

IBM Watson Explorer  
Version 11.0.2

*Installation and Integration Guide*





IBM Watson Explorer  
Version 11.0.2

*Installation and Integration Guide*



**Note**

Before using this information and the product that it supports, read the information in “Notices and trademarks” on page 45.

---

# Contents

## Installing and Integrating IBM Watson Explorer Foundational Components. . . 1

Overview of Watson Explorer Foundational Components . . . . .	1
Requirements . . . . .	2
Installing Watson Explorer Product Modules. . . . .	2
Pre-Install Configuration and Data Collection . . . . .	2
Selecting and Preparing Installation Media . . . . .	7
Installing Watson Explorer Version 11.0.2 . . . . .	8
Configuring Watson Explorer Modules . . . . .	10
Starting Watson Explorer Services . . . . .	34
Accessing Watson Explorer Modules . . . . .	35
Uninstalling Watson Explorer . . . . .	37
Watson Explorer Upgrade Options. . . . .	38
Enabling Javascript in Internet Explorer 7 and Later . . . . .	40
Modifying the Location of the Watson Explorer Knowledge Center URL . . . . .	40

Troubleshooting an Installation . . . . .	41
Getting Help . . . . .	42
Upgrading from Previous Releases . . . . .	42
Upgrading to Watson Explorer Fix Pack versions . . . . .	42
Upgrading from Previous Watson Explorer Versions . . . . .	43
Upgrading from Velocity to Watson Explorer Engine . . . . .	43
Upgrading from IO Pro to Watson Explorer Results Module . . . . .	43

## Notices and trademarks . . . . . 45

## Providing comments on the documentation . . . . . 49



---

# Installing and Integrating IBM Watson Explorer Foundational Components

This portion of the documentation identifies the requirements for successfully installing and running Watson™ Explorer foundational components, explains how to install one or more of the modules that make up Watson Explorer foundational components, and discusses how to integrate Watson Explorer foundational component modules into your system administration environment.

---

## Overview of Watson Explorer Foundational Components

IBM® Watson Explorer is available in two editions, an Enterprise Edition (EE) and an Advanced Edition (AE). The modules described in the following sections are delivered as part of all editions. The functionality of the modules does not differ across editions.

The Watson Explorer foundational components are comprised of the following installable modules:

### **Application Builder**

Simplifies developing 360-degree information applications that deliver targeted, relevant data to users based on their roles or interests within the enterprise or a specific organization.

### **BigIndex**

Provides a Java API for creating, configuring, and deploying information navigation applications in enterprise environments. The BigIndex API includes sophisticated functions that simplify or eliminate traditionally manual enterprise application administration tasks such as load balancing, mirroring, and replication. The BigIndex API Developers Guide PDF can be downloaded from [Using the BigIndex API for Watson Explorer](#).

### **Watson Explorer Engine**

Provides a web-based development and deployment environment for enterprise information navigation applications.

### **Annotation Administration Console**

Provides a web-based administration tool for configuring custom and domain specific text analytics that can enhance Watson Explorer 360-degree view and content mining applications.

### **IBM Knowledge Center**

Provides online help for IBM Watson Explorer Foundational Components and IBM Watson Explorer Analytical Components. By default, online help within these applications contacts the public IBM Knowledge Center server to deliver online help. Installing and starting a local installation of IBM Knowledge Center provides online help for Watson Explorer in "dark" environments where development systems may not have access to the Internet, and in environments where Internet access may be slow or inconsistent.

### **Results Module**

Enables local administrators of enterprise information navigation

applications to easily extend those applications with business-specific highlighting of selected query results and by incorporating domain-specific knowledge such as taxonomies.

### **ZooKeeper**

Provides a distributed coordination service and associated data repository for storing configuration information about distributed applications.

---

## **Requirements**

The following link contains information about the requirements for installing any Watson Explorer modules, provide recommendations for minimum hardware configurations for development and deployed systems, and discuss client requirements for accessing the administrative interfaces for any Watson Explorer modules and applications that have been created using those modules: [Watson Explorer Version 11.0.2 system requirements](#)

---

## **Installing Watson Explorer Product Modules**

The next few sections provide overview information about the ways in which those modules can be installed, and then provide a series of sections that walk you through the pre-installation, installation, and post-installation phases of exploring Watson Explorer. This section concludes with troubleshooting information and information about how to obtain assistance from IBM product support if you cannot resolve any installation, configuration, or integration problems that you are experiencing.

### **Pre-Install Configuration and Data Collection**

Installing one or more Watson Explorer modules on different platforms requires that you collect some information about those systems and configure any associated system servers appropriately. The next sections discuss how to ensure that a system provides the physical resources and performance that are required for Watson Explorer development and deployment, and then discuss the pre-installation requirements that are unique to different supported platforms.

### **Verifying System Capabilities and Resources**

Before installing any Watson Explorer module, you should consider examining the configuration and capabilities of the system on which it is installed to ensure that it provides a suitable hardware and software environment for running the Watson Explorer module(s) that you are installing. System capabilities are especially critical for systems on which you are installing Watson Explorer Engine, because the enterprise information applications that you create with Watson Explorer Engine can require substantial network and system resources in order to crawl remote data, index that data, search large indices, return large result sets, and so on.

To help determine whether a given computer system satisfies Watson Explorer hardware and performance requirements, IBM includes an application named **micro-test** to measure the actual real-world performance of any server or test system on which you plan to run any Watson Explorer module. When executed, the **micro-test** utility performs one or more program execution, read-write, and system analysis tests and records summary data about the results and performance of those test. These results can then be used by customers, product support, and professional services personnel to determine whether a given server is suitable for deploying one or more Watson Explorer modules and satisfy your requirements.



The **micro-test** utility (**micro-test.exe** on Microsoft Windows systems) is installed as part of any Watson Explorer Engine installation, and is located in the **bin** sub-directory of the product's installation directory. Documentation for the **micro-test** utility is available as part of the Watson Explorer Knowledge Center.

IBM strongly recommends that you execute the **micro-test** utility on each host where you have installed any Watson Explorer module, and that you do so and archive the results of its tests before starting any Watson Explorer module for the first time. This will enable the **micro-test** utility to collect information about the default performance of each prospective host, and will also alert you to the presence of existing system software, such as anti-virus software, that you will need to configure so that it does not interfere with the module(s) that you have installed. IBM recommends that you execute the **Test All** batch test, in order to collect data about all of the aspects of system performance and configuration that the **micro-test** utility can collect.

**Note:** The **micro-test** utility is designed to review current system capabilities and performance, and should not be considered a general-purpose benchmarking utility.

## Enabling Software Installation on Linux Systems

**Note:** The instructions in this section are only necessary to perform if you are going to install as an administrator, into system directories such as `/opt`. You do not need to disable SELinux if you are going to install in non-admin mode, which installs them into your home directory.

If you are running a version of Linux that includes support for AppArmor or Security-Enhanced Linux (SELinux), please disable them before installing any Watson Explorer module. You can disable SELinux policy enforcement while installing Watson Explorer modules by executing the following command as the root user:

```
/usr/sbin/setenforce 0
```

Executing this command will put SELinux in permissive mode, where violations of SELinux policies are logged but not prevented, until the next time you boot your system or until you execute the following command, which will put your system back in enforcing mode:

```
/usr/sbin/setenforce 1
```

All Watson Explorer modules require access to resources that are typically managed by SELinux, and will therefore not function correctly on systems where SELinux is actively enforcing its default policies. Contact product support for information about creating SELinux policies that enable the Watson Explorer modules to coexist with active SELinux policy enforcement.

## Pre-Installation Tasks for Watson Explorer Engine

The next few sections explain information and configuration that is required to successfully install and configure Watson Explorer Engine on supported operating systems.

**Collecting Configuration Information for Linux Installations:** As part of the installation process for Watson Explorer, you will be prompted for the Linux user and group who will own the files and directories in the final installation. This is especially important for Watson Explorer Engine, which is either integrated into an existing web server installation or runs an embedded web server that is included

with the product, but is a requirement of other Watson Explorer and related product modules because some of the files and directories that are used by Watson Explorer must be writable by both a specific user and a group which that user belongs to. The user that you specify during installation must also be a member of the group that you specify.

If you are installing Watson Explorer Engine in administrator mode on a Linux system, you will probably want to integrate it with a system web server such as the Apache web server, and should collect the information discussed in “Collecting User/Group Information for Privileged Linux Installations” before beginning your Watson Explorer installation.

If you are installing Watson Explorer Engine in non-admin (un-privileged) mode, you should collect the information discussed in “Collecting User/Group Information for Non-Admin Linux Installations” before beginning your installation.

*Collecting User/Group Information for Privileged Linux Installations:*

Watson Explorer Engine has additional installation requirements than other Watson Explorer modules because it is commonly integrated with a system web server. Before installing Watson Explorer Engine on a Linux platform and integrating it with an existing web server, you will need to know the user and group under which the web server on your system runs. On modern Linux systems, these are typically the user and group **apache** or **www-data**. If you are unsure which of these users and groups is used on your system, you can find out by running the **ps** command at a shell prompt on your Linux system, as in the following example:

```
# ps -eo user,group,comm | egrep 'apache|httpd'

root      root      httpd
apache    apache    httpd
apache    apache    httpd
apache    apache    httpd
```

The three columns of this output identify the user, group, and name of every process that contains either the strings "apache" or "httpd". This output shows that, while the **root** user and group started the first **httpd** process, the others are running as the user and group **apache**, as expected. If you are running a web server other than the Apache web server (which is not recommended), you would change the processes that you are searching for (apache|httpd, in the previous example) to contain the name of your web server's executable.

*Collecting User/Group Information for Non-Admin Linux Installations:*

If you will be installing in non-admin mode on a Linux system, you will need to supply the name of a group of which the user who is doing the installation is a member. To display the list of all groups that this user is a member of, execute the **groups** command at a shell prompt on your Linux system, as in the following example:

```
# groups
users engdev documentation testing root
```

You will need to specify the name of one of these groups when prompted to do so during the non-admin mode installation process. You should generally specify a group that does not have any special system privileges.

## Pre-Configuring Microsoft IIS on Windows Systems:

If you are installing Watson Explorer Engine in administrator mode, to install and use that module with Microsoft Internet Information System (IIS) 7 or higher web servers, specific configuration options must be enabled in those web servers.

### Before you begin

If the following conditions are true, ensure that Microsoft Internet Information System (IIS) is installed:

- You will install Watson Explorer Engine in admin mode.
- You will not use the embedded web server that is included with Watson Explorer Engine.

**Important:** Regardless of the version of IIS that you use, you might need to manually configure some aspects of that virtual directory.

If you will install Watson Explorer Engine in non-admin mode on a Windows system, you do not need to configure IIS to enable integration of Watson Explorer Engine with IIS. Versions of Watson Explorer Engine that were installed in non-admin mode can be accessed only through the embedded web server that is included with Watson Explorer Engine.

Installing and configuring IIS is not required on systems on which you will install only the Results Module or Application Builder, because these Watson Explorer modules include their own WebSphere web servers.

### About this task

When installing Watson Explorer Engine, the installer provides integrated support for creating virtual directories in IIS on systems that are running IIS 7 or higher. For IIS 7 or higher, the installer requires that certain IIS components are installed and activated in your IIS configuration. The following components are required:

- CGI
- ISAPI Extensions
- ISAPI Filters
- IIS Management Scripts and Tools

To verify that the required components are installed and active, do the following on the system where IIS 7 or higher is installed and on which you will install Watson Explorer Engine:

### Procedure

1. Open Server Manager:
  - Microsoft Windows Server 2008: Click **Start > Administrative Tools > Server Manager**. The Server Manager window displays.
  - Microsoft Windows Server 2012: From the desktop, hover your cursor in the upper- or lower- right corner of the screen to show the charms, click the **Start** charm, and then the **Server Manager** tile.
2. Select the web server:
  - Microsoft Windows Server 2008: In the left pane of the Server Manager window, expand **Roles** and select **Web Server (IIS)**.

- Microsoft Windows Server 2012: In the left pane of the Server Manager window, select **IIS**.
3. Verify that the following role services are installed: **CGI, IIS Management Scripts and Tools, ISAPI Extensions, and ISAPI Filters**.
    - Microsoft Windows Server 2008: In the Web Server (IIS) pane, scroll all the way to the end of that pane to find the **Role Services** section. Role services are listed in alphabetical order by role, role group, and then role services:
      - CGI, ISAPI Extensions, and ISAPI filters are located in **Web Server (IIS) > Web Server > Application Development**.
      - IIS Management Scripts is located in **Web Server (IIS) > Web Server > Management Tools**.
    - Microsoft Windows Server 2012: In the right pane of the Server Manager window, scroll all the way to the end of that pane to find the **ROLES AND FEATURES** section.
  4. If some or all of the required role services are not installed:
    - Microsoft Windows Server 2008: From the **Role Services** section, click **Add Role Services**, select the required information in the initial pages of the wizard, and advance to the Select Role Services page.
    - Microsoft Windows Server 2012: From the **TASKS** menu of the **ROLES AND FEATURES** section, select **Add Roles and Features**, select the required information in the initial pages of the wizard, and advance to the Server Roles page.
  5. If any of the required role services are not installed, select them from the Roles list. The following table lists the required role services and their locations:

Role service	Location in the Roles list
CGI	Web Server (IIS) > Web Server > Application Development
ISAPI Extensions	Web Server (IIS) > Web Server > Application Development
ISAPI Filters	Web Server (IIS) > Web Server > Application Development
IIS Management Scripts and Tools	Web Server (IIS) > Web Server > Management Tools >

6. Advance to the confirmation page and click **Install**.

### What to do next

If you had to install the CGI role service, restart IIS:

- Microsoft Windows Server 2008: In Server Manager, expand **Web Server (IIS)**, select **Internet Information Services (IIS) Manager**, and from the Connections pane, select the name of your web server. From the Actions pane, click **Restart**.
- Microsoft Windows Server 2012: From the desktop, hover your cursor in the upper- or lower- right corner of the screen to show the charms, click the **Start** charm, and then the **Internet Information Services (IIS) Manager** tile. From the Connections pane, select the name of your web server. From the Actions pane, click **Restart**.

At this point, you can proceed with the Watson Explorer Engine installation process.

## Selecting and Preparing Installation Media

Watson Explorer and related modules can be installed from the following sources:

- physical DVD media that you ordered and received from IBM or a partner
- an installation archive file in zip format that you downloaded from IBM after authenticating using your Passport Advantage credentials

The next sections explain how to make those installation sources available on Linux or Microsoft Windows systems.

### Accessing DVD Installation Media

This section explains how to make Watson Explorer DVD installation media accessible on Linux and Windows systems so that you can begin the installation process.

### Mounting DVD Media on Linux Systems

If you received Watson Explorer from IBM on physical DVD media, inserting that DVD into a DVD drive on a Linux server may mount it automatically. If this is the case, the DVD will probably be mounted with a name of the form `/media/WEX_VERSION-EDITION`, where:

- *VERSION* is a text string that identifies the version of the Watson Explorer software that is on the DVD, such as **v11.0.2**
- *EDITION* is the edition of the Watson Explorer software that is on the DVD, such as **EE** (enterprise edition)

Listing that directory should show the files and directories that are located at the root of the ISO image or physical DVD. If this is the case, change directory to that directory on your system and proceed to the end of this topic for instructions on how to proceed with the installation process.

More commonly, auto-mounting DVDs is disabled on enterprise Linux systems. In this case, you must manually mount the device that contains the DVD on a directory on your system. To mount your DVD device on a Linux system, execute a command like the following as the root user or via **sudo**:

```
/bin/mount -t iso9660  
DVD-DEVICE MOUNT-DIR
```

In this example command, *DVD-DEVICE* is the name of the device node for the device in which you inserted the DVD, and *MOUNT-DIR* is the name of a directory on your system where you want to mount the DVD. The most common device node for a DVD drive is `/dev/sr0`, though this may differ if your system contains multiple CD or DVD drives. The typical directory on which new filesystems are mounted manually is the directory `/mnt`.

If this command executed correctly, listing the `/mnt` directory should show the files and directories that are located at the root of the physical DVD. If this is the case, change directory to the `/mnt` directory on your system.

Proceed to the end of this topic for instructions on how to proceed with the installation process.

### Using Local Installation Media on Windows Platforms

On Microsoft Windows systems, inserting a DVD into a DVD drive will normally make the contents of that DVD available through the drive letter that is associated

with the DVD drive. Most Windows servers are configured so that they do not automatically execute applications from removable media.

## Extracting the Contents of a Downloaded Installation Archive

The Watson Explorer installation media can also be downloaded as an archive file in zip format that contains the same hierarchy of files that is found on physical DVD media, and is therefore referred to as an **installation archive**. This section explains how to extract the contents of a downloaded Watson Explorer installation archive on Linux and Windows systems so that you can begin the installation process.

### Expanding Zip Archives on Linux Systems

If you have downloaded a Watson Explorer installation archive for use in installing Watson Explorer or related modules on a Linux system, do the following to prepare for the actual install process:

1. Change directory to the directory where you downloaded the Watson Explorer installation archive and execute the following command from a terminal window or console:

```
unzip IMAGE-FILE-NAME
```

Replace *IMAGE-FILE-NAME* with the name of the installation archive file that you downloaded. Executing this command creates a directory with the name of the release contained in the archive, such as WEX\_v11.0.0-EE, in your working directory.

**Note:** If the **unzip** command is not installed on the system where you are installing Watson Explorer or related modules, you can install it by executing the following command as the root user or via the **sudo** command:

```
yum install unzip.x86_64
```

2. Change directory to the directory that was created in the previous step.

### Expanding Zip Archives on Windows Systems

If you have downloaded a Watson Explorer installation archive for use in installing Watson Explorer or related modules on a Windows system, you can use Windows built-in support for archive files in zip format to extract the contents of that archive so that the extracted files and directories can be used for installation. To extract the contents of a zip archive file on Microsoft Windows systems:

1. Right-click on the downloaded archive file and select **Extract All...** from the context-sensitive menu that displays. The Extract Compressed (Zipped) Folders dialog displays.
2. Click **Browse** to navigate to the location where you want the files and directories that are contained in the archive to be extracted.
3. Click **Extract** to begin the extraction process. A progress dialog displays as the contents of the archive are extracted, and closed once the extraction process is complete.

## Installing Watson Explorer Version 11.0.2

To install Watson Explorer version 11.0.2 or later modules, follow the instructions below.

IBM strongly recommends that you are logged in as an Administrator to install Watson Explorer. Installing Watson Explorer requires administrative privileges on both Microsoft Windows and Linux.

In Microsoft Windows, you can install Watson Explorer modules as a non-administrative user, but you must have elevated permissions to do so, see this article for details. However, non-administrators will have certain limitations, for example, you will not be permitted to start the services.

The available Watson Explorer foundational component modules are:

Module	Module file extensions: Linux:.rpm, Windows: .msi
Watson Explorer Engine	wex-engine-11.0.2.x
Zookeeper	wex-zookeeper-11.0.2.x
Application Builder	wex-appbuilder-11.0.2.x
Results Module	wex-resultsmodule-11.0.2.x
BigIndex	wex-bigindex-11.0.2.x
Knowledge Center	wex-knowledgecenter-11.0.2.x

If you wish to install Query Modifier after you install Watson Explorer, see [Installing Query Modifier](#) for instructions.

The Watson Explorer Annotation Administration Console module is also included with Watson Explorer 11.0.2.x, in an archive named: wex-aac-11.0.2.x. See [Installing Annotation Administration Console](#) for installation instructions.

**Note:** Antivirus software, firewalls, and other network security mechanisms can interfere with the installation of Watson Explorer modules. You should disable them before beginning your installation. If you are running a version of Linux that includes support for AppArmor or Security-Enhanced Linux (SELinux), you must disable them before installing any Watson Explorer module.

## Installing Watson Explorer Modules

### About this task

Each Watson Explorer module needs to be installed individually.

### Procedure

1. Run the installation file for your operating system. See [Installing Watson Explorer](#) for the list of installation module names.

Option	Description
Linux Command Line	<p>Start a terminal session (xterm, gnome-terminal, or other terminal application) and use the su or sudo -s command to become the root user.</p> <p>Enter rpm -ivh &lt;file_path&gt;/nameofmodule.rpm</p> <p><b>Note:</b> For a silent install, enter rpm -i &lt;file_path&gt;/nameofmodule.rpm</p> <p>To install to a different directory, enter rpm -ivh --prefix=/alternate/location nameofmodule.rpm</p> <p>Guide to rpm flags</p>

Option	Description
Windows Graphical	<p>Choose the appropriate module installation file from the download location.</p> <p>Double-click on the module installation file.  <b>Note:</b> You can also right-click and choose <b>Install</b> from the menu, but never choose the <b>Repair</b> option.</p>
Windows Command Line	<p>Enter <code>msiexec /i &lt;file_path&gt;\nameofmodule.msi</code> to open the Windows Graphical install.  <b>Note:</b> For a silent install, enter <code>msiexec /i &lt;file_path&gt;\nameofmodule.msi&lt;/file_path&gt; /qn</code></p> <p>To install to a different directory, enter <code>msiexec /i &lt;file_path&gt;nameofmodule.msi&lt;/file_path&gt; /qn INSTALL_ROOT = D:\&lt;file_path&gt;alternate\location&lt;/file_path&gt;</code></p> <p>Guide to msiexec flags</p>

2. Accept the license agreement. If you choose not to accept the agreement the installation will stop.
3. Confirm the destination location for the install.
4. Finish installing the module.
5. Repeat for each module.

## Configuring Watson Explorer Modules

After installation, each installed module needs to be configured. The amount of configuration required varies by module. Most of the Watson Explorer modules have a configuration tool that needs to be run. This tool is used to configure basic settings, such as port location. Additional configuration tasks will be required for some modules.

Default configuration file locations :

- Linux: /opt/ibm/WEX/nameofmodule
- Windows: \Program Files\IBM\WEX\nameofmodule

**Note:** You must configure Zookeeper before you configure Application Builder. The other modules may follow in any order.

Module	Configuration File Name (Microsoft Windows add .exe extension)
Zookeeper	zookeeper-config
Application Builder	appbuilder-config
Results Module	resultsmodule-config
Knowledge Center	knowledgecenter-config
Watson Explorer Engine	embedded-webserver-config



**Note:** There is no configuration tool for the BigIndex module.

After configuring each module, you should integrate it into your system's startup and shutdown processes. See Integrating Other Watson Explorer Modules Into the Startup/Shutdown Process.

**Note:** If you use command line switches to configure a module (for example, `appbuilder-config -quiet -<option_name> <option_value>`), be aware that if you do not specify the `-enable-service` option then any service that was previously created by the configuration tool will be deleted. For example, if you issue the following commands, the first command would create the service, and the second command would change the client port and delete the service you just created.

```
appbuilder-config -quiet -enable-service
appbuilder-config -quiet -clientPort 2182
```

**SILENT CONFIGURATION:** You may configure Watson Explorer modules in non-interactive mode (also referred to as silent or quiet mode) after initially installing and configuring each module. To do so, add the `-quiet` flag to the configuration file call or write a Bash, Powershell, or command prompt script wrapper.

## Configuring Watson Explorer Engine

There are several steps to configure Watson Explorer Engine before starting it up.

They are:

- Generating a new encryption key
- Running the micro-test utility
- Configuring the embedded webserver

See System Administration for tips on integrating Watson Explorer Engine into your IT policies and procedures.

**Note:** As noted in the System Requirements for Watson Explorer version 11.0.0.1 or later, your web browser must have Javascript enabled in order to use the Watson Explorer Engine administration tool.

### Generating a New Encryption Key with the Encryption Key Utility:

Passwords and process-to-process communication in Watson Explorer Engine are secured with the AES-128-GCM encryption algorithm, which uses 128 bit keys, and OpenSSL. A default encryption key is included when you install Watson Explorer Engine. IBM recommends that you generate a new, unique key after installation for maximum security.

### About this task

All of the following in Watson Explorer Engine will be encrypted using this encryption method:

- All Distributed Indexing communications.
- All passwords that are used to authenticate to external systems (e.g. the passwords needed for connectors in seeds, for remote push, collaboration features, etc.) will be stored in an encrypted form on disk.
- Interprocess communication where passwords need to be transmitted.

**Note:** Indexes remain unencrypted in Watson Explorer Engine.

Additional note regarding security: Beginning with Watson Explorer version 11.0.0.1, all Watson Explorer user passwords are secured using the scrypt algorithm, which generates one-way cryptographic hashes. These user passwords are stored in users.xml.

**Warning:** Proceed with caution when generating a new encryption key. This key is used to encrypt and decrypt everything listed above.

- If this key is stolen, all encrypted data would be at risk. It is strongly recommended that this encryption key be stored in a data vault with proper safeguards.
- If it were lost, all encrypted passwords would need to be reentered, and all system services would need to be restarted.

**Important:** You should generate ONE KEY on ONE SERVER and copy it to all other servers. Do not generate multiple keys. If you start using Watson Explorer, and run the utility after that, the encryption key will be replaced - and you risk being unable to decrypt the data encrypted with the original key.

### Procedure

1. To generate a new encryption key:

Option	Description
Linux	Start a terminal session (xterm, gnome-terminal, or other terminal application) and use the su or sudo -s command to become the root user. Enter <file-path>/generate-key. <b>Note:</b> Run generate-key -help if you would like to view additional system information before running the utility. This command does not run the utility. See generate-key to learn more about the <b>generate-key</b> command.
Windows	From a command prompt, run <file-path>\generate-key.exe. This command is located in <install_dir>\Engine\bin. <b>Note:</b> Run generate-key.exe -help if you would like to view additional system information before running the utility. This command does not run the utility.

You will receive this message: "WARNING: An encryption keyfile already exists on disk. Any data encrypted using this key cannot be decrypted after generating a new key." (The encryption keyfile this warning is referring to is the default id\_vcrypt key included with Watson Explorer.) Followed by this message: "Please confirm that you wish to continue [Y/N]"

2. After you enter Y, you will receive the following messages:
  - "The existing keyfile has been backed up to <install\_dir>\Engine\data\static\id\_vcrypt.bak" (The old encryption key is backed up to this location in the event you accidentally generate a new key and need to restore the old one.)
  - "A new key has been generated and written to <install\_dir>\Engine\data\static\id\_vcrypt"
3. Navigate to the id\_vcrypt file to verify the key has been generated. (This file is located by default in <install\_dir>\Engine\data\static\)

4. Immediately store a copy of the encryption key in a safe place (such as a data vault) and secure it using your company's best practices. You should also safeguard the key stored in your Watson Explorer installation.
5. Copy the key to the static directory of all other Watson Explorer servers that have Watson Explorer Engine installed. The default location of this directory is `<install_dir>\Engine\data\static\`

### **Run the micro-test Utility for Engine:**

IBM strongly recommends that after installing Watson Explorer Engine, you should run the micro-test utility included with Watson Explorer and archive the results.

### **About this task**

The micro-test utility helps measure the actual real-world performance of any server that will be running Watson Explorer Engine. When executed, the micro-test utility performs a series of program execution, buffer, and read-write tests. It is recommended that you execute the Test All batch test, in order to collect data about all of the aspects of system performance and configuration that the micro-test utility can collect.

**Note:** The micro-test utility is designed to review current system performance and should not be considered a benchmarking utility. For a list of the tests performed by micro-test and the usage options, see the Using the micro-test Utility.

#### Linux:

- Start a terminal session (xterm, gnome-terminal, or other terminal application) and use the su or sudo -s command to become the root user.
- Enter `<file-path>/micro-test --all`. This command is located in `<install_dir>/Engine/bin`. Other options can be found in the list of Usage and Defaults.

#### Windows:

- From a command prompt, run `<file-path>\micro-test.exe --all`. This command is located in `<install_dir>/Engine/bin`. Other options can be found in the list of Usage and Defaults.

**Note:** On Microsoft Windows systems, you must run `micro-test.exe` from a command prompt. Using a Windows Explorer instance to run the utility causes the results to disappear when the program ends.

### **Configuring the Embedded Web Server for Development Environments:**

An embedded web server is included with Watson Explorer Engine for development (testing) purposes. It is not for production.

See Configuring Watson Explorer Engine's Embedded Web Server for instructions. For information on configuring Watson Explorer Engine for a production environment, see Configuring Watson Explorer Engine for Production Environments.

**Important:** If you enable Watson Explorer Engine's embedded web server, make sure that the port that it uses is not blocked by a network firewall on any system where Watson Explorer Engine is installed. This is the most common cause of

being unable to contact Watson Explorer Engine through the embedded web server after you have started the embedded web server.

*Configuring Watson Explorer Engine's Embedded Web Server:*

**Note:** An embedded web server is included with Watson Explorer Engine for development (testing) purposes. It is not to be used in production environments. For information on configuring Watson Explorer Engine for a production environment, see *Configuring Watson Explorer Engine for Production Environments*.

Watson Explorer Engine includes an embedded web server that can be used to isolate Watson Explorer Engine from any existing web server that is running on your system, or to provide a dedicated web server for use with Watson Explorer Engine API applications. How this web server runs on your system and any security that it uses are configured using the **embedded-webserver-config** command, which is explained in *embedded-webserver-config Command*.

**Important:** If you are using Watson Explorer Engine on a Microsoft Windows system, you should not attempt to use both Microsoft's IIS web server and Watson Explorer Engine's embedded web server to interact with the Watson Explorer Engine administration tool. These web servers typically execute as different users and groups, start processes using different user/group identities, and require different file permission and ownership settings on your Watson Explorer Engine installation. Use either one of these web servers exclusively when using the Watson Explorer Engine administration tool.

Any of the Watson Explorer Engine administration commands discussed in this section or in *System Administration* must be executed with at least the system privileges that you used when installing Watson Explorer Engine. In other words, if you installed Watson Explorer Engine as the root user on a Linux system or as Administrator on a Microsoft Windows system, you must execute the Watson Explorer Engine administration commands as that user or as another user with equivalent or superior privileges.

See *Users and File Permissions* for information about starting the embedded web server with administrative privileges but executing the web server process as another user. If you encounter an error when executing any of the **embedded-webserver-config** commands, see *Command-Line Watson Explorer Engine Administration Commands* for general information about resolving errors related to Watson Explorer Engine's administrative commands.

To start and access the embedded web server:

1. Use the **embedded-webserver-config** command (Linux) or **embedded-webserver-config.exe** command

(Microsoft Windows) to:

- a. Enable the embedded web server, using the following command:  
`embedded-webserver-config webservice/is-enabled=True`
- b. Configure the user and group that the embedded web server should run as, using commands like the following:

```
embedded-webserver-config global/user=apache  
embedded-webserver-config global/group=apache
```

On a Linux system, you must configure the user and group that the embedded web server should run as.

Setting the user and group configuration options is optional on a Microsoft Windows system, and depends on the user as whom you installed Watson

Explorer Engine. See Web Server User Identity and File Permissions for detailed information about the user, file, and directory permissions that are required to create and modify Watson Explorer Engine data.

You may also want to set other, optional configuration parameters. See `embedded-webserver-config` Command for detailed information about available configuration options and about using the command in general.

2. Do one of the following to actually start the web server:
  - (Recommended) The **engine-start** command (Linux) or **engine-start.exe** command (Microsoft Windows) starts the web server, enable access to Watson Explorer Engine, and optionally starts other Watson Explorer Engine related processes. (See `engine-start` for detailed information about using this command.)
  - Use the **embedded-webserver** command (Linux) or **embedded-webserver.exe** command (Microsoft Windows) to manually start the web server, which is explained in `embedded-webserver`. This command will start the web server, but you may still not be able to access Watson Explorer Engine, depending on how it was last shut down on your system.

Once you have started Watson Explorer Engine's embedded webserver, you can complete your Watson Explorer Engine installation by accessing the Watson Explorer Engine administration tool through the embedded web server. If you do not know how to contact Watson Explorer Engine's embedded web server, you can use the **is-running** option of the **embedded-webserver** command (Linux) or **embedded-webserver.exe** command (Microsoft Windows) to show a URL that you can use to contact the web server.

For information about integrating the **engine-start** command into your system's startup and shutdown processes, see `Command-Line Watson Explorer Engine Administration Commands`.

**Note:** See `Additional Command-Line Tools` for information about the command-line tools available in Watson Explorer Engine version 11.0.0.3 or later.

*embedded-webserver-config* Command:

Enables you to configure Watson Explorer Engine's embedded web server, including enabling whether the web server is started automatically as part of the **engine-start** command.

### Synopsis

```
embedded-webserver-config
  [-h|-help]
  [-installation-dir /path/to/engine/installation]
  [-quiet]
  [name=value]
```

### Description

The **embedded-webserver-config** command is located in the `bin` subdirectory of a Watson Explorer Engine installation. This command enables you to display and set

configuration information for Watson Explorer Engine's embedded web server by specifying *name=value* pairs on the command-line. This configuration information is stored in the file `data/embedded-webserver-config.xml`. This file will only exist if you have modified the configuration of the embedded web server. If the file does not exist, the default configuration values (listed later in this section) are used.

Executing the **embedded-webserver-config** command with no options displays the configuration options that are available for the embedded web server and their current values. The command and its output will look something like the following (which shows the default values for all of these options):

```
$ embedded-webserver-config --help
-base-url value
  The virtual directory Engine will use with the embedded webserver (default vivisimo)
-disable-
  Disable debugging mode for the embedded webserver
-disable-webserver
  Disable the embedded webserver
-enable-
  Enable debugging mode for the embedded webserver
-enable-webserver
  Enable Engine's embedded webserver.
-help or -?
  Display this message
-host value
  The hostname to use with Engine.
-language value
  Specifies the language that the configuration prompts use. (default en)
-port value
  The port the embedded webserver listens on (default 9080)
-quiet
  Run in quiet mode; only fatal errors are reported
-service-name value
  The name of the service that the embedded webserver will register (default EngineWebServer)
-user value
  The user to run the embedded webserver's processes as. (default Administrator)
```

**Note:** On Microsoft Windows systems, the **embedded-webserver-config** command does not display the **global/group** option. On Linux systems, this command does not display the **global/password** and **webserver/service-name** options. See the description of these options later in this section for more information about their use and possible values.

As you can see from this example, the **embedded-webserver-config** command enables you to set two classes of configuration options: **global** values, which will affect the webserver but which are intended to be used by other Watson Explorer Engine components, and **webserver** values, which only affect the configuration of the embedded web server. To set any of these values, specify the name of the option that you want to set and the value that you want to set it to, separated by an equals sign (`=`). For example, to set the port that the webserver uses to port 10025, you would execute the command `embedded-webserver-config webserver/port=10025`. This command and its output would look like the following:

```
$ embedded-webserver-config webserver/port=10025
webserver/port 10025
Configuration saved...
```

The **embedded-webserver-config** command does not validate the values of any parameters that you set. For example, port numbers must be less than 65536, but are not checked when you set them.

**Note:** The **global** values that you can configure using the **embedded-webserver-config** command are currently only used by the Watson Explorer Engine embedded web server.

On Microsoft Windows systems, the **embedded-webserver-config** command has the `.exe` file extension.

## Options

- **-h|-help**: Displays a usage message and exits without performing any other actions.
- **-installation-dir/path/to/engine/installation**: Enables you to specify the location of the Watson Explorer Engine installation that contains the web server which you want to configure. If no installation directory is specified, the **embedded-webserver-config** command assumes that the webserver directory and binaries are located in the Watson Explorer Engine installation directory which is the parent of the directory in which the **embedded-webserver-config** application is located.
- **-quiet**: Suppresses normal output and warnings. Errors will still be displayed.

## Configuration Items

The following list shows the configuration items for which values can be specified using the **embedded-webserver-config** command:

**Important:** If you are using Watson Explorer Engine's embedded web server, you must restart it after changing any of the configuration items supported by the **embedded-webserver-config** command.

- **global/admin-email**: The email address shown by the webserver when an error is displayed. The default value of this configuration option is **admin@localhost**.
- **global/admin-url**: The URL that should be used to contact the Watson Explorer Engine administration tool using the embedded web server. This value cannot be set manually - it is constructed internally using the **webserver/base-url** and **webserver/host** configuration options.
- **global/api-url**: The URL that should be used to execute Watson Explorer Engine startup, shutdown, and API commands. The default value for this configuration option is **<unset>**. When this value is not set:
  - If Watson Explorer Engine's embedded web server is not running, Watson Explorer Engine will attempt to derive this URL from **localhost**, the default port number for the web server, the virtual directory (base-url) that is specified in the Watson Explorer Engine installation's **vivisimo.conf** file, and the **cgi-bin** subdirectory.
  - If Watson Explorer Engine's embedded web server is running, Watson Explorer Engine will use the URL that is displayed as part of the output of the **embedded-webserver is-running** command. This URL is derived from the hostname for the current system (if available), the port number specified in the **webserver/port** option, the virtual directory (base-url) that is specified in the installation's **vivisimo.conf** file, and the **cgi-bin** subdirectory.

This option is provided for use when Watson Explorer Engine applications are not installed in their default location, and therefore cannot be accessed through the default URL.

- **global/group**: The name of the group that the web server should run as. On Linux systems, you must set both this configuration option and the **global/user** configuration option in order to start Watson Explorer Engine's embedded web server as the root user. This option is not used on Microsoft Windows systems. The default value of this configuration option is **<unset>**, indicating that no value has been set for this option yet. When this value is not set, the webserver will attempt to run as the group that is associated with the user who starts the webserver.
- **global/password**: The password associated with the user that the web server should run as (which was specified by the **global/user** configuration option). The actual text value of the password is shown as a sequence of asterisks for

security purposes, and is stored internally in a secure fashion. This option is only used on Microsoft Windows systems. The default value of this configuration option is **<unset>**, indicating that no value has been set for this option yet.

- **global/user**: The name of the user that the embedded web server should run as. On Linux systems, the embedded web server cannot run as the root user, but is typically started by a system script that is executed by the root user. You must therefore set both the **global/user** and **global/group** configuration options in order to identify the user and group identity that the root user will use to start and run the embedded web server. The specified user must be a member of the group specified by the **global/group** configuration option.

**Note:** Attempting to set **global/user** to the root user will display error messages, as in the following example:

```
# embedded-webserver-config global/user=root
ERROR: The value for 'global/user' is invalid:
      Running Watson Explorer Engine as root is insecure and not allowed
      (ID: CLI_INVALID_CONFIG_VALUE_ERROR)

WARNING: No changes were made to the stored configuration.
      (ID: CLI_NO_CONFIG_CHANGES_WARNING)
```

On Microsoft Windows systems, you must set both the **global/user** configuration option and the **global/password** option in order to start Watson Explorer Engine's embedded web server as a specified user. Some Windows installations may require that you specify the fully-qualified user name (*DOMAIN\user*) in order to correctly authenticate as a specified user.

The default value of this configuration option is **<unset>**, indicating that no value has been set for this option yet. When this value is not set on Linux systems, the webserver will attempt to run as the user who starts the webserver.

- **webserver/base-url**: The base part of the admin and query-meta URLs. This configuration option is listed for convenience only, and cannot be set by the **embedded-webserver-config** command. It can only be set by editing the `embedded-webserver-config.conf` configuration file. The default value of this option is `/vivisimo/`. This option **must** begin with a `'/'` symbol, or attempting to access Watson Explorer Engine will result in an error.
- **webserver/debug**: Whether the embedded webserver should start in debug mode (true) or not. When in debug mode, access logs and more descriptive error messages are recorded by the embedded webserver. The default value of this configuration option is **false**.
- **webserver/host**: The host name or IP address of the system on which the embedded Watson Explorer Engine web server is running.
- **webserver/is-enabled**: Whether the embedded web server is enabled (true) or not (false). If this variable is set to **true** and the embedded web server has not been removed from your Watson Explorer Engine installation, executing the **engine-start** command will automatically start the embedded web server. The default value of this configuration option is **false**.
- **webserver/is-ssl-enabled**: Whether the embedded web server uses the Secure Socket Layer (SSL), now known as Transport Layer Security (TLS), for secure communication and data integrity verification. If this variable is set to **true**, you must also provide both an SSL certificate and key. The default value of this configuration option is **false**.
- **webserver/port**: The port on which the embedded web server listens for data. The default value of this configuration option is **9080**.



- **webserver/service-name:** (Microsoft Windows only) The name of the Windows Service that is associated with Watson Explorer Engine. The default value of this option is **EngineWebServer**, which is a service that is created by Watson Explorer Engine.
- **webserver/ssl-certificate:** The SSL certificate for the embedded web server. To provide a certificate, save it to a file and specify the name of the file, preceded by an '@' sign, as the value for this configuration option. As an example, the following command would set the SSL certificate for an embedded Watson Explorer Engine web server to the contents of the file `corporate.crt`:  

```
embedded-webserver-config webserver/ssl-certificate=@corporate.crt
```

You can specify any file name as long as it contains the SSL certificate that you want to use. The default value of this configuration option is **<unset>**, indicating that no value has been set for this option yet.
- **webserver/ssl-certificate-key:** the key for the SSL certificate associated with the embedded web server. To provide a key, save it to a file and specify the name of the file, preceded by an '@' sign, as the value for this configuration option. As an example, the following command would set the SSL certificate key for an embedded Watson Explorer Engine web server to the contents of the file `corporate.key`:  

```
embedded-webserver-config webserver/ssl-certificate-key=@corporate.key
```

You can specify any file name as long as it contains a valid key for the SSL certificate that you are using. The default value of this configuration option is **<unset>**, indicating that no value has been set for this option yet.

### Exit Status

Exits with status **0** when the command is successful.

**Note:** On Linux systems, exit codes above 127 are returned by the shell. Please consult the documentation for the shell that you are using for more information.

### Configuring Watson Explorer Engine for Production Environments:

You can use Watson Explorer Engine for production on either a Microsoft Internet Information System (IIS) virtual directory (for Windows) or an Apache web server (Linux or Windows).

*Creating and Configuring an Internet Information Services (IIS) Directory:*

To use Watson Explorer Engine on Windows for production using a Microsoft Internet Information System (IIS) 8.5 or higher web server, you must create a virtual directory and configure it. A Powershell script, `createVDir.ps1`, is included to create an IIS virtual directory. You can also create an IIS virtual directory manually.

## About this task

**Note:** Before running the Powershell script, confirm that the following IIS components are installed and activated in your IIS configuration on Microsoft Windows Server 2012 R2:

- CGI
- ISAPI Extensions
- ISAPI Filters
- IIS Management Scripts and Tools

To do so, open the **Server Manager** window and verify that the following role services are installed: **CGI, IIS Management Scripts and Tools, ISAPI Extensions, and ISAPI Filters**. If you need to install any role services, restart IIS before running the Powershell script.

**Note:** Using Watson Explorer Engine with IIS also requires that execute rights be enabled for CGI applications so that your IIS web server can run executable files that are in the **www\cgi-bin subdirectory** of the directory in which you installed Watson Explorer Engine. To do so, open Internet Information Services (IIS) Manager, and for the **vivisimo** virtual directory, verify that the state of **CGI-exe** is **Enabled**. If it is not, edit the feature permissions to enable applications with the .exe extension to execute in response to CGI requests.

**Note:** Installing and configuring IIS is not required on systems that include only Results Module or Application Builder installations, because these Watson Explorer modules include their own WebSphere web servers.

To run the createVDir.ps1 Powershell script:

- From the command line, enter Powershell C:\Program Files\IBM\WEX\Engine\bin> .\createVDir.ps1 "C:\Program Files\IBM\WEX\Engine\www"
- If successful, the following message will display: Created virtual directory 'vivisimo' at 'C:\Program Files\IBM\WEX\Engine\www' for site 'Default Web Site'

If you prefer to create your IIS virtual directory manually in IIS 8.5 or higher (or have already done so), see the steps below.

## Procedure

1. Open the Internet Information Services (IIS) Manager. Create a new virtual directory by choosing **New** from the Action menu, and then clicking **Virtual Directory**.
2. A wizard will then guide you through creating a Virtual Directory. Enter the following information when prompted:
  - **Alias:** enter the new virtual directory name. The recommended alias for a manually-created virtual directory is **vivisimo**.
  - **Path:** enter the full path name of the **www** folder of the Watson Explorer Engine installation directory (by default C:\Program Files\IBM\WEX\Engine\www).
  - **Permissions:** Select **Read, Run Scripts, and Execute**.
3. Click **Finish** to save the definition for the new virtual directory.

After manually creating a virtual directory, you will need to integrate it into your Watson Explorer Engine installation if the name of the virtual directory that you created (its **Alias**) is not **vivisimo**, which is the default value for a

Watson Explorer Engine virtual directory. To tell your Watson Explorer Engine installation to use a custom virtual directory:

4. Edit the file `vivisimo.conf`, which is located in the directory in which you installed Watson Explorer Engine.
5. Add a statement like the following inside the `<options>` element:

```
<option name="base-url">VIRTUAL-DIRECTORY</option>
```

Replace `VIRTUAL-DIRECTORY` with the name of the virtual directory that you created for Watson Explorer Engine, preceded and followed by a forward slash (/).

For example, if you set the name of the virtual directory for your Watson Explorer Engine installation to search, you would add the following entry to your `vivisimo.conf` file:

```
<option name="base-url">/search/</option>
```

6. Save the updated file and restart your Watson Explorer Engine installation to ensure that it uses the new `base-url` setting when constructing internal URLs. See *Command-Line Watson Explorer Engine Administration Commands* for information on using the **engine-shutdown** and **engine-start** commands to restart your Watson Explorer Engine installation.

#### *Configuring an Apache Web Server:*

To use Watson Explorer Engine on with an Apache web server for production, you need to integrate Watson Explorer Engine with your Apache web server.

Linux:

If you are running the Apache web server on Linux server, the following command will integrate your Watson Explorer Engine installation into your Apache Web server installation:

**Note:** Watson Explorer Engine provides two different configuration files for Linux. If your operating system supports Apache 2.2 use `apache-2.2.conf`; if it supports Apache 2.4; use `apache-2.4.conf`.

```
cd /etc/httpd/conf.d  
cp /opt/ibm/WEX/Engine/apache-2.X.conf
```

**Note:** On Linux systems, creating a symbolic link to the Apache configuration file can cause the Apache web server to fail if you subsequently move your Watson Explorer Engine installation, because the symbolic link will be unresolvable. If you create a symbolic link, it is strongly recommended that the ownership of that file be changed to root. Otherwise, there will be a security issue, because a non-root user can then modify a file effectively executed by root. It is also recommended that you check initialization script performance.

Windows:

If you are running the Apache web server on a Windows system, you can use the DOS or Powershell **copy** command and substitute the appropriate paths in these instructions to add the Watson Explorer Engine configuration data to your Apache installation.

**Note:** If multiple versions of Apache are installed on your system, make sure that you integrate Watson Explorer Engine into the Apache installation that will provide access to the Watson Explorer Engine administration tool. By default, the

Apache integration of Watson Explorer Engine assumes that Apache is configured to allow access to all virtual directories. If you have Apache 2.2 installed, and it is configured with more restrictive permissions, you may need to un-comment the **Allow from all** directive in the apache-2.2.conf file before copying that file or after creating a symbolic link to it. You must then restart your web server before you can access the virtual directory that is used by Watson Explorer Engine. Permissions are handled differently in Apache 2.4, so no changes should be needed to the provided apache-2.4.conf file.

For more detailed information about Apache configuration directives, see the Apache web server documentation.

**vivisimo.conf and Watson Explorer Engine Environment Variables:** In a typical installation, the global settings for your installation are contained in a file at the top level of your installation called vivisimo.conf. For a Linux installation, the default directory is /opt/ibm/WEX/Engine. For a Windows installation, the default directory is \Program Files\IBM\WEX\Engine. Administrators do not normally need to modify this file.

The Watson Explorer Engine software will search for this file by starting from the directory where the binary is located and continuing with containing directories until the file is found. The file can contain any **option** tag. As shown in the previous section, the default file will contain the installation directory setting and the base-url for the installation.

Environment variables are variables that are set in the execution environment of an application. On Microsoft Windows systems, they are set at the system level, whereas on Linux systems, the environment variables used by Watson Explorer Engine are set in the Apache web server's execution PNU.

The following table contains the Watson Explorer Engine options that are typically specified in vivisimo.conf. Most of these options can alternatively be configured as an environment variable. The following table also shows the correspondence between option name and environment variable name.

If the same option is specified in both vivisimo.conf and as an environment variable, the environment variable's setting takes precedence.

*Table 1. Options and Equivalent Environment Variables*

Watson Explorer Engine Option	Environment Variable	Description
admin-authentication-macro	VIVISIMO_ADMIN_AUTHENTICATION_MACRO	The Admin Authentication Macro that will be loaded before any search application in the Watson Explorer Engine administration tool. This macro should contain custom authentication code (which may, for example, check user credentials via LDAP or Active Directory). This value cannot be set in your project using the administration tool or XML mode.

Table 1. Options and Equivalent Environment Variables (continued)

Watson Explorer Engine Option	Environment Variable	Description
authentication-macro	VIVISIMO_AUTHENTICATION_MACRO	The name of a macro that will be loaded before any search application in the query-meta CGI application. This macro should contain custom authentication code (which may, for example, check user credentials via LDAP or Active Directory). This value cannot be set in your project using the administration tool or XML mode. By default, it is set to authentication.
base-url	VIVISIMO_BASE_URL	The base URL for a local Watson Explorer Engine installation. The default value for this option is /vivisimo/. This option <b>must</b> begin with a '/' symbol, or attempting to access Watson Explorer Engine will result in an error.
cleaner.access-time	VIVISIMO_CLEANER_ACCESS_TIME	The temporary files in tmp-dir that have not been accessed in at least this many seconds. The default value for this variable is -1, meaning it is not active.
cleaner.create-time	VIVISIMO_CLEANER_CREATE_TIME	The temporary files in tmp-dir created at least this many seconds ago. The default value for this variable is -1, meaning it is not active.
cleaner.file-limit	VIVISIMO_CLEANER_FILE_LIMIT	The maximum number of temporary files left in the tmp-dir after the cleaner has been called. The default value for this variable is -1, meaning it is not active.
cleaner.frequency	VIVISIMO_CLEANER_FREQUENCY	How often cleaner will run approximately once every cleaner.frequency calls to vivisimo_destroy function to clean the temporary files generated in tmp-dir. The default value for this variable is -1, meaning it is not active.

Table 1. Options and Equivalent Environment Variables (continued)

Watson Explorer Engine Option	Environment Variable	Description
cookie-path	VIVISIMO_COOKIE_PATH	The path used when Watson Explorer Engine sets cookies in the user's browser. If unspecified, the value for base-url is used.
debug-dir	VIVISIMO_DEBUG_DIR	The full or relative pathname of a directory in which debugging information will be stored. Debugging must be enabled in the project or as a URL parameter in order for debugging information to be produced by Watson Explorer Engine. If this option is set to a directory that is not relative to the Watson Explorer Engine installation directory, the user or group under which your web server and the Watson Explorer Engine administration tool are running must be able to read from and write to this directory. By default it is set to the debug sub-directory of the Watson Explorer Engine installation.
default-project	VIVISIMO_DEFAULT_PROJECT	The project loaded by default in the query-meta CGI application. The default is query-meta.
install-dir	VIVISIMO_INSTALL_DIR	The full pathname of the directory in which Watson Explorer Engine is installed. On Linux systems, the default is /opt/IBM/WEX/Engine. On Microsoft Windows systems, the default is C:\Program Files\IBM\WEX\Engine.
key	VIVISIMO_LICENSE	The license key of your Watson Explorer Engine installation.

Table 1. Options and Equivalent Environment Variables (continued)

Watson Explorer Engine Option	Environment Variable	Description
meta.cache-dir	<i>none</i>	The full or relative pathname of the directory in which cache files are saved. By default, this is the cache subdirectory of the Watson Explorer Engine installation directory. If this option is set to a directory that is not relative to the Watson Explorer Engine installation directory and caching is enabled, the user or group under which your web server and the Watson Explorer Engine administration tool are running must be able to read from and write to this directory.
reporting-dir	<i>none</i>	The full pathname of the directory in which the database files that hold raw application reporting information are stored. These files contain information about application queries, not system reporting data (which is stored in the location identified by the <b>system-reporting-dir</b> option). If this option is set to a directory that is not relative to the Watson Explorer Engine installation directory, the user or group under which your web server and the Watson Explorer Engine administration tool are running must be able to read from and write to this directory. By default, reporting files are stored in the data/reporting directory of your installation directory.
repository	VIVISIMO_REPOSITORY	The full or relative path of the XML file containing templates, sources, parsers, forms and syntax which will be retrieved when needed by Watson Explorer Engine.

Table 1. Options and Equivalent Environment Variables (continued)

Watson Explorer Engine Option	Environment Variable	Description
search-collections-dir	VIVISIMO_SEARCH_COLLECTIONS_DIR	<p>The full pathname of the directory that will be used to store Watson Explorer Engine search collection data. By default, search collection data is stored in the data/search-collections directory of your installation directory. The location of search collection data can also be configured on a per-collection basis using the Watson Explorer Engine administration tool.</p> <p>This option should only be set when Watson Explorer Engine is not running.</p> <p><b>Important:</b> If you specify the <b>search-collections-dir</b> option in your <code>vivisimo.conf</code> file, the directory that you specify must be writable by the user that Watson Explorer Engine is executing as, or Watson Explorer Engine will fail when it tries to create or access any of your search collections.</p>
source-test-results-file	VIVISIMO_SOURCE_TEST_RESULTS_FILE	<p>The <b>source-test-service</b> stores files in the Engine/data/source-test directory. To change the location of the source test results file: Create a new local directory whose only purpose is to contain the source test results file. Give the webserver user appropriate permissions. Set the webserver environment variable to point at a results file in this directory. For an Apache webserver, the configuration would look like the following: <code>SetEnv VIVISIMO_SOURCE_TEST_RESULTS_FILE /path/to/new/directory/source-test-results.xml</code>. Restart the webserver to pick up the new configuration. Confirm that the new results file is created when the next source test is run.</p>



Table 1. Options and Equivalent Environment Variables (continued)

Watson Explorer Engine Option	Environment Variable	Description
system-reporting-dir	VIVISIMO_SYSTEM_REPORTING_DIR	The full or relative pathname of the directory where database files for system based reporting information will be stored. These files contain information about system internals, not query-based, application reporting data (which is stored in the location identified by the <b>reporting-dir</b> option). If this option is set to a directory that is not relative to the Watson Explorer Engine installation directory, the user or group under which your web server and the Watson Explorer Engine administration tool are running must be able to read from and write to this directory. By default, system reporting files are stored in the data/system-reporting directory of your installation directory.
tmp-dir	VIVISIMO_TMP_DIR	The full or relative pathname of the directory in which Watson Explorer Engine creates and uses temporary files. If this option is set to a directory that is not relative to the Watson Explorer Engine installation directory, the user or group under which your web server and the Watson Explorer Engine administration tool are running must be able to read from and write to this directory. The default value of this option is the tmp subdirectory of the directory in which Watson Explorer Engine is installed.

Table 1. Options and Equivalent Environment Variables (continued)

Watson Explorer Engine Option	Environment Variable	Description
users-dir	VIVISIMO_USERS_DIR	The full or relative pathname of the directory in which all information about Watson Explorer Engine users is stored. This will create multiple users files and requires a base file to be specified using the users-dir-file option. IBM recommends that the users-file option is used over this one.
users-dir-file	VIVISIMO_USERS_DIR_FILE	A file which contains the base Watson Explorer Engine users information. This must be specified if the users-dir option is specified. IBM recommends that the users-file option is used over this one.
users-file	VIVISIMO_USERS_FILE	The full or relative path of the XML file containing Watson Explorer Engine users information. When specified, no users directory will be created, and a single file will contain all Watson Explorer Engine user information.
vivisimo-conf	VIVISIMO_CONF_FILE	The name of the file containing global properties for this Watson Explorer Engine installation. By default the filename is vivisimo.conf. This option should only be specified as an environment variable.

## Configuring Zookeeper

After installing the Zookeeper module, you need to configure it.

**Note:** You must be logged in as an Administrator to run configuration tool. To do so, log in as the root user in Linux, or as a user with Administrator privileges on Microsoft Windows.

To run the configuration tool:

Linux	<p>Navigate to the installation location for the Zookeeper module (/opt/ibm/WEX/Zookeeper by default)</p> <p>Run ./zookeeper-config</p>
-------	---

Windows	Navigate to the installation location for the Zookeeper module (\Program Files\IBM\WEX\Zookeeper by default)  From a command line, run zookeeper-config.exe
---------	---

The following questions will display. Hit **Return** to accept the defaults.

- Which language do you want to use for the configuration tool? The current language is ['en']. Other available languages are: ['en']
- Which port do you want ZooKeeper to listen on for clients? The port is currently [2181]:  
The number of the port on which Zookeeper will listen for incoming connections.

**Note:** If you are installing Application Builder and modified the port on which Application Builder expects to find a ZooKeeper instance on the current host, you must specify the same port for the current host in this field.

- Specify the directory to store ZooKeeper's data in. The current directory is [/opt/ibm/WEX/ZooKeeperData]:  
The full path-name of the directory in which ZooKeeper stores the configuration data that it is managing.
- Specify the ZooKeeper ID. It is currently [1]:  
The numeric identifier for this ZooKeeper instance.
- Specify the servers and ports ZooKeeper's cluster will use. The current list is [server.1=localhost:16777:16888]:  
A semicolon-delimited list of ZooKeeper instances that are part of an ensemble. Each ZooKeeper instance is of the form server.ID=hostname:leader-port:election-port. The leader-port is the port that followers use to connect to an ensemble leader, and the election-port is the port that all members of an ensemble use when electing the ensemble leader.
- Provide any additional zookeeper configuration as a semicolon-separated list. The configuration is currently:  
[initLimit=1000;syncLimit=1000;tickTime=2000;autopurge.snapRetainCount=3; autopurge.purgeInterval=1]:  
A command-delimited list of configuration options that will be set in the configuration for the ZooKeeper instance you are installing.
- Do you want to install the service for ZooKeeper? [y/N]:  
The Zookeeper service will automatically start Zookeeper whenever you restart a system on which it is installed. Installing the service is recommended.  
**Note:** You should integrate Zookeeper into your system's startup and shutdown processes. See Integrating Other Watson Explorer Modules Into the Startup/Shutdown Process.

**Note:** Verify that the port that Zookeeper uses is not blocked by a network firewall on any system where Zookeeper is installed. This is the most common cause of any Watson Explorer module being unable to contact Zookeeper after you have started Zookeeper.

## Configuring Application Builder

After installing the Application Builder module, you need to configure it.

**Note:** You must be logged in as an Administrator to run configuration tools. To do so, log in as the root user in Linux, or as a user with Administrator privileges on Microsoft Windows.

To run the configuration tool:

Linux	Navigate to the installation location for the Application Builder module (/opt/ibm/WEX/Appbuilder by default)  Run ./appbuilder-config
Windows	Navigate to the installation location for the Application Builder module (\Program Files\IBM\WEX\Appbuilder by default)  From a command line, run appbuilder-config.exe

The following questions will display. Hit **Return** to accept the defaults.

- Which language do you want to use for the configuration tool? The current language is [en']. Other available languages are: [en']
- Which instance of Knowledge Center should Application Builder use for help links? It is currently using [http://www.ibm.com/support/knowledgecenter/SS8NLW\_11.0.2]:
- Which port do you want Application Builder to listen on for clients? The port is currently [8080]:  
The number of the port on which Application Builder will listen for incoming connections. If you change the default port, do not use 8081 or 9090, which are the default ports for Results Module and Knowledge Center.
- What do you want to name the Application Builder application in the configuration file for WebSphere Application Server Liberty profile? The name is currently [AppBuilder]:
- Which ZooKeeper instance or instances do you want to use with Application Builder? The currently used instance or instances are [localhost:2181]:  
A comma-separated list of host:port values that identify the hosts on which ZooKeeper has been installed for use by Application Builder and the port on which ZooKeeper is listening on each host.

**Note:** If you modified the ports on which your ZooKeeper instances will be listening when you configured Zookeeper, you must identify those ports in the host:port entries that you enter for this field.

- Which namespace do you want to use for this instance of Application Builder? The currently used namespace is [production]:  
The namespace that will be associated with Application Builder on any ZooKeeper installations that are configured for use by Application Builder to store configuration data.
- Do you want to install the service for Application Builder? [y/N]:  
The Application Builder service will automatically start Application Builder whenever you restart a system on which it is installed. Installing the service is recommended.

**Note:** You should integrate Application Builder into your system's startup and shutdown processes. See Integrating Other Watson Explorer Modules Into the Startup/Shutdown Process.

**Note:** Verify that the port that Application Builder uses is not blocked by a network firewall on any system where Application Builder is installed. This is the most common cause of any Watson Explorer module being unable to contact Application Builder after you have started Application Builder.

The Application Builder Module is an application powered by a IBM WebSphere Liberty Profile (WLP) server.

- You can adjust the configuration of your Application Builder WLP server after the fact by editing the `server.xml` file in the `AppBuilder/wlp/usr/servers/AppBuilder` directory of your installation. The authentication used by Application Builder is also configured in the `server.xml` file. See [Configuring Application Builder after Installation](#) for more information.
- For useful WLP commands, see [Useful WLP Server Commands for Application Builder](#).

### **Configuring Application Builder after Installation:**

Before starting Application Builder, you might want to modify the default configuration data to adjust options, such as the virtual directory or the port on which Application Builder listens. You can adjust the configuration of the IBM WebSphere® Application Server Liberty profile for Application Builder by editing the `server.xml` file in the `AppBuilder/wlp/usr/servers/AppBuilder` directory of your Watson Explorer installation.

**Important:** The authentication that is used by Application Builder is also configured in the `server.xml` file. However, the authentication can be configured only after the user entity and administrative users have been configured in the Application Builder administrator tool. For more information, see [Configuring the User Registry for Application Builder with WebSphere Application Server Liberty profile](#), which must be performed by using the following default login credentials

- User name: `data-explorer-admin`
- Password `TH1nk1710`

You can adjust the `server.xml` configuration in many ways to meet your site requirements. For example, you can change the port that is used to access Application Builder by changing the value of the `httpPort` attribute to specify a port other than the default value of 8080. You can configure other aspects of the WebSphere Application Server Liberty profile, including https-only configuration, by editing other values in this file.

For more information about WebSphere Application Server Liberty profile, see [Liberty profile overview](#) and [Administering the Liberty profile in the documentation for WebSphere Application Server](#).

### **Useful WLP Server Commands for Application Builder:**

The following additional WLP server commands may be useful:

- To stop the Application Builder WLP Server on Linux, run the `service appbuilder-service stop` command from your installation directory. On Windows, run `StopAppBuilder.bat` from your installation directory.

**Note:** On Microsoft Windows systems, the Application Builder service must be stopped before the associated ZooKeeper service is stopped. If the associated ZooKeeper service is stopped first, the Application Builder service will hang when an attempt to stop the service is made.

- To check the status of the Application Builder WLP Server, run the `service appbuilder-service status` command.

## Configuring Results Module

After installing the Results Module module, you need to configure it.

**Note:** You must be logged in as an Administrator to run configuration tools. To do so, log in as the root user in Linux, or as a user with Administrator privileges on Microsoft Windows.

To run the configuration tool:

Linux	Navigate to the installation location for the Results Module module (/opt/ibm/WEX/Resultsmodule by default)  Run ./resultsmodule-config
Windows	Navigate to the installation location for the Results Module module (\Program Files\IBM\WEX\Resultsmodule by default)  From a command line, run resultsmodule-config.exe

The following questions will display. Hit **Return** to accept the defaults.

- Which language do you want to use for the configuration tool? The current language is ['en']. Other available languages are: ['en']:
- Which port do you want Results Module to listen on? The port is currently [8081]:  
The number of the port on which the Results Module is listening for incoming connections. If you change the default port, do not use 8080 or 9090, which are the default ports for Application Builder and Knowledge Center.
- What do you want to name the Results Module application in the configuration file for WebSphere Application Server Liberty profile? The name is currently [ResultsModule]:
- Do you want to install the service for Results Module? [y/N]:  
The Results Module service will automatically start Results Module whenever you restart a system on which it is installed. Installing the service is recommended.

**Note:** You should integrate Results Module into your system's startup and shutdown processes. See Integrating Other Watson Explorer Modules Into the Startup/Shutdown Process.

**Note:** Verify that the port that Results Module uses is not blocked by a network firewall on any system where Results Module is installed. This is the most common cause of any Watson Explorer module being unable to contact Results Module after you have started Results Module.

**Note:** The Results Module is an application powered by a IBM WebSphere Liberty Profile (WLP) server.

- You can adjust the configuration of your Results Module WLP server after the fact by editing the server.xml file in the ResultsModule/wlp/usr/servers/ResultsModule directory of your installation. See Configuring the Results Module.
- For useful WLP commands, see Useful WLP Server Commands for the Results Module.

**Adjusting Configuration of the Results Module:** You can adjust the configuration of your Results Module WLP server by editing the `server.xml` file in the `ResultsModule/wlp/usr/servers/ResultsModule` directory of your installation. The configuration information that is displayed depends on any changes that you made on the Results Module configuration screen during the installation process, but will look something like:

```
<server description="ResultModule">

    <!-- Enable features -->
    <featureManager>
    <feature>jsp-2.2</feature>
    <feature>servlet-3.0</feature>
    </featureManager>

    <httpEndpoint id="defaultHttpEndpoint"
    host="*"
    httpPort="8081"
    httpsPort="9088" />

    <applicationMonitor updateTrigger="disabled"/>
</server>
```

You can adjust this configuration in many different ways to meet your site requirements. For example, you can change the port used to access the Results Module by changing the `httpPort="8081"` line to specify another port. You can configure other aspects of the WLP server, including https-only configuration, by editing other values in this file.

You can find detailed information about configuring the WLP server in the online WebSphere 8.5 documentation.

#### **Useful WLP Server Commands for the Results Module:**

The following additional WLP server commands may be useful:

- To stop the Results Module use `ResultsModule/wlp/bin/server stop ResultsModule` command from your installation directory.
- To check the status of the Results Module, use the `ResultsModule/wlp/bin/server status ResultsModule` command from your installation directory.

#### **Configuring BigIndex**

BigIndex provides a Java API for creating, configuring, and deploying information navigation applications in enterprise environments. The BigIndex API includes sophisticated functions that simplify or eliminate traditionally manual enterprise application administration tasks such as load balancing, mirroring, and replication.

The BigIndex API Developers Guide PDF can be downloaded from [Using the BigIndex API for Watson Explorer](#).

#### **Configuring Knowledge Center**

After installing the Knowledge Center module, you need to configure it.

**Note:** You must be logged in as an Administrator to run configuration tools. To do so, log in as the root user in Linux, or as a user with Administrator privileges on Microsoft Windows.

To run the configuration tool:

Linux	Navigate to the installation location for the Knowledge Center module (/opt/ibm/WEX/Knowledgecenter by default)  Run ./knowledgecenter-config
Windows	Navigate to the installation location for the Knowledge Center module (\Program Files\IBM\WEX\Knowledgecenter by default)  From a command line, run knowledgecenter-config.exe

The following questions will display. Hit **Return** to accept the defaults.

- Which language do you want to use for the configuration tool? The current language is ['en']. Other available languages are: ['en']:
- Which virtual directory do you want Knowledge Center to use? It is currently [/kc]:  
This is the virtual directory where online help for Watson Explorer will be located.
- Which port do you want Knowledge Center to listen on? The port is currently [9090]:  
This is the number of the port on which the Knowledge Center is listening for incoming connections. If you change the default port, do not use 8080 or 8081, which are the default ports for Application Builder and Results Module.
- Do you want to install the service for Knowledge Center? [y/N]:  
The Knowledge Center service will automatically start Knowledge Center whenever you restart a system on which it is installed. Installing the service is recommended.

**Note:** You should integrate Knowledge Center into your system's startup and shutdown processes. See Integrating Other Watson Explorer Modules Into the Startup/Shutdown Process.

**Note:** Verify that the port that Knowledge Center uses is not blocked by a network firewall on any system where Knowledge Center is installed. This is the most common cause of any Watson Explorer module being unable to contact Knowledge Center after you have started Knowledge Center.

## Starting Watson Explorer Services

The last step of the Watson Explorer installation is starting each Service. Startup scripts are provided to start services on Linux systems, and Services are provided to start Watson Explorer modules on Microsoft Windows systems. These scripts and services are only installed if you installed Watson Explorer in administrator mode.

**Note:** Starting Services requires administrative privileges.

**Important:** When starting the Services always:

- Start ZooKeeper first (Application Builder relies on ZooKeeper for configuration data).
- Start all other services except Application Builder.
- Start Application Builder last.

If you need to stop the Services in the future, stop Application Builder first, and ZooKeeper last.



Module	Service Name	Windows Display Name
Zookeeper	zookeeper-service	IBM Watson Explorer Zookeeper
Results Module	resultsmodule-service	IBM Watson Explorer Results Module
Knowledge Center	knowledgecenter-service	IBM Watson Explorer Knowledge Center
Application Builder	appbuilder-service	IBM Watson Explorer Application Builder

Note: If you are running RHEL 7, the service names are "nameofservice.service".

To start the Services:

Linux	<p>Start a terminal session (xterm, gnome-terminal, or other terminal application) and use the su or sudo -s command to become the root user.</p> <p>Execute service nameofmodule-service start for each service.</p> <p>Additional useful commands:</p> <ul style="list-style-type: none"> <li>• Stop service: service nameofmodule-service stop</li> <li>• Restart service: service nameofmodule-service restart</li> <li>• Run status of service: service nameofmodule-service status</li> </ul>
Windows	<p>Open the <b>Services</b> window (<b>Control Panel &gt; System and Security &gt; Administrative Tools</b>. Double-click on <b>Services</b>.)</p> <p>Choose the Service and click the <b>Start Service</b> button.</p>

## Accessing Watson Explorer Modules

After installing a Watson Explorer module, configuring it, and starting the service, you can now access the module.

By default, the Knowledge Center is available at [http://www.ibm.com/support/knowledgecenter/SS8NLW\\_11.0.2/watsonexplorer\\_11.0.2.html](http://www.ibm.com/support/knowledgecenter/SS8NLW_11.0.2/watsonexplorer_11.0.2.html).

To point to the local instance of Knowledge Center (if installed), modify the de\_help.conf file in the install directories of Watson Explorer Engine and Application Builder to point to:

<http://localhost:9090/kc>

(9090 is the default port for the Knowledge Center.) You can also make this change for Application Builder when running the appbuilder.config tool, at the time you specify which instance of Knowledge Center Application Builder should use for help links.

## Accessing Watson Explorer Engine

After installing Watson Explorer Engine, generating a new encryption key, running the micro-test utility, and configuring the system (production) or embedded (testing) web server, you can now access the Watson Explorer Engine administration tool from a web browser.

- If you configured Watson Explorer Engine for production by integrating it with a system web server, the Watson Explorer Engine administration tool should be available at the following URL, where <hostname> is the name or IP address of the system on which your web server is running and <port> is the port used for this module.

`http://<hostname>:<port>/vivisimo/cgi-bin/admin`

- If you configured Watson Explorer Engine for testing purposes using the embedded web server, see *Configuring Watson Explorer Engine's Embedded Web Server* for instructions for configuring the embedded webserver for use with the Watson Explorer Engine administration tool.

Once the Watson Explorer Engine login prompt displays in your browser, use the default login and password to access the Watson Explorer Engine administration tool.

- Default login: `data-explorer-admin`
- Default password: `TH1nk1710` (T-H-one-n-k-one-seven-one-zero)

Watson Explorer Engine is now ready to use.

**Note:** After accessing the Watson Explorer Engine administration tool for the first time using the default login and password, you should immediately change the password for this login. To change the password for a Watson Explorer Engine account, use the **Management > Users** menu to access and modify the current settings for that user account.

At this point, you may want to create additional Watson Explorer Engine users for development purposes using the **Management > Users** menu. This enables other developers to access the Watson Explorer Engine administration tool while keeping your site's administrative login and password private.

See *Command-Line Watson Explorer Engine Administration Commands* for general information about integrating Watson Explorer Engine into your system startup and shutdown processes, and for detailed information about the Watson Explorer Engine commands that are provided for use in these cases.

Before creating any Watson Explorer Engine search applications or search collections, you may want to take advantage of configuration options that enable you to store Watson Explorer Engine search collection data on other filesystems. For example, if the system on which Watson Explorer Engine is installed also has access to large, high-speed RAID storage devices, you may want to install Watson Explorer Engine in its default location but use Watson Explorer Engine's **search-collections-dir** configuration variable to specify that search collections are created and stored on the high-speed, higher-capacity storage devices. See *Configuring Global Options for Watson Explorer Engine* for more information.

If you are running automatic anti-virus or other system scanning software, see *Integrating Watson Explorer with Anti-Virus Software* for information on excluding specific Watson Explorer Engine directories from the scanning process.

## Accessing Application Builder

After starting the ZooKeeper and the Application Builder services, you can access the Application Builder administration tool at the following URL:

`<hostname>:<port>/AppBuilder/admin`

Where `<hostname>` is the host name or IP or the Application Builder server, and `<port>` is the port used for this module (8080 by default). To access the Application Builder using the Bootstrap user, log in using the username **data-explorer-admin** and the password **TH1nk1710**.

The Application Builder is now installed and operational. Consult the Developing Applications section of the Application Builder documentation for additional configuration details.

If you are running automatic anti-virus or other system scanning software, see Integrating Watson Explorer with Anti-Virus Software for information on excluding specific Application Builder directories from the scanning process.

**Note:** There may be a significant wait period (greater than 1 minute) when accessing the Application Builder or its applications for the first time after the WLP has been started. This is part of the WLP initialization process and only occurs the first time it is accessed after startup.

### Accessing the Results Module

After starting the Results Module service, you can access the Results Module interface at a URL like the following:

```
<hostname>:<port>/ResultsModule
```

Where <hostname> is the host name or IP of the Results Module server, and <port> is the port used for this module (8081 by default). To access the Results Module using the Bootstrap user, use the username **bootstrap** and the password **changeme**.

The Results Module is now installed and operational. For details about configuring the Results Module, see Creating and Configuring Watson Explorer Results Module.

If you are running automatic anti-virus or other system scanning software, see Integrating Watson Explorer with Anti-Virus Software for information on excluding specific Results Module directories from the scanning process.

**Note:** There may be a significant wait period (greater than 1 minute) when accessing the Results Module for the first time after the WLP has been started. This is part of the WLP initialization process and only occurs the first time it is accessed after startup.

## Uninstalling Watson Explorer

To uninstall Watson Explorer version 11.0.0.1 modules or later, follow the instructions below.

To uninstall Watson Explorer 11.0.0.1 or later:

Linux	Start a terminal session (xterm, gnome-terminal, or other terminal application) and use the su or sudo -s command to become the root user.  Enter <code>rpm -e wex-nameofmodule</code> <b>Note:</b> Do not include the file extension (.rpm) or the version number with the name of the module.
Windows	Go to <b>Control Panel &gt; Programs &gt; Programs and Features</b> to view your list of programs.  Choose the Watson Explorer module from the list.  Click the <b>Uninstall</b> button. <b>Note:</b> If you right-click on the module .msi file in Windows Explorer, choose <b>Remove</b> from the menu to uninstall the module.

Default Watson Explorer installation directory:

- Linux: /opt/ibm/WEX/nameofmodule
- Windows: \Program Files\IBM\WEX\nameofmodule

**Note:** The list of module installation file names can be found in Installing Watson Explorer.

**Note:** If you need to uninstall Watson Explorer 11.0.0 or earlier, see Uninstalling Watson Explorer and Installation Manager.

## Watson Explorer Upgrade Options

A new installer was introduced with Watson Explorer version 11.0.0.1. Version 11.0.2 does not use Installation Manager to install Watson Explorer.

You will still need to use Installation Manager to uninstall previous versions. See Uninstalling Watson Explorer and Installation Manager for uninstall instructions.

IBM makes every possible effort to ensure that software upgrades are as easy as possible. However, seamless upgrades are not always possible when product names, installation requirements, and installation locations change.

**Note:** If you need to upgrade from a version of Watson Explorer earlier than 11.0.0 see Upgrading from Previous Releases.

If you have not yet installed any version of Watson Explorer: Do not install 11.0.0 first. Instead, follow these instructions (starting with Installing Watson Explorer) to install Watson Explorer 11.0.2 or later. See the Watson Explorer Version 11.0.2 system requirements to confirm you meet the requirements for:

- Operating Systems
- Hardware
- Prerequisite software
- Supported software
- Client Requirements

For a description of the modules included in Watson Explorer, see Overview of Watson Explorer Foundational Components.

There are two options for upgrading to 11.0.2 or later:

- **Install 11.0.2 (or later) in the same directory as the earlier version on the same machine:**

Uninstall Watson Explorer 11.0.0 or earlier using Installation Manager, then install 11.0.2 (or later) in the same directory. In this scenario your search collection data and other modified files can remain in the same location.

- **Install 11.0.2 (or later) in a different directory than the earlier version on the same machine:**

Uninstall Watson Explorer 11.0.0 using Installation Manager, then install 11.0.2 in a different directory on the same machine.

If you choose to install in a different directory, you will need to copy to the new installation directory:

- Search collection data

- Any modified files (files in your 11.0.0 (or earlier) installation that have been modified since they were installed by the Installation Manager)
- All manually created files from your old installation directory
- Any Application Builder and Results Module data. See Backing Up Application Builder and Upgrading from IO Pro for instructions.

Search collection data is stored in the directory listed below by default. (Changing the location of this directory is common. For information on how to change the location of a search collection directory, see Relocating Collections.):

<install\_dir>/Engine/data/search-collections/

Most modified and manually created files will be stored in (but not limited to) the following directories:

<install\_dir>/Engine/data/axl-pools/\*

<install\_dir>/Engine/data/dictionaries/\*

<install\_dir>/Engine/data/fonts/\*

<install\_dir>/Engine/data/key-match/\*

<install\_dir>/Engine/data/pears/\*

<install\_dir>/Engine/data/reporting/\*

<install\_dir>/Engine/data/repository-supplements/\*

<install\_dir>/Engine/data/sessions/\*

<install\_dir>/Engine/data/static/\*

<install\_dir>/Engine/data/su-tokens/\*

These files may have been modified also:

<install\_dir>/Engine/data/repository-internal.package

<install\_dir>/Engine/data/ISYS-license

**Note:** <install\_dir> = location of your Watson Explorer installation, by default:

- Linux: /opt/ibm/WEX/
- Windows: \Program Files\IBM\WEX\

**Note:** For uninstall instructions for Watson Explorer 11.0.0 or earlier, see the IBM Knowledge Center associated with that version.

## Useful WLP Server Commands for the Results Module

The following additional WLP server commands may be useful:

- To stop the Results Module use ResultsModule/wlp/bin/server stop ResultsModule command from your installation directory.
- To check the status of the Results Module, use the ResultsModule/wlp/bin/server status ResultsModule command from your installation directory.

## Useful WLP Server Commands for Application Builder

The following additional WLP server commands may be useful:

- To stop the Application Builder WLP Server on Linux, run the service `appbuilder-service stop` command from your installation directory. On Windows, run `StopAppBuilder.bat` from your installation directory.

**Note:** On Microsoft Windows systems, the Application Builder service must be stopped before the associated ZooKeeper service is stopped. If the associated ZooKeeper service is stopped first, the Application Builder service will hang when an attempt to stop the service is made.

- To check the status of the Application Builder WLP Server, run the service `appbuilder-service status` command.

## Enabling Javascript in Internet Explorer 7 and Later

### About this task

As noted in the system requirements, a browser must support Javascript in order to use the Watson Explorer Engine administration tool. To enable Javascript in IE 7.0 and later, do the following:

### Procedure

1. Select the Internet Explorer **Tools** menu, click **Internet Options**, and click the **Security** tab.
2. Select the **Internet** zone.
3. Click **Custom level...** The Security Settings - Internet Zone dialog displays.
4. In the Security Settings - Internet Zone dialog, scroll down to the **Scripting** section and click **Enable** under the **Active scripting** entry.
5. Click **OK** to accept your changes and close the Security Settings - Internet Zone dialog.
6. Click **OK** to close the Internet Options dialog.

### Results

If you had tried to visit a page that requires Javascript before enabling support for active scripting, click the **Refresh** button in Internet Explorer to revisit the page and execute the Javascript that it contains.

## Modifying the Location of the Watson Explorer Knowledge Center URL

### About this task

To access the Watson Explorer Knowledge Center through the online help provided in Watson Explorer and other Watson Explorer modules including Watson Explorer Application Builder and Watson Explorer Results Module, you will need to modify the URL in the `de_help.conf` file in your Watson Explorer installation. The URL contained in this configurable file can point to the online Knowledge Center or to a local installation of the Knowledge Center.

To modify the Knowledge Center URL in the `de_help.conf` file, do the following:

## Procedure

1. Navigate to the top level of your Watson Explorer installation directory and open the `de_help.conf` file in a text editor. The following XML node displays:

```
<?xml version="1.0" encoding="UTF-8" ?>
<options>
  <option name="online-help-url">localhost:9999/kc/SS8NLW_11.0.2</option>
</options>
```
2. Modify the URL contained in the XML element option name to point to a local installation of the Knowledge Center or to an externally available Knowledge Center. Use either of the following explicit URLs:
  - **For the external IBM Watson Explorer 11.0.2 Knowledge Center** - [http://www.ibm.com/support/knowledgecenter/SS8NLW\\_11.0.2](http://www.ibm.com/support/knowledgecenter/SS8NLW_11.0.2).
  - **For a locally installed Knowledge Center** - [http://localhost:9090/kc/SS8NLW\\_11.0.2](http://localhost:9090/kc/SS8NLW_11.0.2) (where `localhost:9999` is replaced with the **hostname** and **port** number of the server where your local Knowledge Center is installed)

**Example:** Replace `<option name="online-help-url">localhost:9999/kc/SS8NLW_11.0.2</option>` with `<option name="online-help-url">http://www.ibm.com/support/knowledgecenter/SS8NLW_11.0.2</option>` to have your Watson Explorer Engine point to the externally available IBM Knowledge Center.

**Note:** When modifying or setting a new URL, omit trailing backslashes in the URL path. Trailing backslashes can cause errors.
3. After modifying or setting a new URL, save your changes to the `de_help.conf` file.
4. If you have the Watson Explorer Application Builder module installed, you will need to restart all the Watson Explorer Application Builder services to complete this procedure.

## Results

You have modified the Knowledge Center URL that is associated with your Watson Explorer installation. If you are using a local Knowledge Center installation make sure that the Knowledge Center service has been started using the methods described in the Watson Explorer Installation Guide.

## Troubleshooting an Installation

The Watson Explorer software installation mechanism is designed to make it easy to install any Watson Explorer module(s) successfully on your systems. However, depending upon how your web server, firewalls, and other network security mechanisms are configured, you may encounter problems when first getting started using the module(s) that you have installed. The following are some common errors that you may see, but which can easily be resolved by adjusting the network configuration or file protections on the system where you have installed the Watson Explorer module(s):

- **Corrupt data error when trying to login to the Watson Explorer Engine administration tool immediately after installation.**

Either TCP port 80 (HTTP), TCP port 443 (HTTPS), or both are blocked. Unblock TCP port 80 or 443.

- **Could not create lock file://data/admin.lock.**

The permissions on the `vivisimo.conf` file are set such that the web server does not have access. Change the permissions of the `vivisimo.conf` file (located in the Watson Explorer Engine installation directory).

- **HTTP 404 Error after installation**

One or more of the following issues may be the cause:

- The **Alias** (Apache) or **Virtual Directory** (IIS) is not configured properly, or doesn't point to the correct location
- The URL in the browser is incorrect
- On Linux systems: AppArmor or SELinux is set to restrictive mode and is blocking the execution of the Engine binaries. Set these to permissive or disabled mode.

- **HTTP 500 Error when installing in Windows under IIS**

CGI executables haven't been configured to run correctly, follow the instructions for your version of Windows to allow CGI executables to run.

- **HTTP 403 Error**

The user that is logged in, or is running the web service, has insufficient privileges. Check the permissions on the account.

## Getting Help

If you have questions or encounter problems with this installation procedure or any module of the Watson Explorer software, please file a service request using the IBM Support Portal. (Service requests are also sometimes referred to as PMRs - Problem Management Records.)

You must sign in to the IBM Support Portal to submit or access service requests. If you do not already have a login for the Support Portal, click **Register Now** and specify your email address as your login during the registration process. After registering, you must contact the Site Technical Contact (STC) for your company's account and request that your account be granted the ability to create service requests. Once your account has been authorized, you can click **Support and Download** from the top navigator menu, and select **Open a Technical Service Request** to enter your request.

**Note:** For more detailed information about submitting a service request, see the IBM Service Request Quick Start page.

To take advantage of any support that is available for your national language and locale, select your region and country from the Directory of worldwide contacts page.

---

## Upgrading from Previous Releases

For information about upgrading from earlier versions of Watson Explorer foundational components (Watson Explorer Engine, Application Builder, and Results Module) see Upgrading to Watson Explorer Foundational Components Version 11.0.2.

For information about upgrading to Watson Explorer version 11.0.0.1 or later fix packs, see "Upgrading to Watson Explorer Fix Pack versions."

## Upgrading to Watson Explorer Fix Pack versions

### About this task

The following sections describe how to upgrade to Watson Explorer version 11.0.0.1 or later fix packs.



Watson Explorer Upgrade Options

Installing Watson Explorer Version 11.0.0.1 or Later

Configuring Watson Explorer Modules

Starting Watson Explorer Services

Accessing Watson Explorer Modules

Uninstalling Watson Explorer

## **Upgrading from Previous Watson Explorer Versions**

For information about upgrading from earlier versions of Watson Explorer foundational components (Watson Explorer Engine, Application Builder, and Results Module) see [Upgrading to Watson Explorer Foundational Components Version 11.0.2](#).

## **Upgrading from Velocity to Watson Explorer Engine**

For information about upgrading to Watson Explorer Engine from Velocity, see [Upgrading to Watson Explorer Foundational Components Version 11.0.2](#).

## **Upgrading from IO Pro to Watson Explorer Results Module**

For information about upgrading from IO Pro to Watson Explorer Results Module, see [Upgrading from IO Pro](#).



---

## Notices and trademarks

This information was developed for products and services offered in the U.S.A. This material may be available from IBM in other languages. However, you may be required to own a copy of the product or product version in that language in order to access it.

### Notices

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing  
IBM Corporation  
North Castle Drive  
Armonk, NY 10504-1785 U.S.A.

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing  
Legal and Intellectual Property Law  
IBM Japan Ltd.  
1623-14, Shimotsuruma, Yamato-shi  
Kanagawa 242-8502 Japan

**The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:**

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation  
J46A/G4  
555 Bailey Avenue  
San Jose, CA 95141-1003 U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. \_enter the year or years\_. All rights reserved.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

## **Trademarks**

IBM, the IBM logo, and [ibm.com](http://ibm.com) are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at [www.ibm.com/legal/copytrade.shtml](http://www.ibm.com/legal/copytrade.shtml).

The following terms are trademarks or registered trademarks of other companies:

Adobe is a registered trademark of Adobe Systems Incorporated in the United States, and/or other countries.

Intel and Itanium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows and Windows NT are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

The United States Postal Service owns the following trademarks: CASS, CASS Certified, DPV, LACS<sup>Link</sup>, ZIP, ZIP + 4, ZIP Code, Post Office, Postal Service, USPS and United States Postal Service. IBM Corporation is a non-exclusive DPV and LACS<sup>Link</sup> licensee of the United States Postal Service.

Other company, product or service names may be trademarks or service marks of others.



---

## Providing comments on the documentation

You can provide comments to IBM about this information or other documentation.

### **About this task**

Your feedback helps IBM to provide quality information. You can use any of the following methods to provide comments:

### **Procedure**

- Send your comments by using the online readers' comment form at [www.ibm.com/software/awdtools/rcf/](http://www.ibm.com/software/awdtools/rcf/).
- Send your comments by e-mail to [comments@us.ibm.com](mailto:comments@us.ibm.com). Include the name of the product, the version number of the product, and the name and part number of the information (if applicable). If you are commenting on specific text, include the location of the text (for example, a title, a table number, or a page number).









Printed in USA

GC19-4389-02

