IIIF Image Conversion Guide

1 Goal

IIIF stands for International Image Interoperability Framework (spoken Triple-Eye-F). The framework was created to offer the scientific community a way to productively interact with their digital objects (visual or audio/visual) and use them across different platforms. Here you can learn about the benefits of IIIF.

If you are looking for a general introduction to the IIIF framework, please check out the What is IIIF? training resources for aggregators.

The **goal** of this guide is to give you an example of how to **make your digital images IIIF compliant**. For that purpose you will have to run a **web server** and a **IIIF-compatible image server** and provide your **images in a multi-resolution format (pyramid TIFFs)**. Pyramid TIFFs are layered documents that contain multiple, mapped versions of the same image in different resolutions. This allows image servers to optimize zooming, as they switch to higher resolution images as the user zooms deeper and deeper into the document.

Main contents of the guide

- How to run a **web server** (chapter 3)
- How to run a **IIIF-compatible image server** (chapter 4)
- How to **convert images to a multi-resolution format** (chapter 5)
- How to **access your IIIF images** through an Internet browser URL (Chapter 6), and how to access images via a IIIF manifest (chapter 7).

Glossary and scripts

- You will find links and information on the most important technical terms in the **glossary** of chapter 8.
- And finally, in chapter 9, you will find the information from where to download the **image conversion script** and a **script to install the web server and the IIPImage Server**.

Overall workflow

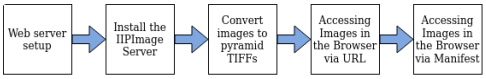

Figure 1 - Workflow

The first two steps “**Web server setup**” and “**Install the IIPImage Server**” are described in detail in chapter 3 and 4 of this guide, but they can be easily executed automatically by running the script “**iiif-install.sh**” on your computer (find the script in chapter 9).

The step “**Convert images to pyramid TIFFs**”, described in chapter 5, can be carried out by running the “**iiif-image-converter.sh**” script (find the script in chapter 9).

2 Prior knowledge and technology

This guide is intended for people with basic **prior knowledge** of the Linux operating system (e.g. opening the terminal, basic Linux commands (12 essential Linux commands for beginners), difference between normal and sudo user, etc.). In addition, it would be good if you would familiarize yourself with the following points: IIIF and Image API (How It Works); multi-resolution image file formats (IIPImage » Images).

In-depth technical knowledge is not required, i.e. it is possible to install and set up the Apache and Image Server, perform image format conversion and access the images via a URL by simply following the steps in this guide.

Technology used within the scope of this guide:

- **Operating system**: Linux Debian, i.e. Ubuntu
  - sudo user with administration permissions
- **Webserver**: Apache2
- **Image server**: IIPImage Server

3 Web server setup

Web servers are used to serve web pages requested by client computers. Apache is a widely used web server application. It is very secure, fast and reliable. It makes it possible to share your content (the offline web page) with other users in your network on a real website.

Update packages on your computer

Enter the following command in the Linux terminal to install and set up Apache server:

```
sudo apt-get update
```

After entering this command, the command line will prompt for user and password.

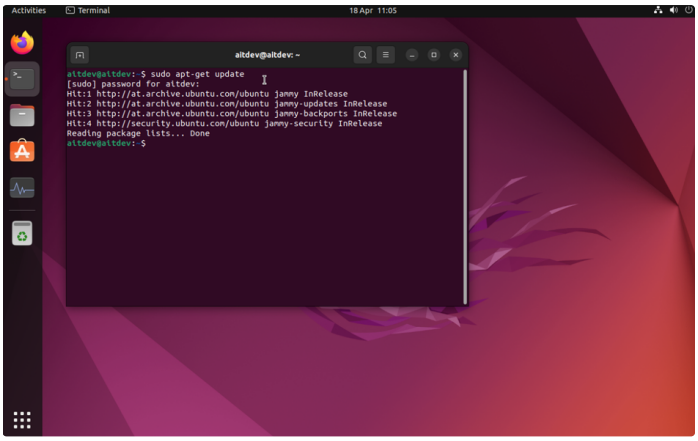

Figure 2 - Output of the command "sudo apt-get update"

Install Apache 2 (additionally install sub-software package fcgid)

```
sudo apt-get install apache2 libapache2-mod-fcgid
```

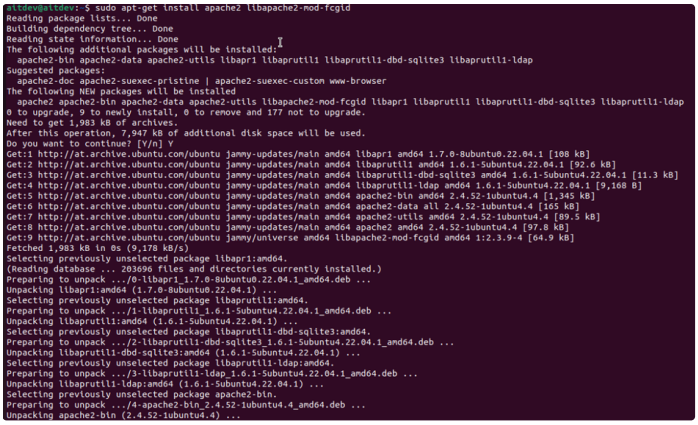

Figure 3 - Output of the command "sudo apt-get install apache2 lib-apache2-mod-fcgid"

Start web server

```
sudo systemctl start apache2
```

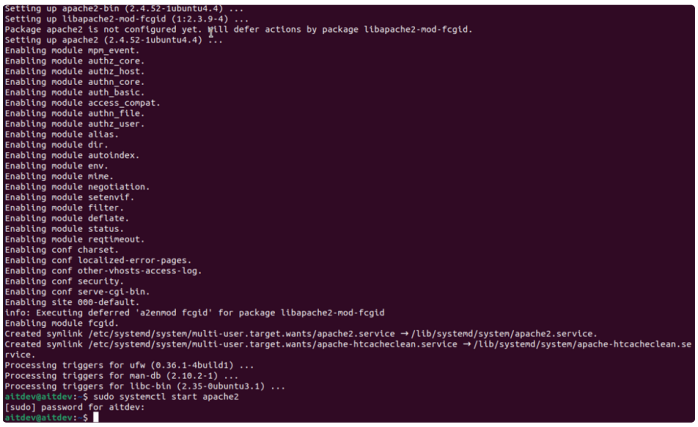

Figure 4 - Output of the command "sudo systemctl start apache2"

Make sure that the Apache server is actually running

Open your browser, enter your IP address into your browser's address bar and the Apache2 Debian Default Page should appear as in the screenshot below.

Figure 5 - Example 'It works!'

Move your content (files you want to access via the browser e.g. html and css files) to the default folder Apache points to (/var/www/html/)

Enter your IP address into your browser and if all goes well, your content will be loaded.

4 Install the IIPImage Server

“The IIPImage Server is a feature-rich high performance image server engineered to be stable, fast and lightweight. It is designed for streaming extremely high resolution images and is capable of handling advanced image features such as 16 and 32 bit per channel depths, floating point data, CIELAB colorimetric images and scientific imagery such as multispectral or hyperspectral images and digital elevation map data.”

IIPImage Server installation and setup is done with the following steps:

Install iipimage-server package with command below

```
sudo apt-get install iipimage-server
```

The command will prompt you for installing some dependencies for this package. To do this, type “y” (yes) and press *Enter*.

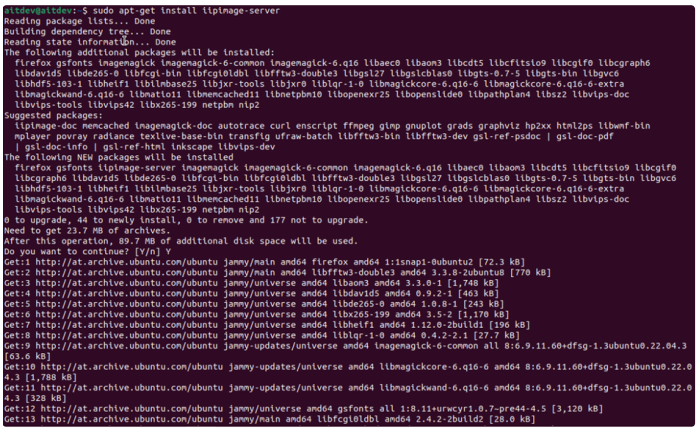

Figure 6 - Output of the command "sudo apt-get install iipimage-server"

Change your image server’s data directory

With the following command the default data directory of your image server */usr/lib/iipimage-server/* is copied to apache2 folder */var/www/*:

```
sudo cp -r /usr/lib/iipimage-server/ /var/www/iipimage-server/
```

Figure 7 - Output of the command "sudo cp -r /usr/lib/iipimage-server/ /var/www/iipimage-server/"

Now you need to run the image server as an Apache module. The modules are configured in directory */etc/Apache2/mods-available/*. Change to this directory and open from there the image server's *iipsrv.conf* config file with following command:

```
sudo nano /etc/apache2/mods-available/iipsrv.conf
```

In this file **change the following line**:

ScriptAlias /iipsrv/ "/usr/lib/iipimage-server/"
to
ScriptAlias /iiif "/var/www/iipimage-server/iipsrv.fcgi"

In addition, you can configure the server to serve through a "cleaner" url. You do that by adding this line in the environment variables:

FcgidInitialEnv URI_MAP "iiif=>IIIF"

Save the file with *Ctrl+o* and press *Enter* to confirm and close nano with *Ctrl+x*.
All these changes are illustrated in Figure 8 (red arrows). With this module enabled, Apache knows where you put the image server's data directory.

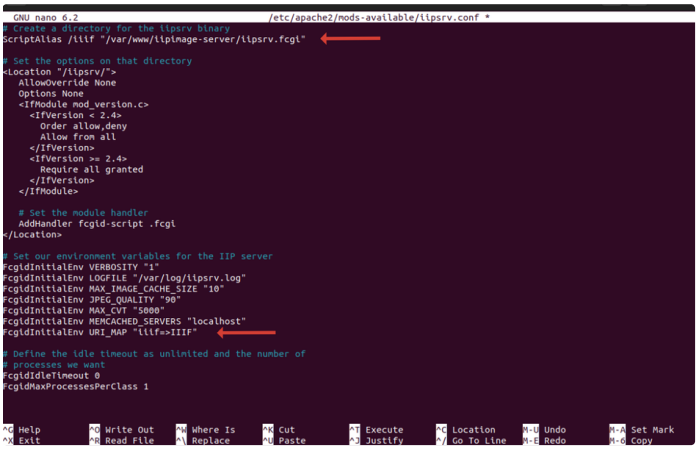

Figure 8 - Editing the "iipsrv.conf" file

Enable the necessary Apache modules for the image server (fcgid already installed and enabled above)

Use the commands:

```
sudo a2enmod headers
```

If fcgid or headers was not enabled before you will have to restart Apache now with the following command:

```
sudo systemctl restart apache2
```

Then you need to check if the image server's module (iipsrv) is enabled:

```
sudo a2enmod iipsrv
```

Now, that all three modules are enabled you need to restart Apache again:

```
sudo systemctl restart apache2
```

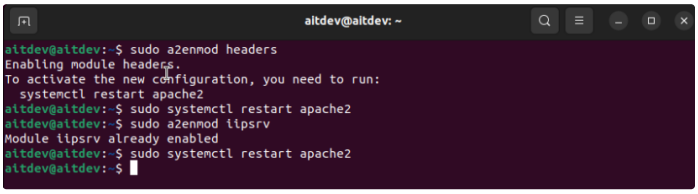

Figure 9 - Output of the commands

Enable CORS

In the image server's config file you enable CORS (cross origin resource sharing) to make sure that the image server is IIIF-compliant, because it allows others to embed your images into their website. To enable CORS open the config file with following command:

```
sudo nano /etc/apache2/apache2.conf
```

Move down to the end of the file and the following line:

Header set Access-Control-Allow-Origin *

It is important that there are no spelling mistakes in the line above. Then you save changes with *Ctrlt+o* and exit nano with *Ctrl+x*.

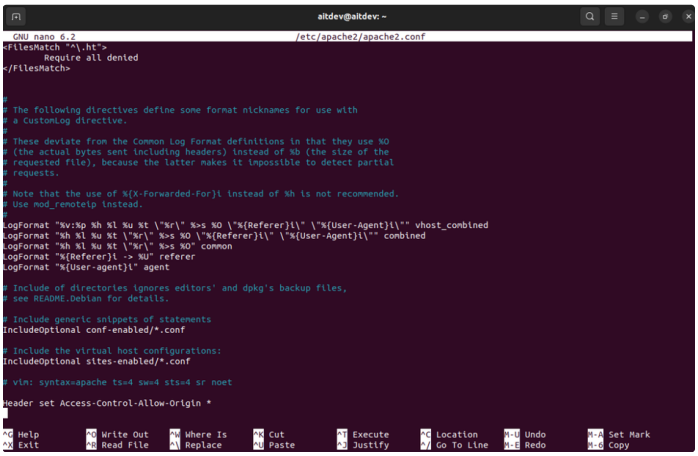

*Figure 10 - Adding the line "Header set Access-Control-Allow-Origin *"*

Now you restart Apache once again:

```
sudo systemctl restart apache2
```

Check if the IIPImage server works

Enter in your browser address bar:

your.ip.address/iiif/

If the start screen of the IIPImage server is shown then you are sure that the server configuration was successful and it runs correctly.

Figure 11 - iipserver working

5 Convert images to pyramid TIFFs

With a multi-resolution format, large raster image files are compressed and can be quickly viewed without having to decompress the entire file. IIPImage Server supports multi-resolution images of the format TIFF and JEPG2000. Thus you need to convert your images to one of these types. In this guide we show how you can convert your images to TIFF format.
Figure 12 shows graphically how pyramid TIFFs are constructed. The Tiled Multi-Resolution (or Tiled Pyramidal) TIFF type allows the image server to enhance zoom options in a way that it switches to higher resolution images of the pyramid as the zoom goes deeper and deeper.

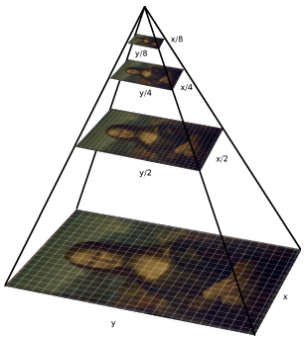

Figure 12 - Graphical representation of the pyramid in tiff format

Installation of the image processor VIPS

Before image transformation the installation of the image processor VIPS is necessary

Use **apt-get** to install the the package **libvips-tools** on Ubuntu:

```
sudo apt-get install libvips-tools
```

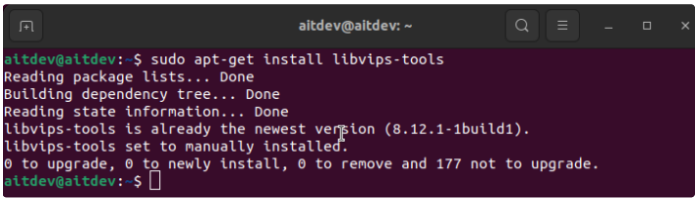

Figure 13 - Install libvips-tools

Now go to the folder where the original images are stored (e.g: **cd /var/www/html/images/**)

Image transformation

This is the main part of this chapter: image transformation.

If you have original images in .png format and you want to transform them into .tif then the following command is executed:

```
sudo vips im_vips2tiff image1.png image1.tif:deflate,tile:256x256,pyramid
```

The execution of the command above can take some time because the pyramid tiffs are huge files.

This command is only for one image!

If you need to convert a large number of images, the shell script (**iiif-image-converter.sh**) that contains the conversion command automates this process for all images. The content of the **iiif-image-converter.sh** shell script looks like this:

```
#!/bin/bash
# This script requires the VIPS software package to be installed. To install it on a Debian distribution run this command:
# sudo apt-get install libvips-tools

# check if command exists
command_exists () {
  type "$1" &amp;> /dev/null;
}
install_vips () {
  if command_exists apt-get; then
    sudo apt-get -y update
    sudo apt-get -y -q install libvips-tools
  fi

  if ! command_exists vips; then
    echo "command vips not found"
    exit 1;
  fi
}

if ! command_exists vips ; then
  echo "command vips not found. Installing..."
  install_vips;
fi

# Convert all images in a directory to tiff
find . -not -name "*.sh" -not -name "*.tif" -type f | while read file
do
  if [ -f "${file%.*}".tif ]; then
    echo "File ${file%.*}.tif already exists. Skipping."
    continue
  fi
  echo "Converting $file to ${file%.*}.tif"
  vips im_vips2tiff "$file" "${file%.*}".tif:deflate,tile:256x256,pyramid;
done

# Move all tiff files to a new directory
mv *.tif /var/www/iipimage-server/
```

iiif-image-converter.sh shell script

You need to place the script in the same folder where the images are located (**e.g.: /var/www/html/images/**) and execute the script by typing the following command into your terminal:

```
./iiif-image-converter.sh
```

Note: If you don't not have the appropriate permissions to make changes to the folder where the images and the script **iiif-image-converter.sh** are located, or if you do not have the permissions for the folder where the resulting images are to be moved after conversion, then you can obtain these permissions by either executing **iiif-image-converter.sh** with the sudo command or by modifying the **iiif-image-convert.sh script** and inserting the sudo command before the vips and mv commands.

Move the converted images to the image servers' data directory

In a final step, the script moves the converted images to the image servers' data directory in order to make them accessible via the image server module.

6 Accessing Images in the Browser via a URL

When you call an image via an IIPImage server URL, the image server will show you this image according to some parameters that you have set in the URL.

For example you can type the following in your browser's address bar:

***your.ip.address*/iiif/image1.tif/full/400,/0/default.jpg**

The IIIF consortium provides a detailed documentation of the IIIF Image API's parameters here.

7 Accessing Images in the Browser via the Manifest

The Manifest is a container file (in the JSON format), that contains metadata about an image collection as well as the IIIF compliant URLs to the contained images.

The sample manifest can be viewed and downloaded from here.


```
{
    "@context": "http://iiif.io/api/presentation/3/context.json",
    "id": "http://MY_IP_ADDRESS/manifest.json",
    "type": "Manifest",
    "label": {
        "en": ["Single Image Example"]
    },
    "items": [{
        "id": "http://MY_IP_ADDRESS/iiif/image_name_without_extension",
        "type": "Canvas",
        "label": {
            "en": ["Canvas with a single IIIF image"]
        },
        "height": height_of_tif_image,
        "width": width_of_tif_image,
        "items": [{
            "id": "http://MY_IP_ADDRESS/iiif/image_name_without_extension",
            "type": "AnnotationPage",
            "items": [{
                "id": "https://MY_IP_ADDRESS/iiif/image_name_without_extension",
                "type": "Annotation",
                "motivation": "painting",
                "body": {
                    "id": "http://MY_IP_ADDRESS/iiif/image_name.tif/full/max/0/default.jpg",
                    "type": "Image",
                    "format": "image/jpeg",
                    "height": height_of_tif_image,
                    "width": width_of_tif_image,
                    "service": [{
                        "id": "https://MY_IP_ADDRESS/iiif/image_name_without_extension",
                        "profile": "level1",
                        "type": "ImageService3"
                    }]
                },
                "target": "http://MY_IP_ADDRESS/iiif/image_name_without_extension"
            }]
        }]
    }]
}
```


Sample manifest

However, this manifest needs to be updated according to your needs. For that you may use for example the nano command. Updating means in this context: changing *MY_IP_ADDRESS* to **your actual** IP Address or host name (if known), changing the content of "label" tag etc. Then if everything is adapted and changed correctly the manifest should be loaded in the browser by entering the following in the browser's address bar:

***your.ip.address*/manifest_name.json**

Using the manifest the images can be accessed and viewed with a **IIIF-compliant image viewer** like Mirador, Universal Viewer, OpenSeadragon etc.

Please see the IIIF Image Conversion Training for a detailed use case description.

8 Glossary

Linux Debian i.e. Ubuntu	Ubuntu develops and maintains a cross-platform, open-source operating system based on Debian, with a focus on release quality, enterprise security updates and leadership in key platform capabilities for integration, security and usability. Debian | Ubuntu

Apache	The Apache HTTP Server Project is an effort to develop and maintain an open-source HTTP server for modern operating systems including UNIX and Windows. Welcome! - The Apache HTTP Server Project
IIPImage Server	The IIPImage server is a feature-rich high performance image server engineered to be stable, fast and lightweight. It is designed for streaming extremely high resolution images and is capable of handling advanced image features such as 16 and 32 bit per channel depths, floating point data, CIELAB colorimetric images and scientific imagery such as multispectral or hyperspectral images and digital elevation map data. IIPImage » Server
Pyramid TIFF	Tiled Multi-Resolution (or Tiled Pyramidal) TIFF is simply a tiled multi-page TIFF image, with each resolution stored as a separate layer within the TIFF. IIPImage » Images
libapache2-mod-fcgid	This package contains mod_fcgid, a high-performance alternative to mod_cgi or mod_cgid. It starts a sufficient number of instances of the CGI program to handle concurrent requests. These programs remain running to handle further incoming requests. libapache2-mod-fcgid
IIIF	IIIF is a way to standardise the delivery of images and audio/visual files from servers to different environments on the Web where they can then be viewed and interacted with in many ways. Home
IIIF Image API's parameters	The IIIF Image API specifies a web service that returns an image in response to a standard HTTP or HTTPS request. The URI can specify the region, size, rotation, quality characteristics and format of the requested image. Image API 3.0

9 Scripts

Access this GitHub repository to download the two scripts:

iiif-image-converter.sh

iiif-install.sh