# File Upload Example

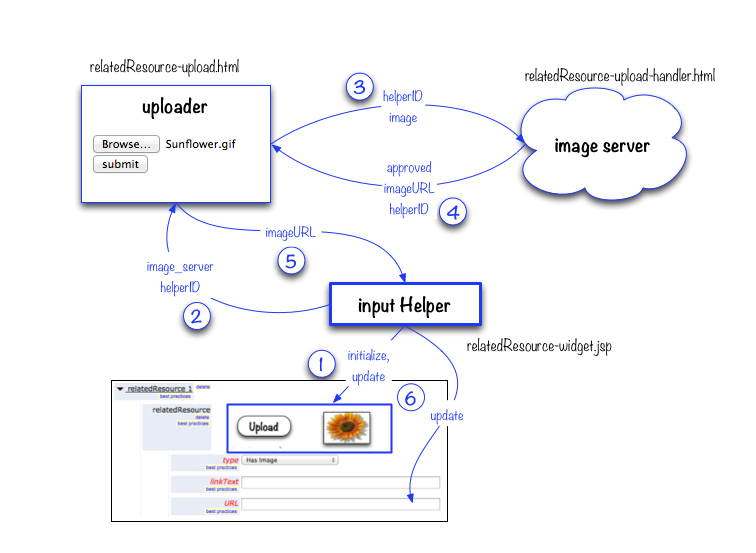
This document details how a “File Upload” Input Helper might be implemented. For Input Helper basics see *Input Helpers in the NCS.*

The files required for this example are contained in the *file\_upload\_example* directory. These files should be copied into the input helper directory for the framework to which this helper is attached (e.g. dcs-project/web/editor/input\_helpers/lar for the *lar* framework). Then the framework config must be modified (as explained in *Input Helpers in the NCS*) to configure the *File Upload Input Helper.*

## Use Case

Allow the user to upload an image file from their local machine to be associated with a particular metadata record. The challenge here is that the NCS does not store files locally, so this must be done by a separate server.

## Flow of control



1. When the input field is rendered, the Input Helper is initialized. The user is presented an *Upload* button that brings up a form for uploading an image file. If there is a value in the URL field, the image at that URL is displayed as a thumbnail.
2. When the user clicks on the *Upload* button, the upload form (*uploader*) is displayed.
3. When an input file is specified, the uploaded image file is sent to an external server (*image server*), which validates the input and stores the image file.
4. *The image server* then returns the *imageURL* (containing the uploaded file’s URL) to the upload form (*uploader*).
5. The upload form returns *imageURL* to *the input helper*.
6. The input helper inserts the *imageURL* into the proper field of the metadata editor and displays the thumbnail in the editor.

## Notes

The File Upload example includes a *simulated* image server (see *relatedResource-upload-handler.html*), to illustrate the flow of control. For actual use, the image server must be implemented.

The included simulated image server code includes a place to specify a *baseImageUrl* path. This variable can be used to construct a imageURL that points to an actual image on the local file system (again, to simulate the path that a real image server might construct).

While testing with the simulated external server code, you will not see thumbnail images unless the imageUrl returned by the simulated image server corresponds to an actual image.