SHAREPOINT 2013 SEARCH CUSTOMIZATION
AND
KNOWLEDGE LAKE IMAGING SEARCH

Jeff Borghoff
Senior Technical Architect
Professional Services – KnowledgeLake, Inc.
CDIA+, SharePoint MOS/MCTS/MCITP, AIIM SharePoint® & SharePoint®
http://www.KnowledgeLake.com
http://blog.KnowledgeLake.com
http://facebook.com/KnowledgeLake
http://twitter.com/KnowledgeLake
http://linkedin.com/company/KnowledgeLake
http://youtube.com/KnowledgeLake
# Table of Contents

**INTRODUCTION**  ........................................................................................................... 4  
**CUSTOMIZING SEARCH IN 2013** .................................................................................. 7  
  - CONTENT SEARCH WEB PART .................................................................................. 7  
  - DISPLAY TEMPLATES ................................................................................................. 8  
  - SEARCH RESULT TYPE ............................................................................................. 9  
  - SEARCH RESULT SOURCE ....................................................................................... 10  
    - Search experiences and corresponding Result Sources ........................................ 10  
    - Available Form Level Result Sources .................................................................. 11  
    - Result Source Protocols ....................................................................................... 11  
  - SEARCH QUERY RULES ......................................................................................... 13  
**CUSTOMIZING SEARCH IN 2013 WITH KNOWLEDGELAKE IMAGING** ......................... 14  
  - SEARCH CENTER .................................................................................................. 15  
    - Search Builder ................................................................................................... 15  
    - Search Results ..................................................................................................... 16  
    - Saved Searches .................................................................................................... 17  
    - Search Export ...................................................................................................... 17  
  - KNOWLEDGELAKE SEARCH WEB PARTS .............................................................. 18  
    - KNOWLEDGELAKE VIEWER ............................................................................... 19  
    - Hover Panel ........................................................................................................ 20  
  - LINE OF BUSINESS INTEGRATION........................................................................... 21  
**CONCLUSION** .............................................................................................................. 22  
**ABOUT KNOWLEDGELAKE** ....................................................................................... 22  
**APPENDIX A – SHAREPOINT 2010 SAMPLE XML FOR ADVANCED SEARCH CUSTOMIZATION** ........................................................................................................ 23  
  - PROPERTIES XML .................................................................................................. 23  
**APPENDIX B – SHAREPOINT 2013 SAMPLE CUSTOM DISPLAY TEMPLATE** .................. 25  
  - ITEM_THREELINES.HTML ...................................................................................... 25  
**APPENDIX C – SHAREPOINT 2013 PDF DISPLAY TEMPLATE FOR THE PDF RESULT TYPE** ........................................................................................................ 26  
  - ITEM_PDF.JS .......................................................................................................... 26
Introduction

This whitepaper is intended to communicate the complexity of customizing the SharePoint 2013 Search features in comparison to the consistent, user friendly and rapidly adoptable KnowledgeLake Imaging Search functionality.

KnowledgeLake introduced its current Imaging search functionality with SharePoint 2010. The KnowledgeLake Search Center and Search Web Parts are a robust and manageable extension of your SharePoint ECM solution.

Whether it’s been driven by industry or pure architecture advances each version of SharePoint has changed drastically in one way or another. In either case, it introduces complexity to the very people you are trying to convince to use SharePoint as a platform.

Despite the changes in SharePoint, the core of the KnowledgeLake product usability remains close to the original functional design. KnowledgeLake continues to fill the gap with products for SharePoint ECM in a consistent, accurate and intuitive way to capture, classify, release and search for SharePoint content.

The following sections will illustrate the customization experience and use of SharePoint 2010/2013 Search and KnowledgeLake Imaging Search.

It is important to recognize that solutions intended to serve as ECM (Enterprise Content Management) or DM (Document Management) systems have the functionality to quickly deliver content based on SharePoint managed properties. Keyword searches are beneficial; however, strong content classification is the overwhelming key to success when searching and finding targeted content. Content must be classified with meaningful names where exact input values and operator condition (=, <,>, <>) will glean precise result, e.g. ProductNumber=93452, keyword based searching will not suffice in most cases.

Both SharePoint Search and KnowledgeLake Imaging Search use SharePoint managed properties. It is interesting to note that until SharePoint 2013, it was a cumbersome and tedious task to create customized and reusable search templates natively in SharePoint that utilized managed properties. In SharePoint 2010, designers were required to use one or more of the Advanced Search web parts for searching by managed properties and the available properties were limited to a very few generic properties (figure 1.) In order to display and navigate the results you need to create a page that contains a Search Core Results, Search Paging and Refinement Panel connected web parts. Another possibility was the use of the Content Query web part, but then again, this requires configuration and for the most part, an undesirable (in terms of ECM/DM) end-user experience. Also, the Content Query web part was limited in scope to the current site collection.
In order to customize the property selection of the Advanced Search you had to have SharePoint page and web part design skills (figure 2,) as well as a background in XML programming (figure 3.) The XML programming can be tedious and is prone to errors if not well constructed. There is no debugging in the interface, so the experience was trial and error.
The designer did not have an easy way to create multiple, reusable query definitions so you were required to create a web part page and customize the XML for each query definition. The result sets were rendered in a results page (figure 4) that gave you some ability to refine, but this is not generally an acceptable format for solutions that are geared to ECM/DM. (We will further discuss the acceptable format under the section below: “Customizing Search in 2013 with KnowledgeLake Imaging”.) You could customize the display properties of the Search Core Results web part, however, again this would require a skilled designer with an expertise in XML/XSLT programming and would require a customized web part page and Search Core Results web part for each variation of the desired display format.
Customizing Search in 2013

With the release of SharePoint 2013, the ability to customize the search experience has been improved, largely due to the deeply merged FAST and SharePoint search architecture. This merger allows for an enhanced search customization experience with new features, such as the Content Search web part and Display Templates. The new Content Search web part is the successor to the functionality provided by the Content Query web part of SharePoint 2010, without the limitation of the site collection boundaries. The Content Search web part is optimized for publishing scenarios, as opposed to ECM/DM, and can be used to retrieve any content from the SharePoint index.

The Content Search web part provides a full screen query builder to create and test queries. It is integrated with Results Source and Query Rule and provides the ability to substitute query variables at query time.

Content Search Web Part

Content Search web part will allow you to show items that are results of a search query you specify. When you add the web part to a page (figure 6 – 9,) the web part will show recently modified items from the current site. This setting can be changed to show items from other site collections or list by editing the web part and changing its search criteria.

However, you will notice a single Content Search web part is limited to its ease of constructing this scope. You have to edit the PATH by hand to add additional source sites and lists, not a favorable option for a designer or end user. Multiple Content Search web parts, on multiple web part pages, may be required to deliver the required search and retrieval requirements of your ECM/DM solution.
With the tabbed ‘Build Your Query’ interface (figures 8 & 9,) you are able to control the results of the Content Search web part. The interface includes:

- **Basic** – for selecting a result source (limited to one per web part.)
- **Refiners** – for preselecting the default refiners for the web part. Additional refiners can be set at runtime.
- **Sorting** – for preselecting the default refiners for the web part.
- **Settings** – for setting load behavior, priority and query rules, etc.
- **Test** – to see what query text will be executed.

**Display Templates**

Display templates allow for the retrieval of managed properties for the rendering of results using HTML and JavaScript, an improvement over the XSLT rendering in SharePoint 2010. The Display Templates come in five layers: Control (the container for the following,) Item (the individual items in the result set,) Filter, Group and Hover Panel (the ability to refine.) (Figure 10)

The built-in SharePoint 2013 Display Templates are limited and do not provide generally acceptable functionality and features for ECM/DM solutions. In order for you to extend the built-in Display Templates you will need to be a skilled SharePoint developer with experience in Master Page design and JavaScript.

The Display Templates, live in the Master pages and page layouts gallery (figure 11,) and can be edited with a source code editor, such as Microsoft Visual Studio.NET of Expression Web. Editing the source of a Display Template is not for the timid. A strong background in HTML, XML, SharePoint and JavaScript development is a requirement. The preferred method for best results is to copy an existing Display Template to a new custom template. Assuming you have the “SharePoint Server Publishing Infrastructure” feature activated, you can view and edit an existing HTML as a base for your new custom template. See Appendix B for examples of the code needed to edit for a custom Display Template. These files can be viewed and edited from the libraries “Open with Explorer.” *Note: You’ll need the Windows WebClient service running on your development machine to use the “Open with Explorer” feature.*
Configurable in the Content Search web part are the Control and List Display Templates

- **Control**
  - List
  - List with Paging
  - Slideshow

- **Item**
  - Diagnostic
  - Large Picture
  - Picture on left, 3 lines on right
  - Picture on top, 3 lines on bottom
  - Recommended Items: Picture on left, 3 lines on right
  - Two Lines
  - Video

In addition to the Content Search web part that uses the Display Templates, are the SharePoint search family of web parts that round out with the Search Box, Search Navigation, Refinement Panel and the Search Core Results web parts.

**Search Result Type**

An important new feature of SharePoint 2013 when considering the use of the built-in search web parts is Result Types. A Result Type works with the Display Templates and property conditions to allow you to tailor the look of your results. SharePoint 2013 comes with many pre-configured Result Types and the ability to create new custom Result Types (figure 13). Result Types use rules to target specific types of content like email MSG, PDF, or Word document files.
Within a Result Type rule are associated Actions that are defined within a bound JavaScript file for the Display Template. Again, any custom Actions will require a web developer with JavaScript skill to create custom Result Types (figure 14 & 15.) See Appendix C for a sample of the PDF Result Type Display Template. Additionally, a Result Type can target specific Result Source.

**Search Result Source**

The SharePoint 2013 Result Source (figure 16) is a hybrid combination of a federated location and the successor to the SharePoint Search Scope, which has been deprecated in SharePoint 2013 (although default Search Scopes are still available for viewing and upgrade backward compatibility.) When a user executes a query the search sub system associates that query with default Result Source to provide the search results. A Result Source contains a definition that specifies a 1.) Search provider or source URL, 2.) a protocol used to get results, e.g. Local SharePoint and 3.) an optional query transform, e.g. Keyword and property filters and sorting. SharePoint 2013 comes with 16 pre-configured result sources available in all sites and site collections in web applications that consume the Search Service Application.

**Search experiences and corresponding Result Sources**

<table>
<thead>
<tr>
<th>This search experience</th>
<th>Uses this preconfigured result source</th>
</tr>
</thead>
<tbody>
<tr>
<td>Everything</td>
<td>Local SharePoint Results</td>
</tr>
<tr>
<td>People</td>
<td>Local People Results</td>
</tr>
<tr>
<td>Conversations</td>
<td>Conversations</td>
</tr>
<tr>
<td>Videos</td>
<td>Local Video Results</td>
</tr>
</tbody>
</table>
### Available Farm Level Result Sources

<table>
<thead>
<tr>
<th>This result source</th>
<th>Specifies these items in the local SharePoint index</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conversations</td>
<td>Discussions in microblogs, newsfeed posts and community sites</td>
</tr>
<tr>
<td>Documents</td>
<td>Microsoft Office documents and PDF documents</td>
</tr>
<tr>
<td>Items matching a content type</td>
<td>Items that match a content type that the incoming query specifies</td>
</tr>
<tr>
<td>Items matching a tag</td>
<td>Documents or list items that match a managed metadata term that the incoming query specifies</td>
</tr>
<tr>
<td>Items related to current user</td>
<td>Documents or list items that are related to the user in a way that the query template specifies</td>
</tr>
<tr>
<td>Local People Results</td>
<td>People items from the profile database of the User Profile service application</td>
</tr>
<tr>
<td>Local Reports and Data Results</td>
<td>Excel, Office Data Connection (ODC, ) Report Definition Language (RDL) items or items in a report library</td>
</tr>
<tr>
<td>Local SharePoint Results</td>
<td>All items from the local SharePoint search index except People items</td>
</tr>
<tr>
<td>Local Video Results</td>
<td>Videos</td>
</tr>
<tr>
<td>Pages</td>
<td>SharePoint web pages</td>
</tr>
<tr>
<td>Pictures</td>
<td>Photos and images</td>
</tr>
<tr>
<td>Popular</td>
<td>Documents and list items sorted by view count</td>
</tr>
<tr>
<td>Recently changed items</td>
<td>Documents and list items sorted by Modified date</td>
</tr>
<tr>
<td>Recommendations</td>
<td>Documents and list items that you recommend for the incoming query</td>
</tr>
<tr>
<td>Wiki</td>
<td>SharePoint wiki pages</td>
</tr>
</tbody>
</table>

### Result Source Protocols

<table>
<thead>
<tr>
<th>A result source that specifies this protocol</th>
<th>Gets search results from this search provider</th>
</tr>
</thead>
<tbody>
<tr>
<td>Local SharePoint</td>
<td>The search index of the local Search service</td>
</tr>
<tr>
<td>Remote SharePoint</td>
<td>The search index of a Search service hosted in another farm</td>
</tr>
<tr>
<td>OpenSearch 1.0/1.1</td>
<td>An external search provider (such as a remote search engine or feed) that uses the OpenSearch protocol to provide search results</td>
</tr>
<tr>
<td>Exchange</td>
<td>Exchange Web Services</td>
</tr>
</tbody>
</table>

---

*Figure 16 - Result Sources*
Figure 17 - Editing a Result Source

Figure 18 - Query Transform

Figure 19 - Result Source Query Builder
Search Query Rules
The final new search feature of SharePoint 2013 we will cover are Query Rules. The Query Rule has replaced and deprecated the search keywords and best bets functionality of SharePoint 2010. Query Rules can enable a single user search request to trigger multiple queries and return multiple Results sets.

The components of a Query Rule are:

- **Query Condition**
  - Defines the context in which the query becomes active, such as when a query:
    - Contains a specific word or words.
    - Contains a word that is specific to a dictionary.
    - Contains an action word that matches a specific phrase or term set.
    - Is common in a different source.
    - Results include a common result type.

- **Query Action**
  - Defines actions that should occur when a condition is met, such as:
    - Assigning a promoted result (similar to a Best Bet in SharePoint 2010.)
    - Creating and displaying a result block.
    - Changing the ranking of the returned results.

- **Publishing Option**
  - Decides when a query is used. This is particularly useful in commerce sites because the publishing options can set a time period on when a rule is available, such as special offer events or sales.

SharePoint 2013 has several built-in Query Rules.
Customizing Search in 2013 with KnowledgeLake Imaging

Now that we have covered the bulk of the new search customization feature of SharePoint 2013, we will take a look at the features and functionality that KnowledgeLake has been providing our customers since SharePoint 2007, with KnowledgeLake Imaging for SharePoint and the KnowledgeLake Search Center (figure 21.) KnowledgeLake Imaging for SharePoint is a powerful suite of applications that enhance SharePoint 2013 for the SharePoint administrator and most importantly, for the end users.

KnowledgeLake Imaging provides you with the correct ECM solution by enhancing native SharePoint capability for capture (scanning and client desktop assets), indexing (classification), searching, retrieving and viewing of ECM/DM content in SharePoint.

KnowledgeLake Imaging for SharePoint adds the following user-friendly functionality and features:

- Search Specific Paths and/or Result Sources
- Pre-configure and store saved searches
- Browser-based interface with multiple tabbed instances
- View document thumbnails and SharePoint properties within the search results
- Take full advantage of native SharePoint Search providers
- Build custom views to personalize the look and feel of search results
- Search by keywords or more importantly by precise managed metadata properties
- Easily Sort, Filter and Group your Search Results in a dynamic and fluid user interface
- Edit metadata directly from Results Sets
- Open Documents in their native application or with our KnowledgeLake Viewer
- Export entire document Search Result sets
- Search content of full-text PDFs directly in our KnowledgeLake Viewer
- Retrieve related and/or linked documents
- Annotate PDF and TIFF documents
- Print document directly from the Search Center
- Index documents
- E-mail documents as attachments or links
- Preform document assembly
- Start Workflows and Tasks
- Tag documents for collaboration
- Perform column lookups
- Perform cascade lookups
- Link from KnowledgeLake Edit forms to SharePoint lists

The KnowledgeLake Imaging functionality is made possible with a number of SharePoint 2013 Farm Solutions and Site Collection features. For this whitepaper, we are going to focus on the KnowledgeLake Search and View features (figure 20.)
Figure 21 - KnowledgeLake Search Center

Search Center

The KnowledgeLake Search Center is a site template that provides a hub for all your ECM/DM searching needs. The core of the Search Center is the Search Builder and Search Results (figure 21). However, the Search Center extends beyond search and retrieval to provide the end user with the tools they need, as we will illustrate below.

Search Builder

The Search Builder relies primarily on SharePoint Managed Properties to perform accurate searching and retrieving. The Search properties of the Search Builder map directly to Managed Properties that are mapped to Crawled Properties giving you the ability to have AND/OR conditions with operators. You can define the Result Columns you want displayed in your Search Results.

Keyword searching (figure 22) remains a valuable feature as it gives you the ability to perform full-text keyword searching of content within a file, and have the resultant metadata displayed in a familiar and preferred table format of the Search Results.

Figure 22 - Keyword Search for Full-Text Searching
The Source editor allows you to select one or more precise target paths for your search as well as the new SharePoint 2013 Result Source. (Figure 23 & 24)

**Search Results**

The Search Results can display multiple result tabs (figures 25) that provide document thumbnail preview (figures 26) and the ability to edit and save metadata directly from the Search Results. The toolbar (figure 27) gives you access to: Imaging User Guide, the ability to export the search result metadata to a CSV file, email the selected documents as links or attachments. Results are displayed in a familiar table layout with filtering, sorting, and drag and drop grouping functionality (figure 28 - 29.)
**Saved Searches**

Similar to the concept of Content Search web part and Display Templates in SharePoint 2013 is the Saved Search functionality of the KnowledgeLake Search Center. These powerful features empower the end user with the ability to save their easily constructed search definitions and retrieve for reuse (figure 30 – 33.) The author of the search can apply Search Visibility permissions to their saved searches as well (figure 31.)

![Figure 30 - KnowledgeLake Search Center Ribbon Bar](image)

![Figure 31 - Edit and Save Search with Search Visibility](image)

![Figure 32 - Edit, Load or Delete Saved Searches](image)

**Search Export**

KnowledgeLake Imaging Search also has a feature that gives you the ability to export a given Search Result metadata and its documents to one of many predefined export paths. The export definition is saved so you can reuse in the future. This is a valuable eDiscovery feature for sending content externally to your clients that do not have access to your internal SharePoint Farm (figures 33 – 35.)

![Figure 33 – Export Definition](image)

![Figure 34 - Saved Export Definition](image)

![Figure 35 - Export Results](image)
KnowledgeLake Search Web Parts

KnowledgeLake Imaging Search provides the site collection designer with a set of web parts to create custom search pages. The Search web parts include the KnowledgeLake Query Builder and Query Results (figure 36.) The View web part adds embedded document viewing to the page.

Figure 36 - KnowledgeLake Search Web Parts

When the KnowledgeLake Search and View web parts are added to a SharePoint web part page, they automatically recognize and connect to one another (figure 37) leaving you to merely set a few properties of the Query Builder or Search Results web part (figure 38.) When the user selects a document from the Query Results, the document will automatically display in the View web part. The Query Builder below is set to use a specific saved query template that was created in the KnowledgeLake Search Center.

There are several customization features available in the web part tool pane. For example, you can hide or disable search property fields. This allows you to further customize the behavior of the saved query. You can also customize the Search Results web part by disabling the context menu, edit and email buttons (figure 39.) Both the Query Builder and Search Results web parts can be used independently by using the “Load From Query String” feature. See Line of Business Integration below for more details.

Figure 37 - An all-in–one custom Search web part page
KnowledgeLake Viewer

The KnowledgeLake Viewer (figure 42) provides advanced SharePoint based functionality for single and multiple document viewing, printing, assembly and classification. The KnowledgeLake viewer can be launched from the Document Library (figure 40,) from the Search Results (figure 41) of the KnowledgeLake Search Center or an integration to a line of business (LOB) application. The viewer supports the following document formats. Document annotation is supported for PDF and TIFF formats.

- Adobe Portable Document Format (.PDF, PDF/A, PDF Encrypted)
- Bitmap (.BMP)
- Extensible Markup Language (.XML)
- Graphics Interchange Format (.GIF)
- Joint Photographic Experts Group (.JPG)
- Microsoft Excel (.XLS, .XLSX, and .XLSM)
- Microsoft PowerPoint (.PPT and .PPTX)
- Microsoft Word (.DOC and .DOCX)
- Microsoft Outlook (.MSG)
- Portable Network Graphics (.PNG)
- Rich Text Format (.RTF)
- Tagged Image File Format (.TIF or .TIFF)
- Text files (.TXT)
- XML Paper Specification (.XPS)

Figure 40 - Launch KnowledgeLake Viewer from Document Library

Figure 41 - Launch KnowledgeLake Viewer from the Search Results in the KnowledgeLake Search Center

Figure 42 - KnowledgeLake Viewer
**Hover Panel**

SharePoint 2013 with the addition of the Office Web Applications add-on server has introduced the Hover Panel: however, this functionality requires the use and expense of an additional Windows server and does not support PDF file types (figure 43). With KnowledgeLake Imaging you are able to view all the file types supported by the KnowledgeLake Viewer including PDF, without the addition of a Windows server and the Office Web Applications add-on (figure 44.)

*Figure 43 - SharePoint 2013 Hover Panel with Office Web Apps Server Add-on does not support PDF Viewing*

*Figure 44 - KnowledgeLake Imaging PDF Hover Panel*
Line of Business Integration
Integrating SharePoint search results with your Line of Business (LOB) systems is possible with an easily constructed URL Query String. This is a no programing approach to integration.

In the example below we are using KnowledgeLake Unify to create an integration between a Great Plains HR Employee Maintenance form and a custom SharePoint web part page. The web part page contains a KnowledgeLake Search Results and Viewer web part. The KnowledgeLake Unify button (figure 45) is configured with an action to glean the employee’s SSN directly from the field and pass it along with the query identifier to the SharePoint web part page (figure 46.) The URL Query String can also contain Keywords, additional search options like Trim Duplicates and the search properties. In the example below the result is all the documents for the employee where SSNText=944-22-9198.

![Figure 45 – Great Plains LOB Integration using KnowledgeLake Unify and URL Query String](image1)

![Figure 46 – URL Query String from LOB Integration](image2)
Conclusion
SharePoint 2013 has achieved significant advances with its new search features and continues to close the gap on customization complexities. The Display Template, Result Type, Result Source and Query Rule functionality is the latest set of tools in the SharePoint search toolbox. These new features allow you to create and extend the usability of search in SharePoint 2013. Over the next few months it will be interesting to see how the SharePoint community adopts these new features and how the search landscape of SharePoint continues to evolve.

KnowledgeLake will continue to sharpen and deliver its proven feature rich functionality for SharePoint; functionality that brings highly usable and practical ECM/DM document capture, classification, search/retrieval, viewing and line of business integration for SharePoint.

About KnowledgeLake
KnowledgeLake, Inc., headquartered in St. Louis, Missouri, is the market leader in Microsoft SharePoint ECM products and solutions. KnowledgeLake is a three-time Microsoft Partner of the Year award winner and is recognized as the founder of the SharePoint document imaging marketplace in 2003. Equity funded by PFU Ltd. (a wholly owned subsidiary of Fujitsu Ltd,) KnowledgeLake is a strong, stable and global company with currently over two million licensed users in 35 countries. KnowledgeLake is entirely focused on building a rich company culture where employee and customer satisfaction are its highest priorities.

Built entirely within Microsoft SharePoint, KnowledgeLake products enable organizations of any size to standardize on SharePoint as a powerful content platform for building and deploying rich solutions that satisfy many diverse business workloads, including: document imaging, workflow, business process management, transactional content management, document management, records management, web content management, collaboration, portals and more. Long-term content viability, open standards, information worker productivity, and compliance are the key drivers of product design.

KnowledgeLake has an industry leading R&D team assembled specifically to focus on the latest Microsoft development platforms. In addition, KnowledgeLake has teams of ECM industry veterans located throughout the US who have decades of experience architecting, selling and implementing ECM products and solutions. KnowledgeLake enables its customers to maximize and extend their already sound investments in proven Microsoft technologies, such as Microsoft Windows Server, Microsoft SQL Server, Microsoft SharePoint and Microsoft Office.

The mission of KnowledgeLake is clear and simple. “To enable our customers to realize their full potential by serving them with new and innovative document technologies.”

KnowledgeLake, Inc.
6 CityPlace Drive, Suite 500
Saint Louis, MO 63141
http://www.KnowledgeLake.com
Phone: 314.898.0500
Fax: 314.898.0501
Toll Free: 888.898.0555
Sales@KnowledgeLake.com
Appendix A – SharePoint 2010 Sample XML for Advanced Search Customization

The following XML included three additional custom managed metadata properties; BatchName, ScanUser and IndexUser.

Properties XML

```xml
<root xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <LangDefs>
    <LangDef DisplayName="Arabic" LangID="ar"/>
    <LangDef DisplayName="Bengali" LangID="bn"/>
    <LangDef DisplayName="Bulgarian" LangID="bg"/>
    <LangDef DisplayName="Catalan" LangID="ca"/>
    <LangDef DisplayName="Simplified Chinese" LangID="zh-cn"/>
    <LangDef DisplayName="Traditional Chinese" LangID="zh-tw"/>
    <LangDef DisplayName="Croatian" LangID="hr"/>
    <LangDef DisplayName="Czech" LangID="cs"/>
    <LangDef DisplayName="Danish" LangID="da"/>
    <LangDef DisplayName="Dutch" LangID="nl"/>
    <LangDef DisplayName="English" LangID="en"/>
    <LangDef DisplayName="Filipino" LangID="tl"/>
    <LangDef DisplayName="French" LangID="fr"/>
    <LangDef DisplayName="German" LangID="de"/>
    <LangDef DisplayName="Greek" LangID="el"/>
    <LangDef DisplayName="Hindi" LangID="hi"/>
    <LangDef DisplayName="Hungarian" LangID="hu"/>
    <LangDef DisplayName="Icelandic" LangID="is"/>
    <LangDef DisplayName="Indonesian" LangID="id"/>
    <LangDef DisplayName="Italian" LangID="it"/>
    <LangDef DisplayName="Japanese" LangID="ja"/>
    <LangDef DisplayName="Kannada" LangID="kn"/>
    <LangDef DisplayName="Korean" LangID="ko"/>
    <LangDef DisplayName="Lao" LangID="lo"/>
    <LangDef DisplayName="Luxembourgish" LangID="lb"/>
    <LangDef DisplayName="Latvian" LangID="lv"/>
    <LangDef DisplayName="Greek" LangID="el"/>
    <LangDef DisplayName="Lt" LangID="lt"/>
    <LangDef DisplayName="Malay" LangID="ms"/>
    <LangDef DisplayName="Malayalam" LangID="ml"/>
    <LangDef DisplayName="Marathi" LangID="mr"/>
    <LangDef DisplayName="Norwegian" LangID="no"/>
    <LangDef DisplayName="Oriya" LangID="or"/>
    <LangDef DisplayName="Pashto" LangID="ps"/>
    <LangDef DisplayName="Persian" LangID="fa"/>
    <LangDef DisplayName="Polish" LangID="pl"/>
    <LangDef DisplayName="Portuguese" LangID="pt"/>
    <LangDef DisplayName="Punjabi" LangID="pa"/>
    <LangDef DisplayName="Romanian" LangID="ro"/>
    <LangDef DisplayName="Russian" LangID="ru"/>
    <LangDef DisplayName="Slovak" LangID="sk"/>
    <LangDef DisplayName="Slovenian" LangID="sl"/>
    <LangDef DisplayName="Spanish" LangID="es"/>
    <LangDef DisplayName="Sri Lankan Sinhala" LangID="si"/>
    <LangDef DisplayName="Swedish" LangID="sv"/>
    <LangDef DisplayName="Thai" LangID="th"/>
    <LangDef DisplayName="Turkish" LangID="tr"/>
    <LangDef DisplayName="Ukrainian" LangID="uk"/>
    <LangDef DisplayName="Urdu" LangID="ur"/>
    <LangDef DisplayName="Vietnamese" LangID="vi"/>
  </LangDefs>
  <Languages>
    <Language LangRef="en"/>
    <Language LangRef="fr"/>
    <Language LangRef="de"/>
    <Language LangRef="ja"/>
    <Language LangRef="zh-cn"/>
    <Language LangRef="es"/>
    <Language LangRef="zh-tw"/>
  </Languages>
  <PropertyDefs>
    <PropertyDef Name="Path" DataType="text" Displayname="URL"/>
    <PropertyDef Name="Size" DataType="integer" DisplayName="Size (Bytes)"/>
    <PropertyDef Name="Write" DataType="datetime" DisplayName="Last Modified Date"/>
    <PropertyDef Name="FileName" DataType="text" DisplayName="Name"/>
    <PropertyDef Name="Description" DataType="text" DisplayName="Description"/>
    <PropertyDef Name="Title" DataType="text" DisplayName="Title"/>
    <PropertyDef Name="Author" DataFormat="text" DisplayName="Author"/>
    <PropertyDef Name="DocSubject" DataFormat="text" DisplayName="Subject"/>
    <PropertyDef Name="DocKeywords" DataFormat="text" DisplayName="Keywords"/>
    <PropertyDef Name="DocComments" DataFormat="text" DisplayName="Comments"/>
    <PropertyDef Name="CreatedBy" DataFormat="text" DisplayName="Created By"/>
    <PropertyDef Name="ModifiedBy" DataFormat="text" DisplayName="Last Modified By"/>
    <PropertyDef Name="owsBatchName" DataFormat="text" DisplayName="BatchName"/>
    <PropertyDef Name="owsScanUser" DataFormat="text" DisplayName="ScanUser"/>
    <PropertyDef Name="owsIndexUser" DataFormat="text" DisplayName="IndexUser"/>
  </PropertyDefs>
  <ResultTypes>
    <ResultType DisplayName="All Results" Name="default"/>
    <ResultType DisplayName="Documents" Name="documents"/>
    <ResultType DisplayName="First Document" Name="first"/>
    <ResultType DisplayName="Last Document" Name="last"/>
  </ResultTypes>
</root>
```
Appendix B – SharePoint 2013 Sample Custom Display Template

Item_ThreeLines.html

```html
<html xmlns:mso="urn:schemas-microsoft-com:office:office" xmlns:mdn="uuid:CF41010-45B3-11DA-A29F-00A0C148827">
<head>
<title>Two lines</title>
<!--[if gte mso 9]><xml>
<mso:CustomDocumentProperties>
<mso:TemplateHidden msdt:dt="string">0</mso:TemplateHidden>
<mso:ManagedPropertyMapping msdt:dt="string">'Link URL'{Link URL}:'Path','Line 1'{Line 1}:'Title','Line 2'{Line 2}:'','FileExtension','SecondaryFileExtension'</mso:ManagedPropertyMapping>
<mso:MasterPageDescription msdt:dt="string">This Item Display Template will show a small thumbnail icon next to a hyperlink of the item title, with an additional line that is available for a custom managed property.</mso:MasterPageDescription>
<mso:ContentTypeId msdt:dt="string">0x0101002039C03B61C64EC4A04F5361F385106603</mso:ContentTypeId>
<mso:TargetControlType msdt:dt="string">;#Content Web Parts;#</mso:TargetControlType>
<mso:HtmlDesignAssociated msdt:dt="string">1</mso:HtmlDesignAssociated>
<mso:_dlc_DocId msdt:dt="string">PFQ6ATWQRSYN-1-253</mso:_dlc_DocId>
</mso:CustomDocumentProperties>
</xml><![endif]--
</head>
<body>
<!-- Warning: Do not try to add HTML to this section. Only the contents of the first <div> inside the <body> tag will be used while executing Display Template code. Any HTML that you add to this section will NOT become part of your Display Template. -->
<script>
includeLanguageScript(this.url, "sitecollection/_catalogs/masterpage/Display Templates/Language Files/(Locales)/CustomStrings.js");
</script>
<!-- Use the div below to author your Display Template. Here are some things to keep in mind:
* Surround any JavaScript logic as shown below using a "pound underscore" (#_ ... _#) token inside a comment.
* Use the values assigned to your variables using an "underscore pound equals" (_#= ... =#) token. -->
<div id="TwoLines">
<!-- -->
var encodedId = $htmlEncode(ctx.ClientControl.get_nextUniqueId() + "_2lines_");
var linkURL = $getItemValue(ctx, "Link URL");
linkURL.overrideValueRenderer($urlHtmlEncode);
var iconURL = Srch.ContentBySearch.getIconSourceFromItem(ctx.CurrentItem);
var line1 = $getItemValue(ctx, "Line 1");
var line2 = $getItemValue(ctx, "Line 2");
line1.overrideValueRenderer($contentLineText);
line2.overrideValueRenderer($contentLineText);
var containerId = encodedId + "container";
var pictureLinkId = encodedId + "pictureLink";
var pictureId = encodedId + "picture";
var dataContainerId = encodedId + "dataContainer";
var line1LinkId = encodedId + "line1Link";
var line1Id = encodedId + "line1";
var line2Id = encodedId + "line2";
<div class="cbs-Item" id="_#= containerId =#_">
  <a class="cbs-ItemLink" title="_#= $htmlEncode(line1.defaultValueRenderer(line1)) =#_" id="_#= pictureLinkId =#_">
    <img class="cbs-Thumbnail" src="_#= $urlHtmlEncode(iconURL) =#_" alt="_#= $htmlEncode(line1.defaultValueRenderer(line1)) =#_" id="_#=
        pictureId =#_" />
  </a>
  <div class="cbs-Detail" id="#_#" dataContainerId="#_#">
    <a class="cbs-lineLink as-noWrap ms-displayLink" href="#_#" linkURL="#_#" title="#_# $htmlEncode(line1.defaultValueRenderer(line1))="#_#" line1="#_#" line2="#_#"></a>
  </div>
</div>
</div>
</body>
</html>
```
Appendix C – SharePoint 2013 PDF Display Template for the PDF Result Type

Item_PDF.js

function ULSVoQ(){var o=new Object;o.ULSTeamName="Search Server";o.ULSFileName="Item_PDF.js";return o;}

function DisplayTemplate_176b8fde3fc4107a1b3b244a43f9f2(ctx) {ULSVoQ; var ms_outHtml=[]; var cachePreviousTemplateData = ctx['DisplayTemplateData']; ctx['DisplayTemplateData'] = new Object(); DisplayTemplate_176b8fde3fc4107a1b3b244a43f9f2.DisplayTemplateData = ctx['DisplayTemplateData']; ctx['DisplayTemplateData']['TargetControlType']="SearchResults"; ctx['DisplayTemplateData']['ManagedPropertyMapping']={'Title':"Title", 'Path':"Path", 'Description':"Description", 'EditorOWSUSER':"EditorOWSUSER", 'LastModifiedTime':"LastModifiedTime", 'CollapsingStatus':"CollapsingStatus", 'DocId':"DocId", 'HitHighlightedSummary':"HitHighlightedSummary", 'HitHighlightedProperties':"HitHighlightedProperties", 'FileExtension':"FileExtension", 'ViewsLifeTime':"ViewsLifeTime", 'ParentLink':"ParentLink", 'FileType':"FileType", 'lContainer':"lContainer", 'SecondaryFileExtension':"SecondaryFileExtension", 'DisplayAuthor':"DisplayAuthor"}; var cachePreviousItemValuesFunction = ctx['ItemValues']; ctx['ItemValues'] = function(slotOrPropName) {ULSVoQ; return Srch.ValueInfo.getCachedCtxItemValue(ctx, slotOrPropName);};

ms_outHtml.push('','');

if(!$isNull(ctx.CurrentItem) && (!$isNull(ctx.ClientControl)))
    {var id = ctx.ClientControl.get_nextUniqueId(); var itemId = id + Srch.U.Ids.item; var hoverUrl = "~sitecollection\_catalogs\masterpage\Display Templates\Search\Item_PDF_HoverPanel.js"; ctx.CurrentItem.csr_Path = ctx.CurrentItem.Path + "#search=" + $urleValueEncode(k); ctx.CurrentItem.csr_Icon = Srch.U.getIconUrlByFileExtension(ctx.CurrentItem); ctx.currentItem_ShowHoverPanelCallback = Srch.U.getShowHoverPanelCallback(itemId, hoverId, hoverUrl);
ms_outHtml.push(''"'"');

ctx['ItemValues'] = cachePreviousItemValuesFunction;
ctx['DisplayTemplateData'] = cachePreviousTemplateData;
return ms_outHtml.join('');}

function RegisterTemplate_176b8fde3fc4107a1b3b244a43f9f2() {ULSVoQ; if (typeof(RegisterModuleInit) == "function" && typeof(Srch.U.replaceUrlTokens) == "function") {
    RegisterModuleInit(Srch.U.replaceUrlTokens("~sitecollection\_catalogs\masterpage\Display Templates\Search\Item_PDF.js"), RegisterTemplate_176b8fde3fc4107a1b3b244a43f9f2);}

if (typeof(RegisterModuleInit) == "function") {
    RegisterModuleInit("~sitecollection\_catalogs\masterpage\Display Templates\Search\Item_PDF.js");
}