

# EWRetrieveAttachedAsSOAPAttachment

---

Retrieves an attached file from the specified field of the table record in your knowledgebase.

## Syntax

```
ew.EWRetrieveAttachedAsSOAPAttachment(String sessionId, String  
    tableName, long id, String fieldName, int position);
```

## Usage

Use the EWRetrieveAttachedAsSOAPAttachment call to retrieve an attached file from a File or Image field in the table record.

## Rules and Guidelines

When retrieving attached files, consider the following rules and guidelines:

- The username that was used to obtain the specified session token must have sufficient access rights to read individual records within the specified table. Please verify specific permissions via **Setup > Access > Manage Groups > (Edit Group) > Table > (Select Table) > Permissions**.
- Agiloft allows specifying fine-grained access permissions on the field level. The username that was used to obtain the specified session token must have sufficient access rights to be able to read the field content. Please verify specific permissions via **Setup > Access > Manage Groups > (Edit Group) > Table > (Edit Table) > Field Permissions**.
- This call requires the position of the attached file in the specified field. Position numbering starts from zero.
- One can use the EWRead method to obtain an array of file names for the attached fields in the specified field. The sequence of file names in the array will correspond to the positions of the files.
- Only one file per call is retrieved.

- In general, you use `EWRetrieveAttachedAsSOAPAttachment` when you know in advance the identifiers of the records to retrieve. The client application may use the likes of the `EWSelectFromTable` call to obtain record identifiers beforehand or take the identifiers from the `id` field of the data structures and the linking classes.

## Basic Steps for Retrieving Attached Files

Retrieving files from records involves the following basic steps:

1. Determine the id of the record you want to update. You may want to use the `EWSelectFromTable` or `EWSearchTable` calls to get the identifiers of the records based on some search condition; or get the identifier from a previous `EWCreate` call; or get the id of a linked record from the linking class after performing `EWRead` or `EWUpdate`.
2. Determine the position of the file in the field.
3. Call `EWRetrieveAttachedAsSOAPAttachment`.
4. Process the results.

## Example Task

In MyKB knowledgebase as user A, retrieve the first file named B.txt currently attached in the field Additional Files of case #456 in the current directory.

The task is completed by performing the following steps:

1. Login to MyKB with "A" and "password" and English as the local language.
2. Read the record to get the list of the files.
3. Find the position of the file.
4. Invoke the retrieve file call.
5. Logout.

## Sample Code - Java

You can generate sample Web Services code for any table by selecting **Setup > Tables > (*Edit Table*) > API > Download Sample**.

---

```

public int retrieve() throws Exception {
    EWServiceAPI binding = new EWServiceAPIServiceLocator().getDemo();
    try {
        String sessionId = binding.EWLogin("MyKB", "A", "password", "en");
        WSCase wsCase = (WSCase) binding.EWRead(sessionId, "case", 456);
        String[] fileNames = wsCase.getAdditional_Files();
        int n = 0;
        if (fileNames!=null) {
            for (int i = 0; i < files.length; i++) {
                if ("B.txt".equals(fileNames[i])) {
                    binding.EWRetrieveAttachedAsSOAPAttachment("case", 456,
                        "additional_files", i);
                    Object[] attachments = binding.getAttachments();
                    AttachmentPart attachmentPart = (AttachmentPart) attachments[0];
                    final InputStream inputStream =
                        attachmentPart.getDataHandler().getInputStream();
                    BufferedOutputStream bos =
                        new BufferedOutputStream(new FileOutputStream("B.txt"));
                    int b;
                    while ((b = inputStream.read()) != -1) bos.write(b);
                    bos.flush();
                    bos.close();
                    break;
                }
            }
        }
        return;
    } finally {
        binding.EWLogout(sessionId);
    }
}

```

## Arguments

Name	Type	Description
sessionId	String	Session token.
tableName	String	The name of the table where the record is.
id	long	The identifier of the record to attach the files to.
fieldName	String	The name of the field to attach the files to.
position	int	The position of the file to be retrieved.

# Response

The file as a SOAP attachment.

## Faults

**EWSessionException** - client not logged in or the session has expired; client should re-login.

**EWPermissionException** - user used to create the session lacks the sufficient privileges to read the record.

**EWWrongDataException** - client has supplied the wrong data, for instance ID cannot be found.

**EWOperationException** - the operation has been blocked by an Agiloft function, for example a table-level lock.

**EWIntegrityException** - the specified table cannot be found or its primary key cannot be identified.

**EWUnexpectedException** - an unexpected exception has happened; user should report this for investigation.