Razor Agile Installation and Configuration Guide

Note: It is highly recommended that testing NOT be performed on a production server.

# Note: This guide will cover an Enterprise version of the Administration Tool. Not all features of this tool are available in the Starter Pack, Standard Pack or Professional versions

# Overview

Razor is a lightweight solution. To try out and test it, simply install all three applications, the Admin, Server, and Client (along with Git and MariaDB) directly on your desktop.  There is no need to set up a separate server and you can run Razor in either a traditional "classic" or "agile" development mode. You can start with the default set of forms, workflows, and automated email notifications, and then customize whatever you need from there.

System Requirements:

Windows Server:

1. Windows Server 2012 or better
2. Operating System Disk (C Drive) > 200GB
3. Separate Git repository drive > 100GB Not on the C drive
4. Do not use a shared drive for the Git repository

Windows Desktop – Test Drive

1. Windows 7, 10 or 11
2. Operating System Disk (C Drive) > 100GB
3. Separate Git repository drive > 50 GB not on the C drive

Linux Server:

1. CentOS 8 Stream, Ubuntu xxx
2. Operating System Disk Size 100GB or Better
3. Separate Git repository drive > 100GB Not on the C drive

Note: Linux servers and clients will require the installation of Mono.

Downloads:

As a first step, type “razoragile” or into your browser or enter the full link to the Razor web page as follows:

<https://resources.visiblesystemscorp.com/razor-6-starterpack>

Halfway down the page you will see images to download and install the following three Razor software applications by clicking on each icon.

[](https://www.visiblesystemscorp.com/Download/RazorAgile/Razor6Admin_v6.0x.exe)Razor Admin Tool

[](https://www.visiblesystemscorp.com/Download/RazorAgile/Razor6Server_v6.0x.exe)Razor Server Software

[](https://www.visiblesystemscorp.com/Download/RazorAgile/Razor6Client_v6.0x.exe)Razor Client Software

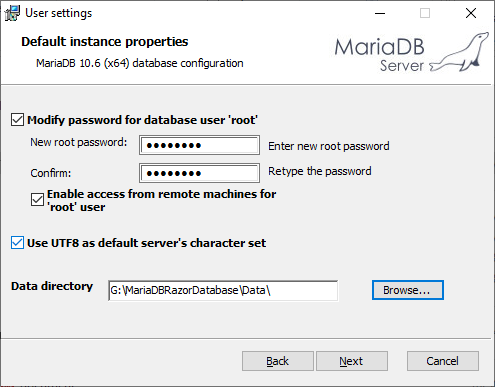
[Git-Logo-1788C](https://4863804.fs1.hubspotusercontent-na1.net/hubfs/4863804/Razor%206/Git-2.32.0.2-64-bit.exe)

[https://resources.visiblesystemscorp.com/hs-fs/hubfs/https___mariadb.com_wp-content_uploads_2019_11_mariadb-logo_blue-transparent.png?width=217&name=https___mariadb.com_wp-content_uploads_2019_11_mariadb-logo_blue-transparent.png](https://4863804.fs1.hubspotusercontent-na1.net/hubfs/4863804/Razor%206/mariadb-10.6.7-winx64.msi)

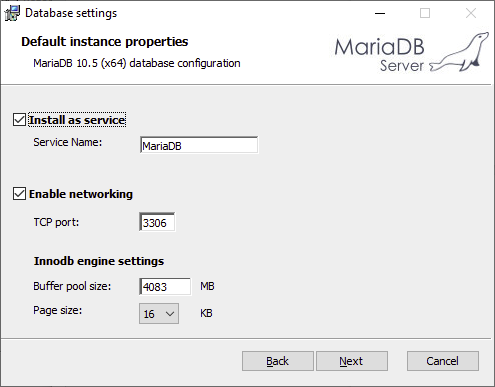
If you haven't already done so, be sure to download and install Git and MariaDB. By clicking on each one of the two icons (just above) you will be able to download and install both applications.

# Razor 6 Administration Tool Installation

1. Once the downloads are completed and ready, you can begin the installation process as follows.
2. First Install the Git package where the Razor server will be running.
   1. Select all the default settings and complete the installation
   2. Note DO NOT change the default installation folder
3. Install the MariaDB on to the C drive but install the Database on the separate volume by following these instructions.
   1. Create the database directory on the separate volume i.e. G:\MariaDBRazorDatabases\Data
   2. Next proceed to install the MariaDB on the C Drive but on the following step:

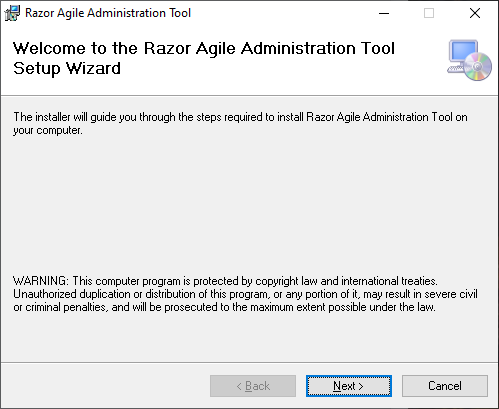


* 1. On the default instance properties window, enable access…, and Use UTF8 as the character set. Also, make sure that you remember the password entered
  2. Use the browse button to point to the new data directory
  3. Since this is where most of the growth is using a separate volume makes some sense.
  4. Click Next
  5. On the second default Instance Properties page make sure it looks exactly like below

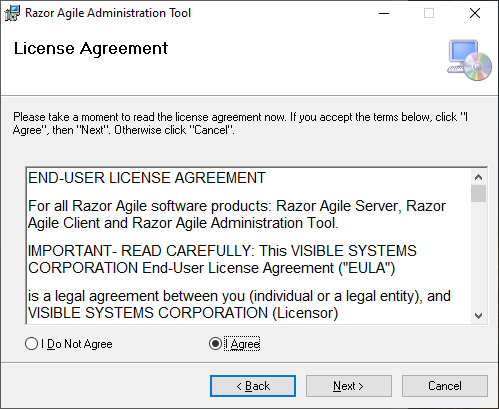


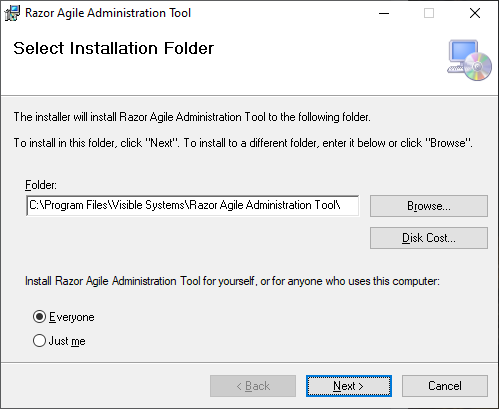
* 1. Click Next
  2. Click Install

1. Next install the Razor Administration tool



* 1. Click on Next
  2. Accept the License Agreement and click next

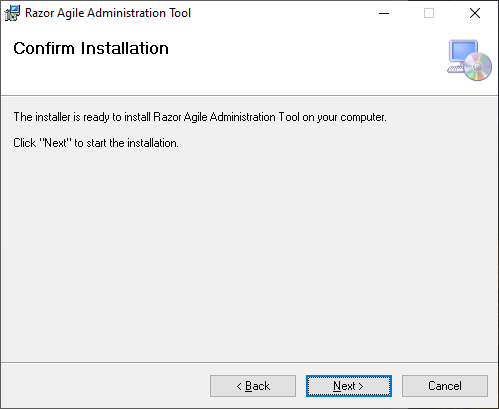




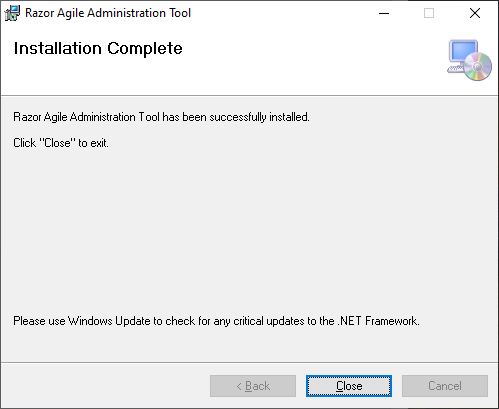
* 1. Leave the Default settings DO NOT change the installation Folder
  2. Click Next



* 1. Since the database is local, leave these settings as they are. If they are not change them to the Database server and port number as needed
  2. Click Next
  3. The Razor Administration tool is now ready to install - Click Next



* 1. Click Close to complete the installation



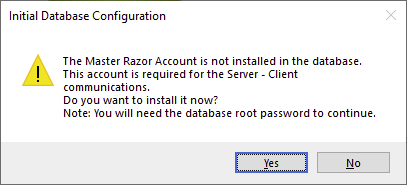
1. There should be an Icon on your desktop that looks like:



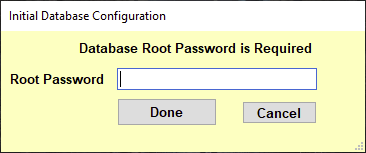
1. Double Click on the Icon to begin the final phase of the installation

# Razor Agile Administration Tool Final Configuration

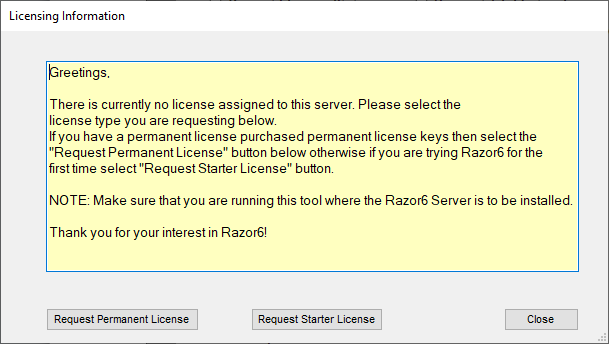
1. When the Icon is clicked for the first time there are some final configuration items to complete.



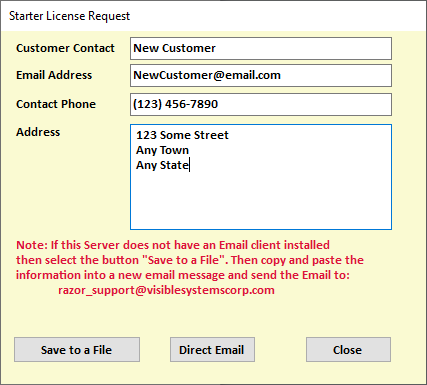
1. This first configuration is to install the user that communicates between the Admin tool, server and client. Without it, Razor6 cannot function. You must have the database root password to install this user (See step b above). If you click on No, the program will exit
2. Click Yes



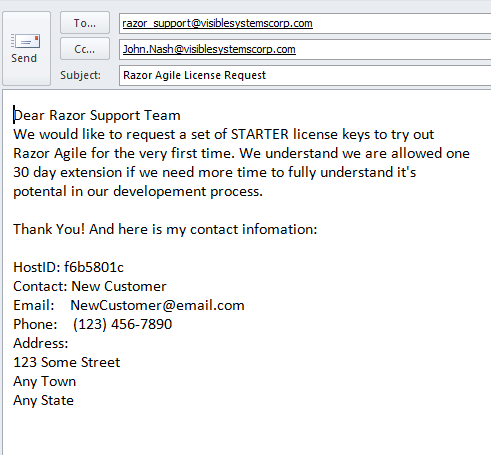
1. Enter the root password for MariaDB from step v above.
2. Click on Done



1. At this time, you will need to request either a Permanent License if you have purchased one or the Starter License. The Starter license is for 30 days for 15 users and is free.
2. Fill out the form as required.



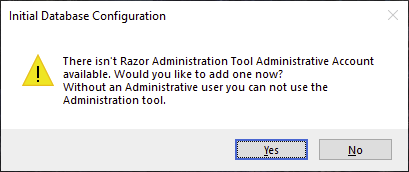
1. If this server has an email client installed, select direct email. It will bring up the email client and have everything filled out.



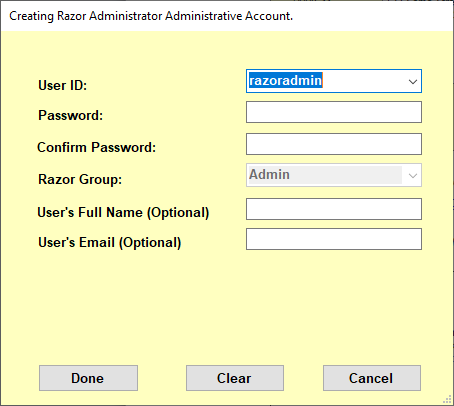
Otherwise select save to file and send the information to:

[Razor\_support@visiblesystemscorp.com](mailto:Razor_support@visiblesystemscorp.com)

1. Click on close and the next window will appear. Select yes



1. The next screen is asking for a Razor Administration tool administrator. This account is only used for the Administration Tool. Click on Yes.



1. The suggested user is presented, but you can change the user if desired. The login and passwords have restrictions such as login length and password strength. Once the login and passwords are set, click Done.

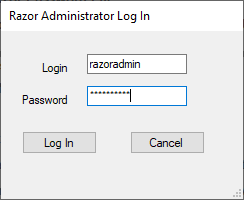
You must provide this login and password to the Razor Administration Tool each time you use the tool.

# Welcome to the Razor 6 Administration Tool

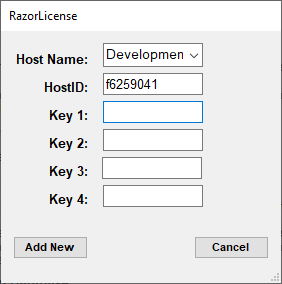


Notice that most of the configuration options are greyed out with the exception of Add Razor licensing and the tool itself is labeled as Unlicensed. Once you receive the return email from Razor support it will include the details of the license issued. There will be four keys that must be installed in order to activate your license tier. So, let’s begin

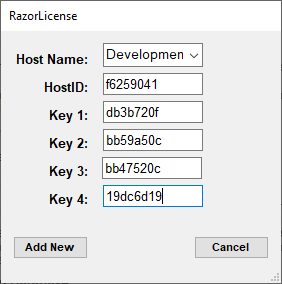
1. Click on Add Razor License and the log in window will pop up. Or you can click on “Click here to log in”



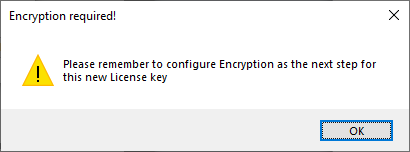
1. The Licensing window will pop up. Ensure that the HostID and Hostname matches the one in the email sent to you with the license keys.



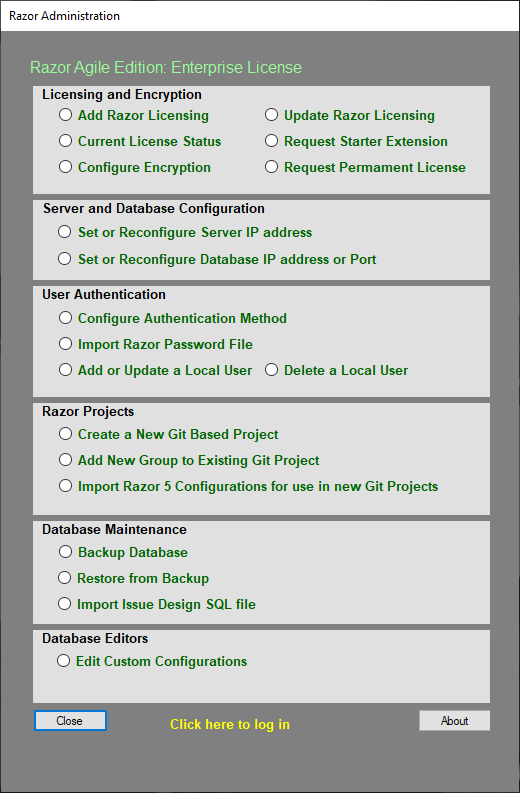
1. Next simply copy and paste the keys into each key slot and click on Add new



1. The next window will remind you to install the encryption key for all server / client communications



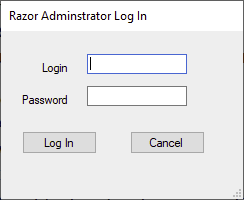
1. Click on install and the Administration Tool will unlock to your available license tier



# Using the Administration Tool

### Logging into the Administration Tool

The administrator can log into the tool by either clicking on any of the desired features or by clicking on the “Not Logged In” text located at the bottom of the tool. The administrator is presented with a request to log in



Use the login and password that you created when the tool was installed (i.e., see page 8)

### Licensing and Encryption

* 1. Add Razor Licensing. Allows the Administrator to add licensing for other servers. This will be discussed when the server is released
  2. Update Razor Licensing. When the license is expired and you have received new keys, this is where the keys are updated
  3. Current License Status. Shows how many users are allowed, how many users are connected and expiration dates for the licenses.
  4. Request trial extension. Provides a template for requesting an extension. A copy can be emailed to [razor\_license@visiblesystemcorp.com](mailto:razor_license@visiblesystemcorp.com)
  5. Configure Encryption. See next section
  6. Request Permanent License. Provides a template for requesting permanent keys.

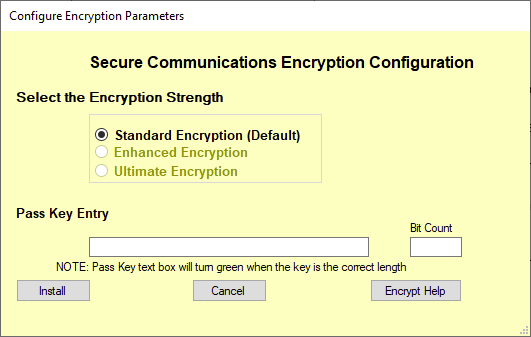
Note: You can contact Visible Systems Corporation Sales by calling 315-363-8000 or sending an email to [razor\_sales@visiblesystemscorp.com](mailto:razor_sales@visiblesystemscorp.com)

The license agreement may be found at:

<https://visiblesystemscorp.com/Products/Razor/license_agreement.htm>

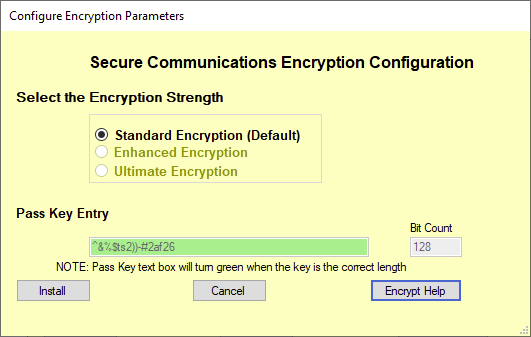
### Configure Encryption

This function configures the pass-phrase between the client and server to fully encrypt the communications between the client and server. At the moment only 128-bit encryption is allowed. In the future, Enhanced encryption will provide 192-bit protection, and Ultimate encryption will provide 256-bit protection for all database data. This added security will ensure that data is securely transmitted between the Razor clients and the Razor server.



The pass-key is entered by the user and should be a random set of characters using upper case, special characters and numbers. It should be a totally meaningless phrase.

Here is an example:



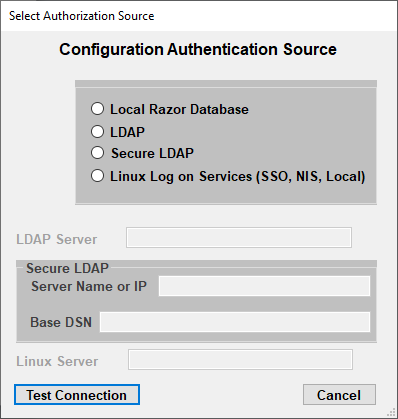
### User Authentication

* 1. Configure Authentication Method. Razor Agile will use the internal MariaDB user/password database by default
  2. LDAP, Secure LDAP and Single Sign On (Linux) or NIS to authenticate users.
  3. Import Razor Password Files. The Administration tool can import the existing Razor 5 Password file leaving current settings for users intact.
  4. Add or update a Local user. Used by the Razor administrator to manage the local users.
  5. Delete a Local User. The administrator can delete a local user here.

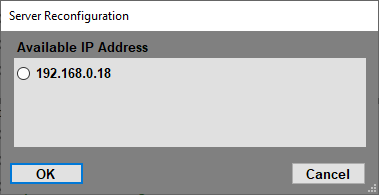
Note: Managing users at the Active Directory, LDAP etc. levels are managed at their source.

Note: When importing a Razor 5 password file, the file must have read permissions and be accessible by the Administration Tool

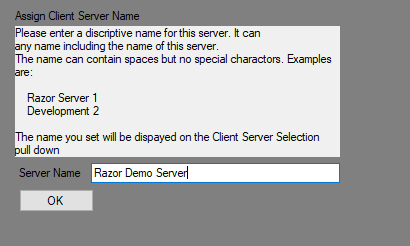
1. Click on the Configure Authentication Method and this screen is displayed



1. Select any of the Authentication methods
   1. Local Razor Database contains all users in MariaDB
   2. LDAP will use the Domain LDAP server
   3. Secure LDAP will use a secure connection to the LDAP server
   4. For NIS, Single Sign on enter the Server name
2. Next Select Set or Reconfigure Server IP Address
   1. The first form will display all of the Active IP address that are available for this server.
   2. Select the IP address that all Razor Agile communications are to happen over



* 1. Click on OK
  2. Next set the name of the server that is to be presented to the users. This name does not have to be the host name of the server.

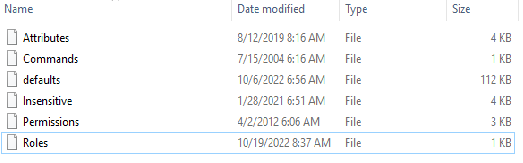


### Optional Import of Existing Issue formats

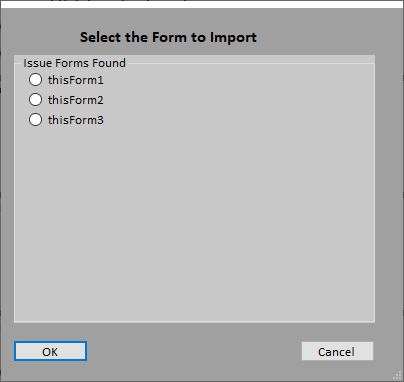
Razor Agile can use existing Issue formats from Razor Classic in all new Projects created. The import process is simple, but will require some adjustments to the final layout of the imported format due to the differences between the old GUI type and the more modern format now available. The Admin tool attempts to make the best guess on the new format, but sometimes not so great. The good news is that the updates to the format are done through the Admin Tool and are quick to do.

So, let’s get started.

* 1. Create a directory that is accessible to the Admin Tool. We will suggest Razor Imports
  2. Add the following files to the directory of the form that you want to import



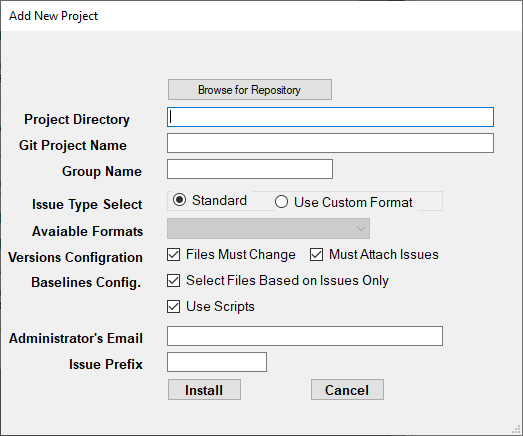
* 1. Note the W or X defaults file is renamed to defaults.
  2. Next start up the Admin tool and select Import Razor 5 Configuration for use in new Git Projects
  3. Log in if needed
  4. Navigate to the Razor Imports directory and click OK
  5. The form should appear listing all the forms possible to import from the defaults file



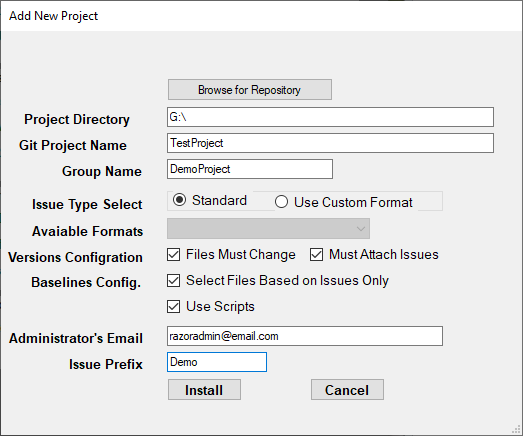
* 1. Notice there are 3 forms in this defaults file. Make sure that the Attributes, Permissions, Insensitive, Roles and Command files match the from you wish to import
  2. Select the form and click on OK. The form will and all supporting information is placed into the database

### Razor Projects

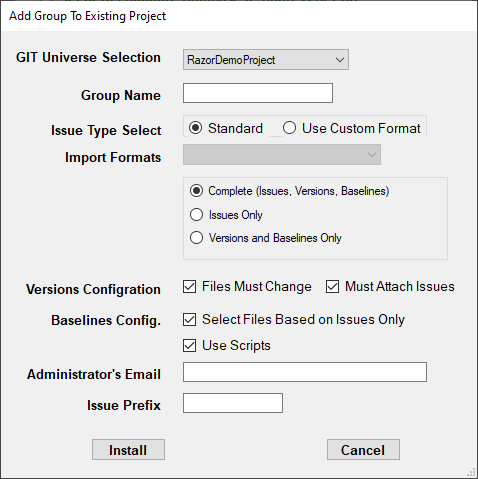
1. Create a New Git Based Project



1. The Browse for Repository option provides a Directory browser so the Administrator can select the location where the new Project is to be created
2. Project Directory will display where the new Project is to be created
3. Git Universe Name - All characters are allowed except spaces
4. Group Name - All characters are allowed except spaces
5. Issue Type Select.
   1. Standard will have the new Project use the built-in form for Issues, Versions and Baselines
   2. Use Custom Format. If the Administrator has imported an existing Razor5 Defaults, Insensitive etc. Files, then when this option is selected the available formats will present themselves in the Available formats pull down.
6. Versions Configuration: Files must change before checking in and Must Attach an Issue to a file check out
7. Baseline Config. Can only update a Baseline based on an Issue
8. Administrator’s Email. Used by Git
9. Issue Prefix: Single letters are still allowed, but this can be any phrase between 1 and 8 characters. For instance, an Issue prefix can be: I, Issue, Docs etc.



1. Add a new group to an Existing Project.

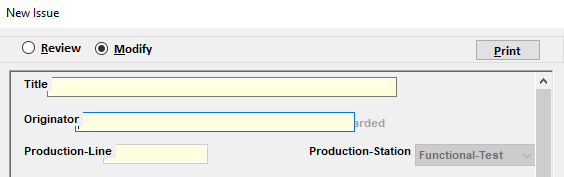


1. Select the Project to add the new group to add to
2. Enter a group name
3. Issue type select. Same as above
4. Select what to add. Complete adds (Issues, Versions and Baselines), Issues only or Versions and Baselines only
5. Versions Configuration
6. Razor Administrator’s Email
7. Issue Prefix as before

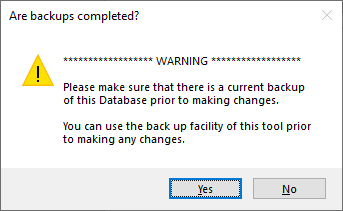


### Fixing an Imported Issues Form

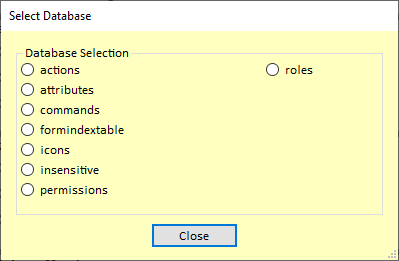
* + 1. Bring up the Razor Client and log into Issues
    2. An imported for may look something like this



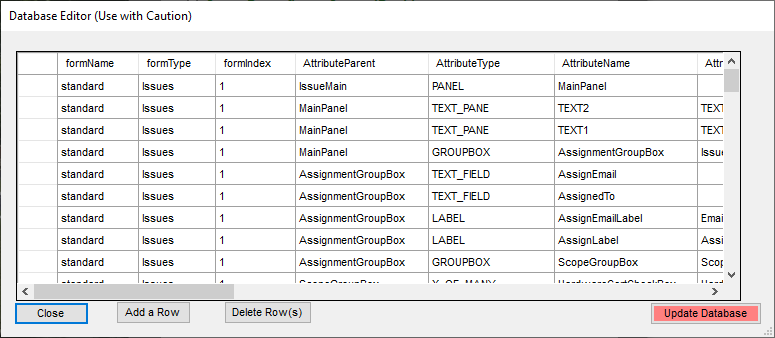
* + 1. Note the text boxes are a bit shifted to the left and some items are located under others. Close the open issue but do not log out of the issues program.
    2. To fix this bring up the Admin tool and select Edit Custom Configurations
    3. A warning message will appear.



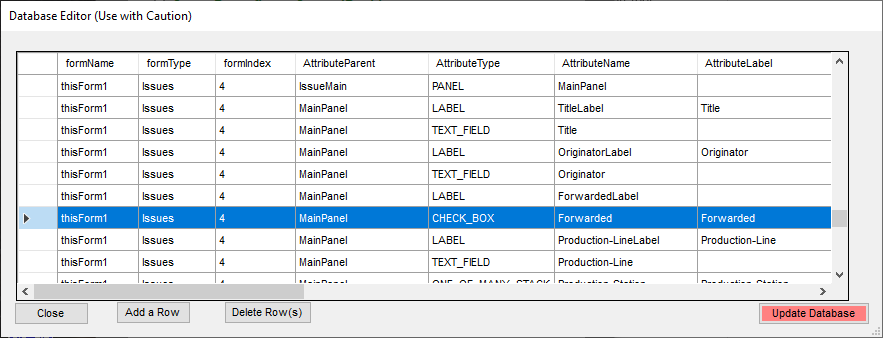
* + 1. Please heed this message and see the section on backing up the databases
    2. The next screen will be which database to edit select attributes



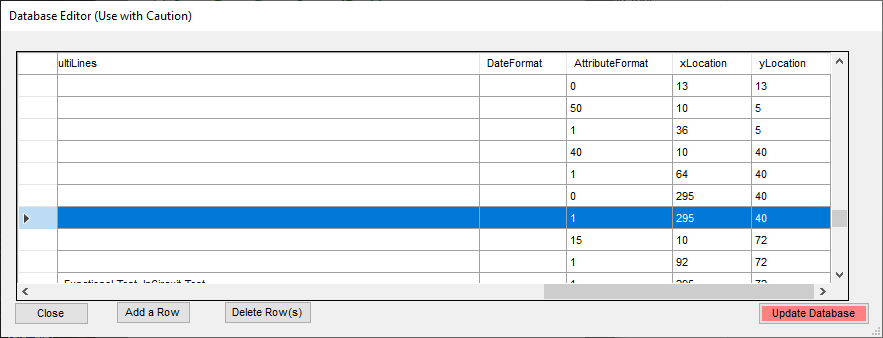
* + 1. The database editor will now be presented.



* + 1. Scroll down until you see the field you need to change



* + 1. Next slide over so the xLocation, yLocation are visible



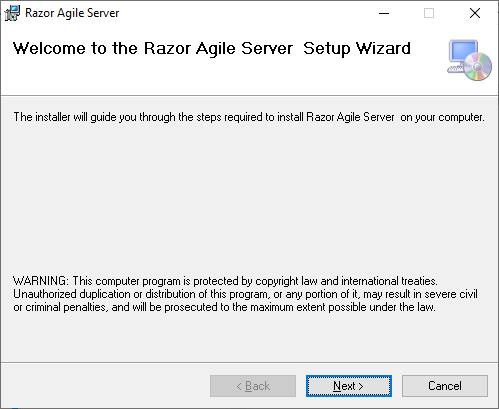
* + 1. Click on the X location field for the attribute and increase its value to move the item to the left or decrease its value to move it to the right. If you need to move the Y location decreasing the value move the attribute up and increasing moves the attribute down.
    2. Click on Update Database and select new issue on the Client tool. If the attribute is in the correct place you are done. Otherwise close the issue and repeat steps 11 and 12 until every look great.

### Database Maintenance

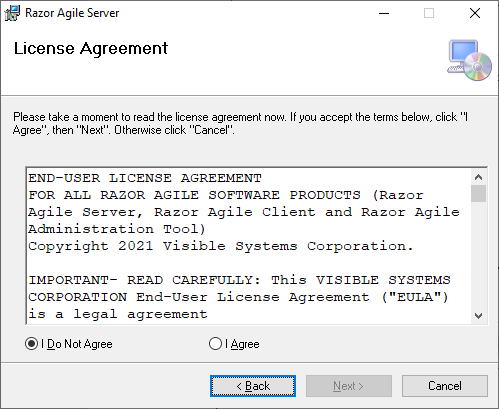
1. Backup Database: Allows for complete backups of all Razor databases or the backup of individual databases
2. Restore Database: Will restore an existing backup

# Razor Server Installation

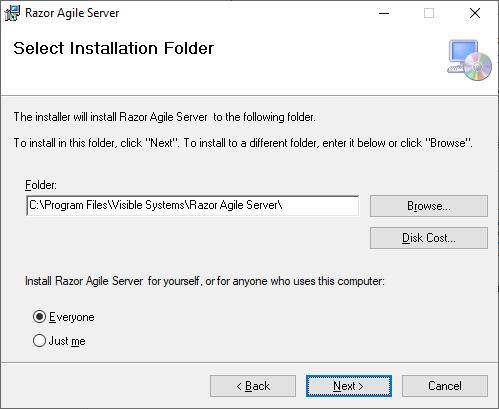
1. Navigate to where the files were downloaded to and double click on the setup.exe file.



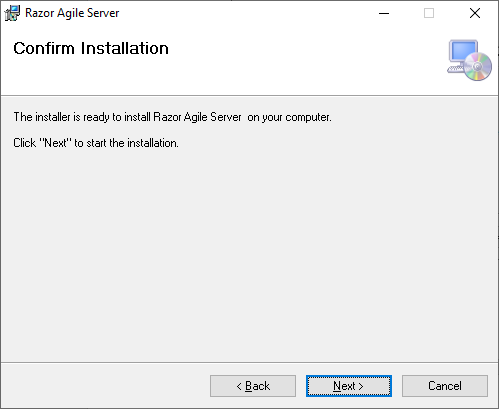
1. Click on Next



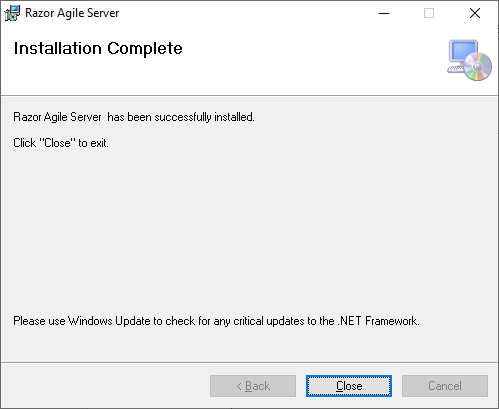
1. Read the License agreement select I Agree if you are OK with the license, Click Next



1. Select all the defaults. Note: DO NOT change the install directory. Click Next



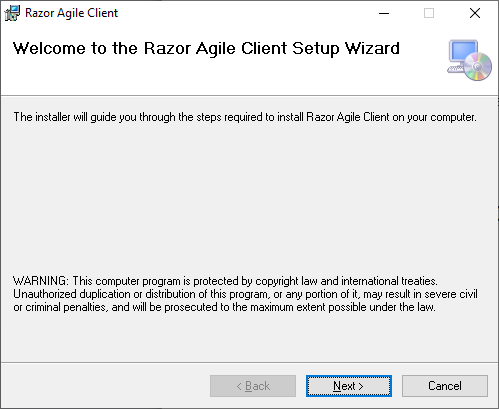
1. Click Next to begin the install process.



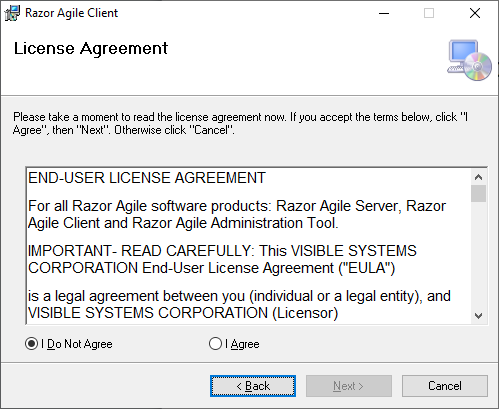
1. The server is now installed and started

# Razor Client Installation

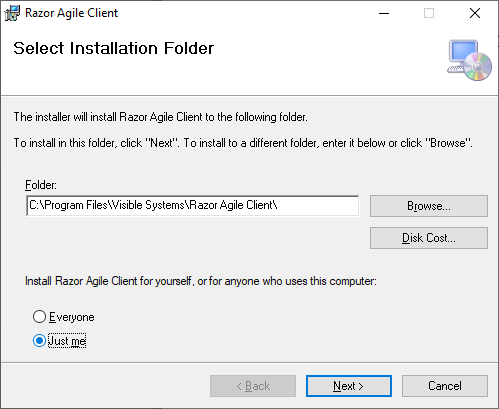
1. Navigate to where the files were downloaded to and double click on the setup.exe file.



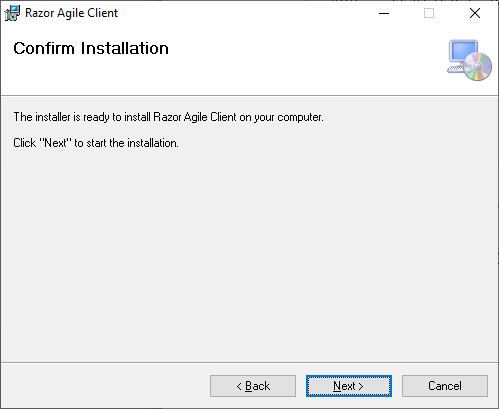
1. Click on Next



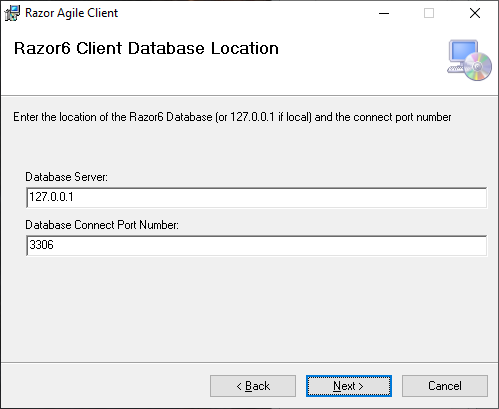
1. Read the License agreement select I Agree if you are OK with the license, Click Next



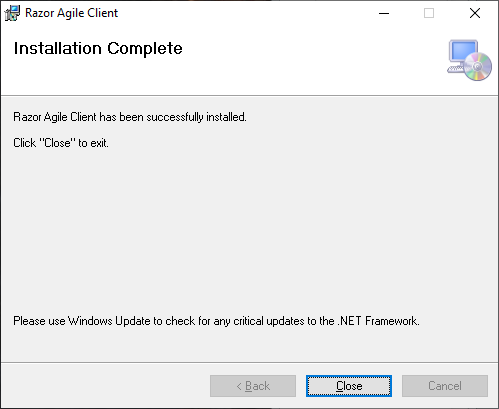
1. Select all the defaults. Note: DO NOT change the install directory. Click Next



1. Click Next to Continue



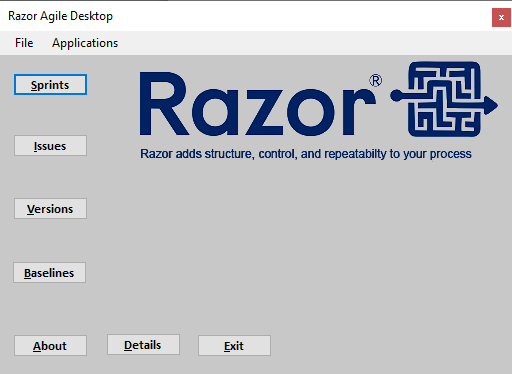
1. If you are not installing the client on the same server as the database as in the Test Drive configuration, then your Razor Administrator should have provided the information as to the Database server and Port number. Edit these fields as necessary and click Next.



1. Installation is now complete and there should be an Icon on your desktop for the client

# Razor Agile Desktop

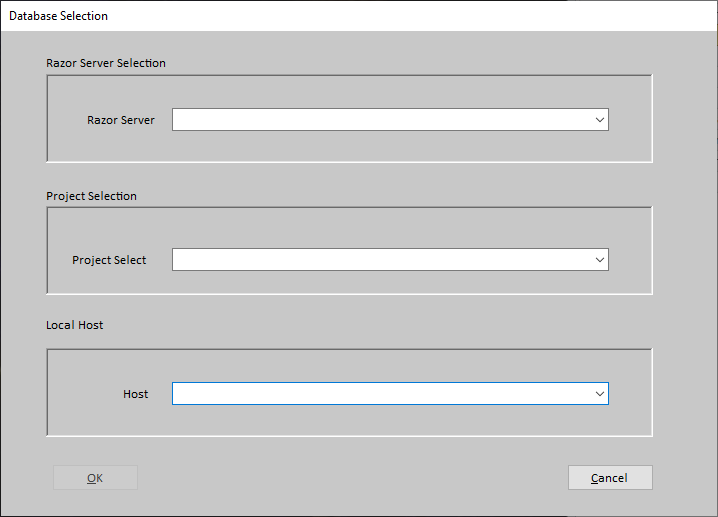
1. Double clicking on the Client Icon will bring up the Razor Agile Desktop



The Razor Agile Desktop is the gateway to all of the tools that are available to the user along the left had side as well as the Applications pull down. The buttons along the bottom provide information on the release via the About button, details on the tools via the Details button and the Exit button exist the Desktop.

One can use the TAB key to navigate through the selections or press the ALT+ the Letter for the desired functions. Pressing the enter key will select the highlighted function.

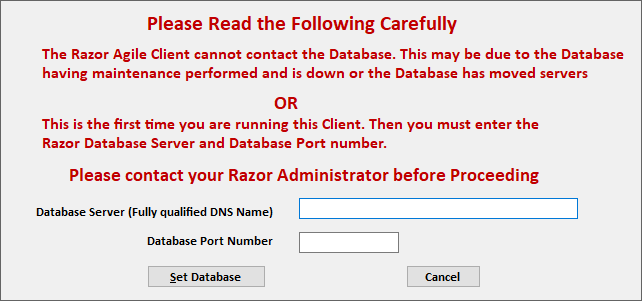
1. Clicking on any of the tools buttons will bring up the Database Selection form.



This tool allows for the selection of the Razor Server to contact in the event that your Razor administrator has set up more than one, the Project to connect to and presents all available local host IP’s to use for commutations to the Razor Server. As before use of the TAB key and the arrow keys make the selections quick and easy.

1. Database is not avaiailable

In the case that the Client detects thaty the database is down or may have moved, the following screen may be displayed. Please read it carefully and contact your Razor Administrator to see if the Database id offline or has moved



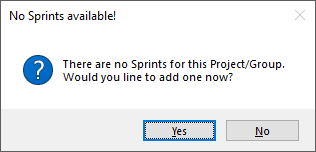
# Sprints

## Razor Agile Sprints

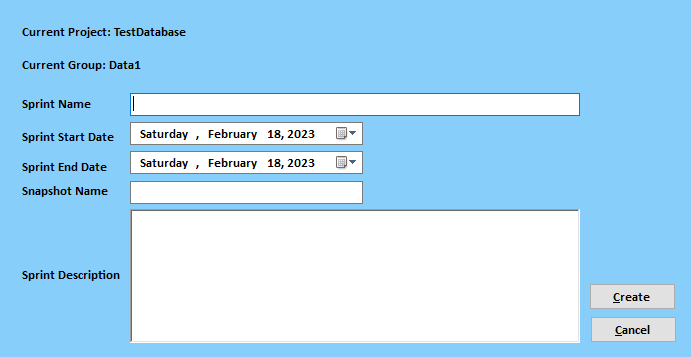
Razor Agile now has Superset to Issues that incorporates Agile Software development. This system uses Sprints and Stories to provide greater insight into the development process. It even incorporates rapid releases in the form when a story is completed, the Snapshot for that story is automatically updated with all the files worked on and associated to that story.

## Start Working with Razor Agile Sprints

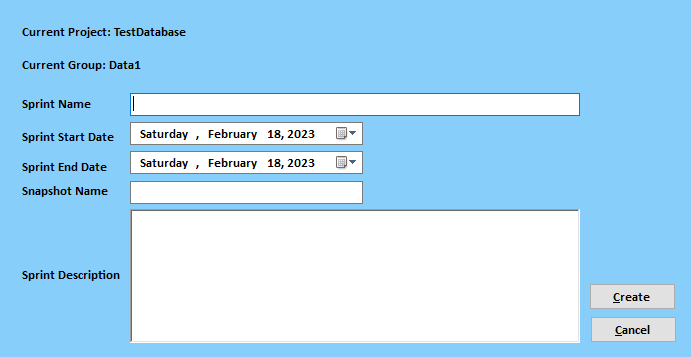
When selecting Sprints for the first time form the Razor Agile Desktop the following screen may be displayed



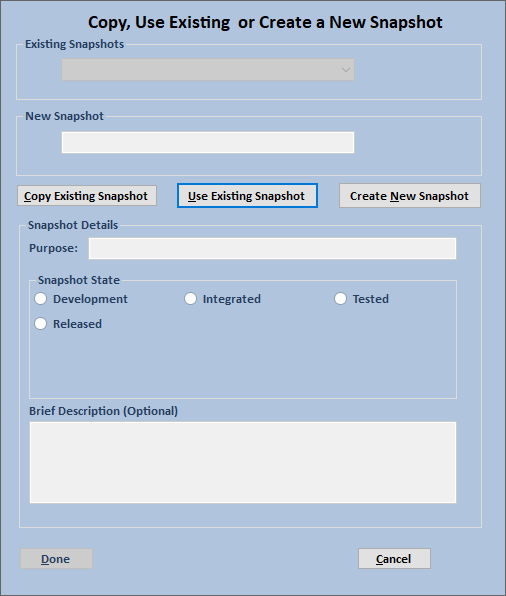
The default is yes and if yes is selected the following screen will be displayed



## Creating your first Sprint

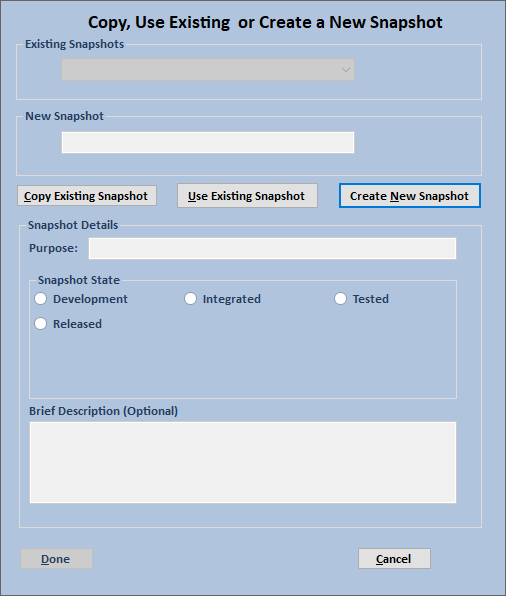


* 1. Step one is to ensure that the correct Project and Group is displayed
  2. Next enter a Definitive Sprint Name. Note there is no checks on the Sprint name but it should be meaning full to the Project / Sprint
  3. Select a Sprint Start Date and End Date. Note the End Date has to be after the Start Date and is verified when the Create button is pressed
  4. The Snapshot Named field tiles this Sprint to and existing release or can create an entirely new release to the software. When clicked the following screen will appear



## Tying a Sprint to a Release

This part of the Sprint System ties any completed story in a sprint to a release in Razor Agile which then if desired can be automatically deployed via Scripting to production.



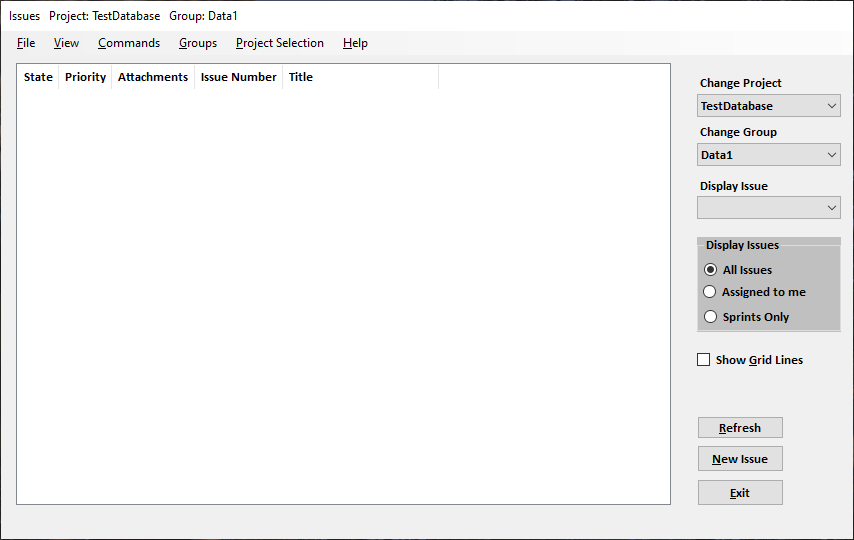
* 1. There are 3 choices that are available to the user
     1. Copy an existing snapshot. This system copies a snapshot that may be a full production release into a test release or a temporary release so the changes can be verified. It can then become the main line release if desired
     2. Use existing snapshot. This will update the current release snapshot of the product
     3. Create New Snapshot. This is used in the case where there a no snapshots to copy or use. Generally, this is a brand new product.

B) Complete the form and select done

# Issues

## The Main Issue View

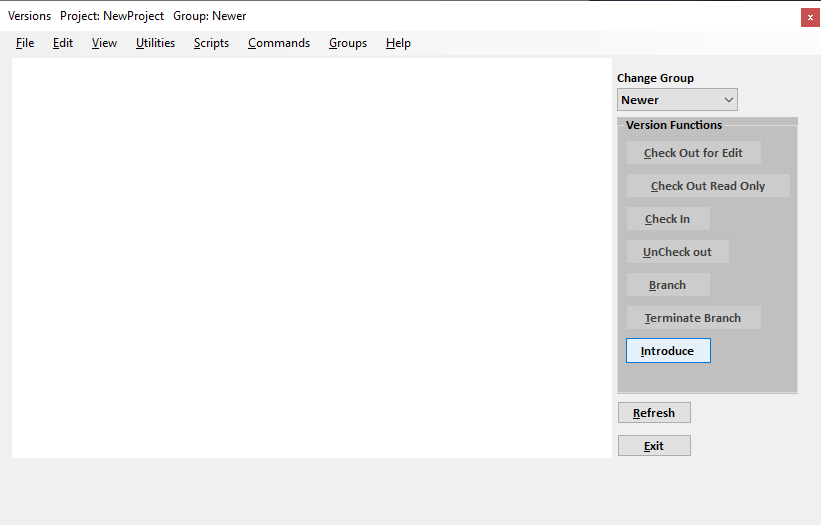
This is a place holder



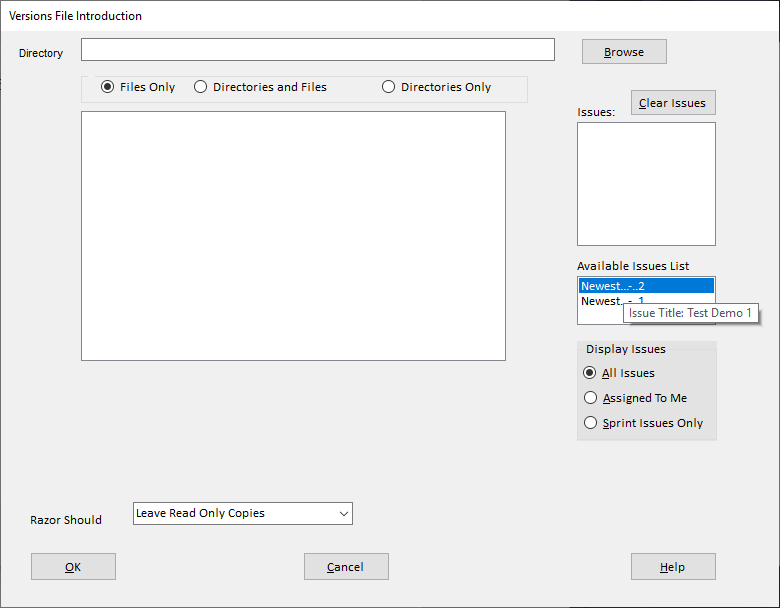
# Versions

## File Introduction

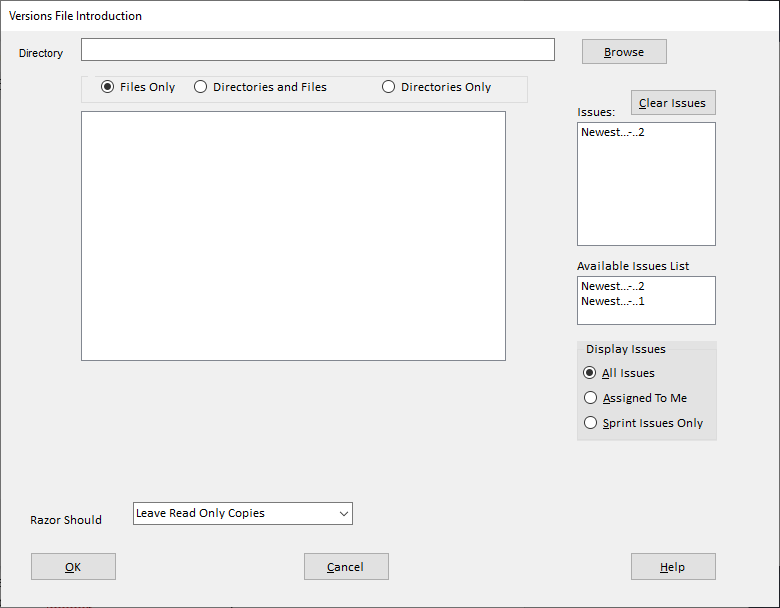
1. There are 2 methods for file introduction: Razor Classic and Drag and Drop
2. Both have similar Interfaces which have been designed to be easier to use
   1. For Classic File Introduction select the Introduce button from the Versions Window



* 1. On the right side of the Introduce window, in the box labeled available issues list you will have a list of the current Issues. By left clicking on an Issue the Issue title will appear

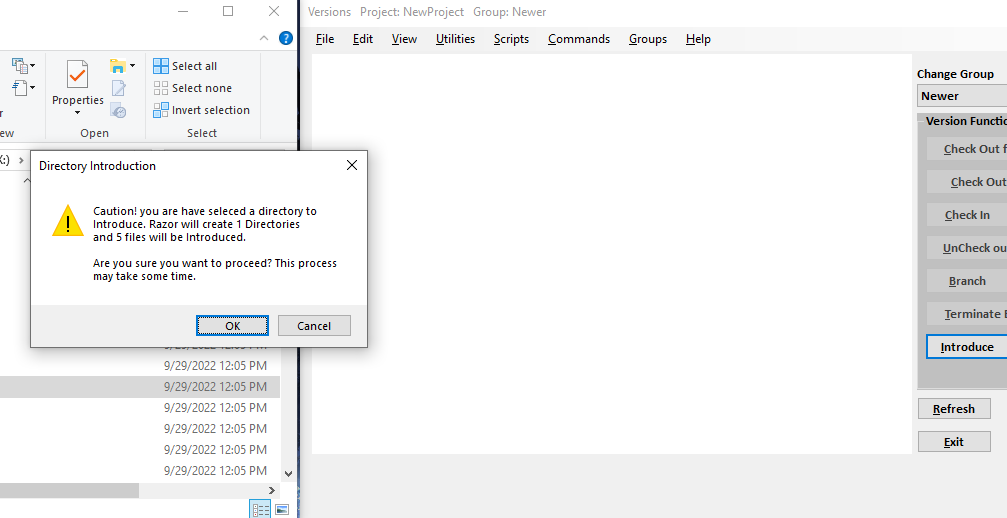


* 1. By right clicking on the selected Issue, it is transferred to the Issues list to be used for this file introduction. One can use the standard Windows feature of Alt clicking or CTL clicking to select one or many issues to be included

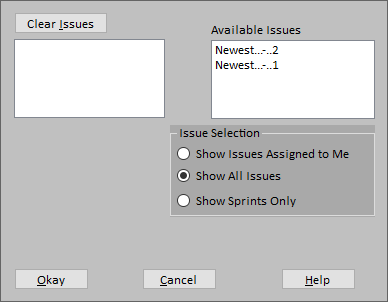


* 1. You can now proceed to complete the file introduction as needed

1. Drag and Drop File introduction
   1. Is basically the same as the standard file introduction
   2. Simply drag the file or folder that you wish to introduce onto the main Versions Window



* 1. Click on the OK button and the Issues selection Window will pop up



* 1. Select the issues as needed and right click to include them and Click Okay

## 2) File Check Out Read Only

## 3) Check Out File for Edit

## 4) Check in file or Un Check Out a File

# Scripting

## Razor Agile Server Environmental Variables

These Environmental Variables are created by the server when a script request is detected by the server. They should not be changed at anytime by any scripts unless noted below

RAZOR\_HOME Where the Binaries for the Server are located

RAZOR\_PROJECT\_DIR Where the Project directory is located

RAZOR\_GROUP What group are we working in

RAZOR\_SCRIPTS\_DIR Where the user scripts are located

RAZOR\_PRELOAD The file that contains Issue preloads

RAZOR\_TMP The Razor temp storage area

RAZOR\_PID Random PID number

RAZOR\_USER Logged in user FULL name

RAZOR\_USER\_EMAIL Logged in User Email

USER User login

TMP Unix Temp Directory

RAZOR6 Yep we be running Razor Agile

PASSED\_PARAMETERS Any passed data from the server to the scripts

PASSED\_FILE\_TO\_SCRIPTS A file passed from the server to the scripts

PASSED\_FILE\_TO\_SERVER A file from the scripts to the server

## Razor Agile API

The API for Razor Agile contains 2 types of commands. Those that do not involve the entire server engine and those that do not. The commands that require the server engine must include the keyword “console” as show in the following items

* 1. Standard commands

To execute these standard commands in the console, first start a Windows Console and then enter:

cd c:\Program Files\Visible Systems\Razor Agile Server

Once there enter the commands as listed below

* + 1. Razor6Server.exe id Returns the ID of the computer
    2. Razor6Server.exe license\_info Returns the License Status
    3. Razor6Server.exe version Returns the release version
    4. Razor6Server.exe who Returns who is logged in
    5. razor.exe rz\_time Returns the Integer timestamp (now)
    6. Razor6Server.exe rz\_fmt\_time Returns the formatted Date time based on integer timestamp
    7. Razor6Server.exe rz\_time\_val: Returns the Integer timestamp based on the input time string
  1. Server Interactive Commands (more to follow)
     1. Promote\_issue Promotes the issue to the passed state
     2. Issue\_submit Submits a new issue
     3. Modify\_issue Modifies or updates an existing issue
     4. Get\_activity Gets the activity associated to an Issue(s)
     5. Check\_out\_readonly
     6. File\_check\_out
     7. File\_check\_in
     8. unCheck\_out

## Standard Command Usage

To execute these standard commands in the console, first start a Windows Console and then enter:

* 1. cd c:\Program Files\Visible Systems\Razor Agile Server
  2. Enter the desired command in as follows

Razor6Server.exe id

* 1. The command will display the output on the console.
  2. To run the command via a script see the next section on Interactive Commands

## Scripted Command Usage

Ruining Server request from a Script no matter what type the user must inform the server that it is to run the command in “console” mode. This is because the Razor Agile Server is running as a service and services cannot normally be accessed from a command line. However, by using the past parameter when calling a RazorService command the server will execute the command and return the results to the Script.

Something to note here is that the error paths return a robust parcel of information back to the script. It may make it more difficult to test for this information, however it will indeed speed up development of a script as the programmer will know what is amiss in their script.

One last thing to note. The system\_setup and rz\_utils scripts supplied by Visible Systems Corporation and installed when the server is installed are for the most part complete and tested.

But if while developing scripts and you are required to update any of these, please send the updates to razor\_support@visible-systems,com so we can include them in our next release. The install program will overwrite these locations.

## Interactive Command Usage

# Appendix

Usage:

Preload File Usage Example:

Note: The Attribute (Originator) and the Result (RAZOR\_USER) are separated

by a single tab

echo "Originator<TAB>$RAZOR\_USER" >> $RAZOR\_PRELOAD

Passed Parameters Usage Example:

This is for 2 passed parameters in a comma separated list.

(PassedParam1,PassedParam2)

if [[ -z "${1}" ]]

then

# Do something based on nothing

echo "Nothing passed in"

else

# This will assume that only 2 parameters are passed.

# It provides a simple method for getting the goodies

passedVar1=`echo "${1}" | "${cutexec}" -d ',' -f 1`

passedVar2=`echo "${1}" | "${cutexec}" -d ',' -f 2`

fi

Format for passed result is as follows:

$updateThisAttribute,$item1,$item2,$item3

Where $updateThisAttribute is the item that is to be updated within Razor followed by a comma

and the items values to be used for the update separated by commas

echo $result

Passed File from server to scripts usage Example

This file will contain a formatted Issue, full file from versions or Baseline output

It can be parsed within the external script as any normal file would be

# $1 is the file path to the passed file, in this case the Issue

# The fourth item in the file is the issue name which we need to

# Get so this is one way to do it

ISSUE=`"${sedexec}" -n '4p' $1`