



**PaperVision® Enterprise
Service API Guide**

Version R82

Information in this document is subject to change without notice and does not represent a commitment on the part of Digitech Systems, LLC. The software described in this document is furnished under a license agreement or nondisclosure agreement. The software may be used or copied only in accordance with the terms of the agreement. It is against the law to copy the software on any medium except as specifically allowed in the license or nondisclosure agreement. No part of this manual may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying and recording, for any purpose without the express written permission of Digitech Systems, LLC.

Copyright © 2016 Digitech Systems, LLC. All rights reserved.
Printed in the United States of America.

PaperFlow, PaperFlow NOW! and the Digitech Systems, LLC logo
are trademarks of Digitech Systems, LLC.

PaperVision Enterprise, ImageSilo, and PaperFlow
are registered trademarks of Digitech Systems, LLC.

Microsoft, Windows, SQL Server, Access, and .NET Framework are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

All other trademarks and registered trademarks are the property of their respective owners.

PaperVision Enterprise contains text search technology provided under license
by dtSearch®. All rights reserved.

PaperVision Enterprise contains technology under license
from Autonomy, Inc. All rights reserved.

PaperVision Enterprise contains portions of imaging code owned and copyrighted
by Pegasus Imaging Corporation, DBA Accusoft Pegasus, Tampa, FL. All rights reserved.

Digitech Systems, LLC.
8400 E. Crescent Parkway, Suite 500
Greenwood Village, CO 80111
Phone: 303.493.6900 Fax: 303.493.6979
www.digitechsystems.com

Using This Guide

Welcome to the *PaperVision Enterprise Service API Guide*. This guide describes how you can use Digitech Systems' application programming interface (API) in your applications to perform tasks such as searching for, and retrieving, any document stored in PaperVision Enterprise. The guide walks you through getting started and provides specifications for the API.

This guide is written for software developers who are familiar with general programming practices and techniques.

Getting Product Support

Digitech Systems, LLC takes great pride in providing software that is extremely simple to use, feature-rich, and flexible both in architecture and in integration capabilities. If, for some reason, you experience any difficulty, please do not hesitate to contact us to experience our legendary customer service toll free at (877) 374-3569 or direct at (402) 484-7777. Additionally, you may contact us at support@digitechsystems.com. Our technical support staff is available to serve you between the hours of 8:00 AM and 6:00 PM Central Time.

Digitech Systems' product features and enhancements are driven by you, our customer. If you have an idea for a future product feature or enhancement, please send an e-mail to dev@digitechsystems.com.

In addition to our legendary technical support staff, we also offer the following resources to help answer your questions and resolve your issues.

- **MyDSI** - MyDSI is an interactive tool for all Digitech Systems customers. Log on to MyDSI at <http://mydsi.digitechsystems.com> to download product updates, license purchased software, view support contract renewals, and check the status of your software support cases and requests.
- **User Forums** - Log in to the User Forums at <http://forums.digitechsystems.com> to exchange answers and ideas with other users in our moderated community.
- **Knowledge Base** - Log in to search our extensive Knowledge Base at <http://kb.digitechsystems.com> for articles on all Digitech Systems products

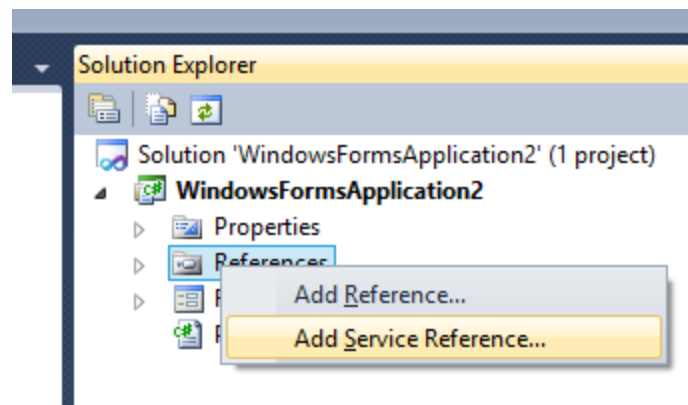
Getting Started

The service APIs are based on the Simple Object Access Protocol (SOAP) 1.1 extensible markup language (XML) (<http://www.w3.org/TR/2000/NOTE-SOAP-20000508/>). You can work with the interface using the programming language of your choice and with the wide array of available development tools that support SOAP 1.1.

For demonstration, the following steps illustrate how to configure Microsoft Visual Studio to use the PaperVision Enterprise document search service. (You can download a free version of Visual Studio Express here: <http://www.visualstudio.com/en-US/products/visual-studio-express-vs>.)

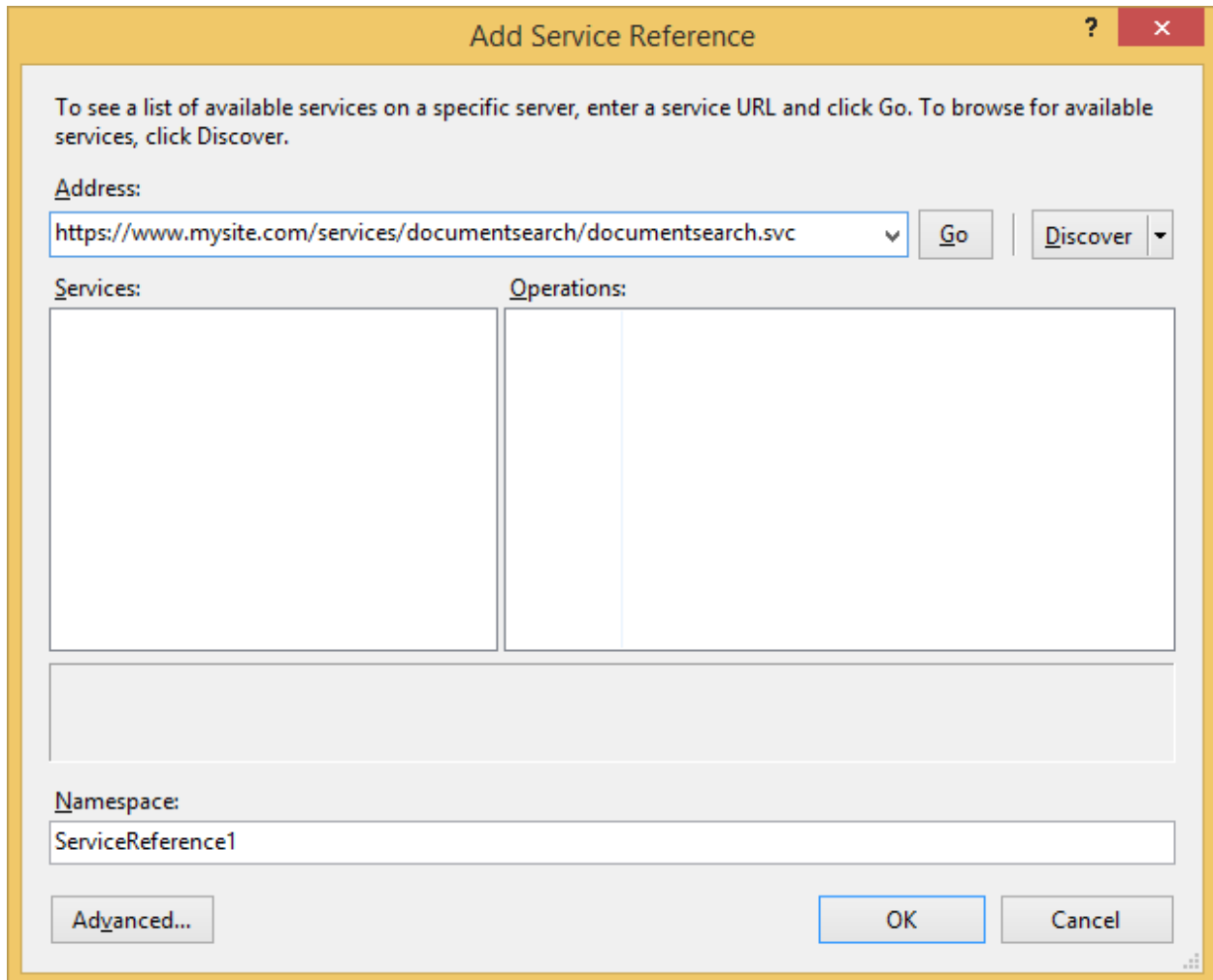
Note: These steps assume that you are familiar with Microsoft Visual Studio and that you already installed PaperVision Enterprise and know the configured address of the service (e.g., <http://www.mysite.com/services/documentsearch/documentsearch.svc>).

Step 1: In your Visual Studio project, within the **Solution Explorer** window, right-click on **References** and select **Add Service Reference**.



Step 2: In the **Add Service Reference** window, type in the document search service address, click **Go**, and then click **OK** to add the reference to your project.

Note: If you are planning to call the API available on ImageSilo at <https://www.imagesilo.com/services/documentsearch/documentsearch.svc>, you will still need to install PVE in a test environment in order to auto-generate code.



The image shows a Windows-style dialog box titled "Add Service Reference". It has a yellow title bar with a question mark icon and a close button. The main area contains instructions: "To see a list of available services on a specific server, enter a service URL and click Go. To browse for available services, click Discover." Below this is an "Address:" label followed by a text box containing the URL "https://www.mysite.com/services/documentsearch/documentsearch.svc". To the right of the text box are two buttons: "Go" and "Discover". Below the "Address:" section are two empty rectangular boxes labeled "Services:" and "Operations:". Below these is a "Namespace:" label followed by a text box containing the text "ServiceReference1". At the bottom of the dialog are three buttons: "Advanced...", "OK", and "Cancel".

Add Service Reference

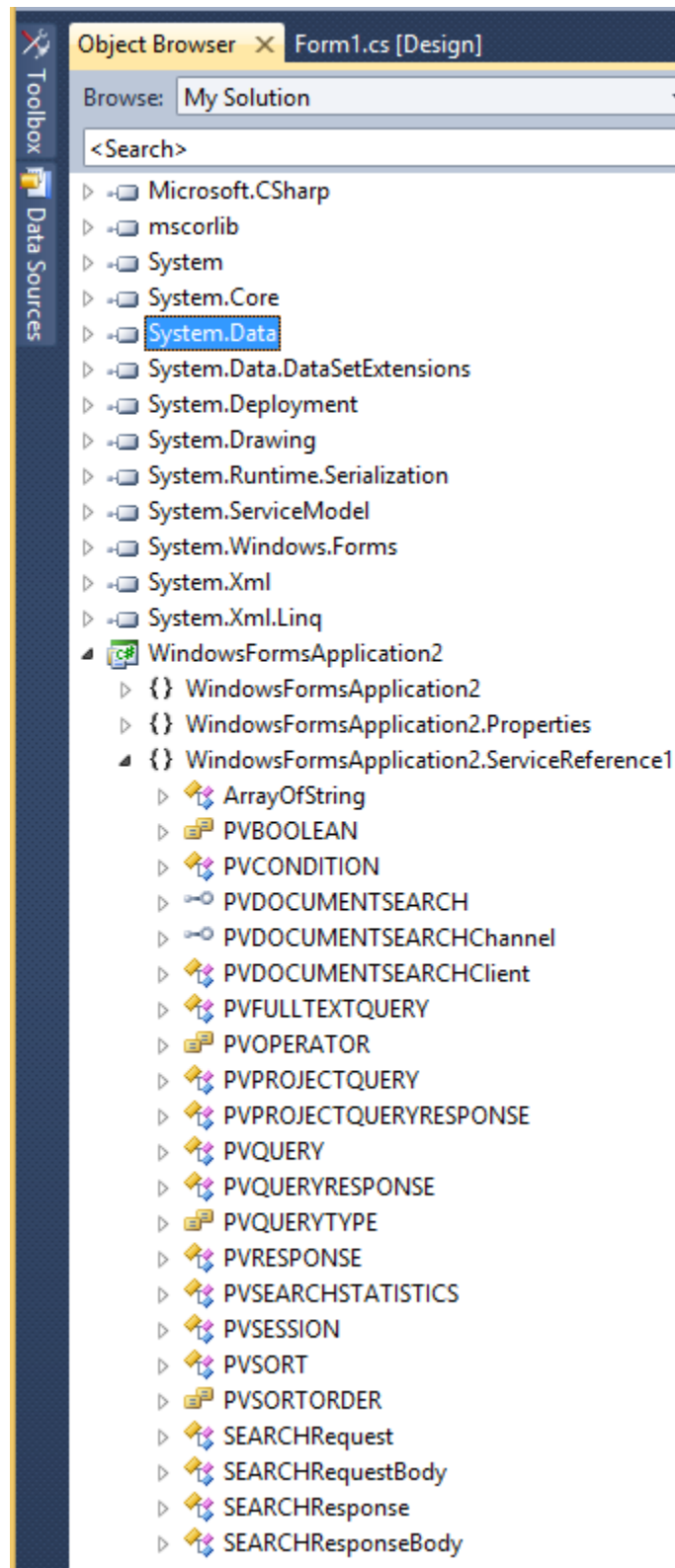
To see a list of available services on a specific server, enter a service URL and click Go. To browse for available services, click Discover.

Address:

Services: Operations:

Namespace:

Step 3: Visual Studio will automatically add the document search object hierarchy to your project as shown in **Object Browser**.



Using PaperVision Enterprise Services

Use of the service proxy and other code auto-generated by Visual Studio (as shown in the previous section), simplifies making calls to a PaperVision Enterprise service. Under the hood, a service is called via an HTTP POST and is composed of an XML SOAP envelope (i.e., the content of the service request). Whether you choose to use a tool such as Visual Studio or to otherwise manually/programmatically compose requests, understanding the underlying XML can help you leverage the full potential of a particular service. The following sections describe the XML requests for PaperVision Enterprise services.

Calling the Document Search Service

A complete, sample document search request XML envelope is listed at the end of this section; you can use the sample as a starting point for manually or programmatically composing a search query.

A search is composed of two primary elements: a session (PVSESSION) and a query (PVQUERY). The following sections describe these XML elements (and the corresponding types auto-generated by Visual Studio).

PVSESSION

The PVSESSION element contains information describing a PaperVision Enterprise session, which is required to search for documents. The sub-elements of PVSESSION are listed in the following table.

Note: To obtain a session, log into PaperVision Enterprise programmatically using the LoginUserEx3 HTTPInterface.aspx XML web service. In addition to calling LoginUserEx3, you must also call LoadSystemAccess on HTTPInterface.aspx. After you have finished using a PaperVision Enterprise session, be sure to call KillSession on HTTPInterface.aspx. For more information on how to use HTTPInterface.aspx, refer to the *PaperVision Enterprise API Reference Guide*.

PVSESSION			
Element	Type	Required?	Description
SESSIONID	String	Yes	Session ID obtained from HTTPInterface.aspx LoginUserEx3 and LoadSystemAccess calls
ENTITYID	Int	Yes	Entity ID
SOURCEIPADDRESS	String	No	Source IP address used in HTTPInterface.aspx LoginUserEx3 call

DEVELOPERID	String	No (currently reserved for future use)	Developer ID obtained from Digitech Systems
-------------	--------	--	---

PVQUERY

The PVQUERY element contains an array of PVPROJECTQUERY. Each PVPROJECTQUERY contains information about a project-specific search. You can execute multiple searches simultaneously; the array of PVPROJECTQUERY can contain multiple searches for the same project or span several projects. PVPROJECTQUERY includes the following sub-elements of note:

- PVFIELD – an index or metadata to be returned in the result set
- PVCONDITION - a single search criterion or logically combined group of search criteria
- PVSORT - how the result set is to be ordered
- PVFULLTEXTQUERY - optionally specifies full-text search parameters

The sub-elements of PVPROJECTQUERY, PVFIELD, PVCONDITION, PVSORT, and PVFULLTEXTQUERY are listed in the following tables.

PVPROJECTQUERY			
Element	Type	Required?	Description
CALLERQUERYID	String	No	Pass-through ID that you provide with each project search; this same ID is returned with each project result set so that you can easily correlate your project search requests with results (e.g., in the case where you want to execute two searches on the same project, assign each a unique CALLERQUERYID)
PROJECTID	Long	Yes	Project ID
FIELDSTORETURN	array of PVFIELD	No	One or more PVFIELDS that specify the document index values and/or metadata to be returned in the result set
CONDITION	PVCONDITION	Yes	You must provide a PVCONDITION if you do not provide a PVFULLTEXTQUERY; you can specify both a PVCONDITION and a PVFULLTEXTQUERY
SORTFIELDS	array of PVSORT	No	One or more PVSORTs that specify how the results are to be ordered
FULLTEXTQUERY	PVFULLTEXTQUERY	No	Full-text search parameters

COMPILEDQUERY	String	No	To improve performance when paging results, populate COMPILEDQUERY with the corresponding COMPILEDQUERY returned from the previous page (so the search service does not have to re-compile your query for each subsequent page)
PAGESIZE	Int	No	When paging results, the number of document records to return per page (must be consistent for all pages); default value is “0” to specify that paging is not requested
PAGEKEY	String	No	When paging results, PAGEKEY is required for all subsequent pages following the first; populate PAGEKEY with the value of PAGENEXTKEY returned with the previous page
IGNOREERRORS	Bool	No	Set to true to ignore recoverable errors such as specifying an index field name that does not exist in a project or for which you do not have permission to view; default value is “false”
RETURNCOUNTONLY	Bool	No	Set to true to receive only a count (RESULTCOUNT) of matching results (result data will be omitted); paging-related options (COMPILEDQUERY, PAGESIZE, PAGEKEY) will be ignored; default value is “false”
RETURNSEARCHSTATISTICS	Bool	No	Set to true to receive performance statistics as part of the results; default value is “false”

PVFIELD			
Element	Type	Required?	Description
FIELDNAME	String	Yes	Document index name or one of the following document metadata constants: DOCID DOCCREATEDATE DOCDUPEID DOCSECURELIST DOCCKEOUTTIME DOCCKEOUTUSERID DOCCKEOUTUSERNAME DOCCKEOUTUSERFULLNAME

			DOCSOURCEDOCID DOCDGID DOCDGNAME DOCDGORIGNAME DOCFTINDEX DOCRECRETLOCKDATE DOCRECRETDESTDATE DOCRECRETDESTLISTID DOCRECRETLOCKINFO DOCRECRETDESTINFO OBJID OBJTYPE OBJHASHTYPE OBJPAGES OBJVERSIONNUM OBJVERSIONTIME OBJVERSIONCREATOR OBJTRASHUSERID OBJTRASHUSERNAME OBJTRASHDATETIME DOCSIGID DOCSIGNAME DOCSIGUSERID DOCSIGDATETIME DOCSIGSOURCEIP DOCSIGUSERNAME DOCSIGUSERFULLNAME DOCFOLDERID DOCFOLDERNAME DOCWFDEFID DOCWFINSTID DOCWFSTARTTIME DOCWFENDTIME DOCWFNAME
HASH	Bool	No	Set to true to receive a hash calculation that includes the returned index value; the hash is useful for quickly identifying and potentially grouping results that have the same returned index values; default value is “false”

PVCONDITION			
Element	Type	Required?	Description
FIELDNAME	String	No	Document index name or one of the following document metadata constants: DOCID DOCCREATEDATE

			DOC DUPE ID DOC SECURE LIST DOC CHECK OUT TIME DOC CHECK OUT USER ID DOC SOURCE DOC ID DOC D G ID DOC FT INDEX DOC RECRET LOCK DATE DOC RECRET DEST DATE DOC RECRET DEST LIST ID OBJ ID OBJ TYPE OBJ HASHTYPE OBJ PAGES OBJ VERSION NUM OBJ VERSION TIME OBJ VERSION CREATOR OBJ TRASH USER ID OBJ TRASH DATE TIME DOC SIG ID DOC SIG NAME DOC SIG USER ID DOC SIG DATE TIME DOC SIG SOURCE IP DOC FOLDER ID DOC WF DEF ID DOC WF INST ID DOC WF START TIME DOC WF END TIME
OPERATOR	PV OPERATOR	No	Operator to be used when comparing FIELDNAME to QUERYVALUE; one of the following constants: EQUAL NOTEQUAL GREATER THAN GREATER THAN OR EQUAL LESS THAN LESS THAN OR EQUAL IS NULL IS NOT NULL LIKE NOT LIKE IN NOT IN
QUERYVALUE	String	No	Value to be used in the condition; can include SQL wildcards such as “%”; supports [CURRENTDATETIME] tag (e.g., “[CURRENTDATETIME+4H]” would resolve to the current date/time

			<p>GMT plus 4 hours) where you can specify “+” or “-“ with the following:</p> <p>“Y” – year “M” – month “D” – day “H” – hour “N” – minute “S” – second</p>
CONDITIONGROUP	Bool	No	PVCONDITION can take the form of a query filter (in which case, FIELDNAME, OPERATOR, and QUERYVALUE should be appropriately populated) or a group of two or more filters joined by a logical connector; in the latter case set CONDITIONGROUP to “true,” set the CONDITIONCONNECTOR to the Boolean connector type and populate CONDITIONS array with the filters to be evaluated per CONDITIONCONNECTOR; default value is “false”
CONDITIONCONNECTOR	PVBOOLEAN	No	<p>Either of the following constants:</p> <p>AND OR</p>
CONDITIONS	array of PVCONDITION	No	Populate with PVCONDITIONs when CONDITIONGROUP is set to “true”

PVSORT			
Element	Type	Required?	Description
FIELDNAME	String	Yes	<p>Document index name or one of the following document metadata constants:</p> <p>DOCID DOCCREATEDATE DOCDUPEID DOCCKEOUTTIME DOCCKEOUTUSERID DOCCKEOUTUSERNAME DOCCKEOUTUSERFULLNAME DOCSOURCEDOCID DOCDGID DOCDGNAME DOCDGORIGNAME DOCFTINDEX DOCRECRETLOCKDATE DOCRECRETDESTDATE</p>

			DOCRECRETDESTLISTID OBJID OBJTYPE OBJHASHTYPE OBJPAGES OBJVERSIONNUM OBJVERSIONTIME OBJVERSIONCREATOR OBJTRASHUSERID OBJTRASHUSERNAME OBJTRASHDATETIME DOCSIGID DOCSIGNAME DOCSIGUSERID DOCSIGUSERNAME DOCSIGUSERFULLNAME DOCSIGDATETIME DOCSIGSOURCEIP DOCFOLDERID DOCFOLDERNAME DOCWFDEFID DOCWFINSTID DOCWFSTARTTIME DOCWFENDTIME DOCWFNAME
SORTORDER	PVSORTORDER	Yes	Either of the following constants: ASCENDING DESCENDING

PVFULLTEXTQUERY			
Element	Type	Required?	Description
QUERYTYPE	PVQUERYTYPE	Yes	One of the following constants: ALLWORDS ANYWORDS BOOLEAN
QUERYVALUE	String	Yes	Value to be used for the search; interpreted based on the value of QUERYTYPE; can contain wildcard characters “*” (matches any number of characters) and “?” (matches any single character)
FUZZINESS	Int	No	Adjust from “0” to “10” (default is “0”) to find search terms even if misspelled; search for “alphabet” with FUZZINESS of “1” would find “alphaqet,” with FUZZINESS of “3” would find both “alphaqet” and “alpkaqet”

PHONIC	Bool	No	Set to "true" to find words that sound alike, like "Smythe" in a search for "Smith"; default is "false"
STEMMING	Bool	No	Set to "true" to find variations on endings, like "applies," "applied," or "applying" in a search for "apply"; default is "false"
NATURALLANGUAGE	Bool	No	Set to "true" to support "plain English" unstructured queries; default is "false"
SYNONYMSALLWORDS	Bool	No	Set to "true" to find synonyms for all search terms, otherwise only words marked with the synonym searching character (&) will be matched; must be used in combination with SYNONYMSUSER, SYNONYMSWORDNET, and/or SYNONYMSWORDNETRELATED; default is "false"
SYNONYMSUSER	Bool	No	Set to "true" to find synonyms for search terms based on a user-defined thesaurus (e.g., search for "fast" would also find "quick"); default is "false"
SYNONYMSWORDNET	Bool	No	Set to "true" to find synonyms for search terms based on the built-in thesaurus (e.g., search for "fast" would also find "quick"); default is "false"
SYNONYMSWORDNETRELATED	Bool	No	Set to "true" to find synonyms and other related words for search terms based on the built-in thesaurus (e.g., search for "fast" would also find "quick"); default is "false"
RETURNTITLE	Bool	No	Set to "true" to return the title of each document in the results; default is "false"
RETURNSTATISTICS	Bool	No	Set to "true" to return the number of search hits, relevance score, and relevance score percent for each document in the results; default is "false"
QUERYAFTERINDICES	Bool	No	Set to "true" to execute the full-text search after querying document indices and/or metadata; default is "false" (by default, full-text search is executed first)

Sample Document Search Request XML Envelope

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
                  xmlns="http://www.digitechsystems.com">
  <soapenv:Header/>
  <soapenv:Body>
    <SEARCH>

      <SESSION>
        <SESSIONID>10F407AA322C7346619CB9BDE29BEBF8</SESSIONID>
        <ENTITYID>1</ENTITYID>
        <SOURCEIPADDRESS>127.0.0.1</SOURCEIPADDRESS>
        <DEVELOPERID></DEVELOPERID>
      </SESSION>

      <QUERY>

        <PROJECTQUERIES>

          <PVPROJECTQUERY>
            <CALLERQUERYID>MyIDToBeReturned</CALLERQUERYID>
            <PROJECTID>1</PROJECTID>
            <FIELDSTORETURN>

              <!-- general metadata that can be returned (in addition to index values)
                DOCCREATEDATE
                DOCDUPEID
                DOCSECURELIST
                DOCCKEOUTTIME
                DOCCKEOUTUSERID
                DOCCKEOUTUSERNAME
                DOCCKEOUTUSERFULLNAME
                DOCSOURCEDOCID
                DOCDGID
                DOCDGNAME
                DOCDGORIGNAME
                DOCFTINDEX
                DOCRECRETLOCKDATE
                DOCRECRETDESTDATE
                DOCRECRETDESTLISTID
                DOCRECRETLOCKINFO
                DOCRECRETDESTINFO
                OBJID
                OBJTYPE
                OBJHASHTYPE
                OBJPAGES
                OBJVERSIONNUM
                OBJVERSIONTIME
                OBJVERSIONCREATOR
                OBJTRASHUSERID
                OBJTRASHUSERNAME
                OBJTRASHDATETIME
                DOCSIGID
                DOCSIGNAME
                DOCSIGUSERID
                DOCSIGDATETIME
                DOCSIGSOURCEIP
```

```

DOCSIGUSERNAME
DOCSIGUSERFULLNAME
DOCFOLDERID
DOCFOLDERNAME
DOCWFDEFID
DOCWFINSTID
DOCWFSTARTTIME
DOCWFENDTIME
DOCWFNAME
-->

<!-- return index named "ABC" and number of pages -->
<PVFIELD>
  <FIELDNAME>ABC</FIELDNAME>
</PVFIELD>
<PVFIELD>
  <FIELDNAME>OBJPAGES</FIELDNAME>
</PVFIELD>

<!-- return indices named "ABC" and "EFG" -->
<!-- include value of "EFG" in returned hash for document grouping -->
<!-- all docs with same value for "EFG" will have the same returned hash-->
<!--
  <PVFIELD>
    <FIELDNAME>ABC</FIELDNAME>
  </PVFIELD>
  <PVFIELD>
    <FIELDNAME>EFG</FIELDNAME>
    <HASH>true</HASH>
  </PVFIELD>
-->
</FIELDSTORETURN>

<!-- fields that can be used in conditions (in addition to indices)
DOCID
DOCCREATEDATE
DOCDUPEID
DOCSECURELIST
DOCCKEOUTTIME
DOCCKEOUTUSERID
DOCSOURCEDOCID
DOCDGID
DOCFTINDEX
DOCRECRETLOCKDATE
DOCRECRETDESTDATE
DOCRECRETDESTLISTID
OBJID
OBJTYPE
OBJHASHTYPE
OBJPAGES
OBJVERSIONNUM
OBJVERSIONTIME
OBJVERSIONCREATOR
OBJTRASHUSERID
OBJTRASHDATETIME
DOCSIGID
DOCSIGNAME
DOCSIGUSERID

```



```

DOCSIGDATETIME
DOCSIGSOURCEIP
DOCFOLDERID
DOCWFDEFID
DOCWFINSTID
DOCWFSTARTTIME
DOCWFENDTIME
-->

<!-- single condition -->
<CONDITION>
  <FIELDNAME>DOCID</FIELDNAME>
  <OPERATOR>ISNOTNULL</OPERATOR>
  <QUERYVALUE></QUERYVALUE>
</CONDITION>

<!-- a AND b -->
<!--
  <CONDITION>
    <CONDITIONGROUP>true</CONDITIONGROUP>
    <CONDITIONCONNECTOR>AND</CONDITIONCONNECTOR>
    <CONDITIONS>
      <PVCONDITION>
        <FIELDNAME>field00</FIELDNAME>
        <OPERATOR>LIKE</OPERATOR>
        <QUERYVALUE>%abcd%</QUERYVALUE>
      </PVCONDITION>
      <PVCONDITION>
        <FIELDNAME>field01</FIELDNAME>
        <OPERATOR>GREATERTHAN</OPERATOR>
        <QUERYVALUE>10000</QUERYVALUE>
      </PVCONDITION>
    </CONDITIONS>
  </CONDITION>
-->

<!-- a OR (b AND c AND (d OR e)) -->
<!--
  <CONDITION>
    <CONDITIONGROUP>true</CONDITIONGROUP>
    <CONDITIONCONNECTOR>OR</CONDITIONCONNECTOR>
    <CONDITIONS>
      <PVCONDITION>
        <FIELDNAME>field00</FIELDNAME>
        <OPERATOR>EQUAL</OPERATOR>
        <QUERYVALUE>text content abc</QUERYVALUE>
      </PVCONDITION>
      <PVCONDITION>
        <CONDITIONGROUP>true</CONDITIONGROUP>
        <CONDITIONCONNECTOR>AND</CONDITIONCONNECTOR>
        <CONDITIONS>
          <PVCONDITION>
            <FIELDNAME>field07</FIELDNAME>
            <OPERATOR>ISNOTNULL</OPERATOR>
            <QUERYVALUE/>
          </PVCONDITION>
          <PVCONDITION>
            <FIELDNAME>field03</FIELDNAME>

```

```

        <OPERATOR>EQUAL</OPERATOR>
        <QUERYVALUE>>true</QUERYVALUE>
        </PVCONDITION>
        <PVCONDITION>
        <CONDITIONGROUP>true</CONDITIONGROUP>
        <CONDITIONCONNECTOR>OR</CONDITIONCONNECTOR>
        <CONDITIONS>
            <PVCONDITION>
            <FIELDNAME>field00</FIELDNAME>
            <OPERATOR>LIKE</OPERATOR>
            <QUERYVALUE>%abcdef%</QUERYVALUE>
            </PVCONDITION>
            <PVCONDITION>
            <FIELDNAME>field01</FIELDNAME>
            <OPERATOR>GREATERTHANOREQUAL</OPERATOR>
            <QUERYVALUE>10000</QUERYVALUE>
            </PVCONDITION>
        </CONDITIONS>
        </PVCONDITION>
    </CONDITIONS>
</CONDITION>
-->

```

<!-- fields that can be used in sorts (in addition to indices)

```

DOCID
DOCCREATEDATE
DOCDUPEID
DOCCKECKOUTTIME
DOCCKECKOUTUSERID
DOCCKECKOUTUSERNAME
DOCCKECKOUTUSERFULLNAME
DOCSOURCEDOCID
DOCDGID
DOCDGNAME
DOCDGORIGNAME
DOCFTINDEX
DOCRECRETLOCKDATE
DOCRECRETDESTDATE
DOCRECRETDESTLISTID
OBJID
OBJTYPE
OBJHASHTYPE
OBJPAGES
OBJVERSIONNUM
OBJVERSIONTIME
OBJVERSIONCREATOR
OBJTRASHUSERID
OBJTRASHUSERNAME
OBJTRASHDATETIME
DOCSIGID
DOCSIGNAME
DOCSIGUSERID
DOCSIGUSERNAME
DOCSIGUSERFULLNAME
DOCSIGDATETIME
DOCSIGSOURCEIP

```

```

        DOCFOLDERID
        DOCFOLDERNAME
        DOCWFDEFID
        DOCWFINSTID
        DOCWFSTARTTIME
        DOCWFENDTIME
        DOCWFNAME
    -->

    <!--
    <SORTFIELDS>
        <PVSORT>
            <FIELDNAME>DOCCREATEDATE</FIELDNAME>
            <SORTORDER>DESCENDING</SORTORDER>
        </PVSORT>
    </SORTFIELDS>
    -->

    <!--
    <FULLTEXTQUERY>
        <QUERYTYPE>ALLWORDS</QUERYTYPE>
        <QUERYVALUE>exceptional</QUERYVALUE>
        <FUZZINESS>0</FUZZINESS>
        <PHONIC>>false</PHONIC>
        <STEMMING>>false</STEMMING>
        <NATURALLANGUAGE>>false</NATURALLANGUAGE>
        <SYNONYMSUSER>>false</SYNONYMSUSER>
        <SYNONYMSWORDNET>>false</SYNONYMSWORDNET>
        <SYNONYMSWORDNETRELATED>>false</SYNONYMSWORDNETRELATED>
        <RETURNRTITLE>>true</RETURNRTITLE>
        <RETURNSTATISTICS>>true</RETURNSTATISTICS>
        <QUERYAFTERINDICES>>false</QUERYAFTERINDICES>
    </FULLTEXTQUERY>
    -->

    <COMPILEDQUERY></COMPILEDQUERY>
    <PAGESIZE>0</PAGESIZE>
    <PAGEKEY></PAGEKEY>
    <IGNOREERRORS>>false</IGNOREERRORS>
    <RETURNCOUNTONLY>>false</RETURNCOUNTONLY>
    <RETURNSEARCHSTATISTICS>true</RETURNSEARCHSTATISTICS>

    </PVPROJECTQUERY>

    </PROJECTQUERIES>

    </QUERY>
</SEARCH>
</soapenv:Body>
</soapenv:Envelope>

```

Receiving Document Search Results

A call to the document search service will return either a SOAP fault or a PVQUERYRESPONSE. Similar to submitting a request to the service, the response contains an array of PVPROJECTQUERYRESPONSE, where a PVPROJECTQUERYRESPONSE is returned for each project query. The following table lists the sub-elements of PVPROJECTQUERYRESPONSE.

| PVPROJECTQUERYRESPONSE | | |
|------------------------|--------------------|--|
| Element | Type | Description |
| CALLERQUERYID | String | Same pass-through ID that you provided with the corresponding project search |
| PROJECTID | Long | Project ID |
| SEARCHRESULT | String | XML result set |
| RESULTCOUNT | Int | Number of documents in the result set |
| COMPILEDQUERY | String | A compiled representation of the query that you submitted; to improve performance when paging, submit the unaltered COMPILEDQUERY as part of the PVPROJECTQUERY for all subsequent pages |
| PAGENEXTKEY | String | When paging results, submit the value of PAGENEXTKEY as the PAGEKEY in PVPROJECTQUERY to obtain the next page of results |
| SEARCHSTATISTICS | PVSEARCHSTATISTICS | Search performance statistics |

| PVSEARCHSTATISTICS | | |
|--------------------|--------|--|
| Element | Type | Description |
| COMPILE | String | Time required to compile the search query |
| EXECUTE | String | Time required to execute the SQL search (if any) |
| EXECUTEFULLTEXT | String | Time required to execute the full-text search (if any) |
| BUILDRESULTS | String | Time required to build search results |
| TOTAL | String | Total time required to service the search request (Note: this time may not be the sum of the other times returned in PVSEARCHSTATISTICS; time for other minor operations may be included in the TOTAL) |

Calling the File Service

To call the file service, compose a PVFILEREQUEST element as described in the following section. Also, a sample file service request XML envelope is provided for you to use as a starting point for manually or programmatically composing a file request.

PVFILEREQUEST

The PVFILEREQUEST element contains information required to request a file from PaperVision Enterprise. You can obtain the content to populate the sub-elements of a PVFILEREQUEST by calling the PaperVision Enterprise HTTPInterface.aspx RetrieveObjects or RetrieveDocuments XML web service. For more information on how to use HTTPInterface.aspx, refer to the *PaperVision Enterprise API Reference Guide*. The following table lists the values to obtain from HTTPInterface.aspx and to populate the PVFILEREQUEST.

Note: There is no need to attempt to decrypt values obtained from HTTPInterface.aspx RetrieveObjects or RetrieveDocuments. You can copy the encrypted values directly to PVFILEREQUEST.

Note: Even though a valid Session ID is a required parameter, requesting a PaperVision Enterprise file does not automatically extend the life of a session.

| PVFILEREQUEST | | | |
|------------------|--------|-----------|---|
| Element | Type | Required? | Content Source* |
| SESSIONID | String | Yes | Session ID obtained from HTTPInterface.aspx LoginUserEx3 and LoadSystemAccess calls |
| EXPIRES | String | Yes | DOCOBJS/DOC/LOCATION[@EXPIRES] |
| FILEPATH | String | No | DOCOBJS/DOC/LOCATION/FILE/FILEPATH |
| DGENCRYPTKEY | String | No | DOCOBJS/DOC/LOCATION[@DGENC] |
| DGPATH | String | No | DOCOBJS/DOC/LOCATION[@DGPATH] |
| OFFSETSTART | String | No | DOCOBJS/DOC/LOCATION/FILE/OFFSET_START |
| OFFSETEND | String | No | DOCOBJS/DOC/LOCATION/FILE/OFFSET_END |
| OFFSETDECOMPRESS | Bool | No | DOCOBJS/DOC/LOCATION[@COMPRESSED] |

*Copy/use the value of listed XML element or attribute as returned from RetrieveObjects or RetrieveDocuments.

Sample File Service Request XML Envelope

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns="http://www.digitechsystems.com">
  <soapenv:Header/>
  <soapenv:Body>
    <GETFILE>
      <PVFILEREQUEST>

        <SESSIONID>1E2A30D734AD88CEA09E38F02681432B1532EEE12A4536228B9476C68717CDD0C5E51B
        B575843D0EE9D8732213F6F9EF7178A8CFE91D118693EC72B6A1A7072</SESSIONID>

      </PVFILEREQUEST>
    </GETFILE>
  </soapenv:Body>
</soapenv:Envelope>
```

```

<EXPIRES>2016-05-17 16:30:42</EXPIRES>

<FILEPATH>0E79C22B8EB7038EAF2F629E1C57D0E5D58F5B18D24B0B79FBCE27A765C36FC3C67D198F
D5EC445FC564F9AEEEBCA17C8005612D7F8806B049AAE5726CB32DEC5E07C11AD3DB1B0B489D7EBDE3
592E4EFC9EA90C80D0E87947FE9F42D7F1E2EB76BAAE197EC1F8C3869066D1E199BF93</FILEPATH>

<DGENCRYPTKEY></DGENCRYPTKEY>

<DGPATH>779E379C337DCDB5EEF5A8AB584E4C579944650C3566CCD758A65C4EFC51C57A0A85377FEE
75FDBBA736D2F97BC07CCAC0CDF9DB4229370A1769CC0FE3FDBFE0</DGPATH>

<OFFSETSTART>0D5F95DC52B63F131A919DA6B547C47F5B575B4ED70FD681EBA9A3AA1F4FD197
</OFFSETSTART>

<OFFSETEND>6B9C105C045E85554D63482F22A7B59002389623627BD43070E654EF23C1681E
</OFFSETEND>

<OFFSETDECOMPRESS>False</OFFSETDECOMPRESS>

</PVFILEREQUEST>
</GETFILE>
</soapenv:Body>
</soapenv:Envelope>

```

Receiving File Service Results

A call to the file service will return either a SOAP fault or streamed content (application/octet-stream) that represents the requested file. Before the returned message is completely delivered, you can begin to process the content (i.e., you are not required to buffer/wait for the entire response message).