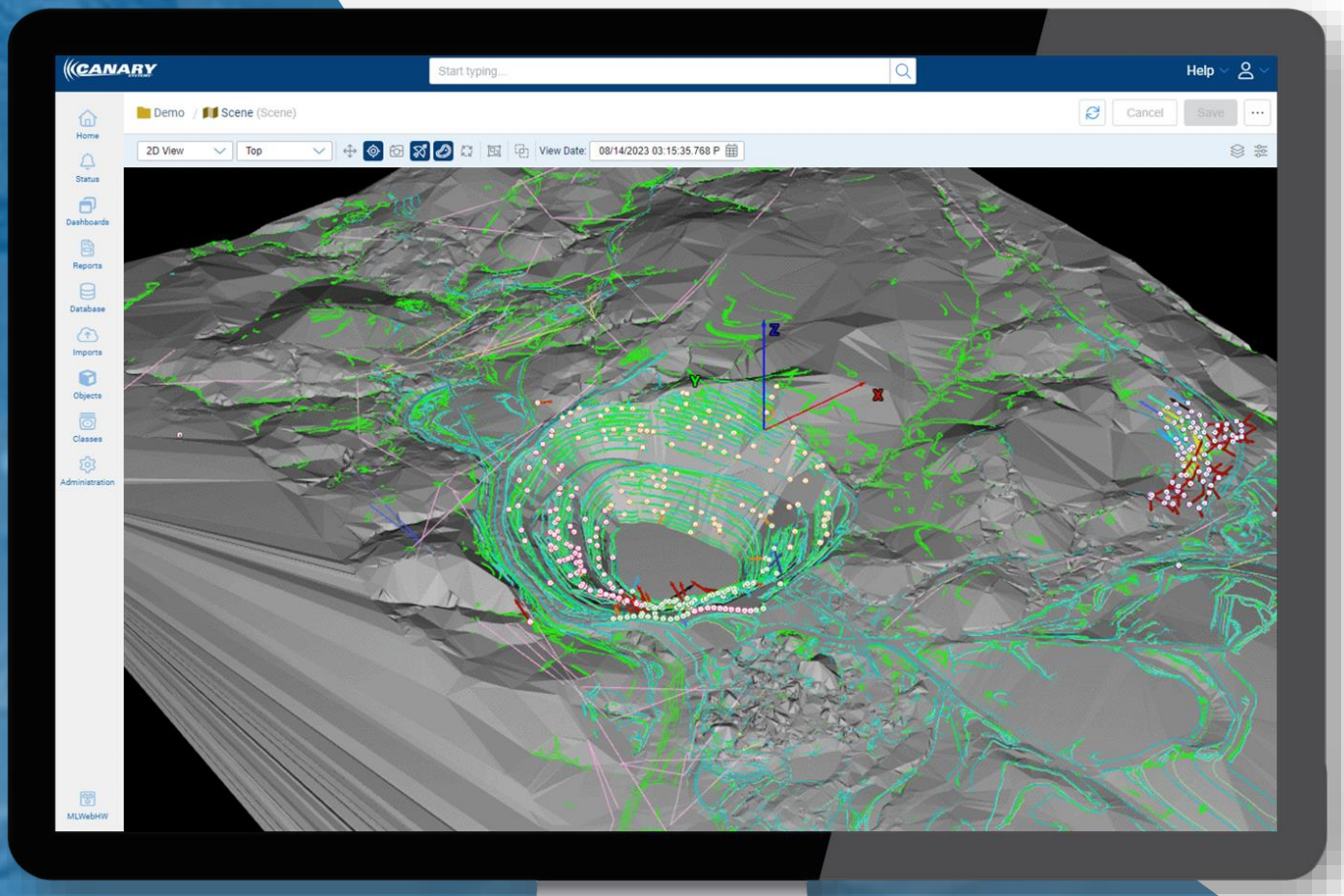
MLSuite® 2023 SP2

Installation Guide

Released January 31, 2024
Revision C



Corporate Office

Canary Systems, Inc.
5 Gould Road
New London, NH 03257
USA
1.603.526.9800
canarysystems.comxa

Copyright and Trademark

End-User License Agreement

By installing the MLSuite software, you agree to the Canary Systems, Inc. End-User License Agreement. The Canary Systems End-Users License Agreement can be found on the Canary Systems website under Software Downloads at canarysystems.com/support/software-downloads

High Risk Activities

Products of Canary Systems are not fault-tolerant and are not designed, manufactured or intended for use as on-line control equipment in hazardous environments requiring fail-safe performance, such as in the operation of nuclear facilities, aircraft navigation or communication systems, air traffic control, direct life support machines, or weapons systems, in which the failure of the products could lead directly to death, personal injury, or severe physical or environmental damage ("High Risk Activities"). Canary Systems, Inc. and its suppliers specifically disclaim any express or implied warranty of fitness for High Risk Activities.

Disclaimer

The following document is provided to assist users with the installation, operation and training in the use of our products. This document and our products are intended to be used by technically qualified personnel. Contained herein is information that is proprietary to Canary Systems® and may not be reproduced or copied in any form, nor disclosed to outside parties by any means whether directly or indirectly, without the written consent of Canary Systems. This document is subject to change without notice and Canary Systems assumes no responsibility for errors, omissions or misinterpretation. Furthermore, Canary Systems makes no warranty as to the suitability of this information and/or products for any given application or use.

Documentation and Release Notes

To obtain the most recent version of all Canary Systems® technical documentation, visit the User's Guides section of our website at **canarysystems.com/support/users-guides**.

Additionally, Canary Systems provides additional information beyond the scope of the technical documentation in the form of application notes, training videos and white papers. Visit the Support section of our website at **canarysystems.com/support** to view application notes, FAQ's and videos, and the Papers section of our website at **canarysystems.com/projects/papers** to view a collection of white papers.

Revison History

Date	Rev.	Comments
10/12/2023	A	Initial Release
12/21/2023	B	Firewall Configuration section amended for accuracy/clarity, Azure Active Directory section added
1/31/2024	C	Advanced MLServer .ini Settings appendix added

How To Use This Guide

When you see a keyboard key's name in bold angle brackets (**< >**), this indicates a particular keystroke you must make in order to perform certain functions.

When you see two bolded locations separated by a vertical bar character (**Location 1 | Location 2**), this indicates that the second location can be accessed only after the first. For example, an Import option that is available in a File drop-down menu in a toolbar would be denoted with **File | Import**.

References to other Canary Systems user's guides, third-party documentation or other external sources will appear bolder and italicized. For example: For more information, see the ***MultiLogger User's Guide***.

Notes & Hints

Notes and Hints appear in boxes like this. **Notes** contain information you need to know to ensure that you use certain features correctly, minimizing any errors. **Hints** indicate tips on how to use certain features more effectively, such as any shortcuts to certain functions.

Related Documentation

Visit the User's Guide page on the Canary Systems website to view and download the most current versions of our technical documentation and user's guides at **canarysystems.com/support/users-guides**.

The following documents contain information that may be useful throughout the course of this manual:

- ***MLWebHardware User's Guide*** for additional information on data collection and import.
- ***MLWeb User's Guide*** for additional information on data visualization and presentation.
- ***MLReport User's Guide*** for additional information on using MLReport.
- The ***Canary Systems End-Users License Agreement*** can be found on the Canary Systems website under Software Downloads at **canarysystems.com/support/software-downloads**.

Technical Support

Canary Systems may be contacted directly via phone or email. Ongoing support via phone, email, and Microsoft Teams,® as well as in-person and field support, is available through the purchase of a support contract. Please contact Canary Systems® directly for more information.

Note

To contact Canary Systems for assistance or regarding the purchase of a support contract, you can reach the Support Team at:

+1.603.525.9800 Ext. 2
support@canarysystems.com

Our website also provides numerous Application Notes, the latest versions of our software components, and all the latest User's Guides. Click the **Support** menu at **canarysystems.com** to access these resources.

Table of Contents

1. Getting Started

Note

As the 3rd generation of the platform, the introduction of MLWeb 2023 to Canary Systems MLSuite has necessitated significant upgrades and changes to the installation processes and functionality of several applications. **Legacy MLWeb users (2021 and previous) should take special care to review section *1.2 Considerations for Legacy MLWeb Users* before moving forward with installation.**

Canary Systems® MLSuite is a collection of powerful desktop, server, web, and mobile applications for Data Acquisition. MLSuite, also referred to MLSuite in this guide, consists of a group of applications, two database servers (Firebird SQL and PostgreSQL), and a web client that operates under the Microsoft .NET Core framework. These various software components may be installed and run on a single server or distributed throughout a LAN/WAN connected group of servers, depending on deployment size (see section *1.1 System Requirements*).

MLSuite is made up of the following applications:

MLServer – The main Windows Service that provides all business logic methods to MLWeb and MLField.

MLGateway – Windows Service that manages connected dataloggers, including automating programming and data collection.

MultiLogger – Windows client application for MLGateway. It provides for configuring the programming and data collection automation of MLGateway.

MLWebHardware – Web based application, used for the configuration and programming of data collection automation of MLGateway. This will eventually replace MultiLogger.

MLWeb – Data management application that provides a browser-based interface to the system. With introduction of this version, data imports are now initiated through this interface.

MLField – Mobile application that bridges the gap between data collection in the field and the project database.

MLReport – A robust report creation tool that can embed data and charts.

In addition, MLSuite includes several support applications and files:

MLAdmin – The MLWeb administration page.

MLEditor – A specialized editor for MultiLogger programming files.

Firebird – The Firebird SQL database.

PostgreSQL – The PostgreSQL database with PostGIS extension.

MLPerfIndex – A tool for optimizing server performance (optional).

IBOConsole – A Firebird database management console (optional).

Icons – Installs a library of icons for data visualization (optional).

PGAdmin – A PostgreSQL database management console (optional).

This Installation Guide provides information on running the MLSuite installer, PostgreSQL and Add-ons, .Net Core, configuring MLAdmin, and uninstalling MLSuite.

1.1 System Requirements

MLSuite is a scalable suite of applications that can be used for one-user one-datalogger deployments all the way to enterprise-wide projects managing tens of thousands of sensors and many users. System Requirements scale along with the project size and scope, with the number of users, frequency and volume of data imports and reports, deployment of 3D and GIS components, and other factors.

The basic system requirements to run the software are listed below. From there, server specifications can vary widely. This section outlines four recommended deployments of increasing scale to be used as guidelines for optimal expected performance of the system.

Note

Due to the high input and output (I/O) requirements of the Firebird and Postgres databases, **the use of virtual machines or NAS storage is not recommended** as virtual servers may lose between 20% and 80% of their speed under a high load.

It is also strongly recommended that solid-state drives (SSD) are used as these allow for significantly more IOPS (up to 500,000 on PCIe-based hardware).

An SSD built on a dedicated database server, which is not virtualized, is the optimal solution for maximum performance for anything more than the most basic deployments.

1.1.1 Basic Requirements

Supported Operating Systems

- MLSuite is designed to run on Windows Server 2016, Windows Server 2019, Windows Server 2022, Windows 10, and Windows 11 operating systems.

Other Minimum Requirements - Server

- MLWeb is designed to run on the ASP .NET Core 7.0 Windows Hosting Bundle.
- 1 GB LAN required
- NAS/CIFS storage is not supported due to high latency of I/O operations
- 3.5 GHz CPUs minimum

Other Minimum Requirements - Clients

- Chromium based browser, such as Chrome or Edge
- Graphics Processor: GPU 1+ GB

1.1.2 Recommended Server Specifications

Minimal Environment

Intended primarily for testing, trial, or backup purposes, or for very small environments with only 1-2 users, limited data imports, and limited 3D data processing.

- Operating System: Windows 10 or Windows 11
- Processor: 3.5 GHz per core, 4 physical cores
- Memory: 32 GB
- Hard Drive: 1 TB SSD

Small Environment

Single server deployment running all MLSuite components for small projects with 1-4 concurrent users and limited 3D data processing requirements.

- Operating System: Windows Server 2016, Windows Server 2019, or Windows Server 2022
- Processor: 3.5 GHz per core, 8 physical cores
- Memory: 32 GB
- Hard Drive 1: 500 GB SSD for Operating System, Program Files, backups, and archives
- Hard Drive 2: 250 GB SSD dedicated to Firebird and PostgreSQL databases
- Hard Drive 3: 250 GB SSD for Page file and Temp folder

Medium Environment

> **Note**
>
> Contact Canary Systems Support for assistance with tiered server setups.

Two server deployment, one dedicated to MLWeb, the second for remaining components. This setup will satisfy the most common project requirements for average size projects including medium sized, multi-user environments, and use of 3D data with infrequent processing.

- Operating Systems (each server): Windows Server 2016, Windows Server 2019, or Windows Server 2022
- Server 1 (running MLGateway, MLWeb, and MultiLogger)
  - Processors : 3 GHz per core, 4 physical cores
  - Memory: 32GB
  - Hard Drive 1: 250 GB SSD with 2 partitions
    - Partition 1 - 100 GB: Operating System
    - Partition 2 - 150 GB: Program Files
  - Hard Drive 2: 250 GB SSD for Page file and Temp folder
- Server 2 (running Firebird, Postgres, MLServer, and MLReport)
  - Processors : 3.5 GHz per core, 8 physical cores
  - Memory: 32GB
  - Hard Drive 1: 500 GB SSD with 2 partitions
    - Partition 1 - 100 GB: Operating System
    - Partition 2 - 400 GB: Program Files, backups, archives
  - Hard Drive 2: 250 GB SSD dedicated to Firebird and PostgreSQL databases
  - Hard Drive 3: 250 GB SSD for Page file and Temp folder

Large Environment

Three server deployment for enterprise sized projects including multi-user access and frequent 3D data processing (such as regular radar data imports).

- Operating Systems (each server): Windows Server 2016, Windows Server 2019, or Windows Server 2022
- Processors (each server): 3.5 GHz per core, 8 physical cores
- Memory (each server): 32 GB
- Server 1 (running MLGateway, MLWeb, and MultiLogger)
  - Hard Drive 1: 250 GB SSD with 2 partitions
    - Partition 1 - 100 GB: Operating System
    - Partition 2 - 150 GB: Program Files
  - Hard Drive 2: 250 GB SSD for Page file and Temp folder
- Server 2 (running PostgreSQL)
  - Hard Drive 1: 500 GB SSD with 2 partitions
    - Partition 1 - 100 GB: Operating System
    - Partition 2 - 400 GB: Program Files, backups, archives
  - Hard Drive 2: 250 GB SSD dedicated to PostgreSQL databases
  - Hard Drive 3: 250 GB SSD for Page file and Temp folder
- Server 3 (running Firebird, Postgres, MLServer, and MLReport)
  - Hard Drive 1: 500 GB SSD with 2 partitions
    - Partition 1 - 100 GB: Operating System
    - Partition 2 - 400 GB: Program Files, backups, archives
  - Hard Drive 2: 250 GB SSD dedicated to Firebird databases
  - Hard Drive 3: 250 GB SSD for Page file and Temp folder

Note

Those installing MLSuite on a system for the first time may skip ahead to section *1.3 Pre-Installation Requirements*.

1.2 Considerations for Legacy MLWeb Users

MLWeb 2023 has been enhanced in numerous critical ways which make it incompatible with previous Canary Systems browser-based applications and associated components. Users maintaining databases in legacy MLWeb should carefully review the below considerations and/or **Appendix C – Example Migration from MLWeb 2021 to MLWeb 2023** of this guide to ensure the most efficient installation of MLSuite with MLWeb 2023.

1. It is recommended users backup existing database files. See **Appendix C**, section *C.1 Backing Up Existing Database Files*.
2. Previous versions of MLSuite and all associated components must be uninstalled. See **Appendix C**, section *C.2 Uninstall MLSuite and Components.*
3. As MLWeb 2023 is no longer configured for 2D scenes, users must adjust system settings for the viewing of 3D scenes. See **Appendix C**, section *C.3 Settings for 3D Scenes*.
4. Certain database items from legacy databases must be adjusted or rebuilt when coming into MLWeb 2023. See **Appendix C**, section *C.4 Migrating Reports, Charts, and Scenes.*

For users with a legacy version of MLSuite on their systems, it is important to complete all of these steps before moving onto the next section of this guide.

Note

Previous generations of MLSuite have required the installation and maintenance of IIS to enable the MLWeb web server. IIS is no longer required as MLWeb 2023 is powered and managed by Microsoft ASP .NET Core Runtime via the .NET framework.

1.3 Pre-Installation Requirements

Several steps must be completed before the installation of MLSuite can begin. Note that these steps apply to full version installations and upgrades, not service pack upgrades.

Required Conditions

- The Windows platform must be fully patched and updated with Microsoft Windows updates. Not having the proper updates will prevent installation of components crucial to the operation of MLSuite.
  - This includes the .NET framework: Version 4.8 must be installed on the computer or server.
  - Windows Firewall must also be enabled. If Windows Firewall is disabled, the installation will fail.
- PostgreSQL must be installed. See section *2.2 Installing PostgreSQL Installer.*
- .NET Core 7.0 Web Service Hosting must be installed. See section *2.3 Installing .NET Core Web Server Hosting.*
- NodeJS must be installed. See section *2.4 Installing NodeJS.*

Additional Pre-Installation Considerations

- If the **firebird.conf** file (**C:\Program Files\MultiLogger\Firebird**) has been edited, the installer will not overwrite the .conf file during installation. To revert an edited file to the default configuration, it is recommended that the firebird.conf file be backed up and deleted before installation of the latest version.

- It is recommended to stop the MLGateway, MLServer, and Firebird services before uninstalling, especially if uninstalling versions of MLSuite older than 2016. Stop MLServer **before** Firebird.
- Folders designated for Firebird and Postgre database files should be created before beginning the installation.
  - Example: C:\Databases\Firebird and C:\Databases\PostgreSQL
- The installer should restart the services automatically, but it is recommended to check that they are running after the installation is complete and the machine has been rebooted. If not, cycle them to ensure that they are running. Start Firebird **before** MLServer.
- An account used by the software to run backend processes should be created. See *Appendix D – Creating Canary Process Account* for instructions on creating a Canary Process Account.

2. Installation

> **Note**
>
> The following example installation shows a new installation of MLSuite 2023 on a 64-bit Windows environment with Admin user rights, unless otherwise specified. The installation process on the operating system may vary (particularly for file locations), especially with desktop operating systems such as Windows 11.

MLSuite with MLWeb 2023 software is available in two formats: a USB flash drive or from a downloaded installer (.msi). The current MLSuite installer is available from the support area of the Canary Systems® website at canarysystems.com/support/software-downloads.

The basics steps for the installation process are as follows:

- Install and configure ***PostgreSQL*** and add-ons – See *2.2 Installing PostgreSQL Installer*
- Install ***.NET Core Web Server Hosting*** – See *2.3 Installing .NET Core Web Server Hosting*
- Install ***NodeJS*** – *See 2.4 Installing NodeJS.*
- Install ***MLSuite*** – See *2.5 Installing MLSuite*

> **Note**
>
> Previous editions of the MLSuite installer have required the manual installation of Microsoft Visual C++ components as a preliminary step. These components are now included with the installation of PostgreSQL and add-ons.

2.1 Installing PostgreSQL Installer

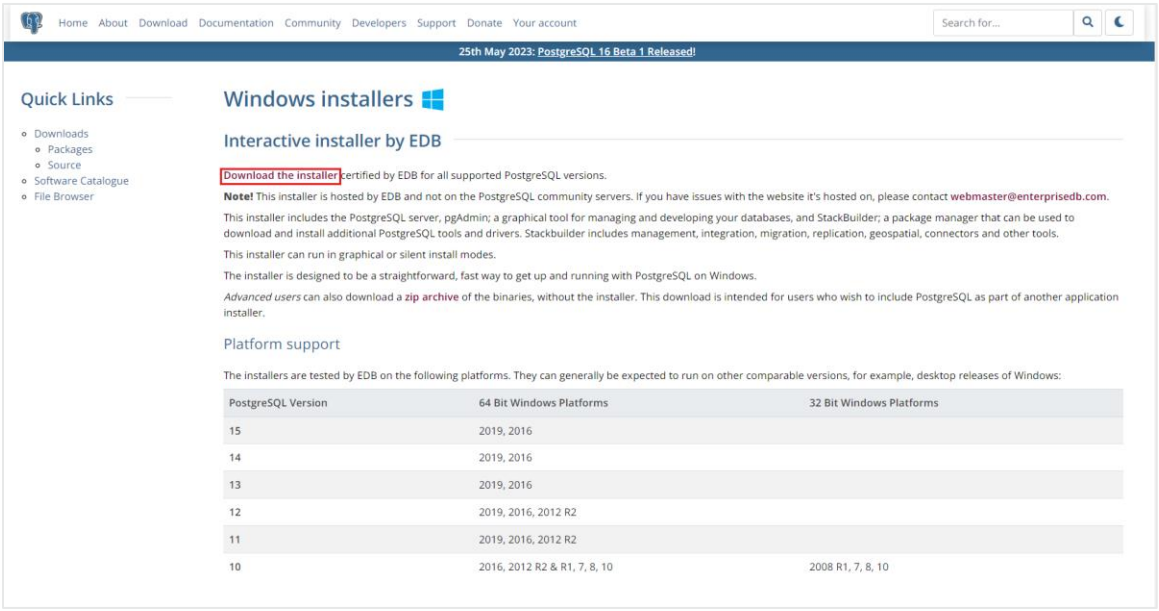
PostgreSQL is a cross-platform database management tool used by MLSuite for storing and processing 3D data. It is recommended that PostgreSQL, along with the Firebird SQL server, be kept on a separate, dedicated database SSD, or even a separate machine from that which MLSuite is installed on. Contact Canary Systems for help determining the optimal setup for your project.

> **Note**
>
> In order to install PostgreSQL, the user must have Admin user rights on Windows.

2.1.1 Downloading the PostgreSQL Installer

The PostgreSQL installer for Windows is available online and can be found on the Canary Systems website under Third Party Downloads & Utilities (canarysystems.com/support/software-downloads).

Click **Download the installer** for a list of available PostgreSQL Versions.



Select the latest version 13 of PostgreSQL and download the installer for the correct operating system.

Note

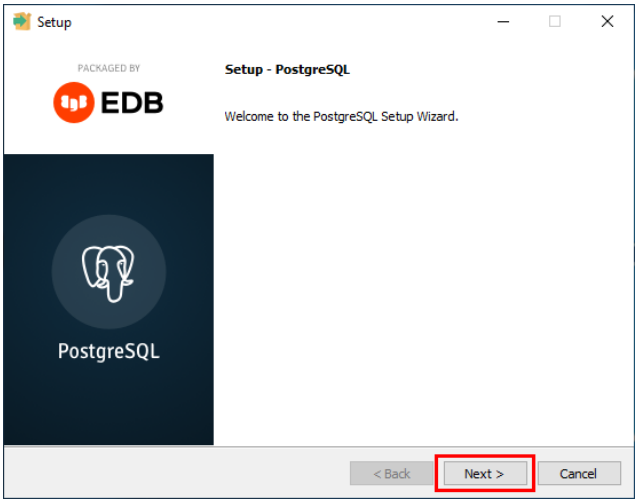
Make sure to install the latest version of 13 as newer versions of the PostgreSQL may not yet be compatible with the MLSuite software.

2.1.2 PostgreSQL and Add-on Installation

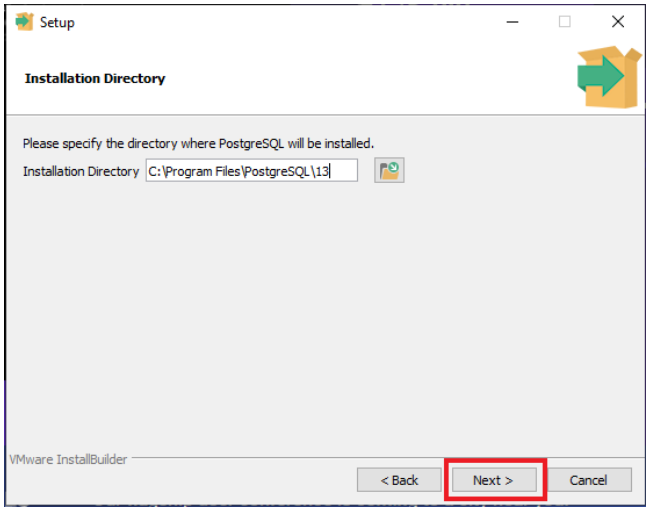
There are several steps required during the installation of PostgreSQL, including customizing settings and enabling add-ons.

PostgreSQL Installation Steps

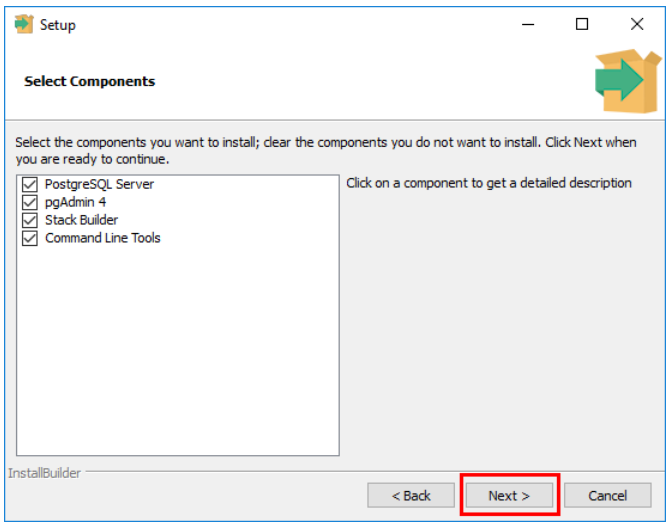
1. Run the PostgreSQL installer.
2. On the initial page, click **Next**.



3. The **Setup** wizard displays the default **Installation Directory** path. The default location can be left, or an alternative drive can be selected. Click **Next**.



4. Ensure that all components are checked on the **Select Components** page. Click **Next**.



Hint

The Stack Builder option requires internet access. If the server the installer is being run on does not have internet access, contact Canary Systems.

5. Update the **Data Directory** location. The Data Directory specifies the path where all **PostgreSQL** data is stored.

If a PostgreSQL database folder does not already exist, one should be created: **Create a new "Databases" folder with a "PostgreSQL" subfolder**.

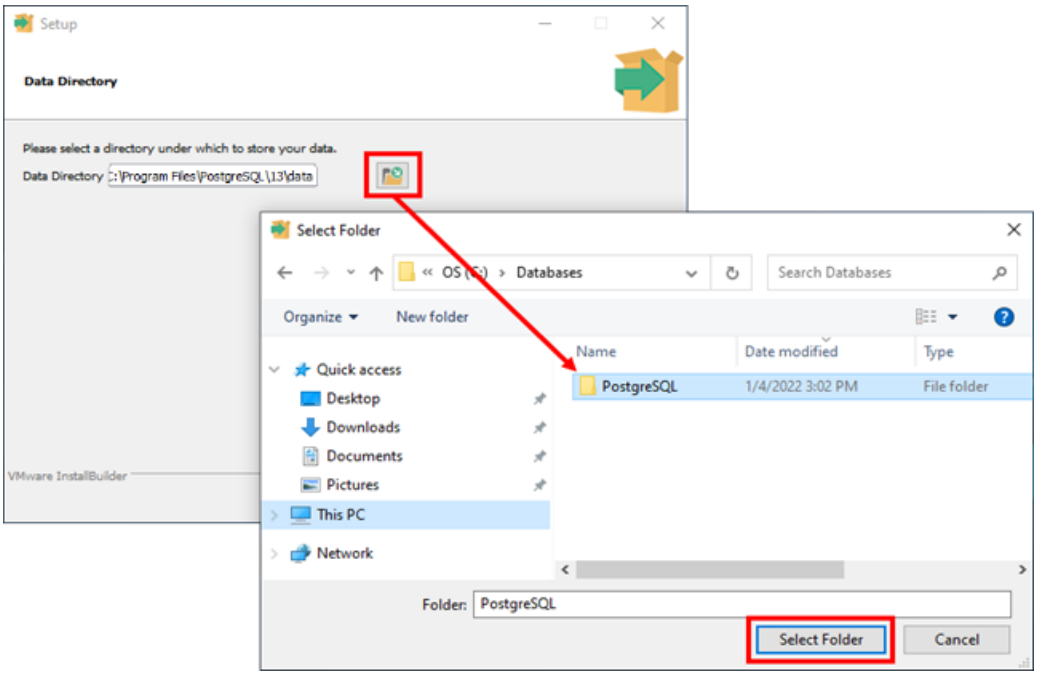
Note

It is highly recommended that PostgreSQL data is stored on a separate, dedicated database SSD.

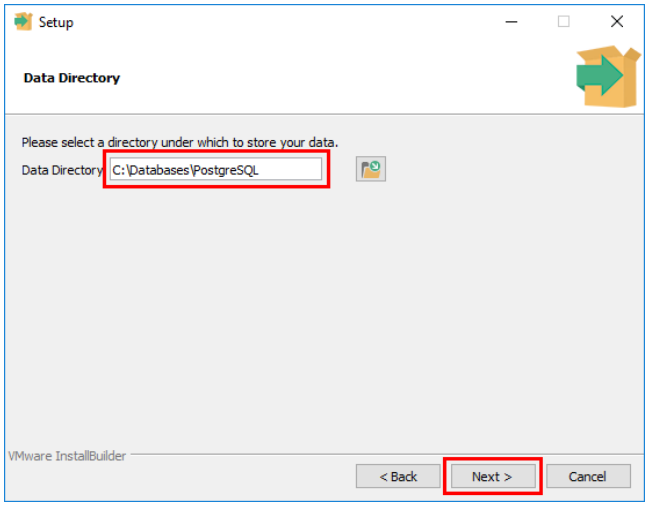
a) Click the Browse icon to select the directory for PostgreSQL data

b) Select the PostgreSQL folder

c) Click Select Folder



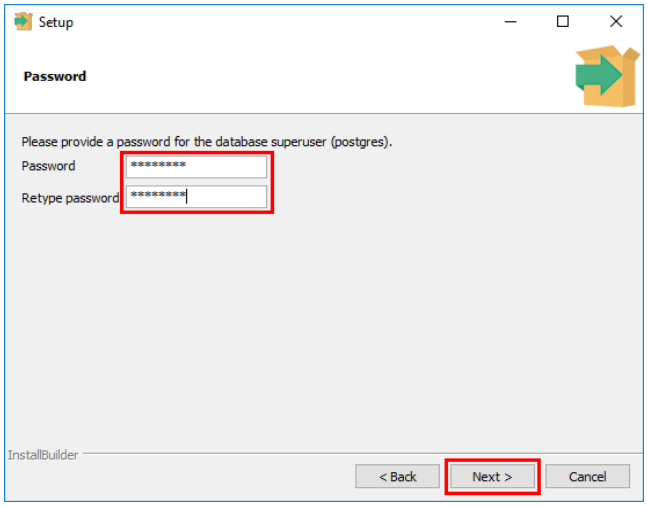
6. After the new directory has been selected, click **Next**.



7. On the **Password** page of the Setup wizard, the postgres superuser password must be set. Enter a password and click **Next**.

Note

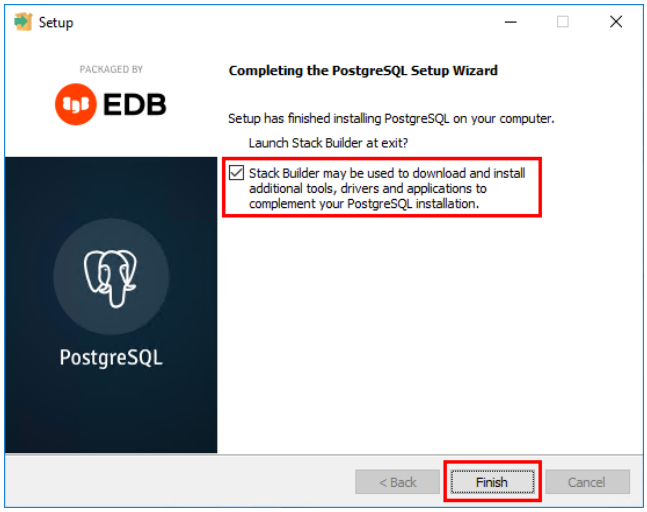
Take note of this password, as it is needed later in the installation process.



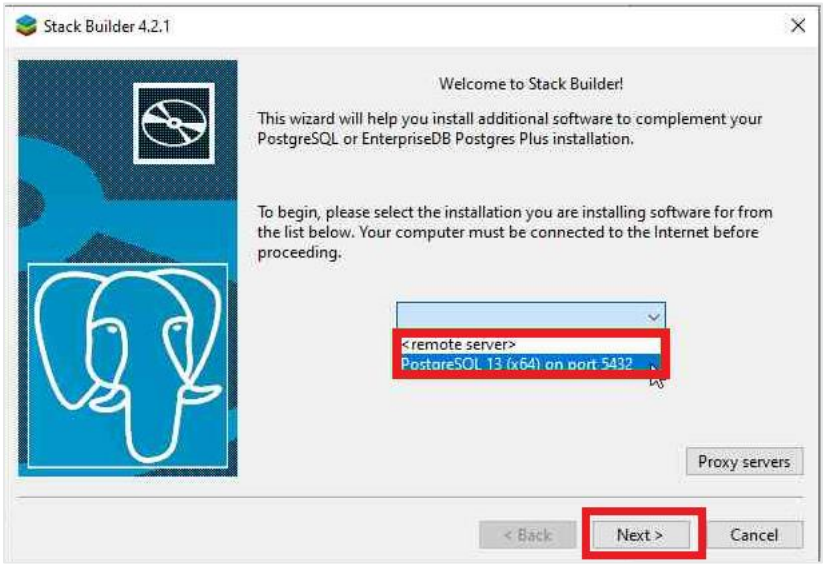
8. Keep the default **Port** setting and click **Next**.
9. Keep the default **Locale** setting and click **Next**.
10. Click **Next** on the **Pre-Installation Summary** page.
11. Click **Next** on the **Ready to Install** page to begin the PostgreSQL installation. This may take a few minutes to complete the installation.

Stack Builder Component Selection and Setup

1. When the installation is complete, the Setup wizard opens a dialog box where **Stack Builder**, which allows for additional third-party PostgreSQL plugins such as PostGIS, can be enabled or disabled. Ensure that it is **enabled** (checkbox is checked), then click **Finish** to open the Stack Builder wizard.



2. Select the **PostgreSQL** database and port from the **Stack Builder** drop-down and click **Next**.

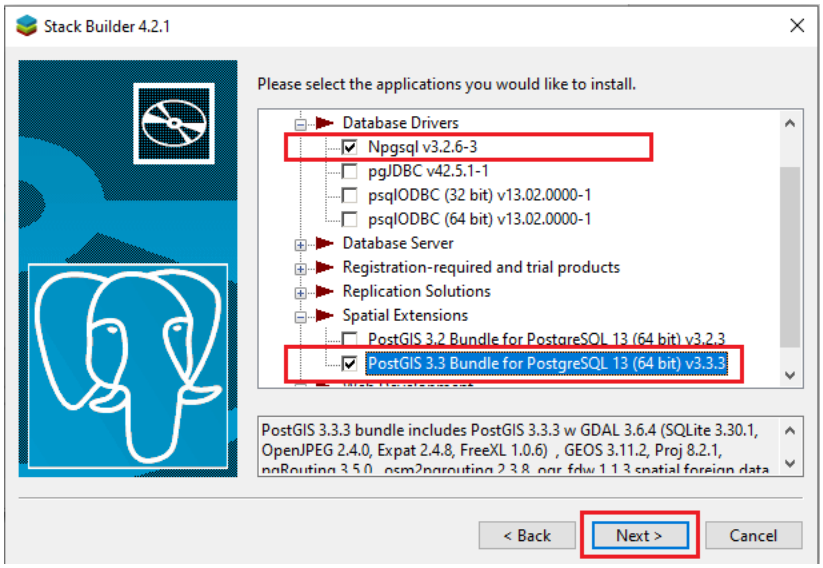


3. From the list of available applications:

   a) Expand the **Database Drivers** group and select **Npgsql**, this allows MLWeb and .NET services to connect to the PostgreSQL databases.

   b) Then expand the **Spatial Extensions group** and select the **PostGIS 3.3 Bundle** option.

   c) Click **Next**.



4. On the next page, the default path for the **Download directory** should not be changed. Click **Next**.

Note

If at this stage you get a Checksum verification failed error, complete the following remedial steps:

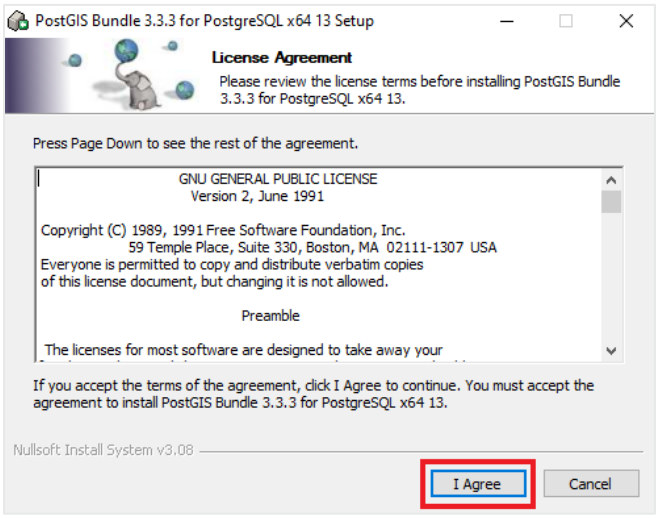
1. Click **OK** to close the Stack Builder Error window.
2. From the Stack Builder wizard, click **< Back** to return to the previous window.
3. Deselect the option for the PostGIS 3.3 Bundle, leaving only Npgsql v3.2.6-3 selected. **Click Next >**.
4. Click **Next >** again. This brings you to step 5 of the main list. Complete steps 5-7 to finish setup of the Npgsql component.
5. Once complete, go to https://ftp.postgresql.org/pub/postgis/pg13/v3.3.3/win64/postgis-bundle-pg13x64-setup-3.3.3-1.exe to download the Postgis bundle directly. Open and then start from PostGIS Installation Step 1 of this guide.

5. Once the applications download, click **Next** on this page and the following page to continue setup of the applications. **Do not check Skip Installation**.
6. The first component to be configured is **Npgsql**. The default Installation Directory should be used, click through and finish setup of Npgsql.
7. Click **Finish** to close the setup dialog and open the PostGIS installer.

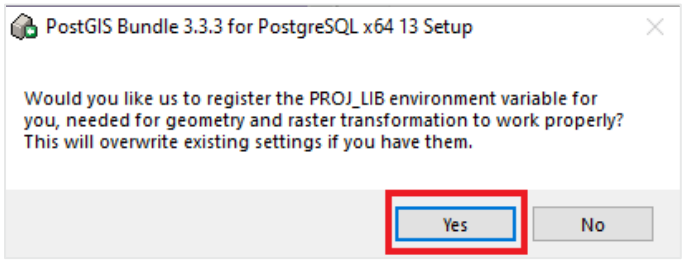
PostGIS Installation

PostGIS is a program that adds enhanced support for spatial queries to the PostgreSQL object-relational database.

1. Read through the License Agreement and click **I Agree** to continue with the installation.



2. On the **Choose Components** page, use the default settings (only PostGIS is checked) and click **Next**.
3. Leave the default **Destination Folder** path as the **Choose Install Location** and click **Next** to install PostGIS.
4. After installation, multiple dialog boxes display asking to register the environment variables, set PostGIS enabled drivers to common drivers, and enable out of db rasters. Click **Yes** for all.



5. After clicking **Yes** on the last dialog box, a message indicating the installation is complete is displayed. Click **Close** and then click **Finish**.

PostgreSQL and add-ons have been successfully installed. PostgreSQL must now be configured. See the following section for configuration details.

2.1.3 Configuring PostgreSQL

To optimize performance, PostgreSQL should be configured for the specific operating system and hardware it is installed on. Tools available online can provide optimal settings for PostgreSQL for the system it is installed on. These settings can be updated in the **postgresql.auto.conf** configuration file.

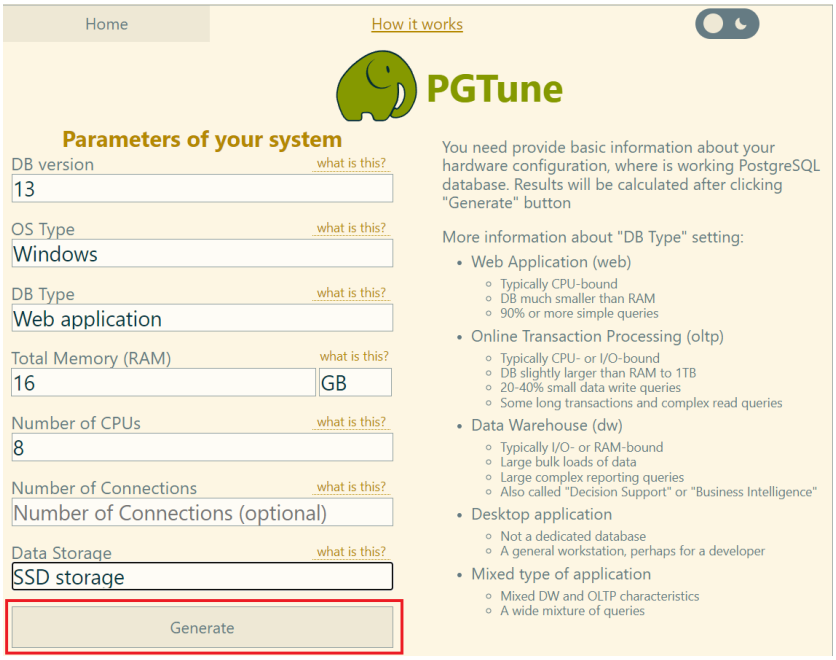
Generating Configurations

Canary Systems recommends ***PGTune*** for PostgreSQL configurations. The PGTune link is available on the Canary Systems website under Third Party Downloads & Utilities: canarysystems.com/support/software-downloads

On the left of the PGTune website, the system parameters are entered:

- **DB Version** – The version of PostgreSQL installed.
- **OS Type** – The machine's operating system.
- **DB Type** – The application of the PostgreSQL database.
- **Total Memory (RAM)** – Maximum memory available for PostgreSQL. (This should not exceed 16)
- **Number of CPUs** – Number of cores, including virtual cores.
- **Number of Connections** – Optional, usually left blank.
- **Hard drive type** – Drive type.

After entering your system parameters, press the **Generate** button.



The configurations are then generated and displayed to the right as either **postgresql.conf** configurations or **ALTER SYSTEM** configurations.

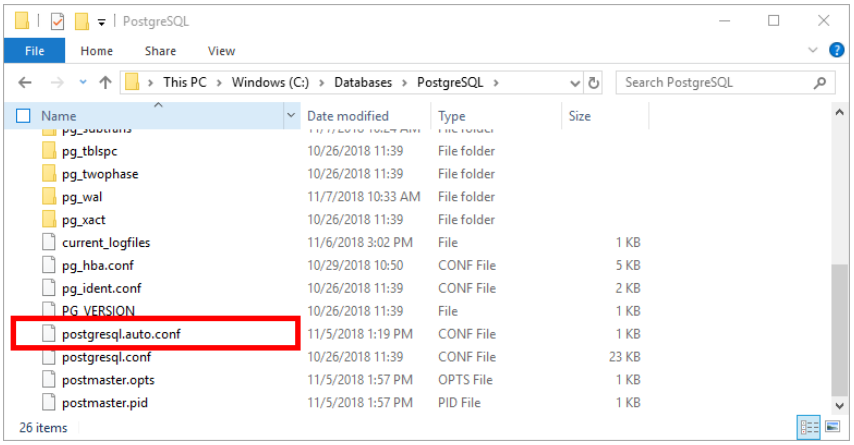
Select **postgresql.conf** and **Copy configuration**.

Updating the PostgreSQL Auto Configuration File

The PostgreSQL files are located in the **Data Directory**, the path specified during the PostgreSQL installation, refer to step 5 in *2.2.2 PostgreSQL and Add-On Installation*.

There are two configuration files for PostgreSQL; **postgresql.auto.conf** and **postgresql.conf**.

It is recommended that you edit the **postgresql.auto.conf** which is typically empty. In the case where there are conflicting entries in both **postgresql.conf** and **postgresql.auto.conf** the value in the **postgresql.auto.conf** take precedence.



1. Open the file using a simple text editor (Canary Systems recommends Notepad++).

Note

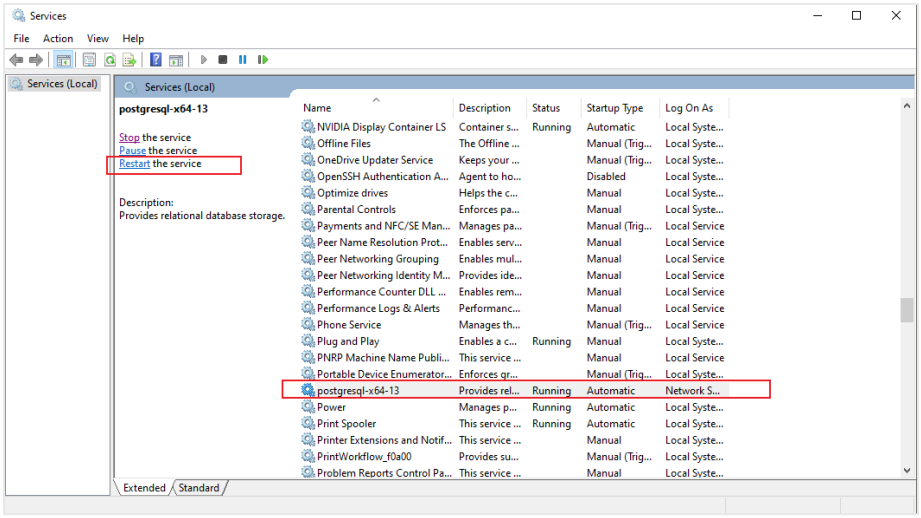
There is verbiage at the top of the postgresql.auto.conf file that reads, “Do not edit this file manually!” It is safe to ignore this message and proceed.

2. Paste the configuration generated by PGTune into the **postgresql.auto.conf** file as shown in the figure below.

```
# Do not edit this file manually!
# It will be overwritten by the ALTER SYSTEM command.

# DB Version: 13
# OS Type: windows
# DB Type: web
# Total Memory (RAM): 16 GB
# CPUs num: 8
# Data Storage: ssd

max_connections = 200
shared_buffers = 4GB
effective_cache_size = 12GB
maintenance_work_mem = 1GB
checkpoint_completion_target = 0.9
wal_buffers = 16MB
default_statistics_target = 100
random_page_cost = 1.1
work_mem = 5242kB
min_wal_size = 1GB
max_wal_size = 4GB
max_worker_processes = 8
max_parallel_workers_per_gather = 4
max_parallel_workers = 8
max_parallel_maintenance_workers = 4
```

3. **Save** and close the posgresql.auto.conf file.

4. Open **Windows Services** and **Restart** the PostgreSQL service.



If the Restart fails, see *Troubleshooting* at the end of this Section.

Once the PostgreSQL service has been successfully restarted, continue to *PostgreSQL Client Authentication Configuration File.*

Hint

If PostgreSQL has been installed on the same server as all other MLSuite components, skip to section *2.3 Installing .NET Core Web Server Hosting.*

PostgreSQL Client Authentication Configuration File

The **pg_hba.conf** file must be edited to allow non-local connections to the PostgreSQL database. The **pg_hba.conf** file is located in the same directory as the **postgresql.auto.conf** file.

Open the **pg_hba.conf** file and scroll all the way to the end of the file. These default settings allow connections to the local database from the same local machine only.

In order to allow remote connections (i.e. in order to allow MLServer service to connect to the database from a different location), add the IPv4 and IPv6 addresses of all servers where PosgreSQL and MLServer components are installed, followed by "/32" for IPv4 or "/128" for IPv6.

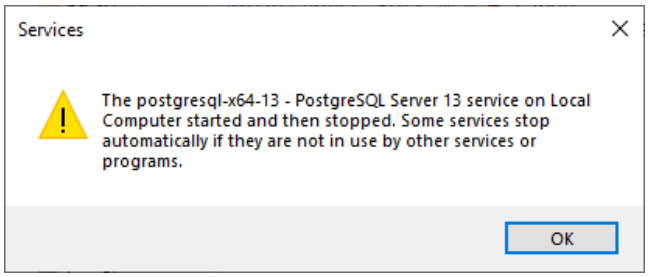
Hint

The IPv4 and IPv6 addresses can be retrieved by running ipconfig on the server hosting MLWeb. To do this, open Command Prompt and enter "ipconfig."

```
# Put your actual configuration here
# ----------------------------------
#
# If you want to allow non-local connections, you need to add more
# "host" records.  In that case you will also need to make PostgreSQL
# listen on a non-local interface via the listen_addresses
# configuration parameter, or via the -i or -h command line switches.



# TYPE  DATABASE        USER            ADDRESS                 METHOD

# IPv4 local connections:
host    all             all             127.0.0.1/32            md5
host    all             all             172.25.10.62/32         md5
# IPv6 local connections:
host    all             all             ::1/128                 md5
host    all             all             fe80::95f3:4ed7:d1:f047%4/128   md5
# Allow replication connections from localhost, by a user with the
# replication privilege.
host    replication     all             127.0.0.1/32            md5
host    replication     all             ::1/128                 md5
```

Troubleshooting

If PostgreSQL service cannot be restarted, a setting generated by PGTune may be invalid. This can be due to a simple typo when copying over the PGTune settings. Keep in mind that the settings are case-sensitive, so common issues to watch for are the use of KB vs. kB, MB vs.GB, and the correct application of # comment tags. The figure below shows the error that is displayed when a restart is attempted when a postgresql.auto.conf configuration value is invalid.



Other common issues may be related to:

- An incorrect **maintenance_work_mem** configuration. It may need to be lowered before PostgreSQL can be restarted.
- Or the total memory being set too high in the PGTune. Anything higher than 16GB is likely to cause issues.

2.2 Firewall Configuration

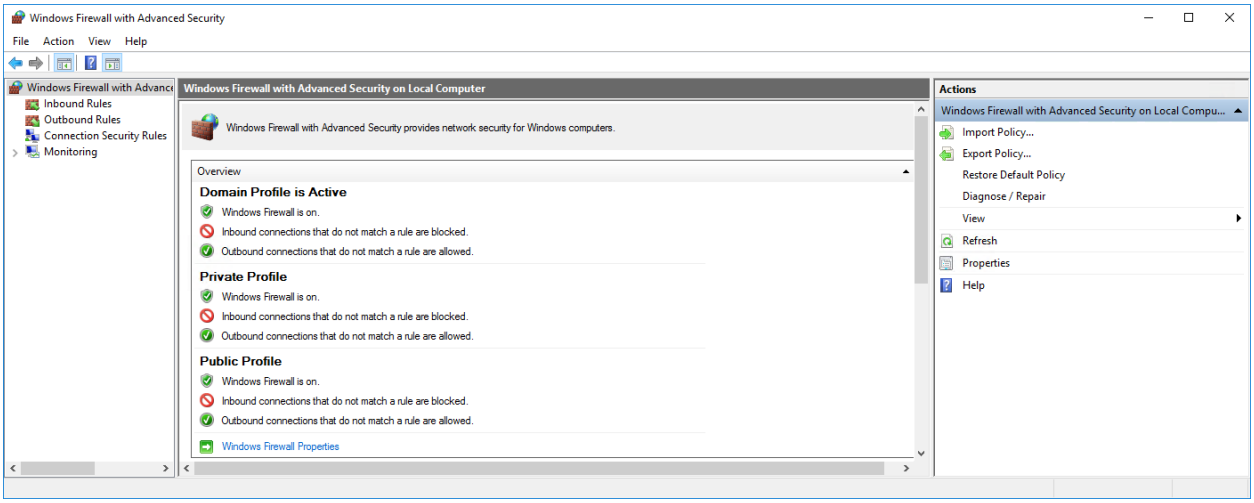
New firewall rules are required for **each** of the following instances:

- If the PostgreSQL database is installed on a server separate from other MLSuite components
- If MLWeb or MLAdmin must be accessed remotely

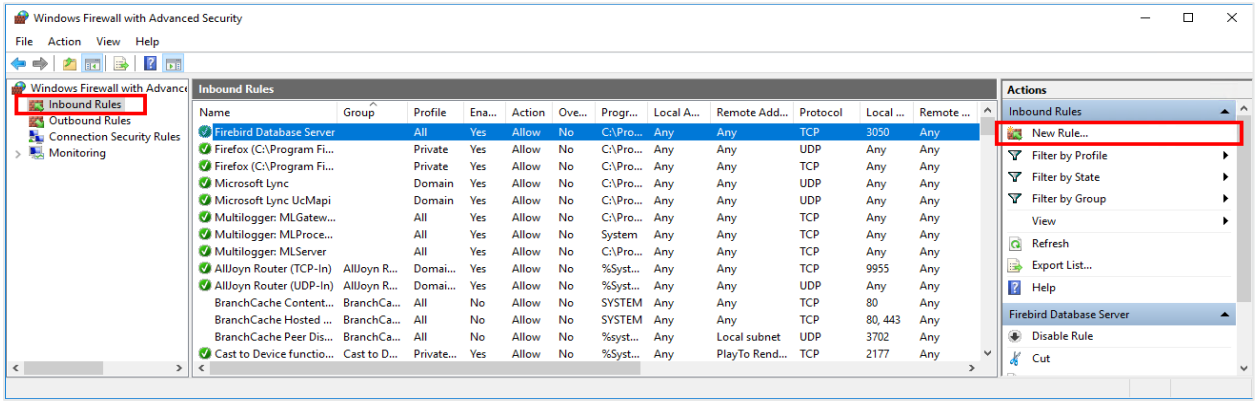
The process for adding each of these rules is similar (but must be done separately). See the respective subsections for each below. Any users without the need for these rules may skip ahead to section *2.3 Installing .NET Core Web Server Hosting*.

PostgreSQL Database on a Separate Server

1. On the server where PostgreSQL is installed, open the **Windows Firewall and Advanced Security** management console.

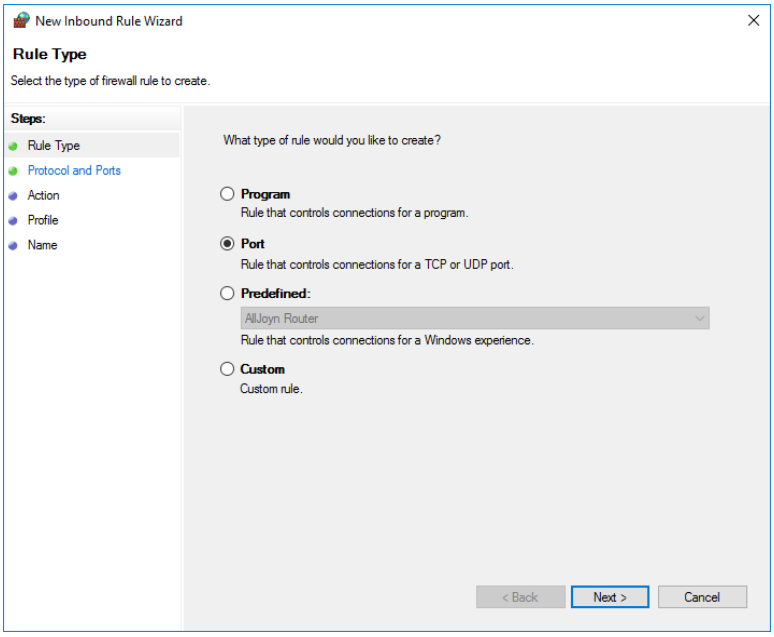


2. Select **Inbound Rules** from the menu on the left and then create a new rule by selecting **New Rule** from the **Actions** menu on the right. This opens the New Inbound Rule Wizard.

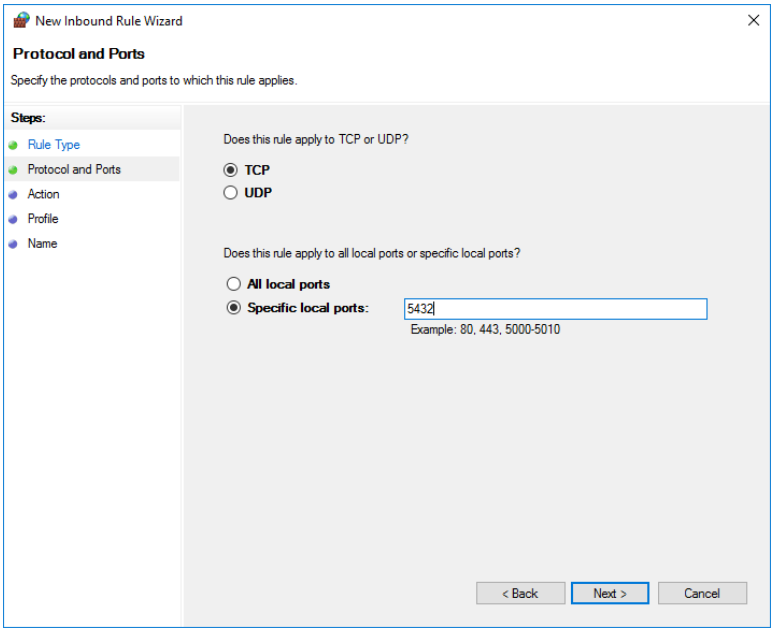


3. Configure the firewall rule using the **New Inbound Rule Wizard**.

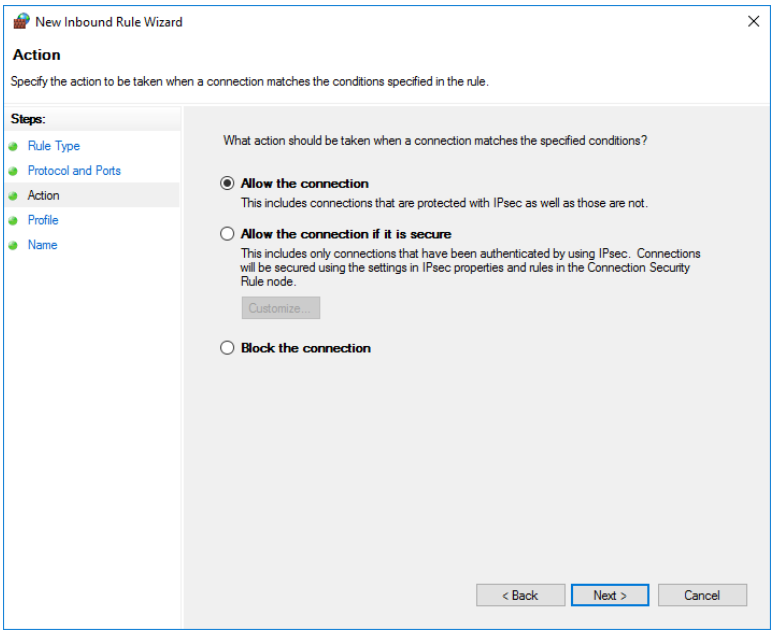
   a) Select **Port** as the **Rule Type** and click **Next**.



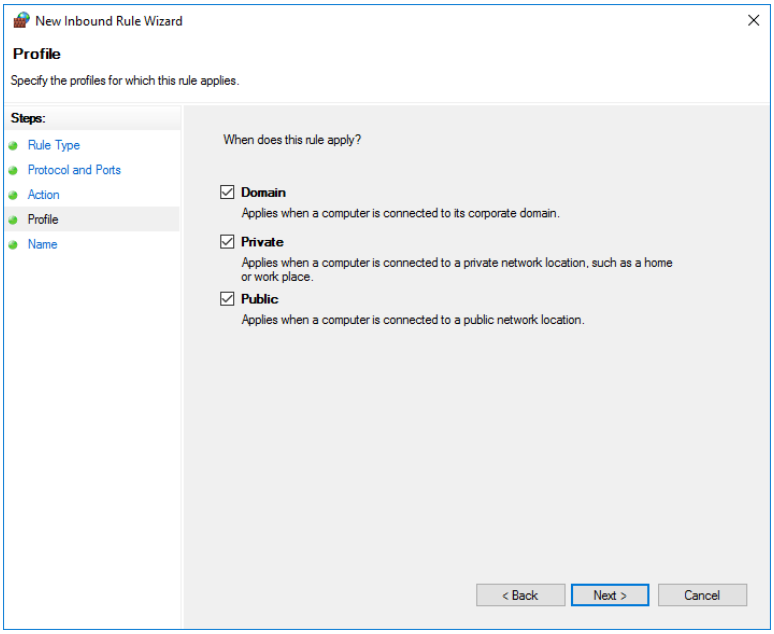
b) On the **Protocol and Ports** tab select **TCP** and **Specific local ports**, then set the local port to "5432." Click **Next**.



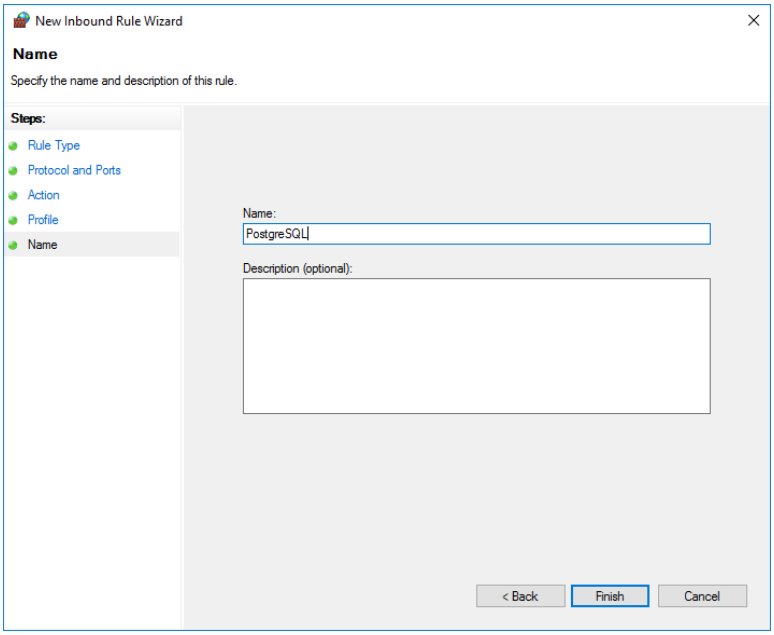
c) On the **Action** tab, select **Allow the connection** and click **Next**.



d) On the **Profile** tab, leave all check boxes selected and click **Next**.

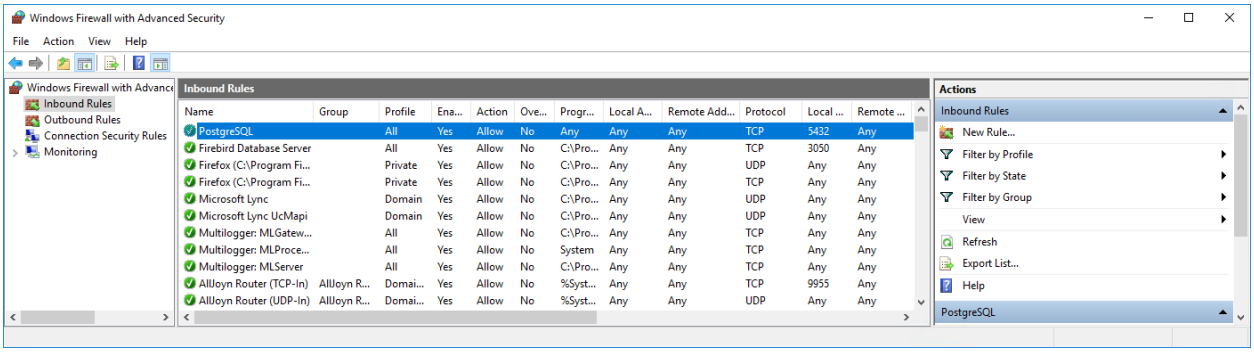


e) In the **Name** field, give your new rule a name and then click **Finish**.



4. Once the new rule has been created, additional settings must be configured.

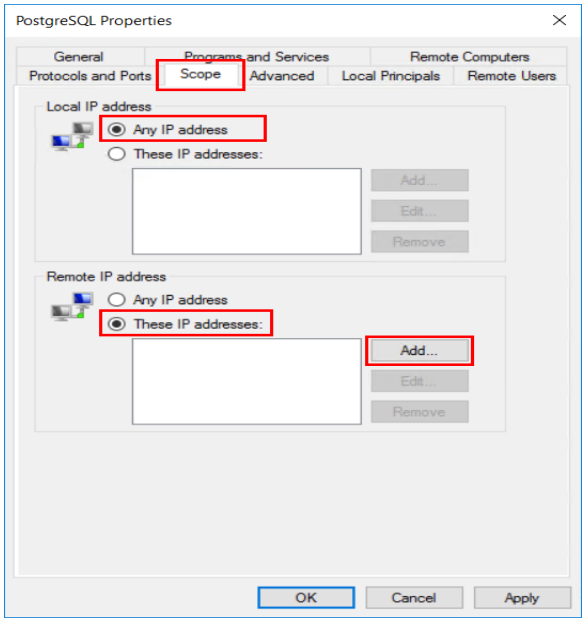
   a) Locate the new rule in the Inbound Rule table and double click it to open the **Properties** form.



   b) From the Properties form, select to the **Scope** tab.

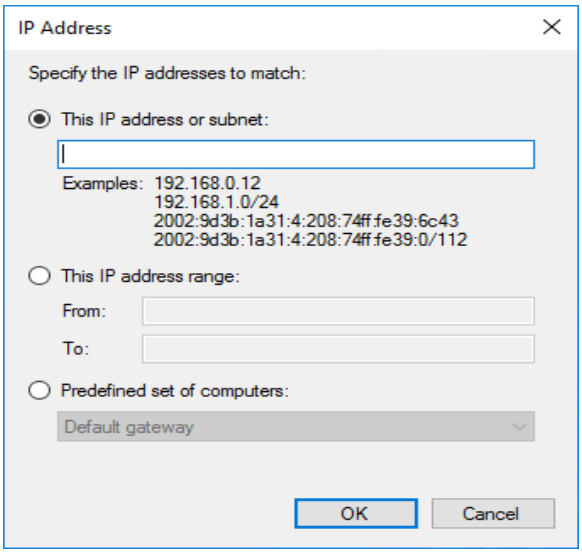
   c) The **Local IP address** can be left as **Any IP address**. However, the **Remote IP address** should be specified.

      i. Select **These IP Addresses:** and click **Add...**

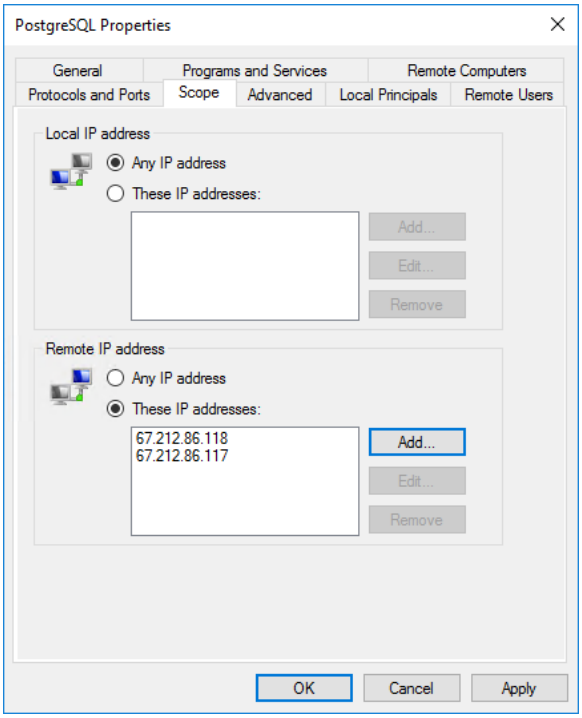


      ii. The IP address for each server that is hosting MLSuite components that will communicate with the PostgreSQL server must be added. Type in the IP and click **OK**.

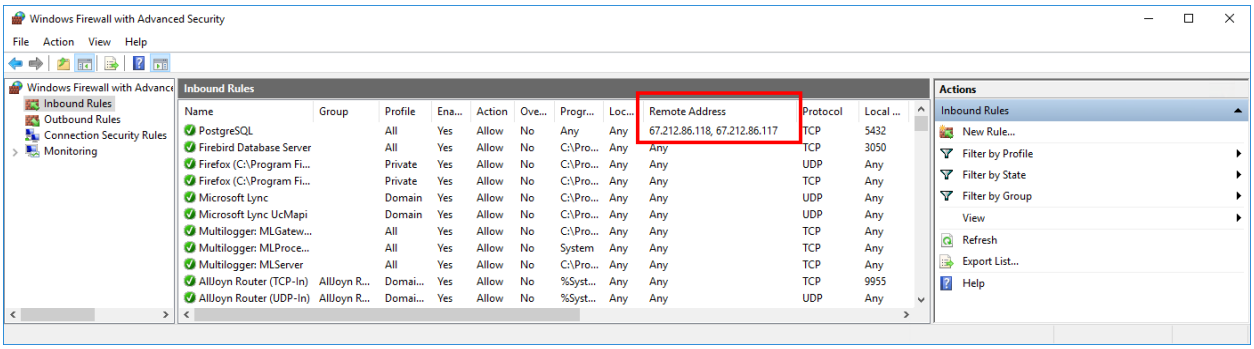
iii. Repeat for each IP address.



iv. Once all the necessary IP addresses have been added click **Apply** and **OK**.

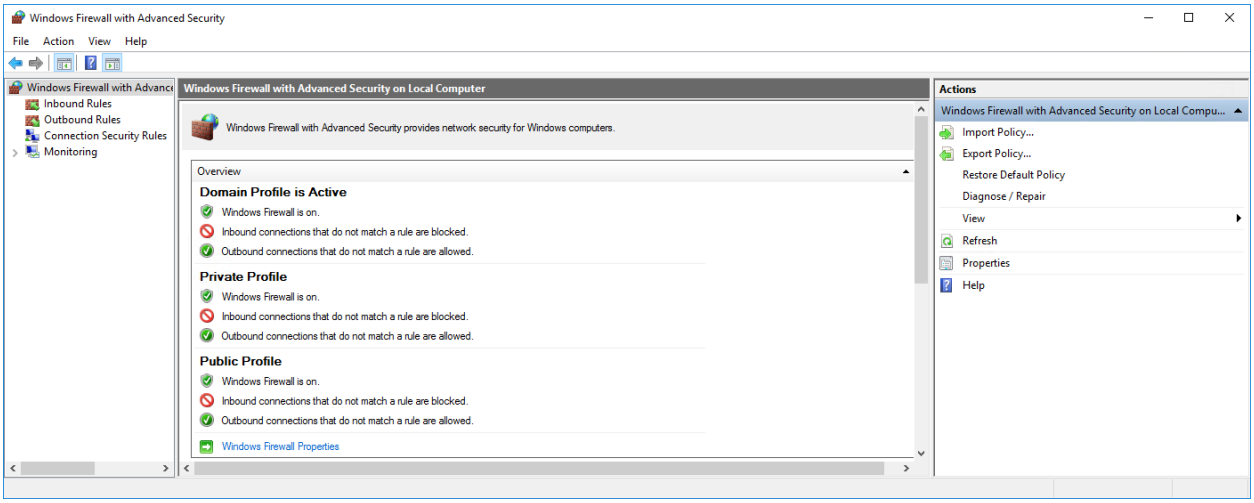


The allowed IP addresses are shown under Remote Addresses for the new Inbound Rule.

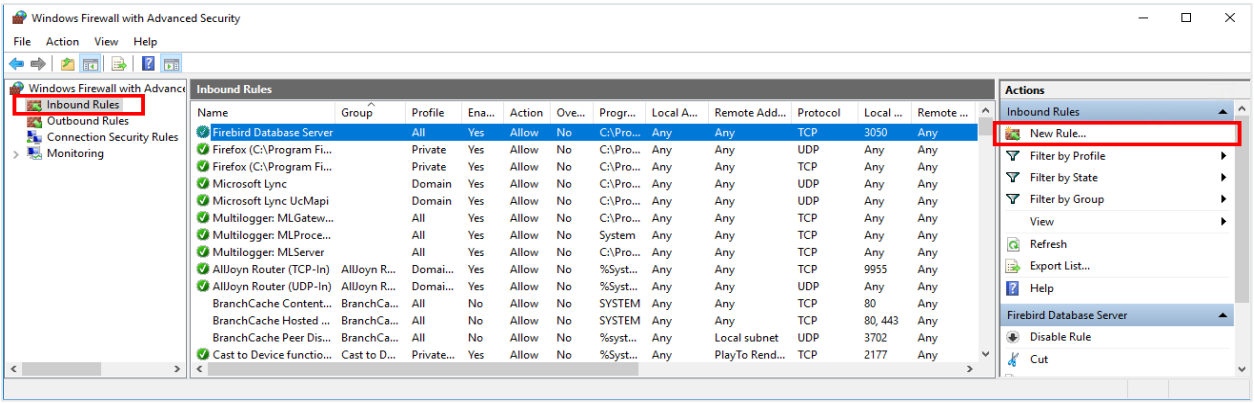


MLWeb or MLAdmin to be Accessed Remotely

1. On the server where MLSuite is installed, open the **Windows Firewall and Advanced Security** management console.

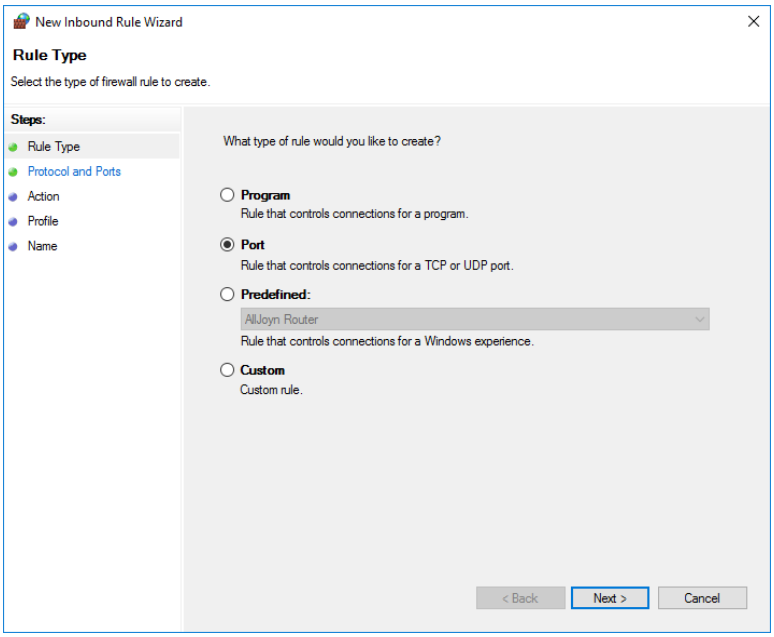


2. Select **Inbound Rules** from the menu on the left and then create a new rule by selecting **New Rule** from the **Actions** menu on the right. This opens the New Inbound Rule Wizard.

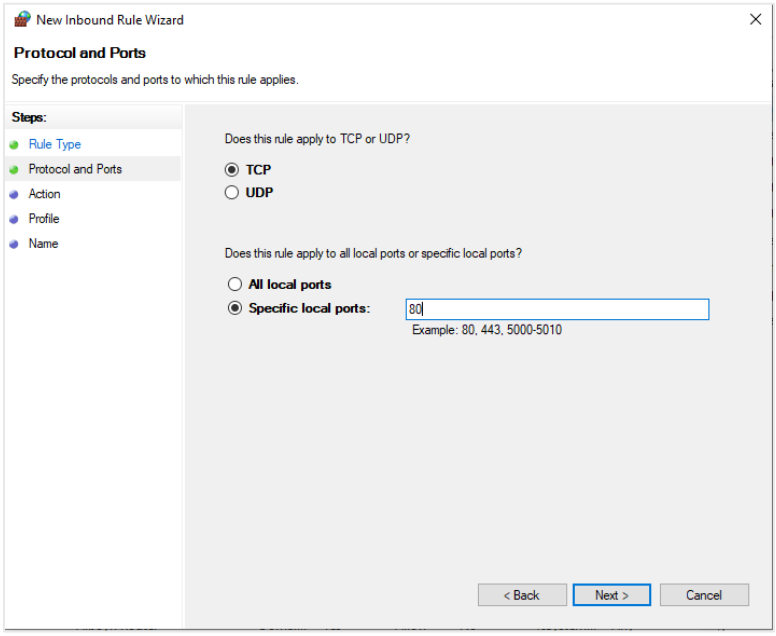


3. Configure the firewall rule using the **New Inbound Rule Wizard**.

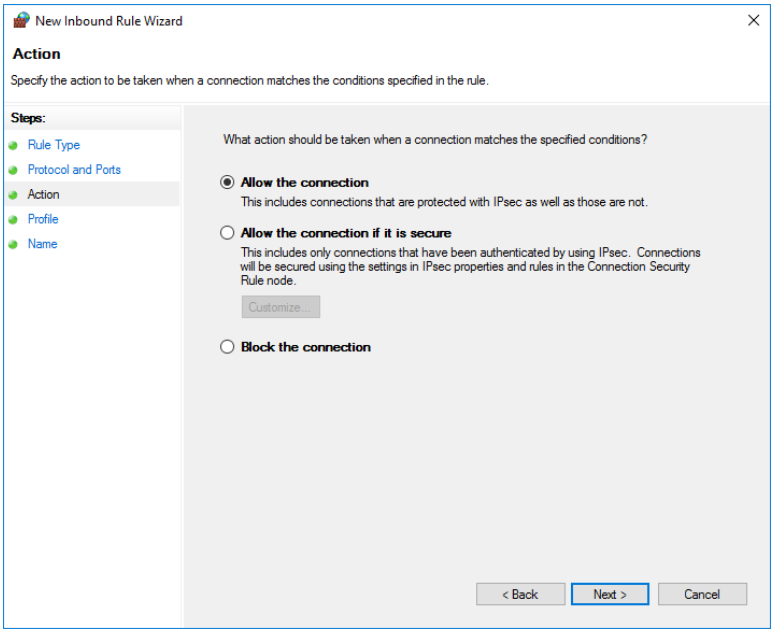
   a) Select **Port** as the **Rule Type** and click **Next**.



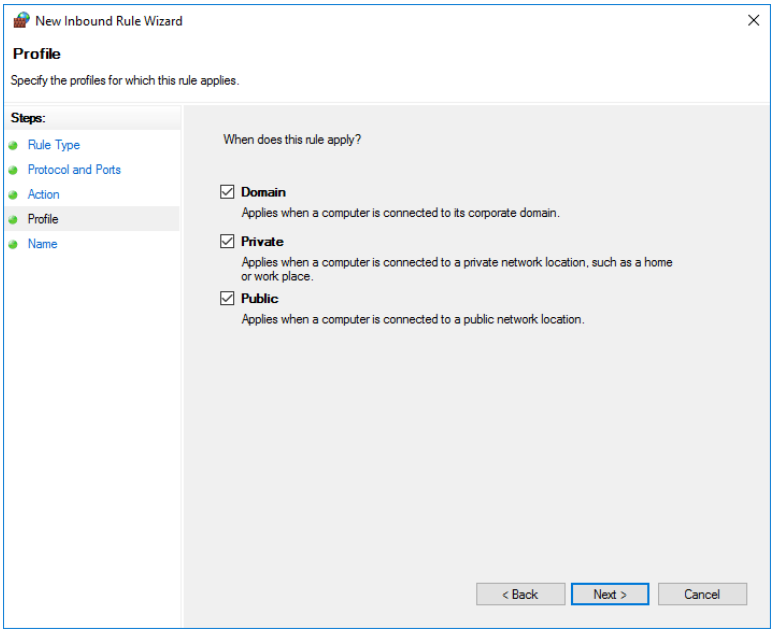
   b) On the **Protocol and Ports** tab select **TCP** and **Specific local ports**, then set the local port to "80" (or "443" for https). Click **Next**.



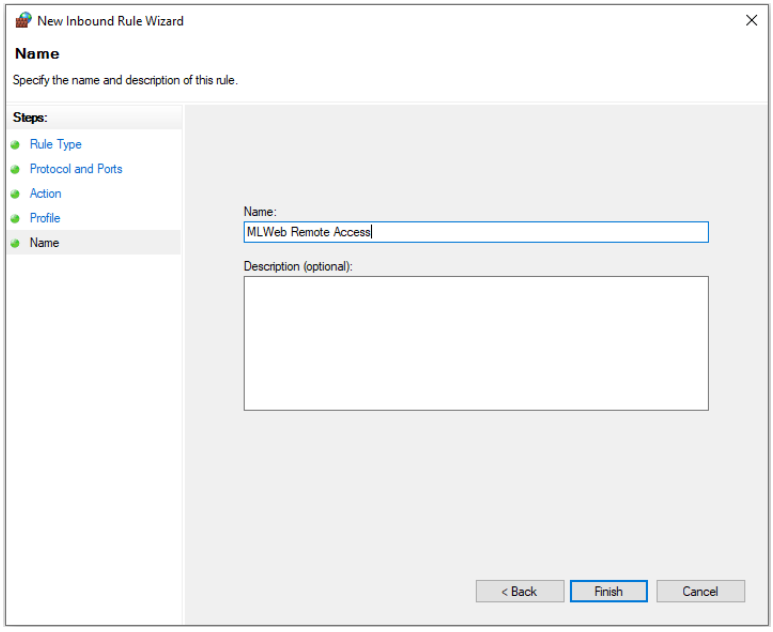
c) On the **Action** tab, select **Allow the connection** and click **Next**.



d) On the **Profile** tab, leave all check boxes selected and click **Next**.

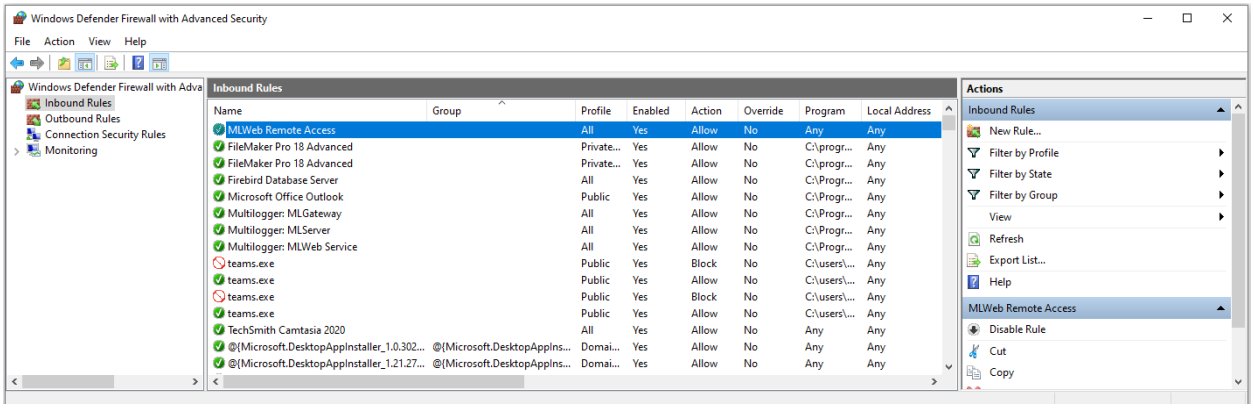


e) In the **Name** field, give your new rule a name and then click **Finish**.



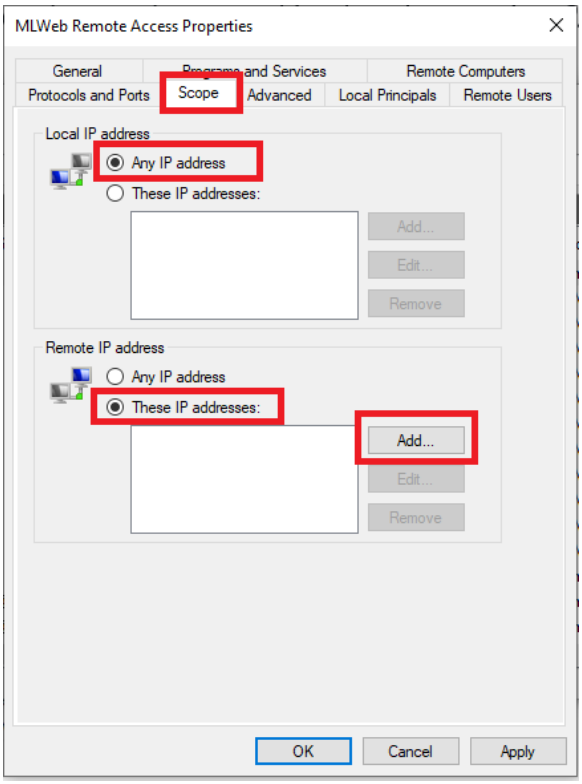
4. Once the new rule has been created, additional settings must be configured.

a) Locate the new rule in the Inbound Rule table and double click it to open the **Properties** form.



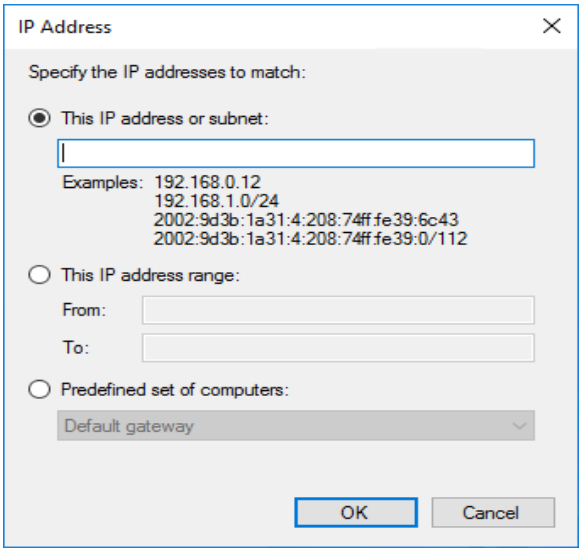
b) From the Properties form, select to the **Scope** tab.

c) The **Local IP address** can be left as **Any IP address**. However, the **Remote IP address** should be specified.



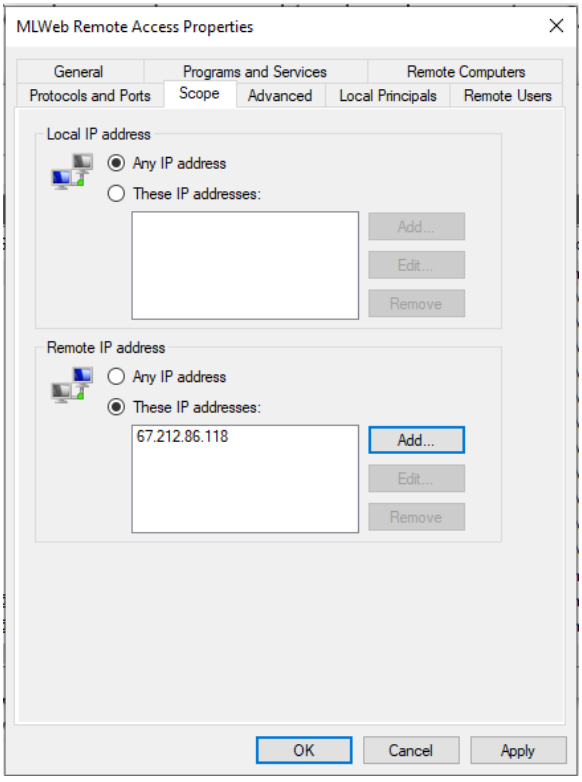
i. Select **These IP Addresses:** and click **Add...**

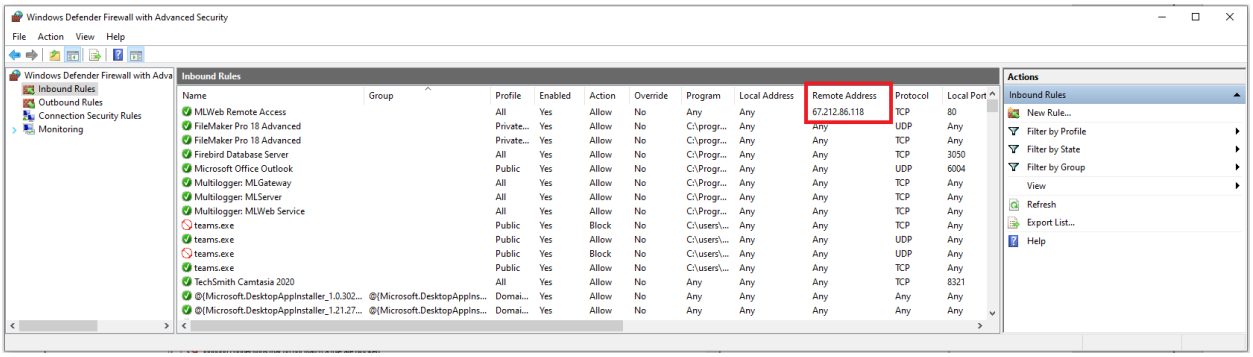
ii. The IP address for each server that is hosting MLSuite components that will communicate with the PostgreSQL server must be added. Type in the IP and click **OK**.



iii. Repeat for each IP address.

iv. Once all the necessary IP addresses have been added click **Apply** and **OK**.





The allowed IP addresses are shown under Remote Addresses for the new Inbound Rule.

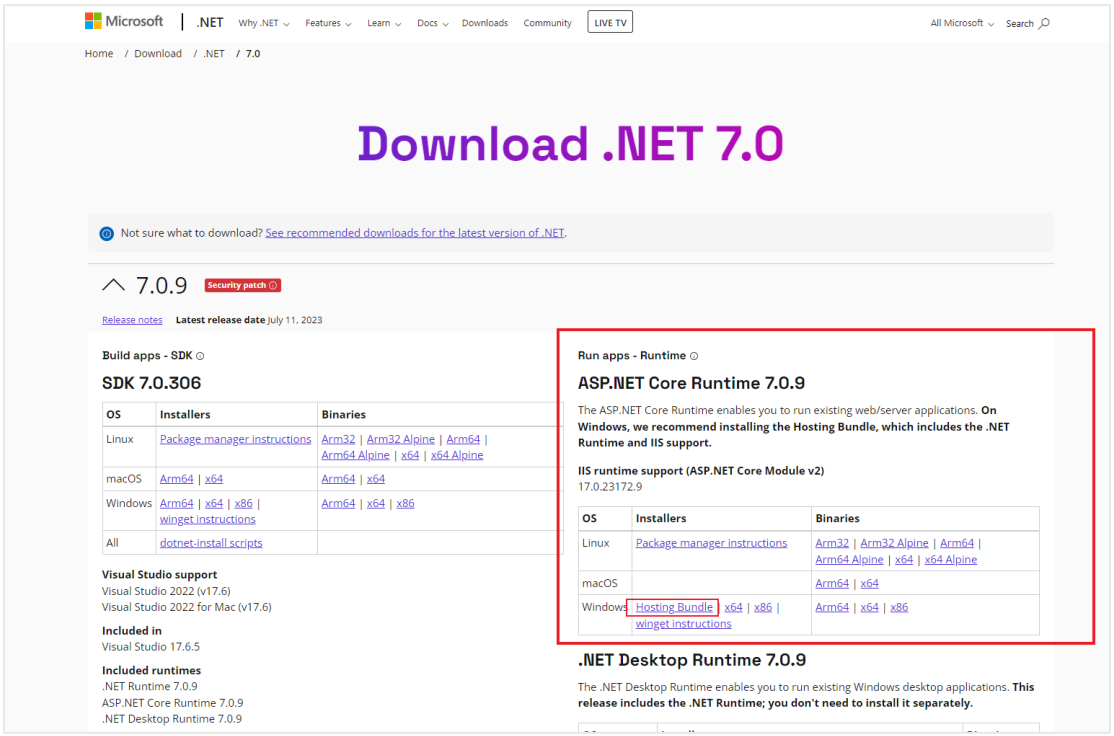
2.3 Installing .NET Core Web Server Hosting

The MLSuite installation processes configures and launches a web server host that is responsible for startup and lifetime management of **MLAdmin**.

Microsoft ASP.NET Core Runtime 7.0 provides the resources for this. Microsoft ASP.NET Core Runtime 7.0 must be downloaded from Microsoft; the **ASP .NET Core Runtime Download** link is available on the Canary Systems website under **Third Party Downloads & Utilities** (canarysystems.com/support/software-downloads).

> **Note**
>
> This is currently the only version of ASP .NET Core Runtime that is compatible with MLSuite. No other version should be installed.

Download the ASP.NET Core Runtime 7.0 (latest version of the Windows Hosting Bundle) as shown below.

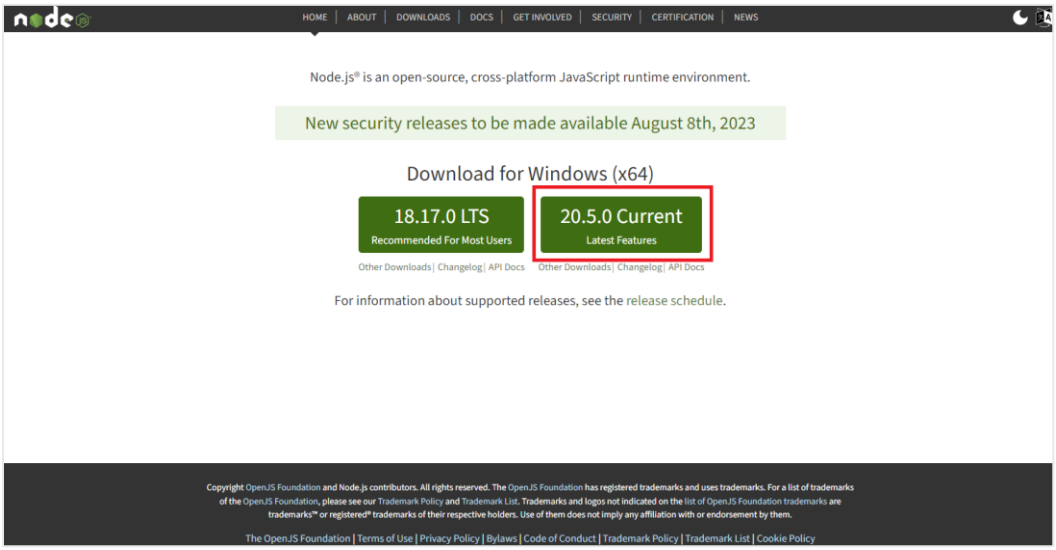


Run the installer once downloaded.

2.4 Installing NodeJS

NodeJS is an asynchronous event-driven JavaScript runtime that serves as a supplement to the new charting engine in MLWeb 2023.

To ensure charts render correctly, download the latest version of NodeJS 20.0; a download link is available on the Canary Systems website under **Third Party Downloads & Utilities** (canarysystems.com/support/software-downloads).



Once downloaded, run the installer with default settings. When the installer is finished, restart the server.

2.5 Installing MLSuite

Before launching the MLSuite Installer, ensure that PostgreSQL, Microsoft ASP.NET Core Runtime 7.0.0, NodeJS, and all their associated extensions as detailed in sections 2.2, 2.3, and 2.4 have been installed.

Software Version

The version of the software is included in the installer's name. The year indicates the major release and the following number the service pack. The installer for MLSuite MLWeb 2023 initial release is **MLSuite_2023.1.0.92_x64.msi**.

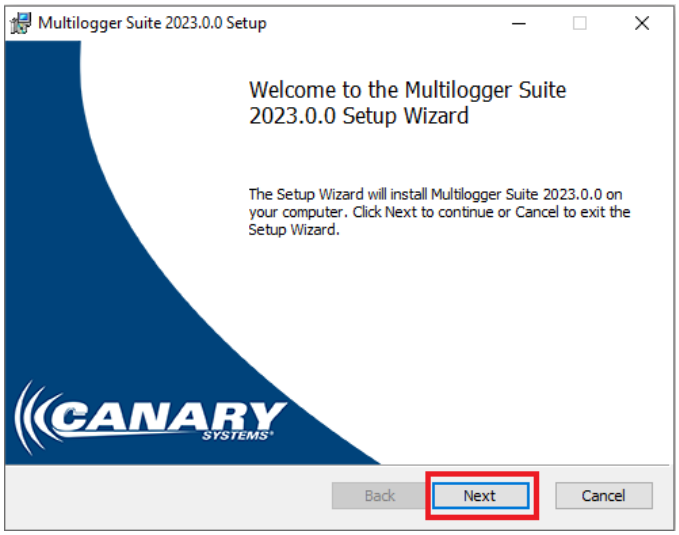
User Installation Restrictions

MLSuite may be installed on any computer. However, if User Account Controls restrict the installer from making changes, a dialog box will appear asking the user to enter the Administrator's credentials to continue the installation.

MLSuite Setup Wizard

Follow the below steps to complete MLSuite installation. If necessary at any step in the process, click **Cancel** to terminate the installation. Note that this action rolls back all progress made from when the Setup Wizard was launched (not including any manual migration work).

1. To begin, Run the MLSuite Installer. Click **Next** to proceed.

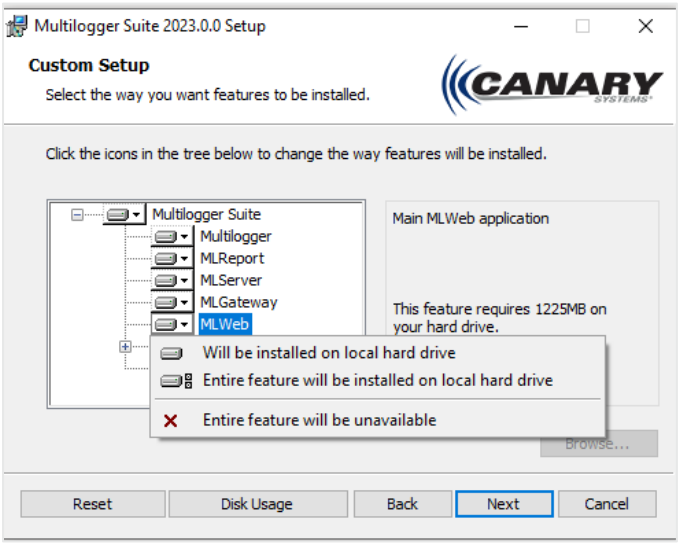


2. Read the entire End-User License Agreement (EULA), displaying in the viewing window. It may be printed for later review and reference purposes by clicking the **Print** button.
3. Check **I accept the terms in the license agreement** to indicate agreement.
4. Click **Next** to continue to open the installers **Custom Setup** wizard, where the application installation options and program file paths can be configured.

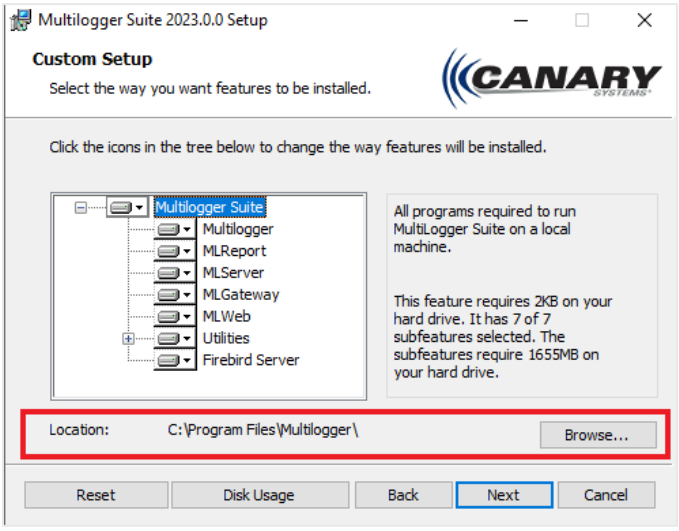
Installation Options

Specific applications can be selected for installation by clicking on their icon and changing their installation options, as shown in the figure below. This allows the installer to choose to only install certain features, or not to install others, dependent on server setup and project requirements.

By default, various MLSuite Utilities are installed with the software. For more information on these Utilities, see section *2.5.2 MLSuite Utilities.*



From the Custom Setup form, the installer is able to designate the installation path for the program files. The default location is displayed at the bottom of the form, as highlighted in the figure below. Typically, a path in the Program Files folder is used.



5. Click the **Browse** button to specify an alternate path if needed.

Additional options are on the installer's Custom Setup wizard include:

Reset – Resets the form to its default options.

Disk Usage – Displays a form showing all available drives, their available storage space and the required storage space for the program and its applications as configured.

6. Click **Next** to continue.

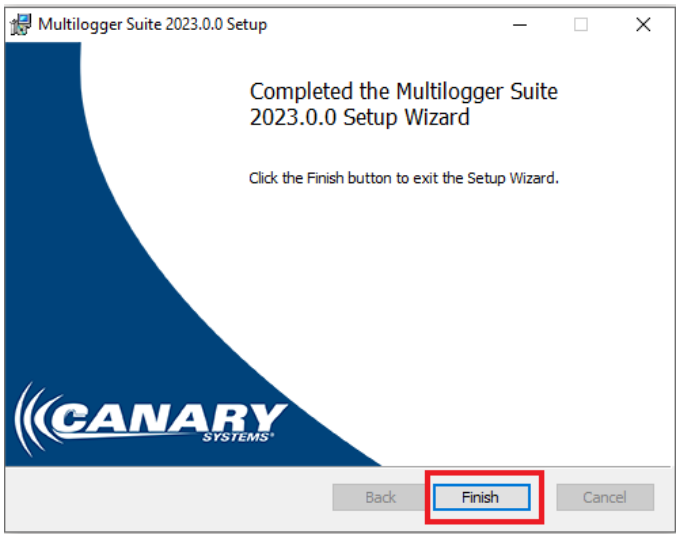
Note

Canary Systems recommends installing MLReport on your local machine. See appendix *B.1 Installing MLRemote Locally* for details on installing MLRemote on your local machine.

7. Before the installation process begins, a settings confirmation screen is displayed. Click **Install** to begin the installation.

8. If prompted to allow the installer to make changes to the drive, select **Yes**. The MLSuite program files are then copied to the destination folder.

9. When the installation has been completed successfully, the screen in Figure 2.5 D is displayed. Click **Finish** to complete the installation.

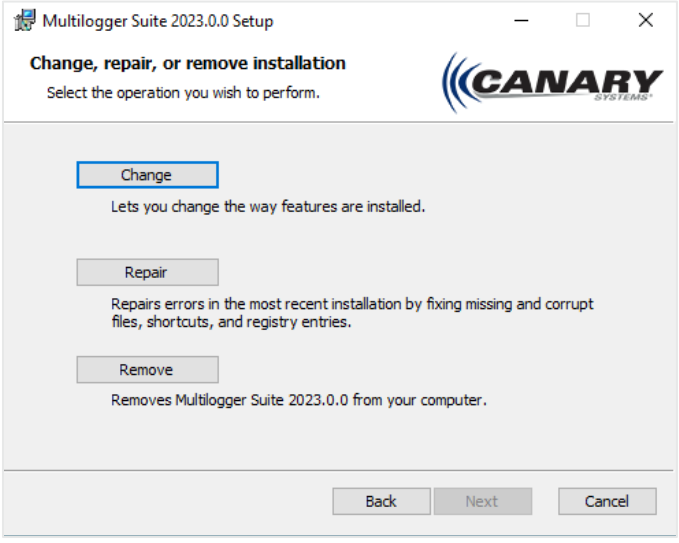


The desktop shortcuts have now been created for MLGateway, MultiLogger, MLServer, MLReport, and MLWebHardware.

Now that MLSuite is installed additional configuration is required. Continue to section *2.6 Creating Firebird Temp Folder*.

2.5.1 Modifying the Installation

After initial installation, the same installer can be run again to add or remove MultiLogger features, repair installation errors, or uninstall MLSuite.

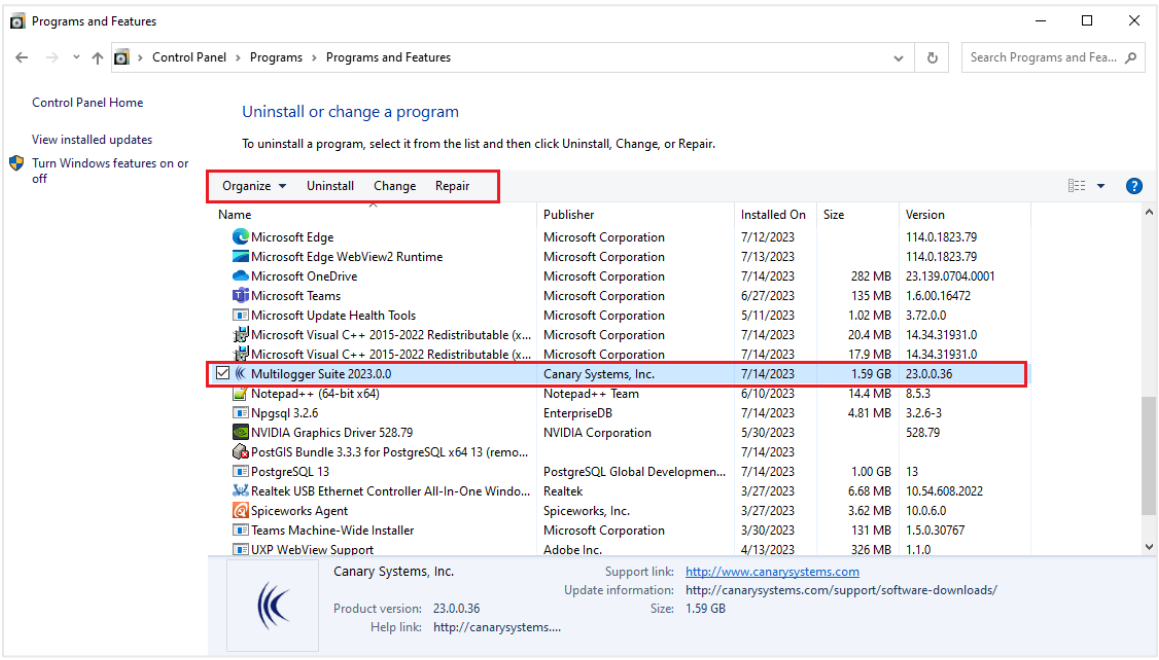


Change – Select this option and click **Next** to display the Custom Setup form. Here, features can be added or removed.

Repair – Select this option and click **Next** to fix errors in the program installation.

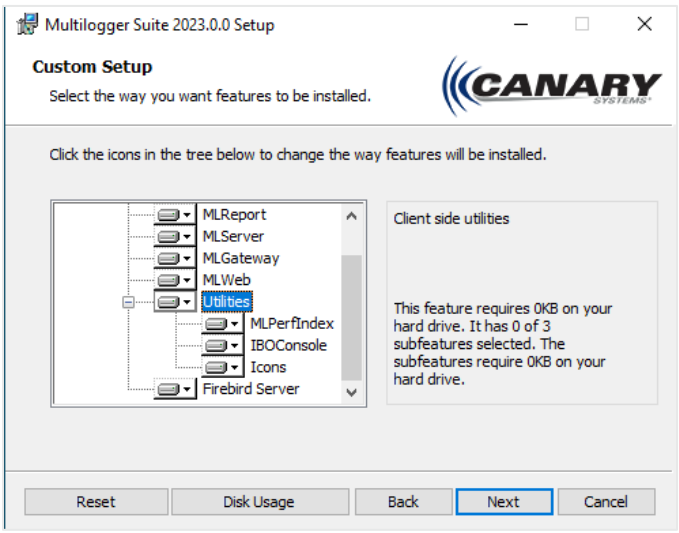
Remove – Select this option to uninstall MLSuite from the computer or server.

The installer can also be run from the Control Panel, in the **Programs and Features** menu. Select **MLSuite** from the list and select **Uninstall**, **Change** or **Repair** to perform the operations described above.



2.5.2 MLSuite Utilities

The MLSuite installer includes three Utilities that are installed by default.



MLPerfIndex – A tool to aid in optimizing server performance, see *Appendix E – Performance Index Tool* for more information.

IBOConsole – Firebird database management console.

Icons – Installs a library of icons that can be used for data visualization in MLWeb, see *Appendix B.2 – Installing the MLWeb Icon Library Locally* for more information.

2.6 Creating Firebird Temp Folder

Firebird databases create temporary files when running **Backup** and **Restore** tasks. By default, these files are written to the Windows Temp folder. These files are typically three times the size of the Firebird's .gdb file.

Note

Canary Systems recommends creating a folder other than Windows Temp to store Temp files. Once created, the path for this folder must be entered into the conf file (see steps below).

Hint

Modify permissions are required to edit this file. Refer to section *2.7 Granting User Permissions*.

1. Create a new folder where the Firebird temp files can be written: "FB_Temp"
2. Locate the **firebird.conf** file (Program Files > MultiLogger > Firebird > firebird.conf)
   a) Open file with notepad and find "TempDirectories"

```
# ----------------------------
# Temporary directories
#
# Provide ';'-separated trees list, where temporary files are stored.
# Relative paths are treated relative to RootDirectory entry
# (see above). Default value is determined using FIREBIRD_TMP,
# TEMP or TMP environment options. Once the first specified
# directory has no available space, the engine will switch to the
# next one, and so on.
#
# E.g.:
# TempDirectories = c:\temp
# or
# TempDirectories = c:\temp;d:\temp
#
# Type: string (special format)
#
# TempDirectories =
```

b) Remove the "#" to uncomment out the TempDirectories and include the path of the new folder: "TempDirectories = C:\Windows\Temp\FB_Temp"

```
# Type: string (special format)
#
TempDirectories = C:\Windows\Temp\FB_Temp
```

3. Save the firebird.conf file

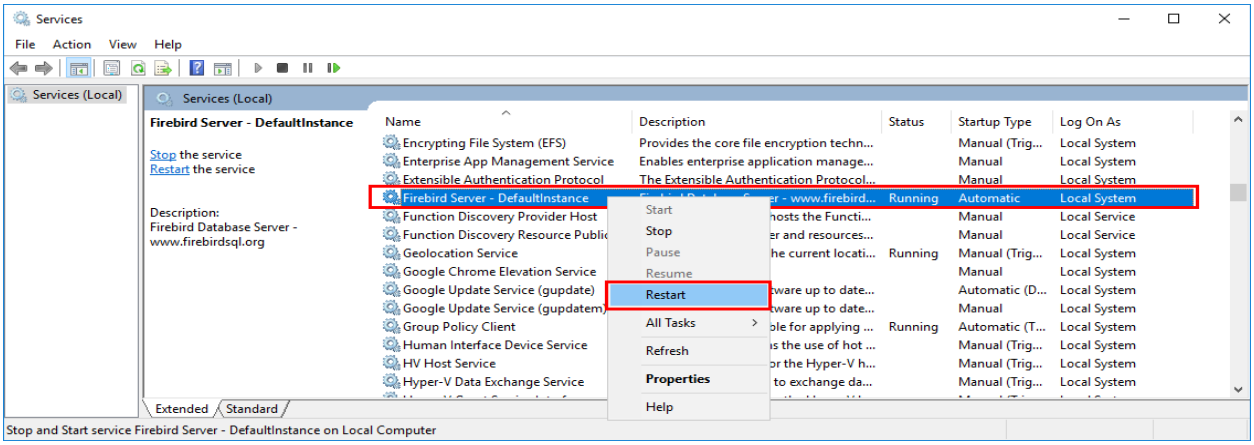
> **Note**
>
> When performing an upgrade, the .conf file will not be updated once it has been edited.

4. Restart the Firebird server

   a) Open **Services**

   b) Navigate to **Firebird Server – DefaultInstance** and right click

   c) **Restart**



2.7 Granting User Permissions

Any user accounts that will be used to log onto the server, to access MLServer or MultiLogger, require permission to specific folders to be granted. The accounts need "Modify" access to the following locations:

- The CanarySys folder (and all subfolders) - by default this is under C:\Users\Public\Documents\CanarySys but is often moved to the root of one of the data drives on the server.
- The Temp folder (and all subfolders) - by default this is located in C:\Windows\Temp, but that can be moved by changing the system environment variables.

Note

Other customized files and folders may require similar permissions. For questions on specific setups, Contact Canary Systems Support.

2.8 Binding SSL Certificate

It is highly recommended that databases hosted on the web have an SSL certificate. SSL certificates authenticate a website's identity and enable an encrypted connection. This security protocol ensures that when users interface with the database, the data therein is protected.

Check with your organization"s IT department to see if an SSL certificate already exists for your website. New SSL certificates can also be purchased from various vendors on the web.

The process for binding the certificate to an IP and port is found below. For this part of the setup, you will need the certificate's fingerprint or "hash," the target machine's IP address, and the port to be used if different from the default.

1. Run the following from elevated PowerShell:

   netsh http add sslcert ipport="<IP>:<PORT>" certhash="<CERTHASH>" appid=("{"+[guid]::NewGuid()+"}")

   Replace **<IP>** with the IP address of NIC web portal will be hosted through. Use 0.0.0.0 to designate any/all.

   Replace **<PORT>** with the port number that the web portal is accessible at via above IP address.

   HTTPS is 443.

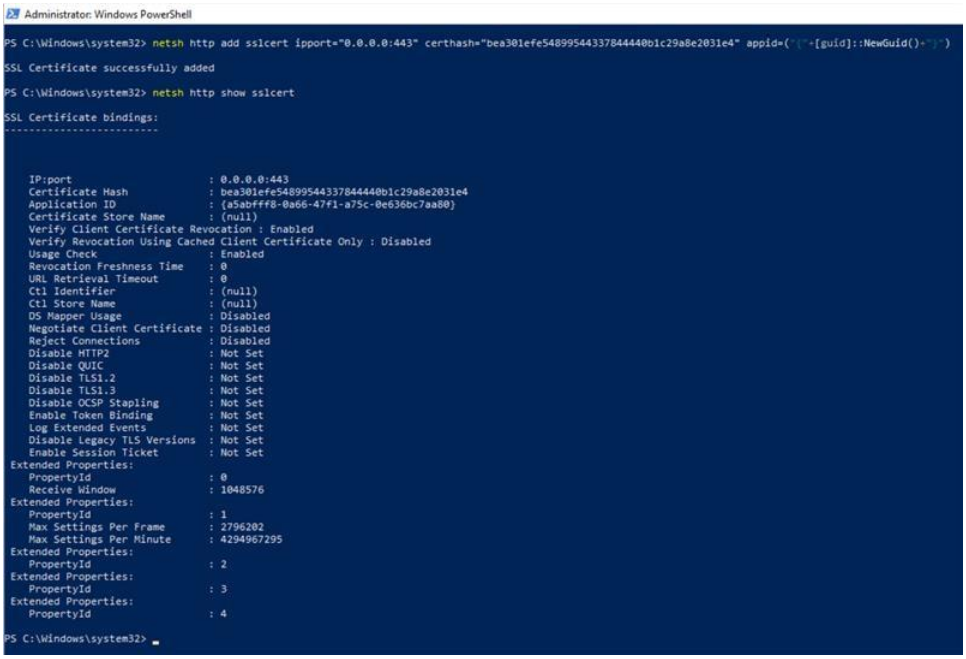
   Replace **<CERTHASH>** with the SSL certificate hash/thumbprint.

   If the command ran correctly, the following line in PowerShell displays, "SSL Certificate successfully added."

2. Restart the MLWeb service associated with the SSL certificate.

3. Run the following command to view/confirm SSL certificate binding:

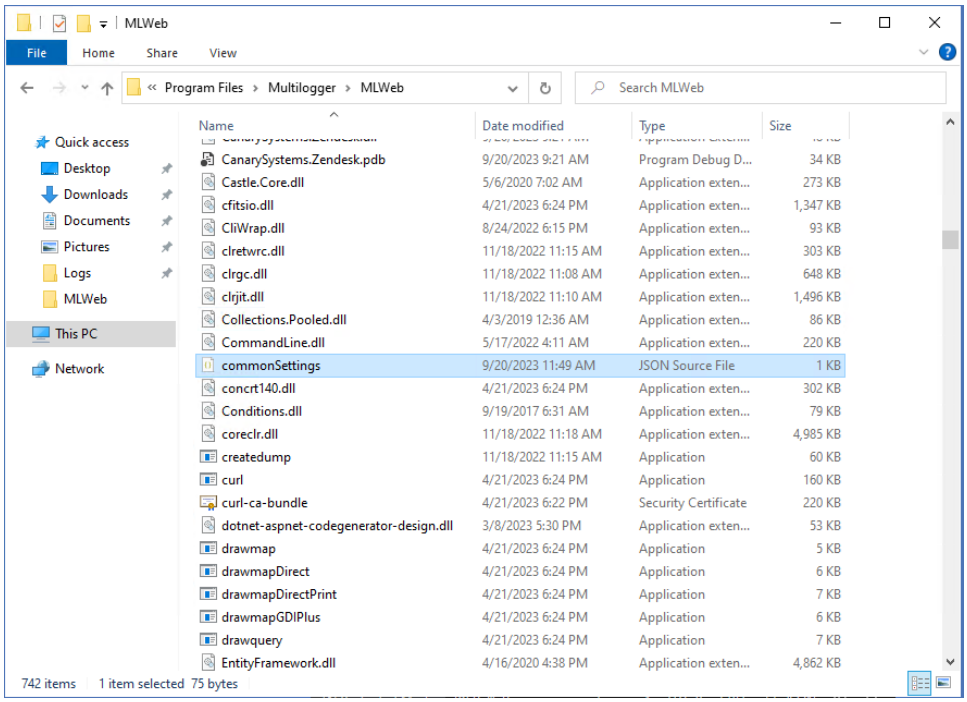
   netsh http show sslcert



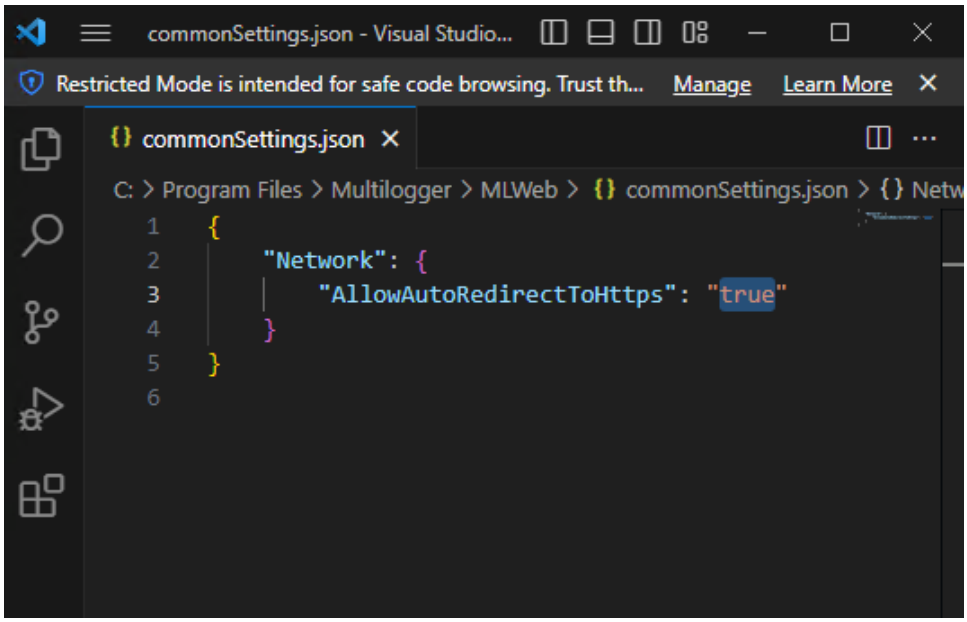
Redirecting Http to Https

To setup autoredirect from http to https, follow the below steps:

1. From the directory where MultiLogger is installed, navigate to Program Files | MultiLogger | MLWeb and then open commonSettings.json.



2. Change the AllowAutoRedirectoHttps command from “false” to “true” as shown in the figure below.



3. Restart MultiLogger MLWebService.

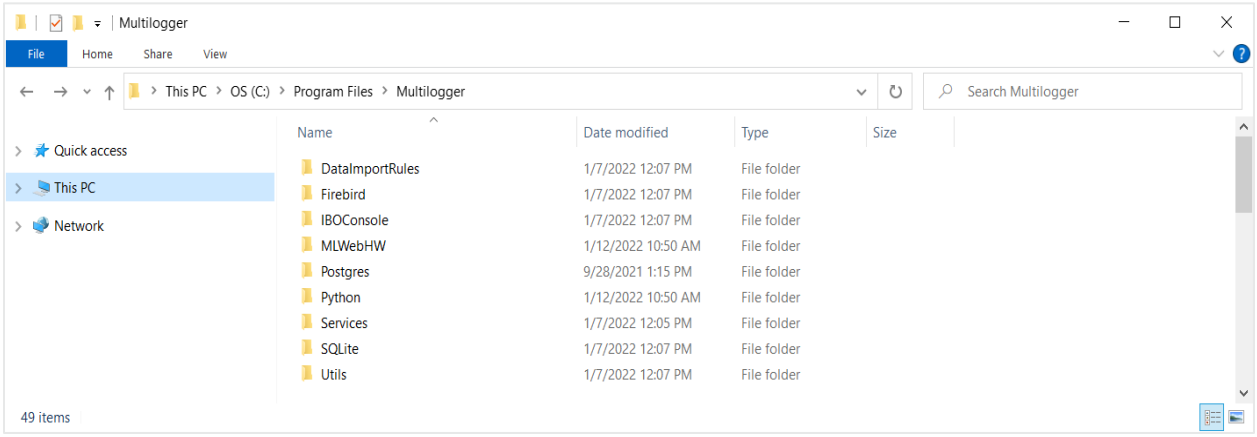
Note

This process performs redirecting from http to https by redirecting traffic from port 80 to port 443. The logic will not not work in cases where other ports are in use or the app is being accessed from an external network that uses different ports mapping. It works strictly with 80 and 443 port numbers.

3. Launching and Configuring Applications

Once the MLSuite installer completes, shortcuts for the installed applications and services are added to the desktop.

Applications and services can also be launched from the **MultiLogger** folder generated during installation. By default, the MultiLogger folder's file path is **\Program Files\MultiLogger**.



Additionally, shortcuts can be placed in the Startup folder so that applications will be launched when a user logs in.

A MLSuite group in the Start menu is created by the installer; this provides shortcuts to the primary applications installed.

3.1 MLGateway

MLGateway is designed to operate as a service; MLWebHardware and MultiLogger are applications used to configure the MLGateway service. MLGateway performs all the actual functions related to system management, programming, monitoring and data collection. MLGateway is installed as a Windows Service. Running as a service provides the advantage of not requiring a user to be logged in for MLGateway to function.

MLGateway performs the following functions:

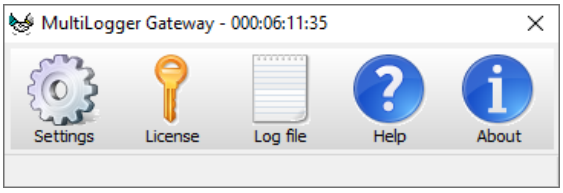
- Multi-user access to dataloggers and other hardware
- MultiLogger text and graphical monitoring functions
- Automation of data collection
- Automation of database data import (through MLServer)

Open the MLGateway interface by clicking on the MLGateway icon , found on the desktop or in the MLSuite group in the start menu.

Note

Closing the Control Panel does not close MLGateway, as it is installed and running as a service. To find out more about managing Windows Services, see Microsoft's Windows Services documentation.

The MLGateway Control Panel is shown in the figure below. It provides status information as well as buttons to access the various forms and functions.



The Function Buttons perform the following actions:

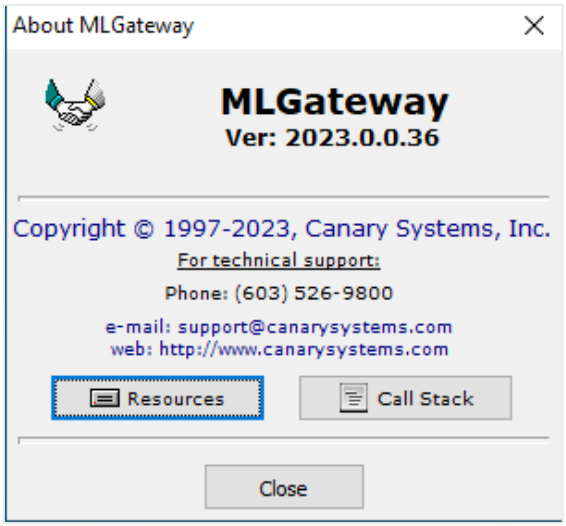
Settings – Configures the Port used for MultiLogger to MLGateway communications.

License – Configures the MLGateway license.

Log File – Displays the log file viewer.

Help – Displays the help file.

About – Displays the current version and support contact information. This includes the **Call Stack** button that displays the call stack information for use in troubleshooting. This information can be copied or emailed directly to Canary Systems support from the dialog.



3.2 MLServer

MLServer is the application included with the MLSuite software that provides the back-end functionality for MLWeb and MLField. MLServer is installed as a Windows Service. Running as a service provides the advantage of not requiring a user to be logged in for MLServer to function. It performs the following functions:

- Importing data sent to it by MLGateway
- Importing data collected by other systems and/or software through import configurations
- Automated backup and restore functions
- Processing of data in the database, generation of alarms and other notifications
- Processing and generation of automated outputs
- Management of MultiLogger Gateway Lists

4. MLAdmin Configuration

MLSuite® includes a browser-based admin page for management of databases and sites. Access to the admin page, MLAdmin, can be restricted to local access only, or can be made available remotely through a network.

Note

Canary Systems recommends contacting our Support team to assist with the initial configuration of MLAdmin. See pg. 4 of this guide for contact details.

MLAdmin is accessed on the machine hosting MLWeb locally at **http://localhost/admin**, or remotely at **http://[Machine Name or IP]/admin**.

4.1 Installation Setup

MLAdmin must be configured after installation of MLSuite. Upon initial login, the setup process begins.

1. MLAdmin has a dedicated Administrator Account for its management. A username and password for this account must be created.

   Be sure to save this information, as this account is needed to view and edit settings in MLAdmin. Click **Submit** when complete.

2. The location of each MLSuite service must be specified. To reach the editor for this function, navigate to **Configuration | Services**. Enter the **machine name or IP** of the machine hosting each service, then click **Save**.

3. The passwords for each required account must be entered. To reach the editor for this function, navigate to **Configuration | Passwords**. Click the icon in each field to display the password. Click the icon in each field to hide the password.

The accounts are as follows:

SYSDBA – Enter the password for the Firebird sysdba user. For initial set up, **masterkey** is required, however it can be changed later through MLWeb.

POSTGRES ml_user – An internal user for accessing PostgreSQL data. Enter a password as desired.

POSTGRES postgres (admin) – The PostgreSQL super user. This internal user is used for database operations. Enter the password created during installation of PostgreSQL, as detailed in section *2.2.2 PostgreSQL and Add-on Installation*.

MLUSER – An internal Firebird user.

Note

The process of setting up or changing the MLUSER password requires a cmd file to be run. From the directory where MLSuite has been installed:

1. Navigate to **Program Files | MultiLogger | Firebird | bin** and then select fb_users_setup.cmd.
2. Run the cmd.
3. When prompted, enter the SYSDBA password you have just set up in MLAdmin. This brings up a series of options.
4. Type 1 and then hit Enter to select 1. Setup ML_User_account
5. Enter a password as desired. Make sure to take note of this.
6. Press Enter; the cmd closes.

Return to MLAdmin and enter the password you have just setup into the MLUSER Password field. **The password entered here must match the password you have set up in the cmd.**

Windows process User Name – Enter the Windows user name created for the Process account.

Note

If one does not yet exist, a windows account called "Canary_Process" must be created before continuing. This can be either a Domain account or a local system account. The Canary Process account should be a user level account. See **Appendix D – Creating Canary Process Account** for steps on how this is done.

Windows process Password – Enter the Windows password configured for the Process account.

4. From the **Admin Dashboard Configuration**:

   a) Specify how MLAdmin is accessed.

      i. If **Local Connections only** is checked, MLAdmin is only accessible on the server hosting MLWeb by typing **http://localhost/admin** into a browser.

      ii. If unchecked, MLAdmin can be accessed remotely by typing **http://[Machine Name or IP]/admin** into a browser.

b) Set the number of hours which backups and other database operations remain accessible through MLAdmin; they are deleted after the set time expires.

Note

Regardless of the Local Connections only toggle setting, MLAdmin cannot be accessed remotely unless a new firewall rule has been set. See section *2.2 Firewall Configuration*.

5. After the Admin Dashboard Configuration page has been configured, click **Save.**

MLAdmin can now be accessed through the Administrator account established in step 1. Continue to section *4.2 Admin Controls* for a list of controls available in MLAdmin.

4.2 MLAdmin Controls

MLAdmin consists of a menu on the left containing tabs for each functionality, and a workspace on the right that displays the selected tab's fields. Menu tabs and subtabs include:

Home – The initial landing page for MLAdmin, displaying current MLSuite Version and real time monitoring.

Dashboard Config – Allows for enabling or disabling remote access to MLAdmin.

Background Jobs – Lists operations recently run or currently running in the background, such as restorations and backups.

Sites – Lists all configured sites. Expanding this menu option provides access to individual sites. Clicking on these sites access their database operations.

Configuration – Provides configurations MLAdmin and database configurations. Expand to access additional configurations. For more detailed info on configuring email, see section *4.3 Configuring Email Sending Information in MLAdmin.*

Continue to the next section of this guide for instructions on creating a new database and Site in MLAdmin.

4.3 Configuring Email Sending Information in MLAdmin

The Email Sender Configuration configures how MLServer connects to the mail server.

1. Enter the SMTP Server host name, fully qualified domain name or IP Address as the **SMTP Host**.
2. Specify the **Port**. The default SMTP port is 25
3. If user authentication is required for relay, enter the **SMTP User** and **SMTP Password**. If the connection is only restricted by IP address then the User and Password are not necessary.
4. The **From Name** and **From Email** will be displayed to the recipient. Generally, the email is the SMTP User name followed by the domain originating the email. In most cases it does not matter what name is entered here; however, some SPAM filtering software may reject mail which does not have a valid From Email identified.
5. Select the SSL Method form the drop-down list. Select **None** for SMTP connections, and **SSL or TLS** for SMTPS connections.

6. Once the information is complete the email sender configuration can be tested. To test the email settings, enter an email address in the Test Email Address field, and click the **Send Test Message** button. A notification indicates if the email was sent successfully, or not. Common error messages include:

   a) **Host Not Found** – Check the Server entry for correct server name.

   b) **Connection Timed Out** – Check the Port setting or check for proper firewall configuration.

   c) **Incorrect Authentication Data** – Check the User Name and Password.

Once all the necessary information has been entered click **Save.**

4.4 Passwords Configuration

The Passwords Configuration workspace allows for viewing and updating passwords for the four required system users as well as the Process Account credentials. Click the icon in each field to display the password. Click the icon in each field to hide the password.

SYSDBA – The Firebird SYSDBA user. Default password is **masterkey**. This password can be changed here.

POSTGRES 'ml_user' – An internal user for accessing PostgreSQL data. This password can be changed here.

POSTGRES 'postgres' (admin) – The PostgreSQL super user created during installation of PostgreSQL. It is an internal account used for performing database operations. The password can **ONLY** be changed from pgAdmin, a PostgreSQL tool. See *Changing the POSTGRES admin passwords* for more details.

MLUSER – An internal Firebird user. This password must be changed from the fb_users_setup.cmd file (see note in section *4.1 Installation Setup*).

Windows process User Name – The user name for the Process Account in MLServer. This account does not require admin rights. It is used for running import scripts, notifications, scheduled outputs and other operations that leave the MLSuite software to interact with the server. If not set up and specified these operations will not work.

Windows process Password – Password for the Process Account in MLServer. When changing this password, it must be changed on the windows account and then changed here to match.

Changing the POSTGRES admin passwords

The POSTGRES admin passwords are used for internal linking of MLServer with the MLWeb and PostgreSQL applications. Mismatched passwords will lead to system failure. In order to change these passwords from their initial values, it's important to do so in the correct order.

1. For the POSTGRES admin password, log into pgAdmin and change the password. Please see the current pgAdmin documentation for details.
2. Change the password in the MLAdmin Passwords workspace to match the new password. Do not forget to **Save** the changes. The system may take a minute to update its internal settings and should continue function normally afterwards.

Note

In MLWeb 2021, the Firebird admin user account doubled as the admin user account for MLWeb. In MLWeb 2023, these accounts are separate.

5. MLWeb Configuration

Once MLSuite has been installed and MLAdmin has been configured, further configuration for MLWeb is possible.

5.1 Creating New Databases and Sites

Databases and sites are created through MLAdmin. Log into MLAdmin using the credentials specified in step 1 of the installation set up completed in section *4.1 Installation Setup*.

Hint

MLAdmin is accessed on the machine hosting MLWeb locally at **http://localhost/admin**, or remotely at **http://[MachineName or IP]/admin.**

5.1.1 Creating a Database

1. Navigate to **Configuration|Databases**.
2. Click **Add** to add a new blank entry.
3. The database *.gdb* file and database name are initially configured in the **Database Path** field.
   a) Enter the path where the Database's *.gdb* database file will be located.
   b) Define the name of the database followed by .gdb. The **.gdb** extension is required.

If the database files are located on the drive (**D:**) under **Databases>Firebird**, and the database is to be titled **example**, enter **D:\Databases\Firebird\example.gdb**.

When creating and naming a new database the following character restrictions need to be considered:

- Only include alphanumeric characters as well as the underscore "_" symbol in the database name.
- All database names must start with an alphanumeric character.

4. For the database to be accessible and to create the associated site, it must be active. Check the **Active** checkbox and click the **Save** button for the database entry.

Hint

If the Active checkbox cannot be checked, confirm that the SYSDBA user name and password have been defined.

5. Once the new Database has been saved, **Save** the Databases Configuration workspace. The database *.gdb* file is then created in the location specified.

The database can now be licensed, and the site can be created.

See section *6.2 Database Licensing* of this guide for more information on Database Licensing.

Continue to *5.1.2 Creating a Site* for instructions on creating a new site, then to *5.1.3 Site Management* for activating the Postgres database for 3D functionality.

5.1.2 Creating a Site

After a database has been created and is active, its site can be created.

1. Navigate to the Sites tab on the left menu, then click **Add New Site** from the dropdown menu.
2. From the Add New Site page, enter the database into the **Database** field to automatically populate the **Site Name** field with the database name. The site name is used as part of the site's URL.
3. Click the **Save** button for the site entry. The site is now created.
4. **Before the site can be accessed, the server where MLSuite is installed must be restarted.**

> **Note**
>
> If MLWeb is published on the internet, it is strongly recommended to enforce HTTPS protocols for it. See section *2.8 Binding SSL Certificate.*

Continue to *5.1.3 Site Management* for details on creating the PostgreSQL database and patching databases when necessary.

5.1.3 Site Management

After saving the Site workspace additional site management buttons become available.

Creating Postgres Database

The first time a site is created in MLAdmin, the Postgres database must also be created. The Postgres database is required in projects using 3D data.

From the **Sites** workspace, click **Create Postgres Database**. A message will be displayed once the PostgreSQL database is created successfully.

> **Hint**
>
> The Postgres Database only needs to be created once; the button only appears for the first site created.

Patch Database

Ensure that the Firebird and Postgres database versions are current. If a patch is required, the **Patch** buttons become active and should be clicked.

If the required versions of the databases already exist, the Patch Database buttons are disabled.

5.1.4 Accessing MLWeb

Once a site has been created **and the server has been rebooted**, MLWeb can be accessed by entering one of the following into the web browser:

1. **[domain name]/mlweb/[site name].** For example: geomonitoring.com/mlweb/demolog.

2. **[ip address]/mlweb/[site name].** For example: 172.25.10.75/mlweb/sitename.
3. **[machine name]/mlweb/[site name].** For example: bobscom/mlweb/sitename.
4. **localhost/mlweb/[site name]** (if browser is running on same server as MLWeb).

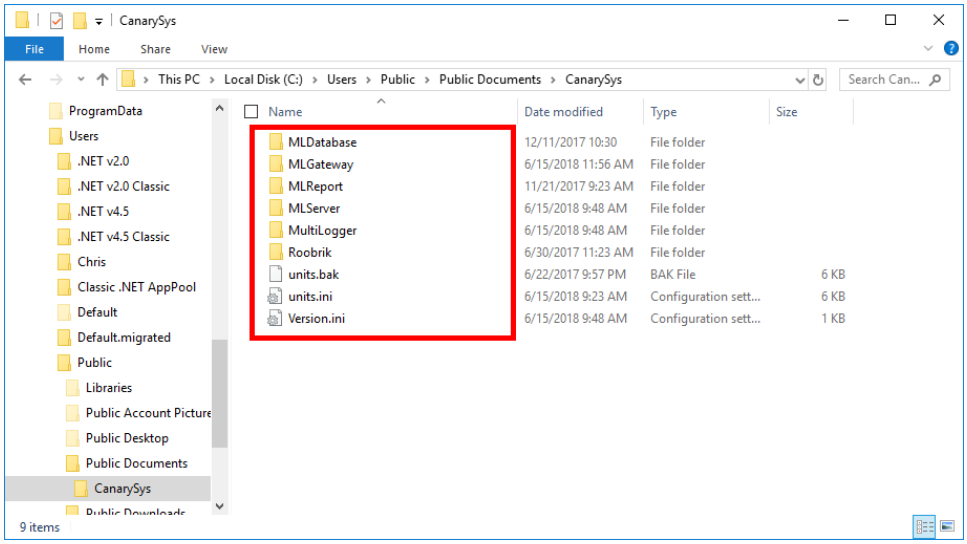
> **Note**
>
> Canary Systems does not recommend installing MLWeb on the same server as the browser.

5.2 Working File Path

MLSuite 2018 and newer uses the Public Documents folder (typically **C:\Users\Public**) to store system configuration files. This path is known as the **[Public Documents]** or **[Shared Docs]** folder.

The **[Shared Docs]** path is used for all system configuration files. These include additions or modifications to the MultiLogger programming options, datalogger configuration files and collected data files, among others. Only program files and other non-customizable files are now stored in **Program File\MultiLogger** (or an alternate installation path, if it was changed).

To find the system configuration files for each application, open **My Computer** and then the **Local Disk (C:)** drive. Next, browse to **Users | Public | Public Documents | CanarySys**. Notice the subfolders in the CanarySys folder for each application installed.



CanarySys Folder Location

It is recommended that the CanarySys folder be relocated to save on resource utilization as well as for ease of access.

The CanarySys folder can be permanently moved to a designated location. This is done by uninstalling MLSuite, moving the folder to the desired location and making adjustments in the Registry Editor. It is not advised to make changes to the Registry Edit without expert guidance. Contact Canary Systems for assistance permanently relocating the CanarySys folder.

5.3 Active Directory Integration

MLSuite allows both Windows and Azure Active Directory user accounts to log into a project's database in MLWeb. The following subsections describe the steps needed for integration with each of these Active Directories.

> **Note**
>
> Some additional configuration settings for the database hosting server are required during installation. Consult your IT department for assistance before beginning the process.

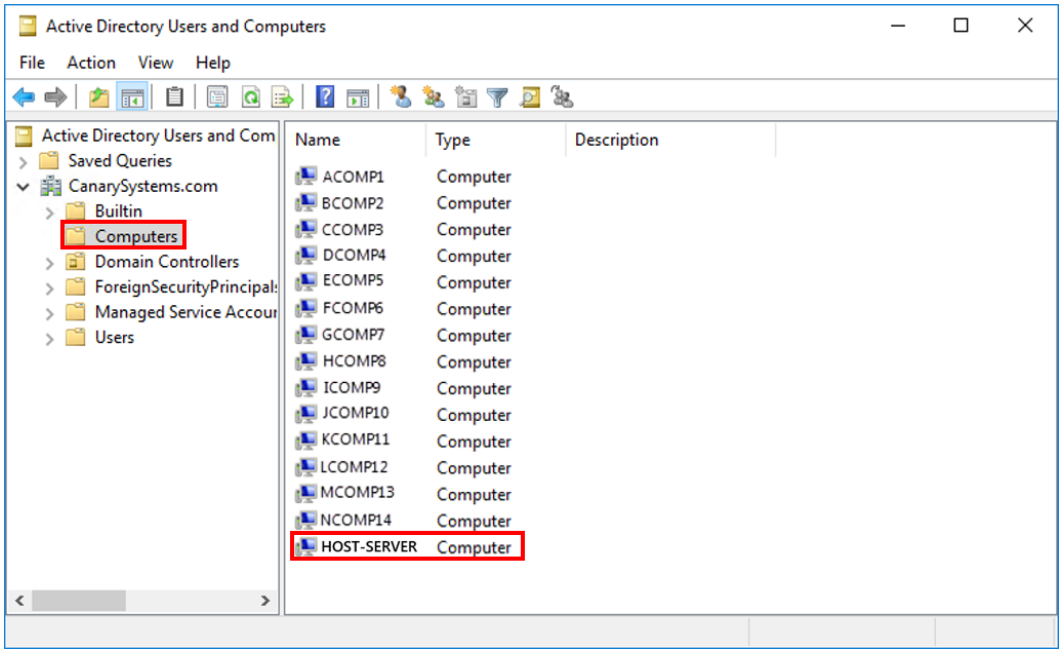
5.3.1 Windows Active Directory

Configuring Active Directory

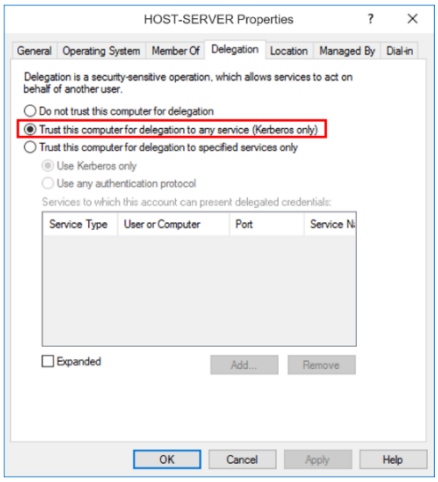
> **Note**
>
> Active Directory must be installed prior to configuring Active Directory integration with MLSuite.

Begin by configuring Active Directory on the **Active Directory server**.

1. Open the Active Directory Users and Computers application from the Start menu and browse to the Computers group under the local domain.



2. Double-click on the server where MLWeb is installed (HOST-SERVER in this example) to bring up its Properties dialog box, as seen in the figure below.



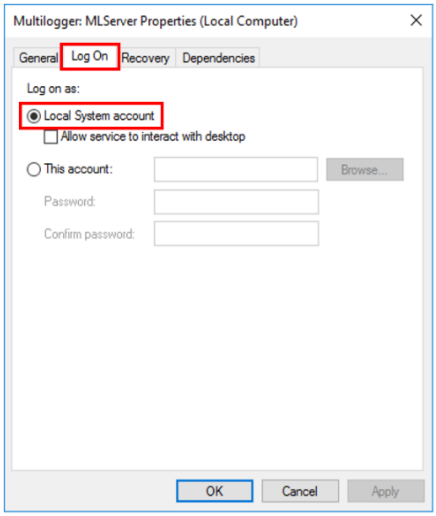
3. From the Delegation tab, select the option **Trust this computer for delegation to any service (Kerberos only)**, then click **OK**.

Note

If MLWeb and MLServer are hosted on different servers, this process needs to be completed for both servers.

Configuring MLServer

1. On the database hosting server, open **Services** (available from **Start | Windows Administrative Controls**) and right-click on the MLServer entry.
2. From the **Log On** tab, ensure that MLServer is logged in via the **Local System account**, as seen in the figure below.



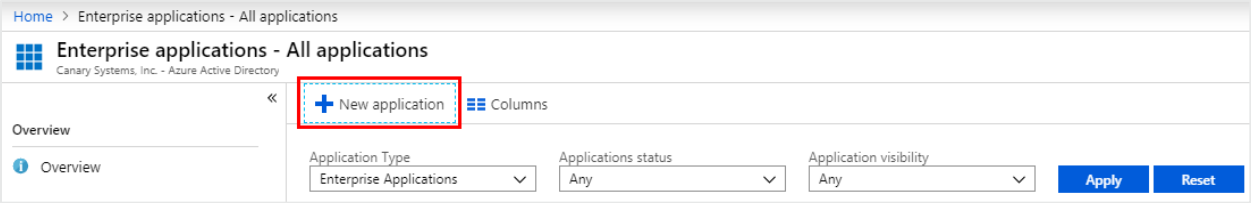
5.3.2 Azure Active Directory

Before you begin, log into the online Azure portal.

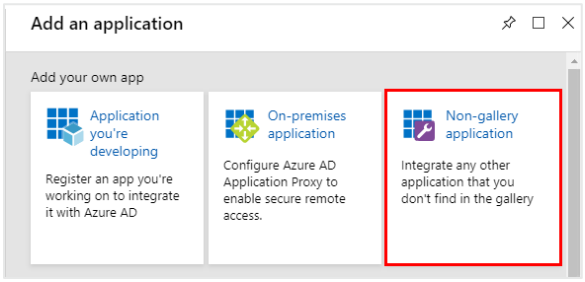
Configuring Active Directory

1. Add MLSuite to the Azure portal as an Enterprise application.

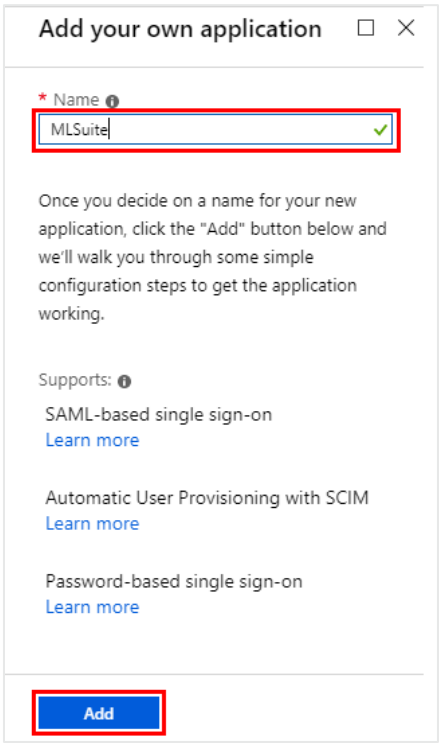
   a) Go to the **Enterprise Applications** menu and select **New application**.



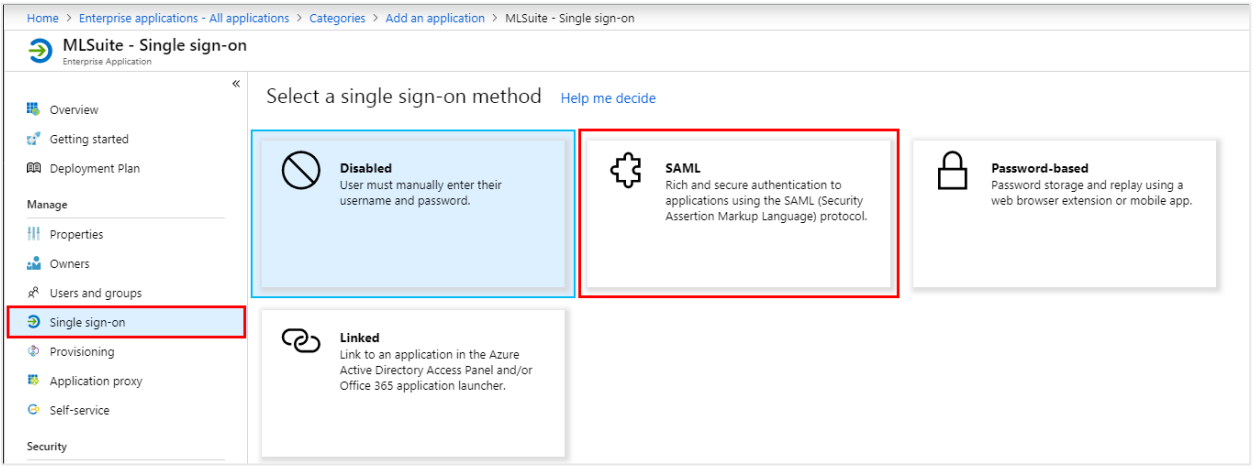
   b) From the **Add an application** window, select **Non-gallery application**.



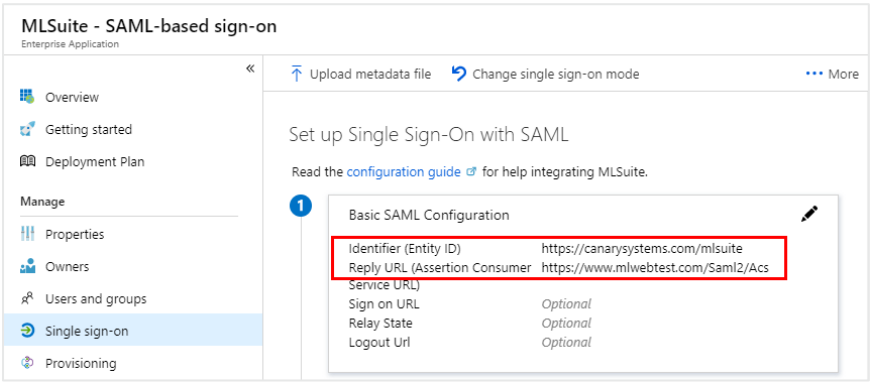
   c) On the right side of the portal, enter application **Name** (MLSuite) and click **Add**:



d) Select **Single sign-on** from the left navigation, then select **SAML**:

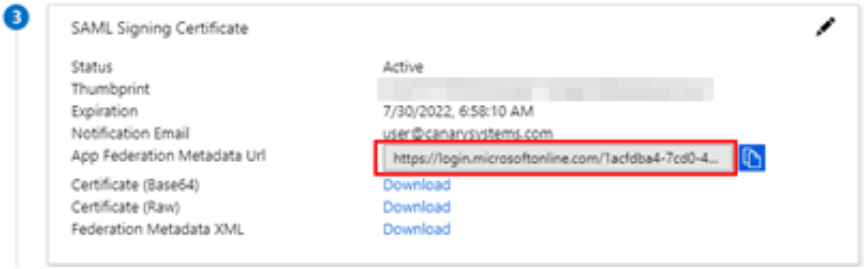


e) Fill in the fields highlighted in the figure below with the details outlined in the table (Identifier SHOULD end with "/" when entered)

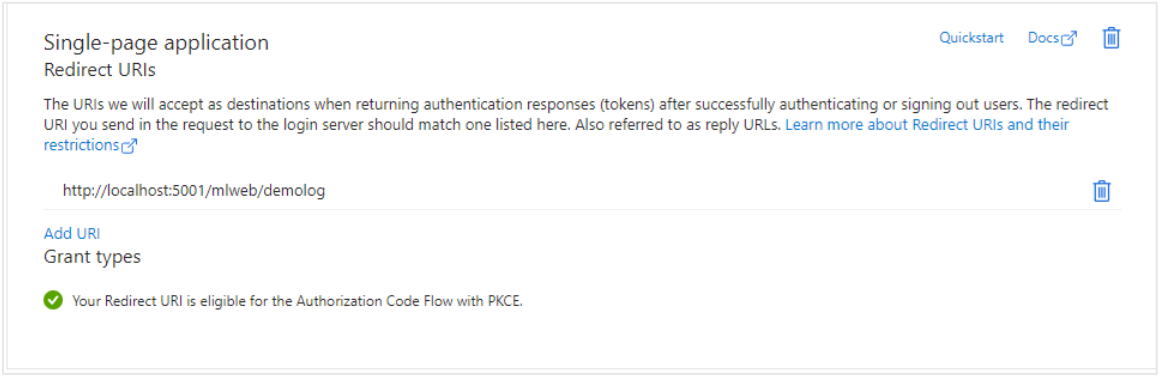


Identifier (Entity ID)	https://canarysystems.com/mlsuite/
Reply URL (Assertion Consumer Service URL)	https://<your mlweb2023 domain>/mlweb/<your database name>

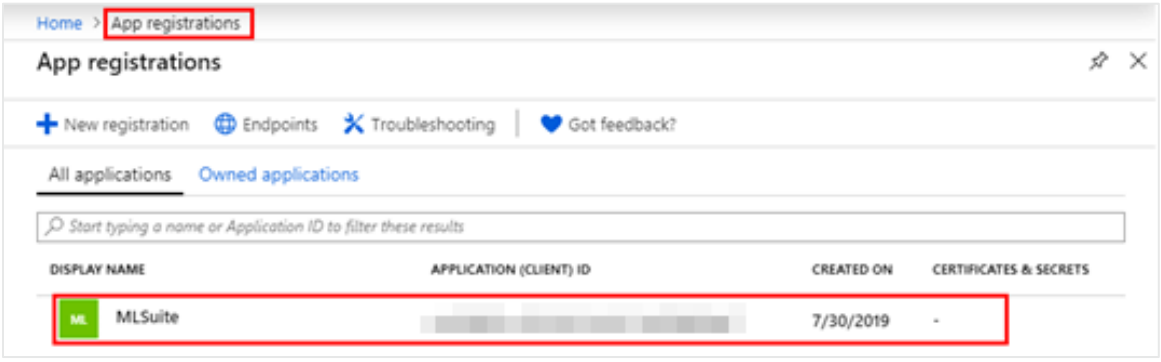
A Metadata URL is generated. Copy and save the URL as it is needed later in the configuration process.



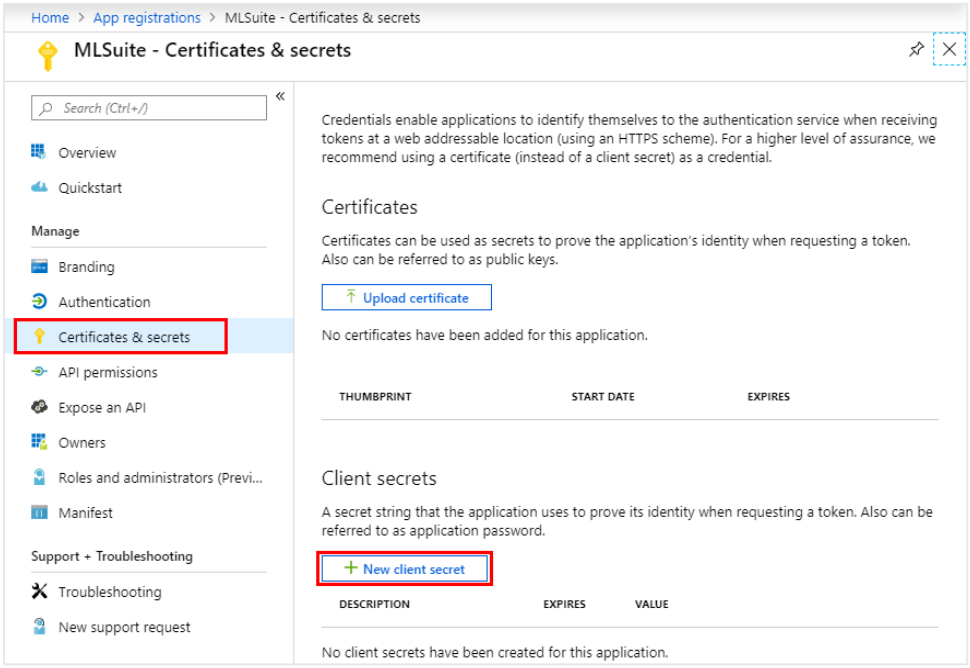
2. Authenticate platform
   a) Go to the Authentification section of your enterprise application
   b) Click **Add platform**
   c) Select **Single-page application**
   d) In the **Redirect URIs** section, fill in your redirect URI (https://<your mlweb2023 domain>/mlweb/<your database name>) and then click **Configure**



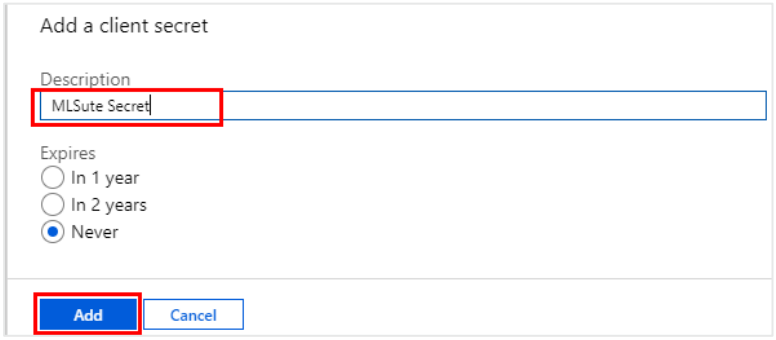
3. Configure MLSuite application:
   a) Go to **App registrations** and open the newly created application.



b) Select **Certificates & secrets** from the left-navigation and click **New client secret**:

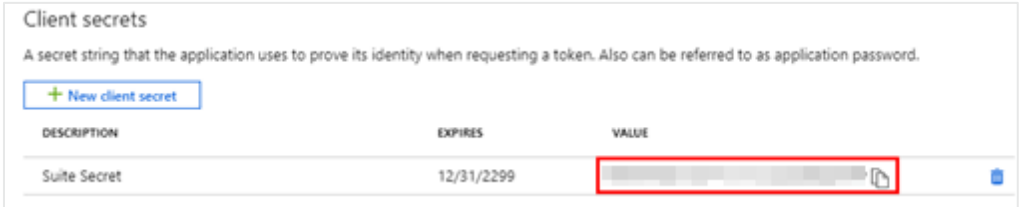


c) Give the secret a **Description** and click **Add**.



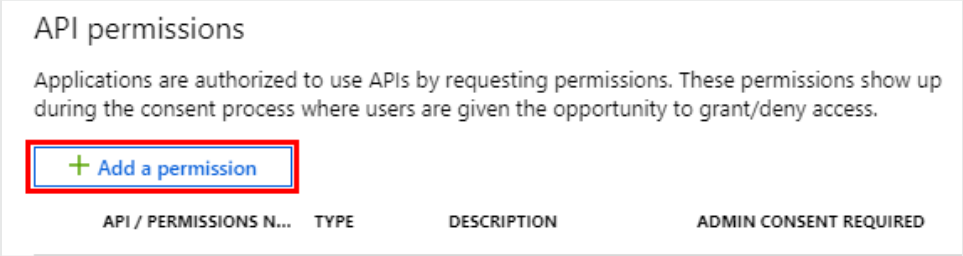
Note

Make sure to copy and save the created secret as it is required for configuring MLWeb as described in the following section.

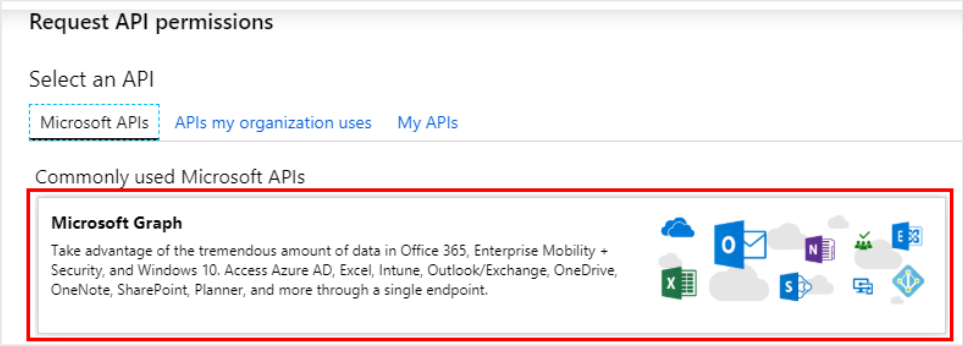


4. Add Permissions

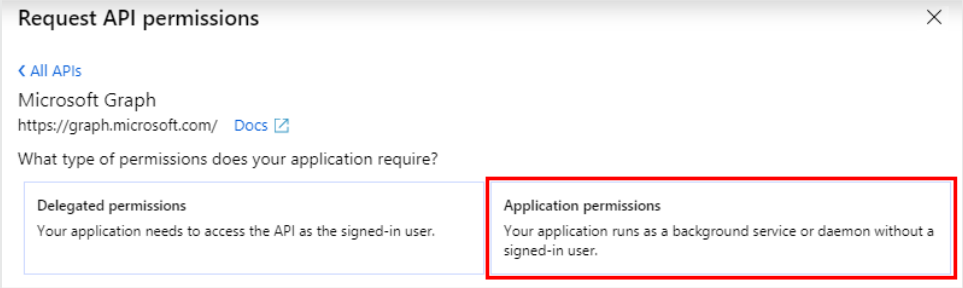
a) Select **API Permissions from the left-navigation and** click **Add a permission**.



b) From the Microsoft APIs tab click **Microsoft Graph**



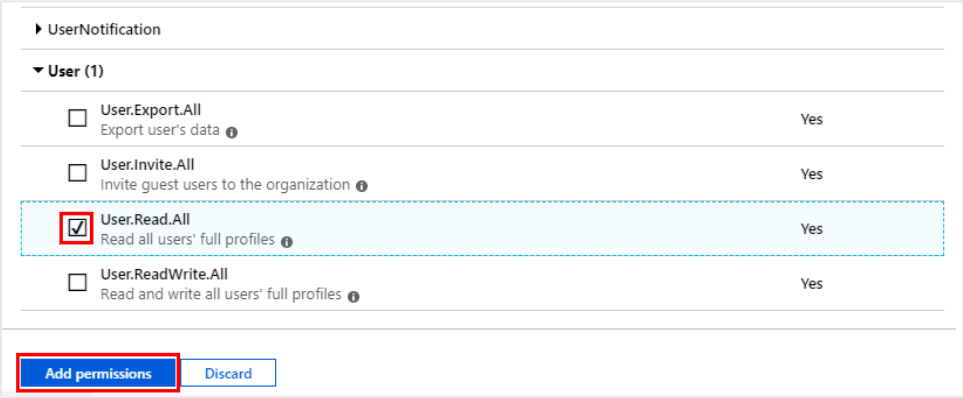
c) Click **Application permissions**



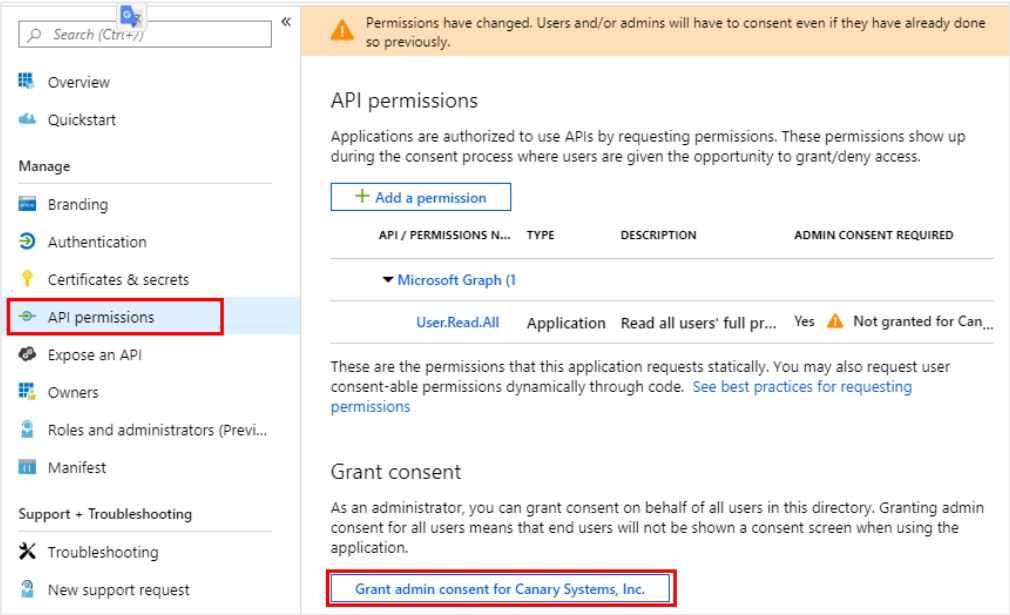
Note

In MLWeb 2023, both Application and Delegated permissions are required for Azure Active Directory setup. At the end of the steps given here for setting up Application permissions, you will be redirected back to this step to start the process over for Delegated permissions. The process for both is the same.

d) Scroll down permissions tree to locate and expand **User** category, check **User.Read.All,** and click **Add permission**

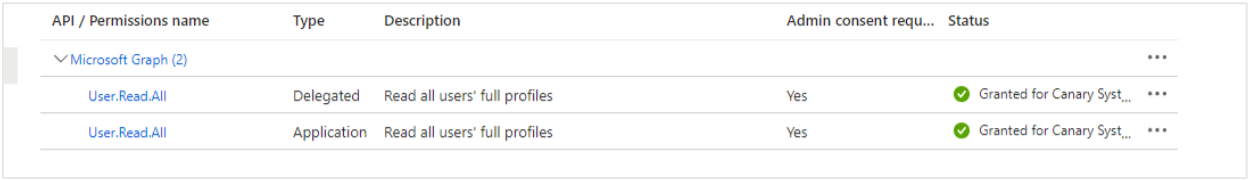


e) Return to **API permissions** area and click **Grant admin consent for <Your Company name>**:

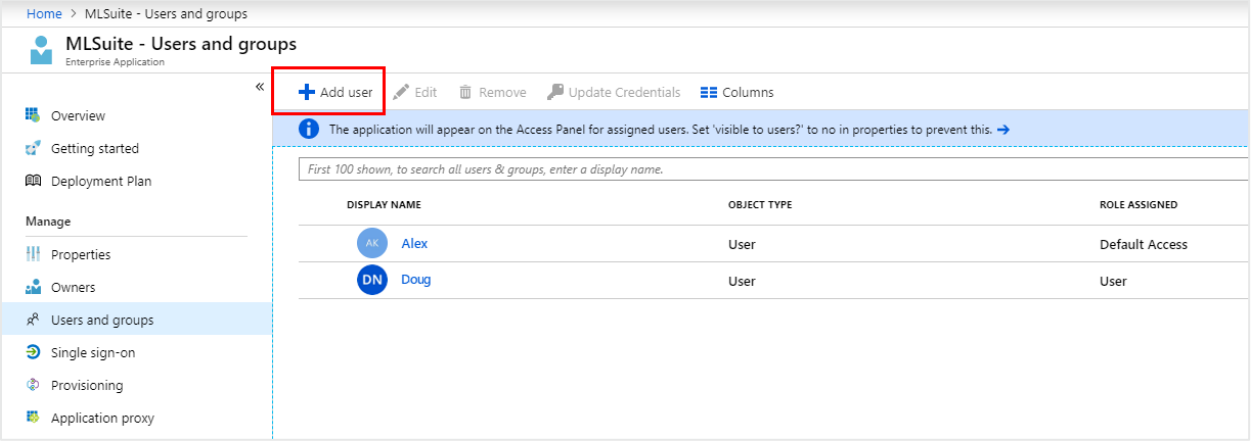


f) **Return to step c) and go through the steps again for Delegated permissions**.

g) Ensure the desired permissions have been granted.



5. Due to Azure security rules, users must be authorized through the Azure portal to be able to log into MLWeb using their Azure credentials. This is done by navigating to the **Enterprise** applications area and



selecting **Users and groups** and **Add users**.

MLWeb Configuration

1. Go to MLWeb folder and open **MLSuite.config.json** file. (C:\inetpub\MLWeb)
2. Create the **samlOptions** entry:

```
"samlOptions": {
    "domainPrefix": "azure",
    "metadataLocation": "<metadata location>"
}
```

   a) **The domainPrefix** parameter should be "azure"

   b) Use the Azure SAML metadata location URL saved from SAML Configuration, as described in the previous subsection, for the **metadataLocation**.

Note

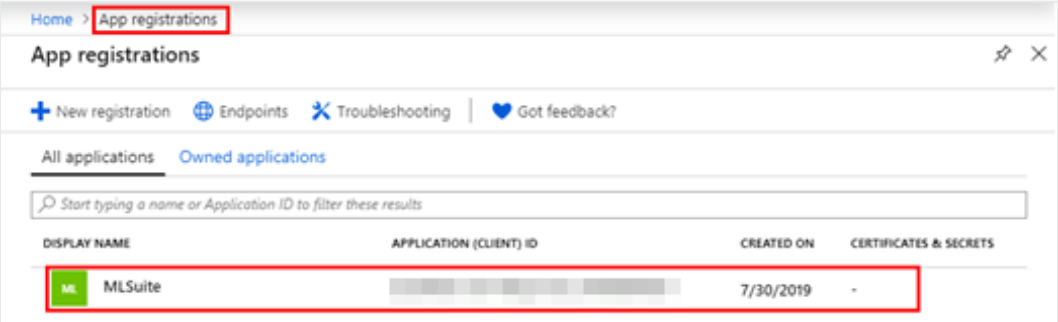
The metadata already contains the security certificate; you do not need to copy/install it manually.

3. Create the **azureActiveDirectory** entry using the details outlined in this step:

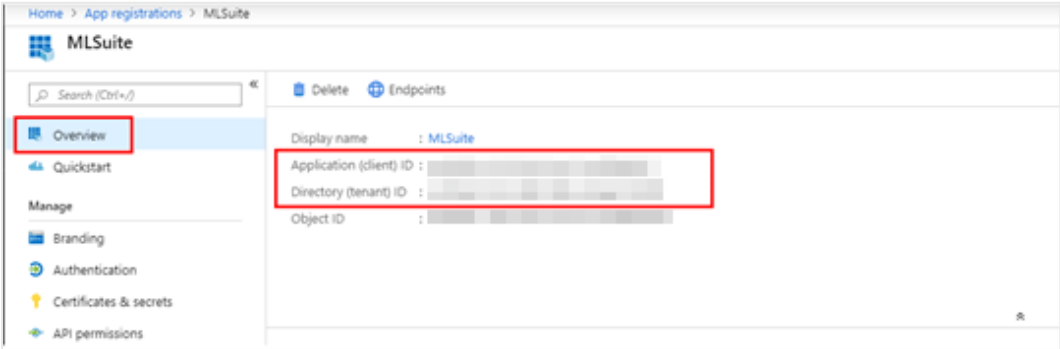
```
"azureActiveDirectory": {
    "credentials": {
      "graphBaseUrl": "https://graph.microsoft.com/v1.0",
      "loginBaseUrl": "https://login.microsoftonline.com",
      "tenantId": "<tenant id>",
      "clientId": "<client id>",
      "clientSecret": "<client secret>"
    }
  }
```

graphBaseUrl	Base URL for Graph API
loginBaseUrl	Base URL for API login
tenantId	Azure tenant Identifier
clientId	Application Identifier
clientSecret	Application secret

a) To get **tenantId** and **clientId,** return to the Azure Enterprise Application, go to **App registrations** and select the previously created application.



b) Select **Overview**:



Note

The base URLs above are valid for public Microsoft Azure Cloud. For a private Cloud, they can be different.

With the above steps completed, users should now see a **Log in with Azure User** option on the MLWeb 2023 login screen. For information on Adding Active directory users in MLWeb, see the ***MLWeb User's Guide 2023***.

6. Licensing

MLSuite uses two separate forms of licensing: **Gateway Licensing** and **Database Licensing**.

- Gateway Licensing allows MLGateway to communicate with data loggers and collect and import their data.
- Database Licensing activates the automation processes for the database, such as imports, alarms, notifications, and calculations.

The processes for Gateway and Database Licensing are detailed in the following sections.

6.1 Gateway Licensing

MLGateway must be licensed and activated on each machine where MLGateway is running. This can be done either through MLWebHardware or through MLGateway.

6.1.1 Licensing MLGateway through MLWebHardware

MLWebHardware is accessed on the machine hosting MLServer locally at **http://localhost/mlwebhw**. It is accessed remotely at **http://[Machine Name or IP]/mlwebhw**.

Log in to MLWebHardware using the admin user and navigate to the License form located under Server on the left side menu.

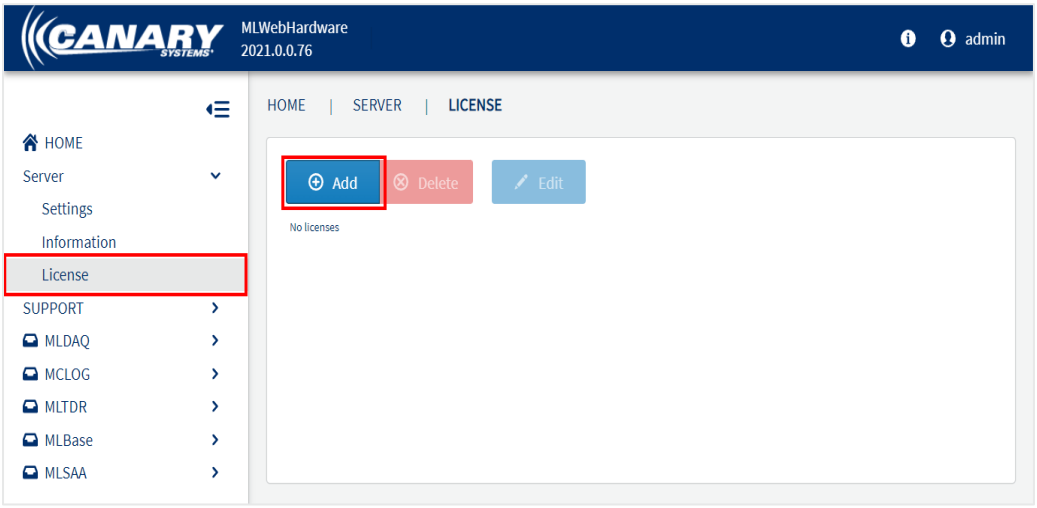
Hint

The default username and password for MLWebHardware is “admin.” The password can be reset once logged in.

For more information on accessing and navigating MLWebHardware, refer to the ***MLWebHardware User’s Guide*** (**canarysystems.com/support/users-guides**).

Entering the License Key

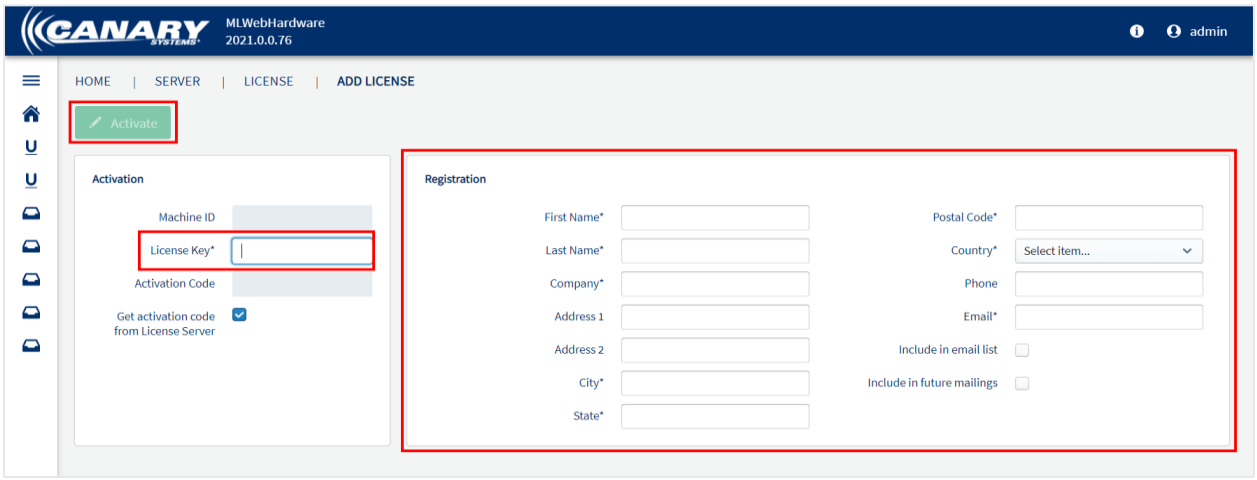
From the License form in MLWebHardware, click **Add** to populate the license Registration form.



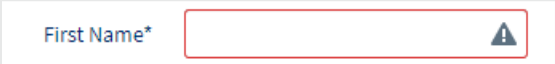
Enter the Registration information and the **License Key** that was supplied with the MLSuite documentation into the License Form.

Once a valid license key and all required fields have been filled in, click **Activate** to obtain the Activation Code and activate the license.

> **Note**
>
> Internet connectivity is required for the automated Authorization Code delivery to function. If you are unable to use the electronic system, call **Canary Systems Software Support** at **1-603-526-9800** ext. **2** to authorize the software over the phone.

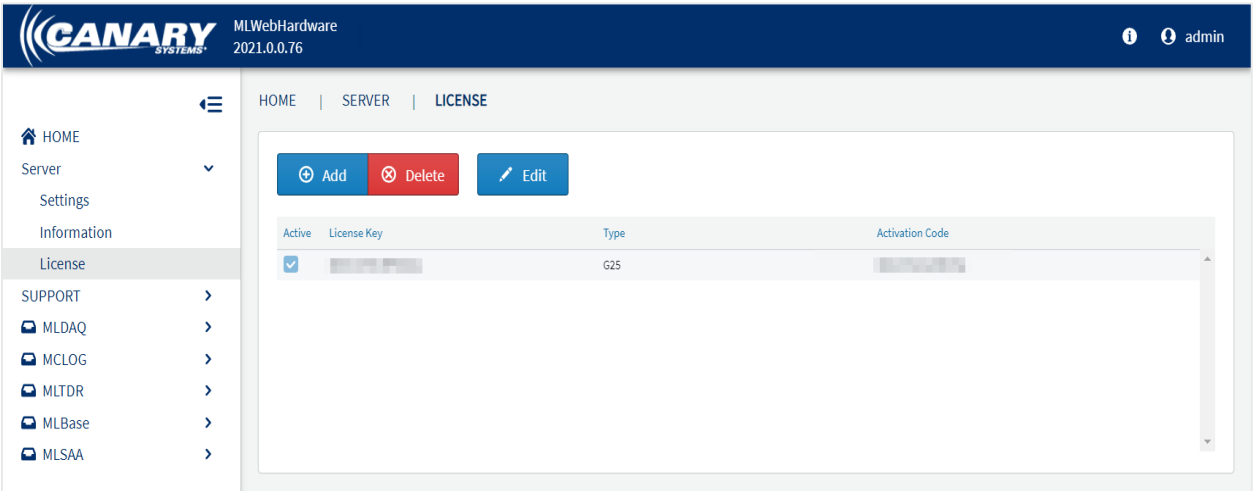


If a mandatory field is missing information, the field will be outlined in red to indicate missing information.



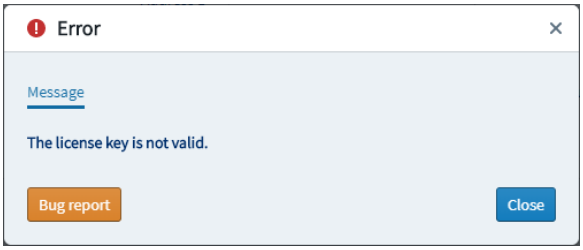
Activation Response in MLWebHardware

Upon successful activation, MLWebHardware returns to the License form where the active license and Activation Code can be viewed.



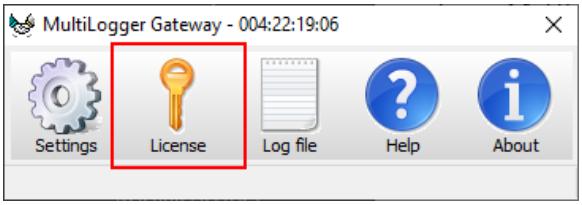
If an invalid key is entered, or the license is unable to activate for any other reason, a message with an option to show details appears in the bottom right corner of MLWebHardware.

Clicking **Show Details** displays additional information pertaining to the error. If the error is believed to be a bug, a **Bug report** can be submitted directly through MLWebHardware by clicking the **Bug report** option.



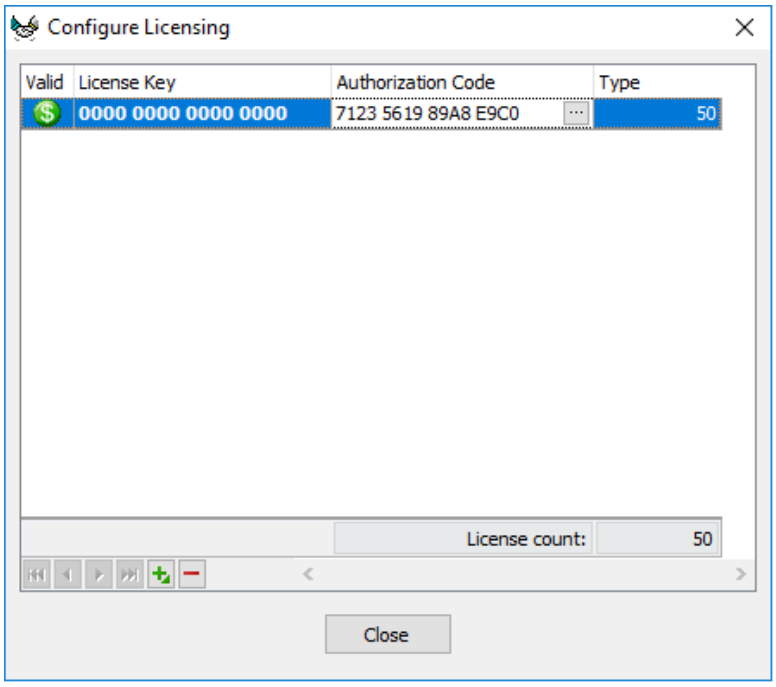
6.1.2 Licensing MLGateway through MLGateway

To activate the Gateway license through MLGateway, launch MLGateway and then click the **License** button to open the **Configure Licensing form**.



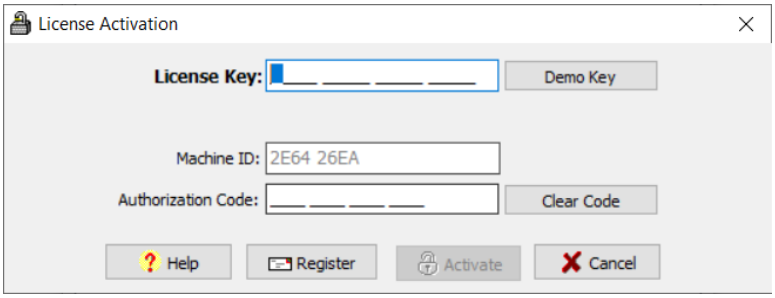
Entering the License Key

From the **Configure Licensing** form, click to add a new record. This opens the **License Activation form**, as shown in the figure below.



At startup, the **License Key** is blank. Enter the key supplied with the MLSuite documentation. This is usually found on the Final QA Report form supplied by Canary Systems.

> **Hint**
>
> Click the **Demo Key** button to obtain a 30-day demo key.

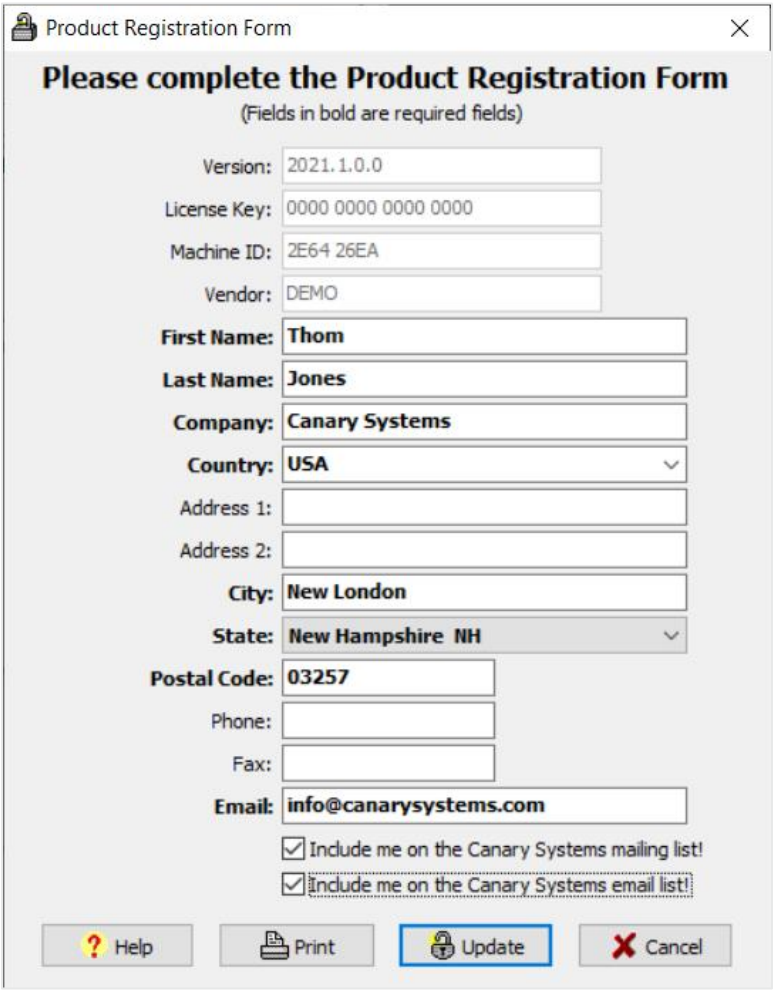


Obtaining the Authorization Code (Activation)

After entering the License Key, MLGateway must be activated.

Click the **Activate** button on the License Activation panel to begin the activation process.

If registration information has not been entered on this computer, the **Product Registration** form must be completed.



The **First Name**, **Last Name**, **Company**, **Country**, **City**, **State**, **Postal Code** and **Email** fields are required, and filling in the remaining fields is recommended.

Once the form is complete, click **Activate** (the button shows **Update** if registration information has been previously entered) to attempt connecting to the Canary Systems Authorization Code Server to obtain your Authorization Code.

Note

Internet connectivity is required for the automated Authorization Code delivery to function. If you are unable to use the electronic system, call **Canary Systems Software Support** at **1-603-526-9800** ext. **2** to authorize the software over the phone.

If all fields are entered correctly, the License Key is valid, and the computer has Internet connectivity, the Authorization Code is uploaded into the appropriate form field and the software is activated.

Activation Response in MLServer

The following messages may display during the activation process:

- **Authorization Granted – Thank you for registering.**

  You have succeeded in activating the software.

- **A required field is blank.**

  One of the required fields is not filled in. The fields in bold are required fields. Click OK to return to the Product Registration form, complete the form and click Activate again.

- **The License Key is not valid.**

  The format is incorrect, or the number is not valid. Verify that the License Key is entered correctly.

- **Unable to reach the Canary Systems Authorization Server.**

  This is usually due to lack of Internet connectivity on the computer attempting to activate the software. This may be due to Firewall configuration or network security configuration. If applicable, contact your systems administrator for assistance in resolving the issue.

  If you are unable to resolve the connectivity issues, the product must be registered offline. First, you will need to restart the software, then click **Register** and **Print** on the Product Registration form after it is completed. Email this form to Canary Systems.

  You will receive an Authorization Code that must be manually entered into the Software Activation form to activate the software. You may also call or email Canary Systems to obtain your Authorization Code.

- **Software Authorization inactive.**

  This may be the result of incorrect configuration of the authorization server. Double-check the License Key entry. If it fails again, contact your vendor or Canary Systems directly.

- **Demo Authorization Denied – Too many requests.**

  MLGateway can be activated with a demo Authorization Code, which lasts 30 days. This period can be extended, but by default it expires after a single period. Contact Canary Systems if you require a longer trial period.

- **Authorization Failed.**

  There was an error in the negotiation for the Authorization Code. Try clicking **Activate** again and if it fails again, contact Canary Systems for further direction.

6.2 Database Licensing

Database Licensing is based on the following items being present (saved) in the database:

- Number of Class Instruments
- Number of unclassed Instruments (certain instruments count as more than 1 license)
- Number of standalone Data Elements (not associated with an instrument)
- Number of 3D Data Sources

If the total number of items exceeds the number of available licenses, an additional Database License Key must be purchased. The active/inactive status of instruments or elements is not considered when determining the number of required licenses. Contact Canary Systems for more information on purchasing license keys.

Database Licensing is accessed and managed through the **Databases** workspace in **MLAdmin**.

From here a list of currently configured databases, including the number of used licenses verses the number available licenses is visible.

Licenses are configured through the Licenses workspace.

To add or view licenses, open the **Database Licenses** workspace by clicking the **License** button while in the Databases workspace.

6.2.1 Adding Licenses

The Licenses workspace lists all active licenses for the database. It allows for adding new licenses and editing and deleting existing licenses.

Click **+ Add** button to add a new license or **Edit License** button to edit an existing one. This opens the **License Activation** form.

Fill in all required fields. The Machine ID field is read-only and is populated automatically. With the License Key entered, the Activation Code populates automatically. License Keys are obtained from Canary Systems directly.

6.2.2 Online activation

If the server is connected to the Internet, enabling the option **Get activation code from License Server** allows for obtaining a valid Activation Code from the Canary Systems licensing server automatically. In this case, clicking **Save** should activate the license and display a message indicating a successful activation.

6.2.3 Offline activation

If no Internet connection is available—or firewall rules or other issues prevent successful communication with the activation server—an Authorization Code will need to be requested. Contact Canary Systems to request an Authorization Code to activate the license. Please have your **Database License Pack Key**, **Machine ID** and **Database Alias** ready. After obtaining the code, uncheck the **Get activation code from License Server** option. This will enable the Activation Code field and allow you fill in the code you have obtained from Canary Systems. If it is correct, clicking **Save** should activate the license and display a message indicating a successful activation.

When successful, the Licensing Activation page updates with the License Key, Authorization Code and license type (showing the number of activated licenses). The **Used/Licensed** field of the Databases Configuration of MLAdmin will update as well. Click **Back** to return to the previous screen.

6.2.4 Deleting Licenses

Click the **Delete** button to delete a license record. **PROCEED WITH CAUTION.** Deleting a license record on a server that is not connected to the Internet, or whose settings may prevent it from communication with the Canary Systems licensing server, may disable the license key.

If the server is online, deleting the license will automatically update the license records in the Canary Systems activation database, and free up the license key to be used again later (on the same or a different server and database). If the server is not able to communicate with the activation database via Internet, you will have to contact Canary Systems Support and request revoking the license activation.

7. Uninstalling MLSuite

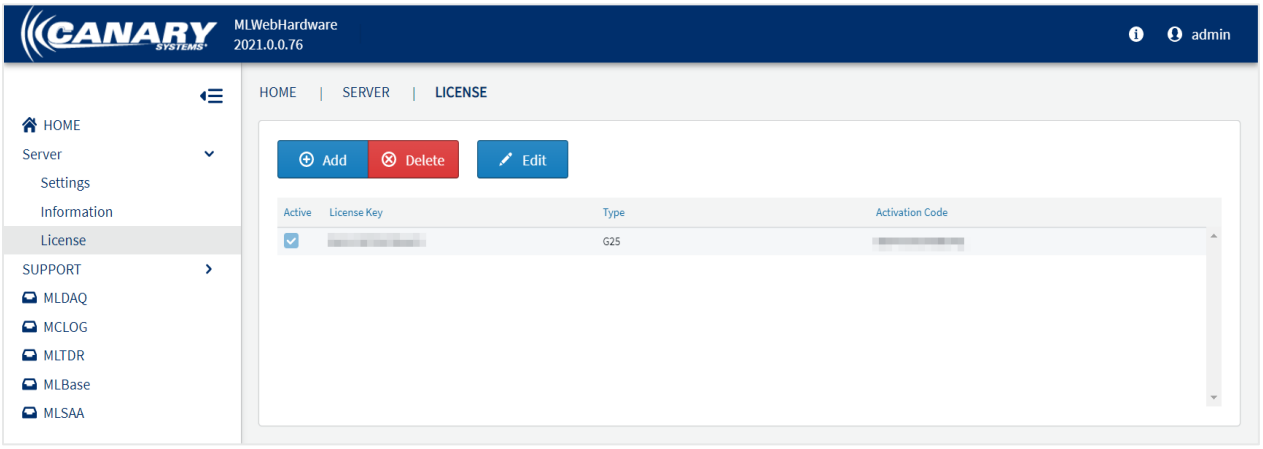
7.1 MLGateway License Deactivation

Before removing MLSuite, the Gateway License for MLGateway must be deactivated on the current machine to allow it to be reactivated on another. This can be done through MLWebHardware or MLGateway, follow the steps outlined in this section.

Once the license for MLGateway is deactivated, the licensed databases in MLAdmin must also be deactivated. This is described in section *7.2 Database Licenses Deactivation*.

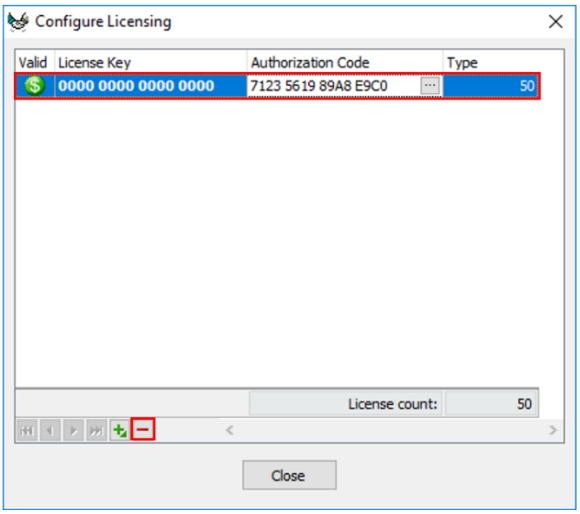
7.1.1 License Deactivation through MLWebHardware

Open **MLGateway** and click the **License** tab. This brings up a list of configured licenses, as shown in the figure below. Select the license you wish to delete and click the **Delete** button to deactivate the license.



7.1.2 License Deactivation through MLGateway

Open **MLGateway** and click the **License** button. This will bring up the **Configure Licensing form**, as shown in the figure below. Highlight the license and click the Delete Record button to deactivate the license.



7.2 Database License Deactivation

In addition to deactivating the Gateway License, all database license keys should also be removed. This will allow them to be preserved and reused later.

Open **MLAdmin** and naviagate to **Configuration|Databases.** Select a licensed database and then navigate to the Licenses workspace. Make note of the License Key for future use, then click **Delete**.

Repeat for all database license keys.

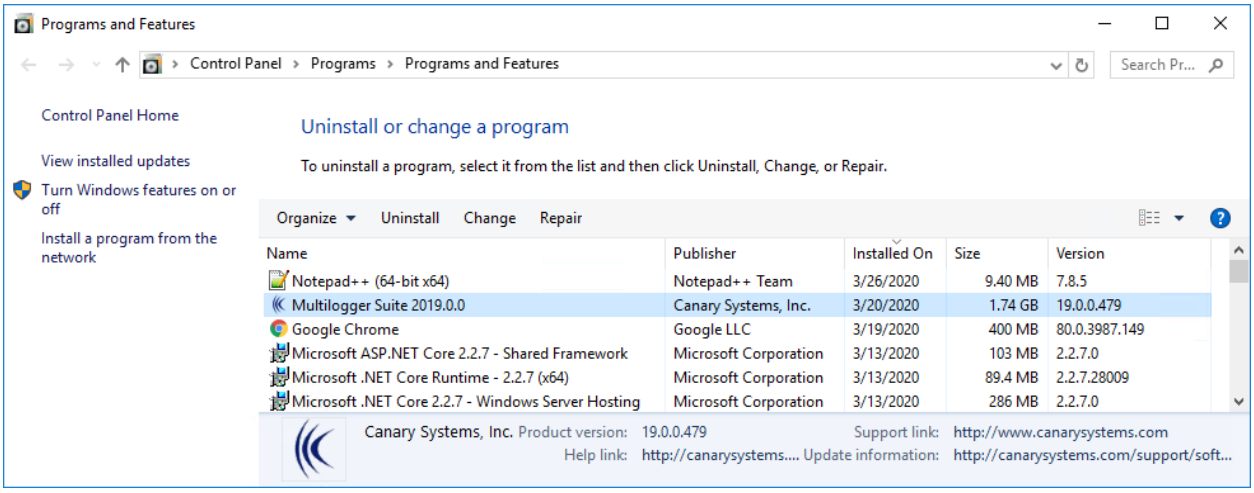
7.3 MLSuite Removal

Note

The following example shows the removal of MLSuite 2023 on a 64-bit Windows environment with Admin user rights. The removal process on your operating system may vary, especially with desktop operating systems such as Windows 11.

7.3.1 Uninstalling Using the Control Panel

To remove MLSuite, open the **Programs and Features** menu of the Control Panel. Locate **MLSuite** in the list of programs and click **Uninstall** to begin the removal process.



A message box is displayed, asking to confirm removal of MLSuite. Click **Yes** to continue to uninstall.

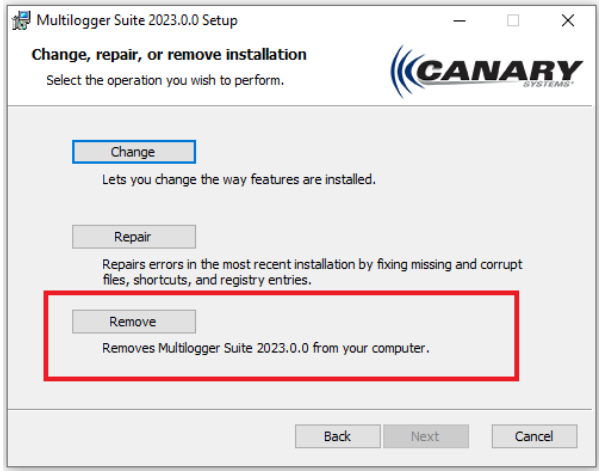
Note

While removal of MLSuite does not delete any user created or modified files, including Network Configuration, datalogger configuration files, collected data files or user customized programming options, it is recommended to backup files before attempting removal. Prior to backing up the CanarySys folder, both MultiLogger and Firebird services should be stopped.

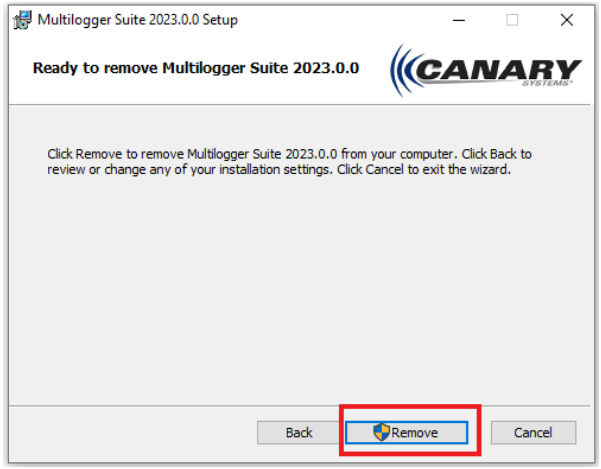
7.3.2 Uninstalling Using the MLSuite Installer

An alternative way to uninstall the software is to run the installation program again. This displays the **Program Maintenance** form, as shown in the figure below.

Select **Remove**, then **Next** to continue.



The next dialog box includes a message asking to confirm the removal of MLSuite. Click **Remove** to continue.



The dialog will update once MLSuite is successfully uninstalled. Click **Finish** to exit.

Appendix A – Server Configuration

A.1 Use of Multiple Drives

Using multiple drives with dedicated functions improves performance. It is recommended for all setups, be it two-tier, three-tier or even single server.

Recommended setup of the multiple drive system is three SSDs on the Back-End server:

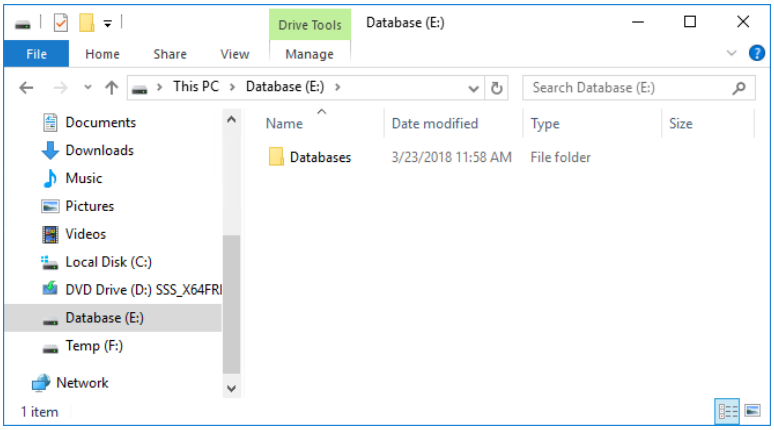
- **Local Drive** for Operating System.
- **Database Drive** for storing database files.
- **Temp Drive** for storing TEMP folders and Page files.

Database Drive Setup

The Database drive is where the database files are to be stored. Ensure that a folder exists, that is accessible from MLWeb where these files can be stored.

If the multiple drive method is being deployed on a machine with MLSuite already installed, **the Firebird and MLServer services must be stopped in Windows Services.** If they are not stopped, errors may occur, and the databases may be corrupted. **Firebird must be stopped first and then MLServer**.

After both services have been stopped, the Database folder can be moved. Ensure that the Database folder is copied from the previous drive onto the Database drive, and then removed entirely from the previous drive.



If MLSuite was already installed and databases were present, their location has now changed and MLAdmin must be updated to reflect this.

Updating Database Location in MLAdmin

Open MLAdmin and navigate to the Databases workspace. In the **Database Path** field, update the path to the new location and **Save**.

After the databases have been moved and the file paths have been updated to reflect the move, restart the Firebird service first, then restart the MLServer service.

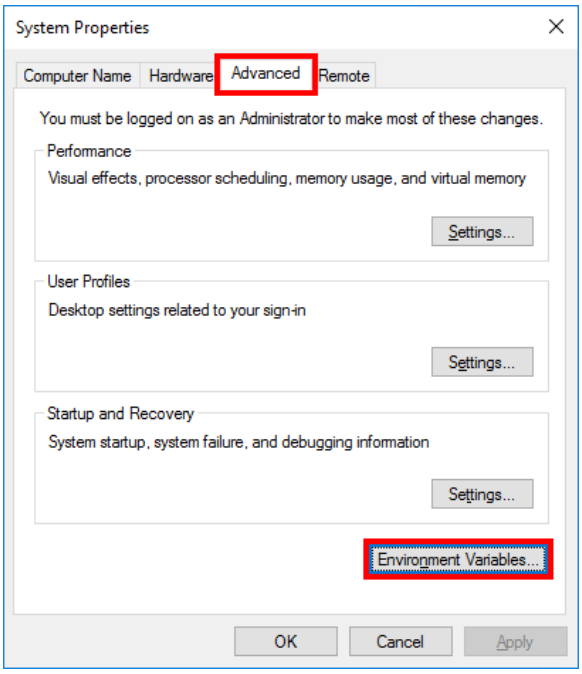
Temp Drive Setup

The Temp drive must be set to the location where TEMP folders and Page files are generated.

Open the **System Environment Variables** feature by clicking the start menu and type "System Environment Variables."

Moving TEMP Folders to the Temp Drive

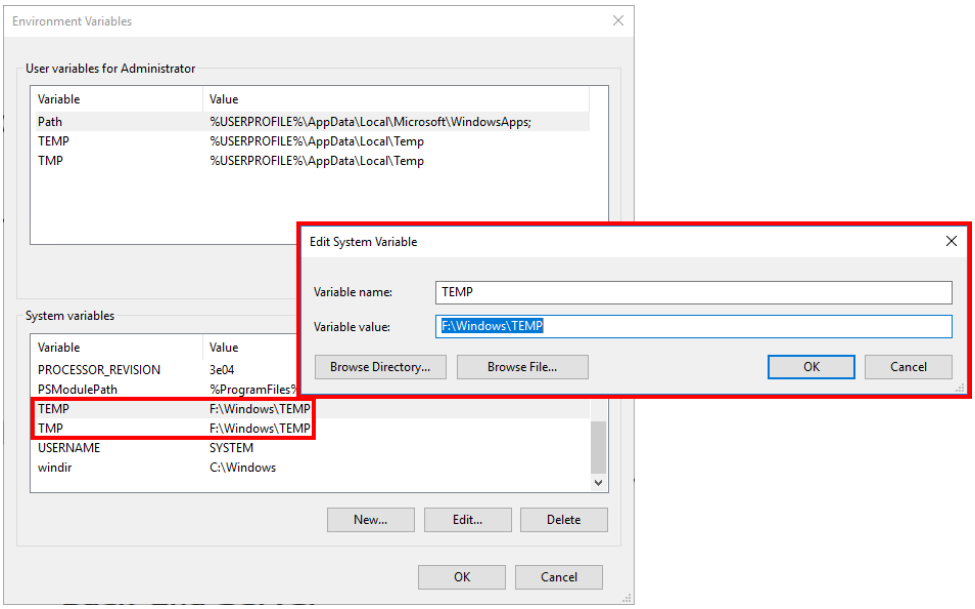
From the **Advanced** tab of the System Environment Variables form, click the **Environment Variables** button.



In the **System Variables** list, scroll to **TEMP** and select it.

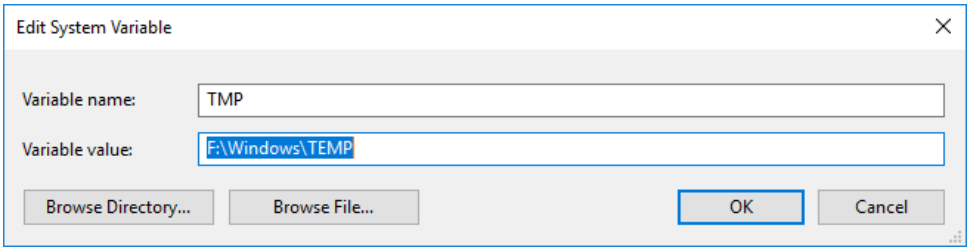
From the **Edit System Variables** dialog, change the **Variable Value** to the path of the Temp drive followed by **\Windows\Temp**.

For example, if the Temp drive is **F:**, enter **F:\Windows\TEMP**.



Repeat this step for **TMP**.

Using the previous example, **F:\Windows\TEMP** would be entered in the **Variable Value** field for TMP.



After these steps are complete, the server should be rebooted.

Confirm that the changes were made correctly by checking the TEMP folder in the Temp drive and seeing that TEMP files are being generated.

Login to MLWeb and confirm that it is running.

Update MLServer with Temp Location (Optional)

If using a custom temp folder:

Open the MLServer.ini file located within the CanarySys folder.

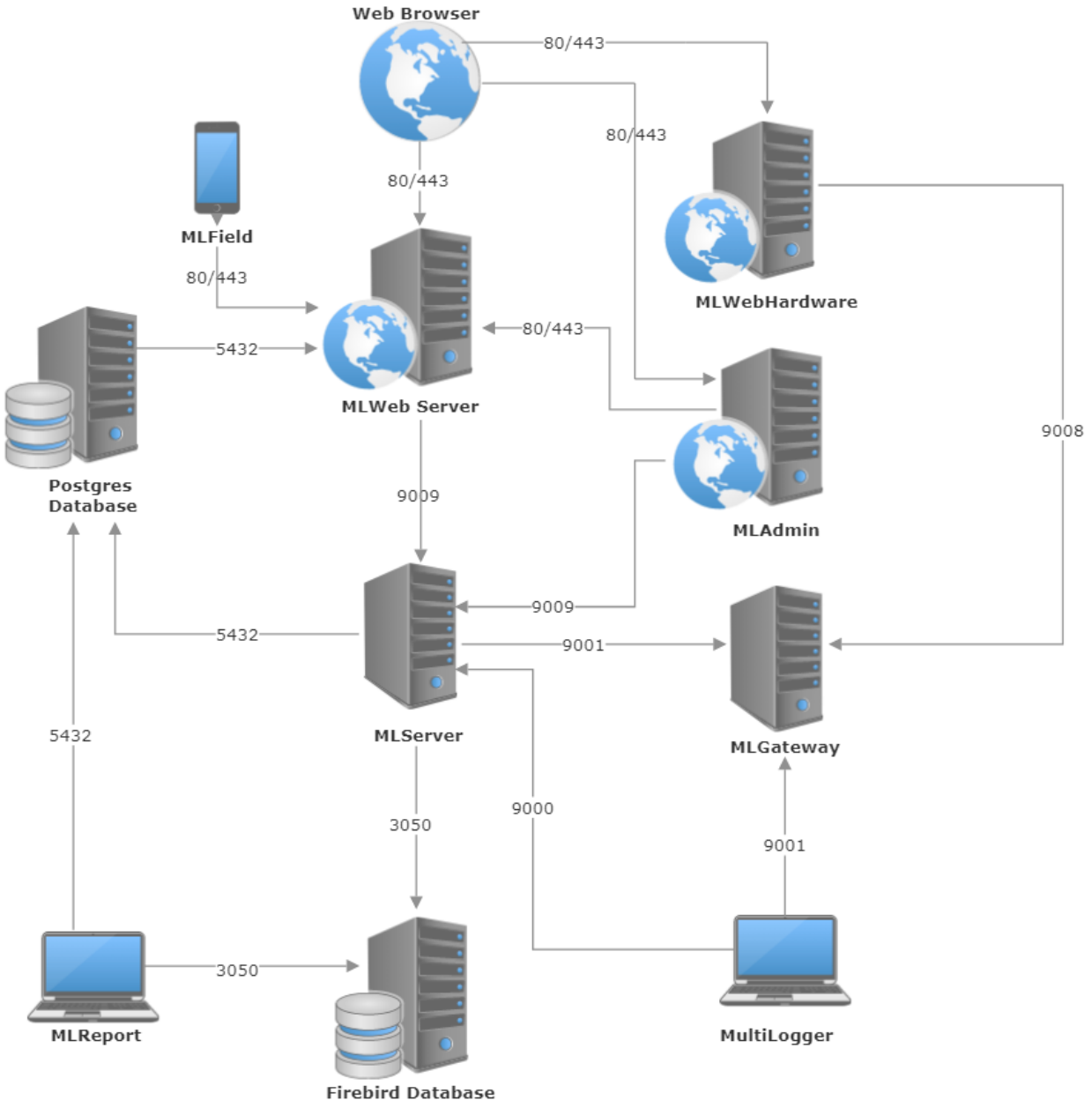
Open the file using Notepad++, under [General] there is a "TempFolder=" field. Add the path to the Temp drive.

```
[General]
Statistics=False
Statistics.TopEventTimeFilter=0
Statistics.ChildEventTimeFilter=0
PythonPath=
ReportJPEGHiRes=False
ReportJPEGQuality=95
CalcEngineThreadCount=0
DomainController=
ProcessAccount.UserName=canary_process
TempFolder=F:\Windows\TEMP
```

Note

By default, MLServer uses the Windows temp folder (MLServer subfolder).

A.2 MLSuite TCP Port Diagram

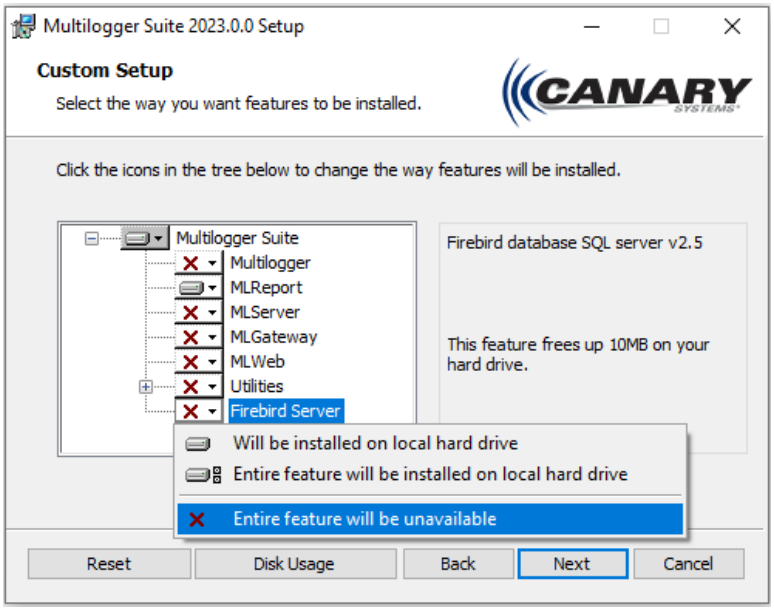


Appendix B – Installing MLSuite Components Locally

Installing MLReport Locally

Canary Systems recommends installing MLReport on your local machine and **not** on the same server that hosts MLWeb. To do this, follow the steps outlined below:

1. Download the installer from the Canary Systems website and begin the installation as outlined in section 2.5 Installing MLSuite.
2. MLReport is the **ONLY** feature that should be installed. From the **Custom Setup** wizard, disable all features other than MLReport. This is done by selecting **Entire feature will be unavailable** as shown in the figure below.



See section *1.2 Configure Database Connection* of the ***MLReport User's Guide*** (canarysystems.com/support/users-guides) for details on connection configuration.

Hint

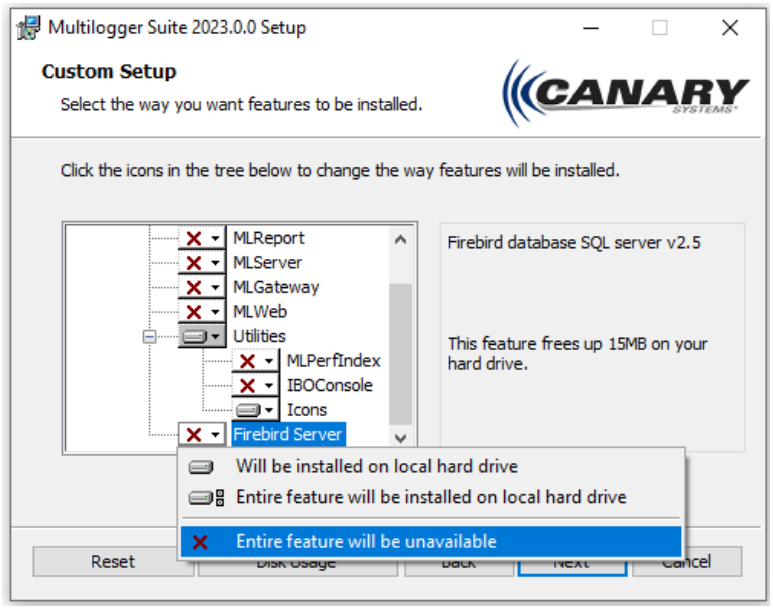
If connecting to MLServer on a remote server, firewall configuration may be required. For more information on configuring firewall rulesm see section *2.2 Firewall Configuration.*

B.2 Installing the MLWeb Icon Library Locally

The MLSuite installer includes a utility option for installing a library of standard MLWeb Icons.

If accessing MLWeb from the server where MLSuite is installed these can be found in a folder called **Icons** under the **CanarySys** folder.

If accessing MLWeb from a machine other than the server MLSuite is installed on, the Icon Utility option can be installed on a local machine. This is done by running the installer and setting all features to “Entire feature will be unavailable” except for the Icon utility.



This creates the CanarySys folder along with the Icon subfolder where the icon library can be accessed through MLWeb from the local machine.

Appendix C – Migration from MLWeb 2021 to MLWeb 2023

MLWeb 2023 has been enhanced in numerous critical ways which make it incompatible with previous Canary Systems browser-based applications and associated components. Follow the example migration in this appendix to ensure the most complete transfer of databases from the previous generation MLWeb to MLWeb 2023.

Note

Depending on the complexity of the database, Canary Systems recommends using a dedicated migration server for MLWeb 2021 to MLWeb 2023. upgrades. Contact Canary Systems Support for questions regarding specific setups.

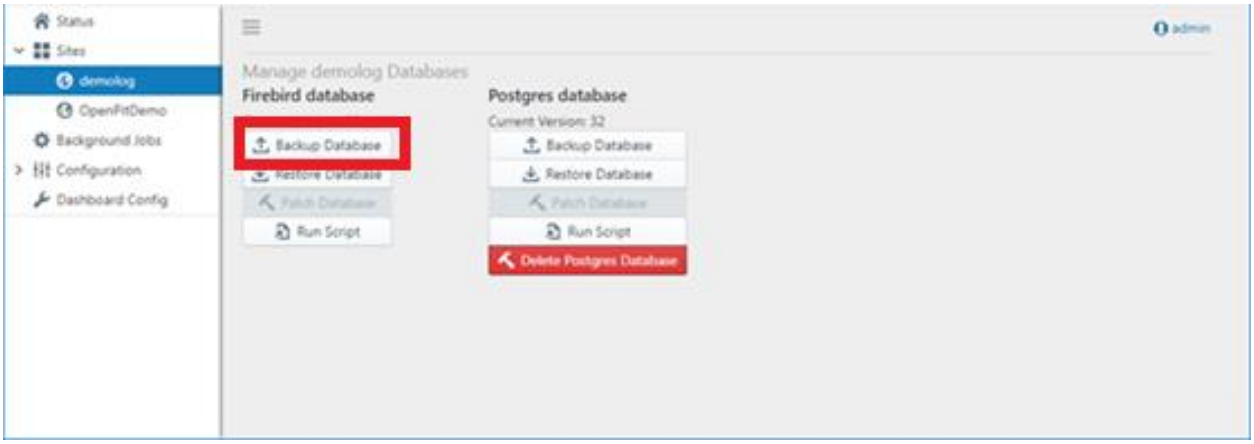
C.1 Backing Up Existing Database Files

Databases can be manually backed up using the Backup Database operation in MLAdmin. For migration from MLWeb 2021 to MLWeb 2023, only the Firebird database needs to be backed up.

The operation creates a **Database Backup** file, or ***.gbk*** file, stored in a zipped folder (*.zip*). The *.gbk* file's name format is the database name. The zipped folder's name format is the database name followed by the date and time of the backup: *Database_yyyymmddhhnn.zip*.

To manually backup a database:

1. Navigate to MLAdmin.
2. Expand the **Sites** tab and then select the site to be backed up. Doing this open's the site's Manage Databases workspace. Alternately, clicking Manage within the general Sites workspace opens the Manage Databases workspace.
3. In the Manage Databases workspace, under the Firebird database heading, click Backup Database. A zip file (*.zip*) containing the backup file (*.gbk*) is created and downloaded.



This .gbk file can be used to restore the database in MLWeb 2023 once the new version of the software has been installed.

For more information on MLAdmin, see *Chapter 21 – MLAdmin* of the ***MLWeb User's Guide 2019.***

> **Note**
>
> If **not** using a migration server, 3D objects, as well as inclinometer, TDR, and SAA chart data must be noted and/or saved separately from the database backup. **IMPORTANT: these database items will not transfer over to MLWeb 2023 and must be manually added after databases are restored in the upgraded software.** For more information on this process continue to section *C.3 Migrating Reports, Charts, Objects, and Scenes.*

C.2 Uninstall MLSuite and Components

> **Note**
>
> The steps here are specific to the process for uninstalling previous editions of MLSuite in preparation for an upgrade to new software and differ from the process as laid out in section 7 of this guide.

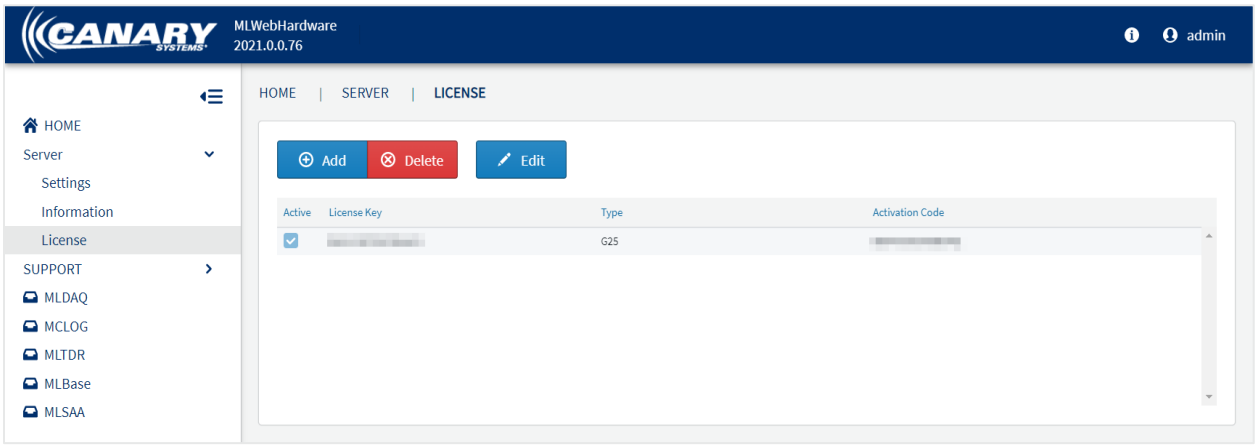
C.2.1 MLGateway License Deactivation

Before removing MLSuite, the Gateway License for MLGateway must be deactivated on the current machine to allow it to be reactivated on another. This can be done through MLWebHardware or MLGateway.

Once the license for MLGateway is deactivated, the licensed databases in MLAdmin must also be deactivated. This is described in section *C.2.2 Database Licenses Deactivation*.

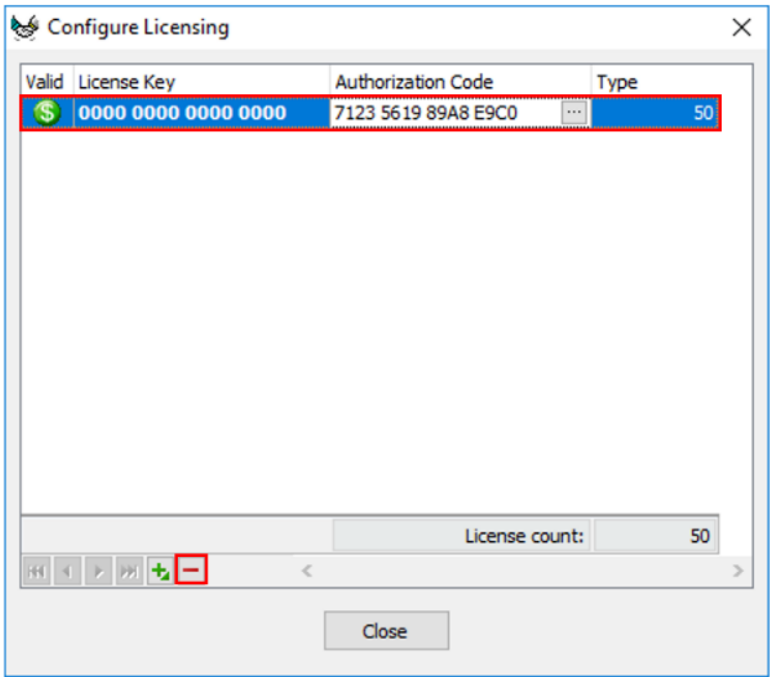
License Deactivation through MLWebHardware

Open **MLGateway** and click the **License** tab. This brings up a list of configured licenses, as shown in the figure below. Select the license you wish to delete and click the **Delete** button to deactivate the license.



License Deactivation through MLGateway

Open **MLGateway** and click the **License** button. This will bring up the **Configure Licensing form**, as shown in the figure below. Highlight the license and click the Delete Record button to deactivate the license.



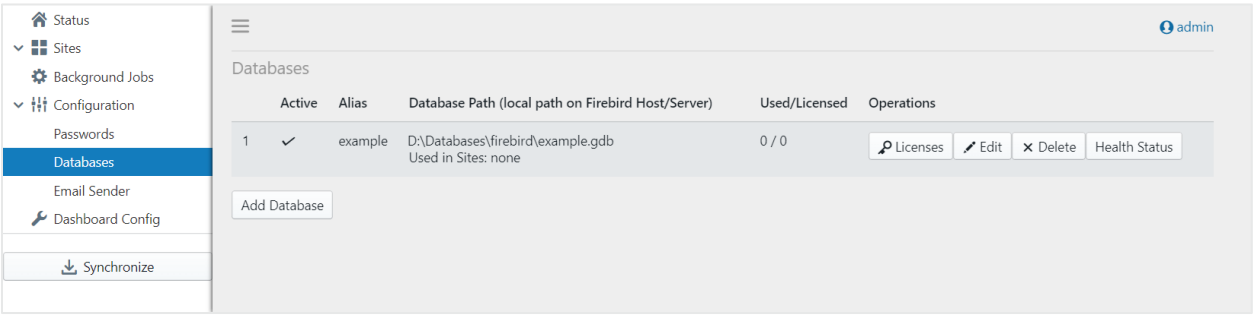
C.2.2 Database License Deactivation

In addition to deactivating the Gateway License, all database license keys should also be removed. This will allow them to be preserved and reused later.

Open **MLAdmin** and click the **Database** tab. From the Databases workspace click the Edit button to make note of the License Key for future use.

Close the License form and click **Delete** to remove the license from the database.

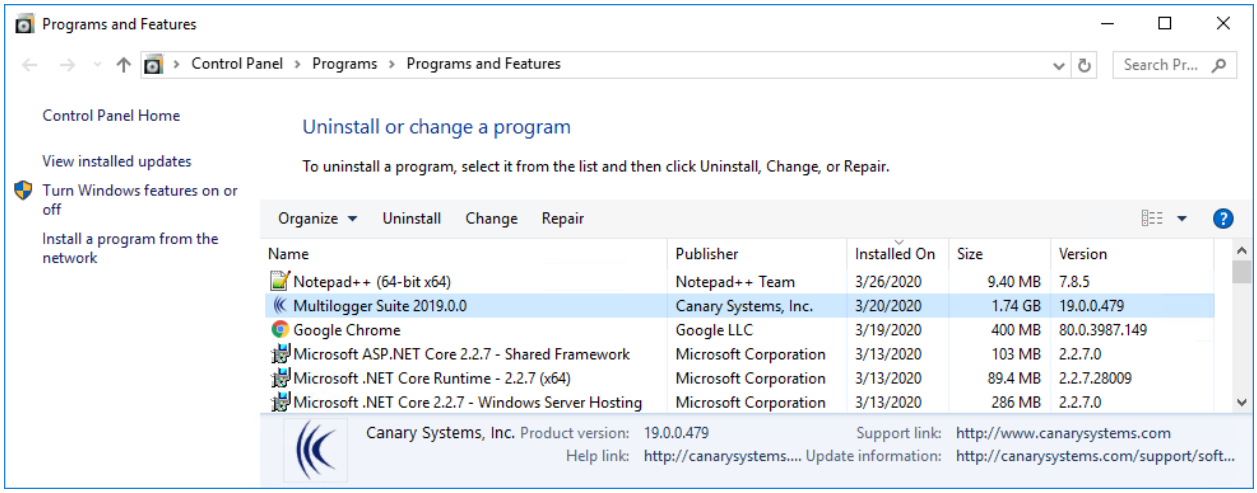
Repeat for all database license keys.



C.2.3 MLSuite Removal

Uninstalling Using the Control Panel

To remove MLSuite, open the **Programs and Features** menu of the Control Panel. Locate **MLSuite** in the list of programs and click **Uninstall** to begin the removal process.

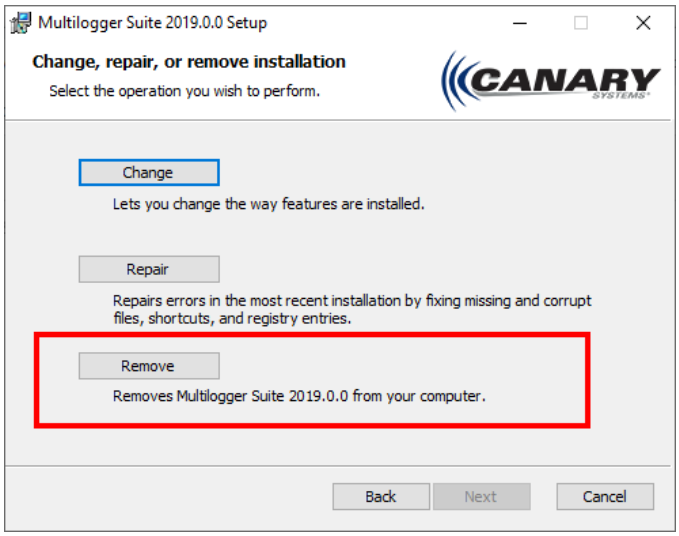


A message box is displayed, asking to confirm removal of MLSuite. Click **Yes** to continue to uninstall.

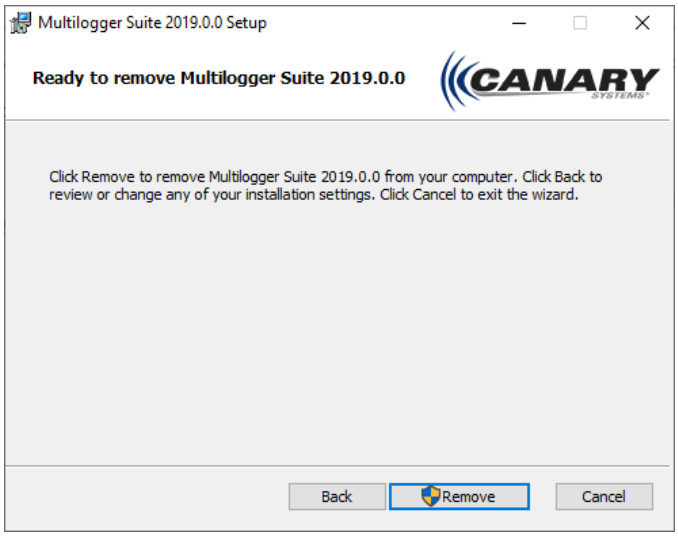
Uninstalling Using the MLSuite Installer

An alternative way to uninstall the software is to run the installation program again. This displays the **Program Maintenance** form, as shown in the figure below.

Select **Remove**, then **Next** to continue.



The next dialog box includes a message asking to confirm the removal of MLSuite. Click **Remove** to continue.

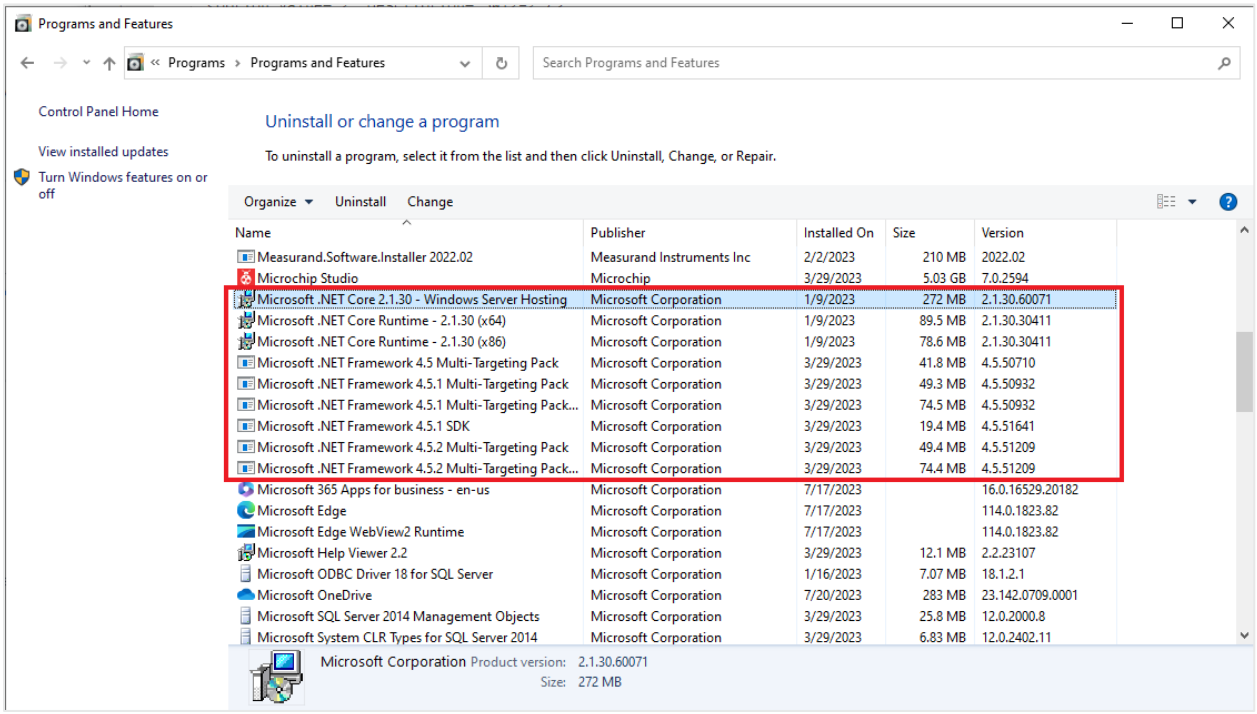


The dialog will update once MLSuite is successfully uninstalled. Click **Finish** to exit.

C.2.4 Uninstall Components

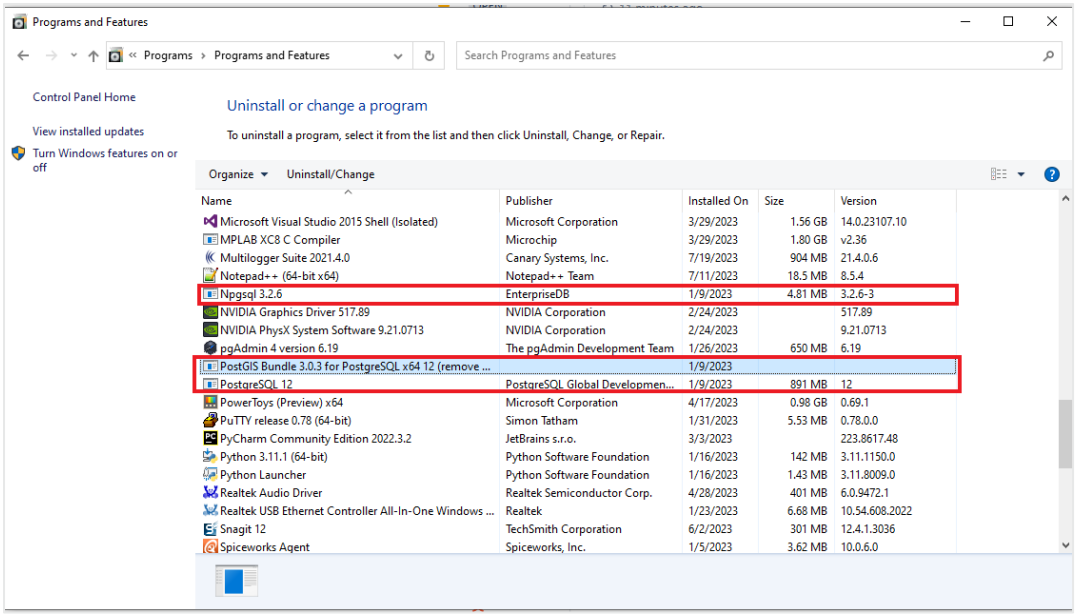
Return to the **Programs and Features** menu of the Control Panel. Unistall the associated components of MLSuite in the reverse order to which they were originally installed.

1. From Programs and Features, uninstall existing version of .NET Core Runtime and all associated components (in any order).



2. From Program & Features, uninstall existing PostGIS bundle as well as Npgsql (any order).

3. Finally, uninstall existing version of PostgreSQL.



C.3 Migrating Reports, Charts, Objects, and Scenes

Because of enhancements to MLWeb 2023, such as new charting and GIS engines, special attention is called for when migrating certain database items from legacy software. Carefully review the following guidelines to ensure the most complete and accurate migration of databases from MLWeb 2021 to MLWeb 2023.

Note

Canary Systems has implemented an entirely new mapping/GIS engine in MLWeb 2023 featuring numerous enhancements including dual 2D/3D scenes. **Image Windows as found in legacy versions of the software are no longer supported.** For information on machine settings for best reaults when viewing 3D data, reference the **MLWeb User's Guide 2023.**

C.3.1 Custom Reports

Certain table and column names have changed in MLWeb 2023 and must be updated in any SQL code via MLReport in order for Custom Reports to display correctly in the software. For more information on scripting in MLReport, refer to the ***MLReport User's Guide (***canarysystems.com/support/users-guides***)***.

For changes to the table and column names, see *Appendix F – Table and Column Name Updates in MLWeb 2023.*

> **Note**
>
> The referenced appendix does not contain an exhaustive list of changes. Canary Systems recommends contacting their Support team for assistance with SQL scripting.

C.3.2 Charts

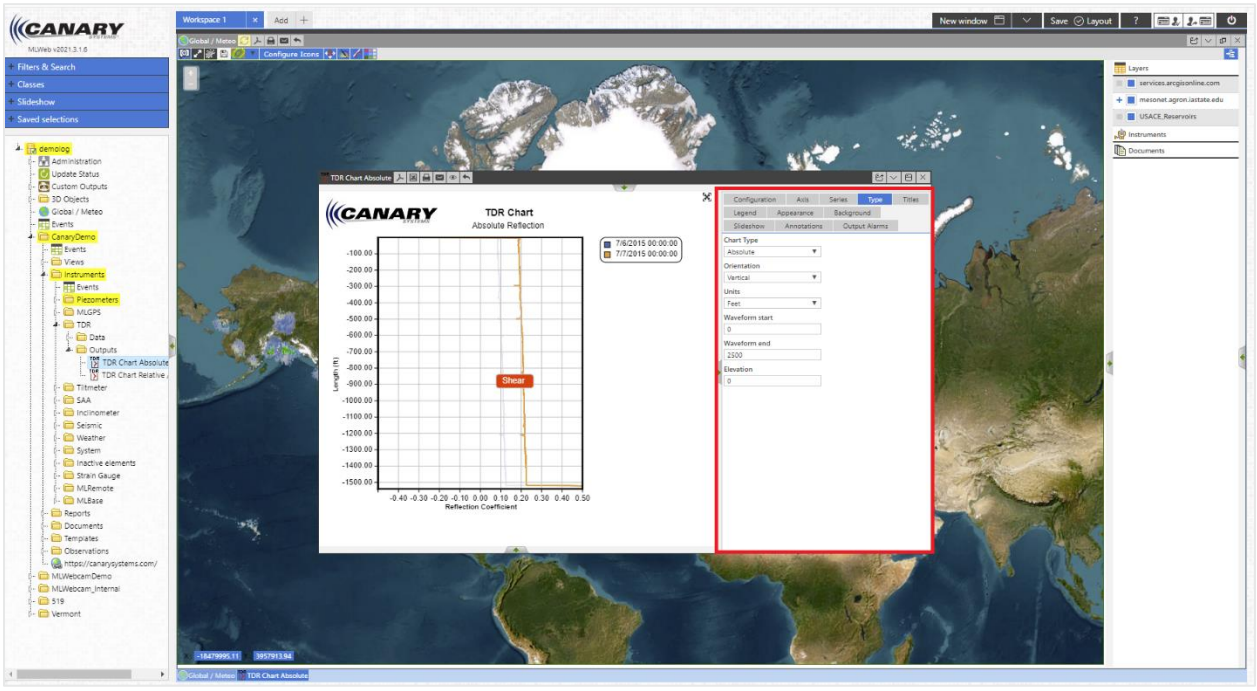
Select fields from the Outputs Configuration Side Panel in MLWeb 2021 are now instrument properties in MLWeb 2023. These configurations do not migrate to MLWeb 2023 automatically and must be entered manually. **TDR, Inclinometer, and SAA charts will not display accurately until these properties have been updated.**

> **Note**
>
> For assistance with the migration of TDR, Inclinometer, and SAA charts, contact Canary Systems Support.

Use the below sections to determine how fields from MLWeb 2021 Output Configuration Side Panel fields correlate to MLWeb 2023 instrument properties.

TDR Charts

The fields that must be entered as properties in MLWeb 2023 are found by opening the Outputs Configuration Side Panel for the chart being worked on in MLWeb 2021 and then navigating to the **Type** tab.



In MLWeb 2023, navigate to the corresponding Property within the Instrument and then enter the data from each MLWeb 2021 field using the **Add new** button. A list of MLWeb 2021 Outputs Configuration Side Panel fields and the properties that each correspond to in MLWeb 2023 is found below.

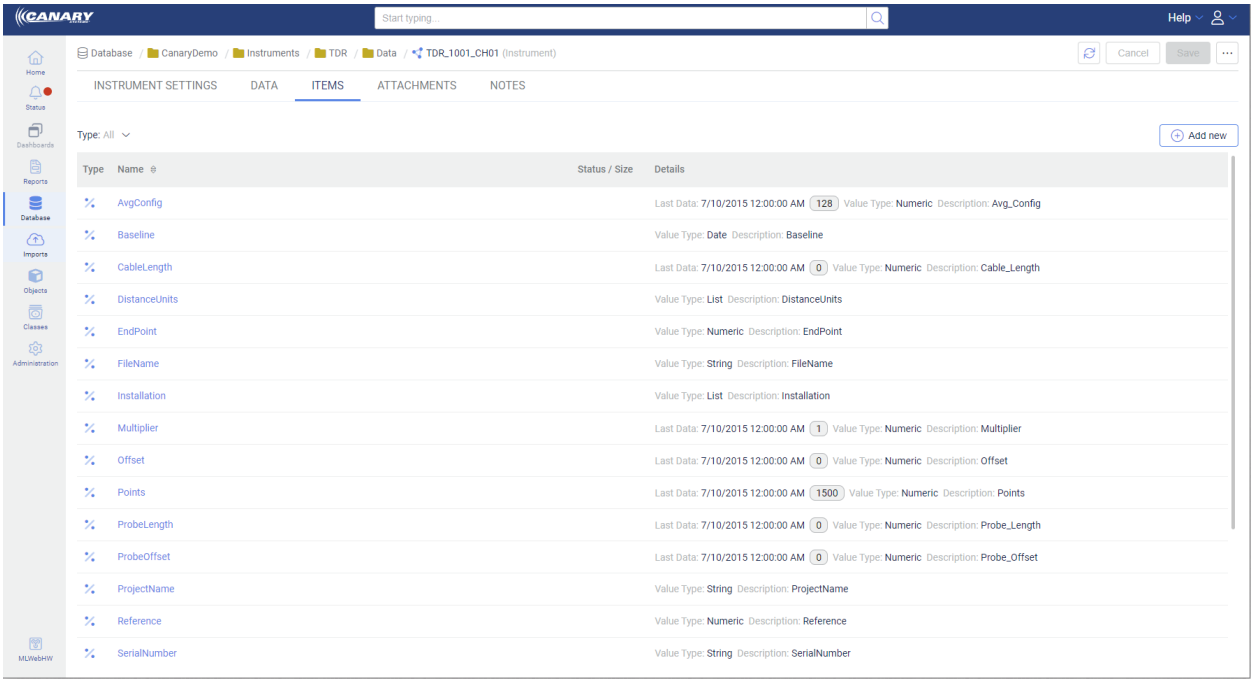


Chart Type – This field migrates automatically; no action is necessary.

Orientation – Corresponds to the Installation Property in MLWeb 2023.

Units – Corresponds to the DistanceUnits Property in MLWeb 2023.

Waveform start – Corresponds to the Start point Property in MLWeb 2023.

Waveform end – Corresponds to the End point Property in MLWeb 2023.

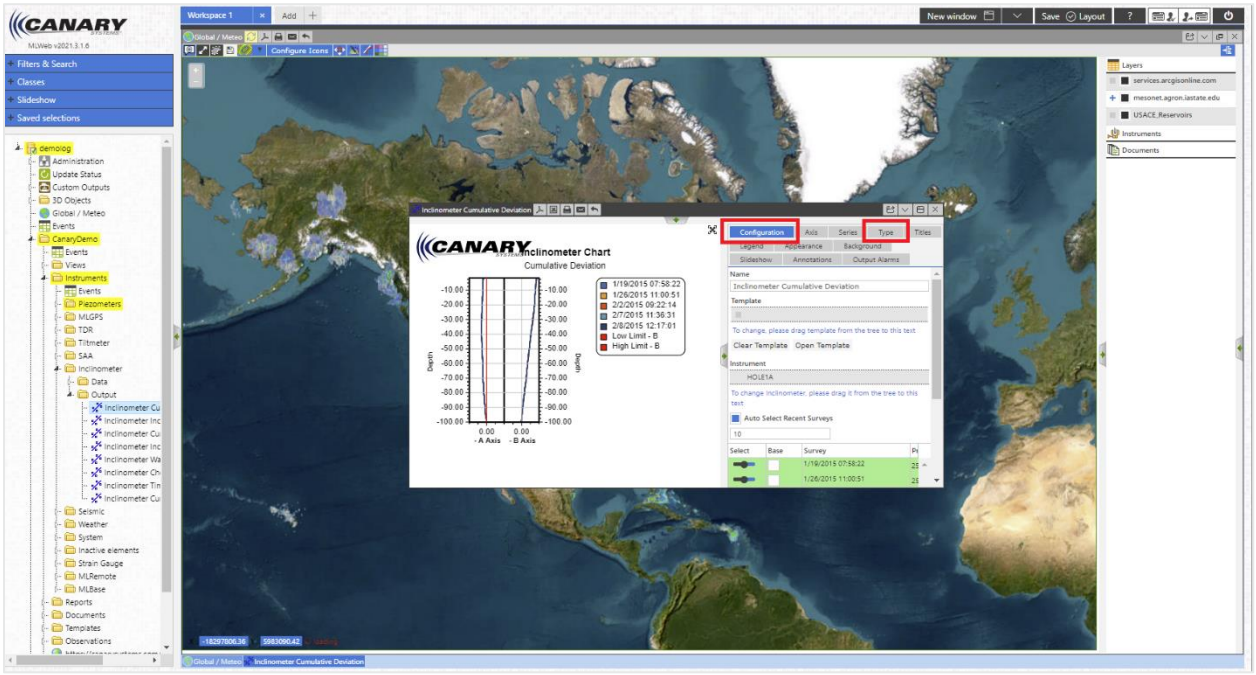
Elevation – Corresponds to the Reference Property in MLWeb 2023.

Note

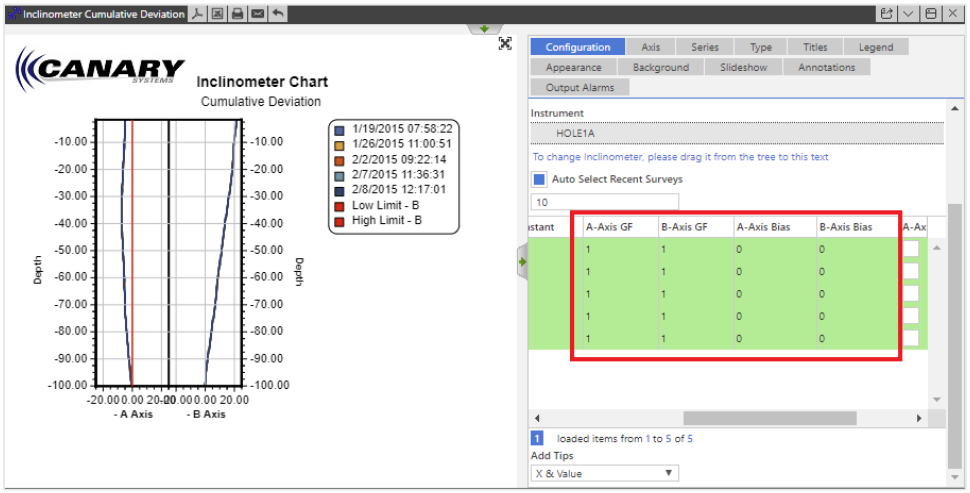
When entering this data in MLWeb 2203, it is important to remember to always set the Recorded field of the **New Data** side panel to a date and time prior to the first data included in the chart being migrated.

Inclinometer/SAA Charts

For Inclinometer/SAA charts in MLWeb 2021, there are two tabs in the Outputs Configuration Side Panel that contain information which must be manually added to the migrated charts in MLWeb 2023: the Configuration tab and the **Type** tab.



A list of applicable Configuration tab chart columns and the properties that each correspond to in MLWeb 2023 is found below.



A-Axis GF – Corresponds to the GageFactorA Property in MLWeb 2023.

B-Axis GF – Corresponds to the GageFactorB Property in MLWeb 2023.

A-Axis Bias – Corresponds to the BiasA Property in MLWeb 2023.

B-Axis Bias – Corresponds to the BiasB Property in MLWeb 2023.

A list of applicable Type tab fields and the properties that each correspond to in MLWeb 2023 is found below.

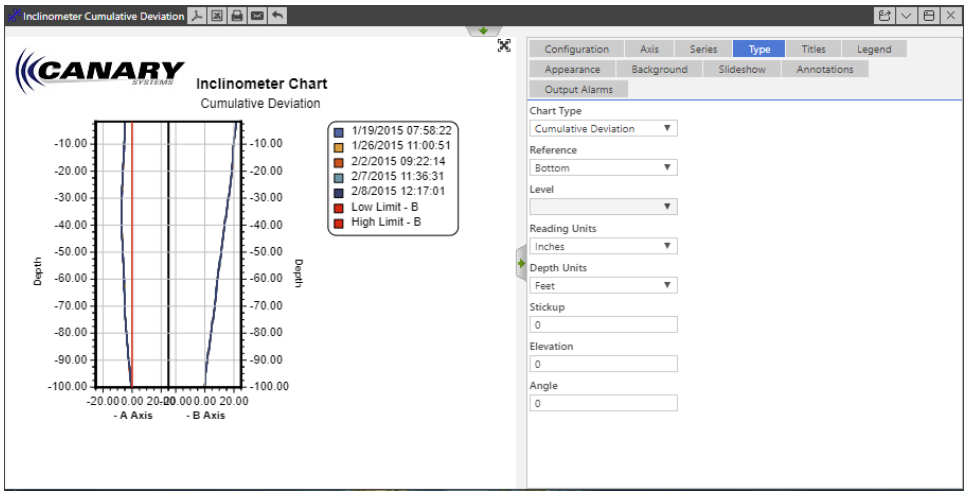


Chart Type – This field migrates automatically; no action is necessary.

Reference – Corresponds to the Direction Property in MLWeb 2023. After selecting the **Add new** button to bring up the New Data slider, the NearToFar selection corresponds to the Bottom selection in MLWeb 2021; the Far ToNear selection corresponds to the Top selection in MLWeb 2021.

Reading Units – Corresponds to the ReadingUnits Property in MLWeb 2023.

Depth Units – Corresponds to the DistanceUnits Property in MLWeb 2023.

Stickup – Corresponds to the Stickup Property in MLWeb 2023.

Elevation – Corresponds to the Reference Property in MLWeb 2023.

Angle – Corresponds to the Azimuth Property in MLWeb 2023.

Note

When entering this data in MLWeb 2023, it is important to remember to always set the Recorded field of the **New Data** side panel to a date and time prior to the first data included in the chart being migrated.

C.3.3 Objects and Scenes

For full functionality given the enhanced GIS engine, scenes from legacy edition software must be rebuilt for MLWeb 2023. For information on scenes, refer to the ***MLWeb User's Guide 2023*** (canarysystems.com/support/users-guides).

Appendix D – Creating Canary Process Account

The MLAdmin site includes an option for a Windows process username and password. This is referred to as the Process Account or Canary Process Account and is required for various processes to run properly without requiring elevated privileges.

D.1 Windows Process User Requirements/Rights

The Windows Process user can be a local account on the server or a Windows Domain account.

The account should be setup so that the password meets the complexity requirements and configured so that the password never expires.

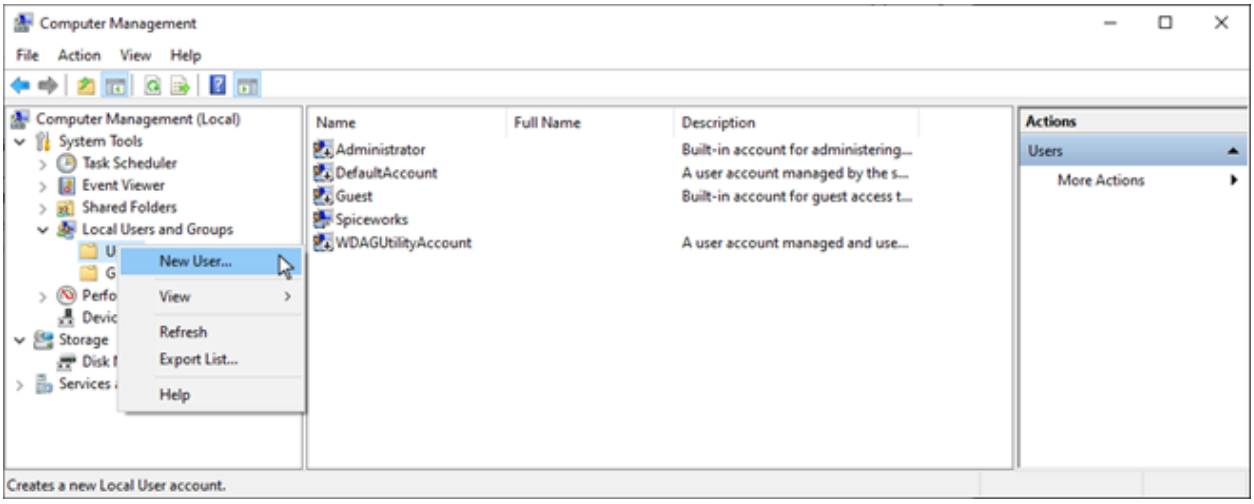
When entering the username and password in MLAdmin, if the account is a local account (created on the server), enter the username of the account with the password you have configured (refer to section *4.1 Installation Setup*). If the account is a Windows Domain account, the username must be entered as username@domain (i.e. Canary_Process@canary.local).

D.2 Standardize name as Canary_Process

To standardize deployments, use **Canary_Process** for the username for either a local account or a Windows Domain account.

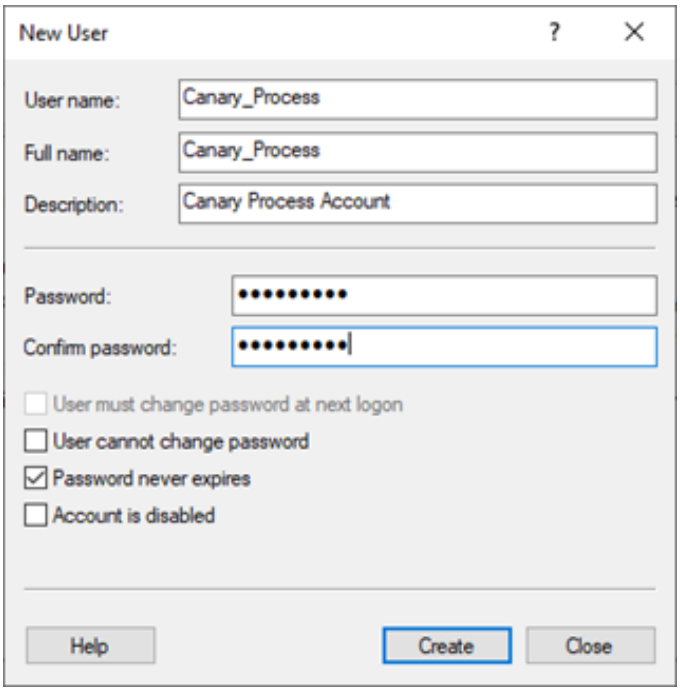
Creating a Local User Account

1. From the Start menu, navigate to **Computer Management**.
2. In the lefthand panel, expand **Local Users and Groups** and then right click on **Users**.
3. Choose **New User** from the drop-down menu. A new window appears.



4. Enter the name “Canary_Process” and add a description if desired.

5. Enter a password that meets the local system's password policy, confirm the password, uncheck the box. **User must change password and next logon**, and check the box **Password never expires**.



6. Click the **Create** button to create the Process Account.

D.3 Permissions needed for Canary_Process Account

After a Windows Process user has been created and set up in MLAdmin, permissions to specific folders on the server running MLServer need to be granted for this account. The account needs "modify" access to:

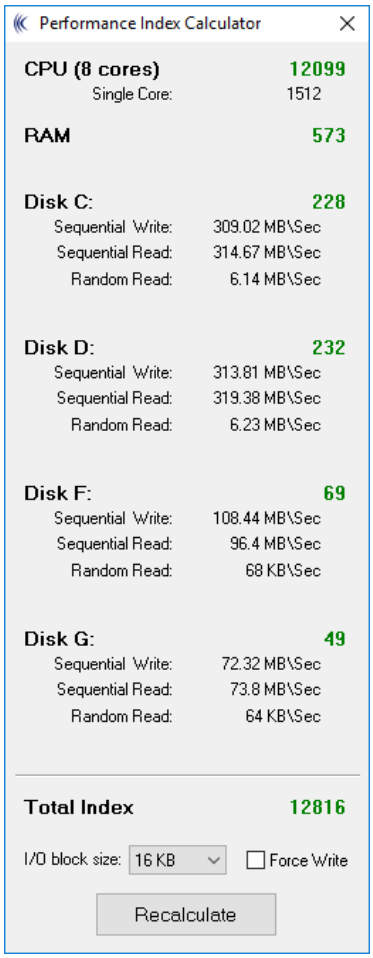
- The **CanarySys** folder (and all subfolders) wherever that is located. By default, this is under C:\Users\Public\Documents\CanarySys, but is often moved to the root of one of the data drives on the server.
- The **Windows\Temp** folder (and all subfolders) wherever that is located. By default, this is located in C:\Windows\Temp, but that can be moved by changing the system environment variables.
- In addition, if Python scripts are executed through MLServer, the Canary_Process account requires modify access to the location where the **Python executables** are installed.

Appendix E – Performance Index Tool

MLSuite includes a tool which can aid in optimizing server performance: **MLPerfIndex**.

After launching MLPerfIndex, simply click on the **Recalculate** button to launch the indexing process.

Be advised that this process makes heavy use of all system resources while it's running. It is advisable to run indexing only during times of low or preferably no other traffic in order to avoid disruptions to other users and processes, as well as to obtain as good a baseline as possible without external interference.



Two options are available, and should be left at their default values:

- **I/O Block Size:** 16 KB
- **Force Write:** (unchecked)

Acceptable index values are as follows:

- **Single logical CPU:** 1100 - 1500
- **Total CPU (8 cores):** 9000 - 12000
- **RAM:** 500-600
- **Disk:** 200 or higher (values lower than 200 indicate non-SSD hard drives with higher read/write latency)

This tool simulates the day-to-day workload of navigating and reading/writing the Firebird database at the core of MLSuite. Due to the specific environment simulated with MLPerfIndex, the results are likely to differ from third-party tools such as *CrystalDiskMark* for drive benchmarking or *Cinebench* for CPU benchmarking.

The *MLPerfIndex* score is split into three broad categories: CPU, Memory (RAM) and Storage. These scores are then combined into a single, final score.

CPU

The CPU test measures the processor's floating-point performance by leveraging all available cores at once. The threads run in real-time (highest) priority to avoid the influence of other processes, but it is recommended to close other applications anyway. During the test, the mouse pointer might appear frozen or lagging; this is normal.

The more cores the test machine has, the higher the performance index score. Also shown in the CPU index is the single-core performance index where only a single core is tested to represent a lighter workload.

Memory

Memory performance is measured in a single thread (one CPU core) by allocating and releasing 1MB blocks of memory in a loop. The score on this section of the test scales with the number of cycles per second (MHz) the RAM is capable of, so the faster the RAM speed, the higher the performance index score. More RAM, such as having 64GB instead of 16GB, does not necessarily result in a higher performance index score. The DDR revision of the memory tested also impacts scores, mainly because newer revisions have much higher base speed when compared to older revisions. Basic DDR5 memory (4800MHz), such as that found in newer workstations, can score around 1000 points. Basic DDR4 memory (2133MHz) scores closer to 300-500.

File System

The file system performance is measured in a single thread (one CPU core) by creating, writing, reading, and then deleting a 16K block in a loop. Higher scores indicate faster sequential read and write speeds, along with faster random read speeds. Much of the score is dependent on the IOPS (Input/Output Operations per Second) of the drive and the access time, where higher IOPS and lower access times will lead to higher scores. A basic spinning disk hard drive (HDD) with a mechanical, moving drive head will score low due to long (up to 5ms) access time and low IOPS (max 400 for high-RPM drives). A solid-state drive (SSD) with no mechanical parts is no longer limited by the access time of a moving drive head, leading to lower access time and higher IOPS (75000+). An NVMe SSD is even faster, with IOPS over 1,000,000 in certain cases.

Canary Systems recommends *at least* a solid-state drive (SSD) for Firebird database storage and operation.

Note

An additional consideration regarding drive speed is physical vs. virtual storage. In the case of a virtual machine (VM) running MLSuite, in which case storage is often managed via a SAN or NAS drive, speed losses of between 20% and 80% are expected when under load. SAN/NAS are effective for transferring hundreds of MBs from one location to another but fail to keep up with hundreds of thousands of tiny blocks of data commonly seen with Firebird databases. A physical drive dedicated to just Firebird is recommended for maximum performance.

Appendix F – Firebird Table and Column Name Updates in MLWeb 2023

Old Table Name	New Table Name	Old Column Name	New Column Name
Dataelement		data_element	ID
		Entity_class_nbr	Entity_class_id
Data	**Dataelement_data**	data	num_value
		data_element	Dataelement_id
Entity_Class		Entity_class_nbr	ID
Users	**Security_user**		

Appendix G – Advanced MLServer .ini Settings

The MLServer *.ini* settings may need to be modified for various reasons. The .ini file is located in the **CanarySys\MLServer** folder. To edit the file, open it using Notepad or a similar tool, navigate to the section where changes are needed. Once the changes are made **Save** and **Close** the file.

Note

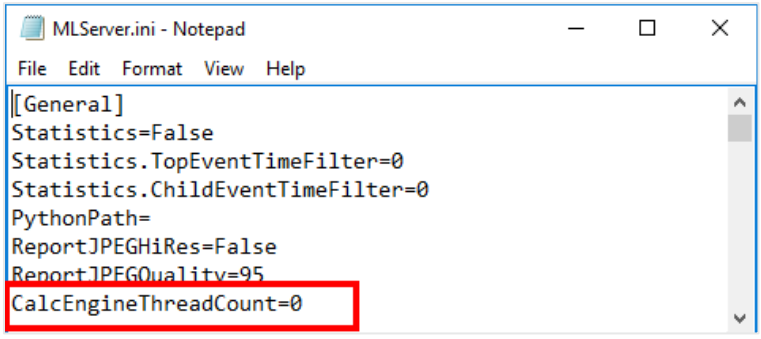
After the settings are changed and the MLServer .ini file is saved, the **MultiLogger: MLServer** service must be restarted from the **Windows Services** application.

G.1 CalcEngineThreadCount

The default value is "CPU Virtual Cores" multiplied by 4. Default value is used when the override is missing in **MLServer.ini** or it is equal to 0.

```
[General]

CalcEngineThreadCount=8
```

In some cases, to avoid server overload, default should be changed to a lower value.



CalcEngineTreadCount highlighted in the .ini file.

G.2 Processing

In a database with data going back for an extended period of time restricting how far back to look before setting off alarms is possible by adding a setting to the .ini file.

```
[Processing]

DontProcessOlderThan=45
```

Any data older than the number of days specified as the DontProcessOlderThan value will not be processed for alarms, notifications and calculations. If the DontProcessOlderThan value is not set in **MLServer.ini,** the database looks back 90 days by default.

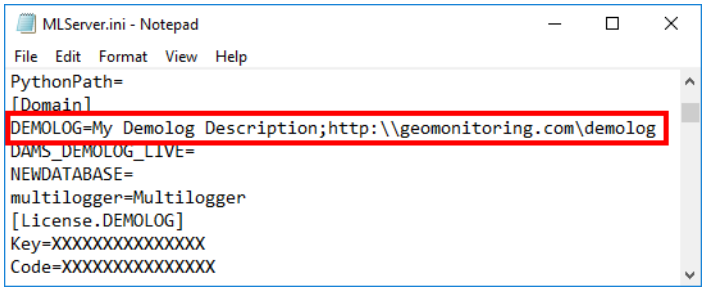
Note

This setting is not in the .ini file by default. If specifying the number of days to for processing it needs to be added manually.

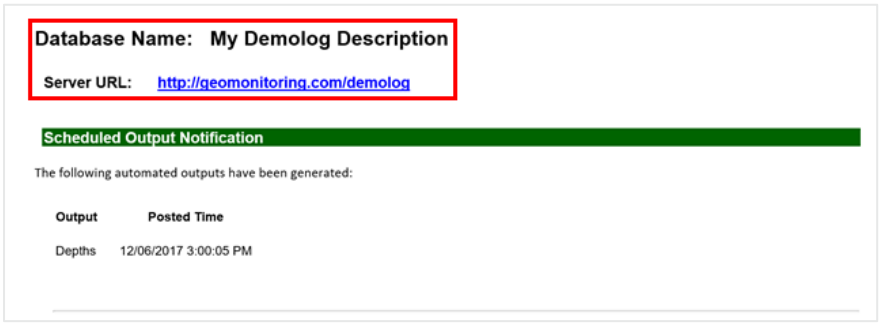
G.3 Notification Email Fields

The Database Name and Server URL fields of Notification Emails can be configured with the **MLServer.ini** file. This file is found in the CanarySys folder at the following path: **C:\Users\Public\Documents\CanarySys\MLServer**.

Under the **[Domain]** section of MLServer.ini, enter the desired database alias or server URL to be used for the database. If both are to be used, separate them with a semicolon.



The Database Name and Server URL fields configured in the MLServer.ini file.

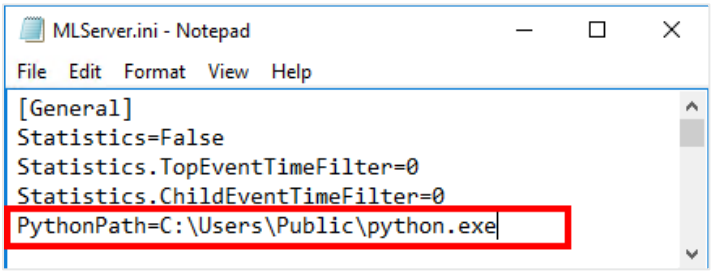


The Database Name and Server URL fields as they appear in a Notification Email.

G.4 Configuring MLServer for Python Script Imports

Before MLServer can run Python scripts, the path to the location of **python.exe** must be specified in the **MLServer.ini** file. This file is found in the CanarySys folder at the following path: **C:\Users\Public\Documents\CanarySys\MLServer**.

Under the **[General]** section of MLServer.ini is the **PythonPath=** parameter. The path to where – on the machine running MLServer – python.exe is entered in this parameter.



The PythonPath= parameter of the MLServer.ini file.

After the path is entered and the MLServer.ini file is saved, the **MultiLogger: MLServer** service must be restarted from the Windows **Services application**. MLServer is then able to run Python Scripts.

Note

If Python was installed to the system path then this is unnecessary.