

# FotoWare Archive Agent Specification

Copyright (C) FotoWare a.s 1997-2007. All rights reserved.

*Document summary: This document contains the specification of the FotoWare Archive Agent and can be used as a reference when writing clients to an archive agent or implementing an archive agent server for third-party systems.*

## Table of contents

1	Introduction .....	3
2	Data formats .....	4
3	Access URLs .....	5
4	Interface .....	6
4.1	'Information' method .....	6
4.1.1	URL .....	6
4.1.2	Parameters .....	6
4.1.3	Return value .....	6
4.2	'GetSmallLogo' method .....	7
4.2.1	URL .....	7
4.2.2	Parameters .....	7
4.2.3	Return value .....	7
4.3	'GetLargeLogo' method .....	8
4.3.1	URL .....	8
4.3.2	Parameters .....	8
4.3.3	Return value .....	8
4.4	'Search' method .....	9
4.4.1	URL .....	9
4.4.2	Parameters .....	9
4.4.3	Return value .....	9
4.5	'FileInfo' method .....	10
4.5.1	URL .....	10
4.5.2	Parameters .....	10
4.5.3	Return value .....	10
4.6	'Download' method .....	11
4.6.1	URL .....	11
4.6.2	Parameters .....	11
4.6.3	Return value .....	11
4.7	Getting previews .....	12
5	Portal Agent Information XML .....	13
5.1	Elements .....	13
5.2	Example .....	13
6	Archive Agent File List XML .....	14
6.1	Introduction .....	14
6.2	Elements .....	14
6.3	Example .....	14
7	Example of XSL Transform .....	15
8	Licensing .....	16

# 1 Introduction

---

FotoWare's Archive Agent interface is an open standard for making document archives accessible from any other application. The interface was first developed for the FotoWare Portal Server project. FotoWare Portal Server is capable of presenting search results from many archives in a single interface by communicating with the various archives through their archive agents.

The archive agent interface also makes it easy for third-party applications to search in archives without writing interfaces to proprietary software. FotoWeb publishes an archive agent for each archive in the system, which makes it very easy for applications that integrate with FotoWeb to find files, get previews and download these files.

The interface is based upon known industry standards, which makes it easy to develop both server and client software for this system. Server and client software can be developed in any development language on any platform that supports communication via HTTP and reading of XML documents.

The agents communicate with clients via HTTP, and uses standard query string parameters to control the requests. Answers from the agent are delivered as well-formed XML that can be read using any standard XML parser.

FotoWare has two software modules that support the Archive Agent interface:

- FotoWare Archive Agent ISAPI Module
- FotoWeb Enterprise

Currently, the Archive Agent is only commercially available through FotoWeb Enterprise.

## 2 Data formats

---

The Archive Agents communicate via HTTP URLs and return results as XML documents. The previews are returned in binary JPEG format.

URLs are expected to be in the UTF-8 character format. Reserved characters should be encoded using the Quoted Printable standard. (More information can be found in the HTTP specification.)

The XML documents returned from the Archive Agent should be encoded using UTF-8. Agents may return data in UTF-16 format, in which case that should be specified in the Content-Type header:

```
Content-Type: text/xml; charset=UTF-16
```

Using UTF-16 encoding may not be ideal due to the large overhead of transmitting the data.

It is recommended that Archive Agent implementations support GZIP transfer encoding of the data. This will dramatically reduce the download time of the XML documents. To enable GZIP transfer compression, use Accept-Encoding header in the request to the server:

```
Accept-Encoding: gzip
```

The server signals that the data is in fact compressed using the Content-Encoding response header:

```
Content-Encoding: gzip
```

---

## 3 Access URLs

---

To access Archive Agent, you need to send an HTTP request following a specific format. The format of the URLs has been standardized, so that you can write client software without knowing anything about the architecture of the archive agent server. The URLs have been designed so that it is possible to develop agents in any web application technology.

A typical URL for an archive agent looks like this:

```
http://servername/ArchiveAgent/ArchiveAgent.dll/<Command>
```

The <Command> part is replaced by the appropriate command the client wants to execute. Additional parameters are passed on the query line.

A URL for an archive agent implementation in ASP.NET can look like this:

```
http://servername/ArchiveAgent.aspx/<Command>
```

To access the archive agent interface for an archive in FotoWeb, use this URL:

```
http://servername/fotoweb/fwbin/fotoweb_isapi.dll/ArchiveAgent/5000/<Command>
```

In the URL above, we access the agent for the archive with id 5000. To access another archive, replace this part with the name or id of another archive.

## 4 Interface

---

### 4.1 'Information' method

This method returns general information about the company running the Archive Agent.

#### 4.1.1 URL

```
http://servername/ArchiveAgentPath/Information
```

#### 4.1.2 Parameters

None

#### 4.1.3 Return value

A 'Portal Agent Information' XML document:

```
<?xml version="1.0" ?>
<PortalAgentInformation>
  <Company>FotoWare a.s.</Company>
  <Address>Lille Grenden 5, N-0159 Oslo, Norway</Address>
  <SalesEmail>sales@fotoware.com</SalesEmail>
  <SupportEmail>support@fotoware.com</SupportEmail>
  <Phone>+47 22 00 30 30</Phone>
  <Fax>+47 22 00 30 20</Fax>
  <Url>http://www.fotoware.com</Url>
  <BriefDescription>FotoWare a.s is the leading...</BriefDescription>
  <Description>FotoWare a.s is the leading company...</Description>
</PortalAgentInformation>
```

## 4.2 'GetSmallLogo' method

This method returns the a small logo. This logo can be used when presenting the archive to users.

### 4.2.1 URL

```
http://servername/ArchiveAgentPath/GetSmallLogo
```

### 4.2.2 Parameters

None

### 4.2.3 Return value

An image file in JPEG, GIF or PNG format representing the logo of the company running the agent.



## 4.3 'GetLargeLogo' method

This method returns the large company logo. This logo can be used when presenting the archive to users.

### 4.3.1 URL

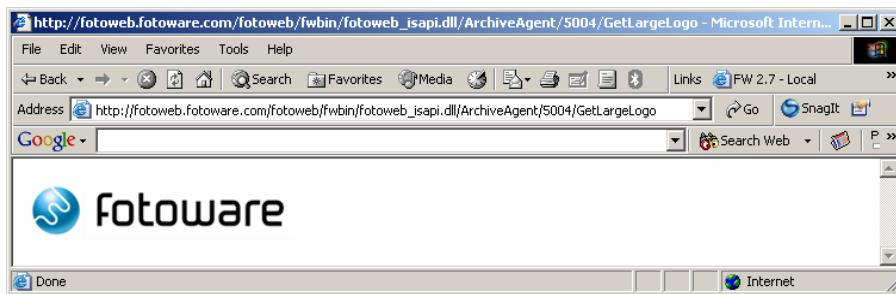
```
http://servername/ArchiveAgentPath/GetLargeLogo
```

### 4.3.2 Parameters

None

### 4.3.3 Return value

An image file in JPEG, GIF or PNG format representing the logo of the company running the agent.



## 4.4 'Search' method

This method executes a search through the agent and returns the resulting file list as an XML document.

### 4.4.1 URL

http://servername/ArchiveAgentPath/Search

### 4.4.2 Parameters

#### Search[1]

The string to search for.

#### PreviewSize[\*] := [ 0 . . . 1024 ]

The maximum dimension in pixels for a preview. The URL for this preview will be returned.

#### FileInfo[1] := [ 0 | 1 ]

Boolean value indicating if file information should be returned with the file list.

#### MetaData[1] := [ 0 | 1 ]

Boolean value indicating if meta data should be returned with the file list.

### 4.4.3 Return value

An ['Archive Agent File List XML'](#) document containing a list of files matching the search string and any requested data:

```
<?xml version="1.0" ?>
<FileList
  Version="1.0"
  CreatorApplication="FotoWeb/6.0"
  Created="Mon, 06 Jun 2005 11:50:38 GMT"
  TotalHits="2"
  ReturnedHits="2"
  SearchTime="0.582941850472644"
  ProcessingTime="4.4580299341163">

  <File Name="182492.jpg" Id="fileIdString">
    <PreviewLinks>
      <PreviewUrl Id="0" Size="200">previewUrl</PreviewUrl>
    </PreviewLinks>

    <FileInfo>
      <Path>D:\Media Library\Archive\Images\Alain Ernoult</Path>
      <Created>Tue, 30 Sep 2003 09:23:38 GMT</Created>
      <LastModified>Wed, 16 Mar 2005 12:59:15 GMT</LastModified>
      <FileSize>4203447</FileSize>
      <MimeType>image/jpeg</MimeType>
    </FileInfo>

    <MetaData>
      <PixelWidth>3565</PixelWidth>
      <PixelHeight>2371</PixelHeight>
      <Resolution>300.00</Resolution>
      <ColorSpace>Rgb</ColorSpace>

      <Text>
        <Field Id="IPTC2:0" Name="Iptc Field 0" />
        <Field Id="IPTC2:120" Name="Caption">AUTO VOITURE</Field>
      </Text>
    </MetaData>
  </File>
</FileList>
```

## 4.5 'FileInfo' method

This method returns information about a file such as file attributes and/or meta data etc. This function requires input delivered by the 'Search' method.

### 4.5.1 URL

```
http://servername/ArchiveAgentPath/FileInfo
```

### 4.5.2 Parameters

**Id[ 1 . . . \* ]**

Valid Id(s) of the file(s) to return information for. These ids must be those returned by the 'Search' method.

**PreviewSize[\*] := [ 0 . . . 1024 ]**

The maximum dimension in pixels for a preview. The URL for this preview will be returned.

**FileInfo[1] := [ 0 | 1 ]**

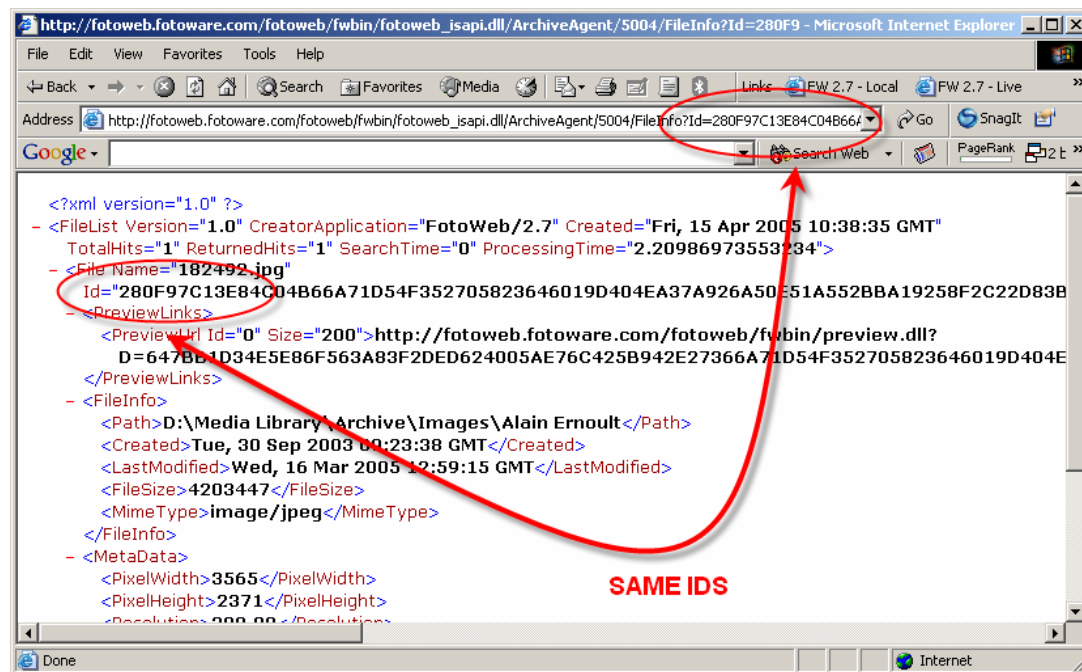
Boolean value indicating if file information should be returned with the file list.

**MetaData[1] := [ 0 | 1 ]**

Boolean value indicating if meta data should be returned with the file list.

### 4.5.3 Return value

An 'Archive Agent File List XML' document containing information for the requested files. If a file is not found, it is left out of the document, unless the request is for a single file, in which case a '404 Not Found' message will be returned.



Screenshot 1 Result of a FileInfo request to an archive agent

## 4.6 'Download' method

This method downloads the specified file. You must have download permissions in the archive to call this method.

### 4.6.1 URL

```
http://servername/ArchiveAgentPath/Download
```

### 4.6.2 Parameters

#### Id[ 1 ]

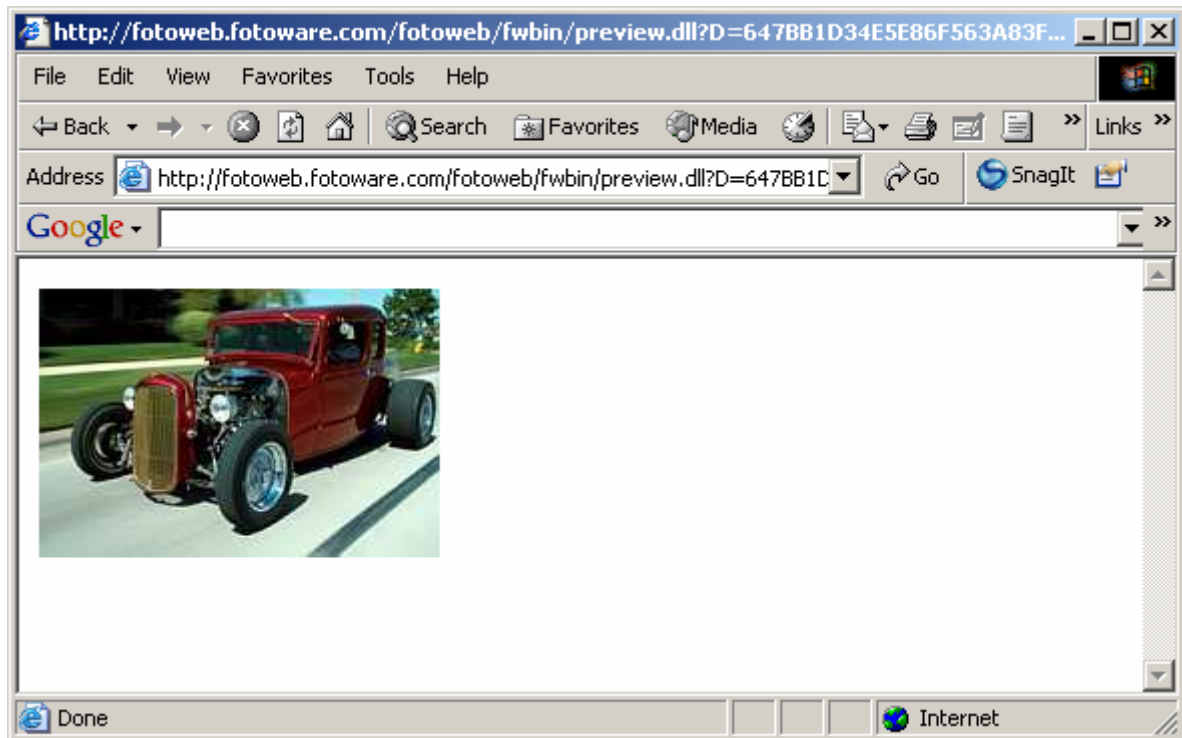
A valid Id of the file to download. This id must be that returned by the 'Search' method.

### 4.6.3 Return value

The data of the specified file in binary format if the calling user has download permissions in the archive.

## 4.7 Getting previews

To get the JPEG RGB previews of the files reported by the Archive Agent, you can use the URL provided by the 'PreviewURL Id' node if the XML returned by the 'Search' and 'FileInfo' methods.



Screenshot 2 Pasting a preview URL from an Archive Agent into Internet Explorer

---

## 5 Portal Agent Information XML

---

The Archive Agent Information XML document contains information about the company or entity running the Archive Agent and any contact information they choose to disclose.

### 5.1 Elements

All the elements of the Xml document are required. The BriefDescription and Description elements may contain pure text or valid XHTML.

### 5.2 Example

```
<PortalAgentInformation>
  <Company />
  <Address />
  <SalesEmail />
  <SupportEmail />
  <Phone />
  <Fax />
  <Url />
  <BriefDescription />
  <Description />
</PortalAgentInformation>
```

## 6 Archive Agent File List XML

### 6.1 Introduction

The Archive Agent File List XML document contains information about the files matching a request sent to the Archive Agent.

### 6.2 Elements

All elements are required, except under these circumstances:

- When `MetaData=0` is specified, the `MetaData` element may be missing.
- When `FileInformation=0` is specified, the `FileInfo` element may be missing.
- If no `PreviewSize` parameters are specified, the `PreviewLinks` element must be missing.

The number of `PreviewUrl` elements returned must match the number of `PreviewSize` parameters passed on the query string. If one of the requested preview sizes is out of range, the server *must* return a 400 Bad Request message. It *must not* ignore the parameter and return an incomplete XML document.

### 6.3 Example

```
<?xml version="1.0" ?>
<FileList Version="1.0"
  CreatorApplication="FotoWare Archive Agent"
  Created="Tue, 07 Jan 2003 15:01:56 GMT"
  TotalHits="258"
  ReturnedHits="50"
  SearchTimeMs="322.2212"
  ProcessingTimeMs="2826.432">
<File Name="gjestp_bw.jpg" Id="fileId_1">
  <FileInfo>
    <Path>C:\Database\2003\01\12<Path>
    <Created>Tue, 07 Jan 2003 15:01:56 GMT</Created>
    <LastModified>Tue, 07 Jan 2003 15:01:56 GMT</LastModified>
    <FileSize>355422</FileSize>
    <MimeType>image/jpeg</MimeType>
  </FileInfo>
  <PreviewLinks>
    <PreviewUrl Id="0"
Size="120">http://.../ArchiveAgent.dll/Preview?Id=previewId_1</PreviewUrl>
    <PreviewUrl Id="1"
Size="180">http://.../ArchiveAgent.dll/Preview?Id=previewId_2</PreviewUrl>
  </PreviewLinks>
  <MetaData>
    <PixelWidth>1312</PixelWidth>
    <PixelHeight>2000</PixelHeight>
    <Resolution>300.00</Resolution>
    <ColorSpace>Rgb</ColorSpace>
    <Text>
      <Field Id="IPTC2:30" Name="Release Date">20020303</Field>
      <Field Id="IPTC2:55" Name="Created Date">20020302</Field>
      <Field Id="IPTC2:80" Name="Byline">Boesen Michael</Field>
      <Field Id="IPTC2:110" Name="Credit">All Over Press</Field>
      <Field Id="IPTC2:115" Name="Source">Polfoto</Field>
      <Field Id="IPTC2:120" Name="Caption">POLFOTO, COPENHAGEN, DENMARK
3MAR2002: Australian singer Kylie Minogue</Field>
      <Field Id="IPTC2:122" Name="Caption Writer">Trine Hansen</Field>
      <Field Id="IPTC2:200" Name="Custom Field 1">69d0xk8o</Field>
      <Field Id="IPTC2:201" Name="Custom Field 2">211475</Field>
    </Text>
  </MetaData>
</File>
</FileList>
```

## 7 Example of XSL Transform

The returned XML output from the Archive Agent can easily be converted to HTML that can be displayed by any web browser using a technique called 'XSL Transforms'. A sample XSL Transform document for showing thumbnails from the result of an Archive Agent is shown below:

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:template xmlns:xsl="http://www.w3.org/TR/WD-xsl">
  <HTML>
  <HEAD>
  <TITLE>FotoWare Archive Agent Thumbnail Result</TITLE>
  </HEAD>
  <BODY bgcolor="#ffffff">
    <H2>Hits: <xsl:value-of select="FileList/@ReturnedHits" /> (<xsl:value-of
  select="FileList/@TotalHits" />), Search time: <xsl:value-of
  select="FileList/@SearchTimeMs" /> ms, Processing time: <xsl:value-of
  select="FileList/@ProcessingTimeMs" /> ms</H2>
    <xsl:for-each select="FileList/File">
      <xsl:for-each select="PreviewLinks/PreviewUrl">
        <IMG>
          <xsl:attribute name="src">
            <xsl:value-of select="." />
          </xsl:attribute>

          <xsl:attribute name="title">
            <xsl:value-of select="../../MetaData/Text/Field[@Id='IPTC2:120']" />
          </xsl:attribute>
        </IMG>
      </xsl:for-each>
    </xsl:for-each>
  </BODY>
</HTML>
</xsl:template>
```

## 8 Licensing

---

FotoWare has two applications that publish the Archive Agent interface:

- FotoWeb Enterprise
- Archive Agent (stand-alone application)

FotoWare's Archive Agent is licensed through Index Manager. Each user that directly or indirectly (e.g. through a third party system) accesses the Index Manager through the Archive Agent needs a Client Access License (CAL) in Index Manager. If you build a web portal that shows images using the Archive Agent, each user of the web system must have a CAL in Index Manager. If your system has 10 different users, you need at least 10 CALs. If your system has 500 users, you need at least 500 CALs. (The Archive Agent is *not* licensed per simultaneous user.) The Archive Agent software itself is licensed per server, and requires the customer to have a valid Software Support Agreement with FotoWare.

If you have a license of FotoWeb Enterprise, the requirements for CALs in Index Manager do not apply. You can then access Index Manager through the Archive Agent of FotoWeb, and the FotoWeb license will be valid for all users. If your web system has 500 users, you still need only one FotoWeb Enterprise license to cover all users. (The FotoWeb license includes the CALs for Index Manager for the FotoWeb users.)