



Search in SharePoint 2013 and SharePoint Online solution pack

Microsoft Corporation

September 2014

**Applies to:** SharePoint 2013 and SharePoint Online

**Summary:** This solution pack includes code and documents that describe search and demonstrate techniques for customizing search for SharePoint 2013 and SharePoint Online.

©2014 Microsoft Corporation. All rights reserved.

This document is provided "as-is." Information and views expressed in this document, including URL and other Internet website references, may change without notice. You bear the risk of using it.

Some examples are for illustration only and are fictitious. No real association is intended or inferred.

This document does not provide you with any legal rights to any intellectual property in any Microsoft product. You may copy and use this document for your internal, reference purposes. You may modify this document for your internal, reference purposes.

Module 1: Introduction to SharePoint Search Architecture, APIs and Search Apps

# What is a solution pack?

A solution pack is a set of evolving guidance created in a partnership between the SharePoint product team and the SharePoint community. Solution packs are aligned around a specific scenario and exist to clarify complicated stories and help users meet goals with new patterns and technologies.

The core building block of each solution pack is the *module*. The scope and purpose of the module remains constant, but the content within each module can grow and change over time. This is version 1.0 of the Search in SharePoint 2013 and SharePoint Online solution pack, which contains two module documents, associated code samples, inline code examples, and numerous links to additional Microsoft and SharePoint community resources.

Some examples of content types that can be included in a solution pack:

* **Documents**. Written guidance you can use to understand concepts, choose from among options, and complete tasks.
* **Pictures**. Visual elements such as diagrams and infographics that present complex information simply and clearly.
* **Code**. Code samples that do complicated customization work for you or provide an easily customizable resource you can use to save coding and design time.
* **Case studies**. Case studies that describe how others in the same situation used new technologies and patterns to achieve business and design goals.

# How is this different?

Historically, Microsoft has published product documentation that has come mostly from the product team, and we’ve provided guidance to enterprise customers largely from the field via consulting services.

With solution packs, the product team is engaging directly with consulting services and the community to use the cycle of services consumption and production to deliver a dynamic story that cycles around customer needs and goals.

**Acknowledgements**

The solution pack was made possible by contributions and/or review by the following individuals:

*Cathy Anderson, Todd Baginski, Scarlet Bulterman* *Linda Lu Cannon, Jim Crowley, Bella Engen, Rob Howard, Vesa Juvonen, Aditi Maheshwari, Frank Marasco, Brian Michely, Ken Milne, Kavindra Palaraja, Kiki Shuxteau, Lucas Smith, Jo-Anne Stamerjohn, Jeremy Thake, Cindy Yan, Benny Zhang*

# Contents

The Search in SharePoint 2013 and SharePoint Online solution pack includes two documents, or modules, listed in Table 1.

**Table 1. Search in SharePoint 2013 and SharePoint Online solution pack modules**

|  |  |  |  |
| --- | --- | --- | --- |
| **Module** | **Name** | **Description** | |
| 1 | Introduction to SharePoint Search Architecture, APIs and Search Apps | Introduces the solution pack concept and contents, including SharePoint search fundamentals and architecture. It also introduces the search Query CSOM and REST APIs and provides an introduction to developing search apps. | |
| 2 | Search customizations for SharePoint 2013 and SharePoint Online | Describes search customizations scenarios and discusses techniques and samples to customize search for SharePoint 2013 and SharePoint Online. | |
|  | | |  |

## Where are the samples?

**Important** Each module document is associated with one or more code samples, which are downloadable individually from Code Gallery.

# Key Terms and Concepts

Table 2 lists terms and concepts that are useful to know as you start to work with SharePoint site provisioning and branding with the remote provisioning pattern.

**Table 2. Key terms and concepts**

|  |  |  |  |
| --- | --- | --- | --- |
| **Term** | **Definition** | **File system, path, syntax, or UI location** | **Guidance** |
| [App for SharePoint](http://msdn.microsoft.com/library/office/fp179930.aspx) |  | Site Contents tab | [Build apps for SharePoint](http://msdn.microsoft.com/en-us/library/office/apps/jj163230.aspx)  [Apps for SharePoint sample pack](http://code.msdn.microsoft.com/sharepoint/Apps-for-SharePoint-sample-64c80184) |
| App web | The website from which an app is deployed. | Structure of an app web URL:  *https:// App\_Prefix - App\_ID . App\_Base\_Domain / Domain\_Relative\_URL\_of\_Host\_Web / App\_Name* | [Host webs, app webs, and SharePoint components](http://msdn.microsoft.com/en-us/library/office/fp179925.aspx) |
| ClientContext | A central object that serves as a “center of gravity” for all SharePoint CSOM and JSOM operations. | public class ClientContext : ClientRuntimeContext | [ClientContext](http://msdn.microsoft.com/en-us/library/office/microsoft.sharepoint.client.clientcontext.aspx) class |
| Cloud App Model (CAM) | Apps for SharePoint are self-contained pieces of functionality that extend the capabilities of a SharePoint website. You can use CAM to author and deliver secure, performant, flexible, and consistent apps for SharePoint. |  | [Apps for SharePoint overview](http://msdn.microsoft.com/en-us/library/office/fp179930.aspx) |
| CSOM | Client-side object model. A model for writing client-side code for SharePoint with the .NET framework. |  | [Using the client-side object model (CSOM)](http://msdn.microsoft.com/en-us/library/ff798388.aspx)  [[MS-CSOM]: SharePoint Client Query Protocol](http://msdn.microsoft.com/en-us/library/dd912094(v=office.15).aspx)  [SharePoint 2013 .NET Server, CSOM, JSOM, and REST API index](http://msdn.microsoft.com/en-us/library/office/dn268594.aspx) |
| Hive | SharePoint’s physical files; the files in the file system. These files are distinct from content stored in a content database. | %program files%/Common Files/Microsoft Shared/Web Server Extensions/15/ | See also “File system” and “content database.” |
| Host web | The website on which an app is installed. |  | [Host webs, app webs, and SharePoint components](http://msdn.microsoft.com/en-us/library/office/fp179925.aspx) |
| REST | REpresentional State Transfer. A stateless architectural style that abstracts architectural elements and uses HTTP verbs read and write data from Web pages that contain XML files. |  | [Get Started with the SharePoint 2013 REST service](http://msdn.microsoft.com/en-us/library/office/apps/fp142380.aspx)  [Using the REST interface](http://msdn.microsoft.com/en-us/library/ff798339.aspx) |
| Root web | The first web inside of a site collection. |  | The root web is also sometimes referred to as the “Web Application Root.” |
| [SharePoint Online](http://office.microsoft.com/en-us/sharepoint/sharepoint-online-online-collaboration-software-FX103789366.aspx) | SharePoint Online is the cloud-based SharePoint offering in Office 365. |  |  |
| Site | A group of sites that share the same owner and administrative settings, such as permissions. |  | [Site](http://msdn.microsoft.com/en-us/library/office/microsoft.sharepoint.client.site.aspx) class  [Create a site collection in SharePoint 2013](http://technet.microsoft.com/en-us/library/cc263094.aspx) |
| Web | Represents a Microsoft SharePoint Foundation Web site. |  | [Web](http://msdn.microsoft.com/en-us/library/office/microsoft.sharepoint.client.web.aspx) class |
| Subsite | A single SharePoint site in a SharePoint site collection. A subsite can inherit navigation and permissions from a parent site, or it can have unique permissions and navigation. |  | You can create subsites that are based on the root site collection, or sites based on other site collections. You can choose to inherit permissions from the site collection or specify unique permissions for the subsite. |

# SharePoint search architecture

In SharePoint 2013, search has been re-architected, combining the ease of configuration and deployment in SharePoint 2010 with the scalability and extensibility of FAST Search Server to a single enterprise search platform.

With this release, SharePoint introduced common patterns into the search platform to help you use SharePoint’s out-of-the-box capabilities to easily customize search for different scenarios.

Some examples are:

* Video search and conversation search added as out-of-the-box search verticals.
* Topic pages and Content by Search, which greatly enhances the web content management capabilities and supporting scenarios such as search-driven sites and knowledge management sites.
* My tasks, which pulls together project tasks so that users can track tasks assigned in multiple sites in a central location, their OneDrive for Business site.

Figure 1 shows the common patterns in the SharePoint search platform.

**Figure 1. SharePoint search platform common patterns**



Before you start customizing search, you should become familiar with the search architecture in SharePoint 2013, which is comprised of components and databases that work cohesively to enable search functionality. The following tables describe these components and databases.

**Table 3. Search components in SharePoint 2013**

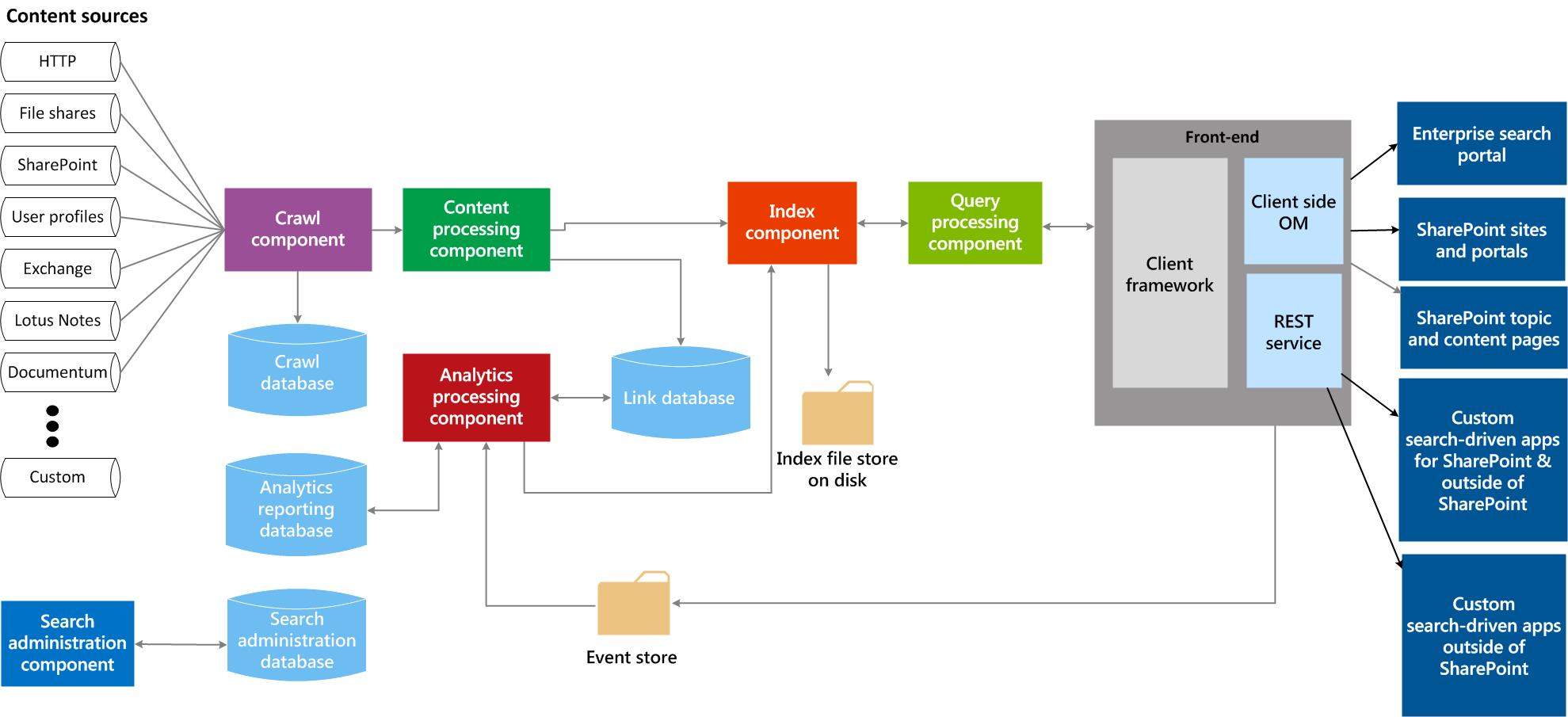
|  |  |
| --- | --- |
| **Component** | **Description** |
| Crawl component | Crawls content sources to collect crawled properties and metadata from crawled items and sends this information to the content processing component. |
| Content processing component | Transforms the crawled items and sends them to the index component. This component also maps crawled properties to managed properties. |
| Analytics processing component | Carries out search analytics and usage analytics. |
| Index component | Receives the processed items from the content processing component and writes them to the search index. This component also handles incoming queries, retrieves information from the search index and sends back the result set to the query processing component. |
| Query processing component | Analyzes incoming queries. This helps optimize precision, recall and relevance. The queries are sent to the index component, which returns a set of search results for the query. |
| Search administration component | Runs the system processes for search, and adds and initializes new instances of search components. |

**Table 4. Search databases in SharePoint 2013**

|  |  |
| --- | --- |
| **Database** | **Description** |
| Crawl database | Stores tracking information and historical information about crawled items such as documents and URLs. It also stores information such as the last crawl time, the last crawl ID and the type of update (add, update, delete) during the last crawl. |
| Link database | Stores unprocessed information that is extracted by the content processing component and information about search clicks. The analytics processing component analyzes this information. |
| Analytics reporting database | Stores the results of usage analysis. |
| Search administration database | Stores search configuration data. |

Figure 2 provides an overall view of how these components and databases interact.

**Figure 2. Search components interaction**



## Crawl and content processing

The crawl process starts with the different sources of content (ex. HTTP, file shares, SharePoint, etc...) represented on the top left part of the diagram. For content to be added to the index, the crawler uses connectors that tell the crawler how to connect to the content source and access the content items within the source. Once the crawler has found the content items, it uses applicable format handler to parse the content.

After retrieving the content, the crawl component passes crawled items to the content processing component, where the component processes crawled items and sends these items to the index component.

This processing includes:

* Document parsing
* Mapping crawled properties to managed properties
* Linguistics processing such as language detection and entity extraction.

The content processing component also writes information about links and URLs to the link database.

## Query processing

The query processing component analyzes and processes search queries to optimize precision, recall and relevance, including performing linguistics processing such as word breaking and stemming. The processed query is then submitted to the index component, which returns a result set based on the processed query to the query processing component, which in turn processes that result set.

## Search analytics

SharePoint analyzes both the content itself, (search analytics) and also the way that users interact with it, (usage analytics) and uses this information to improve search.

Search analytics is about extracting information, such as links, the number of times an item is clicked, anchor text, data related to people, and metadata, from the link database. This information is important to relevance.

Usage analytics is about analyzing usage log information received from the front-end via the event store. Usage analytics generates usage and statistics reports.

The results from the analyses are added to the items in the search index. In addition, results from usage analytics are stored in the analytics reporting database.

# Search feature availability

See [What's new in search in SharePoint Server 2013](http://technet.microsoft.com/library/ee667266(v=office.15).aspx) for information about the new search capabilities in SharePoint 2013. To learn about what search feature availability across SharePoint solutions, see [Search Features in SharePoint Online](http://technet.microsoft.com/en-us/library/sharepoint-online-search-service-description.aspx).

# Building blocks for customizing the search experience

Search in SharePoint 2013 and SharePoint online include a lot of new functionality and improvements that allow you to extensively customize the search experience. These new features enable a lot of customization out-of-the-box without you having to write any custom code. SharePoint search also includes CSOM and REST APIs when you do need to write code for your customizations, or if you want to create apps to access SharePoint search results outside of SharePoint.

To take full advantage of the new functionality, you should understand what new features have been added, how existing functionality has been improved, and how everything works together to provide the search experience. This section covers what you need to know to begin customizing SharePoint search.

## Search Center site

The Search Center is a SharePoint site oriented around the task of search. It is a portal where you can search for content on your organization's intranet, providing a centralized and highly customizable user interface.

This module describes the Search Center’s pages and web parts, along with the search configuration settings that you can modify to create custom search experiences without requiring that you write a lot of code, or create custom search applications.

When you create a Search Center site, SharePoint creates a default search home page and a default search results page. In addition, several pages known as *search verticals* are also created. Search verticals are search results pages that customized for searching specific content, such as **People** and **Videos**, and they display search results that are filtered and formatted for a specific content type or class.

The following pages are created in a Search Center site collection, in the **Pages** library:

* **default.aspx:** the home page for the Search Center, and the page where end-users enter their queries.
* **results.aspx:** the default search results page for the Search Center. It is also the search results page for the **Everything** search vertical.
* **peopleresults.aspx:** the search results page for the **People** search vertical.
* **conversationresults.aspx:** the search results page for the **Conversations** search vertical.
* **videoresults.aspx:** the search results page for the **Videos** search vertical.
* **advanced.aspx:** the search page where end-users can apply some restrictions to their search phrases — for example, limiting the search to an exact phrase.

All of the search vertical pages contain the Search results web part, however it is configured differently for each search vertical. For each search vertical page, the query in the Search Results web part is directed to a particular result source, applicable to that search vertical. For example, the query in the Search Results web part on the peopleresults.aspx page is limited to the **Local People Results** result source. Understanding how the default search verticals in SharePoint 2013 are configured can help you when you want to create your own search vertical or make other Search Center customizations.

Some additional resources for working with the Search Center:

[Set up a Search Center in SharePoint 2013](http://technet.microsoft.com/en-us/library/dn794206(v=office.15).aspx)

[How to create a Search Center Site Collection and enable crawling of your content in SharePoint 2013](http://technet.microsoft.com/en-us/library/dn794245(v=office.15).aspx)

[Create a Search Center site in SharePoint Server 2013](http://technet.microsoft.com/en-us/library/hh582314(v=office.15).aspx)

[Manage the Search Center](http://technet.microsoft.com/en-us/library/jj851144(v=office.15).aspx)

[Manage the Search Center in SharePoint Online](http://office.microsoft.com/en-us/office365-sharepoint-online-enterprise-help/manage-the-search-center-in-sharepoint-online-HA103994122.aspx)

## Search web parts

The Search Center pages contain the following web parts:

* Search Box
* Search Results
* Search Navigation
* Refinement

This section describes these web parts and how you can configure them to customize search and improve how search results are displayed.

### Search Box web part

The Search Box web part shows a text box where users can enter words or phrases to search for information. By default, the Search Box web part is used on the Search Center home page (default.aspx), and all default search results pages (results.aspx, peopleresults.aspx, conversationresults.aspx, and videoresults.aspx).

By editing the Search Box web part properties in the web part tool pane you can:

* Change where the search results should be displayed — for example, show results in a custom Search Results web part or a custom search results page.
* Turn off query suggestions and people suggestions.
* Show links to a search preference page and an advanced search page.
* Change the display template for the web part.

For more information see:

* [Configure properties of the Search Box Web Part in SharePoint Server 2013](http://technet.microsoft.com/en-us/library/gg576963(v=office.15).aspx)
* [How to change the text that is displayed in the Search Box Web Part in SharePoint Server 2013](http://technet.microsoft.com/en-us/library/dn794238(v=office.15).aspx)

### Search Results web part

The Search Results web part displays the results of a search query. By default, the Search Results web part is used on all default search vertical pages (results.aspx, peopleresults.aspx, conversationresults.aspx, and videoresults.aspx). The Search Results web part also sends the search results to the Refinement web part and the Search Navigation web part, so there must be a Search Results web part on a search results page for the other search web parts to work.

You can edit the Search Results web part properties in the web part tool pane to change the search query, and the behavior and appearance of results on the search results page.

By changing these properties you can do the following:

* Change the result source to specify which content should be searched.
* Add query variables or property filters to customize search results for different users or user groups.
* Promote or demote items or pages within the search results.
* Change the sorting of the search results.
* Change the display template.

For more information, see:

* [Configure properties of the Search Results Web Part in SharePoint Server 2013](http://technet.microsoft.com/en-us/library/gg549987.aspx)
* [How to configure the Search Results Web Part to use a new result source in SharePoint 2013](http://technet.microsoft.com/en-us/library/dn794200(v=office.15).aspx)

### Search Navigation web part

The Search Navigation web part shows links that let users move quickly between the different search verticals (**Everything**, **People**, **Conversations**, and **Videos**). The Search Navigation web part uses search results from the Search Results web part so that when users click a search vertical link, the search results are filtered and displayed according to how the search vertical is set up. By editing the Search Navigation web part properties in the web part tool pane you can:

* Specify a different web part to get the results from.
* Change the number of search vertical links to show.
* Change the appearance and layout of the web part.

You can make other Search Navigation related changes in **Site Settings** -> **Search Settings**, such as:

* Changing the link display names.
* Changing the link order.

Adding a link to a new search vertical to be shown in the web part.

### Refinement web part

The Refinement web part filters search results into categories called refiners. Users can click these refiners to narrow search results to find what they're looking for more easily. Refiners are managed properties that are marked as *Refinable* and *Queryable*. For information about these settings, see **Managed property setting overview** in [Overview of the search schema in SharePoint Server 2013](http://technet.microsoft.com/en-us/library/jj219669(v=office.15).aspx).

By editing the Refinement web part properties in the web part tool pane you can specify:

* Which Search Results web part to filter search results from.
* The refiners to use in the Refinement web part.
* The display template that is applied to each refiner.
* The appearance, layout, and behavior of the Refinement web part.

By default, the Refinement web part doesn't show the number of results for each refiner value. You can add refiner counts by modifying the display template for the refiner, see **Add refiner counts to the Refinement Web Part** in [Configure properties of the Refinement Web Part in SharePoint Server 2013](http://technet.microsoft.com/en-us/library/gg549985(v=office.15).aspx).

For more information, see:

* [Plan to use refiners on a search results page in SharePoint 2013](http://technet.microsoft.com/en-us/library/dn794223(v=office.15).aspx)
* [How to add refiners to your search results page in SharePoint 2013](http://technet.microsoft.com/en-us/library/dn794243(v=office.15).aspx)

## Result sources

Result sources limit searches to certain content or to a subset of search results.  A result source is defined by the following:

* A search provider or source URL to get search results from — for example, the search index of the local SharePoint Search service
* A protocol to use to get search results — for example, the OpenSearch protocol
* A query transform, which can narrow results from the given search provider or URL to a specific subset of results— for example, a set of results that has a particular content type

SharePoint Server 2013 provides 16 pre-configured result sources, including **Local SharePoint Results**, **Conversations**, **Items related to current user**, etc…

You can view details about result sources from the **Manage Result Sources** page. (**Site Settings** -> **Search** -> **Result Sources**)

From the **Manage Result Sources** page, you can create new result sources in either of the following two ways:

* You can click **New Result Source**. For more information, see [Configure result sources for search in SharePoint Server 2013](http://technet.microsoft.com/en-us/library/jj683115(v=office.15).aspx).
* You can point to the arrow next to an existing result source, click **Copy**, and then modify the copy as necessary and save it with a new name.

A result source specifies one of four protocols to use to get search results, and if the result source uses a protocol other than **Local SharePoint**, the result source must also specify a URL from which to get search results, as shown in the following table.

**Table 5. Result source protocols**

|  |  |  |
| --- | --- | --- |
| **Result source protocol** | **Provider** | **URL** |
| Local SharePoint | The search index of the local Search service | N/A |
| Remote SharePoint | The search index of a Search service hosted in another farm | The address of the root site collection of the remote SharePoint farm |
| OpenSearch 1.0/1.1 | An external search provider (such as a remote search engine or feed) that uses the OpenSearch protocol to provide search results | The URL of the RSS feed of a search provider that uses the OpenSearch protocol |
| Exchange | Exchange Web Services | An Exchange Web Services URL |

For more information, see:

[Understanding result sources for search in SharePoint Server 2013](http://technet.microsoft.com/en-us/library/dn186229(v=office.15).aspx.)

[About result sources and federation](http://technet.microsoft.com/en-us/library/jj219577(v=office.15).aspx#Section12)

[Understanding result sources](http://office.microsoft.com/en-us/sharepoint-server-help/understanding-result-sources-HA102848849.aspx?CTT=1)

## [Manage result sources](http://office.microsoft.com/en-us/office365-sharepoint-online-enterprise-help/manage-result-sources-HA103639370.aspx)

## Query rules

Query rules enable you to customize the search experience for the kinds of queries that are important to your users without needing any custom code. In a query rule, you specify the context, conditions and correlated actions. In the applicable context, when a query meets the conditions, search performs the actions to improve the relevance of the search results.

For context, you can restrict a rule to queries:

* performed on a particular result source
* from a particular category of topic page
* by a user matching a particular user segment

Table 3 lists the conditions you can specify to cause the rule to fire.

**Table 6. Query rule conditions**

|  |  |  |
| --- | --- | --- |
| **Condition** | **Description** | **Example** |
| Query matches keyword exactly | Apply the query rule when the query exactly matches a word or phrase that you specify. | You specify "picture; pic" as the keywords. The query rule will apply when users type the query "picture" or "pic" in a search box. The rule will not apply if a user types "pictures" or "sunny picture". |
| Query contains action term | Apply the query rule when the query contains a term in the form of a single word or phrase that indicates that the user is trying to do something. The term must be at the beginning or end of the query and might be a verb, a command, or a filter. | If a query contains the phrase "download", the user is probably not looking for items that contain the word "download", but is probably trying to download something. |
| Query matches dictionary exactly | Apply the query rule when the query exactly matches a dictionary entry. This entry can be a term in the term store, or an entry in the people names dictionary. | A word that a user types in a search box matches an entry in the pre-configured dictionary. |
| Query more common in source | Apply the query rule if the user’s query is more commonly performed against a different result source than the current one. This condition uses an analysis of queries that users entered in the various result sources. | You can create a query rule that checks if a query is more commonly performed in a Video vertical. It will make video results more prominent if it is. |
| Result type commonly clicked | Apply the query rule if the query often ends in users clicking results of a particular result type. When you create a new result type, you can indicate that these clicks should be recorded to be used in query rules. | If this is a query where people often click the result type "pictures", it may be appropriate to provide picture-related results in a result block. |
| Advanced query text match | Apply the query rule if the query matches a regular expression. It also allows you to use variations of the keyword, dictionary and action term conditions explained earlier, but with more advanced control. | To match all phone numbers that are in the format nnn-nnn-nnnn, you specify the regular expression "\(?(\d{3})\)?-?(\d{3})-(\d{4})". |

A query rule can specify the following three types of actions:

* Add **Promoted Results** (formerly called **Best Bets**) that appear above ranked results. For example, for the query "sick leave", a query rule could specify a particular Promoted Result, such as a link to a site that has a statement of company policy regarding time off work.
* Add one or more groups of results, called result blocks. A result block contains a small subset of results that are related to a query in a particular way. Like individual results, you can promote a result block or rank it with other search results.
* Change the ranking of results by changing the query. For example, for a query that contains “download toolbox”, a query rule could recognize the word “download” as an action term and boost search results that point to a particular download site on your intranet.

See [Manage query rules in SharePoint Server 2013](http://technet.microsoft.com/en-us/library/jj871676.aspx) for more information.

## Query Transforms

To provide search results that are appropriate for a user query, sometimes the query needs to be modified; this is done with query transforms. Default search verticals included with SharePoint 2013, such as **Videos**, **People**, and **Conversations** all contain predefined query transforms to optimize the search experience for that vertical. For example, queries submitted from the **Videos** search vertical are changed so that only video files are returned in the search results.

You can configure query transforms in three places:

* In a web part, such as a Search Results web part.
* In a query rule, which specifies that certain actions will be performed only if certain conditions are satisfied.
* In the result source that the query uses to get search results.

See [Plan to transform queries and order results in SharePoint 2013](http://technet.microsoft.com/en-us/library/60a1110d-27dc-45d0-86e2-cddc72d072b2(v=office.15)) for more information.

A user query is transformed first by the web part, then by any query rules that apply, and finally by the result source. When you configure a transform in a result source, you know that the transform changes will not be discarded or overridden, because the result source transforms the query last. You can re-use a result source query transform in web parts or result blocks, and you can create query rules or result types that are only applied to results from certain result sources.

You can use the Query Builder to help you write and test query transforms. You can test the query from within the Query Builder by setting temporary test values for the query variables, run the query and preview the search results.

You can launch the Query Builder from:

* The web part properties dialog, click **Change Query** in the **Search Criteria** section.
* The **Add Query Rule** or **Edit Query Rule** pages, click:
  + **Add Result Block**  -> **Launch Query Builder**
  + **Change ranked results by changing the query**

The **Add Result Source** or **Edit Result Source** pages, click **Launch Query Builder** in the **Query Transforms** section.

## Result types and display templates

SharePoint 2013 search includes a new results framework that makes it easy to customize the way search results are displayed. Now, instead of writing custom XSLT to change how search results are displayed, you can customize the appearance of important types of results by using display templates and result types.

### Result types

To display search results differently, search results have to be sorted into different result types. A result type is a classification of a search result, distinguishing one search result from another, and is comprised of a collection of the following:

* **Rules** One or more characteristics or conditions to compare each search result against, such as the result source or content type of the search result. Rule conditions can be joined by using equality, comparison, and logical operators.
* **Properties** The list of managed properties for the search result. You must add managed properties to the properties list before you map the managed property to a display template.
* **Display templates** Controls the way in which all results that meet the conditions appear and behave on a search results page.

SharePoint search includes several default result types. To see them, go to **Site Settings** --> **Site Collection Administration** --> **Search Result Types**. You cannot edit any of the default result types, however you can create new result types by copying existing ones, and modifying the new ones. For more information about the default result types included with SharePoint 2013, see [Result types and display templates that are used to display search results in SharePoint Server 2013](http://technet.microsoft.com/en-us/library/dn386874(v=office.15).aspx).

### Display templates

Display templates define the visual layout and behavior of search results. They control which managed properties are shown in search results and how they appear. SharePoint stores display templates in the **Search** subfolder of the **Display Templates** folder in the **Master Page Gallery**. Each display template consists of two files: an HTML version of the display template that you can edit in your HTML editor, and a .js file that SharePoint uses. When working with display templates, you modify the HTML file; the .js file is created and modified by SharePoint, you do not edit this file at all.

There are two primary types of display templates

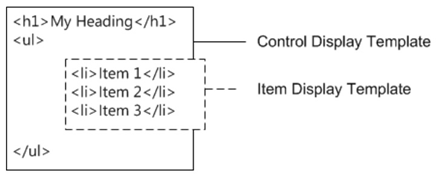
* **Control display templates** which determine the overall structure of how the results are presented.
* **Item display templates** which determine how each result in the set is displayed.

The control display template provides HTML to structure the overall layout for how you want to present the search results. For example, the control display template might provide the HTML for a heading and the beginning and end of a list. The control display template is rendered only once in the web part.

The item display template provides HTML that determines how each item in the result set is displayed. For example, the item display template might provide the HTML for a list item that contains a picture, and three lines of text that are mapped to different managed properties associated with the item. The item display template is rendered one time for each item in the result set. So, if the result set contains ten items, the item display template creates its section of HTML ten times.

When used together in this way, the control display template and the item display template combine to create a cohesive block of HTML that is rendered in the Web Part, as shown in Figure 2.

**Figure 1. Combined HTML output of a control display template and item display template**



For an in-depth look at display templates and their structure, see [SharePoint 2013 Design Manager display templates](http://msdn.microsoft.com/en-us/library/jj945138.aspx) and **Search-driven Web Parts and display templates** in [Overview of the SharePoint 2013 page model](http://msdn.microsoft.com/en-us/library/jj191506.aspx#bk_SearchDriven).

For more information about display templates available in SharePoint 2013, see [Display template reference in SharePoint Server 2013](http://technet.microsoft.com/en-us/library/jj944947.aspx).

#### Customizing display templates

If you want to customize display templates included with SharePoint, instead of customizing the existing version, you should create new display template by copying the content from the one you want to modify, and then customize the new version. Starting from a copy of an existing display template is also the easiest way to create a new one, as it ensures that you are starting with all the required elements.

Another tip when working with display templates is to map a network drive to the Master Page Gallery, for details, see [How to: Map a network drive to the SharePoint 2013 Master Page Gallery](http://msdn.microsoft.com/en-us/library/office/jj733519(v=office.15).aspx).

The HTML file that is used for a display template is a fully-formed HTML document, with a **head** element and **body** element.

Within the **head** element, there's a **title** element which specifies the display name for the display template. The text in this tag is what will be shown when you do configurations in the SharePoint UI, for example, when you configure a result type.

After the title element, there’s a custom document properties element, **mso:CustomDocumentProperties**. In item display templates, this element contains a **mso:ManagedPropertyMapping** element, which is where the managed properties used by SharePoint search are mapped to values used by the display template.

The syntax for this is:

'<*display template reference name*>':<*managed property name*>'

Here’s an example:



Within the **body** element, there's a **script** element where you can include external resources such as CSS files or JavaScript files outside of the display template. See the **Script block** section of [SharePoint 2013 Design Manager display templates](http://msdn.microsoft.com/en-us/library/jj945138.aspx) for examples showing how to include external resources in the script element.

The next element is a **div** element; this is where you place any HTML or script that you want as part of the display template.

A good way to become familiar with the display template structure is to download copies of the default display templates for search results, **Control\_SearchResults.html**, the control display template, and **Item\_Default.html**, the item display template.

### Additional resources – result types and display templates

Some additional resources for display templates and result types:

[Customize search result types in SharePoint 2013](http://technet.microsoft.com/en-us/library/dn135239(v=office.15).aspx)

[How to change the way search results are displayed in SharePoint Server 2013](http://technet.microsoft.com/en-us/library/dn794204(v=office.15).aspx)

[Understanding how search results are displayed in SharePoint Server 2013](http://technet.microsoft.com/en-us/library/dn794212(v=office.15).aspx)

[Understanding how item display templates and hit highlighting work in SharePoint Server 2013](http://technet.microsoft.com/en-us/library/dn794249(v=office.15).aspx)

[How to create a new result type in SharePoint Server 2013](http://technet.microsoft.com/en-us/library/dn794234(v=office.15).aspx)

[How to display values from custom managed properties in search results - option 1 in SharePoint Server 2013](http://technet.microsoft.com/en-us/library/dn794209(v=office.15).aspx)

[How to display values from custom managed properties in search results – option 2 in SharePoint Server 2013](http://technet.microsoft.com/en-us/library/dn794240(v=office.15).aspx)

[How to display values from custom managed properties in the hover panel in SharePoint Server 2013](http://technet.microsoft.com/en-us/library/dn794247(v=office.15).aspx)

# Query APIs and Search apps

SharePoint 2013 search includes .NET and JavaScript client object models and a REST service that enables access to search results for online, on-premises, and mobile development.

Table 7 shows the APIs that you can use to program search queries and the path to the source file on the server.

**Table 7. Search Query APIs**

|  |  |  |
| --- | --- | --- |
| **API name** | **Class library or schema and path** | **Example** |
| .NET CSOM | Microsoft.SharePoint.Client.Search.dll  [SharePoint Server 2013 Client Components SDK download](http://www.microsoft.com/en-us/download/details.aspx?id=35585)  %ProgramFiles%\Common Files\Microsoft Shared\web server extensions\15\ISAPI  [SharePoint Online Client Components SDK download](http://www.microsoft.com/en-us/download/details.aspx?id=42038)  %ProgramFiles%\Common Files\Microsoft Shared\web server extensions\16\ISAPI | Personalized search results (URL TBD)  [SharePoint 2013: Query Search with the Managed Client Object Model (Code Gallery)](http://code.msdn.microsoft.com/Query-Search-with-the-649f1bc1) |
| JavaScript CSOM | SP.search.js  %ProgramFiles%\SharePoint Client Components\Scripts | [Querying Search with the JavaScript Client Object Model (Code Gallery)](http://code.msdn.microsoft.com/SharePoint-2013-Querying-a629b53b) |
| Search REST service | http://server/\_api/search/query  http://server/\_api/search/postquery  http://server/\_api/search/suggest | [SharePoint 2013: Using the search REST service from an app for SharePoint (Code Gallery)](http://code.msdn.microsoft.com/sharepoint/SharePoint-2013-Perform-a-1bf3e87d) |

## Search Query .NET CSOM

To use the query .NET CSOM, get a [ClientContext](http://msdn.microsoft.com/en-us/library/office/microsoft.sharepoint.client.clientcontext(v=office.15).aspx) instance (located in the [Microsoft.SharePoint.Client](http://msdn.microsoft.com/en-us/library/office/microsoft.sharepoint.client(v=office.15).aspx) namespace in the Microsoft.SharePoint.Client.dll). Then use the query object model in the [Microsoft.SharePoint.Search.Client.Query](http://msdn.microsoft.com/en-us/library/office/microsoft.sharepoint.client.search.query(v=office.15).aspx) namespace in the Microsoft.SharePoint.Search.Client.dll.

Here’s a basic example:

using Microsoft.SharePoint.Client;

using Microsoft.SharePoint.Client.Search.Query;

…

using (ClientContext clientContext = new ClientContext("http://intranet.contoso.com"))

{

KeywordQuery keywordQuery = new KeywordQuery(clientContext);

keywordQuery.QueryText = "Argument";

SearchExecutor searchExecutor = new SearchExecutor(clientContext);

ClientResult<ResultTableCollection> results = searchExecutor.ExecuteQuery(keywordQuery);

clientContext.ExecuteQuery();

}

Now you can iterate through the search results. The following example writes out the title of each result:

foreach (var row in results.Value[0].ResultRows)

{

Console.WriteLine(row["Title"]);

}

## Search Query REST service

The Search REST service supports both HTTP **POST** and **GET** requests. When you make a call to the Search REST service, you specify query parameters with the request and search uses these query parameters to construct the search query.

With a **GET** request, you specify the query parameters in the URL. For **POST** requests, you pass the query parameters in the body in JavaScript Object Notation (JSON) format.

GET requests:

<http://server/_api/search/query>

POST requests:

<http://server/_api/search/postquery>

The following table shows some GET request examples.

**Table 8. Sample GET requests for Search REST service**

|  |  |
| --- | --- |
| **Type** | **Request URL** |
| Keywords | http://server/site/\_api/search/query?querytext='{KQL Query}‘ |
| Selecting Properties | <http://server/site/_api/search/query?querytext='test'&selectproperties='Title,Rank>' |
| Sorting | http://server/site/\_api/search/query?querytext='test'&sortlist='LastModifiedTime:descending'  http://server/site/\_api/search/query?querytext='test'&sortlist='LastModifiedTime:descending,Rank:ascending' |

For a complete list of the query parameters available and how to use them, see [SharePoint Search REST API overview](http://msdn.microsoft.com/en-us/library/jj163876(v=office.15).aspx). For sample code, see [SharePoint 2013: Using the search REST service from an app for SharePoint](http://code.msdn.microsoft.com/sharepoint/SharePoint-2013-Perform-a-1bf3e87d).

## Search apps

Apps for SharePoint are self-contained pieces of functionality that extend the capabilities of a SharePoint website. A search app is an app for SharePoint that uses search functionality. In a search app, you can use the search query APIs to retrieve search results. In addition, you can also use a search app to distribute search configurations from one SharePoint installation to another.

### Search app development environment

You have the following options for setting up a development environment to create search apps:

* Microsoft Visual Studio 2012 or Microsoft Visual Studio 2013: [How to: Set up an on-premises development environment for apps for SharePoint (MSDN)](http://msdn.microsoft.com/en-us/library/office/fp179923(v=office.15).aspx).

Office 365 Development Tools: [How to: Set up an environment for developing apps for SharePoint on Office 365 (MSDN)](http://msdn.microsoft.com/en-us/library/office/fp161179(v=office.15).aspx).

### Search app permissions

Search apps require only user-level permissions, known as **QueryAsUserIgnoreAppPrincipal**. This permission lets you query the search app based on the user’s permissions. This means that search results will be returned based on the user’s ACLs. To grant permissions to the app to use search:

1. From **Solution Explorer**, double-click **AppManifest.xml**.
2. On the **Permissions** tab, select **Search for Scope** and then select **QueryAsUserIgnoreAppPrincipal** for **Permissions**.

See [App permissions in SharePoint 2013 (MSDN)](http://msdn.microsoft.com/en-us/library/fp142383(v=office.15).aspx) for more information about app permissions.

### Query APIs for Search apps

You can use the .NET CSOM, JavaScript CSOM, or search REST service to retrieve search results in a search app. The following example demonstrates how to use the query .NET CSOM to retrieve search results in a search app:

var spContext = SharePointContextProvider.Current.GetSharePointContext(Context);

using (var clientContext = spContext.CreateUserClientContextForSPHost())

{

KeywordQuery keywordQuery = new KeywordQuery(clientContext);

keywordQuery.QueryText = "Argument";

SearchExecutor searchExecutor = new SearchExecutor(clientContext);

ClientResult<ResultTableCollection> results = searchExecutor.ExecuteQuery(keywordQuery);

clientContext.ExecuteQuery();

}

### Additional resources - Search apps and apps for SharePoint

See the following resources for more information about Apps for SharePoint and search apps.

[Apps for SharePoint (MSDN)](http://msdn.microsoft.com/en-us/office/dn448479.aspx)

[Get started developing apps for SharePoint (MSDN)](http://msdn.microsoft.com/en-us/library/office/jj163980(v=office.15).aspx)

[Search apps in SharePoint 2013 (MSDN)](http://msdn.microsoft.com/en-us/library/office/dn554260(v=office.15).aspx)

[How to: Deploy custom search configurations by using Visual Studio (MSDN)](http://msdn.microsoft.com/library/office/dn194077.aspx)

[Add search capabilities to your apps for SharePoint (MSDN Blog)](http://blogs.msdn.com/b/officeapps/archive/2013/05/30/add-search-capabilities-to-your-apps-for-sharepoint.aspx)

For examples, see the following search app code samples:

[SharePoint 2013: Using the search REST service from an app for SharePoint (Code Gallery)](http://code.msdn.microsoft.com/sharepoint/SharePoint-2013-Perform-a-1bf3e87d)

[SharePoint 2013: Personalizing search results in an app for SharePoint](http://code.msdn.microsoft.com/SharePoint-2013-Personalizi-fb6ddcf9)