

# Update CAST Extend local server with new content

- [Introduction](#)
- [Update process](#)
  - [Creating the bundle \(.extarchive file\)](#)
  - [Upload the bundle to CAST Extend local server](#)
    - [Via AIP Console](#)
    - [Via the CAST Extend local server GUI](#)
    - [Via the CAST Extend local server built in API and curl](#)



**Summary:** step-by-step instructions for manually updating **CAST Extend local server with new content**.

## Introduction

CAST Extend local server is delivered "empty" without any preconfigured extensions and in normal functioning mode, extensions will be automatically downloaded from CAST Extend when requested by AIP Console or Extension Downloader. However, you can also populate CAST Extend local server with specific extensions manually should you need to.

The **process of updating** is as follows:

- Create a new **bundle (an .extarchive file)** containing the extensions you require using [ExtendCli](#)
- Upload the bundle to update CAST Extend local server:
  - using [AIP Console 1.12.0](#)
  - using the [CAST Extend API](#) and [curl](#)

When CAST Extend local server is updated with a new bundle of extensions and extensions have already been added, a "**merge**" is actioned, rather than a "delete and add". This means that no extensions are ever removed, i.e.:

- Any extensions that already exist in CAST Extend local server but do not exist in the new bundle will remain as they are in CAST Extend local server
- Any extensions that already exist in CAST Extend local server at a given version number will remain as they are in CAST Extend local server when:
  - an older version of the same extension is added to the bundle (both versions will be available for use)
  - a newer version of the same extension is added to the bundle (both versions will be available for use)
  - an older and a newer version of the same extension are added to the bundle (all three versions will be available for use)
  - the same version of the extension is added to the bundle

When an **.extarchive** is uploaded, the packages are stored by default in:

```
%PROGRAMDATA%\CAST\Extend\packages
```



The ability to create a bundle using the live CAST Extend website is no longer supported. You must use [ExtendCli](#).

## Update process


### Creating the bundle (.extarchive file)

See [ExtendCli](#) for more detailed information. If you want to simply add all "**product**", "**user community**" and "**labs**" extensions to a bundle (.extarchive file), you can use the following commands:

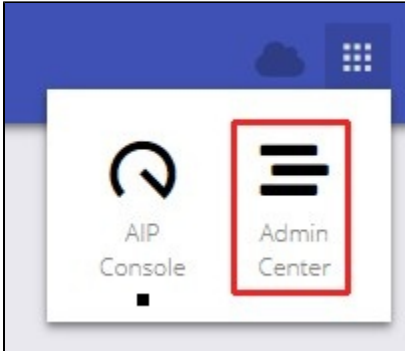
```
ExtendCli.exe config set api-key <YOUR API KEY>
ExtendCli.exe bundle clear
ExtendCli.exe bundle template add product
ExtendCli.exe bundle template add community
ExtendCli.exe bundle template add labs
Extendcli.exe bundle pack -o <Folder path>
```

### Upload the bundle to CAST Extend local server

## Via AIP Console

 This method has been removed in **Console 2.5**.

If you are using CAST Extend local server with AIP Console, the process of updating CAST Extend local server is handled automatically by AIP Console. Login to AIP Console with a user account that already has the **Admin role** and switch to the **Admin Center**:



Now move to the **Administration Center - System Settings** panel and expand the **CAST Extend** option:

*Click to enlarge*

The screenshot displays the 'SYSTEM SETTINGS' interface for CAST Extend. At the top, navigation tabs include 'APPLICATIONS', 'SECURITY', 'NODES', 'GLOBAL CONFIGURATIONS', and 'SYSTEM SETTINGS' (highlighted with a red box). Below these are expandable sections for 'Dashboard Integration' and 'Imaging Settings'. The 'CAST Extend' section is expanded and highlighted with a red box. It contains 'Server Settings' with three radio button options: 'Extend official service', 'Extend offline service', and 'Extend proxy' (selected). A 'SYNCHRONIZE EXTENSIONS' button is present. Below are input fields for 'CAST Extend URL\*' (containing 'http://win10test:8085') and 'API key\*' (masked with dots). A checked checkbox 'Allow CAST to automatically collect anonymous statistical data' and a 'SAVE' button are also visible. The 'Extension Package Upload' section below features a large red-bordered box with the text 'Drop your extensions package file here' and a button 'or click to choose a file'. A red arrow points from the 'CAST Extend' header to this upload area.

Drag and drop your downloaded bundle in to the options screen, or click the button to choose the **.extarchive file** directly. The bundle will then be uploaded and added to CAST Extend local server automatically and all new extensions added to the bundle will be available immediately.

### Via the CAST Extend local server GUI

Starting v. 1.1.0, CAST Extend local server has its own GUI which you can use to upload the **.extarchive** file. See [Manage CAST Extend local server via Administration Center](#) for more information about this.

GENERAL  UPDATES	Date ↑	Type	Status	UPLOAD	SYNCHRONIZE
	4 Jun 2021 - 14:59		complete		
	4 Jun 2021 - 15:29		complete		
	4 Jun 2021 - 15:48		complete		

## Via the CAST Extend local server built in API and curl

CAST Extend local server has a built in [API](#) that can be accessed in a manner of ways. It is therefore possible to upload a new bundle using the API, for example with [curl](#):

```
curl -H "x-cxproxy-apikey:<APIKEY>" -F "data=@path_to_file.extarchive" http://<server>:<port>/api/update
```

<b>-H</b>	This option defines an extra header to include in the request when sending HTTP to a server. In the case of CAST Extend local server, this is used to send the APIKEY value to allow access. See <a href="https://curl.se/docs/manpage.html#-H">https://curl.se/docs/manpage.html#-H</a> .
<b>APIKEY</b>	The APIKEY that grants access to Extend local server. The APIKEY value is displayed at the end of the CAST Extend local server installation process. Alternatively you can find the APIKEY value in the following location on the server on which CAST Extend local server is installed:  <pre>%PROGRAMFILES%\CAST\Extend\config.proxy.json</pre> <p>The key is located on the line <b>APIKEY</b>:</p> <pre>{   "PORT": 8085,   "STORAGE_DIR": "c:/ProgramData/CAST/Extend",   "PACKAGE_DIR": null,   "LOG_DIR": "C:/Program Files/CAST/Extend/logs",   "PUBLIC_URL": "http://WIN10TEST:8085/",   "EXTEND_URL": "https://extend.castsoftware.com",   "APIKEY": "&lt;api_key&gt;",   "ADMI":     "admi194c45b7d34a12adb81df7838b66e127e280838b2585276604865460316d6813353000e1b800c50e3d2cd6b2740b     f47a99568c70ca52424c77f7d5a2b8016b810",   "SYNC_TYPE": "auto" }</pre>
<b>-F</b>	This causes curl to POST data using the Content-Type multipart/form-data. See <a href="https://curl.haxx.se/docs/manpage.html#-F">https://curl.haxx.se/docs/manpage.html#-F</a>
<b>path_to_file.extarchive</b>	The <b>path</b> to the .extarchive bundle file. Replace this with your
<b>&lt;server&gt;</b>	Forms part of the URL to your on premises CAST Extend local server. For example:
<b>&lt;port&gt;</b>	<ul style="list-style-type: none"> <li><b>&lt;server&gt;</b>: the local server on which CAST Extend local server is <a href="#">installed</a>.</li> <li><b>&lt;port&gt;</b>: the port number which CAST Extend local server runs on. This is determined during the <a href="#">installation</a> (default port number is set to 8085).</li> </ul>

For example:

```
curl -H "x-cxproxy-apikey:<api_key>" -F "data=@D:\downloads\b2b430df976c412ca29e799fff0fd86e.extarchive"  
http://WIN10TEST:8085/api/update
```