# netTerrain 10.1

## Import / Export Guide

# Contents

Document Code.**GN_D_nT10-06**

Last revision: **02/18/2026**

Image: **Leonard Kleinrock** (born June 13, 1934)

Leonard Kleinrock is an American computer scientist known for his work on queueing theory and the pioneering of packet switches, the precursors of the modern router as well as the supervision for the development of the original NCP ARPANET protocol.

**GRAPHICAL NETWORKS**

Graphical Networks LLC

Telephone: +1-240-912-6223

Fax: +1-240-912-6339 (anybody still uses these?)

# 1 About This Guide

This document is divided into the following chapters:

- Chapter 1, About This Guide
- Chapter 2: PDF, Visio and Static Methods
- Chapter 3, netViz Import
- Chapter 4, Excel Import
- Chapter 5, KML and KMZ Files
- Chapter 6: ArcGIS import

## 1.1 Who should use it

This guide is intended for netTerrain administrators tasked with the configuration and set up of various bulk import and export mechanisms.

netTerrain does provide many other import and export mechanisms, including third-party integrations, discovery, database imports and exports as well as API integrations. Those mechanisms are usually better suited for continuous processes as opposed to bulk or one-time imports. They are achieved via the netTerrain discovery and collector interface, the Integration Toolkit (ITK) and the RESTful/SOAP API. None of these continuous mechanisms are covered in this Mickey Mouse guide, for those other methods consult one of the other guides for grown-ups:

- Collector guide
- ITK guide
- Database description and scripting guide
- API programming guide.

## 1.2 Assumptions

This guide assumes that the administrator has knowledge of the following:

- SQL Server administration, including login/security management, backup and restore procedures
- Knowledge of IIS Management, including creation of virtual directories/applications
- Overall knowledge of server administration procedures (installing, uninstalling and managing application settings)

# 2 PDF, Visio, Static Methods

This chapter goes over the following bulk import and export processes:

• PDF exports
• Visio exports and imports
• PowerPoint exports
• Static methods (HTML and PNG)

Since some of the export mechanisms can be executed from the server as well as from the client, whenever applicable, we will cover both.

> **Attention!**
>
> The import and export options will only be available if a netTerrain administrator enabled them in the admin console. Please check the netTerrain Administrator guide to see how that is done.

## 2.1 Client-side PDF export

One of the easiest ways to export diagrams from netTerrain is by using the PDF export option. The PDF option is available from the Import & Export ribbon in netTerrain. If the PDF icon is not present, then it was not enabled by the administrator (see netTerrain admin guide on how to enable the feature). There is no additional software that needs to be installed for the PDF export option to work.

To use the PDF export, follow these steps:

1) Press the PDF icon available from the Export ribbon:
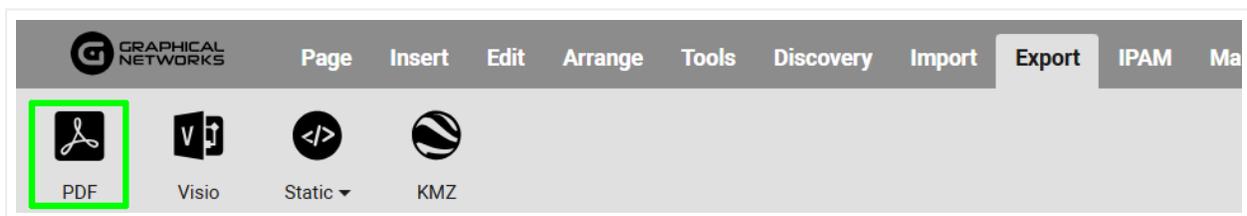


2) A PDF Export options dialog appears
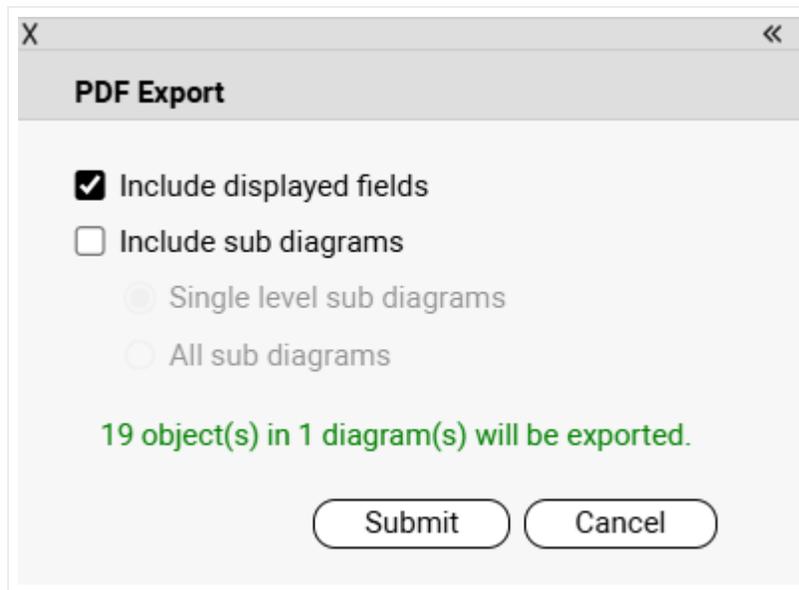
The Include displayed field option will preserve any displayed fields present in the diagrams when exported. The include sub diagrams provides two options:

a. Single Level sub diagrams : will export the current diagram and one level below

b. All sub diagrams: will export the current diagram and all diagrams below

3) Press 'Submit'

> Attention!
>
> This can be a time-consuming process when several diagrams are part of the export. When performing a large export, it may be best to do this during a time of the day when there is less user activity. For instance, when they are all sleeping or at the water cooler chatting non-stop.

Notice that before hitting submit, netTerrain will provide you with a count of nodes to be exported so that you can decide whether a reduction in number of diagrams to export is in order.
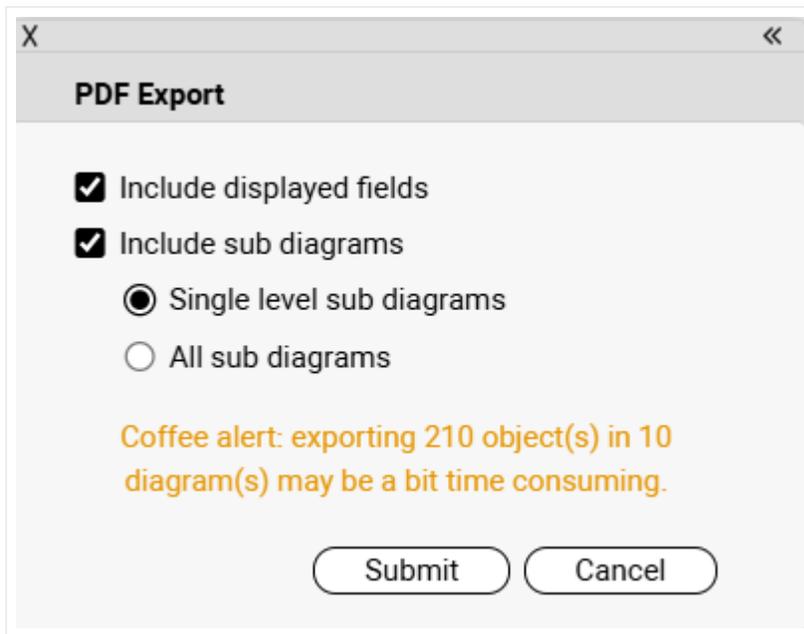
*Message indicating the number of nodes and diagrams to export*

## 2.2 Exporting netTerrain diagrams to Visio, HTML and PNG

netTerrain gives the users the ability export diagrams to other graphical tools:

• HTML, which comes in handy for embedding diagrams on another website or perhaps as part of a frame inside another application
• PNG, which is useful for easy distribution of individual diagrams
• PowerPoint, for that presentation that you need along with your TPS report
• Visio, if you are a masochist, or perhaps for that curmudgeon that also likes to heat up leftover fish in the office

## 2.2.1 Server-side vs client-side exports

Visio exports rendered from the application server require the application to be installed on the server. Often installing these applications on the application server is not possible. netTerrain provides an alternative export mechanism where the client renders the Visio exports, instead of the server.

To enable server-side exports, the administrator can enable that setting from the admin console:
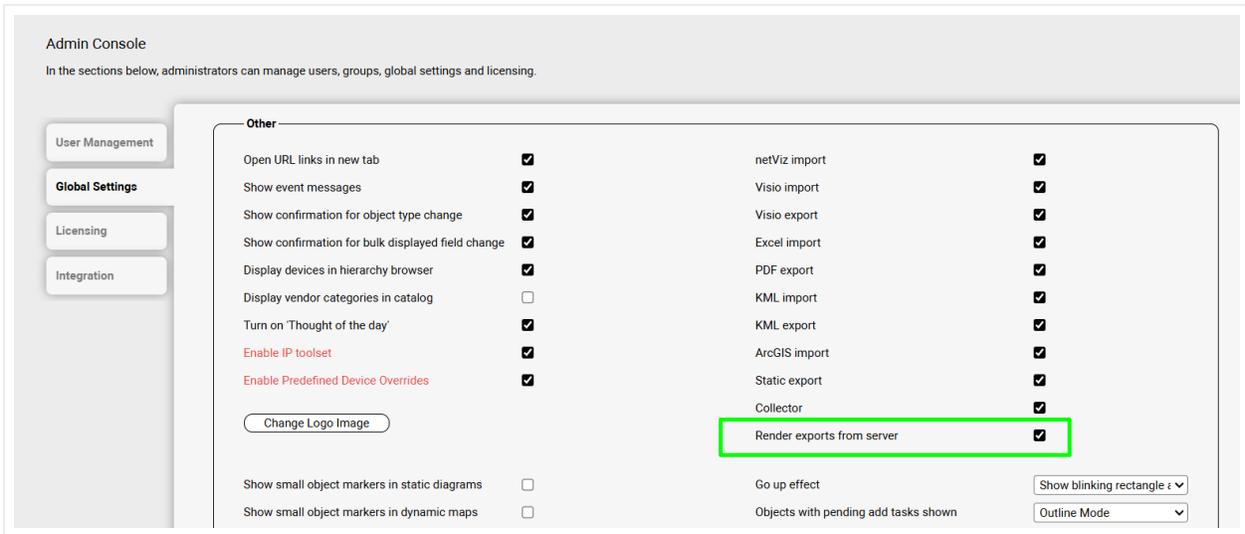
However, when the server-side exports are disabled, users must download the export utility to take advantage of Visio exports.

Also, for the Visio export to show up as an option in the user menus it must enabled from the admin console as shown in the image below:
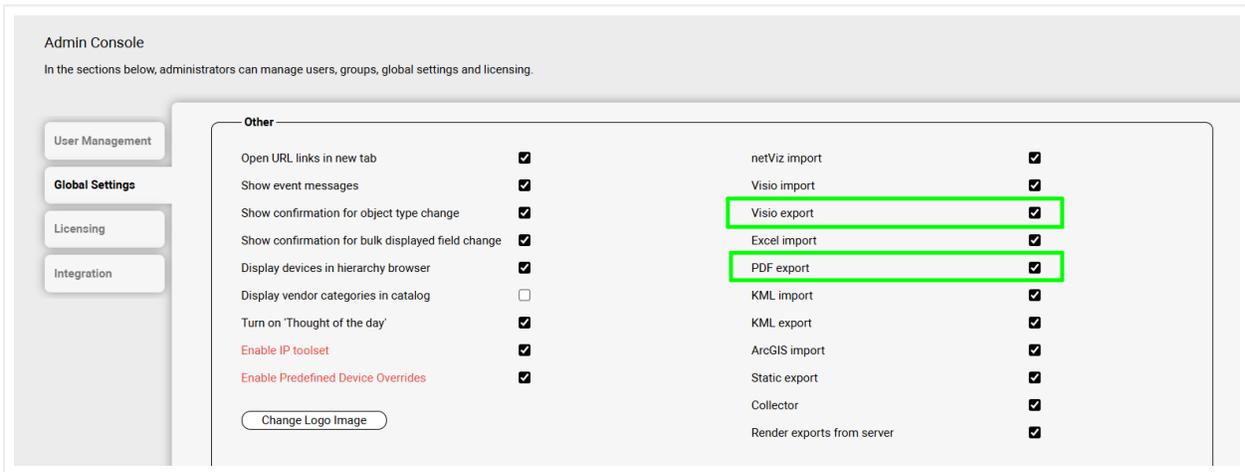


To download the Exporter tool, simply go to Tools->Utilities and click on 'Exporter Tool'.
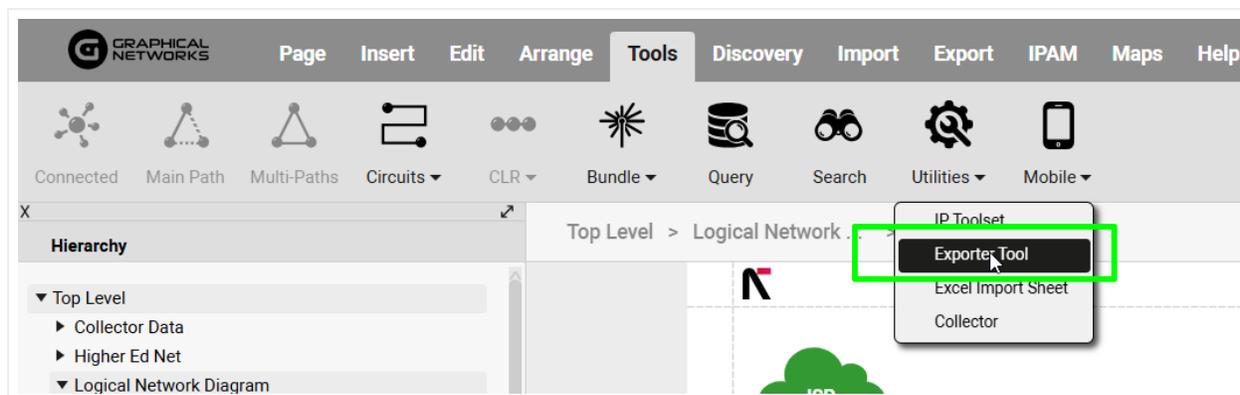
*Exporter tool to export to Visio client-side*

## 2.2.2 Starting the export process

To start the export process click on one of the export buttons in the Export ribbon.
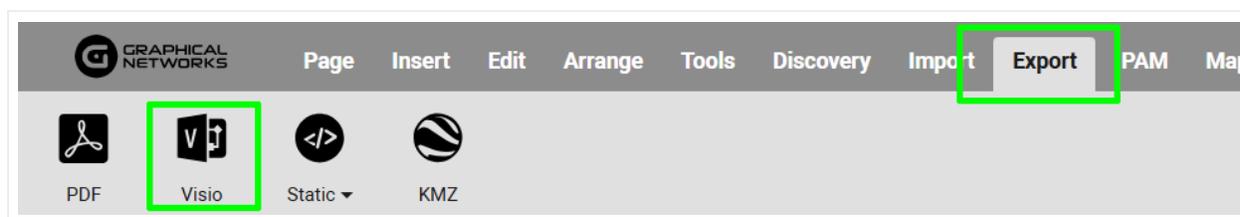


*Visio export option*

## 2.2.3 Visio export options

After clicking on the Visio export button, a dialog showing export options will be displayed, as shown below in the Visio export example:
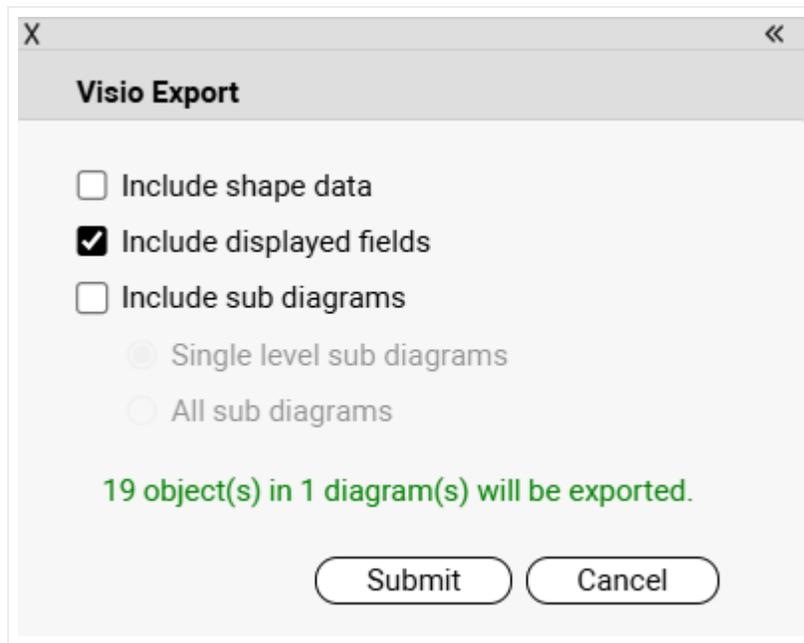
*Visio export options*

The following options are available when exporting Visio files:

- Include Shape Data: exported objects from diagrams will include all field data in the exported Visio diagrams.
- Include displayed fields, which means that any displayed data in the diagram will also be in the Visio exports.
- Include sub diagrams, which will export child diagrams to individual Visio sheets depending on the sub diagram options that are checked. These are:
- Single level sub diagram, which exports the immediate level below
- All sub diagrams exports, which, well, you guessed it, exports everything

Note that the "all sub diagrams" option can take a very long time to process and is not recommended for large sets of diagrams. If your boss really must see the entire collection of diagrams then get ready for a really long coffee break (or better yet, give him access to netTerrain, it's a much better app anyways).

If the file is rendered on the server the file will be made available for download by a server process.

## 2.2.4 Exporting and fetching the files

After clicking on Submit, the Visio export process commences.
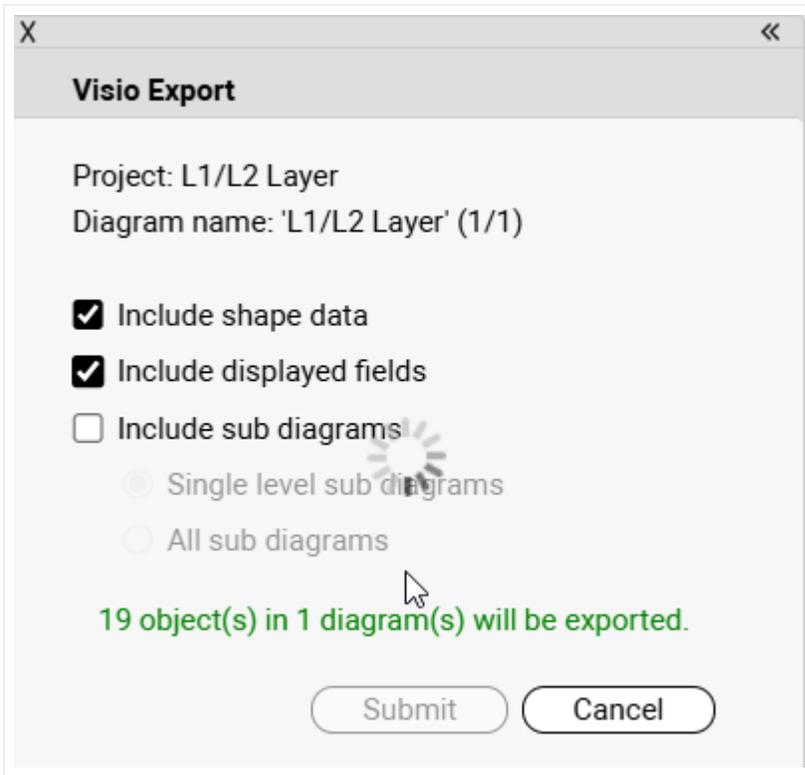
*An example of a Visio export being processed*
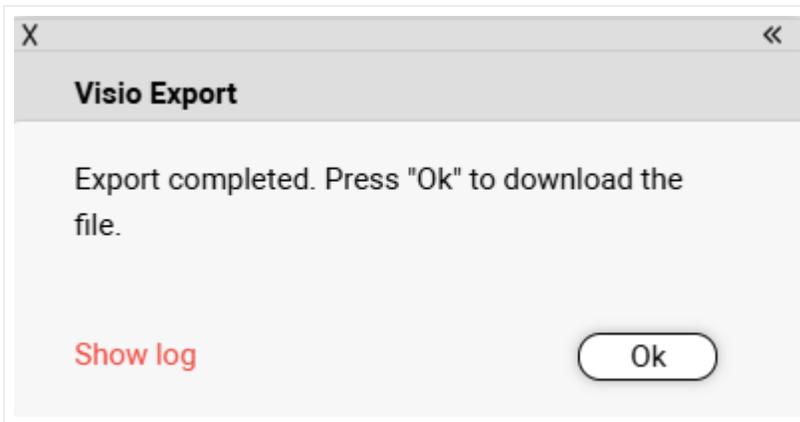


*Completed export process*

If the export is enabled on the client side, an .ntd extension file should be available from your browser download tab.

*Visio export file downloaded for local rendering*

After you double click the .ntd file extension the netTerrain toolset will begin the rendering process.
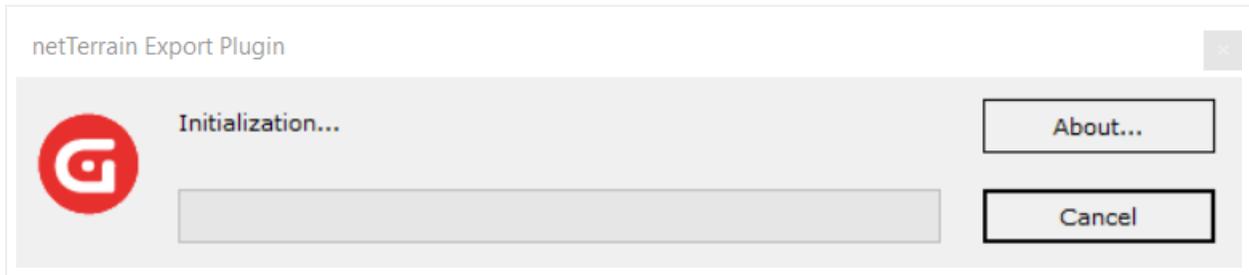


*Visio export file being rendered locally*

When the process is complete the file should automatically open.

If you do not have Visio installed, netTerrain gives you the option to generate the file in PDF.
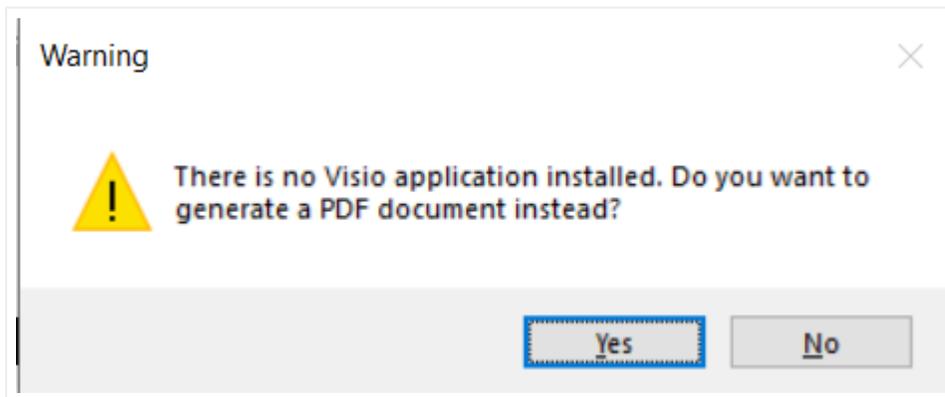


## 2.3 Server-side Visio Import and Export

For the server-side Visio import to work properly, Visio 2010 or 2013 needs to be installed on the netTerrain application server. If you only need to export diagrams and do not want to install Visio on the server, you can render diagrams from a local installation of Visio.

## 2.3.1 Setting it up

Follow these steps to configure the Visio import in netTerrain:

1) On the application server run the 32-bit version of the Microsoft MMC to change the Visio settings. If Visio is 64-bit, then use MMC, only without the -32.
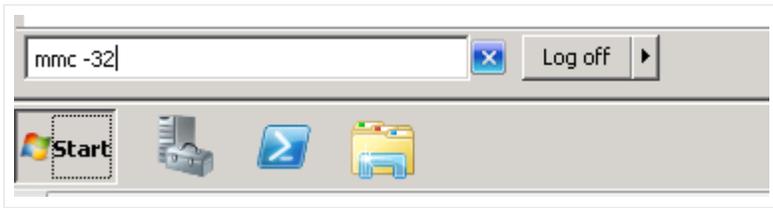
2) The MMC console should start up.
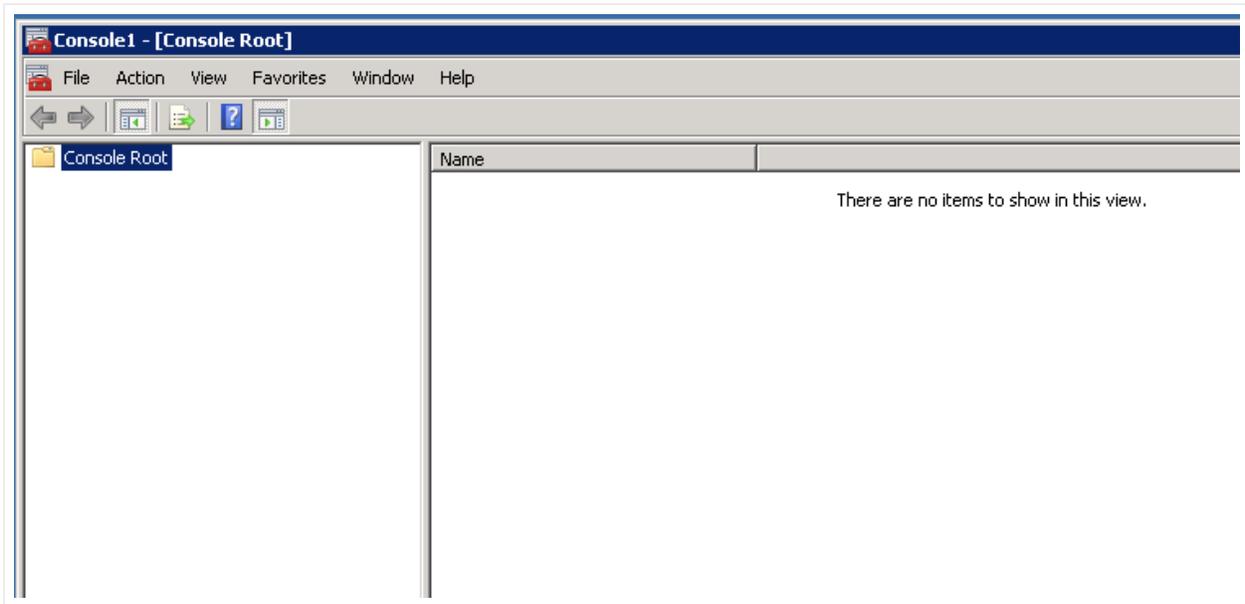


3) Select Add/Remove Snap In.
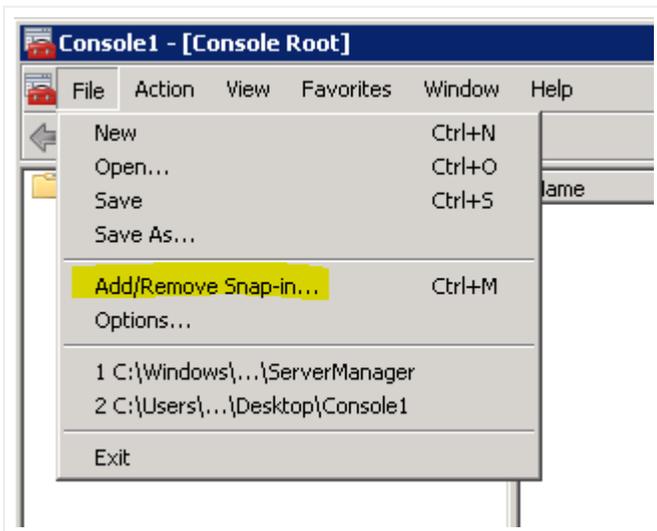


4) Select Component services, press "Add" and then press OK.
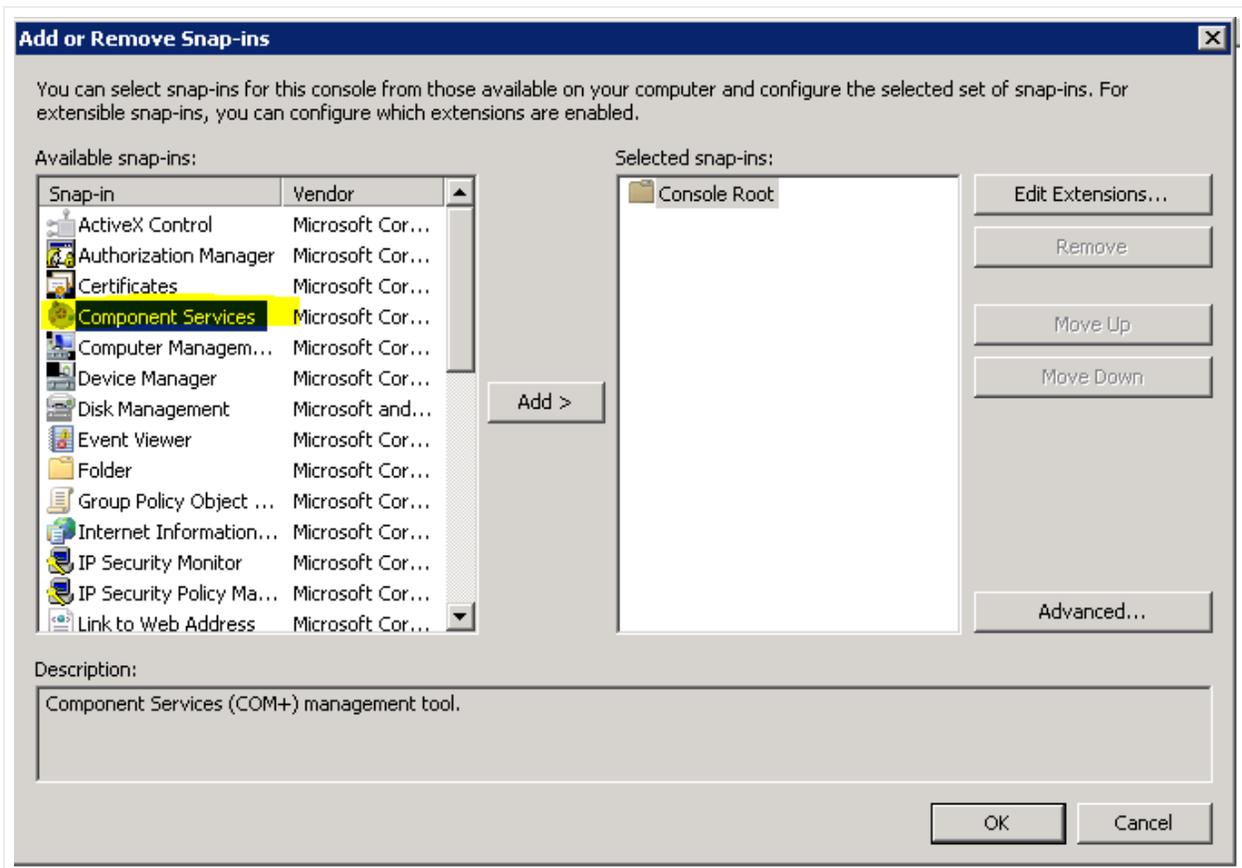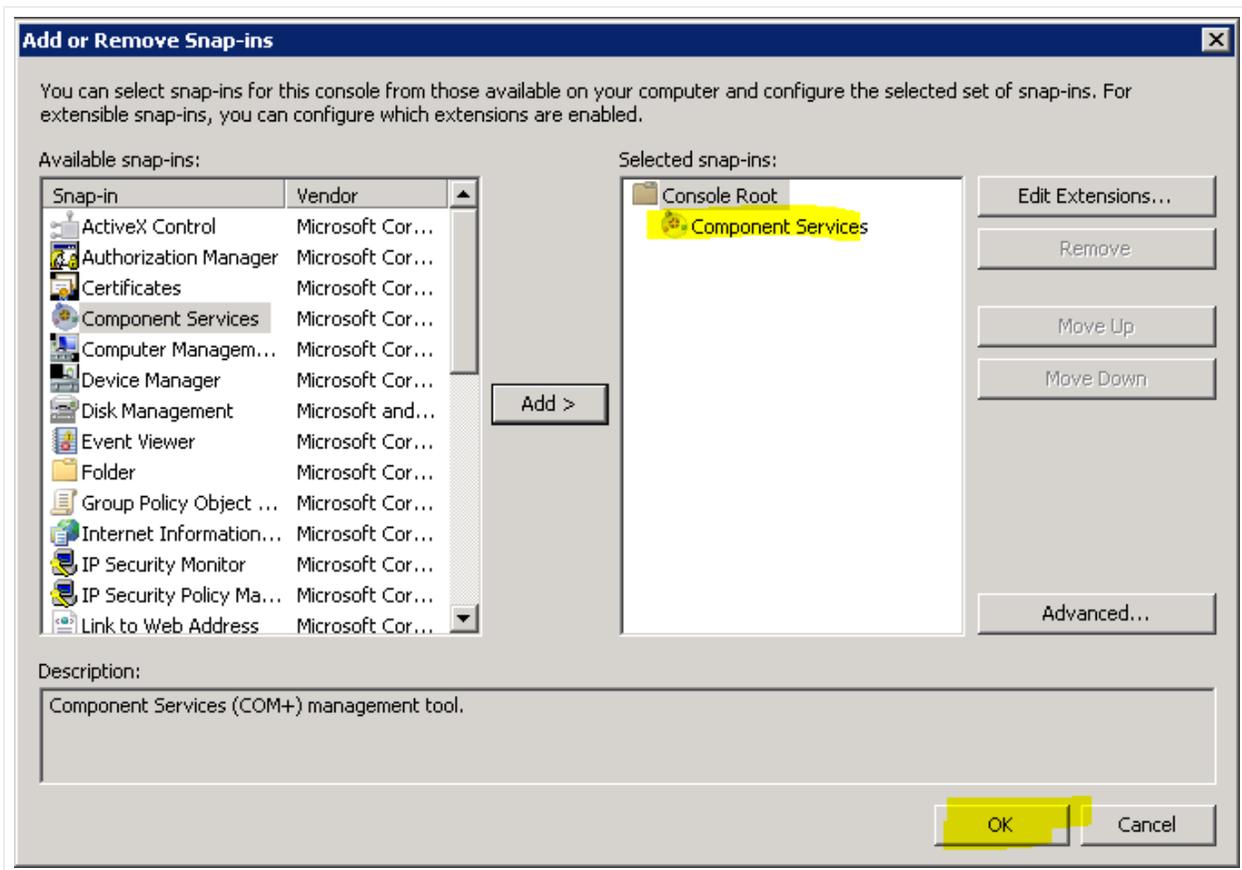
5) Open "DCOM config" and then scroll down to "Microsoft Visio Drawing".
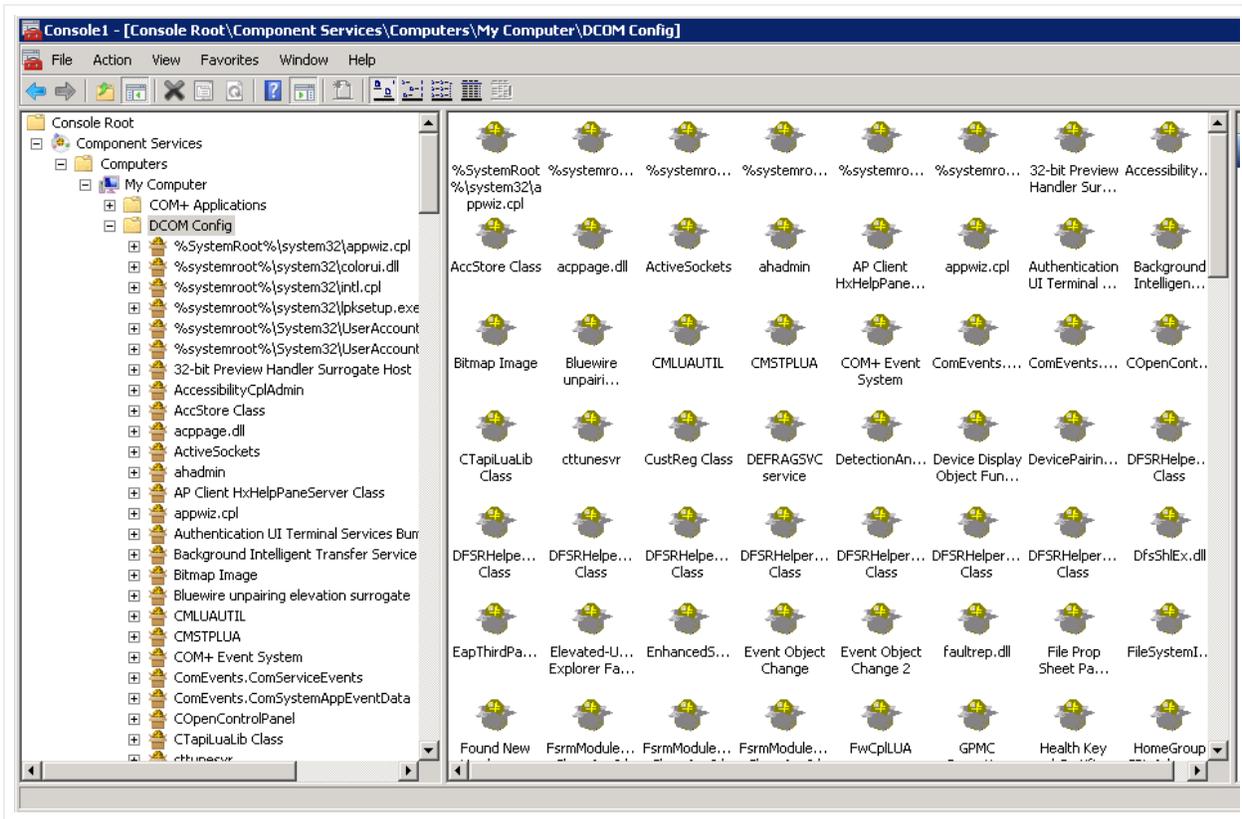
6) Right click on "Microsoft Visio Drawing" and choose properties.

7) Click on the Security Tab and under "Launch and Activation Permissions" click "Customize", and then click "Edit".
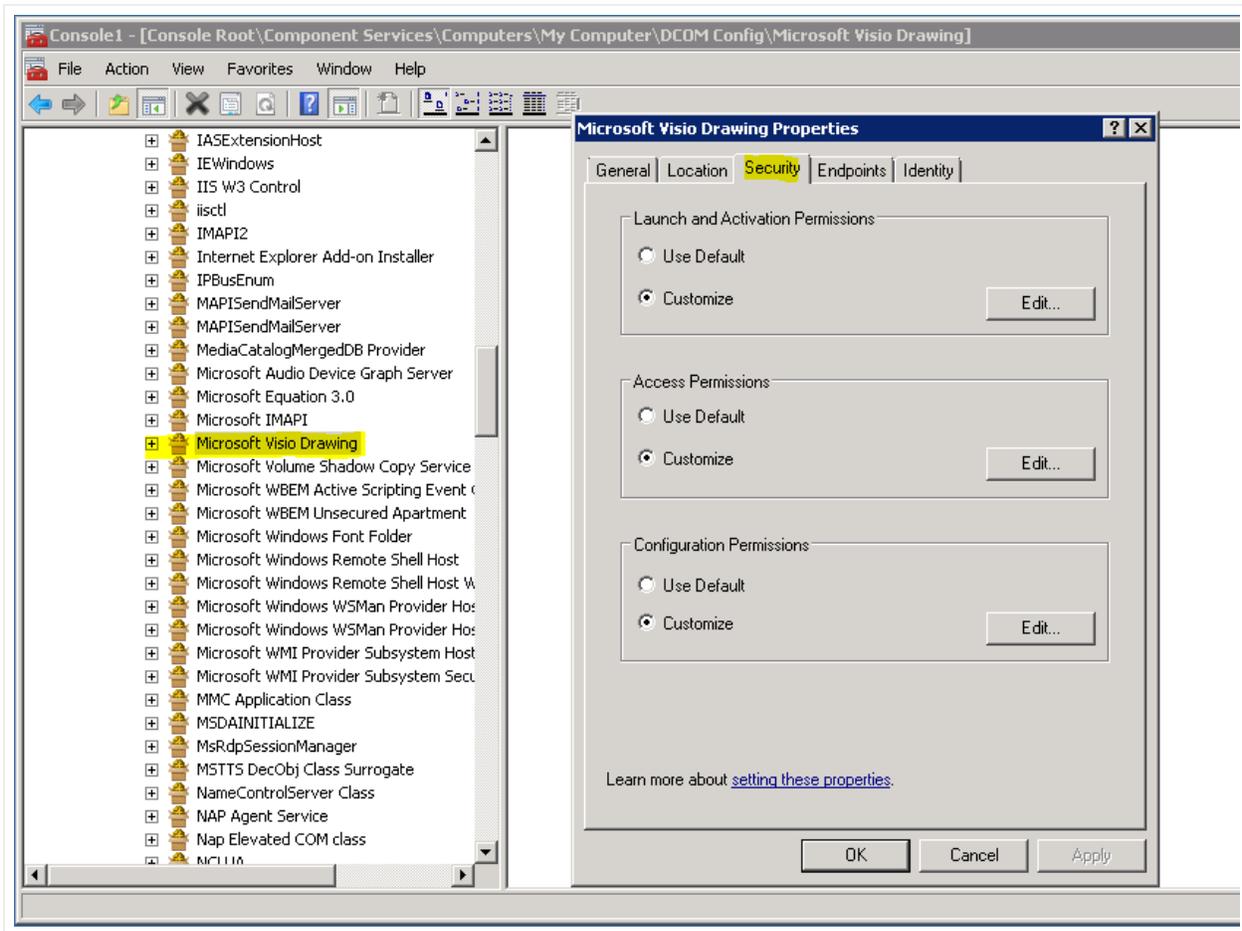
8) Add the user "IIS APPPOOL[AppPoolName]". This account will not be listed or be searchable. You must type it in as follows:

IIS AppPool\ASP.NET v4.5.

> **Attention!**
>
> Make sure you type in the appropriate application pool name. For instance, if you setup an application pool called "netTerrain" then you must use this application pool name and not the default application (such as ASP.NET v4.5).
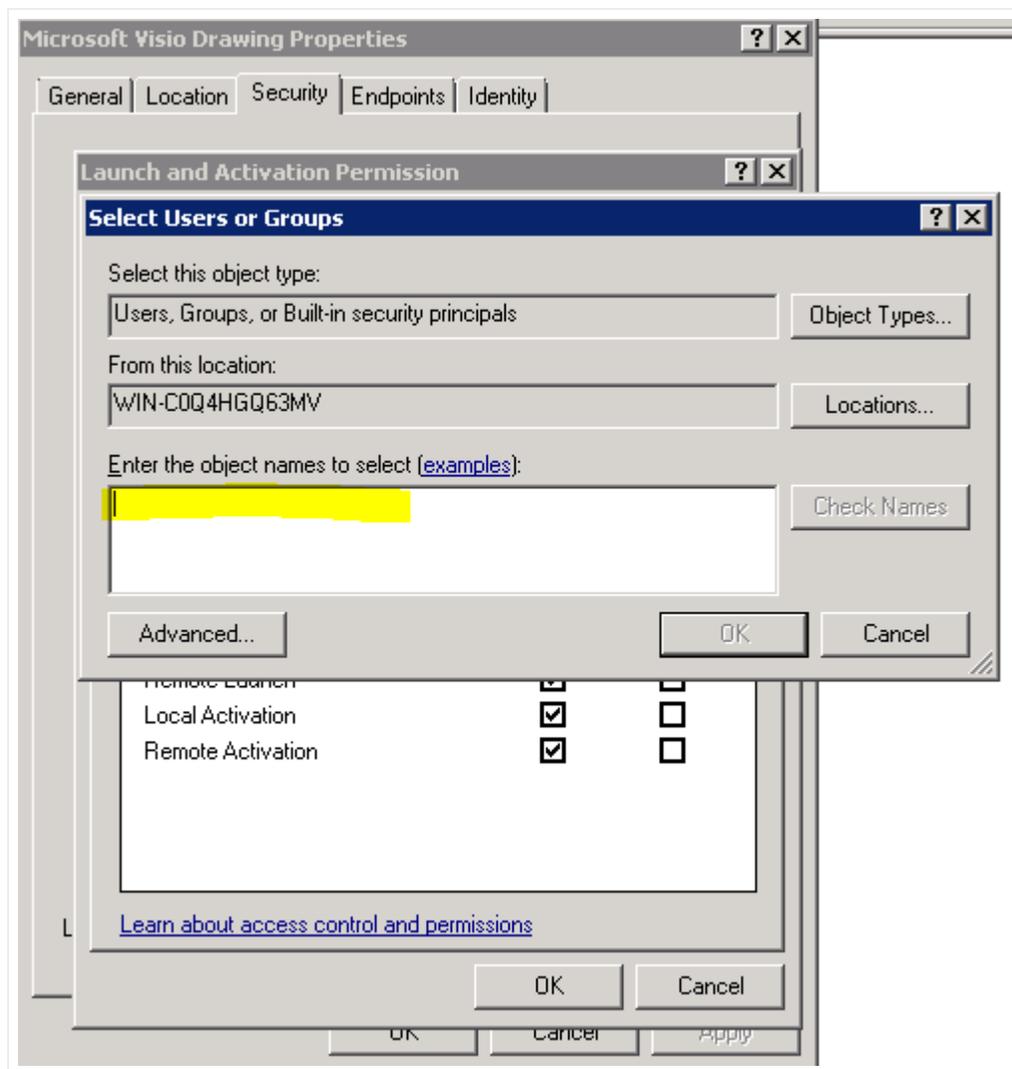
9) Press 'OK'. If it was typed in correctly, the name will be accepted, and the permissions can be configured. Additional permissions may be needed, but this could depend on your environment. If needed, enable all permissions.
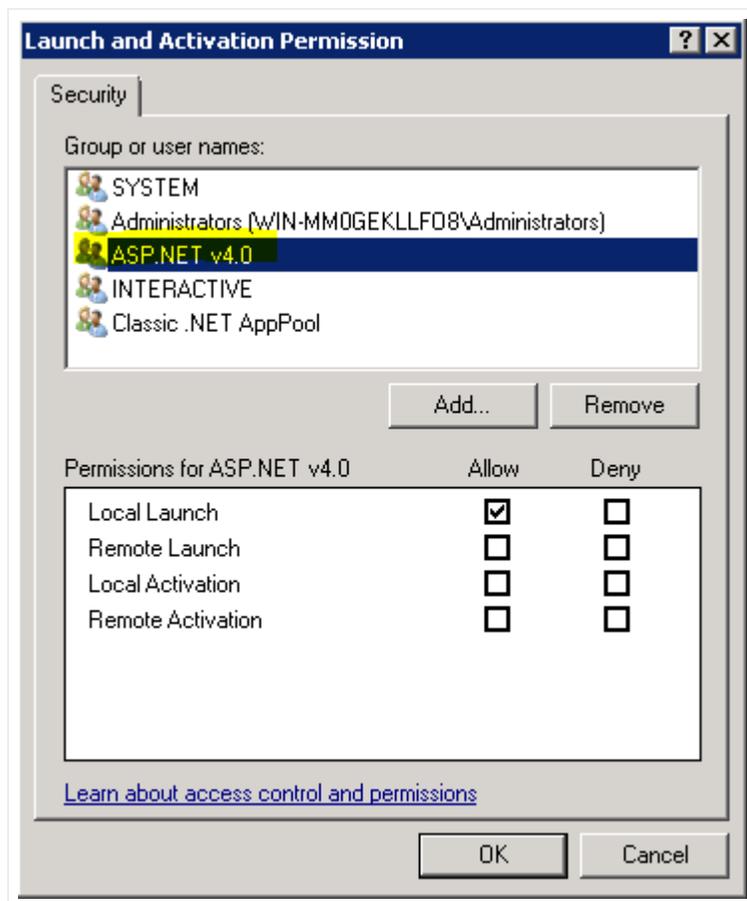
10) Follow the same steps for "Access Permissions".
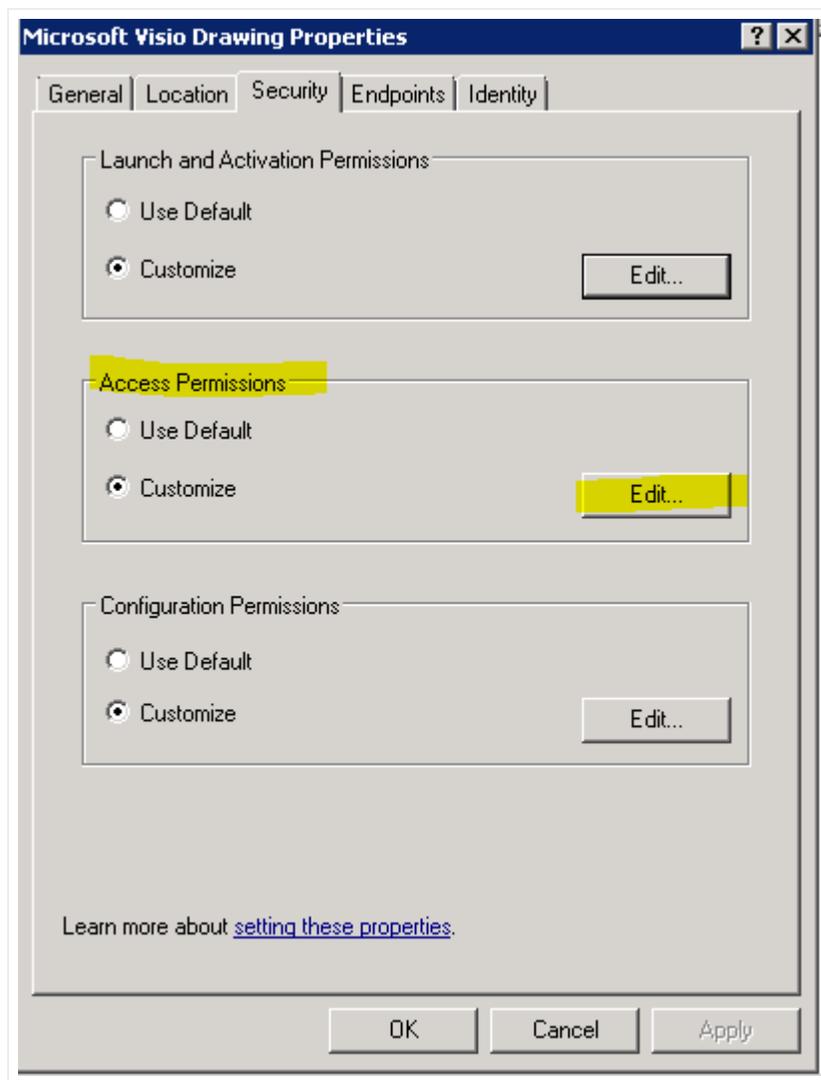
11) Select the Identity Tab and then check "The interactive user". Then press OK. You can now shut down the console.
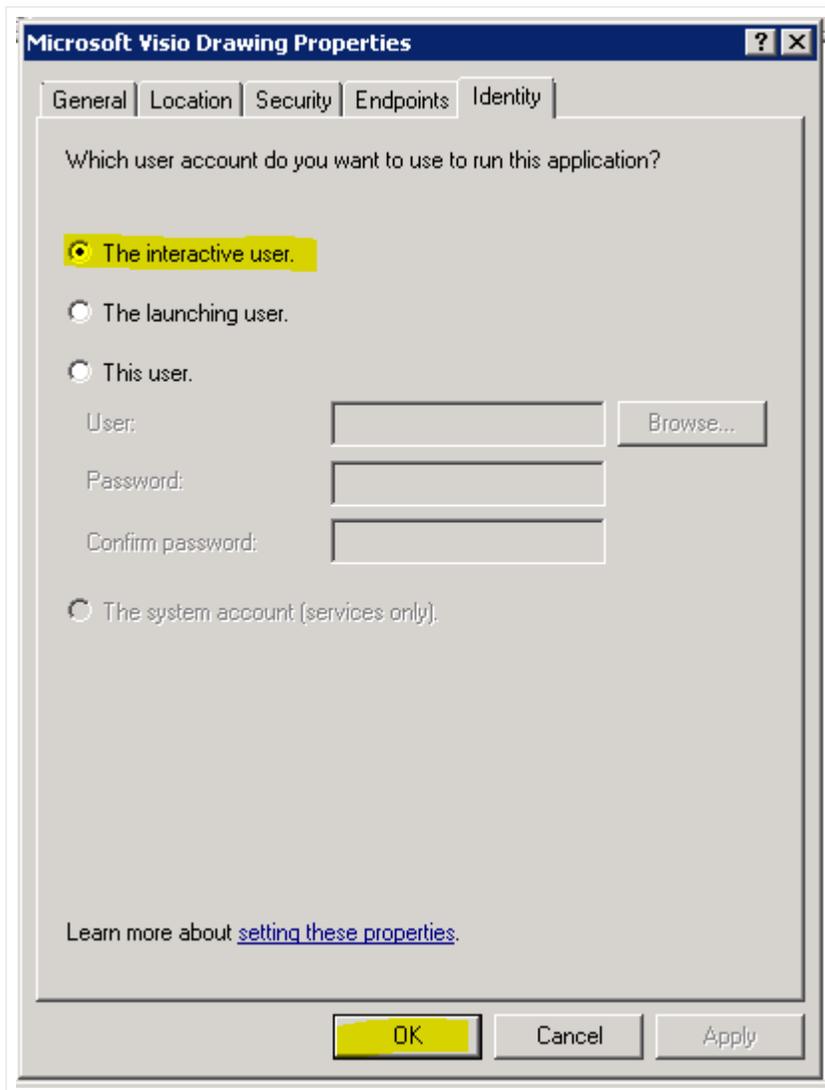
## 2.3.2 Other settings and components

The following settings and components should be noted:

1) Log files: In case of an import problem, you can review the Visio import log files from the "uploads" folder.

2) Large files: To handle large Visio files (greater than 10MB) you need to enable this property in IIS and update the web.config file section shown below:

```
<system.webServer>
    <security>
  <requestFiltering>
  <requestLimits maxAllowedContentLength="524288000"/></
```

```
requestFiltering>
  </security>
</system.webServer>
```

### 2.3.3 Using the Visio importer

Once the Visio importer has been configured you can import a project using the netTerrain web interface. After logging in you should see a Visio icon in the Import ribbon.
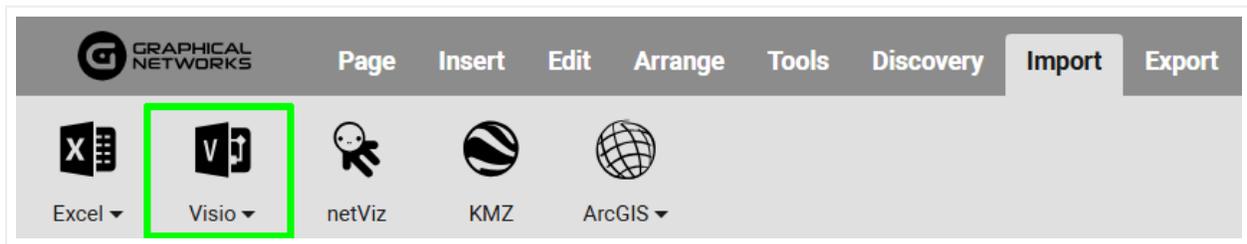


*Visio import button*

There are two ways to import a Visio file: as a static searchable image or as a full project.

Static Visio imports simply convert the contents of each Visio sheet into a static image (no selectable nodes and links). As a bonus, all the attributes for each Visio object are indexed in netTerrain so that you can search for them. However, netTerrain will not pinpoint the object that matches a search, since it is simply part of a static image, but at least it will take you to the image that contains it.

Full project imports are a lot more elaborate since they do convert the Visio objects into netTerrain nodes and links.

To start an import simply click the on the corresponding Visio import option and you should get a popup prompting you to upload a drawing.
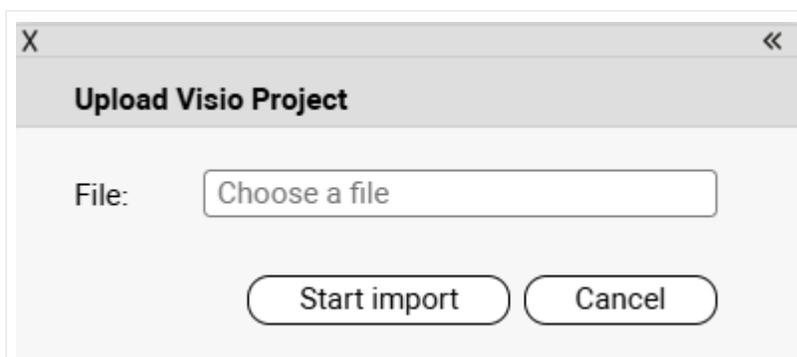


*Uploading a Visio file*

The drawings will be uploaded into the diagram you were in when you pressed the import button, under a top-level Visio icon that bears the name of the Visio project.

Because Visio drawings will not generate smart devices or racks in netTerrain (that concept does not exist in Visio), it is probably better to upload Visio diagrams onto a netTerrain diagram that holds (or will hold) logical data. Better yet, don't use Visio, it's not good for your health.

netTerrain currently only supports .vsd Visio file formats. If the file is larger than 10MB, then IIS may conspire on you and time out. Make sure you configure the settings in the web.config file to allow for large file uploads.

### 2.3.3.1 Using Visio stencils in the netTerrain catalog

Users can also import Visio stencils from netTerrain directly. After logging into the project, you should see a Visio icon under the Import ribbon. Simply click the icon and you should get a popup to upload a stencil file in .vss format. After the upload completes the stencils will be turned into netTerrain objects in the catalog nodes section. Any fields associated with the stencils will also be converted into netTerrain fields.

### 2.4 Static exports: HTML and PNG

Sometimes you may need to create a static output for a netTerrain diagram that does not require any specific application to run. For that purpose, we introduced HTML and PNG exports as a mechanism to one or multi diagrams to a format that can be opened with a browser or an image viewer. An additional advantage of these methods is that it does not require any specific set up on the netTerrain server.

### 2.4.1 Exporting to static HTML

netTerrain diagrams are already in HTML5 format, so why this export option? Native netTerrain diagrams may be HTML, but you need to be connected to the netTerrain server to view them. Sometimes you may have a user that needs to look at a diagram without being connected to netTerrain. This is what this export feature is for.

To start the export process to HTML, click on the Static button on the Export toolbar and select the Export as HTML package option.
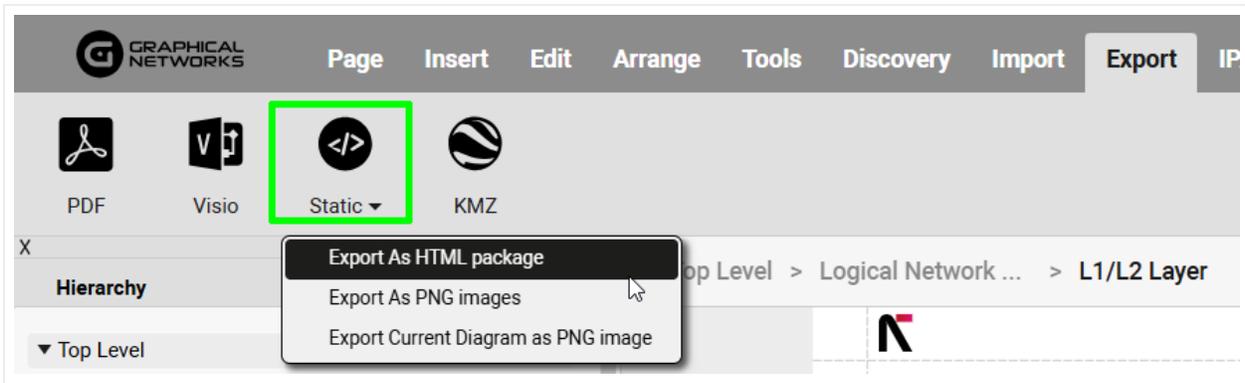
*HTML package export*

## 2.4.1.1 HTML export options

After clicking on the HTML export sub menu, a dialog showing export options will be displayed, as shown below:
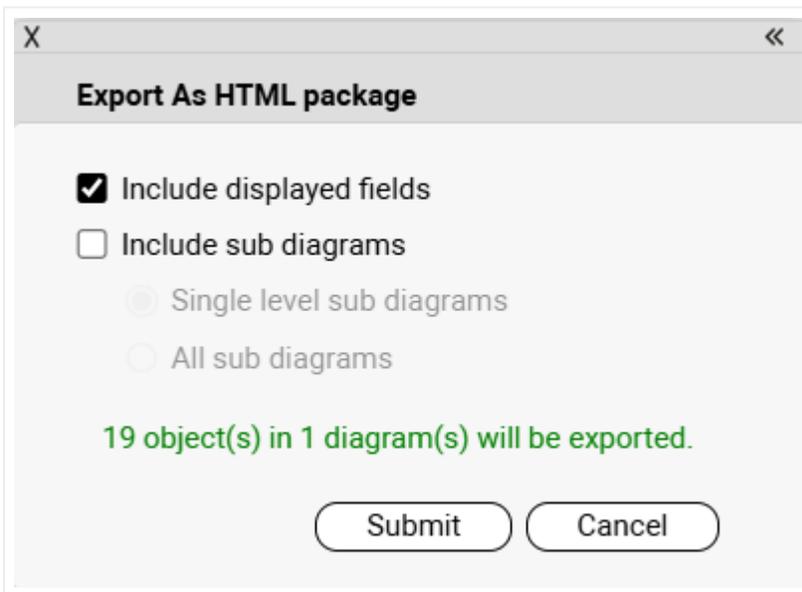


*HTML export options*

The following options are available when exporting to an HTML package:

- Include displayed fields, which means that any displayed data in the diagram will also be in the exported HTML diagrams.
- Include sub diagrams, which will export child diagrams to individual HTML pages (hence the name HTML package) depending on the sub diagram options that are checked. These are:
- Single level sub diagram, which exports the immediate level below
- All sub diagrams exports, which, well, you guessed it, exports everything

Note that the all sub diagrams option can take a very long time to process and is not recommended for large sets of diagrams. If your boss really must see the entire collection of diagrams, then get ready for a long coffee break.

## 2.4.2 Exporting to PNG

To start the export process to PNG, click on the Static button on the Import & Export toolbar and select one of the PNG export options.
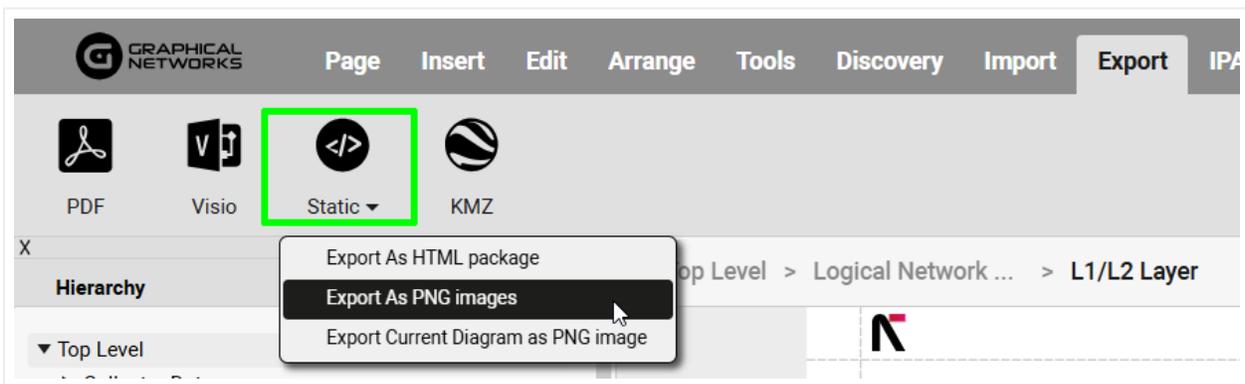


*PNG export*

## 2.4.2.1 PNG export options

The first option of the highlighted sub menus above can export multiple diagrams in PNG format:
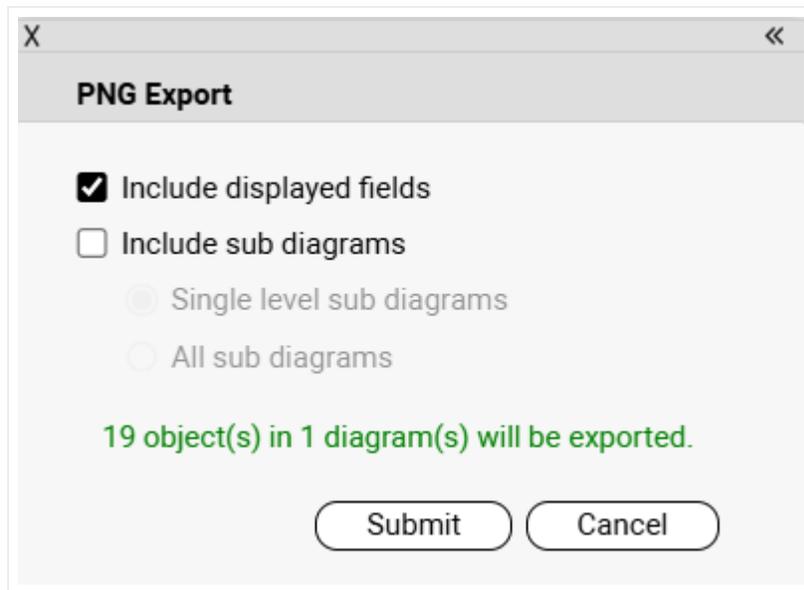
*PNG package export options*

The following options are available when exporting to a PNG package:

- Include displayed fields, which means that any displayed data in the diagram will also be in the exported PNG diagrams.
- Include sub diagrams, which will export child diagrams to individual PNG files. These are:
- Single level sub diagram, which exports the immediate level below
- All sub diagrams exports, which, well, you guessed it, exports everything

The second option of the highlighted sub menus in the Static export button exports the current diagram only.

# 3 netViz Import

This chapter explains the process to set up netViz imports in netTerrain. For the netViz import to work properly, netViz 7.5 needs to be installed on the netTerrain application server.

Before starting the netViz import set up, make sure you place the file called nvbashlp.dll (usually located in C:\Program Files (x86)\Graphical Networks\netTerrain5) in the netViz program folder:

```
C:\Program Files (x86)\netViz\7\Author\Program.
```

## 3.1 Modes of operation

The netViz importer can run in two modes: manual or automatic. No, manual does not mean that you must manually create everything in netTerrain again; it simply means that netTerrain will not automatically fetch the data from the netViz .net file; instead you must prepare the import data beforehand.

The manual import involves:

1) Opening the netViz project from the netViz application.

2) Running a netViz basic script from the embedded basic editor inside the netViz program.

3) Taking the xml output generated by the script for a netTerrain import.

This process is easier than it sounds, and we recommend following it if you only have to import a few projects. The automatic import process is easier for end users but setting it up could be somewhat tricky. As a simple rule of thumb, if you have more than 10 projects, try to set up netViz on the app server to run in automatic mode, otherwise import netViz projects in manual mode and skip the next section.

## 3.2 Setting up the automatic import

If you want to run the import in automatic mode, then you need to set up certain parameters on the netTerrain application server. Why is that? Because netTerrain is a web-based client server application and Microsoft security (for very good and obvious reasons) is quite iffy about letting a web app in IIS take control of a desktop executable.

The convenience of automatic importing lies in the fact that end users can import netViz projects directly from the netTerrain browser by uploading them with the click of a button. The automatic import is convenient when you have many netViz projects to import (e.g.: more than 10).

Note that for netViz projects larger than 30MB you may need to change your IIS settings for the automatic import to work, or you may have to use the manual mode.

Follow these steps to configure the netViz import in netTerrain:

1) On the application server run the 32-bit version of the Microsoft MMC to change the netViz settings.
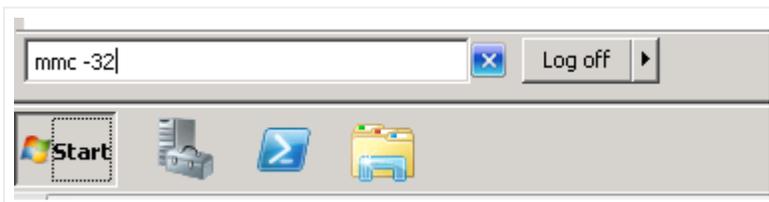
2) Make sure the MMC console starts up.
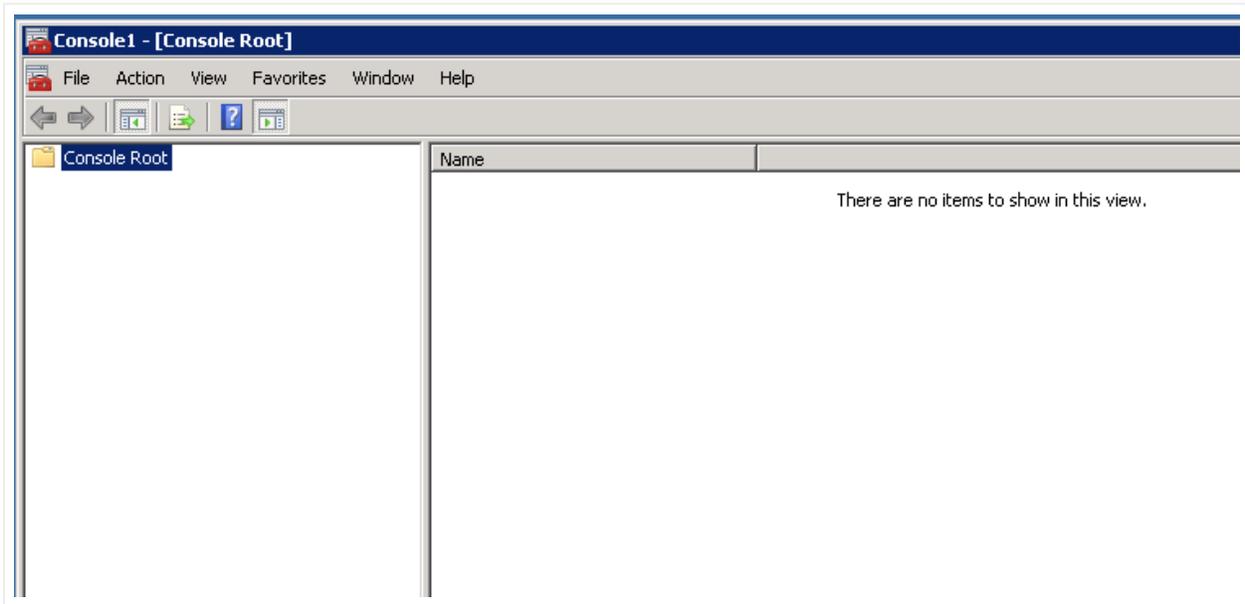


3) Select Add/Remove Snap In.
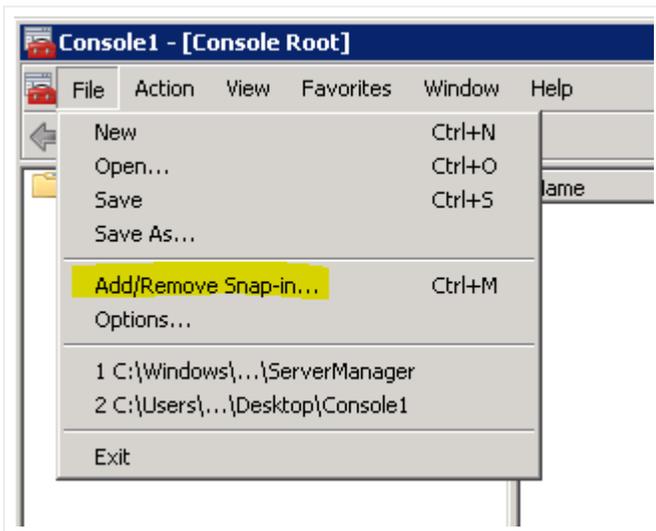


4) Select Component services, press "Add" and then press OK.
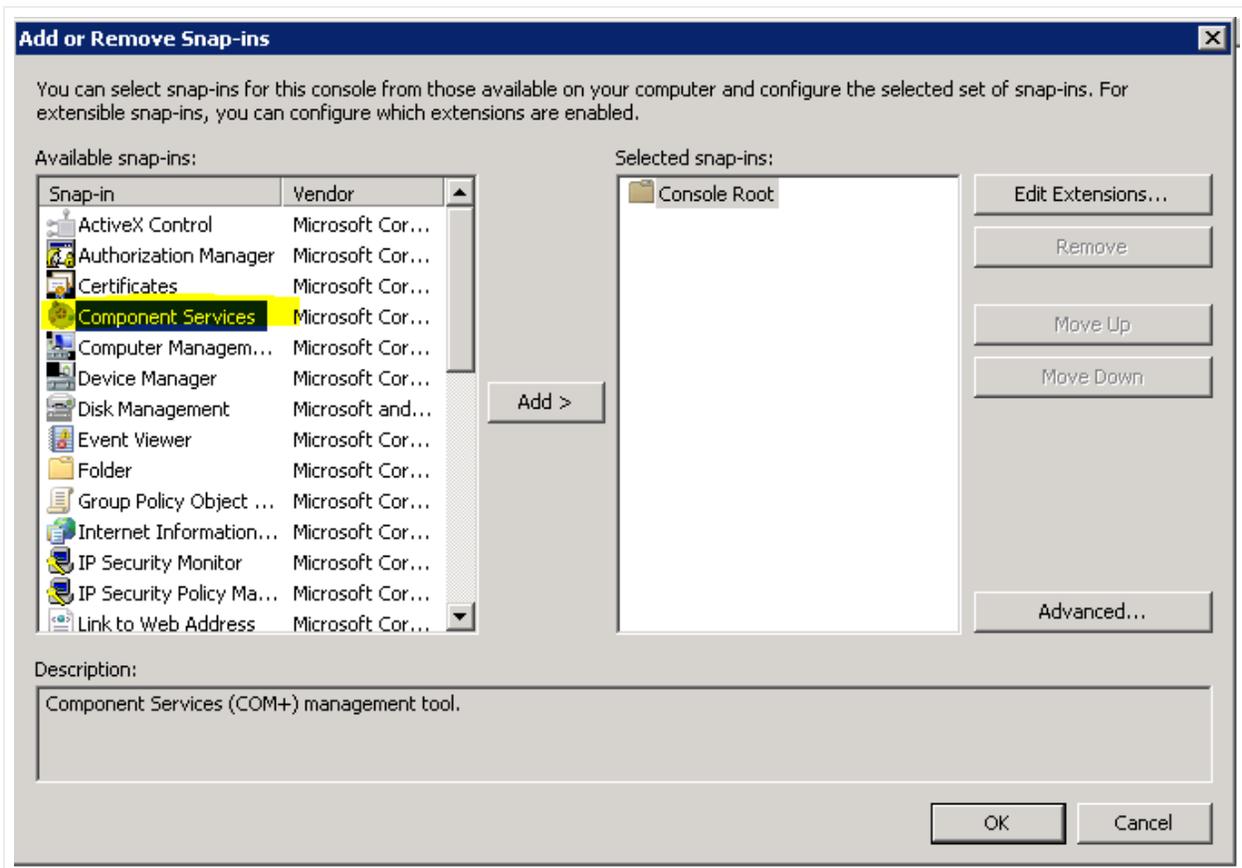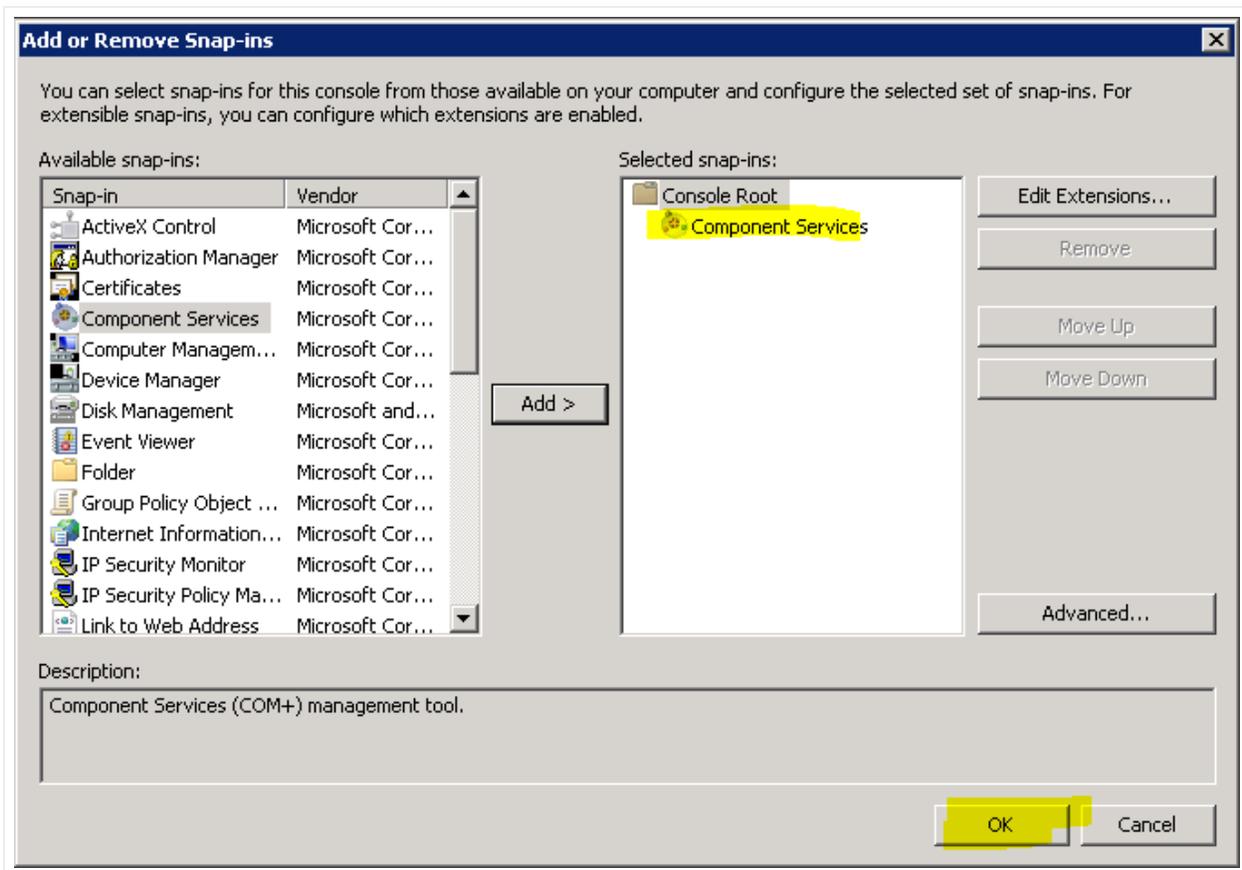
5) Open "DCOM config" and then scroll down to "netViz".
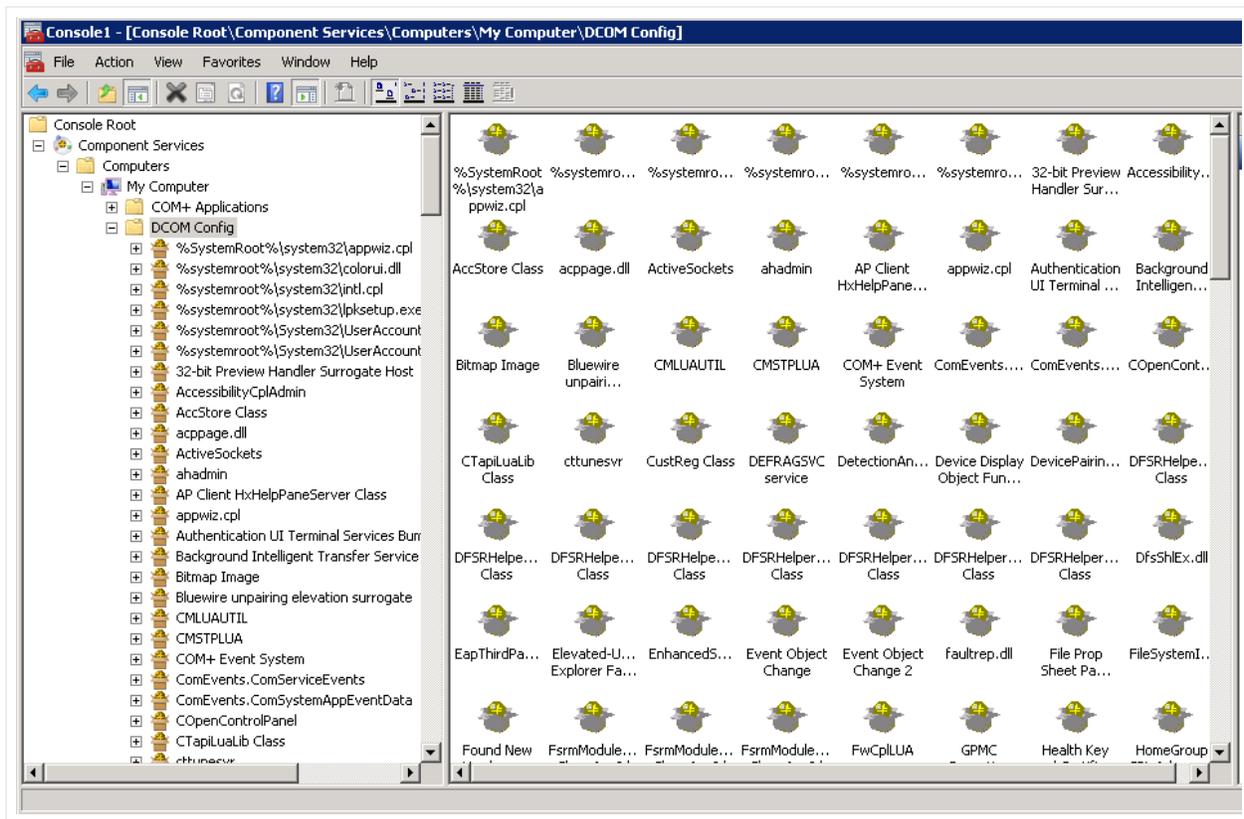
6) Right click on "netViz" and choose properties.

7) Click on the Security Tab: Under "Launch and Activation Permissions", click "Customize", and then click "Edit".
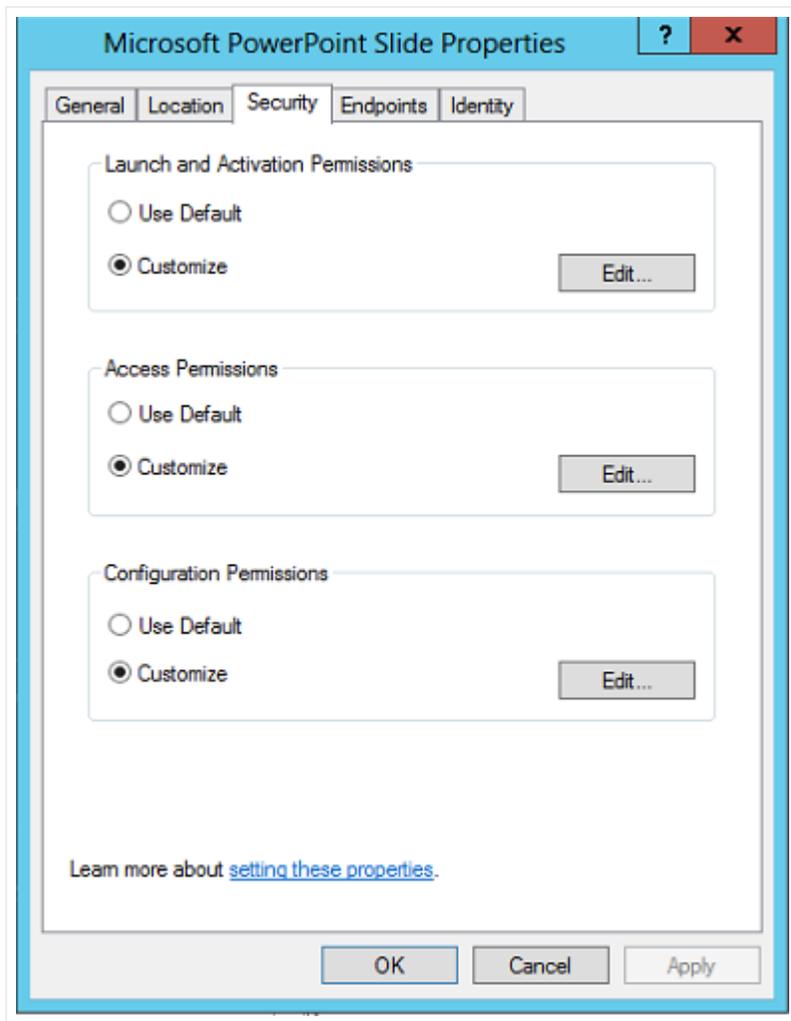
8) Add the user "IIS APPPOOL[AppPoolName]". This account will not be listed or searchable. You must type it in as follows:

```
IIS AppPool\ASP.NET v4.0 (or 4.5).
```

Attention!

Make sure you type in the appropriate application pool name. For instance, if you setup an application pool called "netTerrain" then you must use this application pool name and not the default application pool for instance ASP.NET v4.0.
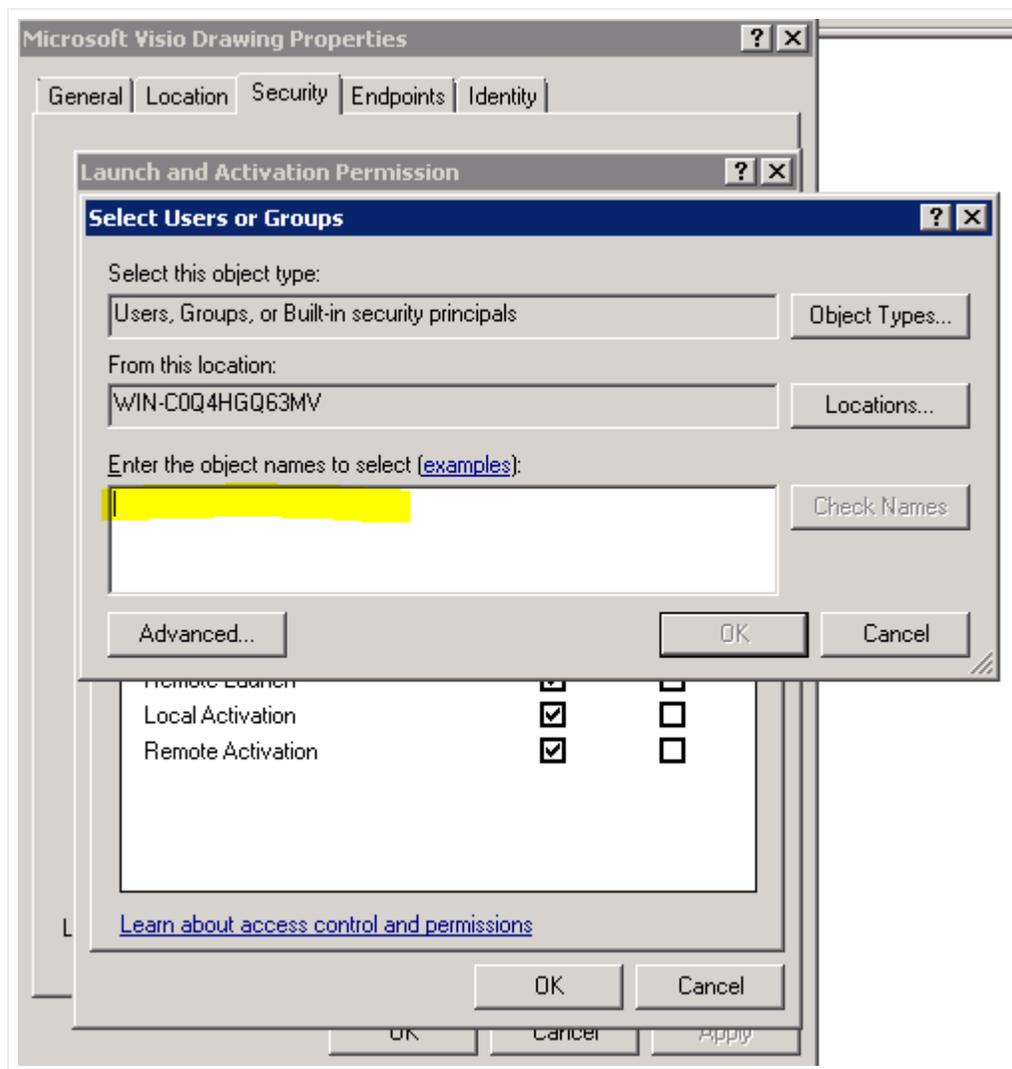
9) Press OK. If the user was typed in correctly the name will be accepted and the permissions can be configured. Additional permissions may be needed, which depends on your environment. If needed, enable all features.

10) Follow the same steps for the "Access Permissions" part.
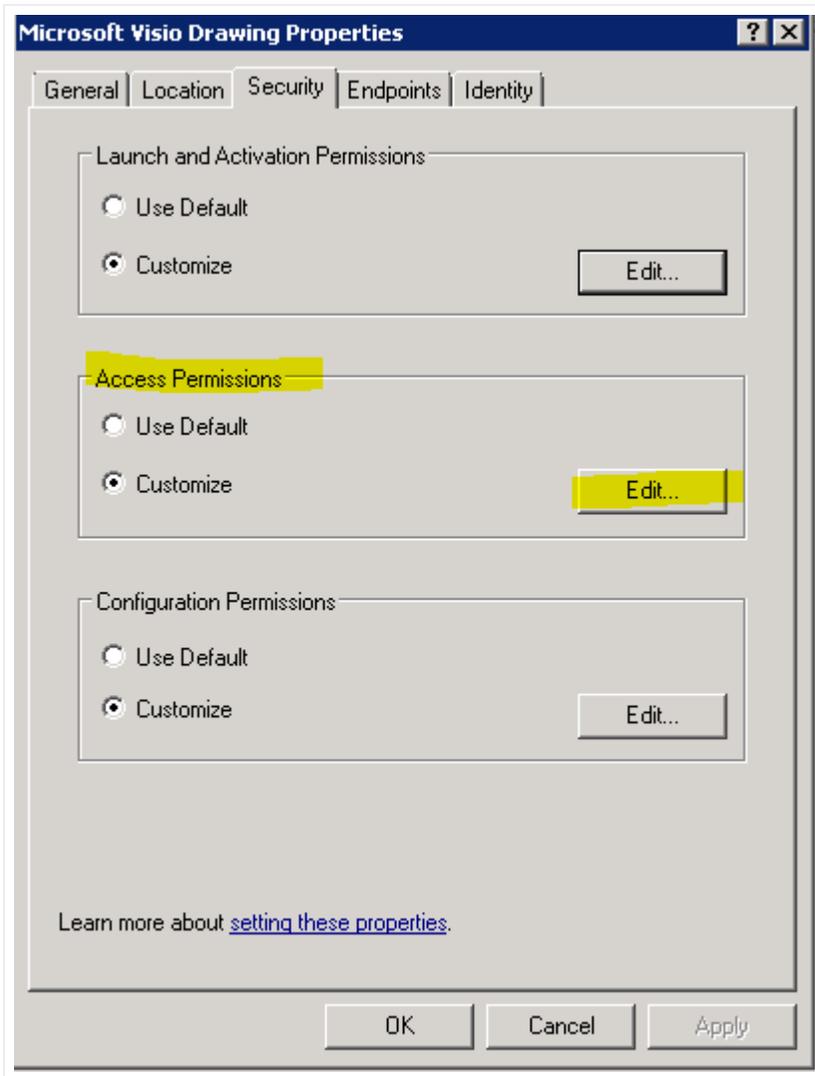
11) Select the Identity Tab and then check "The interactive user". After pressing OK, you can close the console.
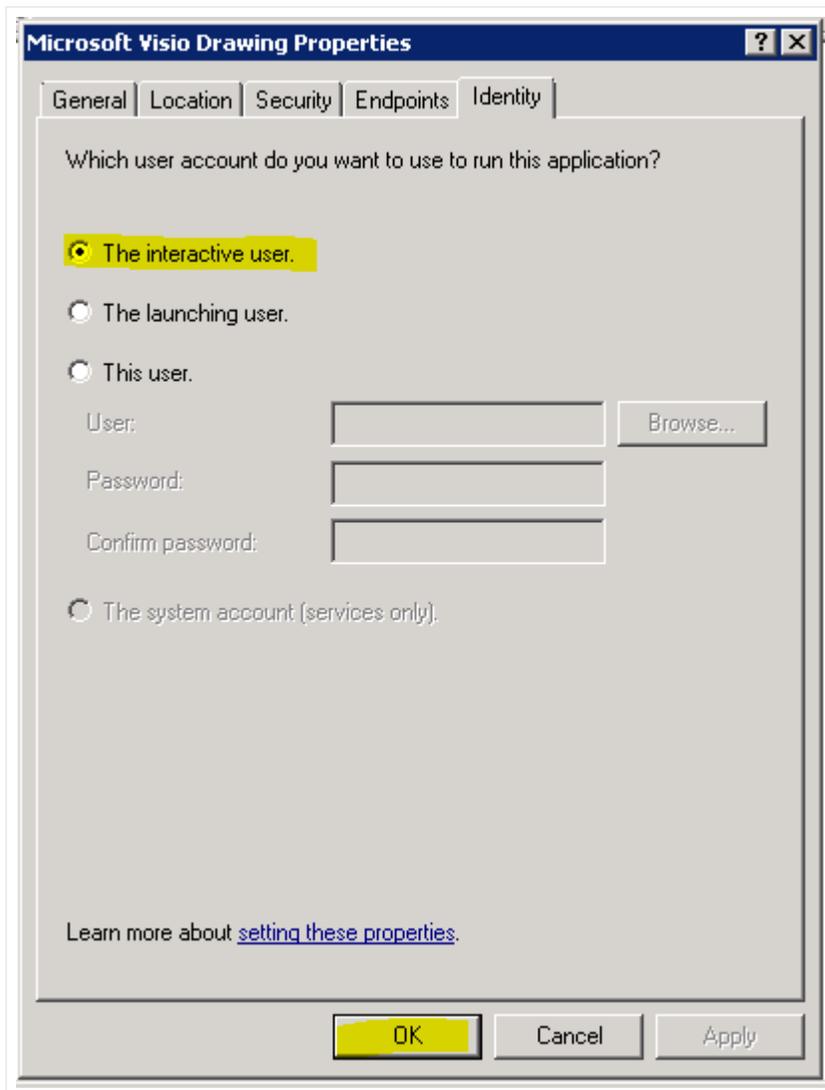
Next, we will review other settings and components to review before importing your netViz data.

## 3.3 Other settings and components

The following settings and components should be noted:

1) Log files: In case of an import problem, you can review the netViz import log files from the "uploads" folder.

2) Large files: To handle large netViz files (greater than 10MB) you need to enable this property in IIS and update the web.config file section:

```
<system.webServer>
  <security>
  <requestFiltering>
 <requestLimits maxAllowedContentLength="524288000"/>
  </requestFiltering>
  </security>
</system.webServer>
```

## 3.4 Pre-import checklist

Before you start the netViz importer, review this checklist for an optimal process:

1) **Back up** your netViz project.

2) **Print templates:** remove print templates and composite views (we cannot import these due to netViz export limitations).

3) **Unused types:** delete any catalog types that are not used (unless you want to import the unused catalog objects). To do this in netViz:

a) Go to file/catalog/delete types

b) Then choose the option to delete all unused. Do this for node and link types.

4) **Repeated type names:** combine any nodes or links that have the same name or change the name of these nodes (if you do not change redundant names then some of the nodes and links will be dropped). You can use replace type in netViz, if needed.

5) **Field naming:** hit F3 to edit the internal catalog to check all node and link fields for repeated names. All field names must be unique for each node and link type. Also, if any node or link field is called "name", "id" or "type", please rename them since these are netTerrain reserved field names.

6) **netViz "blue" field:** the field that has the unique name information should be moved to the first position for the types in the catalog. This is usually the blue or green field in netViz. This field will then be mapped with the name field in netTerrain.

7) **Run a save and compact** on the project.

8) **Import the netViz project** as described below

## 3.5 Using the netViz importer manual process

Follow these steps to run the netViz import process in manual mode:

1) Open the basic script called "NVEXPORTscript.bas" located in the C:\Program Files (x86)\Graphical Networks\netTerrain\Utilities folder using notepad and change the output folder path value.
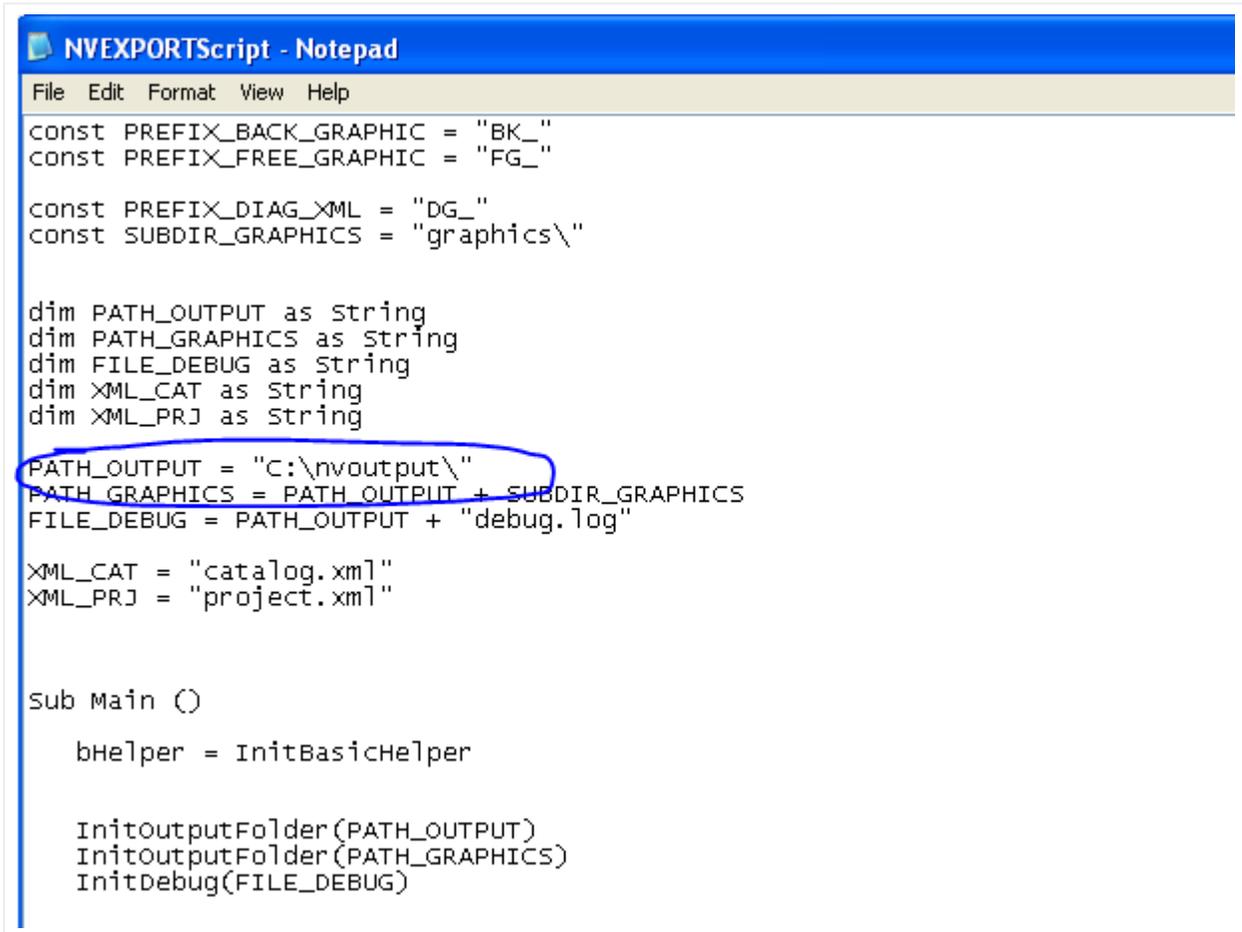


2) Create the output path folder in this instance called C:\nvoutput.

3) Start netViz and open the project that will be converted.

4) Open the Basic Editor feature in netViz.
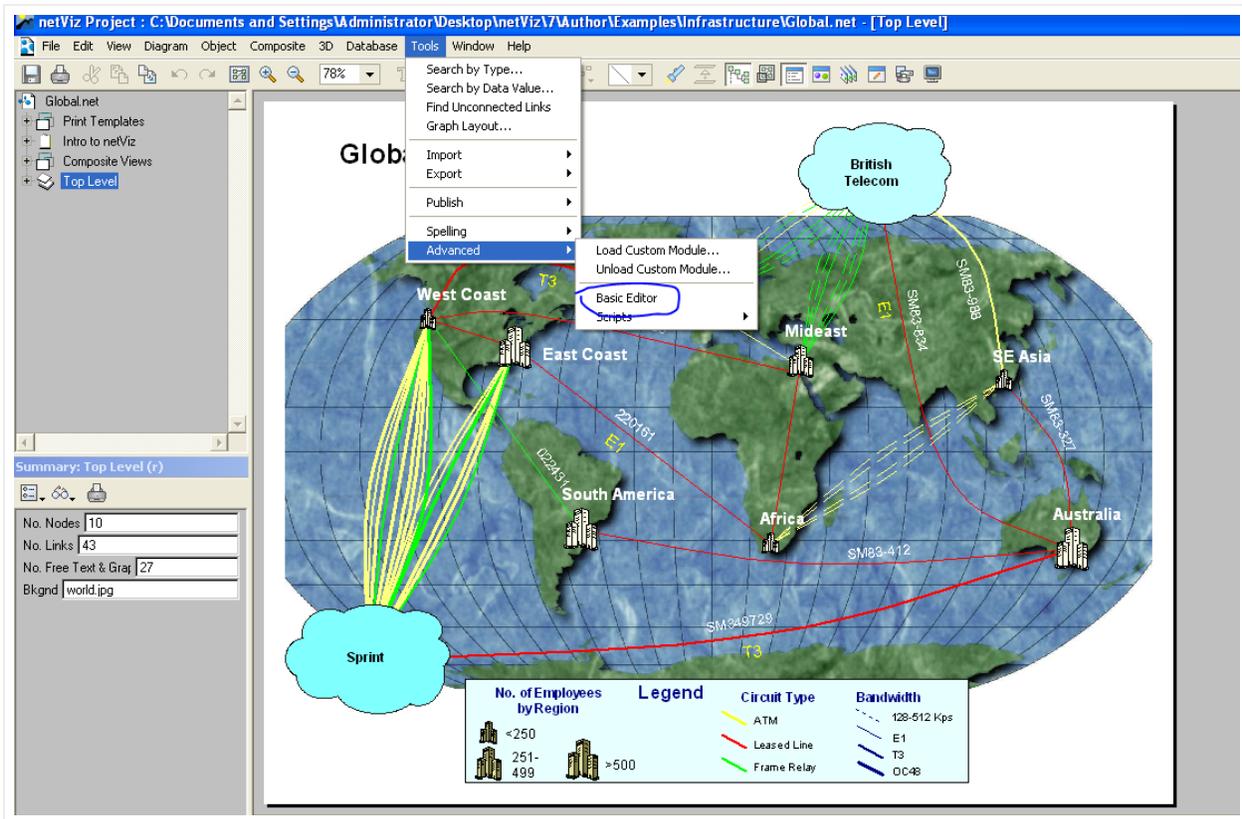
5) Load the "NVEXPORTscript.bas" module.
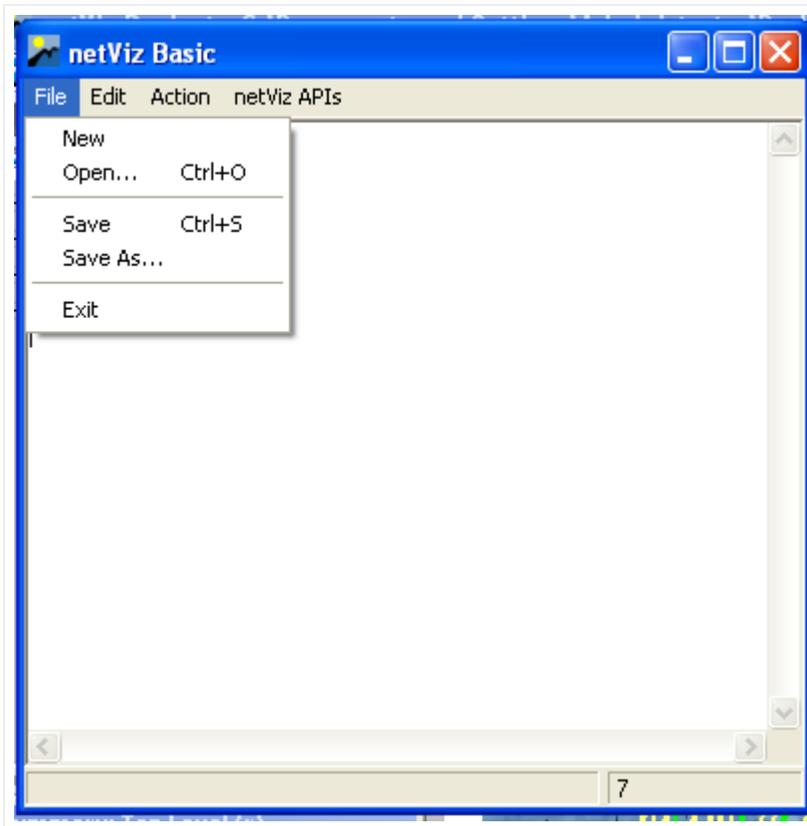
6) Press 'run'.
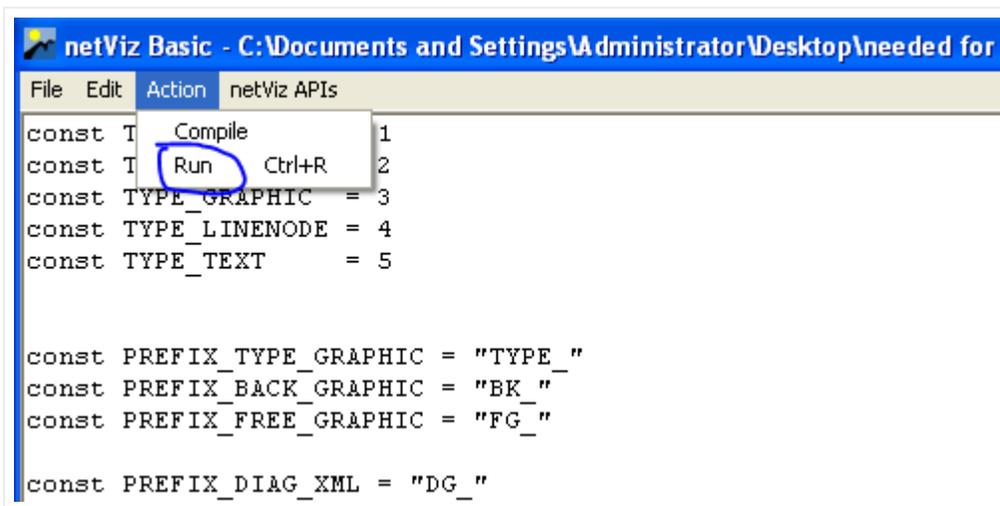


7) When it is completed you should see this message:

8) ZIP the folder called "nvoutput" (the folder you created on the C drive).

### Zip this folder

| Name | Date modified | Type | Size |
|---|---|---|---|
| inetpub | 11/10/2013 7:05 PM | File folder | |
| ☑ nvoutput | 1/13/2014 8:44 AM | File folder | |
| Program Files | 1/7/2014 7:03 PM | File folder | |
| Program Files (x86) | 1/10/2014 4:03 PM | File folder | |
| temp | 1/13/2014 8:43 AM | File folder | |
| Users | 11/12/2013 9:32 AM | File folder | |
| Windows | 1/10/2014 4:57 PM | File folder | |
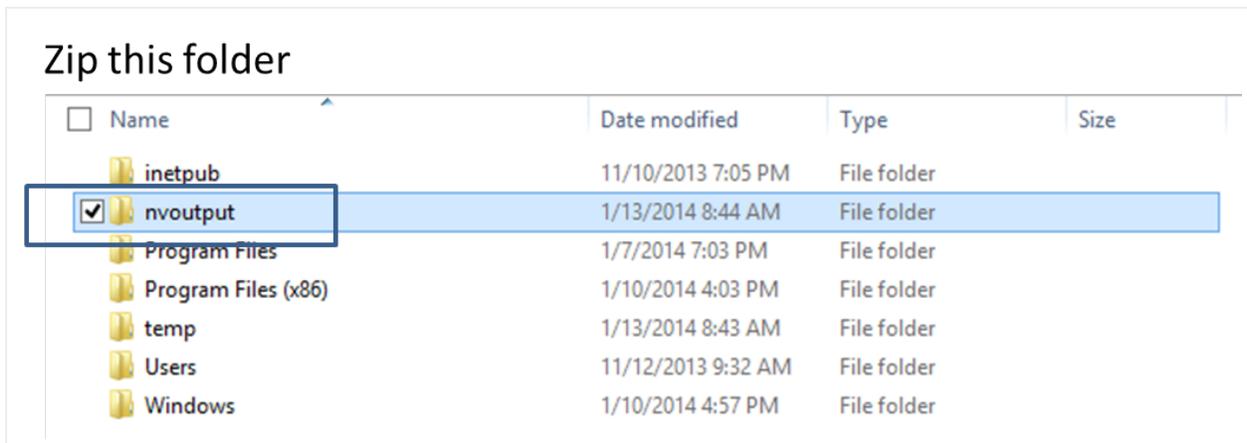
9) Log into the netTerrain server and press the netViz import button. Upload the ZIP file.
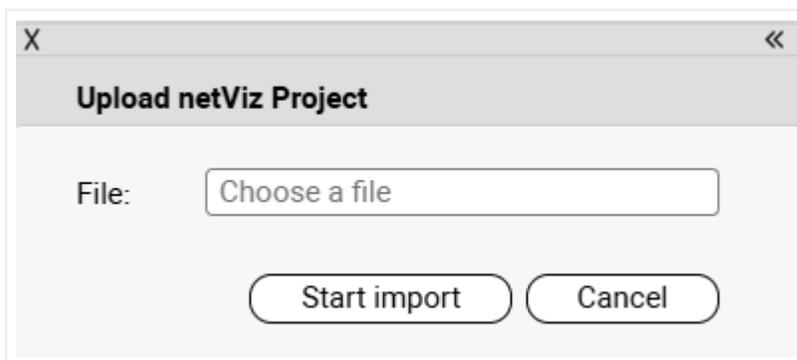


The import process will begin.
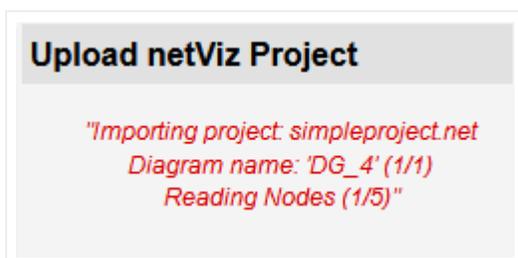
10) After the process completes you will be able to see the imported project.

This process takes about half a second per instance. If you want to check the number of objects in netViz before importing them, open netViz and select Summary->brief from the File menu.

You should also check the netviz.log file for errors (usually located in C:\ProgramData\Graphical Networks\netTerrain\vis\App_Data).

After the import is completed, a green generic node will be placed on the diagram where the import was launched. The node will have the name of the netViz project and the entire contents of the project will be placed underneath that node.

## 3.6 Using the netViz importer automated process

Log into netTerrain and press the "netViz import" button in the Import & Export ribbon. Upload a netViz project and wait for the project import to complete. Yeah, that's it. The tricky part is to get IIS to work in concert with netViz access permissions, as described earlier in the chapter.
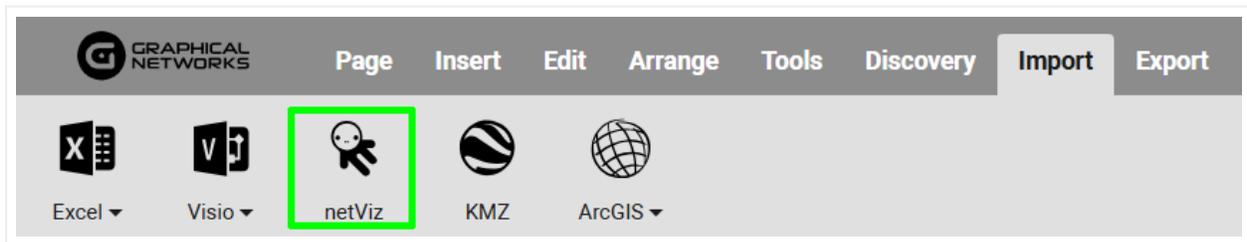


*netViz import button*

After the import is completed, a green generic node will be placed on the diagram where the import was launched. The node will have the name of the netViz project and the entire contents of the project will be placed underneath that node.

## 3.7 Bulk importing netViz projects

When needing to import numerous netViz projects, using the import button from the Import & Export ribbon may be inefficient and tedious. netTerrain allows for bulk imports of netViz projects but requires server file access to the webserver to upload netViz projects.

1) Ensure the following setting is present in the web.config file above "

```
<!-- NetViz import queue.-->

<add key="NetVizImportQueuePath" value="App_Data/netvizque"></add>
```

2) Add the netViz projects to the folder (or the custom installation folder):

```
_C:\ProgramData\Graphical Networks\netTerrain\vis\App_Data\netvizque_
```

3) From the netTerrain project click on + + " to access the netViz queue. Make sure the number of netViz projects matches the number of projects uploaded. Click 'Import' to begin the process."

4) At the end of the import process, logs will be generated in the 'App_Data' folder. We recommend that you at least review the '[time-date] NV QUEUE Import' log for any errors that may have occurred during import. There are also logs for each netViz project in the 'App_Data' folder.

> Attention!
>
> The value App_Data/netvizque refers to a folder path. Make sure the folder netvizque exists in the App_Data folder.

## 3.8 Post netViz import checking

It is recommended to perform a basic sanity check in netTerrain to check that the netViz projects have been imported in a satisfactory manner. This includes comparing the catalogs, the projects and other miscellaneous checks.

## 3.8.1 Comparing catalogs

Perform a catalog node and link comparison between the netViz catalog and the imported netTerrain catalog.

One way to do this is to print the catalog from netViz (files/summary/catalog/print). Use this printout to check off each field and verify all fields have been imported. Make sure all visual overrides are available and check the drop-down lists. Currently, drop down lists need to be recreated in netTerrain. If any fields are missing, it is usually because the fields in netViz are repeated.

## 3.8.2 Comparing projects

As a second step, you can check the node counters for both netViz and netTerrain. In netTerrain, the counter for nodes will be displayed in the diagram properties of the netViz import node sub diagram.

*netTerrain node counter*

The netViz counter can be obtained by opening the netViz project and then going to Project->Summary. Compare both counters and if they are close enough then the imports were successful. It is not uncommon to miss some nodes and links. This is usually due to extra system nodes added in netTerrain and/or dangling links not being imported.

## 3.8.3 Other checks

You can also:

1) Recreate any node and link field drop down lists that were not imported. Drop downs are not imported unless they are visual overrides in netViz.

2) Check visual overrides, to be sure they look correct.

3) Check link settings to be sure they have the appropriate color.

4) Check the nodes to see if any images need to be replaced.

5) Check background images for diagrams to see if any need to be replaced.

6) Clean up font sizes and positions for nodes and links. "Shape" object displayed fields should be set to the "center".

## 3.9 Importing a netViz catalog

It is possible to just import a netViz catalog into netTerrain without importing any data:

1) Log into netTerrain.

2) Press the "netViz import" button under the Import & Export ribbon.

3) Create an empty netViz project that has a catalog collection that you want imported into the netTerrain nodes catalog.

4) Upload the netViz catalog project and wait for the project to be processed.

The node and link types from the netViz project should be imported as types in the netTerrain catalog.

# 4 Excel Import

This chapter covers the process and techniques required for importing devices, nodes and other objects into the netTerrain project using the Excel import spreadsheet.

The spreadsheet can be downloaded from netTerrain by clicking on the 'Excel Import Sheet' menu located in the Tools section, as explained later.

> **Attention!**
>
> It is important that you maintain the existing format of the ExcelImportSheet.xls file and preserve its compatibility level; otherwise the Excel importer may not be able to comprehend its contents and will result in errors.

Also, in order for the import to work, it must be enabled from the Admin console in the Global Settings section.

## 4.1 Excel import worksheets

The Excel import spreadsheet contains three worksheets:

- Objects: creates new node or updates existing ones in the netTerrain project. Devices and cards are also allowed.
- Links: creates new links in the netTerrain project
- Subnets: creates subnets in bulk for the IPAM functionality

Through the nodes worksheet, devices, cards and racks can also be imported.

> **Attention!**
>
> You cannot generate catalog devices using the Excel spreadsheet. However, we support a large catalog of pre-modeled devices. If there are devices you would like to see modeled, please contact us at support@graphicalnetworks.com. As part of the maintenance we can model any types for you. Of course, our training and certifications also teach you how to do this in our device type modeler.

## 4.1.1 Obtaining the Excel import sheet

As mentioned above, the Excel import sheet can be downloaded directly from netTerrain if the administrator has enabled downloads from the netTerrain page.

Simply log into netTerrain, go to the Tools ribbon, and click on the Tools menu option. If the Excel import sheet download has been enabled by the admin then you should be able to download the template excel import sheet directly.

If the administrator disabled the direct download of the Excel template, you must request from the administrator directly.

If you are the administrator, the spreadsheet can be found in the web installation directory of netTerrain. The default directory is: `C:\ProgramData\Graphical Networks\netTerrain\vis` . There should be a BulkImport.xls file in the base folder. This file must be used for the Excel import and should be copied and then modified. The file may be renamed.
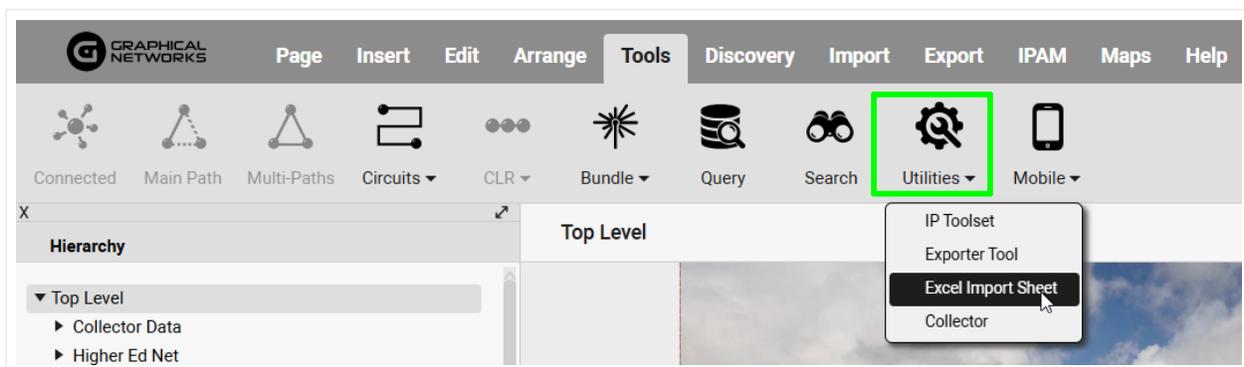


*Excel import sheet download*

## 4.1.2 Objects

This worksheet is used for creating and updating nodes, devices and cards in the netTerrain project.

| | A | B | C | |
|---|---|---|---|---|
| 1 | **Name** | **Type** | **ParentNode** | |
| 2 | Alpha | Building | Excel Import | {Latitude}4 |
| 3 | 1st | Floor | Alpha | {Comment: |
| 4 | 101 | Room | Alpha>1st | |
| 5 | R1 | 42d | Alpha>1st>101 | |
| 6 | PP1 | Corning CCH-04U (empty) | Alpha>1st>101>R1>Back{20} | |
| 7 | A | Generic Vertical 12 port card (LC Ports) | Alpha>1st>101>R1>PP1>A | |
| 8 | B | Generic Vertical 12 port card (LC Ports) | Alpha>1st>101>R1>PP1>B | |
| 9 | C | Generic Vertical 12 port card (LC Ports) | Alpha>1st>101>R1>PP1>C | |
| 10 | R10987655 | Juniper MX80-48T-AC | Alpha>1st>101>R1>Front{10} | {IP_Addres |
| 11 | Bentley | Building | Excel Import | {Latitude}3 |
| 12 | 1st | Floor | Bentley | |
| 13 | 101 | Room | Bentley>1st | |

*Objects worksheet*

The columns include:

- Name: mandatory field that represents the name of the object
- Type: mandatory field that identifies the type associated with the object. Once a type is defined, the custom fields in the spreadsheet will correspond to the custom fields associated with that object type in the netTerrain catalog
- ParentNode: the name of the parent node containing this object. If this field is blank, the new object will be placed on the Top Level diagram.
- Fields: the custom fields associated with the type of the object. The import process does not require you to put specific custom values in specific fields in the spreadsheet, instead, you simply need to add the netTerrain custom field name as a prefix in brackets. In the example below, the nodes Alpha and 1st' have custom fields Latitude, Longitude and Comments which can be populated by specifying the netTerrain custom field names in brackets:

| A | B | C | D | E |
|---|---|---|---|---|
| **Name** | **Type** | **ParentNode** | | |
| Alpha | Building | Excel Import | {Latitude}40.4851 | {Longitude}-78.7247 |
| 1st | Floor | Alpha | {Comments}1500 Sq Feet | |
| 101 | Room | Alpha>1st | | |
| R1 | 42d | Alpha>1st>101 | | |

*Pointing to the proper netTerrain custom fields using brackets*

## 4.1.2.1 Assigning cards to a device slot

Before assigning cards to devices through the bulk import process make sure the card types that you will use have been modeled and mapped to the device type and device's modeled slots in the netTerrain catalog.

Let's assume we have a device type called 'deviceTypeName' which has been modeled with slots named 'S1', 'S2', and 'S3'. We now want to assign the card types 'Card101', 'Card202' and 'Card303' to 'S1', 'S2', and 'S3' respectively. We can accomplish this type of import by using angle brackets '>'.

The format for 'ParentNode' is as follows:

'Project Object Name>Slot Name'

| | A | B | C | D |
|---|---|---|---|---|
| 1 | **Name** | **Type** | **ParentNode** | |
| 2 | Alpha | Building | Excel Import | {Latitude}40.4851 |
| 3 | 1st | Floor | Alpha | {Comments} 1500 |
| 4 | 101 | Room | Alpha>1st | |
| 5 | R1 | 42d | Alpha>1st>101 | |
| 6 | PP1 | Corning CCH-04U (empty) | Alpha>1st>101>R1>Back{20} | |
| 7 | A | Generic Vertical 12 port card (LC Ports) | Alpha>1st>101>R1>PP1>A | |
| 8 | B | Generic Vertical 12 port card (LC Ports) | Alpha>1st>101>R1>PP1>B | |
| 9 | C | Generic Vertical 12 port card (LC Ports) | Alpha>1st>101>R1>PP1>C | |
| 10 | R10987655 | Juniper MX80-48T-AC | Alpha>1st>101>R1>Front{10} | {IP_Address}10.0.( |
| 11 | Bentley | Building | Excel Import | {Latitude}39.2813 |
| 12 | 1st | Floor | Bentley | |
| 13 | 101 | Room | Bentley>1st | |

*Example of card assigned to parent device*

Attention!

Make sure the order of appearance of records in the spreadsheet follows the natural hierarchy associated with the objects. For example, if a card is associated to a specific device, make sure the device was created first. A common mistake is referring to a parent node which has not yet been created.

## 4.1.2.2 Mounting devices on racks

To mount a device on a rack, you must provide the rack unit number at the end of the hierarchy path of each device to be rack mounted. In the example below, a Corning CCH-04U device is mounted in position 20 in a rack that has front and back views. If you want to mount the device on the back of the rack, simply prepend the name of the back container (usually named 'back') to the rack position, such as 'Back{20}'.

| PP1 | Corning CCH-04U (empty) | Alpha>1st>101>R1>Back{20} |
|---|---|---|

## 4.1.3 Links

This worksheet is used for creating links in the netTerrain project.

| | A | B | C | D | E |
|---|---|---|---|---|---|
| 1 | **Name** | **Type** | **Node1** | **Node2** | |
| 17 | Fast Ethernet 6 | Ethernet | Alpha>1st>101>R1>R10987655>6 | Franklin>1st>101>R1>8934jue8>6 | |
| 18 | Fast Ethernet 7 | Ethernet | Alpha>1st>101>R1>R10987655>7 | Franklin>1st>101>R1>8934jue8>7 | |
| 19 | Fast Ethernet 8 | Ethernet | Alpha>1st>101>R1>R10987655>8 | Franklin>1st>101>R1>8934jue8>8 | |
| 20 | Fast Ethernet 9 | Ethernet | Alpha>1st>101>R1>R10987655>9 | Franklin>1st>101>R1>8934jue8>9 | |
| 21 | Fast Ethernet 10 | Ethernet | Alpha>1st>101>R1>R10987655>10 | Franklin>1st>101>R1>8934jue8>10 | |
| 22 | Fast Ethernet 11 | Ethernet | Alpha>1st>101>R1>R10987655>11 | Franklin>1st>101>R1>8934jue8>11 | |
| 23 | Fast Ethernet 12 | Ethernet | Alpha>1st>101>R1>R10987655>12 | Franklin>1st>101>R1>8934jue8>12 | |
| 24 | Fast Ethernet 13 | Ethernet | Alpha>1st>101>R1>R10987655>13 | Franklin>1st>101>R1>8934jue8>13 | |
| 25 | Fast Ethernet 14 | Ethernet | Alpha>1st>101>R1>R10987655>14 | Franklin>1st>101>R1>8934jue8>14 | |
| 26 | Fast Ethernet 15 | Ethernet | Alpha>1st>101>R1>R10987655>15 | Franklin>1st>101>R1>8934jue8>15 | |
| 27 | Fast Ethernet 16 | Ethernet | Alpha>1st>101>R1>R10987655>16 | Franklin>1st>101>R1>8934jue8>16 | |
| 28 | Fast Ethernet 17 | Ethernet | Alpha>1st>101>R1>R10987655>17 | Franklin>1st>101>R1>8934jue8>17 | |
| 29 | Fast Ethernet 18 | Ethernet | Alpha>1st>101>R1>R10987655>18 | Franklin>1st>101>R1>8934jue8>18 | |
| 30 | Fast Ethernet 19 | Ethernet | Alpha>1st>101>R1>R10987655>19 | Franklin>1st>101>R1>8934jue8>19 | |
| 31 | Fast Ethernet 20 | Ethernet | Alpha>1st>101>R1>R10987655>20 | Franklin>1st>101>R1>8934jue8>20 | |
| 32 | Fast Ethernet 21 | Ethernet | Alpha>1st>101>R1>R10987655>21 | Franklin>1st>101>R1>8934jue8>21 | |
| 33 | Fast Ethernet 22 | Ethernet | Alpha>1st>101>R1>R10987655>22 | Franklin>1st>101>R1>8934jue8>22 | |

*Links worksheet*

The columns include:

- Name: mandatory field that represents the name of the link
- Type: mandatory field that identifies the type associated with the link. Once a type is defined, the custom fields in the spreadsheet will correspond to the custom fields associated with that node type in the netTerrain catalog
- Node1: mandatory field identifying the name of the starting node or device.
- Node2: mandatory field identifying the name of the ending node or device
- Fields: the custom fields associated with the type of the link, which work just like the case with objects specified above.

## 4.1.3.1 Creating links between ports

Sometimes you need to create links between ports. This creates an additional naming convention challenge because ports are usually just named based on their sequential port number. Naturally, in such a case just using the port name to identify endpoints would be a problem since many ports with the same number might exist in the project.

To overcome this, you can identify ports by appending the device ancestry information in both the Node1 and the Node2 columns:

'Device Name>Port Name'

For instance, let's say we have a device called 'myDeviceName' with two ports 'P1' and 'P2' and another one called 'myOtherDevice' also with two ports 'P1' and 'P2'

To link both 'P1' ports together and the 'P2' ports together we could format the Node1 and Node2 columns like indicated below:

| | A | B | C | D |
|---|---|---|---|---|
| 1 | **Name** | **Type** | **Node1** | **Node2** |
| 2 | MyFirstLink | LinkTypeNameHere | myDeviceName>P1 | myOtherDevice>P1 |
| 3 | MySecondLink | Logical | myDeviceName>P2 | myOtherDevice>P2 |

*Creating links between ports*

In a similar fashion we can embed and concatenate card information in our endpoint columns, such as the example below:

| | A | B | C | D |
|---|---|---|---|---|
| 1 | **Name** | **Type** | **Node1** | **Node2** |
| 2 | MyFirstLink | LinkTypeNameHere | myDeviceName>Card1>P1 | myOtherDevice>P1 |
| 3 | MySecondLink | Logical | myDeviceName>Card1>P2 | myOtherDevice>P2 |

*Creating links between ports residing in cards*

## 4.1.3.2 Bundling Links

The Excel import sheet can be used to bundle links as well. As we can see in the image below for cell 'E3', you can bundle links by using the "bundle" label inside curly braces ({ }) .

Outside the curly braces, enter the name of the link you want to bundle. Repeat this process for each link you wish to bundle. In the following example the fiber link "24SF1" will be linked to the conduit link "C100":

| | Name | Type | Node1 | Node2 | |
|---|---|---|---|---|---|
| 1 | **Name** | **Type** | **Node1** | **Node2** | |
| 2 | C100 | Conduit | Alpha | Bentley | {Length}75 Miles |
| 3 | 24SF1 | 24 Strand Fiber | Alpha>1st>101>R1>PP1 | Bentley>1st>101>R1>PP1 | {bundle}C100 |

## 4.1.3.3 Assigning Fiber Strands

Users can assign fiber strands with the Excel Import sheet. For each fiber link, follow these steps:

1) Enter the name of the fiber link in column A, followed by the ">" symbol.

2) Append the strand number to the fiber link using a three-digit format (e.g., 001, 002, 003, etc.).

3) Example: If the fiber link name is "FiberLink1" and the strand number is 001, enter "FiberLink1>001".

Leave column B (Type) blank. It will not be used in the case of fiber strand assignments.

Like other links, columns C and D are used for providing the full hierarchy path.

| Name | Type | Node1 | Node2 |
|------|------|-------|-------|
| FiberLink1 | 24 Strand Fiber | Alpha>1st>101>R1>PP1 | Bentley>1st>101>R1>PP1 |
| FiberLink1>001 | | Alpha>1st>101>R1>PP1>C>B1 | Bentley>1st>101>R1>PP1>A>B1 |
| FiberLink1>002 | | Alpha>1st>101>R1>PP1>C>B2 | Bentley>1st>101>R1>PP1>A>B2 |
| FiberLink1>003 | | Alpha>1st>101>R1>PP1>C>B3 | Bentley>1st>101>R1>PP1>A>B3 |
| FiberLink1>004 | | Alpha>1st>101>R1>PP1>C>B4 | Bentley>1st>101>R1>PP1>A>B4 |
| FiberLink1>005 | | Alpha>1st>101>R1>PP1>C>B5 | Bentley>1st>101>R1>PP1>A>B5 |
| FiberLink1>006 | | Alpha>1st>101>R1>PP1>C>B6 | Bentley>1st>101>R1>PP1>A>B6 |
| FiberLink1>007 | | Alpha>1st>101>R1>PP1>C>B7 | Bentley>1st>101>R1>PP1>A>B7 |
| FiberLink1>008 | | Alpha>1st>101>R1>PP1>C>B8 | Bentley>1st>101>R1>PP1>A>B8 |

## 4.1.4 Updating values

The Excel import sheet can be used to update values of objects in netTerrain. This is quite trivial: simply add the objects to be updated to the spreadsheet along with the fields to be updated. If netTerrain can uniquely identify the objects from the spreadsheet (whether due to the unique name of the object or the combination of the name and path) it will not create a new object, but simply update the fields specified in the spreadsheet.

## 4.1.4.1 Using curly brackets to insert and update values

In the examples above, values for fields are inserted or updated in netTerrain by virtue of the name of the column in the spreadsheet matching a field in netTerrain. However, you can also update values without worrying about the name of the column by prepending the value to be updated with the name of the field to be updated using curly brackets like this:

| | A | B | C | D | E |
|---|---|---|---|---|---|
| 1 | Name | Type | ParentNode | | |
| 2 | Alpha | Building | Excel Import | {Latitude}40.4851 | {Longitude} -78.7247 |
| 3 | 1st | Floor | Alpha | {Comments} 1500 Sq Feet | |

In the example above, the latitude and longitude fields for the first record and the comments field for the second record are updated using curly brackets. The advantage of that, as can be seen, is that you don't need to worry about the order or sequence of fields between one record and the next.

## 4.1.4.2 Changing the Name of objects

Since the name of an object is also a common data field you can also change the name of an object the same way you would insert data into other fields.

## 4.1.5 Positioning Objects and reparenting

The Excel import spreadsheet can be used to automatically position nodes on netTerrain diagrams. Options include x/y positioning, lat/long positioning and reparenting.

## 4.1.5.1 Changing x/y coordinates

The x/y coordinates of a node can be applied from the Excel import spreadsheet to place objects on a diagram. This is the default method if the diagram does not have a map.

The format for specifying x/y coordinates is the same as with any field, using curly brackets:

```
{x}Coordinate

{y}Coordinate
```

For the standard 8.5 by 11 inch diagram, the x coordinate ranges from 0 to 950 units and the y coordinate ranges from 0 to 700 units.

## 4.1.5.2 Lat & Long Coordinates

A node can be placed on a diagram containing a georeferenced map using latitude and longitude coordinates. The format for lat/long coordinates, as expected, uses curly brackets:

```
{Latitude}Coordinate

{Longitude}Coordinate
```

## 4.1.5.3 Moving Objects to Different Diagrams

Using the 'reparent' label, you can move a node from one diagram to another with the Excel import sheet. To reparent a node, place the {reparent} label followed by the new path in a custom field for the given record. The example below shows the 101 room being moved from the 1st floor to the 2nd floor.

| | 101 | Room | | Charles>1st | | {reparent}Charles>2nd | |

## 4.1.6 Troubleshooting and Tips

As you build your import spreadsheet, we recommend testing the import in phases to catch input mistakes early.

When errors occur on input, please refer to the 'App_data' in the netTerrain web directory folder. The log files are named as follows: '[date time] EI [name of uploaded spreadsheet]'.

Here are some common mistakes and their resolution:

## 4.1.6.1 Input string not in correct format issue:

Below is a typical logged error related to input string problems:

```
Input string was not in a correct format.

 at System.Number.StringToNumber(String str, NumberStyles options,
NumberBuffer& number, NumberFormatInfo info, Boolean parseDecimal)

 at System.Number.ParseInt32(String s, NumberStyles style,
NumberFormatInfo info)

 at System.Convert.ToInt32(String value)

 at NetTerrain.Importers.ExcelImporter.ExcelTypeToImportFormat(DataRow
nodeTypeRow, String nodeTypeName)…
```

Usually this error stems from two possible data entry mistakes:

1) The Excel import spreadsheet had its structure (the columns) modified. This is most likely due to a change the compatibility level. You should make a new copy of the original bulk import spreadsheet and copy over the information.

2) Not all mandatory fields were filled out

## 4.1.6.2 Type image ignored

You may get the following error in the log:

Type image ignored. File does not exist.

Make sure that when you are importing node types, you zip together the spreadsheet with a folder called 'graphics', which contains all the referenced icons.

## 4.1.6.3 Parent node missing

You may get the following error in the log:

Cannot import node xyz as its parent node xyz cannot be found.

This one is self-explanatory and usually the resolution is simple. Make sure that the parent node exists or, if it is supposed to be imported during the same process, that it is higher up in the row sequence than the nodes that produce the error.

## 4.2 Proceeding to import from Excel

It is not uncommon for organizations to have more than one user to fill out data in Excel that should be imported into netTerrain. In many cases you could end up with several Import files waiting to be queued for netTerrain import. For that reason, netTerrain can import one file at a time or import a whole batch.
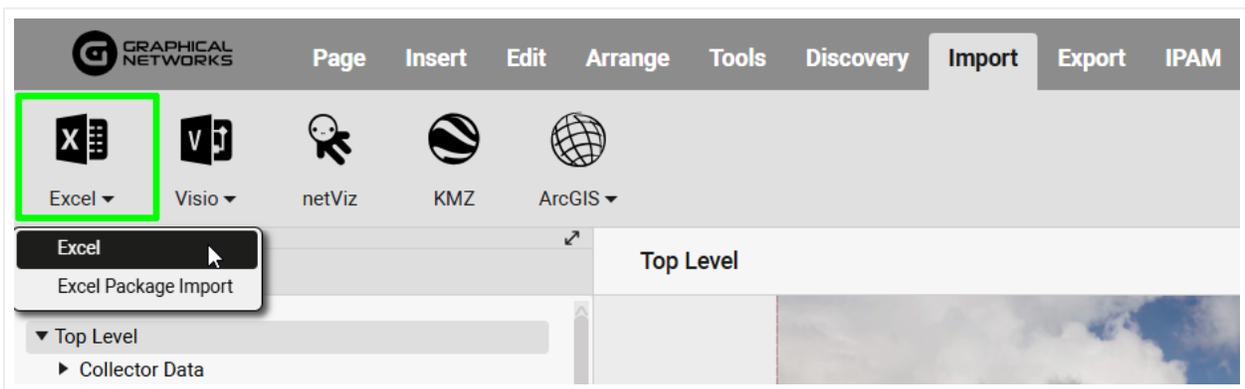


*Excel bulk import options*

## 4.2.1 Importing single files

Once you have completed your bulk import spreadsheet, you can proceed to import the data into netTerrain one file at a time by pressing on the 'Excel Import' sub menu under the Excel Import menu option, as shown above.
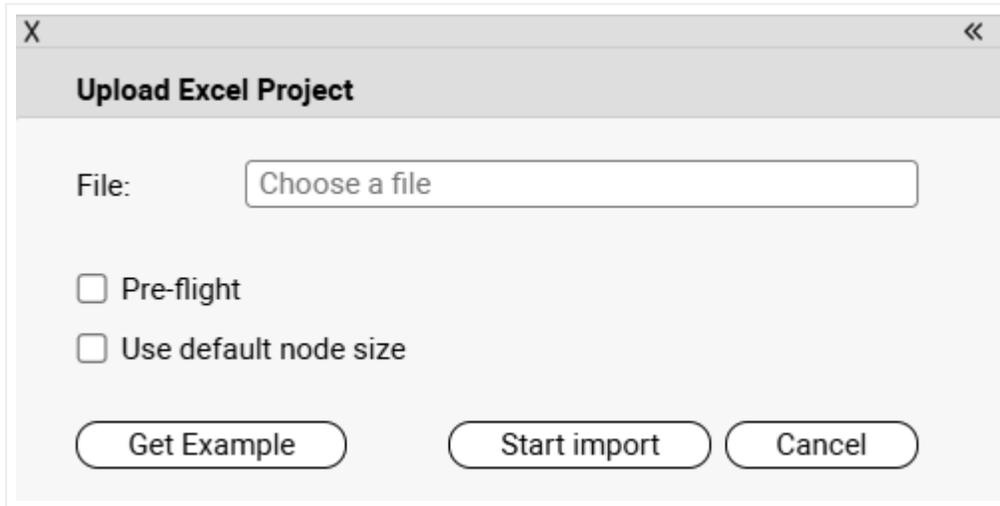


*Single file import option*

After prompted to choose a bulk import file from your local machine or connected network folders, you can fine-tune the import process with the 'pre-flight' and 'Use default node size' options.

The Pre-flight option previews the import before it actually happens. This gives you the ability to review and fix any issues with the data being imported, before it is committed. Additionally, if you want to override any node sizes with the default node size defined in the catalog, make sure you check the 'Use default node size' checkbox.

Once you are ready, click on the 'Start import' button to start the process.

## 4.2.1.1 Default Node Size

When importing from excel, the sizing of objects will use dynamic sizing based on the diagram page size. To use the default node size regardless of the diagram page size, check the "Use default node size" option.
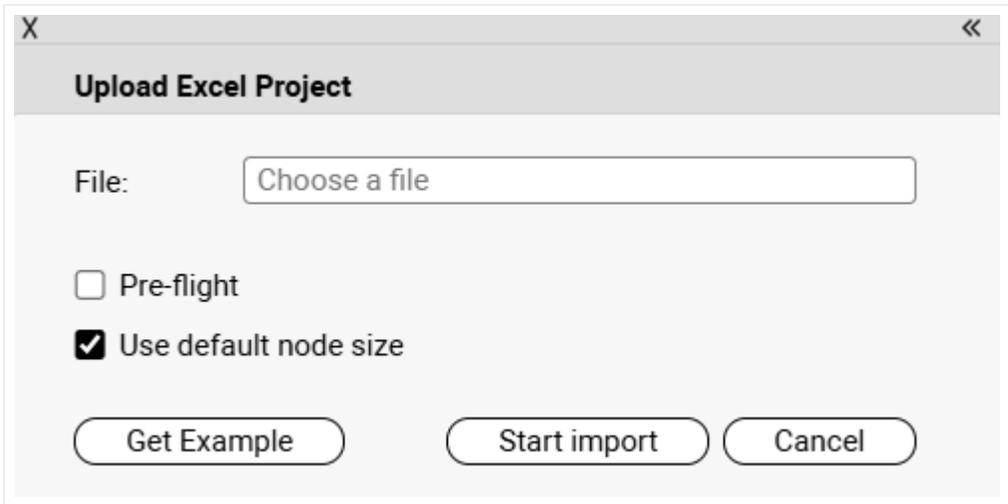
## 4.2.2 Importing a package

The 'Import package' option is quite self-explanatory and prompts the user to choose multiple Excel files and import them as a package.
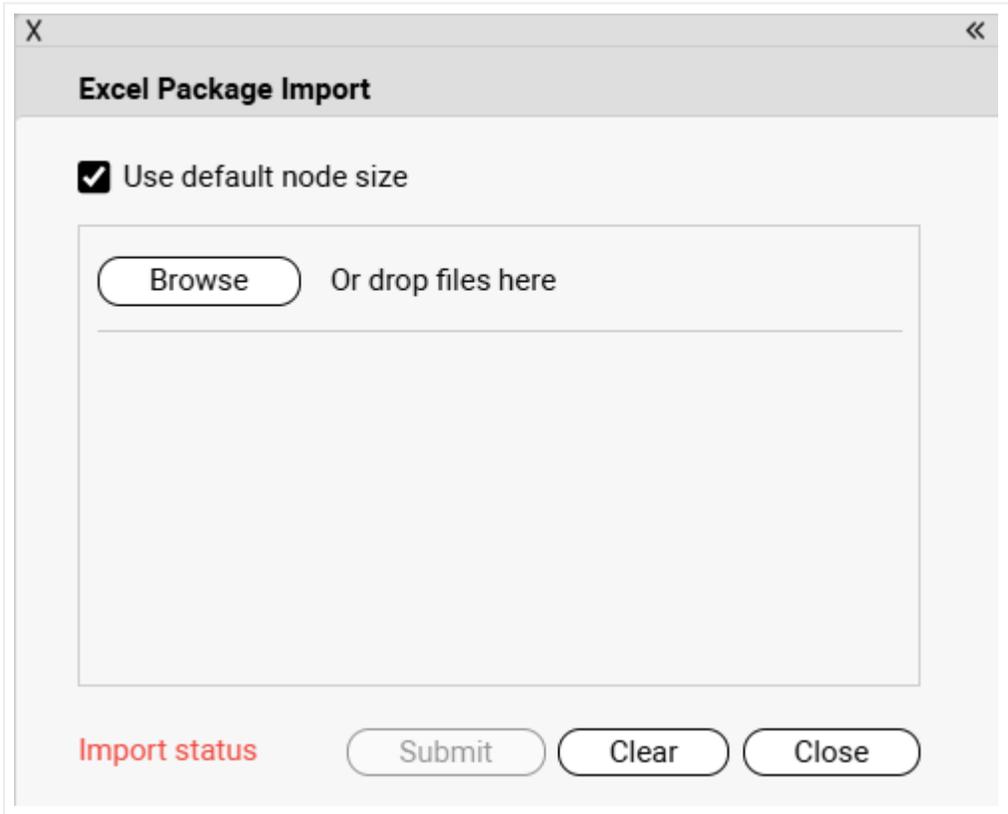


*Package file import option*

Remember that these files need to be formatted based on the base ExcelImport file standard. After selecting the list of files, a dialog showing each individual file, with the status Ready for upload will be displayed.
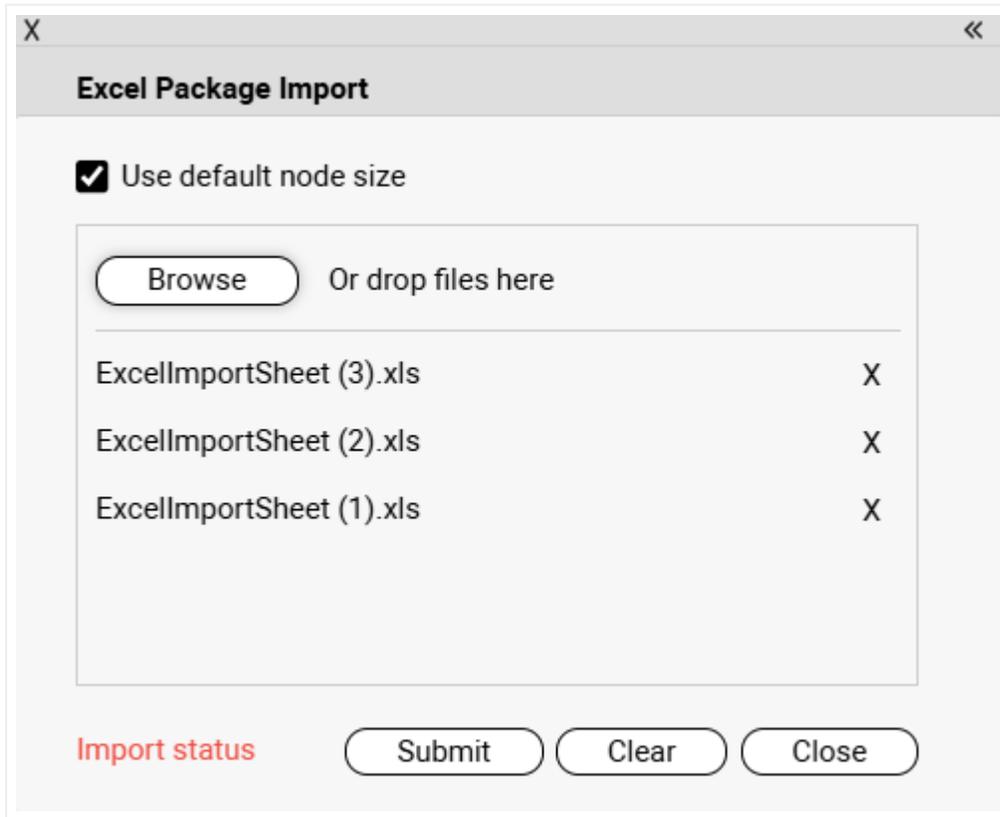


*File upload dialog*

This list can be cleared if you want to start over. Once you are ready, press the Submit button to get the process started. And get yourself a big cup of coffee, as this can take a while.

It is also recommended to do this type of process involving large files during the night, or when users are not actively on netTerrain because it can slow the server down considerably.

Users can access the Import status list, by clicking on the Import status link as shown above. This will display a list of the files in the queue. The status column will show files that are still queued up as 'New', files currently processed as 'In process', the ones that are completed as 'Done" and any files that errored with the label 'Error".

Logs for individual files already processed can also be accessed by clicking on the 'Show log' link.

An administrator can remove any pending files from the queue. A non-admin user can remove their own files from the queue.

# 5 KML and KMZ Files

KML and its cousin, KMZ (zipped KML) are typical formats used by mapping or GIS applications, such as Google Earth, and they can be easily imported into netTerrain. netTerrain can also export an OSM-based diagram (a netTerrain diagram that has an Open Street Map reference) to KML. KML/KMZ imports and exports are used in the context of a netTerrain OSP project, specifically to import data that exists in another OSP type application and minimize manual data entry.

Nothing needs to be setup on the netTerrain server or on the client side to make the import and export work but note that these features only work for netTerrain diagrams containing a dynamic map. Dynamic maps are only available with a valid netTerrain OSP license.

## 5.1 Importing from KMZ

Before importing a KML file, users need to be aware of certain basic considerations and requirements, such as which diagrams are valid for imports, as well as the characteristics of the map to be used.

### 5.1.1 Prerequisites for importing KML / KMZ files

As stated, a KMZ file is basically a compressed version of a KML file, but there is, however, an important distinction to make: KMZ files can contain images, typically located under a "files" folder. These images would not be present in the KML file since the latter is essentially a text file with a special XML style format.

netTerrain can use images contained in the KMZ file and associate them with the corresponding nodes that are imported. If you want to preserve the same icons used in your application that exported the data, you may want to utilize the raw KMZ format instead of the unzipped KML file and make sure that those images are indeed included. From now, we will mostly just refer to KMZ files, when explaining this feature.

Before importing a KMZ file into netTerrain, make sure that you are on a diagram in netTerrain that already has a dynamic map in it.

This map should ideally be large enough to correctly place all nodes that exist in the KMZ file. This means that every latitude and longitude coordinate associated with a node in the file should be contained within the boundaries of your netTerrain map, otherwise, any nodes falling outside of the boundaries will be discarded. Since importing the data is easy, if your original map is not correct, you can delete the imported data, change the map, and retry. Also, it is better to err on the side of a larger than needed map. You can change it

to a smaller map later, and netTerrain will preserve the coordinates (if they are still within the boundaries of the smaller map).

> **Note**
>
> Since KMZ data is flat in nature, netTerrain will not produce any hierarchy of any kind. As such, all the data will be placed in one diagram, which in some cases may produce a very large diagram that takes a long time to load.

## 5.1.2 Importing a KMZ/KML file

To import a KMZ file navigate to the diagram that contains the map that will hold the KMZ data and click on the import from KMZ button found under the Import & Export ribbon.
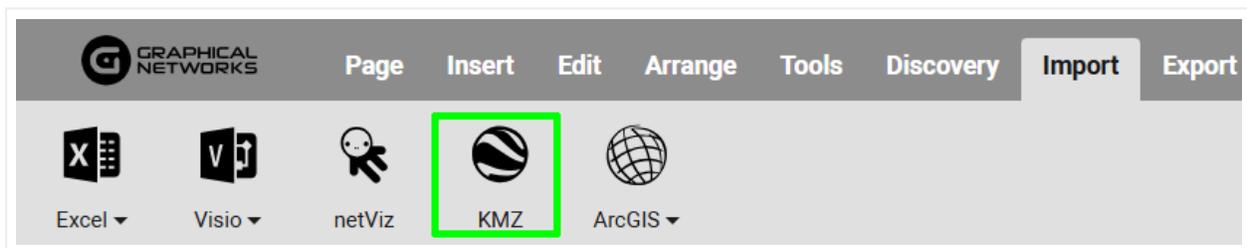


*KMZ import button*

In the upload KMZ file dialog navigate to the folder that has the KMZ file and click on Start Upload. Once the import finished, an Import completed dialog appears. After you click on Ok the map is refreshed, containing the KMZ data.
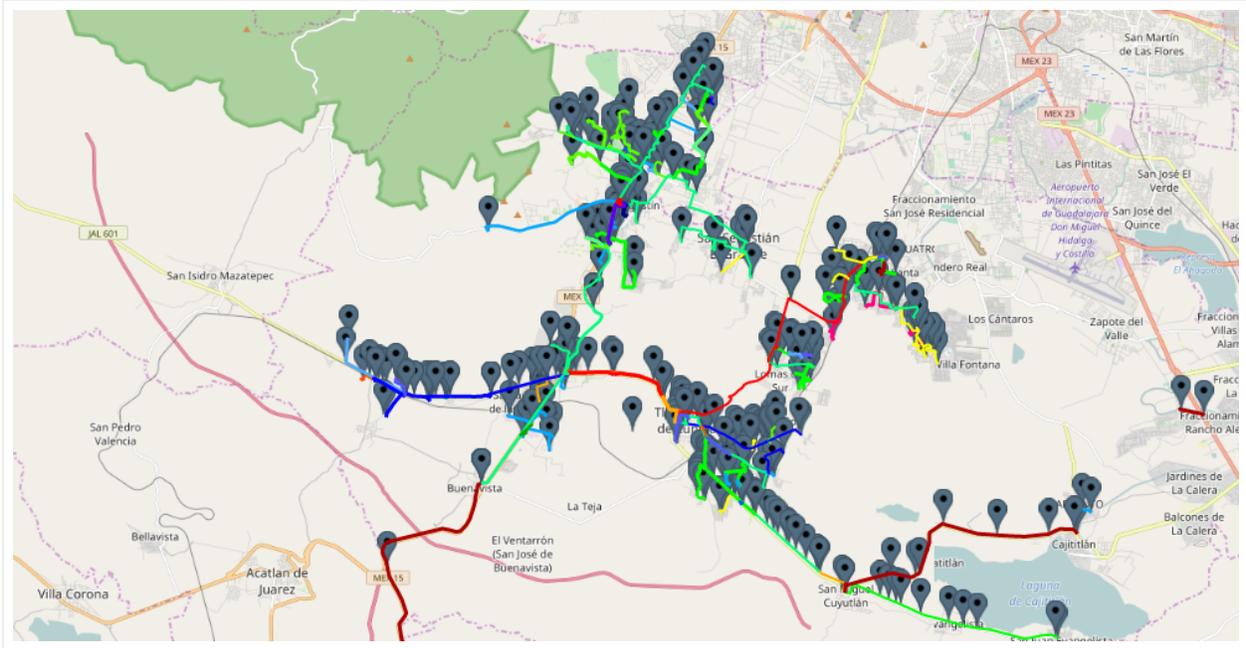
*Sample KMZ import into netTerrain*

## 5.2 Exporting to KMZ

Exporting a netTerrain diagram to KMZ is a very straightforward task.

First, navigate to the netTerrain diagram that contains the data to export. Remember, it must be a diagram that contains an OSM map.

Then click on the Import & Export ribbon and click on the KMZ Export button on the right.
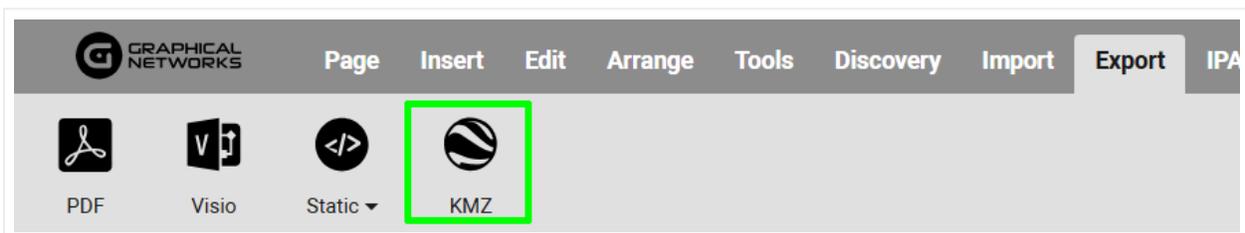


*KMZ export button*

In the upload KML / KMZ file dialog navigate to the folder that has the KMZ file and click on 'Start Upload'. Once the import finished, an "Import completed" dialog appears. After you click on 'Ok' the map is refreshed, containing the KMZ data.

*Sample KMZ import into netTerrain*

A few things to note about the results you see after the import:

- Nodes are automatically assigned a type
- The same icon you used in Google Earth (or your original tool) should be seen in netTerrain
- Lat long positioning is, as expected, automated
- Lines should include bend points to follow the same path you used in the source
- Line colors should be preserved
- "Dangling" links are not supported, as such, netTerrain creates fictitious, transparent endpoints to represent these lines

# 6 ArcGIS Import

This chapter explains the process to import ArcGIS data into netTerrain.

The ArcGIS Importer is a standalone Windows Console application that collects information about the desired ArcGIS features using the ArcGIS REST API and imports those features into netTerrain using the netTerrain REST API.

This application must first be downloaded from netTerrain.

## 6.1 Downloading the ArcGIS Importer ZIP

To download a ZIP file containing the necessary files to perform an ArcGIS Import, press on the Download ArgGIS Importer sub menu under the ArgGIS Import menu option. The ZIP file can be unpacked anywhere on your machine.
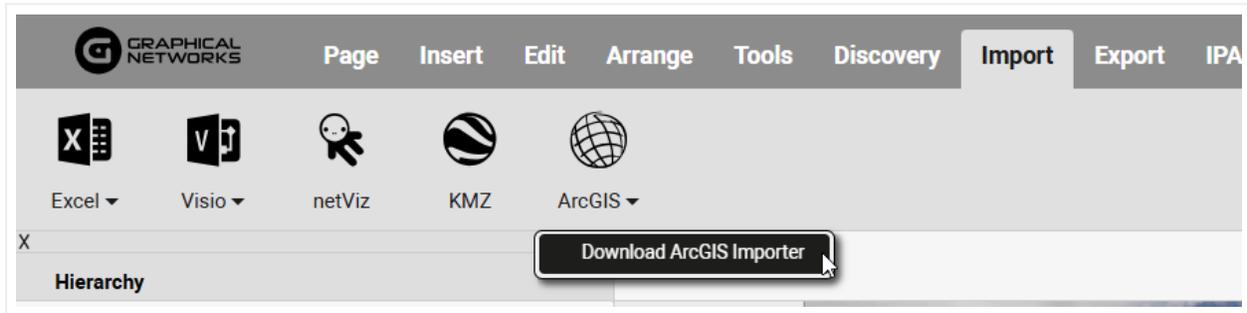


## 6.2 Prerequisites in netTerrain

For the ArcGIS import to work correctly, you must set up netTerrain for the import, which involves the following tasks:

- Creating object types
- Preparing diagrams
- Setting up the API
- Creating a user token in netTerrain

## 6.2.1 Creating Object Types

In netTerrain, create a node type (or link type for polyline features), or use an existing one, for each desired ArcGIS feature type to be imported.

When importing ArcGIS features, the only supported values for "geometryType", as specified in the feature layer information, are "esriGeometryPoint" and "esriGeometryPolyline".

Add fields to the netTerrain object type for every ArcGIS attribute to be imported for each feature type. For simplicity, if field names are added that match the feature attribute names (comparing is case-insensitive), then no additional mapping is required later.

## 6.3 Preparing diagrams and the API

Before attempting to import ArcGIS features into netTerrain, make sure that the target diagram(s) have a dynamic map for a background.

Any map should ideally be large enough to correctly place all objects that exist on the feature layer to be imported. The import process will skip any features with latitude and/or longitude values outside the boundaries of the defined map.

In the Admin Console, select the "Global Settings" tab, and in the section labelled "API Settings", check the "Enable RESTful API" option as show below.

## 6.4 Creating a User Token in netTerrain

If you do not wish to enter user credentials (username and password) in plain text in the application's config file (see AGImport.exe.config below), then a user token is required.

The user token must be created for a User that has at least PowerUser level permissions.

To create a user token, login as the desired User, then press on the 'My Settings' sub menu under the User menu option. In the My Settings dialog box, click on 'My User Tokens'.

On the resultant web page, click on 'Add New'.
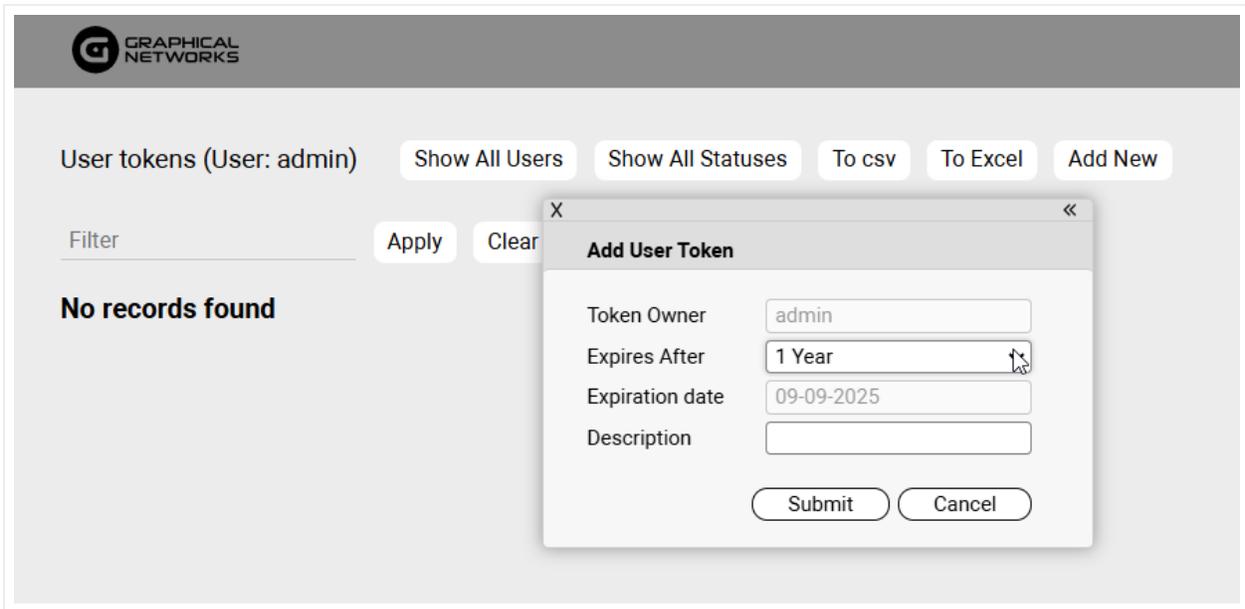
*A screenshot of a computer Description automatically generated*

In the 'Add User Token' dialog box, change the expiration period and add a description as desired, then click on 'Submit' and follow the instructions.

## 6.5 Supplied Files

The standalone Windows Console application to perform the import is called AGImport.exe and downloaded from netTerrain directly.

Following is the syntax used within the console application:

```
AGImport [/W] [/C] [/T] [/P] [/S] [/M] [/R] [control_file]

AGImport /?

control_file Specifies a control file (.xml) - optional - default is
AGImport.xml

/W Waits for user input to exit program

/C Directs verbose output to console instead of log file

/T Reports would-be changes but does NOT update netTerrain (TESTING)

/P Update position of existing objects in netTerrain
```

```
/S Update size of existing objects in netTerrain

/M Mark unmatched objects for deletion in netTerrain

/R Really delete unmatched objects in netTerrain

/? Displays this help text only
```

Example:

AGImport /C /T /W

## 6.5.1 AGImport.exe.config

The application automatically looks for this file in the same folder as the executable resides. This file is provided but the contents should be updated. It specifies the URL for the netTerrain server (which is the base URL for the REST endpoints) and provides authorization, either by supplying user credentials or a user token generated in netTerrain. The netTerrain user in either case must have at least PowerUser level permissions.

Sample file contents (using credentials):

```xml
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <startup>
    <supportedRuntime version="v4.0"
sku=".NETFramework,Version=v4.7.2" />
  </startup>
  <appSettings>
    <add key="NetTerrainUrl" value="http://localhost/netTerrain" />
    <add key="UserName" value="someuser" />
    <add key="Password" value="somepwd" />
  </appSettings>
</configuration>
```

Sample file contents (using user token):

```xml
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <startup>
    <supportedRuntime version="v4.0"
sku=".NETFramework,Version=v4.7.2" />
  </startup>
  <appSettings>
    <add key="NetTerrainUrl" value="http://localhost/netTerrain" />
    <add key="Token"
value="TCLe0mPLzSBhL6yha84oKqhe170skyqYlN6zAMaO0dA8h24EEK7FYeuPOK2QMKfreZ
>
  </appSettings>
</configuration>
```

## 6.5.2 AGImport.xml

The application automatically looks for this file in the same folder as the executable resides.

Sample file contents:

```xml
<?xml version='1.0' encoding='UTF-8'?>
  <esri>
    <diagram id="24000000000290"
file="sampleserver6_AGP_USA_MapServer_Cities.xml" />
  </esri>
```

Each `<diagram>` element in the XML must have the following attributes:

| | |
|---|---|
| id | The ID of a Diagram in netTerrain |
| file | An XML configuration file for importing into that Diagram |

Multiple `<diagram>` elements are allowed, using the same or different Diagram IDs.

## 6.5.3 arcgisonline_AGP_USA_MapServer_Cities.xml

This is the import configuration file referenced by the supplied AGImport.xml and is provided as an example. Any number of configuration files may be created to import from multiple sources and/or multiple layers.

The format of any import configuration file is standard XML. There are 3 root elements:

```
<api-key>
<deleted-postfix>
<layers>
```

The `<api-key>` is optional. If access to your ArcGIS REST endpoints requires an API Key, and that key is universal, then that value should be supplied here.

The `<deleted-postfix>` is optional. If the AGImport application is run with the /M option to "mark" objects for deletion, this can be used to modify the text string that will be appended to the name of objects that are marked. If nothing is supplied here, the default value applied by the application is "(DELETED)".

The `<layers>` element can contain any number of `<layer>` elements. Each `<layer>` element will consist of these possible child elements:

```
<jsonfile>
<endpoint>
<api-key>
<nt-type>
<nt-size>
<name-field>
<lat-field>
<long-field>
<field-mappings>
```

The `<jsonfile>` is optional. If specified, it takes precedence over the value of `<endpoint>`. Use this if you have used another tool to capture the query response for a particular feature layer (as a JSON file) using your ArcGIS REST endpoint.

See the following files for examples:

· Sample_Layer_0_QUERY_from_sampleserver6.arcgisonline.com.json

· Sample_Layer_1_QUERY_from_sampleserver6.arcgisonline.com.json

The `<endpoint>` is required if `<jsonfile>` is not specified. This is the ArcGIS REST endpoint associated with the feature layer that is being imported. All features are requested from this endpoint. There is no filtering of the results.

The `<api-key>` is optional. It only needs to be specified here if access to this ArcGIS REST endpoint requires a different API Key than any universal key supplied earlier.

The `<nt-type>` is required. It specifies the object Type in netTerrain to use for this feature layer.

The `<nt-size>` is optional. It specifies the node size to use for this layer. The size is defined in netTerrain's internal units (100 units per inch) with a minimum size (width and height) of 0.001. The proper aspect ratio for the node Type image is maintained. If left blank, the node size is calculated based on the coverage area of the map.

The `<name-field>` is optional. If the ArcGIS feature attributes do not include a field called "Name", or the feature layer information does not designate a "displayFieldName", then the feature attribute to map to the netTerrain "Name" field must be supplied here.

The `<lat-field>` and `<long-field>` tags are optional. The import into netTerrain requires object positioning by latitude and longitude coordinates. If the feature geometry specifies X and Y values in some other coordinate system, but the feature also specifies the latitude and longitude using other attribute fields, then those field names can be supplied using these two elements. This only applies to nodes ("geometryType" = "esriGeometryPoint").

The `<field-mappings>` element can contain any number of `<mapping>` elements.

Each `<mapping>` element in the XML must have the following attributes:

| field | The name of the field (attribute) in the features being imported |
|---|---|
| property | The name of the field (property) to map to in the netTerrain object Type |

Multiple `<mapping>` elements are used to map multiple fields.

## 6.5.4 Use Notes and Recommendations

The ArgGIS import uses an Object ID value that uniquely identifies every feature to be imported for the specified layer. The attribute field containing this ID is typically named "objectid", but it may be named otherwise provided that the field is designated as "type" = "esriFieldTypeOID".

When performing an import, features will be created in netTerrain if their Object ID does not yet exist on the specified Diagram. If that Object ID already exists, then the data fields will be updated for the matching object. By adding the appropriate command line switches, the position of objects may also be updated, and unmatched objects may be deleted (or marked for deletion).

If the application is run with the /T option (including in combination with other options), it will only report what would be changed in netTerrain without making those changes.

It is strongly recommended that importing into a specified diagram be done at a time when no users are actively viewing that diagram in a client browser.

For the Name field for each object, if specific positioning is required (such as to the right or below the object), it is suggested that a node of the desired type be temporarily added to any standard diagram (i.e., without a dynamic map) and the displayed field be positioned as desired. After doing so, right-click on the displayed field for the Name and select the option 'Set Default' in the context menu.

This should be done prior to importing any nodes of the specified node type. The temporary node can then be deleted. It is not necessary to resize the temporary node. When objects are created during the import process, they are sized as specified in the import configuration file and the Name field is scaled accordingly.

If it is better to have an additional field displayed on the diagram for each imported node, go to the catalog, select the node type, show its fields, and check 'Is Displayed' for the field to be displayed. Also give that field a non-blank default value. Then repeat the same process as described above for the Name field, i.e., add a node, position the displayed field, and select the option 'Set Default' in the context menu. A non-blank default value is needed for the displayed field to be created, scaled, and positioned properly during import prior to any actual data value being set.

If the size of the nodes imported is not acceptable, the value of `<nt-size>` can be changed in the import configuration file before running the application again with the /S option to update the size of the nodes. However, if the size needs to be changed dramatically, it is recommended that the nodes be deleted in netTerrain before running the import application again to recreate the nodes with the desired value of `<nt-size>`.

Even if the size change is modest, it may be necessary, after importing with the /S option, to manually update the displayed field positions on the diagram. To do so, select the displayed field for any node on the diagram, move and size it as desired, then right-click on the displayed field that has been changed and select the option 'Apply to type on Diagram' in the context menu.

This will update the displayed field position for all nodes of the same node type on the diagram.

This guide is part of the official documentation for netTerrain, developed by Graphical Networks.

**GRAPHICAL NETWORKS**