



Find

Software Version: 11.6

Administration Guide

Document Release Date: February 2018

Software Release Date: February 2018

Legal notices

Warranty

The only warranties for Seattle SpinCo, Inc. and its subsidiaries ("Seattle") products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. Seattle shall not be liable for technical or editorial errors or omissions contained herein. The information contained herein is subject to change without notice.

Restricted rights legend

Confidential computer software. Except as specifically indicated, valid license from Seattle required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

Copyright notice

© Copyright 2015-2018 EntIT Software LLC, a Micro Focus company

Trademark notices

Adobe™ is a trademark of Adobe Systems Incorporated.

Microsoft® and Windows® are U.S. registered trademarks of Microsoft Corporation.

UNIX® is a registered trademark of The Open Group.

Support

Visit the Micro Focus Software Support Online website at <https://softwaresupport.softwaregrp.com>.

This website provides contact information and details about the products, services, and support that Micro Focus offers.

Micro Focus online support provides customer self-solve capabilities. It provides a fast and efficient way to access interactive technical support tools needed to manage your business. As a valued support customer, you can benefit by using the support website to:

- Search for knowledge documents of interest
- Submit and track support cases and enhancement requests
- Access the Software Licenses and Downloads portal
- Download software patches
- Access product documentation
- Manage support contracts
- Look up Micro Focus support contacts
- Review information about available services
- Enter into discussions with other software customers
- Research and register for software training

Most of the support areas require you to register as a Passport user and sign in. Many also require a support contract.

You can register for a Software Passport through a link on the Software Support Online site.

To find more information about access levels, go to

<https://softwaresupport.softwaregrp.com/web/softwaresupport/access-levels>.

Contents

Part I: Getting Started	7
Chapter 1: Introduction	9
Chapter 2: Prerequisites	11
Supported Operating Systems and Browsers	11
Required IDOL Components	11
IDOL Content Data Index Requirements	11
IDOL Document Security	12
Optional IDOL Components	12
Third-Party Software	12
Find Home Directory	12
Chapter 3: Install Find	15
Available Formats	15
Interactive Installer	15
Zip Package	15
Install Find with the Interactive Installer (Recommended)	15
Install Find Manually Using the Zip Package	16
Use the Executable .war File to Run Find	16
Install Find as a Service on Linux	16
Install Find as a Windows Service	17
Upgrade Find	18
Revert to a Previous Version of Find	18
Chapter 4: Set Up Find	19
Run Initial Find Configuration	19
Configure Statistics Server	19
User Roles	20
Create IDOL Community Roles	20
Clustering	21
Part II: Configure Find	23
Chapter 5: Configure Find on the Settings Page	25
Configure IDOL View Server	25
Configure MMAP Integration	26
Configure Map Visualizations	26
Enable Polling for Saved Searches	27
Upload a Custom Logo	28
Chapter 6: Configure Find Using the Configuration File	29
Configure User-Friendly Names for Fields and Values	29
Configure Fields to Show or Hide in Document Previews	31
Configure Fields to Export	32

Configure the Order of Filters	33
Configure the Order of Parametric Fields	34
Configure Parametric Filters to Never Show or Always Show	35
Configure Find with Different IDOL Document Root Fields	36
Change or Add Location Field Names	36
Configure the Trending Visualizer	39
Configure the Date Field	39
Configure the Number of Field Values to Display	40
Configure the Number of Date Ranges to Use	40
Change the Maximum Number of Documents Used to Calculate a Topic Map	41
Change the Maximum Number of Documents to Use to Compare Searches	42
Enable or Disable the Filter Search Bar	42
Enable or Disable the Related Concepts Panel	43
Open a Shared Search as a New Query	44
Customize Error Messages	44
Exclude Databases from Queries	45
Configure Entity Search	45
Modify the Length of a Document Summary	47
Chapter 7: Configure Dashboards	48
Find Dashboards	48
Dashboard Configuration	49
Dashboard Grid Size	51
Widgets	52
Widget Configuration	52
Topic Map Widget	54
Results List Widget	55
Sunburst Widget	56
Trending Widget	57
Map Widget	59
Video Panel Widget	60
Static Content Widget	61
Static Image Widget	62
Current Time/Date Widget	62
Time Last Refreshed Widget	63
Dashboard Export Notes	64
Example Dashboard Configuration	64
Chapter 8: Configure Applications	72
Configure Application Links in Find	72
Chapter 9: Configure Templates	74
Configure Templates in Find	74
Create Template Files	75
Example Template	78
Chapter 10: Configure Find with System Properties	80
Configure the Log File Locations	80
Advanced Log Configuration	80

Tomcat Access Log Configuration	81
Configure IDOL Logs	81
Configure Cache Size	82
Configure Default User Roles	83
Configure Find to Use SSL	84
Configure Session Timeout	85
Rename the Session Cookie	85
Configure Find with a Reverse Proxy	86
Configure Find to Use the Reverse Proxy	86
Set the Reverse Proxy System Property	87
Configure the Reverse Proxy	87
Apache 2	87
SAML	88
NGINX	89
Internet Information Services (IIS)	90
Configure Find to Use a Single Pre-Authenticated User	90
 Send documentation feedback	 91

Part I: Getting Started

This section provides an introduction to Micro Focus Find, and describes how to install the application and perform the initial set up.

- [Introduction, on page 9](#)
- [Prerequisites, on page 11](#)
- [Install Find, on page 15](#)
- [Set Up Find , on page 19](#)

Chapter 1: Introduction

Micro Focus Find is an end-user search interface for IDOL.

Find supports the following functionality:

- **Advanced search.** Refine searches by database, date, or parametric values.
- **Document preview.** Find uses the IDOL View component to render near-native views of the source documents for your search results in a web browser. The Document Preview feature works with View server and documents in source repositories.
- **Query manipulation.** Find applies synonym and blacklist rules when you use the IDOL Query Manipulation Service as a back end for your search. You can use IDOL Data Admin to create and manage these rules.
- **Results manipulation.** Find applies pin-to-position promotions when you use the IDOL Query Manipulation Service as a back end for your search. You can use IDOL Data Admin to create and manage these promotions.
 - **Results augmentation.** Find applies spotlight promotions when you use the IDOL Query Manipulation Service as a back end for your search. You can use IDOL Data Admin to create and manage these promotions.
- **Document security.** You can set up document security so that only users with appropriate permissions can access the documents, or even see them in search results.
- **Visualizations.** You can view results as topic maps, sunburst charts, maps, or in table format, as well as in list format.
- **Save searches.** You can save searches and run them again later, save a result set as a snapshot, or compare two or more saved searches.

NOTE:

From the 11.0 release, Find includes Business Information for Human Intelligence (BIFI); visualizations, saved searches, and comparisons are available only to users with the `FindBI` role (see [User Roles, on page 20](#) for more information).

For more information on additional IDOL components that work with Find, see [Required IDOL Components, on page 11](#) and [Optional IDOL Components, on page 12](#).

Chapter 2: Prerequisites

This section describes the operating systems that Find supports, and the required and optional IDOL components and third-party software needed to run Find. It also specifies how to set up the Find home directory.

• Supported Operating Systems and Browsers	11
• Required IDOL Components	11
• Optional IDOL Components	12
• Third-Party Software	12
• Find Home Directory	12

Supported Operating Systems and Browsers

Find supports the following operating systems:

- Windows Server 2008, 2008 R2, 2012, or 2012 R2
- Ubuntu 14.04 LTS
- CentOS 6 or 7

Find supports the following web browsers:

- Chrome (latest version)
- Firefox (latest version)
- Internet Explorer 11
- Microsoft Edge

Required IDOL Components

Before you install Find, you must install and configure the following IDOL components:

- IDOL Community 11.3.0 (Find requires access to the ACI and service ports).

NOTE:

IDOL Community requires an IDOL Agentstore subcomponent. However, you do not need to configure the details of the IDOL Community Agentstore component in Find.

- IDOL Content 11.6 (Find requires access to the ACI and service ports).
- IDOL View Server 11.3.0 (Find requires access to the ACI and service ports).

If you want to use a Connector for viewing, Find also requires access to the ACI and service ports for the connector. See [Configure IDOL View Server, on page 25](#).

IDOL Content Data Index Requirements

In general, Find does not place restrictions on the data that you store in the IDOL Content component.

However, if you index your data in XML format, you must use <DOCUMENTS><DOCUMENT> as your top level fields, to ensure that Find treats the contents consistently.

IDOL Document Security

You can use IDOL Document Security in Find. In this case, Find requests security information from the IDOL Community component, and uses it to manage your user sessions. The security token contains information about the groups and permissions that the user has, which the IDOL Content component uses to control access to your documents.

For information about how to set up document security in IDOL, refer to the *IDOL Document Security Administration Guide*.

NOTE:

The IDOL Community security tokens have an expiration time. When the security token expires, the user session in Find ends. The user must log in to Find again to generate a new token.

Optional IDOL Components

The following IDOL components are optional:

- IDOL Query Manipulation Server (QMS) 11.6 (Find requires access to the ACI and service ports).

NOTE:

IDOL QMS sends queries to an IDOL Content component data index. The IDOL Content component that you use must be the same for both QMS and the Find configuration (see [Required IDOL Components, on the previous page](#)).

IDOL QMS also requires an IDOL Agentstore subcomponent. However, you do not need to configure the details of the IDOL QMS Agentstore component in Find.

- IDOL Statistics Server 11.3.0 (Find requires access to the ACI and service ports).
- IDOL Answer Server 11.6 (Find requires access to the ACI and service ports).

Third-Party Software

Find requires the following third-party software:

- Java 1.8

Find Home Directory

You must specify the Find home directory as a Java system property. Depending on how you are deploying the application, you can do this either in a startup script or within the appropriate application server configuration.

The home directory contains the following files:

- The Find configuration file (`config.json`). If this file is not found, Find generates a default configuration file when the application starts.
You can edit the configuration file either from the Settings page in the application (recommended), or manually in a text editor. If you manually edit the file, you must restart Find for any configuration changes to take effect.
- The Find customization folder. This directory contains additional configuration files for additional customizable features, such as dashboards and application links. See [Configure Dashboards, on page 48](#) and [Configure Applications, on page 72](#).
- The Find logs folder. This directory contains the application log files, which contain details of application usage and any errors.
- The Find data folder. This directory contains the local database for the saved searches.

If you need to contact support, include the Find home directory as a zip file when you send your query.

NOTE:

By default, the installer stores the Find home directory in the `ProgramData` folder, which might be hidden by default in Windows.

Chapter 3: Install Find

Find is available as an interactive installer.

Alternatively, you can install Find as an IDOL component by running the IDOL Server installer. For more information, see the *IDOL Server Getting Started Guide*.

NOTE:
To upgrade to Find 11.6 from an earlier version, you must delete your existing configuration file before you install the new version. Micro Focus also recommends that you back up your database for saved searches before you upgrade.
For more information, see [Upgrade Find, on page 18](#).

- [Available Formats](#)15
- [Install Find with the Interactive Installer \(Recommended\)](#)15
- [Install Find Manually Using the Zip Package](#) 16
- [Upgrade Find](#) 18

Available Formats

Find is available in two different formats:

- an interactive installer
- a zip package containing the components

You can also install Find as an IDOL component by running the IDOL Server installer. For more information, see the *IDOL Server Getting Started Guide*.

Interactive Installer

Micro Focus recommends that you install Find by using the interactive installer. This process creates all required directories, and sets up Find to run as a service. During installation, you specify the Find home directory and the paths for installation.

Zip Package

The Find zip package contains an executable Java `.war` file. Unlike traditional Java web applications, Find does not require a stand-alone application server (for example, Tomcat), because one is embedded within the `.war` file.

Install Find with the Interactive Installer (Recommended)

Use the following procedure to install Find on Windows or Linux by using the interactive installer.

To install Find

- Run the installer as an Administrator user, and then follow the instructions, supplying the correct information when prompted.

Find is installed at the location you specify. Find is available on your machine as a service named `MicroFocus-Find`, which you can start and stop from the Windows services manager.

NOTE:

By default, the installer stores the Find home directory in the `ProgramData` folder, which might be hidden by default in Windows.

Before you start Find, follow the steps in [Run Initial Find Configuration, on page 19](#), and ensure that you have set up the required user roles in IDOL Community. A user must be a member of at least the `FindUser` role to log into Find. See [User Roles, on page 20](#) for more information.

Install Find Manually Using the Zip Package

The following sections describe how to install Find by using the zip package.

Use the Executable .war File to Run Find

Use the following command to run Find:

```
java -Didol.find.home=[home directory] -Dserver.port=[port] -jar find.war -
uriEncoding utf-8
```

NOTE:

On Linux, Micro Focus recommends that you use `/opt/find` as the install directory and `/opt/find/home` as the home directory. On Windows, the home directory might be `C:\Micro Focus\Find`. In either case, ensure that the user running Find has read and write permissions for the home directory.

Install Find as a Service on Linux

Use the following procedure to install Find as a service on Linux.

1. Create a user and group to run Find. Micro Focus recommends that you use `find` as both the user name and the group name:

```
$ useradd find
```

2. Create a directory in which to install Find. Micro Focus recommends that you use `/opt/find` as the installation directory:

```
$ mkdir /opt/find
```

3. Create a directory to serve as the home directory for Find. Micro Focus recommends that you use `/opt/find/home` as the home directory:

```
$ mkdir /opt/find/home
```


4. Copy `find.war` to the installation directory:

```
$ cp find.war /opt/find/
```

5. Recursively change the ownership of the installation directory:

```
$ chown -R find:find /opt/find
```

6. Deploy the startup script.

System V

```
$ cp install/linux/sysv/find.sh /etc/init.d/find
```

```
$ chmod +x /etc/init.d/find
```

```
$ update-rc.d find defaults 99
```

Upstart

```
$ cp install/linux/upstart/find.conf /etc/init/
```

```
$ chmod +x /etc/init/find.conf
```

7. If you have used any non-default paths or settings during installation, edit the startup script to contain the corresponding values.
8. Start the application.

System V

```
$ sudo /etc/init.d/find start
```

Upstart

```
$ sudo service find start
```

Install Find as a Windows Service

To run Find as a Windows service, you must download a copy of the Windows service wrapper from <https://github.com/ft-idol/winsw>, and change the name of the `winsw.exe` executable file to `find.exe`.

Ensure that the Find home directory and port are correctly configured in the `find.xml` configuration file. You must also ensure that `find.exe`, `find.xml`, and `find.war` are all in the same directory.

Run the following command from the Windows command line to install Find as a service:

```
find.exe install
```

NOTE:

You can also use the following commands with `find.exe`:

- `uninstall`
- `start`
- `stop`
- `status`

For more information, see <https://github.com/ft-idol/winsw>.

Upgrade Find

To upgrade to Find 11.6 from an earlier version, you must delete your existing configuration file. The configuration files might not be compatible, and if you attempt to run Find with a configuration file from an earlier version, it might have unexpected results.

Micro Focus also recommends that you back up your database for saved searches before you upgrade (that is, copy the `data` directory in the home directory). During the upgrade process, Find migrates the database to the new version. If you need to revert to the earlier version of Find, you must restore the `data` directory and configuration file.

To upgrade Find

1. Back up the `data` directory in the Find home directory. This directory contains the database file that stores saved search data. The backup is required if you need to revert to the previous version of Find after upgrading.
2. Back up the `config.json` file in the Find home directory, to use as a reference.
3. Delete the `config.json` file from the home directory.
4. Run the new Find installer, or replace the `.war` file in the Find home directory. You must use the same home directory as the installed version of Find.
5. Follow the installation and setup instructions in [Install Find, on page 15](#).

The installer migrates your database to the latest version, and generates a new `config.json` file for the new

TIP:

You can use the backup copy of the `config.json` file as a reference to specify your configuration settings in the upgraded version.

Revert to a Previous Version of Find

The Find upgrade process modifies the database, and the configuration settings might not be compatible between different versions. If you need to revert to a previous version of Find, you must restore the configuration file and `data` directory to the earlier version.

CAUTION:

Any data that you save after you upgrade Find is lost when you revert to an earlier version.

To revert to a previous version of Find

1. Uninstall the existing version of Micro Focus Find Administration Guide.
2. Follow the instructions in [Install Find, on page 15](#) to install the previous version.
3. Restore the backup copies of `config.json` and the `data` directory that you made during the upgrade process to the Find home directory.

Chapter 4: Set Up Find

This section describes how to run the initial required Find configuration to get the application running and working with your back end IDOL components.

For more information about configuration and customization, see [Configure Find, on page 23](#)

• Run Initial Find Configuration	19
• Configure Statistics Server	19
• User Roles	20
• Clustering	21

Run Initial Find Configuration

Use the following procedure to configure Find.

1. Run Find once without a `config.json` file in your Find home directory to generate a default file.
2. Open the generated `config.json` file in a text editor. Find the generated default user name and password (`login.defaultLogin.username` and `login.defaultLogin.password`).
3. Go to `http://FindHost:FindPort`, where *FindHost* is the IP address or name of the machine on which Find is installed, and *FindPort* is the port that you have installed Find on (for example, 8080).
4. Log in with the default user name (this is normally set to `admin`) and password from the configuration file.
The Settings page opens.
5. Fill in the details of your backend IDOL servers. Click **Test Connection** for each server to confirm that the Find server can establish a connection to each server.
6. Click **Save Changes**.
Find is now ready for use.

NOTE:

To return to the Settings page at any time, click the cog in the top right corner of the application, and then click **Settings** in the list of options.

For more information about configuring Find, see [Configure Find, on page 23](#).

Configure Statistics Server

Find can optionally use Statistics Server to store statistics about user searches. For example, it tracks statistics about the times users open a source document or preview, and the times a user leaves a search without viewing any results.

By default, Find statistics are not enabled. You can enable statistics by configuring Statistics Server on the Find Settings page. See [Run Initial Find Configuration, above](#).

In addition, you must use a special configuration file for your Statistics Server, which contains the appropriate statistics definitions for Find. This configuration file is included in the Find ZIP package.

NOTE:

The Find interactive installer does not include the configuration file.

To configure Statistics Server

1. Open `install/statsserver/statsserver-required-config.cfg`. This file contains the statistics definitions that the Statistics Server requires.
2. Copy the definitions from `statsserver-required-config.cfg` to your Statistics Server configuration file.

NOTE:

If you want to include extra statistics definitions that are not included in `statsserver-required-config.cfg`, those extra statistics must have an IDOL name in order for the server to start.

3. Restart Statistics Server.

For information on how to send statistics to Statistics Server, see the *Query Manipulation Server Administration Guide*.

User Roles

Find user roles control the access that a particular user has to the Find user interface. Each role provides access to a particular set of Find features, and you assign your users to one or more roles that they need to access the features that you want them to use.

Users and roles are stored in the IDOL Community component. Find requires three roles in IDOL Community:

- `FindUser`. The search end-user role.
- `FindAdmin`. This role controls access to the Settings page. Micro Focus does not recommend that you grant the `FindAdmin` role to end users who do not have system administrator responsibilities.
- `FindBI`. Users with the `FindBI` role have access to BIFHI features in Find (Sunburst charts, topic maps, comparisons, saved searches, and so on).

NOTE:

Users must belong to the `FindUser` role to log in to the application; users with only the `FindAdmin` or `FindBI` roles cannot log in.

Create IDOL Community Roles

Find does not automatically create the roles in IDOL Community. You, or your IDOL system administrator, must create these roles.

The following procedure describes how to create the roles directly in IDOL Community.

To create IDOL Community roles

1. In your web browser, go to `http://CommunityHost:CommunityPort/action=admin` (where *CommunityHost* is the IP address or name of the machine on which Community is installed, and *CommunityPort* is the port that you have installed Community on).

The IDOL Admin interface for Community opens.

2. Click **Control > Roles**.
3. Click **Add Role** to create the roles.

TIP:

You can also use this page in the IDOL Admin interface to assign users to the roles, by selecting a role and clicking **Add User**.

To manage the users in your system, click **Control > Users**.

Clustering

Find does not currently support clustering of the web application.

Part II: Configure Find

This section provides information about more advanced configuration for Find to customize the display and functionality.

- [Configure Find on the Settings Page, on page 25](#)
- [Configure Find Using the Configuration File, on page 29](#)
- [Configure Dashboards, on page 48](#)
- [Configure Find with System Properties, on page 80](#)

Chapter 5: Configure Find on the Settings Page

This section describes the features that you can configure by using the Settings page in the Find interface, and gives instructions on how to set up those features.

• Configure IDOL View Server	25
• Configure MMAP Integration	26
• Configure Map Visualizations	26
• Enable Polling for Saved Searches	27
• Upload a Custom Logo	28

Configure IDOL View Server

IDOL View Server converts original documents into HTML format for viewing in a Web browser. It can also highlight search terms in the document, when displaying a search results. Find uses IDOL View Server to create the document previews in searches.

There are three viewing modes. The appropriate one to use depends on how you index your documents, and your View Server configuration:


- **Universal.** In this mode, Find requests document from a View Server that is configured in Universal Viewing mode. This mode allows View to use information in your data index to work out how to retrieve the original documents for viewing, by using a file, Web URL, or the Distributed Connector. To use this mode, you must enable universal viewing in your View Server configuration file. For more information, refer to the *IDOL Server Administration Guide*.
- **Reference Field.** In this mode, Find requests documents from View Server by using a document reference field, which you can configure. In this case, the reference must be sufficient to identify the original location of the document, such as a URL or file path (if you configure IDOL View Server to view local files).
- **Connector.** In this mode, Find uses a connector to retrieve the original document. The connector can retrieve the original document from the source repository, which simplifies the storage and retrieval. You must provide the host and port of the connector to use. Find uses the `AUTN_IDENTIFIER` document field and connector group to request the original document.

You can use any connector that supports the View action. If you want to retrieve documents from multiple external repositories, you can use the Distributed Connector. For more information about the IDOL configuration for this set up, refer to *IDOL Expert*.

The option that you choose depends on how you store and access your documents. If you can access all your documents by file path or URL, use the **Reference Field** option. If you use an external repository with its own authentication to store your documents (such as Microsoft Exchange, or Microsoft SharePoint), then you can use the **Connector** mode to simplify access and retrieval.

Users with the **FindAdmin** role can use the Settings pages to configure the IDOL View Server.

To configure IDOL View Server

1. In the toolbar on the top right of the page, click .
2. In the list, click **Settings**.

The Settings page opens.

3. Find the **View** section, and specify the host and ACI port of your IDOL View Server.
4. From the **Viewing Mode** list, select the mode to use. Depending on the mode that you use, further options are required:
 - In **Reference Field** mode, specify the IDOL reference field to use to identify your documents in IDOL View Server.
 - In **Connector** mode, specify the host name and ACI port of the connector to use to retrieve documents.
5. Click **Test Connection** to ensure that Find can connect to your IDOL View Server and Distributed Connector (if required).

NOTE:

In Universal mode, **Test Connection** checks the connection to the IDOL View Server, but does not check whether the other components that View needs to access are configured correctly. You must check any additional components in your View Server configuration.

6. Click **Save Changes**.

Configure MMAP Integration

You can configure Find to integrate with Media Management and Analysis Platform (MMAP). Configuring Find in this way means that you can open indexed rich media documents in the MMAP application from a button in preview mode or document detail view, for richer video and audio exploration.

To configure MMAP integration

1. Click the cog in the top right corner of the application, and then click **Settings**.
2. In the **MMAP** section, enter the MMAP host and port, for example `http://localhost:8080`.

NOTE:

This URL must be resolvable by the Find end-user's Web browsers, because it is used for hyperlinks between Find and MMAP (rather than being used for Find and MMAP server-to-server communications).

3. Click **Enable MMAP**.
4. Click **Save Changes**, then confirm your changes.

Configure Map Visualizations

You can enable the Map visualizer to display the location data for results documents. When you enable this functionality, the Find user interface includes a Map View results visualizer tab.

NOTE:

The default configuration uses the IDOL fields `LAT` and `LON` to retrieve location information from your documents. If you want to use different fields, or multiple location field sets, you must configure them in the Find `config.json` configuration file. See [Change or Add Location Field](#)

Names, on page 36.

To enable map visualizations

1. Click the cog in the top right corner of the application, and then click **Settings**.
2. In the **Mapping** section, specify the following settings:
 - **Tile Server Url Template.** The full Slippy Map Tilenames (SXYZ) URL for requesting a tile from an accessible tile server, with the *x*, *y*, and *z* coordinates and *S* server replaced with curly brace variables. For example, `https://{s}maps.example.com/tiles/{z}/{x}/{y}.png`.
 - **Attribution.** An optional string to display in the bottom right corner of the map (for example, a copyright statement).
3. In the **Results to load each time** section, choose the maximum number of location points that you want to be able to render on your map view.

NOTE:

Setting this to a high number makes the map slow to render.

4. Click **Enable Mapping**.
5. Click **Save Changes**, and then confirm your changes.

You can use any image file type that your browser supports.

There are many compatible tile servers available as a service over the internet, including Mapbox (<https://www.mapbox.com>), MapQuest (<http://www.mapquest.com/>), and most OpenStreetMap servers.

CAUTION:

Ensure that you check the terms of use before you use these services.

You can also implement your own Tile Map Service to work with the Map Visualizer, as long as you use an accepted format for the tile names. For more information, see <http://leafletjs.com/reference.html#url-template>.

Enable Polling for Saved Searches

Users with the `FindBI` role can save their searches in Find as a tab which they can return to later. Because databases are often updated or changed, the result set might change over time. You can configure Find to poll for any changes to the saved search results since the search was saved, and then display the number of new results in the tab.

To enable or disable polling

1. Click the cog in the top right corner of the application, and then click **Settings**.
2. In the **Saved Searches** section, click **Enable Polling** or **Disable Polling** as appropriate.
3. Click **Save Changes**.

To specify the interval between polling attempts

1. Click the cog in the top right corner of the application, and then click **Settings**.
2. In the **Saved Searches** section, type the number of minutes that should elapse in between polling attempts in the **Polling Interval (in minutes)** field.
3. Click **Save Changes**.


NOTE:

Polling for saved searches counts as user activity, and delays a session timeout for the user. If your polling interval is shorter than your session timeout, the polling causes the session to continue until you close the browser window. See [Configure Session Timeout, on page 85](#).

Upload a Custom Logo

You can modify Find to display a custom logo, by using the Customizations settings page. This page allows you to upload and select a logo for the Find splash screen (available to users who do not have the **FindBI** user role), and the navigation bar.

To upload a custom logo

1. In the toolbar on the top right of the page, click .
2. In the list, click **Customizations**.
The Customizations page opens.
3. To add a new main splash screen logo, click the upload box in the Splash Screen Logo section of the page and navigate to the file that you want to upload, or drag a logo file to the upload box.
4. To add a new navigation bar logo, click the upload box in the Small Logo section of the page and navigate to the file that you want to upload, or drag a logo file to the upload box.
5. To change the logo to use, find your logo in the appropriate list and click **Apply**.
6. Refresh your Web browser to display your changes.

Chapter 6: Configure Find Using the Configuration File

This section describes the features that you can configure by editing the `config.json` file in the Find home directory, and gives instructions on how to set up those features.

• Configure User-Friendly Names for Fields and Values	29
• Configure Fields to Show or Hide in Document Previews	31
• Configure Fields to Export	32
• Configure the Order of Filters	33
• Configure the Order of Parametric Fields	34
• Configure Parametric Filters to Never Show or Always Show	35
• Configure Find with Different IDOL Document Root Fields	36
• Change or Add Location Field Names	36
• Configure the Trending Visualizer	39
• Change the Maximum Number of Documents Used to Calculate a Topic Map	41
• Change the Maximum Number of Documents to Use to Compare Searches	42
• Enable or Disable the Filter Search Bar	42
• Enable or Disable the Related Concepts Panel	43
• Open a Shared Search as a New Query	44
• Customize Error Messages	44
• Exclude Databases from Queries	45
• Configure Entity Search	45
• Modify the Length of a Document Summary	47

Configure User-Friendly Names for Fields and Values

You can configure Find to display user-friendly names for fields and values in the user interface, in place of the values from your raw data. Find uses the display names that you specify in the parametric filter lists, as well as document metadata and the fields in document previews.

NOTE:

You cannot modify the display name for the standard metadata fields that display in the full page document preview (for example, `title`, `reference`, and `database`).

You configure user-friendly display names by adding and modifying the `fieldsInfo` section of the Find configuration file.

To configure display names for fields and values

1. Go to the Find home directory.
2. Open `config.json` in a text editor.
3. Find or add the `fieldsInfo` section.

4. Create a property and definition object for each field that you want to modify. You can use the following properties:

names	(Array, strings) An array of raw field names that you want to apply the field and value display names to.
displayName	(String) The user-friendly display name to use in the Find user interface. <div> NOTE: If you specify a display name, Find applies it to all the fields in the <code>names</code> array. This might result in multiple identical field names listed in the parametric filter list, corresponding to the different raw fields in your data. In general, Micro Focus recommends that you do not specify the display name if you set multiple values in <code>names</code>. You can specify multiple field names without the display name to normalize the field values to consistent display values. </div>
values	(Object) A JSON object to specify a field value and the display value to use for that value. This object has the following subproperties: <ul style="list-style-type: none"> <code>value</code> (string). The raw data value. <code>displayValue</code> (string). The user-friendly display name to use for this value.
advanced	(Boolean) A Boolean value that specifies whether to display this field in document previews. For more information, see Configure Fields to Show or Hide in Document Previews, on the next page .
csvExport	(Boolean) A Boolean value that specifies whether to include this field in the CSV export. For more information, see Configure Fields to Export, on page 32 .

For example:

```
"fieldsInfo": {
  "country": {
    "names": [ "COUNTRY_ORIGIN" ],
    "displayName": "Country",
    "values": [ { "value": "UK", "displayValue": "United Kingdom of Great
Britain and Northern Ireland" } ]
  },
  "elevation": {
    "names": [ "PLACE_ELEVATION" ],
    "displayName": "Place Elevation (ft)"
  }
}
```

The user interface displays a field called *Country*, with the value *United Kingdom of Great Britain and Northern Ireland*, and a *Place Elevation (ft)* field.

5. Save the file, and then restart Find to apply your configuration changes.

Configure Fields to Show or Hide in Document Previews

You can configure Find to show or hide particular fields in the results list document previews.

In the default Find configuration, the document preview does not show any fields for text documents. For videos, it shows the URL, content type, and transcript (if one exists).

When you expand the preview, the full document display page has a metadata panel. This shows some standard document metadata, such as the title, reference, and date, as well as an automatically generated document summary. By default, it also includes the URL for all documents, and the content type for videos. You can also display any additional IDOL fields associated with the document by clicking **Show Advanced**.

You can modify your Find configuration to show additional fields in the document previews, or to hide certain fields.

NOTE:

You cannot hide the standard metadata or document summary in the full document display page.

You control the display of fields in the document preview by modifying the `fieldsInfo` section of the Find configuration file.

To modify the fields that show in the document preview

1. Go to the Find home directory.
2. Open `config.json` in a text editor.
3. Find or add the `fieldsInfo` section.
4. Create a property and definition object for each field that you want to show, or modify an existing one. The following table describes the properties that you can use to modify the field preview display.

names	(Array, strings) An array of raw field names that you want to apply the settings to.
advanced	<p>(Boolean) A Boolean value that specifies whether to display this field in the document preview.</p> <p>Set <code>advanced</code> to <code>false</code> if you want to display the field as part of the document preview. In this case, the field is displayed in the preview, and also in the metadata section of the document display page when a user expands the preview.</p> <p>Set <code>advanced</code> to <code>true</code> to hide the field in the document preview. In this case, the field is available only in the document display page when a user clicks Show Advanced to show all available fields.</p> <div><div>NOTE:</div><div>For videos, the document preview always includes the transcript field</div></div>

	if there is a transcript (that is, you cannot hide this transcript in the preview).
displayName	(String) The user-friendly display name to use in the Find user interface. For more information, see Configure User-Friendly Names for Fields and Values, on page 29 .
values	(Object) A JSON object to specify a field value and the display value to use for that value. For more information, see Configure User-Friendly Names for Fields and Values, on page 29 .
csvExport	(Boolean) A Boolean value that specifies whether to include this field in the CSV export. For more information, see Configure Fields to Export, below .

For example:

```
"fieldsInfo": {
  "previewFields": {
    "names": [ "author", "description" ],
    "advanced": false
  },
  "advancedFields": {
    "names": [ "contenttype" ],
    "advanced": true
  }
}
```

In this example, Find displays the `author` and `description` fields in the document preview. It hides the `contenttype` field, which is then available only in the expanded document display view when a user clicks **Show Advanced**.

5. Save the configuration file, and then restart Find to apply your configuration changes.

Configure Fields to Export

You can control the fields that Find exports when a user exports the results to CSV.

By default, Find exports all document fields. You can exclude particular fields from the export, in which case Find exports all field except for the excluded ones. You can also specify particular fields to explicitly include, in which case Find exports only the allowed fields.

You configure CSV export fields by modifying the `fieldsInfo` section of the Find configuration file.

To configure the fields to export

1. Go to the Find home directory.
2. Open `config.json` in a text editor.
3. Find or add the `fieldsInfo` section.
4. Create a property and definition object for each field that you want to modify. You can use the following properties:

names	(Array, strings) An array of raw field names that you want to apply the settings to.
csvExport	<p>(Boolean) A Boolean value that specifies whether to export this field in the CSV export.</p> <p>Set <code>csvExport</code> to <code>false</code> to exclude the field from CSV export. By default, Find includes all fields. You can use this option to exclude some fields.</p> <p>Set <code>csvExport</code> to <code>true</code> to explicitly include the field in the CSV export. If any fields in your configuration have <code>csvExport</code> set to <code>true</code>, Find exports only those fields.</p>
displayName	(String) The user-friendly display name to use in the Find user interface. For more information, see Configure User-Friendly Names for Fields and Values, on page 29 .
values	(Object) A JSON object to specify a field value and the display value to use for that value. For more information, see Configure User-Friendly Names for Fields and Values, on page 29 .
advanced	(Boolean) A Boolean value that specifies whether to display this field in document previews. For more information, see Configure Fields to Show or Hide in Document Previews, on page 31 .

For example:

```
"fieldsInfo": {
  "country": {
    "names": [ "COUNTRY_ORIGIN" ],
    "displayName": "Country",
    "values": [ { "value": "UK", "displayValue": "United Kingdom of Great Britain and Northern Ireland" } ]
  },
  "elevation": {
    "names": [ "PLACE_ELEVATION" ],
    "csvExport": false
  }
}
```

When a user exports a document, the CSV export does not include the `PLACE_ELEVATION` field.

5. Save the file, and then restart Find to apply your configuration changes.

Configure the Order of Filters

You can change the order in which different groups of filters appear in the sidebar.

To configure the order of filter groups

1. Go to the Find home directory.
2. Open `config.json` in a text editor.
3. In the `filterOrder` section, modify the order of the array options to the order in which you want them to appear. You can also add a hyphen (-) as an array item to add a space between two groups in the list of filters.

For example, the default order is:

```
"filterOrder": ["metaFilter", "indexesFilter", "datesFilter",
"geographyFilter", "-", "parametricFilter"]
```

where

<code>metaFilter</code>	is the filter search box to allow you to filter the list of filters. See Enable or Disable the Filter Search Bar, on page 42 .
<code>indexesFilter</code>	is the list of data indexes (databases) for your data.
<code>datesFilter</code>	is the list of date filters.
<code>geographyFilter</code>	is the list of location filters.
<code>parametricFilter</code>	is the list of parametric fields. You can also modify the order that these parametric fields are displayed by updating the <code>parametricOrder</code> parameter. See Configure the Order of Parametric Fields, below .

You can add the filters in any order. For example:

```
"filterOrder": ["metaFilter", "-", "parametricFilter", "-", "indexesFilter",
"datesFilter", "geographyFilter"]
```

If you do not include one or more of the filter groups, Find lists it at the end, after the specified groups.

4. Save the file, and then restart Find to apply your configuration changes.

Configure the Order of Parametric Fields

You can change the order in which parametric fields appear in the sidebar.

To configure the order of parametric fields

1. Go to the Find home directory.
2. Open `config.json` in a text editor.
3. In the `parametricOrder` section, list the fields in the order in which you want them to appear. For example:

```
"parametricOrder": ["company", "author", "category"]
```

NOTE:

The names of the parametric fields need to match the IDOL field name.

4. Save the file, and then restart Find to apply your configuration changes.

Any parametric fields that you list in the `parametricOrder` section appear at the top of the list.

If you do not specify a parametric order, by default the fields are listed alphabetically.

If there are parametric fields that you did not list in the `parametricOrder` section, these appear after the ordered fields, in the default sort order (that is, alphabetical). For example:

```
Company
Author
Category
Date Identified
Family
Genus
```

NOTE:

Any fields that you add to the parametric never show list do not appear in the user interface. Similarly, if you have a parametric always show list, the interface does not display any fields that you specify in the parametric order that do not appear in the always show list. See [Configure Parametric Filters to Never Show or Always Show, below](#).

Configure Parametric Filters to Never Show or Always Show

You can configure Find to always show or never show particular parametric filters. For example, if you have internal filters, you can configure Find so that they do not appear to your users.

To configure parametric filters that you never want to show

1. Go to the Find home directory.
2. Open `config.json` in a text editor.
3. In the `parametricNeverShow` section of the configuration file, specify a list of parametric fields that you never want to show. For example:

```
"parametricNeverShow": ["INTERNAL_FIELD", "FILE_TOO_LARGE", "HAS_BEEN_VIRUS_CHECKED"]
```

This setting removes the `INTERNAL_FIELD`, `FILE_TOO_LARGE`, and `HAS_BEEN_VIRUS_CHECKED` parametric fields from the lists of parametric fields sent to the client.

The user cannot see or work with these fields, although the fields are still visible in the metadata display.

NOTE:

The names of the parametric fields need to match the IDOL field name.

4. Restart Find to apply your configuration changes.

To configure parametric filters that you want to always show

1. Go to the Find home directory.
2. Open `config.json` in a text editor.

3. In the `parametricAlwaysShow` section of the configuration file, specify a whitelist of parametric fields. For example:

```
"parametricAlwaysShow": ["COLOR", "PRICE", "MODEL"]
```

In the above example, the `COLOR`, `PRICE`, and `MODEL` fields are the only parametric fields that are available to filter by in the Find user interface. Any fields not included in the always show list are not displayed. If there are any fields in both the always show and the never show lists, the never show list takes priority and the field is not visible.

NOTE:

The names of the parametric fields need to match the IDOL field name.

4. Restart Find to apply your configuration changes.

Configure Find with Different IDOL Document Root Fields

In most standard IDOL configurations, index XML files have a root field `DOCUMENTS`, which contains all the individual documents and fields. Depending on your IDOL configuration, you might have other root fields. In this case you must also configure Find to recognize the different document root fields so that it can extract the parametric fields from different documents.

To configure Find to recognize different IDOL root fields

1. Go to the Find home directory.
2. Open `config.json` in a text editor.
3. Find the `idolFieldPathNormalizerXMLPrefixes` property (or add it if it does not exist), and set it to an array of XML root fields that you want to use. For example:

```
"idolFieldPathNormalizerXMLPrefixes" : [ "DOCUMENTS", "ADMISSIONLIST" ]
```

4. Save the file, and then restart Find to apply your configuration changes.

NOTE:

If you define custom field roots, you cannot use full field paths in your `fieldsInfo` configuration (in the `names` array). You must specify the leaf field node only. For example, if your documents include the field `ADMISSIONLIST/DOCUMENT/COUNTRY_ORIGIN`, specify it as `COUNTRY_ORIGIN` only:

```
"fieldsInfo": {
  "country": {
    "names": [ "COUNTRY_ORIGIN" ]
  }
}
```

Change or Add Location Field Names

By default, Find retrieves latitude and longitude information from the `LAT` and `LON` IDOL fields, respectively. When you are using these fields in your documents, you can configure map visualizations by using the Find configuration settings page (see [Configure Map Visualizations, on page 26](#)).

You can also update the map configuration manually if you want to use different fields as the latitude and longitude values, or if you want to use multiple location values. In this case, you modify the `map` configuration object in your `config.json` file, and update the `fieldsInfo` configuration with any required fields.

The following table describes the properties that you can set in the `map` configuration section.

Property	Type	Description
<code>enabled</code>	Boolean	Set to <code>true</code> to enable the map visualizations in Find.
<code>attribution</code>	String	An optional string to include in the bottom right of the map visualization, for example a copyright notice.
<code>initialLocation</code>	Object	The location to use as the initial center point of the map visualization. This object contains the following properties: <ul style="list-style-type: none"> <code>latitude</code> (number). The latitude value of the initial location. <code>longitude</code> (number). The longitude value of the initial location.
<code>tileUrlTemplate</code>	String	The full Slippy Map Tilenames (SXYZ) URL for requesting a tile from an accessible tile server, with the <code>x</code> , <code>y</code> , and <code>z</code> coordinates and <code>S</code> server replaced with curly brace variables. For example, <code>https://{s}maps.example.com/tiles/{z}/{x}/{y}.png</code> .
<code>resultsStep</code>	Integer	The maximum number of location points that you want to be able to render on your map view. Setting this parameter to a high value can make the map slow to render.
<code>locationFields</code>	Array, objects	The details of the fields that you want to use to plot locations. Each location field object in the array contains the following properties: <ul style="list-style-type: none"> <code>displayName</code> (string). The name of the location set. <code>latitudeField</code> (string). The name of the field to use to retrieve latitude values. You must also define this field in the <code>fieldsInfo</code> configuration object. <code>longitudeField</code> (string). The name of the field to use to retrieve longitude values. You must also define this field in the <code>fieldsInfo</code> configuration object. <code>iconName</code> (string). The name of the icon to use for this location set. <code>iconColor</code> (string). The color to use for the icon. <code>markerColor</code> (string). The color to use for the location markers on the map visualization.

For each object in your `locationFields` array, Find displays a set of location markers on your map visualization. For example, the following map configuration configures two sets of location fields to use in the map visualization. The map visualization includes two sets of markers, for `DefaultLocation` and `OGLocation`.

```
"map": {
  "enabled": false,
  "attribution": null,
  "initialLocation": {
```

```

        "latitude": 51.5074,
        "longitude": 0.1278
    },
    "tileUrlTemplate": null,
    "resultsStep": 2500,
    "locationFields" : [
        {
            "displayName" : "DefaultLocation",
            "latitudeField" : "latitude",
            "longitudeField" : "longitude",
            "iconName" : null,
            "iconColor" : null,
            "markerColor" : null
        }, {
            "displayName" : "OGLocation",
            "latitudeField" : "oglatitude",
            "longitudeField" : "oglongitude",
            "iconName" : "hp-pin",
            "iconColor" : "blue",
            "markerColor" : "orange"
        }
    ]
}

```

Where `latitude`, `longitude`, `oglatitude`, and `oglongitude` are fields defined in the `fieldsInfo` configuration:

```

"fieldsInfo": {
    "latitude": {
        "names": [
            "NODE_PLACE/LAT",
            "LAT"
        ],
        "type": "NUMBER",
        "advanced": true
    },
    "longitude": {
        "names": [
            "NODE_PLACE/LON",
            "LON"
        ],
        "type": "NUMBER",
        "advanced": true
    },
    "oglatitude" : {
        "names" : [ "OG_LATITUDE" ]
        "type" : "number",
        "advanced" : true,
    },
    "oglongitude" : {

```

```

    "names" : [ "OG_LONGITUDE" ]
    "type" : "number",
    "advanced" : true,
  }
}

```

TIP:

If you want to use multiple location fields in your map visualization, but you want only one set of location markers, you can update the `fieldsInfo` configuration to add the new fields without adding a new `locationFields` option to the map configuration. For example:

```

"fieldsInfo": {
  "latitude": {
    "names": [
      "NODE_PLACE/LAT",
      "LAT",
      "OG_LATITUDE"
    ],
    "type": "NUMBER",
    "advanced": true
  },
  "longitude": {
    "names": [
      "NODE_PLACE/LON",
      "LON",
      "OG_LONGITUDE"
    ],
    "type": "NUMBER",
    "advanced": true
  }
}

```

For more information about `fieldsInfo` configuration, see [Configure User-Friendly Names for Fields and Values](#), on page 29.

Configure the Trending Visualizer

The Trending visualization displays a chart of document rate against time for the values in a particular field, to show how particular topics change in popularity. The document rate is the number of documents added in the specified time unit (second, minute, hour, day, or year, as labeled on the chart). The appropriate time unit is chosen according to the document rates in your data.

You can configure the date field to use to specify the date and time for your documents, and you can set the number of field values and date values to display on the chart.

Configure the Date Field

The Trending visualizer uses the value of the specified date field as the document date when it creates the chart. By default, Find uses the `AUTN_DATE` field as the date field, but you can modify this field in the `config.json` configuration file.

To configure the date field to use for the Trending visualizer

1. Go to the Find home directory.
2. Open `config.json` in a text editor.
3. Find the `trending` section, or create one if it does not exist.
4. Set the `dateField` property to the name of the field to use. You can use any date field, as long as it was configured as a date field in your IDOL Content component before you indexed the data.

For example:

```
"trending" : {
  "dateField" : "DOCUMENTDATE",
  "numberOfValues": 10,
  "defaultNumberOfBuckets": 20
}
```

5. Save the file, and restart Find to apply your configuration changes.

Configure the Number of Field Values to Display

By default, the Trending visualizer displays the trends for ten values in the selected parametric field. You can modify this value in the `config.json` configuration file.

To configure the number of values to display in the Trending visualizer

1. Go to the Find home directory.
2. Open `config.json` in a text editor.
3. Find the `trending` section, or create one if it does not exist.
4. Set the `numberOfValues` property to the number of values to include in the visualization.

For example:

```
"trending" : {
  "dateField" : "DOCUMENTDATE",
  "numberOfValues": 5,
  "defaultNumberOfBuckets": 20
}
```

NOTE:

In the visualization, each value has a line with a different color. There are ten possible colors, so if you set `numberOfValues` higher than ten, some values share a color.

5. Save the file, and restart Find to apply your configuration changes.

Configure the Number of Date Ranges to Use

The Trending visualizer displays document rate against time. It selects date values from your documents and organizes the available dates into the specified number of ranges.

By default, the visualizer divides the total range of dates into 20 equal-sized buckets. Users can modify this number by using a slider, which by default allows them to change the number of buckets between three and 100.

You can modify the default number of buckets to display in the `config.json` configuration file, and the maximum and minimum values to add on the slider.

To configure the number of values to display in the Trending visualizer

1. Go to the Find home directory.
2. Open `config.json` in a text editor.
3. Find the `trending` section, or create one if it does not exist.
4. Set the `defaultNumberOfBuckets` property to the number of date ranges to include in the visualization. Set `minNumberOfBuckets` and `maxNumberOfBuckets` to the minimum and maximum values to use in the UI slider, respectively.

For example:

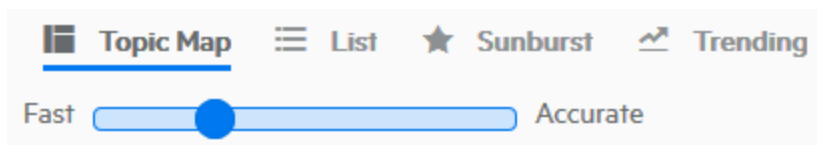
```
"trending" : {
  "dateField" : "DOCUMENTDATE",
  "numberOfValues": 5,
  "defaultNumberOfBuckets": 30,
  "minNumberOfBuckets": 5,
  "maxNumberOfBuckets": 40
}
```

5. Save the file, and restart Find to apply your configuration changes.

Change the Maximum Number of Documents Used to Calculate a Topic Map

The topic map shows a collection of related concepts that are related to the results generated by a search. The choice of related concepts is affected by the number of result documents that are used to generate them. A larger number of documents provides a more accurate reflection of the full result set, but a smaller number is faster to calculate.

The user can control this by using the fast-accurate slider in the topic map view.



The slider enables the user to choose any number of documents for the calculations, between 50 at the fast end, and a configurable maximum value at the accurate end. By default, the maximum value is 1000.

To change the maximum number of documents to use

1. Go to the Find home directory.
2. Open `config.json` in a text editor.
3. Set `topicMapMaxResults` to the maximum number of results to use to generate the topic map. For example:

```
"topicMapMaxResults": 500
```

This example sets the range to be 50 (fast) to 500 (accurate).

4. Restart Find to apply your configuration changes.

Change the Maximum Number of Documents to Use to Compare Searches

The Compare option allows you to compare two saved searches. By default, Find compares all the available results for both searches, which provides the most accurate comparison. For very large data sets, complete comparisons for large queries might be quite slow. In this case, you can improve the performance by reducing the maximum number of results to use in the comparison.

To change the maximum number of documents to use

1. Go to the Find home directory.
2. Open `config.json` in a text editor.
3. Add the `comparisonStoreStateMaxResults` property, and set it to the maximum number of results to use in the comparison. For example:

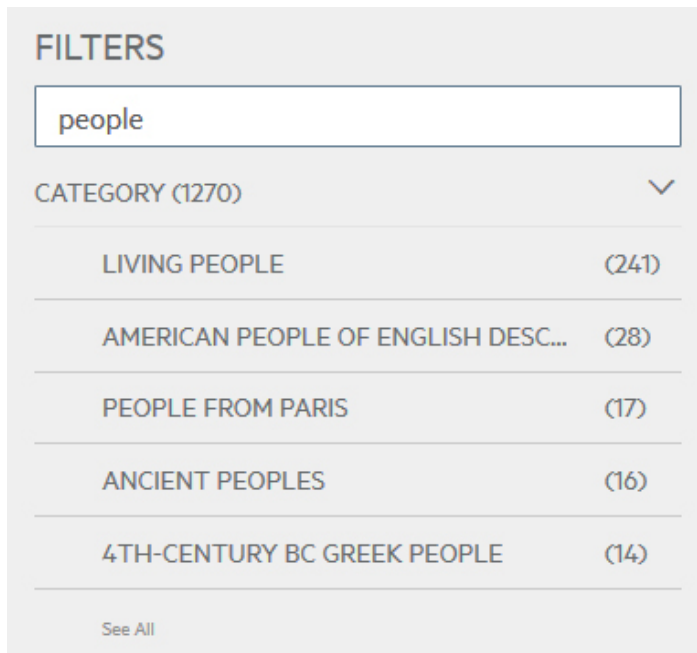
```
"comparisonStoreStateMaxResults": 100000
```

This example compares the first 100000 results.

4. Restart Find to apply your configuration changes.

Enable or Disable the Filter Search Bar

The filter search bar is a search bar that enables users to search through the filters in the left-hand panel in the Find user interface.



By default, this feature is present only for users with the `FindBI` user role. You can enable or disable the filter search bar for either user role.

To enable or disable the filter search bar

1. Go to the Find home directory.
2. Open `config.json` in a text editor.
3. In the `enableMetaFilter` section, set a user role to `false` to disable the filter search bar, or `true` to enable it. For example:

```
"uiCustomization": {
  "options": {
    "enableMetaFilter": {
      "user": false,
      "bi": true
    },
    "enableRelatedConcepts": {
      "user": true,
      "bi": false
    }
  }
},
```

Enable or Disable the Related Concepts Panel

The related concepts panel is shown on the right of the Find search page. By default, it is enabled for users without the `FindBI` user role, and disabled for those with the `FindBI` user role. You can disable or enable related concepts for either user role.

To enable or disable the related concepts panel

1. Go to the Find home directory.
2. Open `config.json` in a text editor.
3. In the `enableRelatedConcepts` section, set a user role to `true` to enable the display of the related concepts panel, or `false` to disable it. For example:

```
"uiCustomization": {
  "options": {
    "enableRelatedConcepts": {
      "user": true,
      "bi": false
    },
    "enableMetaFilter": {
      "user": false,
      "bi": true
    }
  }
}
```

4. Restart Find to apply your configuration changes.

Open a Shared Search as a New Query

When a user opens a shared search that they did not create, for example by clicking on a dashboard, by default Find opens the shared search directly if they have permission to edit it, or in read-only mode if they do not. The user can create a new query from the read-only one by using the **Open as Query** option.

You can configure Find to automatically always open a new query for a shared search.

To configure Find to open shared searches as a new query by default

1. Go to the Find home directory.
2. Open `config.json` in a text editor.
3. In the `uiCustomization` section, add the `openSharedDashboardQueryAsNewSearch` property, and set it to `true` to automatically open a shared search as a new query. For example:

```
"uiCustomization": {
  "openSharedDashboardQueryAsNewSearch": true
}
```

Customize Error Messages

When an error occurs in Find, the application displays an error message describing the nature of the problem. If the error is of a nature such that a user cannot solve it, a message prompting the user to contact support is shown.

To customize error messages

1. Go to the Find home directory.
2. Open `config.json` in a text editor.
3. To add your relevant support contact details, add an `errorCallSupportString` option to the `uiCustomization` section. For example:

```
"uiCustomization": {
  "errorCallSupportString": "Contact Support on 01 234 56."
}
```

4. Restart Find to apply your configuration changes.
Find uses the new message instead of the default message.

Exclude Databases from Queries

In the default configuration, Find sends queries to all your IDOL databases. You can change the default database selection by excluding some databases in your Find configuration file (`config.json`). In this case, the databases are not included in a query by default, but users can explicitly select the databases if they want to search them.

To exclude certain databases from queries by default

1. Go to the Find home directory.
2. Open `config.json` in a text editor.
3. In the `uiCustomization` section, set the `defaultDeselectedDatabases` property to an array of databases that you want to exclude from the list of databases that Find queries by default.

For example:

```
"uiCustomization": {
  "defaultDeselectedDatabases": [
    "multilingual"
  ]
}
```

4. Restart Find to apply your configuration changes.

Configure Entity Search

Entity searches allow users to run an additional entity search against a special IDOL Content component.

When you configure entity search, users can select text in the Find user interface and it opens a dialog box with search results from the entity Content. You can also configure an Answer Server, in which case the dialog box includes a question input field.

You configure entity search by adding the `entitySearch` section to your `config.json` configuration file, with details of the search components to use, and any custom fields.

To configure entity search

1. Go to the Find home directory.
2. Open `config.json` in a text editor.
3. Find or add the `entitySearch` section.
4. Set the `enabled` property to **true** to enable entity search.
5. Add a `server` property to set the details for your entity search component. This property is an object, which accepts the subproperties described in the following table. In most cases, you must change at least `host` and `port` to the appropriate values for your IDOL Content component.

<code>host</code>	(String) The host name or IP address of the component to use for entity search. The default value is <code>localhost</code> .
<code>port</code>	(Integer) The port of the component to use for entity search. The default value is <code>9050</code> .
<code>protocol</code>	(String) The protocol to use to connect to the IDOL Content component. The default value is <code>HTTP</code> .

6. To enable an Answer Server to use for entity search, add an `answerServer` property. This property is an object.

To enable Answer Server, set the `enabled` subproperty to **true** and add a `server` subproperty to set the details for your entity search Answer Server.

This `server` object contains the same subproperties as for the main entity search component (see [Step 5](#)), with the following default values:

```
"server": {
  "protocol": "HTTP",
  "host": "localhost",
  "port": 7700}
```

In most cases, you must change at least `host` and `port` to the appropriate values for your Answer Server.

In the `answerServer` object, you can optionally also set `systemNames` to an array of Answer Server systems that you want to query. By default, Find uses all systems.

7. Set any additional optional parameters in the `entitySearch` object. The following table lists the available parameters.

<code>agentBooleanField</code>	(String) A field in your entity search index that contains AgentBoolean expressions. For more information, refer to the <i>IDOL Server Reference</i> .
<code>combine</code>	(String) The combine option to use in queries to the entity search Content. For more information, refer to the <i>IDOL Server Reference</i> .
<code>idolPrintFields</code>	(Array, strings) A list of fields from your IDOL documents to display in the entity search results.

8. Save the file, and then restart Find to apply your configuration changes.

For example:

```

"entitySearch" : {
  "server" : {
    "protocol" : "HTTP",
    "host" : "entitycontent.example.com",
    "port" : 9000
  },
  "enabled" : true,
  "agentBooleanField" : "AGENTBOOL",
  "combine" : "FieldCheck",
  "idolPrintFields" : [ "WIKIPEDIA_ENG", "WIKIPEDIA_IMAGE", "PERSON_DATE_OF_
BIRTH", "PERSON_DATE_OF_DEATH", "PERSON_PROFESSION", "WIKIPEDIA_IMAGE", "LAT",
"LON", "PLACE_COUNTRY_CODE", "PLACE_REGION1", "PLACE_POPULATION", "PLACE_TIMEZONE",
"WIKIPEDIA_IMAGE", "URL_HOMEPAGE", "COMPANY_GOOGLE" ],
  "answerServer": {
    "server": {
      "protocol": "HTTP",
      "host": "10.2.21.210",
      "port": 16000
    },
    "systemNames": ["FactBank", "PassageExtractor"],
    "enabled": true
  }
}

```

After you configure entity search, you might also want to configure a custom template to display entity search results. See [Configure Templates in Find, on page 74](#).

Modify the Length of a Document Summary

Find generates document summaries for the results that it returns in queries to give users a quick idea of what each document is about. By default, Find uses a maximum summary length of 250 characters (where a character is a single byte). This value is generally suitable for producing summaries for English and most European languages.

You might need to increase the summary length if you have documents in multibyte languages, such as Chinese, Japanese, Korean, and Thai.

For example, if you want to generate summaries for Thai text, you might increase the maximum summary length to 750, because each Thai text character corresponds to three UTF-8 bytes.

To change the document summary length

1. Go to the Find home directory.
2. Open `config.json` in a text editor.
3. Find or add the `documentSummaryMaxLength` property, and set it to the maximum length to use. This value must be an integer.

```
"documentSummaryMaxLength": 750
```

4. Save the file, and then restart Find to apply your configuration changes.

Chapter 7: Configure Dashboards

This section describes how to set up and configure dashboards.

• Find Dashboards	48
• Dashboard Configuration	49
• Dashboard Grid Size	51
• Widgets	52
• Widget Configuration	52
• Dashboard Export Notes	64
• Example Dashboard Configuration	64

Find Dashboards

Dashboards in Find are fully configurable UI pages, which you can use to set up and display the content, search results, and visualizations that you refer to most regularly.

Each dashboard contains one or more *widgets*, each of which displays a particular set of content, and can be backed by an Find saved search. The standard set of widgets provide a variety of visualizations for your saved searches, as well as static content, images, or videos. For a full list of available widgets, see [Widgets, on page 52](#).

The widgets that use a saved search display the latest available results for a particular visualization or results list. The widgets are not interactive. However, you can click a particular widget to open the saved search that the widget displays, to view and explore the information in more detail.

You configure dashboards by modifying the `dashboards.json` configuration file, in the `customizations` directory of your Find installation home directory.

Dashboards are available only to users with the `FindBI` user role. You can also optionally restrict individual dashboards to a specific set of roles.

The list of available dashboards is displayed in the **Dashboards** section of the Find navigation side bar.

You can also access the dashboard directly by using a link of the form

`FindURL/public/dashboards/DashboardName`, where *FindURL* is the base URL for your Find user interface, and *DashboardName* is the configured name of the dashboard that you want to display.

The dashboards have a full screen option. In this case, the session continues indefinitely without logging the user out of Find, unless you are using IDOL document security (in which case the session ends when the user security token expires).

Users can also export dashboards to .pptx presentation format, either as a single slide or with each widget on a separate slide.

NOTE:

There are some restrictions for export on certain types of widget. See [Dashboard Export Notes, on page 64](#).

Dashboard Configuration

You create and configure dashboards by modifying the `dashboards.json` configuration file, in the `customization` directory in your Find installation home directory.

This configuration file contains the `dashboards` object, which is an array of dashboard configuration objects. To add a dashboard, you add an object to this array.

In the Find user interface, the **Dashboards** section in the navigation sidebar lists your dashboards in the same order that you list them in the configuration file.

When you add a dashboard, you can configure the name and the number of grid squares to divide your dashboard into, as well as how often to update the contents of the widgets.

TIP:

If you belong to the **FindAdmin** role, you can reload the dashboard configuration to make any configuration changes available to your users without restarting Find.

To reload the dashboards, you must send the following API call to Find directly, by typing the URL into your Web browser address bar. You must be logged into Find as a user with the **FindAdmin** role in the same Web browser, because Find uses the session cookie to authorize the action.

```
http://FindURL/api/admin/customization/config/reload
```

Where *FindURL* is the URL of your Find server.

This option also reloads any other customization configurations (for example, applications).

The following table lists the configuration properties that you can use for the dashboards.

Property	Required	Type	Description
<code>dashboardName</code>	Yes	String	The name to use for the dashboard. This name is displayed in the Dashboards navigation section.
<code>enabled</code>	Yes	Boolean	Set to true to enable the dashboard and display it in the Dashboards navigation section. Set to false to hide the dashboard.
<code>width</code>	Yes	Integer	The width of the dashboard grid, in grid squares (where a widget takes up a minimum of one grid square). For more information, see Dashboard Grid Size, on page 51 .
<code>height</code>	Yes	Integer	The height of the dashboard grid, in grid squares (where a widget takes up a minimum of one grid square). For more information, see Dashboard Grid Size, on page 51 .
<code>updateInterval</code>	Yes	Integer	The time (in seconds) to wait between each refresh of the widgets in this dashboard. This parameter affects only widgets that reference a

Property	Required	Type	Description
			query saved search. It does not apply to static widgets and widgets that reference a snapshot.
widgets	Yes	Object Array	An array of objects that define the widgets to include in the dashboard, their positions and sizes, and the individual widget configuration. For details of the properties to use for widget configuration, see Widget Configuration, on page 52 .
displayWidgetNames	No	String Enum	<p>A string that determines whether to display the names of the widgets. Use one of the following options:</p> <ul style="list-style-type: none"> • always. Always display names, if provided in the widget configuration. • onhover. Display names only when a user mouses over the widget. • never. Never display names, even if they are provided in the widget configuration. This is the default value. <p>You can override this parameter for individual widgets by using the <code>displayWidgetName</code> option in the widget configuration. See Widget Configuration, on page 52.</p>
roles	No	String Array	<p>An array of strings that specify the role names of roles that can access the dashboard. Users can see the dashboard only if they belong to at least one of the roles that you specify.</p> <p>If you do not set the roles property, the dashboard is accessible to all users.</p>

The following example shows a simple configuration for one dashboard that contains a single widget:

```
{
  "dashboards": [
    {
      "dashboardName": "Static Content Dashboard",
      "enabled": true,
      "width": 4,
      "height": 4,
      "updateInterval": 1000,
      "roles": ["CustomRole"]
      "widgets": [
        {
          "name": "Static Image",
          "type": "staticImageWidget",
          "x": 1,
```

```
        "y": 1,
        "width": 2,
        "height": 2,
        "widgetSettings": {
            "url": "http://example.com/image.png"
        }
    }
]
}
```

Dashboard Grid Size

You define the size of the dashboard by defining a number of grid squares that the dashboard can contain. Each widget takes up at least one grid square, although you can also configure it to use more space.

You cannot define the absolute size of a dashboard. The grid scales with the size of the dashboard window.

You can size and place your widgets anywhere on the grid, and define the size of each widget as a number of grid squares in width and height.

For example, if you define your grid to be four width by four height, you have sixteen grid squares, which you can use in several ways, such as:

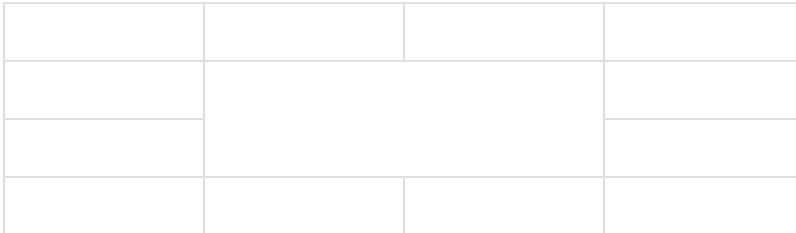
- Sixteen one-by-one widgets:

- Four two-by-two widgets:

- One two-by-four widget, and two two-by-two widgets:



- One two-by-two widget, placed centrally, surrounded by one-by-one widgets:



Widgets

There are several standard widgets available that you can use in your dashboards.

The following widgets display results or visualizations from a saved search:

- **Topic Map.** Displays the topic map visualization for the specified saved search.
- **Results List.** Displays the result list for the specified saved search.
- **Sunburst.** Displays the sunburst visualization for the specified saved search.
- **Trending.** Displays the trending visualization for the specified saved search.
- **Map.** Displays the map visualization for the specified saved search.
- **Video Panel.** Displays a video result from the specified saved search.

Clicking on one of these widgets opens the appropriate saved search page. When the current user owns the saved search page, it opens as normal, and the user can modify the search. If the current user does not own the saved search page, the page displays the results and visualizations, but the user cannot modify the search settings.

The following widgets display information that does not relate to a specific saved search:

- **Static Content.** Displays some static HTML content that you provide.
- **Static Image.** Displays an image that you reference.
- **Current Time/Date.** Displays the current date and time.
- **Time Last Refreshed.** Displays the time that the widgets on the dashboard were last refreshed, and the time that the next update is scheduled.

Widget Configuration

For each widget, you must specify the name, and the size and position of the widget in the dashboard grid.

The following table describes the parameters for all widgets. All options are required unless otherwise specified.

Property	Description	Type
name	The name of the widget.	String
type	The type of widget. See the individual widget sections for the value to use.	String Enum
x	The position of the left side of the widget, from the left side of the dashboard (the left side is 0). Widgets align to grid squares. You must set the position so that the whole widget appears within the dashboard width.	Integer
y	The position of the top of the widget, from the top of the dashboard (the top is 0). Widgets align to grid squares. You must set the position so that the whole widget appears within the dashboard height.	Integer
width	The width of the widget, in grid squares. This value must be less than the total width of the dashboard, and less than the combined width of the widgets on the same row of the dashboard.	Integer
height	The height of the widget, in grid squares. This value must be less than the total height of the dashboard, and less than the combined height of the widgets in the same column in the dashboard.	Integer
cssClass	Optional. A CSS class to use for the widget.	String
displayWidgetName	Optional. A string that determines whether to display the name of the widget. Use one of the following options: <ul style="list-style-type: none"> • always. Always display the name for this widget, if it has one. • onhover. Display the name only when a user mouses over the widget. • never. Never display the name, even if it has one in the widget configuration. <p>This option overrides the <code>displayWidgetNames</code> option in the dashboard configuration, which has a default value of never. See Dashboard Configuration, on page 49.</p>	String Enum

For more information about the dashboard grid, see [Dashboard Grid Size, on page 51](#).

For widgets that require a saved search, you must also specify the ID and type of the saved search. You add the saved search by using the `datasource` property, which is a JSON object with the following properties:

source	The type of data source to use for the widget. For a saved search, set this option to <code>SavedSearch</code> .
config	A JSON object that defines the configuration for the data source. For the <code>SavedSearch</code>

source, this object has the following required subproperties:

- **id** (string). The ID of the saved search to use for this widget. You find the ID by opening the saved search tab in the Find user interface, and checking the URL. In the URL, the saved search type and ID appear after `tab/`, as `QUERY:ID` or `SNAPSHOT:ID`, where ID is a number.
- **type** (string enum). The type of saved search: **QUERY** or **SNAPSHOT**.

The following example shows the basic widget configuration.

```
"widgets": [
  {
    "name": "TopicMap",
    "type": "TopicMapWidget",
    "x": 1,
    "y": 1,
    "width": 2,
    "height": 2,
    "datasource": {
      "source": "SavedSearch",
      "config": {
        "type": "QUERY",
        "id": "1234"
      }
    },
    "widgetSettings": {
      "maxResults": 1000
    }
  }
]
```

Each widget has some specific configuration details, which you specify in the `widgetSettings` object in the dashboard configuration. The available parameters are described in the following sections.

Topic Map Widget

For the Topic Map widget, set `type` to **TopicMapWidget**.

This widget type requires a saved search. You must set the `source` property to **SavedSearch** in the widget configuration.

The following table describes the options that you can use in the `widgetSettings` object.

Property	Required	Description	Default
<code>maxResults</code>	No	The maximum number of results to retrieve from the saved search to form the topic map. Requesting more results generally gives a more accurate topic map, but the widget might load and refresh more slowly.	300

Example widget configuration:

```

{
  "name": "TopicMap",
  "type": "TopicMapWidget",
  "x": 0,
  "y": 0,
  "width": 1,
  "height": 1,
  "datasource": {
    "source": "SavedSearch",
    "config": {
      "type": "QUERY",
      "id": "1234"
    }
  },
  "widgetSettings": {
    "maxResults": 1000
  }
}

```

Results List Widget

For the Results List widget, set type to **ResultsListWidget**.

This widget type requires a saved search. You must set the `source` property to **SavedSearch** in the widget configuration.

The following table describes the options that you can use in the `widgetSettings` object.

Property	Required	Type	Description	Default
<code>columnLayout</code>	No	Boolean	Set to true to display results in columns instead of rows.	false
<code>maxResults</code>	No	Integer	The maximum number of results to include in the results list.	6
<code>sort</code>	No	String enum	The sort option to use to order the results. Micro Focus recommends that you use only date and relevance , which match the options available in the Find user interface. However, you can use any sort method that is supported by the IDOL Content component. In this case, the result order might be different to the rest of your user interface. For details of the available sort options, refer to the <i>IDOL Content Component Reference</i> .	relevance

Example widget configuration:

```
{
  "name": "List",
  "type": "ResultsListWidget",
  "x": 1,
  "y": 1,
  "width": 2,
  "height": 2,
  "datasource": {
    "source": "SavedSearch",
    "config": {
      "type": "QUERY",
      "id": "1234"
    }
  },
  "widgetSettings": {
    "maxResults": 10
  }
}
```

Sunburst Widget

For the Sunburst widget, set type to **SunburstWidget**.

This widget type requires a saved search. You must set the source property to **SavedSearch** in the widget configuration.

The following table describes the options that you can use in the widgetSettings object.

Property	Required	Type	Description	Default
firstField	Yes	String	The name of the field to display on the inner ring.	
maxLegendEntries	No	Integer	The maximum number of values to display in the legend, for each tier. If the search returns more than the specified number of values for the field, colors and legend entries are assigned to the values with the largest contribution, and the remaining values are displayed in white.	5
secondField	No	String	The name of the field to display on the outer ring. If you do not set this property, the widget displays only a single ring.	

Example widget configuration:

```
{
  "name": "Sunburst",
  "type": "SunburstWidget",
  "x": 2,
```



```

    "y": 2,
    "width": 1,
    "height": 1,
    "datasource": {
      "source": "SavedSearch",
      "config": {
        "type": "QUERY",
        "id": "1234"
      }
    },
    "widgetSettings": {
      "firstField": "category",
      "secondField": "person"
    }
  }
}

```

Trending Widget

For the Trending widget, set `type` to **TrendingWidget**.

This widget type requires a saved search. You must set the `source` property to **SavedSearch** in the widget configuration.

The following table describes the options that you can use in the `widgetSettings` object.

Property	Required	Type	Description	Default
<code>parametricField</code>	Yes	String	The name of the field to display in the trending visualization.	
<code>dateField</code>	No	String	The name of the date field to use to find the date associated with particular values in the trending visualization.	AUTN_ DATE
<code>numberOfBuckets</code>	No	Integer	The number of points to include on the trending lines. Each point represents the document rate for a particular time range (bucket). The number of buckets controls how many buckets to split the full time range into.	20
<code>maxValues</code>	No	Integer	The maximum number of top values to display from the specified parametric field. If you define the <code>values</code> object for this widget, it overrides the <code>maxValues</code> setting.	10
<code>maxDate</code>	No	String	The maximum date to include in the visualization. Specify a date in the format <code>YYYY-MM-DDT hh:mm:ssZ</code> . This value becomes the maximum value on the x-axis. By default, the visualizer displays the maximum date that occurs in the documents	unlimited

Property	Required	Type	Description	Default
			with the specified parametric field.	
minDate	No	String	The minimum date to include in the visualization. Specify a date in the format YYYY-MM-DDT hh:mm:ssZ. This value becomes the minimum value on the x-axis. By default, the visualizer displays the minimum date that occurs in the documents with the specified parametric field.	unlimited
values	No	Object	<p>A JSON object that specifies particular values that you want to display in the visualization. If you define values, the visualization includes only the specified values, regardless of <code>maxValues</code>.</p> <p>This object contains the following properties:</p> <ul style="list-style-type: none"> • <code>name</code> (string). Required. The field value to display in the visualization. • <code>color</code> (string). Optional. The color to use for the trending line for this field value. You can use the values blue, light-blue, orange, pink, light-pink, green, light-green, red, purple, and yellow. If you do not specify a color, Find assigns colors at random. 	

Example widget configuration:

```
{
  "name": "Trending",
  "type": "TrendingWidget",
  "x": 0,
  "y": 0,
  "width": 2,
  "height": 2,
  "datasource": {
    "source": "SavedSearch",
    "config": {
      "type": "QUERY",
      "id": "1234"
    }
  },
  "widgetSettings": {
    "parametricField": "OVERALL_VIBE",
    "dateField": "AUTN_DATE",
    "numberOfBuckets": 25,
    "values": [
      {
```

```

        "name": "POSITIVE",
        "color": "purple"
      },
      {
        "name": "NEGATIVE",
        "color": "orange"
      }
    ]
  }
}

```

Map Widget

For the Map widget, set type to **MapWidget**.

NOTE:

Map widgets display correctly only if you configure the map visualizers. See [Configure Map Visualizations, on page 26](#).

This widget type requires a saved search. You must set the `source` property to **SavedSearch** in the widget configuration.

The following table describes the options that you can use in the `widgetSettings` object.

Property	Required	Type	Description	Default
<code>centerCoordinates</code>	Yes	Object	The coordinates to center the map on. This object has the following properties: <ul style="list-style-type: none"> <code>latitude</code> (number). The latitude coordinate to use as the center of the map. <code>longitude</code> (number). The longitude coordinate to use as the center of the map. 	
<code>locationFieldPairs</code>	Yes	String Array	The names of the configured field pairs to use as the location for the result markers. You find the names to use here in the <code>displayName</code> property of the <code>maps</code> section in the <code>config.json</code> configuration file.	
<code>zoomLevel</code>	Yes	Number	The level of zoom to use for the map. The value that you use for this option depends on your map tile server.	
<code>clusterMarkers</code>	No	Boolean	Set to true to cluster the result markers.	false
<code>maxResults</code>	No	Integer	The maximum number of search results to display on the map.	1000

Example widget configuration:

```

{
  "name": "Map",
  "type": "MapWidget",
  "x": 1,
  "y": 1,
  "width": 2,
  "height": 2,
  "datasource": {
    "source": "SavedSearch",
    "config": {
      "type": "QUERY",
      "id": "1234"
    }
  },
  "widgetSettings": {
    "maxResults": 10000,
    "locationFieldPairs": ["DefaultLocation", "OtherLocation"],
    "centerCoordinates": {
      "latitude": 51.5,
      "longitude": 0.12
    },
    "zoomLevel": 3,
    "clusterMarkers": true
  }
}

```

Video Panel Widget

For the Video Panel widget, set `type` to **VideoWidget**.

This widget type requires a saved search. You must set the `source` property to **SavedSearch** in the widget configuration.

The following table describes the options that you can use in the `widgetSettings` object.

Property	Required	Type	Description	Default
audio	No	Boolean	Set to true to play the audio for the video.	false
crossOrigin	No	String	The CORS settings to use (anonymous or use-credentials). You might need this option to enable Find to export video as an image in a PowerPoint report, if you host videos on a different server to Find. Use this option only if the server you use to host your videos supports CORS headers. You do not need this setting if your video is hosted on the same site as Find (including by reverse-proxy), or if you do not want to export video dashboards.	

Property	Required	Type	Description	Default
loop	No	Boolean	Set to false if you do not want to replay the video when it finishes. In this case, the video stops after it reaches the end and does not replay until the widget refreshes. By default, the video loops.	true
restrictSearch	No	Boolean	Set to true to restrict the saved search to return only results with the content type field value video .	false
searchResultNumber	No	Integer	The number of the result that you want to play (for example, 2 to play the second result in the list).	1

NOTE:

The widget displays a result only if it is type video. To ensure that the results include only videos, set `restrictSearch` to **true**.

Alternatively, you can set `searchResultNumber` to display a specific video result (for example, if you want to display a particular result from a snapshot saved search).

Example widget configuration:

```
{
  "name": "Video Widget",
  "type": "Videowidget",
  "x": 1,
  "y": 1,
  "width": 2,
  "height": 2,
  "datasource": {
    "source": "SavedSearch",
    "config": {
      "type": "QUERY",
      "id": "1234"
    }
  },
  "widgetSettings": {
    "restrictSearch": true
  }
}
```

Static Content Widget

For the Static Content widget, set `type` to **StaticContentWidget**.

You cannot use a saved search for this widget type, because its content is independent from any search.

The following table describes the options that you can use in the `widgetSettings` object.

Property	Required	Type	Description	Default
html	Yes	String	The HTML to display in the widget.	

Example widget configuration:

```
{
  "name": "Static Content",
  "type": "StaticContentWidget",
  "x": 1,
  "y": 1,
  "width": 2,
  "height": 2,
  "widgetSettings": {
    "html": "<h1>Hello World!</h1>"
  }
}
```

Static Image Widget

For the Static Image widget, set type to **StaticImageWidget**.

You cannot use a saved search for this widget type, because its content is independent from any search.

The following table describes the options that you can use in the `widgetSettings` object.

Property	Required	Type	Description	Default
url	Yes	String	The URL of the image to display in the widget.	

Example widget configuration:

```
{
  "name": "Static Image",
  "type": "StaticImageWidget",
  "x": 1,
  "y": 1,
  "width": 2,
  "height": 2,
  "widgetSettings": {
    "url": "http://example.com/image.png"
  }
}
```

Current Time/Date Widget

For the Current Time/Date widget, set type to **CurrentTimeWidget**.

You cannot use a saved search for this widget type, because its content is independent from any search.

The following table describes the options that you can use in the `widgetSettings` object.

NOTE:

The current time/date widget uses the `moment.js` library to format dates. For more information about the available formats, refer to the `moment.js` documentation.

Property	Required	Type	Description	Default
<code>dateFormat</code>	No	String	The date format to display. This option takes a <code>moment.js</code> date format string.	ll
<code>timeFormat</code>	No	String	The time format to display. This option takes a <code>moment.js</code> time format string.	HH:mm Z
<code>timeZone</code>	No	String	The timezone to use in the date/time display. This option takes a <code>moment.js</code> timezone string. By default, the <code>moment.js</code> library attempts to display the local timezone.	

Example widget configuration:

```
{
  "name": "Current Time Date",
  "type": "CurrentTimeWidget",
  "x": 1,
  "y": 1,
  "width": 2,
  "height": 2,
  "widgetSettings": {
    "timeFormat": "HH:mm:ss Z"
  }
}
```

Time Last Refreshed Widget

For the Time Last Refreshed widget, set `type` to **TimeLastRefreshedWidget**.

You cannot use a saved search for this widget type, because its content is independent from any search.

The following table describes the options that you can use in the `widgetSettings` object.

NOTE:

The time last refreshed widget uses the `moment.js` library to format times. For more information about the available formats, refer to the `moment.js` documentation.

Property	Required	Type	Description	Default
<code>dateFormat</code>	No	String	The date format to display. This option takes a <code>moment.js</code> date format string.	HH:mm Z
<code>timeZone</code>	No	String	The timezone to use in the date/time display. This option takes a <code>moment.js</code> timezone string. By default,	

Property	Required	Type	Description	Default
			the <code>moment.js</code> library attempts to display the local timezone.	

Example widget configuration:

```
{
  "name": "Time Last Refereshed",
  "type": "TimeLastRefreshedWidget",
  "x": 1,
  "y": 1,
  "width": 1,
  "height": 1
}
```

Dashboard Export Notes

The Find dashboard user interface includes an option to export your dashboards to a Microsoft PowerPoint presentation Open XML format (.pptx).

For the **Video** and **Static Image** widgets, users can export the widgets only if you serve the videos or images from a domain that their browser is allowed to access by the CORS policy.

For the **Sunburst** widget, users can export only a single ring of the sunburst (if they attempt to export a two-level sunburst visualization, the presentation includes only the inner ring).

Example Dashboard Configuration

The following example JSON has a configuration for four example dashboards.

NOTE:

This example configuration might not work directly in your Find without modification. For example, to use these dashboards, you must have a `QUERY` type saved search with ID 1, and you must have appropriate fields configured.

```
{
  "dashboards": [
    {
      "dashboardName": "Single widget dashboard",
      "enabled": true,
      "width": "1",
      "height": "1",
      "displayWidgetNames": "never",
      "widgets": [
        {
          "name": "Static Content",
          "type": "StaticContentWidget",
          "x": 0,
```



```

        "y": 0,
        "width": 1,
        "height": 1,
        "widgetSettings": {
            "html": "<h1>Welcome to the Find Dashboard</h1>"
        }
    }
]
},
{
    "dashboardName": "3 widget dashboard",
    "enabled": true,
    "width": "4",
    "height": "4",
    "updateInterval": 1000,
    "displayWidgetNames": "onhover",
    "widgets": [
        {
            "name": "Sunburst",
            "type": "SunburstWidget",
            "x": 0,
            "y": 0,
            "height": 2,
            "width": 2,
            "datasource": {
                "source": "SavedSearch",
                "config": {
                    "id": 1,
                    "type": "QUERY"
                }
            },
            "widgetSettings": {
                "firstField": "COMPANY",
                "secondField": "OVERALL_VIBE"
            }
        },
        {
            "name": "Trending",
            "type": "TrendingWidget",
            "x": 0,
            "y": 2,
            "height": 2,
            "width": 2,
            "datasource": {
                "source": "SavedSearch",
                "config": {
                    "id": 1,
                    "type": "QUERY"
                }
            }
        }
    ]
}

```

```

    },
    "widgetSettings": {
      "parametricField": "OVERALL_VIBE",
      "dateField": "AUTN_DATE",
      "numberOfBuckets": 20,
      "minDate": "2016-04-02T00:00:00Z",
      "maxDate": "2016-04-05T00:00:00Z",
      "values": [
        {
          "name": "POSITIVE",
          "color": "purple"
        },
        {
          "name": "NEGATIVE",
          "color": "orange"
        }
      ]
    }
  },
  {
    "name": "Results List",
    "type": "ResultsListWidget",
    "x": 2,
    "y": 0,
    "height": 4,
    "width": 2,
    "datasource": {
      "source": "SavedSearch",
      "config": {
        "id": 1,
        "type": "QUERY"
      }
    },
    "widgetSettings": {
      "maxResults": 10
    }
  }
]
},
{
  "dashboardName": "Alternative 3 widget dashboard",
  "enabled": true,
  "width": "4",
  "height": "2",
  "displayWidgetNames": "always",
  "widgets": [
    {
      "name": "Static Image",
      "type": "StaticImageWidget",

```

```

    "x": 0,
    "y": 0,
    "width": 1,
    "height": 2,
    "widgetSettings": {
      "url": "http://example.com/image.png"
    }
  },
  {
    "name": "Video Panel Widget",
    "type": "VideoWidget",
    "x": 1,
    "y": 0,
    "width": 2,
    "height": 2,
    "datasource": {
      "source": "SavedSearch",
      "config": {
        "type": "QUERY",
        "id": "1"
      }
    },
    "widgetSettings": {
      "restrictSearch": true
    }
  },
  {
    "name": "Map Widget",
    "type": "MapWidget",
    "x": 3,
    "y": 0,
    "width": 1,
    "height": 2,
    "datasource": {
      "source": "SavedSearch",
      "config": {
        "type": "QUERY",
        "id": "1"
      }
    },
    "widgetSettings": {
      "locationFieldPairs": [
        "DefaultLocation"
      ],
      "centerCoordinates": {
        "latitude": 51.5,
        "longitude": 0.12
      },
      "zoomLevel": 3
    }
  }
]

```

```

        }
    }
]
},
{
    "dashboardName": "Multi-widget dashboard",
    "enabled": true,
    "width": "8",
    "height": "9",
    "displayWidgetNames": "always",
    "widgets": [
        {
            "name": "Trending",
            "type": "TrendingWidget",
            "x": 0,
            "y": 0,
            "width": 4,
            "height": 3,
            "datasource": {
                "source": "SavedSearch",
                "config": {
                    "type": "QUERY",
                    "id": "1"
                }
            },
            "widgetSettings": {
                "parametricField": "OVERALL_VIBE",
                "dateField": "AUTN_DATE",
                "numberOfBuckets": 15
            },
            "values": [
                {
                    "name": "POSITIVE",
                    "color": "red"
                }
            ]
        },
        {
            "name": "Sunburst",
            "type": "SunburstWidget",
            "x": 4,
            "y": 0,
            "width": 2,
            "height": 3,
            "datasource": {
                "source": "SavedSearch",
                "config": {
                    "type": "QUERY",
                    "id": "1"
                }
            }
        }
    ]
}

```

```

    }
  },
  "widgetSettings": {
    "firstField": "CATEGORY",
    "secondField": "COMPANY"
  }
},
{
  "name": "Sunburst",
  "type": "SunburstWidget",
  "x": 6,
  "y": 0,
  "width": 2,
  "height": 3,
  "datasource": {
    "source": "SavedSearch",
    "config": {
      "type": "QUERY",
      "id": "1"
    }
  },
  "widgetSettings": {
    "firstField": "COMPANY",
    "secondField": "OVERALL_VIBE"
  }
},
{
  "name": "Current Time/Date",
  "type": "CurrentTimeWidget",
  "x": 0,
  "y": 3,
  "width": 1,
  "height": 2,
  "widgetSettings": {
    "timeFormat": "HH:mm z"
  }
},
{
  "name": "Last Time Refreshed",
  "type": "TimeLastRefreshedWidget",
  "x": 1,
  "y": 3,
  "width": 1,
  "height": 2
},
{
  "name": "Static Content",
  "type": "StaticContentWidget",
  "x": 2,

```

```

        "y": 3,
        "width": 3,
        "height": 3,
        "widgetSettings": {
            "html": "<h1>Find Multi-Widget Dashboard</h1>"
        }
    },
    {
        "name": "Map",
        "type": "MapWidget",
        "x": 5,
        "y": 3,
        "width": 3,
        "height": 3,
        "datasource": {
            "source": "SavedSearch",
            "config": {
                "type": "QUERY",
                "id": "1"
            }
        },
        "widgetSettings": {
            "locationFieldPairs": [
                "DefaultLocation"
            ],
            "centerCoordinates": {
                "latitude": 51.5,
                "longitude": 0.12
            },
            "zoomLevel": 3
        }
    },
    {
        "name": "Video",
        "type": "VideoWidget",
        "x": 0,
        "y": 5,
        "width": 2,
        "height": 4,
        "datasource": {
            "source": "SavedSearch",
            "config": {
                "type": "QUERY",
                "id": "1"
            }
        },
        "widgetSettings": {
            "restrictSearch": true
        }
    }

```

```

    },
    {
      "name": "Results",
      "type": "ResultsListWidget",
      "x": 2,
      "y": 6,
      "width": 4,
      "height": 3,
      "datasource": {
        "source": "SavedSearch",
        "config": {
          "type": "QUERY",
          "id": "1"
        }
      },
      "widgetSettings": {
        "maxResults": 10
      }
    },
    {
      "name": "Topic Map",
      "type": "TopicMapWidget",
      "x": 6,
      "y": 6,
      "width": 2,
      "height": 3,
      "datasource": {
        "source": "SavedSearch",
        "config": {
          "type": "QUERY",
          "id": "1"
        }
      }
    }
  ]
}

```

Chapter 8: Configure Applications

This section describes how to configure Find to include other applications in the navigation sidebar.

- [Configure Application Links in Find](#)72

Configure Application Links in Find

You can configure Find to list some external applications in the navigation sidebar. For example, you might want to link to administrative user interfaces to allow your users to quickly access those applications, or you can link to other applications that are important to your organization.

You configure application links by using the `applications.json` configuration file, in the `customization` directory in your Find installation home directory. This configuration file contains the `applications` object, which is an array of application configuration objects. To add an application, you add an object to this array.

In the Find user interface, the **Applications** section in the navigation sidebar lists your applications in the same order that you list them in the configuration file.

When you configure an application, you set the name and URL to use for each application, and whether to open the application link in a new Web browser tab, or to use the current one.

TIP:

If you belong to the **FindAdmin** role, you can reload the application configuration to make any configuration changes available to your users without restarting Find.

To reload the application configuration, you must send the following API call to Find directly, by typing the URL into your Web browser address bar. You must be logged into Find as a user with the **FindAdmin** role in the same Web browser, because Find uses the session cookie to authorize the action.

```
http://FindURL/api/admin/customization/config/reload
```

Where *FindURL* is the URL of your Find server.

This option also reloads any other customization configurations (for example, dashboards).

The following table lists the configuration properties that you can use for each application.

Property	Required	Type	Description
<code>applicationName</code>	Yes	String	The name to use for the application. This name is displayed in the Applications navigation section.
<code>url</code>	Yes	String	The URL to navigate to when a user clicks this menu item. This value must be a well-formed URL.
<code>openInNewTab</code>	No	Boolean	Set to true to open application links in a new Web browser tab. Set to false to open the link in the current tab. The default value is false .

Property	Required	Type	Description
enabled	No	Boolean	Set to true to enable the application link and display it in the Applications navigation section. Set to false to hide the application link. The default value is true .

The following example shows a simple configuration for three applications:

```
{
  "applications": [
    {
      "applicationName": "IDOL Data Admin",
      "url": "http://dataadmin.example.com:8080"
    },
    {
      "applicationName": "IDOL Admin (Content)",
      "url": "http://ida-content.example.com:9000/a=admin",
      "openInNewTab": true,
      "enabled": false
    },
    {
      "applicationName": "IDOL Admin (Community)",
      "url": "http://ida-community.example.com:9030/a=admin",
      "enabled": true
    }
  ]
}
```

For this example, the IDOL Data Admin link and the IDOL Admin (Community) link are listed in the Application list in Find. The IDOL Admin (Content) link is not enabled, and so it is not displayed in Find.

Chapter 9: Configure Templates

This section describes how to configure Find to use additional templates for the results list and preview pane.

- [Configure Templates in Find](#) 74
- [Create Template Files](#) 75

Configure Templates in Find

You can configure Find to use a custom template to display results and promotions in the search results list, and metadata in the preview panel. For example, you might want to use a larger thumbnail image and title in the result list, and to remove the summary.

You configure how Find uses the custom templates by using the `templates.json` configuration file, in the `customization` directory in your Find installation home directory. This configuration file contains JSON objects that allow you to configure different aspects of the Find results.

TIP:

If you belong to the **FindAdmin** role, you can reload the template configuration to make any configuration changes available to your users without restarting Find.

To reload the template configuration, you must send the following API call to Find directly, by typing the URL into your Web browser address bar. You must be logged into Find as a user with the **FindAdmin** role in the same Web browser, because Find uses the session cookie to authorize the action.

`http://FindURL/api/admin/customization/config/reload`

Where *FindURL* is the URL of your Find server.

This option also reloads any other customization configurations (for example, dashboards).

The following table describes the configuration objects that you can include in your template configuration file. These objects are all optional.

Object	Type	Description
<code>searchResult</code>	Array	Defines the templates to use for search result items in the results list.
<code>promotion</code>	Array	Defines the templates to use for promotion items in the results list.
<code>previewPanel</code>	Array	Defines the templates to use for document metadata in the preview panel.
<code>entitySearch</code>	Array	Defines the templates to use for the entity search dialog box.

In each of these objects, you specify an array of template configurations, each of which specifies the template file to use and the circumstances in which to use the template.

The following table lists the configuration properties that you can use to specify a template.

Property	Required	Type	Description
file	Yes	String	The template file name. See Create Template Files, below .
triggers	Yes	Object Array	<p>The triggers that specify when to use a template. This object contains the following properties:</p> <ul style="list-style-type: none">• <code>field</code>. The field ID (the key in the <code>fieldsInfo</code> section of the Find configuration file).• <code>values</code>. An array of field values that the specified field must contain for the trigger to activate. <p>A document activates the trigger if it contains at least one of the specified values in the specified field. If the list of values is empty, a document activates the trigger if it contains any value in the specified field.</p> <p>You can specify multiple field conditions in the trigger. In this case, a document must match all the specified field conditions to activate the trigger.</p>

When displaying results, Find processes the templates in the order that you list them in the configuration file. For a particular document, Find uses the first template that the document matches. If there are no matching templates in your configuration, Find displays the result with the application default template.

The following example shows a simple search result configuration that applies to documents that contain the value **Living people** in the **categories** field.

```
{
  "searchResult": [
    {
      "file": "person.handlebars",
      "triggers": [
        {
          "field": "categories",
          "values": ["Living people"]
        }
      ]
    }
  ]
}
```

Create Template Files

Find processes templates by using `handlebars.js` (for more information, see <http://handlebarsjs.com/>).

Each template file must produce HTML output that represents one document.

Find runs the templates with the following Handlebars context:

```
interface ResultTemplateData {
    reference: string;
    title: string;
    date: string;
    database: string;
    promotionName: string|undefined;
    summary: string; // The highlighted summary, should not be HTML escaped
    url: string|undefined; // URL of the original document or media file
    icon: string; // Icon class based on content type
    similarDocumentsUrl: string|undefined; // URL for linking to the similar
documents view, only in result and promotion
    thumbnailSrc: string|undefined, // Source attribute to load the thumbnail in an
<img> tag
    age: string; // Internationalised age of the document (e.g. "3 years ago")
    fields: {id: string, values: string[], displayName: string, advanced: boolean}
[];
}
```

The following table also describes some Find custom helpers that you can use in your templates.

Helper	Description
equal	A block helper that takes two or more arguments. The block is printed only if the first argument is referentially equal to any of the subsequent arguments.
hasField	A block helper that takes one string argument (the field). The block is printed only if the document has a value for the field.
hasFieldValue	A block helper that takes two string arguments, the field and the value. The block is printed only if the document contains the specified value in the specified field.
getFieldValue	Prints the first value for the specified field.
getFieldValues	Prints all values for the specified field. You can optionally add the following parameters: <ul style="list-style-type: none"> • <code>max</code>. The maximum number of values to display. For example, <code>max=2</code>. By default, the helper prints all available values. • <code>delimiter</code>. The delimiter to use between field values. For example, <code>delimiter=' '</code>. The default value is <code>' , '</code> (a comma with a space). • <code>ellipsis</code>. The string to use after the final value, when there are more than the specified maximum. For example, <code>ellipsis=', and more.'</code>. The default value is <code>....</code>. You can set <code>ellipsis</code> to <code>' '</code> to turn off the ellipsis.
withField	A block helper that executes the block in the context of the given field.
i18n	Prints a string from the application internationalization file.
percentage	Formats the specified number as a percentage. You can optionally add the following parameters:

	<ul style="list-style-type: none"> • <code>min</code>. Min. • <code>max</code>. Max. • <code>round</code>. The option to use for rounding the value. This parameter accepts the following values: <ul style="list-style-type: none"> ◦ <code>round</code>. Use standard rounding rules. ◦ <code>ceil</code>. Round up to the next integer. ◦ <code>floor</code>. Round down to the next integer. ◦ <code>N</code>. A number of decimal places to round to. ◦ <code>false</code>. Disable rounding.
<code>placeholderTemplate</code>	<p>Fills in a template that contains placeholder values, by using variables that you specify. This helper takes two arguments. The first argument is a hash, array, or string that defines the values to use in the specified template. The second argument is the template string that contains placeholder values. You can also optionally add the following parameters:</p> <ul style="list-style-type: none"> • <code>placeholder</code>. The symbol to use to specify placeholders. Each placeholder value must have this placeholder symbol at the start and end of the placeholder. The default symbol is a dollar sign (\$). For example, <code>\$username\$</code> or <code>\$0\$</code>. You can repeat the placeholder symbol to escape it. For example, <code>\$\$100</code> represents \$100. • <code>delimiter</code>. The delimiter to use to split the value when the value argument contains a string. The default value is a comma (,) <p>The placeholder value is a key that links to the values in your hash, array, or string. If you use a hash, you can use string keys. If you use an array or string value, use a zero-indexed numeric key that specifies which of the listed values to use.</p> <p>For example: <code>{{ placeholderTemplate 'liftoff!,one,two,three' 'countdown: \$3\$, \$2\$, \$1\$, and \$0\$' placeholder='\$' delimiter=', ' }}</code> generates the string <i>countdown: three, two, one, and liftoff!</i>.</p>
<code>prettyPrintNumber</code>	<p>Pretty-prints the specified number (for example, it converts 1000 to 1,000). You can optionally add the following parameters:</p> <ul style="list-style-type: none"> • <code>delimiter</code>. The thousands delimiter to use. The default value is a comma (,) • <code>round</code>. The option to use for rounding the value. This parameter accepts the following values: <ul style="list-style-type: none"> ◦ <code>round</code>. Use standard rounding rules. ◦ <code>ceil</code>. Round up to the next integer. ◦ <code>floor</code>. Round down to the next integer. ◦ <code>N</code>. A number of decimal places to round to. ◦ <code>false</code>. Disable rounding. This value is the default. • <code>plusIfPositive</code>. Set to true to add a plus symbol (+) in front of positive numbers (for example +8). The default value is false.

toExternalUrl	Adds a URL schema prefix to the specified string, if it does not already have one. You can optionally add the following parameter: <ul style="list-style-type: none"> defaultProtocol. The prefix to use. The default value is http://.
toLowerCase	Converts the specified string to lower case characters.
toUpperCase	Converts the specified string to upper case characters.
toRelativeTime	Prints a specified date string as a relative time (for example, <i>50 years ago</i>). You can optionally add the following parameters: <ul style="list-style-type: none"> dateFormat. The date format of the input string argument. This option takes a <code>moment.js</code> date format string. The default format is DD-MM-YYYY. For more information, refer to the <code>moment.js</code> documentation. hideSuffix. Set to true to hide the suffix (for example, to print <i>50 years</i> rather than <i>50 years ago</i>). The default value is false. round. The option to use for rounding the value. This parameter accepts the following values: <ul style="list-style-type: none"> round. Use standard mathematical rounding rules. This value is the default. ceil. Round up to the next integer. floor. Round down to the next integer. N. A number of decimal places to round to.
wikiThumbnailHelper	Converts a Wikipedia Commons image URL to one with a fixed thumbnail size. This helper takes two arguments. The first argument is the URL, and the second is the maximum thumbnail width (in pixels).

You reference fields in custom helpers by using the field ID (that is, the corresponding key in the `fieldsInfo` section of the Find configuration file). Document fields are available only if you explicitly reference them in that configuration section.

Example Template

The following template is a simple search result template that displays a thumbnail image for documents that contain a `thumbnail` field. It also displays the document summary, an `Author` field, and a link to get similar documents.

```
<div>
  <h1><i class="{{icon}}"></i>{{title}}</h1>
  {{#hasField 'thumbnail'}}
    
  {{/hasField}}
  <p>{{summary}}</p>
  <p>Author: {{getFieldValue "Author"}}</p>
  <a href="{{similarDocumentsUrl}}">See similar documents</a>
</div>
```


Chapter 10: Configure Find with System Properties

This section describes the features that you can configure by using system properties.

• Configure the Log File Locations	80
• Configure IDOL Logs	81
• Configure Cache Size	82
• Configure Default User Roles	83
• Configure Find to Use SSL	84
• Configure Session Timeout	85
• Rename the Session Cookie	85
• Configure Find with a Reverse Proxy	86
• Configure Find to Use a Single Pre-Authenticated User	90

Configure the Log File Locations

By default, Find creates a folder of log files under the home directory. You can configure Find to store these log files in a different location by setting the `logging.path` property in the java run command.

To change the location of the log files

1. At the command line, send the java run command with the `logging.path` argument set to the location where you want to store your log files. For example:

```
java -Dlogging.path=[log file directory] -Didol.find.home=[home directory] -
Dserver.port=[port] -jar find.war
```

2. Restart Find to apply your configuration changes.

Find creates the specified directory if it does not exist, as long as the service has the appropriate permissions. It creates the logs directly in the specified directory.

If you run Find as a service on Windows, you can also add the `logging.path` property to the `find.xml` file.

If you run Find as a service on Linux, you can modify the `FIND_LOGGING_DIR` variable in the start scripts (`find.sh` for SystemV, or `find.conf` for Upstart).

When you modify the Find configuration by modifying the XML file or start scripts, you must restart Find to apply your configuration changes.

Advanced Log Configuration

In the default configuration, Find produces a new log file every day.

Find uses a time-based rolling policy as part of a standard Logback configuration to create logs (see <https://logback.qos.ch/index.html>). If required, you can modify the Find logging configuration by updating the `logback-spring.xml` configuration file inside the Find war file.

To find the logback-spring XML configuration file

1. Open the `find.war` file as a ZIP archive.
2. Navigate to the `/WEB-INF/lib/` folder and find a JAR file with a name of the form `core-*.jar`, where `*` is a string that includes the Find version number.
3. Open the `core-*.jar` as a ZIP archive and extract the `logback-spring.xml` file.

For information about the settings that you can use to modify the way that Find generates the log file, refer to the Logback documentation. For example, the following link provides information about options that you can add to the existing log

format: <https://logback.qos.ch/manual/appenders.html#TimeBasedRollingPolicy>.

When you have made your modifications, you can update the version of the `logback-spring.xml` file in the `core-*.jar` file, and then update this file in the `find.war` archive.

Tomcat Access Log Configuration

In addition to the Find log files, there is also a Tomcat access log, `tomcat-access.log`. This uses the same logging mechanism as the other logs, and you can configure the details by using the `logback-spring.xml` file (see [Advanced Log Configuration, on the previous page](#)).

You can set the logging pattern to use by using the `server.tomcat.accesslog.pattern` system property when you start Find. This option uses the standard Apache log format. For more information on the patterns that you can use in the log string, refer to the Apache documentation at

<https://tomcat.apache.org/tomcat-8.0-doc/api/org/apache/catalina/valves/AbstractAccessLogValve.html>.

For example:

```
java -Dserver.tomcat.accesslog.pattern=combined -Didol.find.home=[home directory] -
Dserver.port=[port] -jar find.war
```

NOTE:

In Find 11.5 and earlier, the Tomcat access log was called `access.log` and used a different logging configuration. If you upgrade to Find 11.6 from 11.5 or earlier, you must remove the `server.tomcat.accesslog.enabled=true` from your system properties.

If you previously disabled Tomcat access logging by setting `server.tomcat.accesslog.enabled=false`, you disable the new version by setting `server.tomcat.accesslog.pattern` to an empty string (that is, `server.tomcat.accesslog.pattern=`).

Configure IDOL Logs

By default, Find generates an `idol-access.log` file, and uses it to log all the actions it makes to IDOL component ACI and service ports. The log information includes:

- the user that sent the query.
- the IP address the user is connecting from.
- the IDOL component that the query is sent to (for example, QMS or Content)

- the host and port of the IDOL component.
- the full query that is sent to the IDOL component.
- the status code that the IDOL component returns.
- the time taken to complete the request.

You can turn off IDOL logging by setting the `idol.log.enabled` system property. You can also turn off just the information about time taken to complete requests by setting the `idol.log.timing.enabled` system property.

To turn off IDOL logging

1. At the command line, send the java run command with the `idol.log.enabled` argument set to **false**. For example:

```
java -Didol.log.enabled=false -Didol.find.home=[home directory] -Dserver.port=[port] -jar find.war
```

2. Restart Find to apply your configuration changes.

To turn off timing information in the IDOL logs

1. At the command line, send the java run command with the `idol.log.timing.enabled` argument set to **false**. For example:

```
java -Didol.log.timing.enabled=false -Didol.find.home=[home directory] -Dserver.port=[port] -jar find.war
```

2. Restart Find to apply your configuration changes.

If you run Find as a service on Windows, you can also add the `idol.log.enabled` and `idol.log.timing.enabled` properties to the `find.xml` file.

If you run Find as a service on Linux, you can add the argument to the existing arguments variable in `find.sh` for SysV, or `find.conf` for Upstart.

When you modify the Find configuration by modifying the XML file or start scripts, you must restart Find to apply your configuration changes.

Configure Cache Size

If you see a Catalina cache warning in your log files when you run Find, you can use the `server.tomcat.resources.max-cache-kb` system property to increase the size of the cache.

The cache warning might appear similar to the following example:

```
org.apache.catalina.webresources.Cache.getResource Unable to add the resource at [/WEB-INF/Find/example.jpg] to the cache because there was insufficient free space available after evicting expired cache entries - consider increasing the maximum size of the cache
```

The maximum size of the cache is set in kilobytes. The default value is 20480 (20 MB).

To set the cache max size property

- Start Find with the Java run command, and include the `server.tomcat.resources.max-cache-kb` argument. For example:

```
java -Dserver.tomcat.resources.max-cache-kb=30480 -Didol.find.home=[home
directory] -Dserver.port=[port] -jar find.war -uriEncoding utf-8
```

If you run Find as a service on Windows, you can also add this argument to the `find.xml` file.

If you run Find as a service on Linux, you can add the argument to the existing arguments variable in `find.sh` for SysV, or `find.conf` for Upstart.

When you modify the Find configuration by modifying the XML file or start scripts, you must restart Find to apply your configuration changes.

For more information on installing and running Find as a service on Windows or Linux, see [Install Find, on page 15](#).

Configure Default User Roles

You can configure Find to set some default roles for users that do not have a valid Find role in the IDOL Community component.

You can use this option if you are using an IDOL Community component that authenticates by using a third-party repository (such as LDAP). In this case, the third-party repository handles the authentication, and the IDOL Community component creates the Find user when they log in for the first time.

When you define default roles, Find applies the configured default roles to the new user when they log in for the first time.

To configure default user roles in Find

- Start Find with the Java run command, and include the following arguments:

```
-Dfind.defaultRoles=list_of_roles
```

where *list_of_roles* is a comma-separated list of the user roles that you want to assign by default to new users.

For example:

```
java -Dfind.defaultRoles=[Users] -Didol.find.home=[home directory] -Dserver.port=
[port] -jar find.war -uriEncoding utf-8
```

If you run Find as a service on Windows, you can also add the same arguments to the `find.xml` file.

If you run Find as a service on Linux, you can add the argument to the existing arguments variable in `find.sh` for SysV, or `find.conf` for Upstart.

When you modify the Find configuration by modifying the XML file or start scripts, you must restart Find to apply your configuration changes.

For more information on installing and running Find as a service on Windows or Linux, see [Install Find, on page 15](#).

Configure Find to Use SSL

By default, Find runs in plain HTTP mode. You can configure it to use SSL communications, by setting the appropriate system properties in the Java run command.

NOTE:
If you enable SSL, you cannot use plain HTTP with Find.

To configure Find to use SSL

- Start Find with the Java run command, and include the following arguments:

```
-Dserver.ssl.key-store=path_to_keystore
-Dserver.ssl.key-store-password=keystore_password
-Dserver.ssl.key-password=key_password
-Dserver.port=server_port
```

where,

<i>path_to_keystore</i>	is the path to a keystore. You must create a keystore and import your certificates into it. By default, Find expects this keystore to be in JKS format. For more information on JKS keystore creation, see https://docs.oracle.com/cd/E19509-01/820-3503/ggfen/index.html . <div>NOTE: If you want to use a keystore in a different format, you must also set the -Dserver.ssl.key-store-type argument to the type of keystore that you want to use. For example: -Dserver.ssl.key-store-type=pkcs12</div>
<i>keystore_password</i>	is the password for the keystore.
<i>key_password</i>	is the password for the key inside the keystore.
<i>server_port</i>	is the port to user for Find. To use SSL, you must change this value from the default port (that is, you must not use port 8080). Typically you use port 8443, but you can use any valid port value.

For example:

```
java -Dserver.ssl.key-store=/etc/ssl/private/keystore.jks -Dserver.ssl.key-store-
password=MyPassword -Dserver.ssl.key-password=KeyPassword -Didol.find.home=[home
directory] -Dserver.port=8443 -jar find.war -uriEncoding utf-8
```

If you run Find as a service on Windows, you can also add the same arguments to the find.xml file.

If you run Find as a service on Linux, you can add the argument to the existing arguments variable in find.sh for SysV, or find.conf for Upstart.

When you modify the Find configuration by modifying the XML file or start scripts, you must restart Find to apply your configuration changes.

For more information on installing and running Find as a service on Windows or Linux, see [Install Find, on page 15](#).

Configure Session Timeout

By default, a user session can remain idle for one hour before Find terminates it automatically. You can edit this setting by setting the `server.session.timeout` system property to the number of seconds after which sessions should end. For example, `server.session.timeout=7200` sets a session timeout of 2 hours.

NOTE:

The session timeout does not apply when a user opens a dashboard in full screen mode. See [Find Dashboards, on page 48](#).

To set the session timeout property

- Start Find with the Java run command, and include the `server.session.timeout` argument. For example:

```
java -Dserver.session.timeout=3600 -Didol.find.home=[home directory] -  
Dserver.port=[port] -jar find.war -uriEncoding utf-8
```

If you run Find as a service on Windows, you can also add the same `-Dserver.session.timeout` argument to the `find.xml` file.

If you run Find as a service in Linux, you can add the argument to the existing arguments variable in `find.sh` for SysV, or `find.conf` for Upstart.

When you modify the Find configuration by modifying the XML file or start scripts, you must restart Find to apply your configuration changes.

For more information on installing and running Find as a service on Windows or Linux, see [Install Find, on page 15](#).

NOTE:

Polling for saved searches counts as user activity, and delays a session timeout for the user. If your polling interval is shorter than your session timeout, the polling causes the session to continue until you close the browser window. See [Enable Polling for Saved Searches, on page 27](#).

Rename the Session Cookie

By default, Find uses the session cookie `FINDSESSIONID` to store session details. You can optionally rename the session cookie, for example if you want to run multiple instances of Find on the same domain and keep the session details separate.

To rename the session cookie

- Start Find with the Java run command, and include the following arguments:

```
-Dorg.apache.catalina.SESSION_COOKIE_NAME=MYNEWSESSIONNAME
```

where *MYNEWSESSIONNAME* is the new session cookie that you want to use.

For example:

```
java -Dorg.apache.catalina.SESSION_COOKIE_NAME=FIND1SESSIONID -Didol.find.home=[home directory] -Dserver.port=[port] -jar find.war -uriEncoding utf-8
```

If you run Find as a service on Windows, you can also add the same arguments to the `find.xml` file.

If you run Find as a service on Linux, you can add the argument to the existing arguments variable in `find.sh` for SysV, or `find.conf` for Upstart.

When you modify the Find configuration by modifying the XML file or start scripts, you must restart Find to apply your configuration changes.

For more information on installing and running Find as a service on Windows or Linux, see [Install Find, on page 15](#).

Configure Find with a Reverse Proxy

You can use Find with a reverse proxy over either HTTP(S) or Apache JServ Protocol (AJP).

With a reverse proxy, you can choose to manage authentication for your users, and the reverse proxy allows users to access the Find application with a default set of user roles, which you can define.

NOTE:

You cannot configure the reverse proxy to assign an administrator user role. Your administrators must have access to the Find application by another route.

Using a reverse proxy can allow you to:

- add security and protect the identity of your backend services.
- pre-authenticate your users, enabling a single sign-on (SSO) workflow.
- provide SSL termination.
- improve performance by adding compression or caching.

The following sections describe how to configure Find to use a reverse proxy, and how to configure various types of reverse proxy.

Configure Find to Use the Reverse Proxy

You configure Find to use a reverse proxy by setting the `server.reverseProxy` system property.

When you set `server.reverseProxy` to **true**, the Find application listens for AJP connections on a port that you can define by using the `server.ajp.port` system property.

The reverse proxy must translate incoming HTTP(S) connections from your clients into AJP connections to Find. In AJP mode, the application reads the user name from incoming requests, so you

must set up remote authentication. You cannot encrypt the connection from the reverse proxy to the application.

By default, `server.reverseProxy` is **false**. In this case, the Find application is available only over HTTP or HTTPS. Remote authentication is not possible in this mode. However, you can encrypt the connection with SSL. In this mode, users authenticate by using the proxied login page.

In both these methods, the reverse proxy must rewrite the request path, response Location headers, and cookie paths.

Set the Reverse Proxy System Property

You can use the following procedure to set the `server.reverseProxy` system property.

To configure Find to use a reverse proxy

1. At the command line, send the java run command with the `server.reverseProxy` argument set to **true**. For example:

```
java -Dserver.reverseProxy=true -Didol.find.home=[home directory] -
Dserver.port=[port] -jar find.war
```

By default, this option opens an AJP port on port 8009. To use a different port, set the `server.ajp.port` argument to the correct port number.

You can also specify the roles to assign to your users by setting the `find.reverse-proxy.pre-authenticated-roles` system property to a comma-separated list of Find Community role names (see [User Roles, on page 20](#)). The default value for this property is `FindUser`.

NOTE:

This setting does not assign any roles in the Community component. It gives all users that use the reverse proxy the same permissions to access Find.

Another option is to set `find.reverse-proxy.pre-authenticated-username` to a specific user name. This option bypasses authentication so that any user that accesses Find is logged in automatically as that user name. See [Configure Find to Use a Single Pre-Authenticated User, on page 90](#).

2. Restart Find to apply your configuration changes.

If you run Find as a service on Windows, you can also add the `server.reverseProxy` property to the `find.xml` file.

If you run Find as a service on Linux, you can add the arguments to the existing arguments variable in the start scripts (`find.sh` for SystemV, or `find.conf` for Upstart).

Configure the Reverse Proxy

The following sections provide configuration examples for using different reverse proxy servers.

Apache 2

Apache 2 is a free open source Web server that you can configure as a reverse proxy by using the `mod_proxy` module.

The following configuration example exposes the Find running on `find.example.com` at `https://localhost:443/find`, using AJP to enable remote authentication. The application AJP port is 8009, and the HTTPS port is 8443.

For this configuration, you must enable the `mod_proxy`, `mod_proxy_ajp`, `mod_proxy_wstunnel`, and `mod_ssl` modules.

```
SSLEngine on
SSLProxyEngine on

SSLCertificateFile /path/to/my-certificate.crt
SSLCertificateKeyFile /path/to/my-key.key

<Location /find/>
    AuthType YOUR_AUTH_MODULE
    require valid-user

    ProxyPass "ajp://find.example.com:8009/"
    ProxyPassReverse /

</Location>
```

The following configuration example exposes the Find running on `find.example.com` at `https://localhost:443/find`, using HTTPS to communicate with the application.

For this configuration, you must enable the `mod_proxy`, `mod_proxy_http`, `mod_proxy_wstunnel`, and `mod_ssl` modules.

```
SSLEngine on
SSLProxyEngine on

SSLCertificateFile /path/to/my-certificate.crt
SSLCertificateKeyFile /path/to/my-key.key

<Location /find/>
    AuthType YOUR_AUTH_MODULE
    require valid-user

    ProxyPass "https://find.example.com:8443/"
    ProxyPassReverse /

</Location>
```

SAML

You can configure Apache 2 to act as a SAML service provider, which ensures that users are authenticated before passing them through to the application, and sends them to an identity provider if they are not.

For this configuration, you must install and enable the `mod_auth_mellon` module.

```
SSLEngine on
SSLProxyEngine on
```



```

SSLCertificateFile /path/to/my-certificate.crt
SSLCertificateKeyFile /path/to/my-key.key

<Location /find/>
    AuthType "Mellon"
    AuthName YOUR_AUTH_NAME
    Require valid-user

    MellonEnable "auth"
    MellonSecureCookie On
    MellonVariable "session_cookie"
    MellonUser "NAME_ID"
    MellonEndpointPath "/mellon"
    MellonDefaultLoginPath "/"

    # Session TTL seconds
    MellonSessionLength 86400

    MellonSPMetadataFile /path/to/service/provider/metadata
    MellonSPPrivateKeyFile /path/to/service/provider/key/file
    MellonSPCertFile /path/to/service/provier/cert/file
    MellonIdPMetadataFile /path/to/idp/metadata/file

    ProxyPass "https://find.example.com:8443/"
    ProxyPassReverse /
</Location>

```

NGINX

Nginx is a free open source Web server and reverse proxy. It does not support the AJP protocol, so remote authentication is not possible.

The following example configuration exposes the Find instance running on `https://find.example.com:8443/find` at `https://localhost:443/find`.

```

server {
    listen 443 ssl default_server;

    ssl_certificate /path/to/my-certificate.crt;
    ssl_certificate_key /path/to/my-key.key

    location /find/ {
        proxy_pass https://find.example.com:8443/find;
        proxy_cookie_path /find;
    }
}

```

Internet Information Services (IIS)

IIS is a Web server created by Microsoft that is built into many versions of Windows. You can configure it as a reverse proxy in HTTP mode, by using the URL Rewrite and Application Request Routing (ARR) modules. In later versions of IIS, the Web server can automatically proxy WebSockets connections.

You can enable the AJP protocol by using the Apache Tomcat ISAPI redirector module.

To reverse proxy in HTTP mode, you must configure a URL Rewrite rule that redirects incoming requests to the application, and another rule that rewrites Location headers in the response.

Configure Find to Use a Single Pre-Authenticated User

You can configure Find to always log in with a single pre-authenticated user. This option bypasses authentication so that any user that accesses Find is logged in automatically with that user name and any associated permissions.

To use this option, you must set the `server.reverseProxy` system property to **true**, even if you do not want to use a reverse proxy. The reverse proxy is not required, because the normal Find HTTP(S) ports still work.

You then set the `find.reverse-proxy.pre-authenticated-roles` and `find.reverse-proxy.pre-authenticated-username` system properties to the role and user name that you want to use for all users.

To configure Find to use a single pre-authenticated user

1. At the command line, send the `java` run command with the `server.reverseProxy` argument set to **true**, the `find.reverse-proxy.pre-authenticated-roles` argument set to the roles to use, and the `find.reverse-proxy.pre-authenticated-username` argument set to the user name to use. For example:

```
java -Dserver.reverseProxy=true -Dfind.reverse-proxy.pre-authenticated-
roles=FindUser,FindBI -Dfind.reverse-proxy.pre-authenticated-username=anon -
Didol.find.home=[home directory] -Dserver.port=[port] -jar find.war
```

2. Restart Find to apply your configuration changes.

If you run Find as a service on Windows, you can also add the `server.reverseProxy` and `find.reverse-proxy.pre-authenticated-username` properties to the `find.xml` file.

If you run Find as a service on Linux, you can add the arguments to the existing arguments variable in the start scripts (`find.sh` for SystemV, or `find.conf` for Upstart).

Send documentation feedback

If you have comments about this document, you can [contact the documentation team](#) by email. If an email client is configured on this system, click the link above and an email window opens with the following information in the subject line:

Feedback on Administration Guide (Micro Focus Find 11.6)

Add your feedback to the email and click **Send**.

If no email client is available, copy the information above to a new message in a web mail client, and send your feedback to swpdl.idoldocsfeedback@microfocus.com.

We appreciate your feedback!