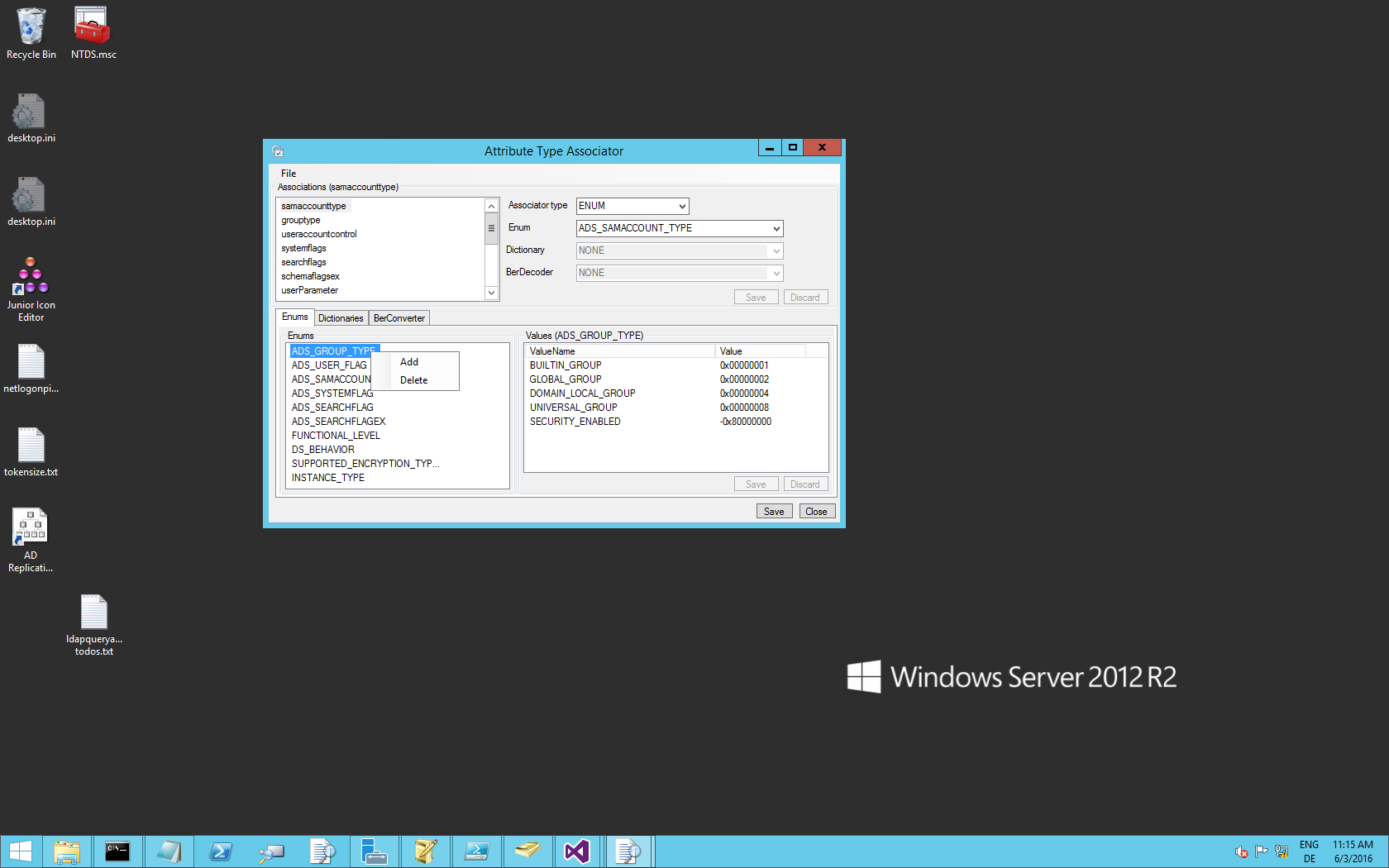
If as GUID attribute is not marked with GUID syntax, you may check isGuid checkbox to get the expected result.

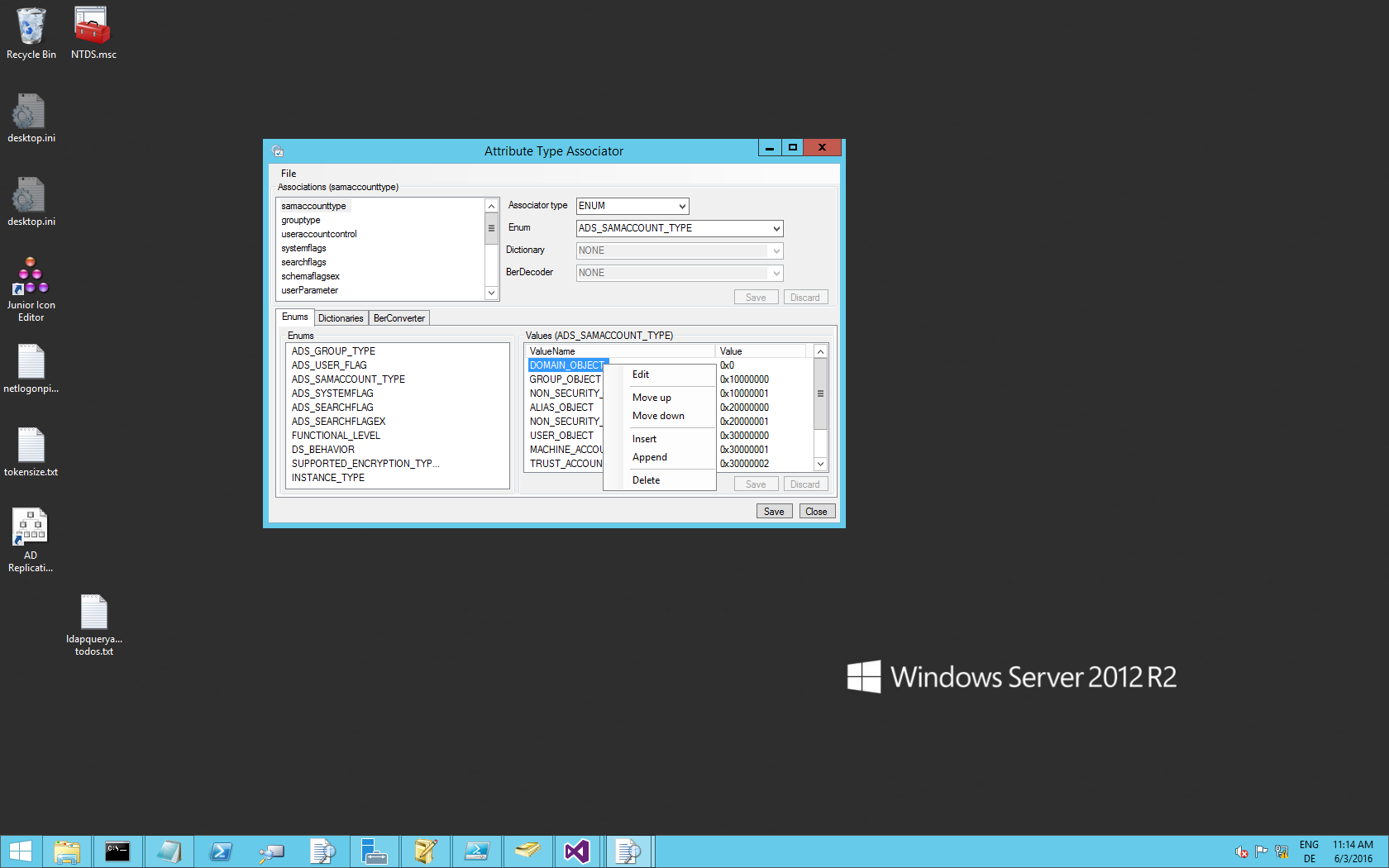


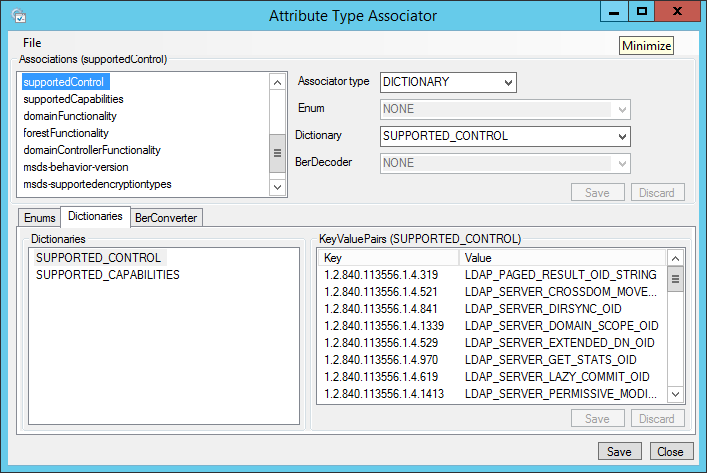


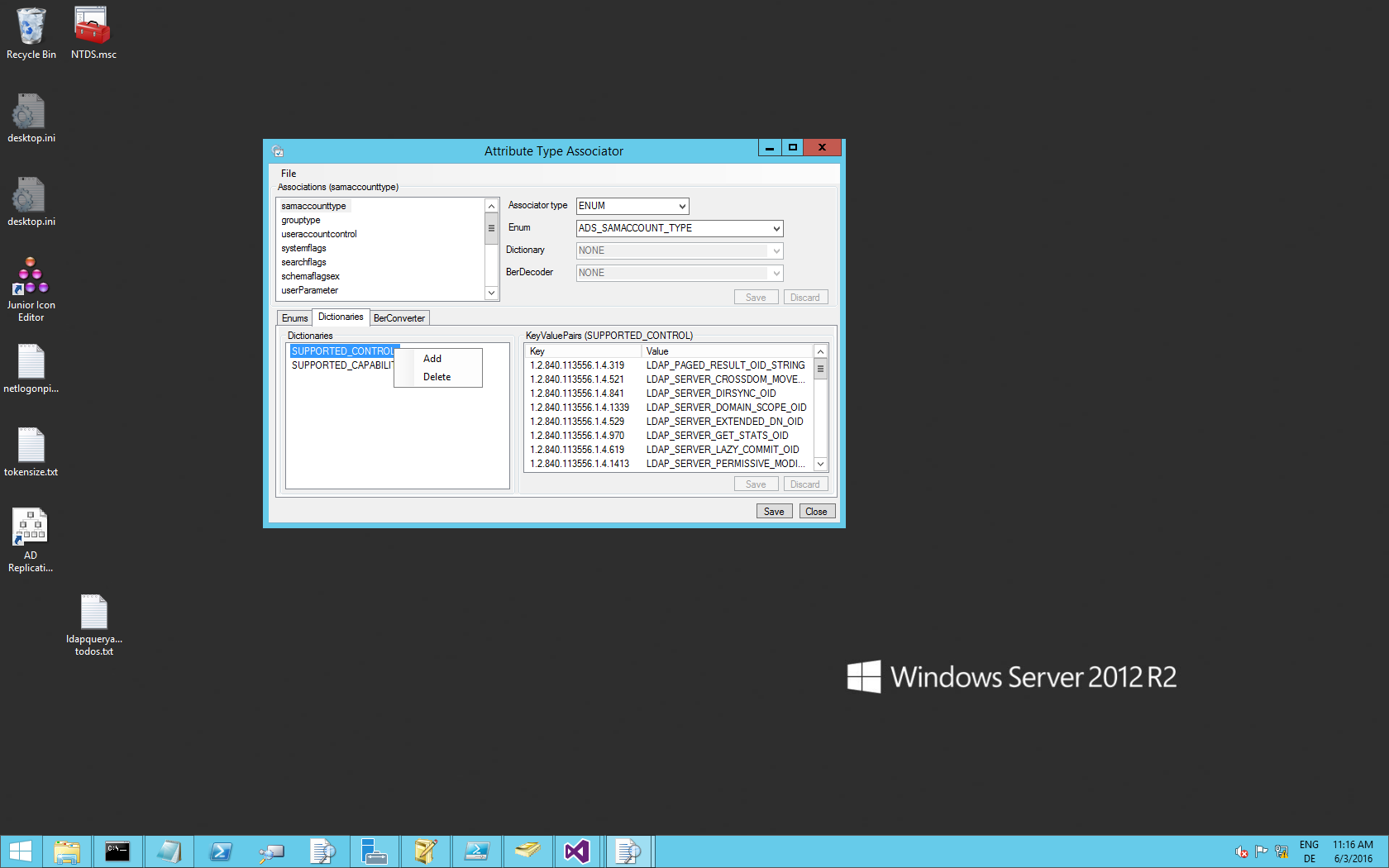
## DynamicTypeBuilder

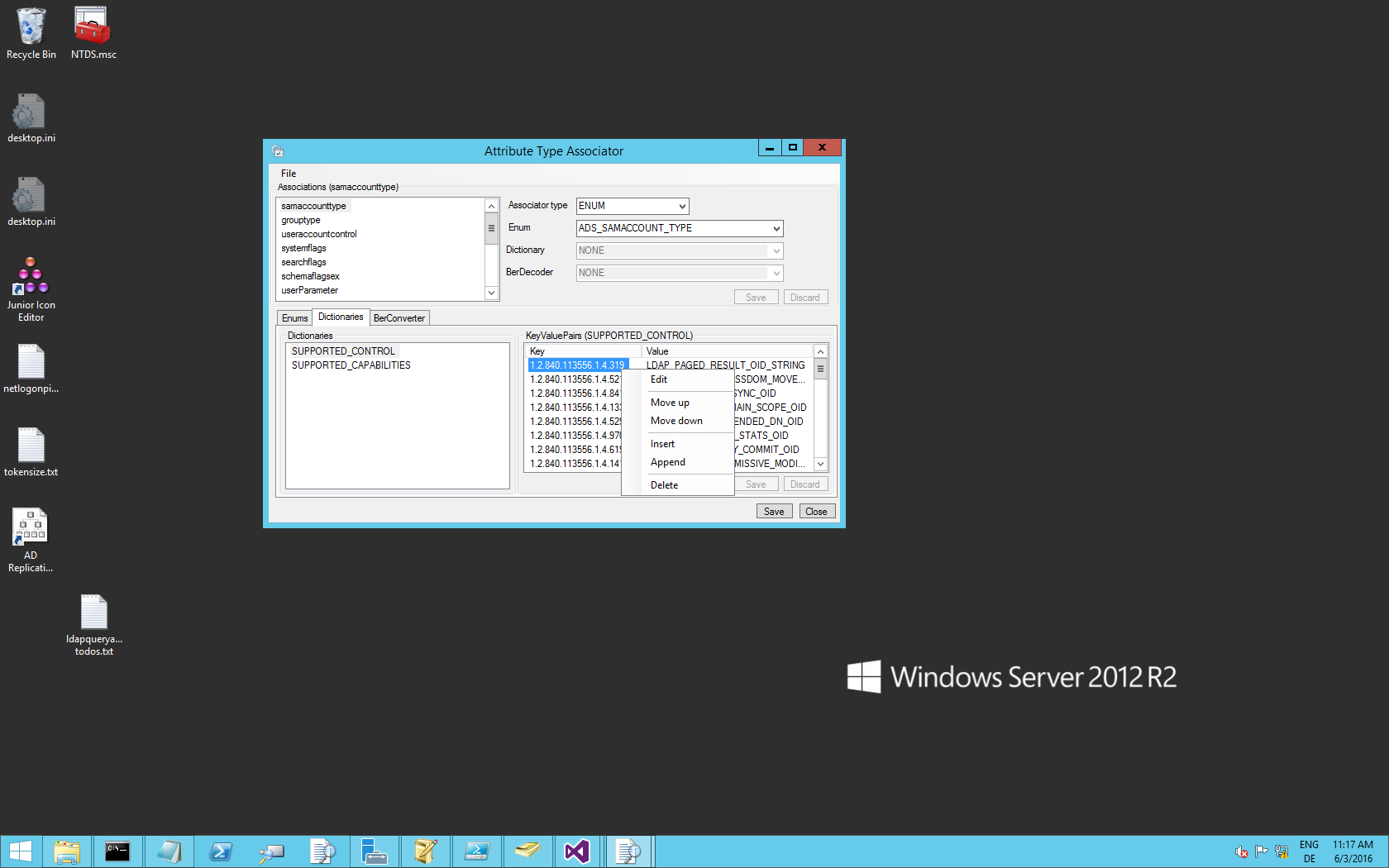


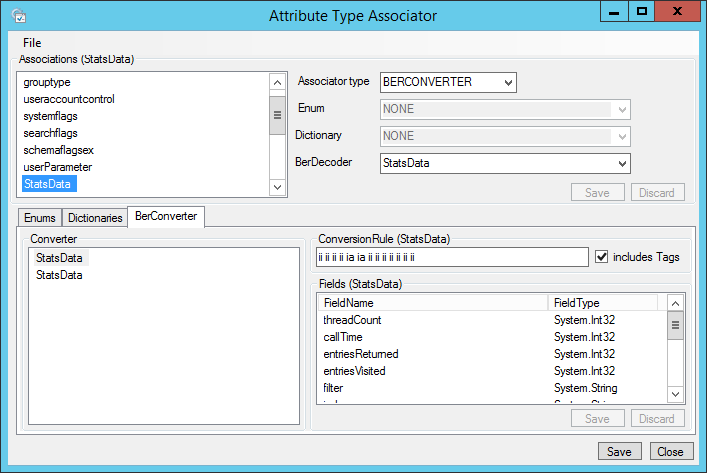


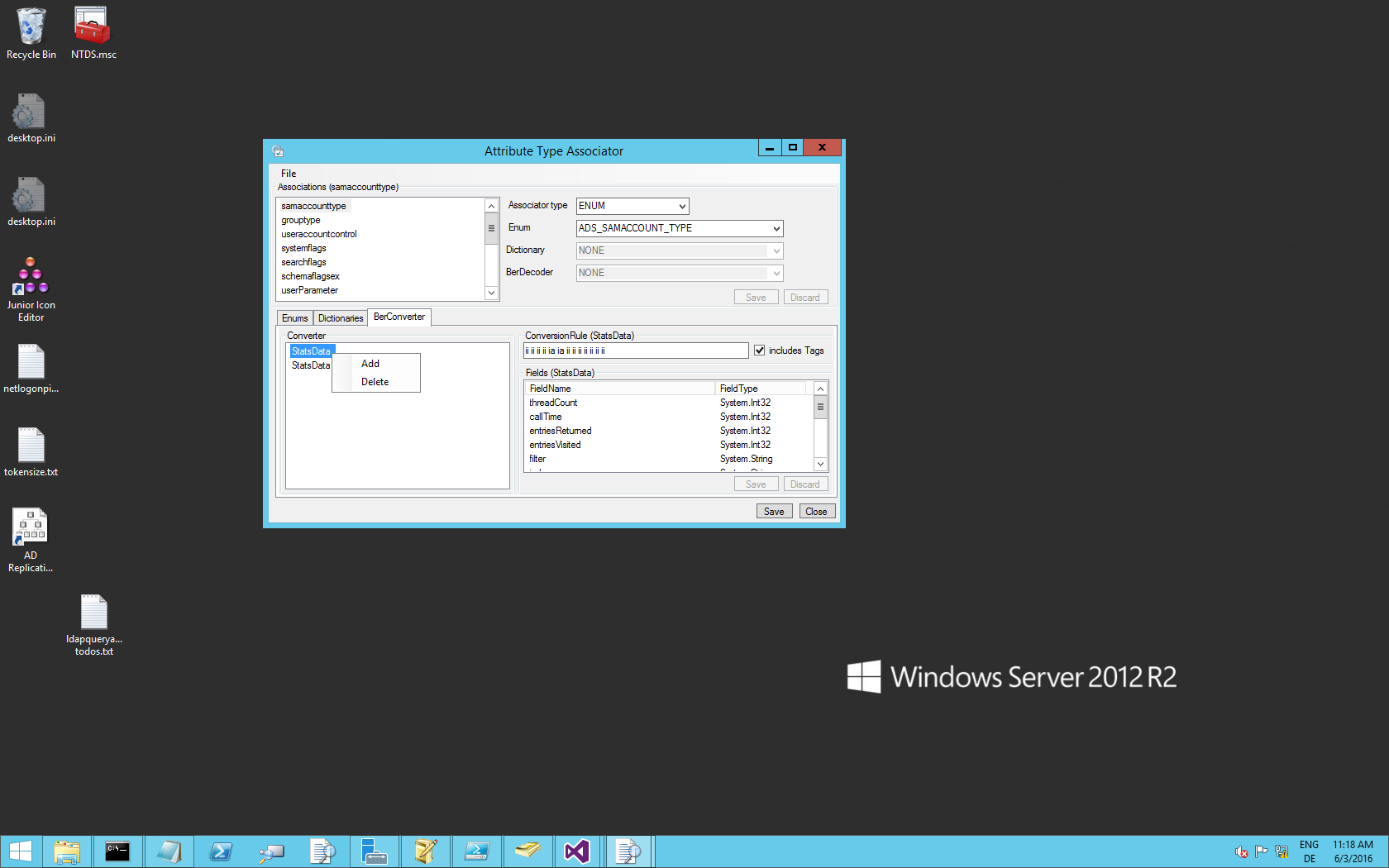


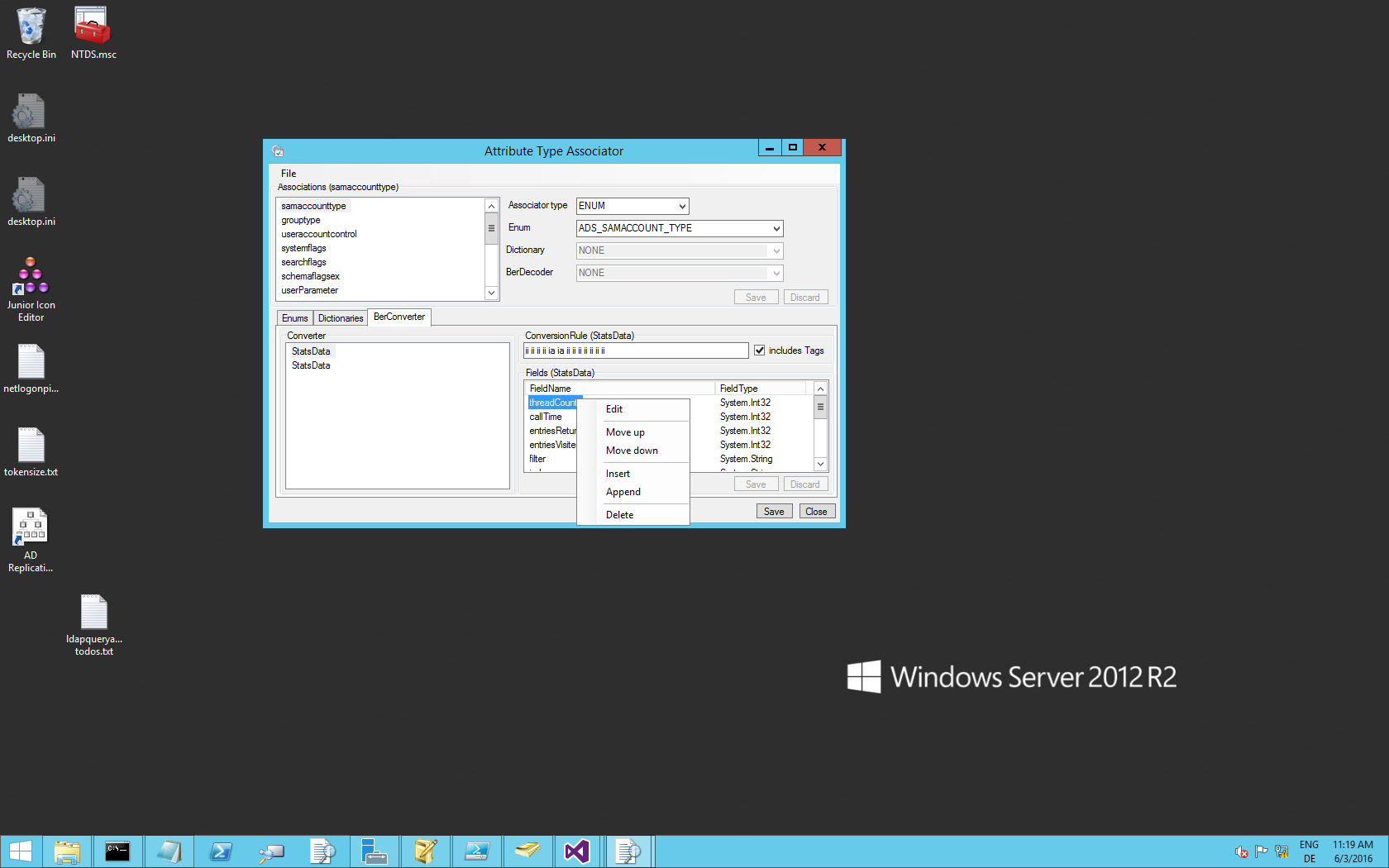












## Disclaimer

// =========================================================================

// THIS CODE-SAMPLE IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER

// EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES

// OF MERCHANTABILITY AND/OR FITNESS FOR A PARTICULAR PURPOSE.

// This sample is not supported under any Microsoft standard support program

// or service. The code sample is provided AS IS without warranty of any kind.

// Microsoft further disclaims all implied warranties including, without

// limitation, any implied warranties of merchantability or of fitness for a

// particular purpose. The entire risk arising out of the use or performance

// of the sample and documentation remains with you. In no event shall

// Microsoft, its authors, or anyone else involved in the creation,

// production, or delivery of the script be liable for any damages whatsoever

// (including, without limitation, damages for loss of business profits,

// business interruption, loss of business information, or other pecuniary

// loss) arising out of the use of or inability to use the sample or

// documentation, even if Microsoft has been advised of the possibility of

// such damages.

//=========================================================================