Loading Image Stacks and Movies

In this context, the term *image sequence* is used to refer to a collection of images which can be from a time-lapse assay, a three-dimensional (3D) Z-stack assay, or both. This tutorial will instruct you how to load these collections in order to properly represent your data for processing.

Example #1: Sequences of individual files

For some microscopes, the simplest method of capturing image sequences is to simply acquire them as a series of individual images, where each image represents a single timepoint or Z-slice (for simplicity, we will refer to *timepoints* in the rest of this example). Typically, the image filename reflects the timepoint, such that the alphabetical image listing corresponds to the proper sequence, e.g., *img000.png*, *img001.png*, *img002.png*, etc. It is also not uncommon to store the movie such that one movie's worth of files is stored in a single folder.

For example, let's say you have a time-lapse movie of individual files set up as follows:

- Three folders, one for each image channel, named *fluo2*, *fluor* and *phase*.
- In each folder, the files are named for the imaging modality used as follows:
 - o fluo2: calibrate2-P01.001.TIF, calibrate2-P01.002.TIF,..., calibrate2-P01.287.TIF
 - o fluor: calibrated-P01.001.TIF, calibrated-P01.002.TIF,..., calibrated-P01.287.TIF
 - o phase: phase-P01.001.TIF, phase-P01.002.TIF,..., phase-P01.287.TIF

where the file names are in the format <*Stain>-<Well>.<Timepoint>.TIF*.

• There are 287 timepoints per movie, and a movie of the 3 channels above is acquired from each well.

In this case, the first step to set up the input modules to handle these files would be to select the **Images** module and drag-and-drop your folders of images into the File list panel. If necessary, set your filtering rules accordingly in order to omit any files that are not part of a movie sequence.

In the above example, you would drag-and-drop the *fluo2, fluor* and *phase* folders into the File list panel. In this case, the default "Images only" filter is sufficient to capture the necessary files.



In the **Metadata** module, check the box to enable metadata extraction. The key step here is to obtain the necessary metadata tags to do two things:

- Distinguish the movies from each other. This information is typically encapsulated in the filename and/or the folder name.
- For each movie, distinguish the timepoints from each other, ensuring their proper ordering. This information is often contained in the filename.

To accomplish this, do the following:

- Select "Extract from file/folder names" or "Import from file" as the metadata extraction method. You will use these to extract the movie and timepoint tags from the images. Use "Extract from file/folder names" to create a regular expression to extract the metadata from the filename and/or path name.
- Or, use "Import from file" if you have a comma-delimited file (CSV) of the necessary metadata columns (including the movie and timepoint tags) for each image.

If there are multiple channels for each movie, this step may need to be performed for each channel.

In this example, you could do the following:

- Select "Extract from file/folder names" as the method, "From file name" as the source, and specify
 .*-(?P<Well>[A-P][0-9]{2})\.(?P<Timepoint>[0-9]{3})
 - as the regular expression. This step will extract the well ID and timepoint from each filename.
- Click the "Add" button to add another extraction method.
- In the new group of extraction settings, select "Extract from file/folder names" as the method, "From folder name" as the source, and

```
.*[\\/](?P<Stain>.*)[\\/].*$
```

as the regular expression. This step will extract the stain name from each folder name.

• Click the "Update" button below the divider and check the output in the table to confirm that the proper metadata values are being collected from each image.

CellProfiler 2.1.0 (rev e4cc0fa)		at the spate spectrum.					A-T-A- #1		x	
<u>File Edit Test Data Tools Window H</u> elp										
Pipeline	- Module notes -									
Input modules	The Metadata r	nodule optionally allows you to ext	ract information	describing	your images (i	.e, metada	ta) which will be stored along	with your measurements. T	'his 🔺	
Images	Information can	be contained in the file name and,	or location, or li	an exter	nai file.				T	
Metadata	-Module setting									
Groups	Metadata ext	raction method Extract from file/fi	older names	-				7	1 ^	
Analysis modules	м								i	
									4	
	Regular expression *-(?P <wel>[A+P][0-9](2))).(?P<timepoint>[0-9]*) Extract metadata from All images</timepoint></wel>									
	I (rev e4cc0ia) Data Tools: Window Help Module notes: The Metadata module optionally allows you to extract information descriping your images (i.e., metadata) which will be stored along with your measurements. The information can be contained in the file name and/or location, or in an external file. Module settings: Metadata source [rile name * Regular expression: *-(Pc:VMel>(A+P)[0-9](2)), (P-(Timepoint>[0-9]*) Extract method Extract from file/folder names * Regular expression: *-(Pc:VMel>(A+P)[0-9](2)), (P-(Timepoint>[0-9]*) Extract method Extract from file/folder names * Regular expression: *-(VPC:VMel>(A+P)[0-9](2)), (P-(Timepoint>[0-9]*) Extract method Extract from file/folder names * Regular expression: *(VV(PC-Stain>.*))(V), *\$ Extract method Extract from file/folder names * Regular expression: *(VV(PC-Stain>.*))(V), *\$ Regular expression: *(VV(PC-Stain>.*))(V), *\$ Extract method Extract from file/folder names * Regular expression: *(VV(PC-Stain>.*))(V), *\$ Extract method Extract from Ali mages * Regular expression: *(VV(PC-Stain>.*))(V), *\$ C: Upersylvins2P0100.1TP: None None fluc2 Metadata source regular expression: *(VV(PC-Stain>.*))(V), *\$ C: Upersylvins2P0100.1TP: None None fluc2 Regular expression: *(VV(PC-Stain>.*))(V), *\$ C: Upersylvins2P0100.1TP: None None fluc2 Regular expression: *(VV(PC-Stain>.*))(V), *\$ C: Upersylvins2P0100.1TP: None None fluc2 Regular expression: *(VV(PC-Stain>.*))(V), *\$ <td< td=""><td>_</td></td<>						_			
Regular expression **(?P <wel>[A +][0-9] (2))\(?P <timepoint>[0-9] *) Extract metadata from Al images • Metadata extraction method Extract from file/folder names • Metadata source Folder name • Regular expression *[[V](?P <stain>,*][V].*\$ • Extract metadata from Al images •</stain></timepoint></wel>										
CellProfiler 21.0 (rev edcoting) File Edit Jest Data Tools Window Help Produce Images Produce Images Produce Produce Pro										
	м	etadata source Folder name 👻						?	1 _	
	Reg	ular expression .*[\\](?P <stain>.</stain>	*)[\V].*\$?	1	
	Extract	metadata from All images	•					?	1	
		Remove this extr	action method						I	
	<u> </u>	Nemove and exa	action method							
	Update	Path / URL	Series Fran	e Stain	Timepoint	Well			Â	
	1	C:\Users\mbra2-P01.001.TIF	None None	fluo2	001	P01				
	2	C:\Users\mbra2-P01.002.TIF	None None	fluo2	002	P01				
	3	C:\Users\mbra2-P01.003.TIF	None None	fluo2	003	P01				
	4	C:\Users\mbra2-P01.004.TIF	None None	fluo2	004	P01				
	5	C:\Users\mbra2-P01.005.TIF	None None	fluo2	005	P01				
	6	C:\Users\mbra2-P01.006.TIF	None None	fluo2	006	P01				
	7	C:\Users\mbra2-P01.007.TIF	None None	fluo2	007	P01				
	8	C:\Users\mbra2-P01.008.TIF	None None	fluo2	008	P01				
Output	9	C:\Users\mbra2-P01.009.TIF	None None	fluo2	009	P01				
View output settings	10	C:\Users\mbra2-P01.010.TIF	None None	fluo2	010	P01				
Men ouput setungs	11	C:\Users\mbra2-P01.011.TIF	None None	fluo2	011	P01				
	12	C:\Users\mbra2-P01.012.TIF	None None	fluo2	012	P01				
P Adjust modules: + - ^ v	13	C:\Users\mbra2-P01.013.TIF	None None	fluo2	013	P01				
	14	C:\Users\mbra2-P01.014.TIF	None None	fluo2	014	P01			-	
Analyze Images	Welcome to Cell	Profiler								

In the **NamesAndTypes** module, assign the channel(s) to a name of your choice. If there are multiple channels, this would need to be done for each channel. For this example, you could do the following:

- Select "Assign images matching rules".
- Make a new rule [Metadata][Does][Have Stain matching][fluor] and name it OrigFluor.
- Click the "Add another image" button to define another image with a rule.
- Make a new rule [Metadata] [Does] [Have Stain matching] [fluo2] and name it OrigFluo2.
- Click the "Add another image" button define another image with a rule.



- Make a new rule [Metadata][Does][Have Stain matching][phase] and name it OrigPhase.
- In the "Assign channel by" setting, select "Metadata".
- Select "Well" for the OrigFluor, OrigFluo2, and OrigPhase channels.
- Click the 🖃 button to the right to add another row, and select "Timepoint" for each channel.

• Click the "Update" button below the divider to view the resulting table and confirm that the proper files are listed and matched across the channels. The corresponding well and frame for each channel should now be matched to each other.

Image set ma	atching method Metadat	a 🔻		?	
Μ	latch metadata Well Timepoir	uo2 OrigFluor	OrigPhase ell ▼ + - nepoint ▼ + -	- ↓ -û	4 III
Update	OrigFluo2	OrigFluor	OrigPhase		Â
P01:001	calibrate2-P01.001.TIF	calibrated-P01.001.TIF	phase-P01.001.TIF	 IF	
P01:002	calibrate2-P01.002.TIF	calibrated-P01.002.TIF	phase-P01.002.TIF	IF	
P01:003	calibrate2-P01.003.TIF	calibrated-P01.003.TIF	phase-P01.003.TIF	IF	
P01:004	calibrate2-P01.004.TIF	calibrated-P01.004.TIF	phase-P01.004.TIF	IF	
P01:005	calibrate2-P01.005.TIF	calibrated-P01.005.TIF	phase-P01.005.TIF	(F	
P01:006	calibrate2-P01.006.TIF	calibrated-P01.006.TIF	phase-P01.006.TIF	(F	
P01:007	calibrate2-P01.007.TIF	calibrated-P01.007.TIF	phase-P01.007.TIF	(F	
P01:008	calibrate2-P01.008.TIF	calibrated-P01.008.TIF	phase-P01.008.TIF	íF	
P01:009	calibrate2-P01.009.TIF	calibrated-P01.009.TIF	phase-P01.009.TIF	íF	
P01:010	calibrate2-P01.010.TIF	calibrated-P01.010.TIF	phase-P01.010.TIF	íF	
P01:011	calibrate2-P01.011.TIF	calibrated-P01.011.TIF	phase-P01.011.TIF	(F	
P01:012	calibrate2-P01.012.TIF	calibrated-P01.012.TIF	phase-P01.012.TIF	(F	
P01:013	calibrate2-P01.013.TIF	calibrated-P01.013.TIF	phase-P01.013.TIF	íF	
P01:014	calibrate2-P01.014.TIF	calibrated-P01.014.TIF	phase-P01.014.TIF	íF	
P01:015	calibrate2-P01.015.TIF	calibrated-P01.015.TIF	phase-P01.015.TIF	ίF	
P01:016	calibrate2-P01.016.TIF	calibrated-P01.016.TIF	phase-P01.016.TIF	iF	Ŧ

In the **Groups** module, enable image grouping for these images in order to select the metadata that defines a distinct movie of data. For this example, do the following:

- Select "Well" as the metadata category.
- The tables below this setting will update themselves, and you should be able to visually confirm that each well is defined as a group, each with 287 frames' worth of images.

Module settings										
		٦					?	-		
Do you want to group your images?	• Yes 💿 No									
Metadata category	Well 🔻	_								
	WCII -									
	Add another m	Add another metadata item								
Grouping list										
		Group: Well	Count							
	1	P01	287							
Image sets										
mage sets		Group numbe	er Group index	Group: Well	Path: OrigFluo2	File: 0		=		
	1	1	1	P01	C:\Users\mbraata\fluo2\P01	calibrate2-P		-		
	2	1	2	P01	C:\Users\mbraata\fluo2\P01	calibrate2-P				
	3	1	3	P01	C:\Users\mbraata\fluo2\P01	calibrate2-P				
	4	1	4	P01	C:\Users\mbraata\fluo2\P01	calibrate2-P				
	5	1	5	P01	C:\Users\mbraata\fluo2\P01	calibrate2-P				
	6	1	6	P01	C:\Users\mbraata\fluo2\P01	calibrate2-F				
	7	1	7	P01	C:\Users\mbraata\fluo2\P01	calibrate2-F				
	8	1	8	P01	C:\Users\mbraata\fluo2\P01	calibrate2-F				
	9	1	9	P01	C:\Users\mbraata\fluo2\P01	calibrate2-F				
	10	1	10	P01	C: Users (mbraata (fluo2)/P01	calibrate2-F				
	11	1	11	P01	C:\Users\mbraata\fluo2\P01	calibrate2+				
	12	1	12	P01	C: Users/mbraata/mu02/PU1	calibrate2.P				
	14	1	13	P01	C:\Users\mbra_ata\fluo2\P01	calibrate2-P				
	15	1	15	P01	C:\Users\mbraata\fluo2\P01	calibrate2-	-			
	4	· ·	15	1.01	cripsers (noral rata (noo2 (FOT	Control CE2-F				
	· [· · · ·				

Without this step, CellProfiler would not know where one movie ends and the next one begins, and would process the images in all movies together as if they were constituents of only one movie.

Example #2: Basic image sequences consisting of a single file

Another common means of storing time-lapse/Z-stack data is as a single file containing the entire movie. Examples of this approach include image formats such as:

- Multi-frame TIF
- Metamorph stack: STK
- Evotec/PerkinElmer Opera Flex
- Zeiss ZVI, LSM
- Standard movie formats: AVI, Quicktime MOV, etc

CellProfiler uses the Bio-Formats library for reading various image formats. For more details on supported files, see <u>http://www.openmicroscopy.org/site/support/bio-formats4/supported-formats.html</u>. In general, we recommend saving stacks and movies in a format such as .TIF.

For our first example, let's say you have two image stacks in the following format:

- The stacks are Opera's FLEX format.
- Each FLEX file contains 8 fields of view, with 3 channels at each site (DAPI, GFP, Texas Red).
- Each channel is in grayscale format.

In this case, the procedure to set up the input modules to handle these files is as follows (please note that this procedure is basically identical whether the file is for a time-lapse assay or a Z-stack assay):

In the **Images** module, drag-and-drop your folders of images into the File list panel. If necessary, set your rules accordingly in order to filter out any files that are not images to be processed.

CellProfiler 2.1.0 (rev e4cc0fa)	
<u>File Edit Test Data Tools Window H</u> elp	
Pipeline	Module notes
Input modules	To begin creating your project, use the Images module to compile a list of files and/or folders that you want to analyze. You can also specify a set of rules to
V Images	Include only the desired hies in your selected folders.
Metadata	Fielist
Groups	C:\Users\mbray\Pictures\StadData2
	Example2.ftex
Analysis modules	
	Show his excluded by hiters
	Module settings
	Filter images? Images only
Output	Apply filters to the file list
view output setungs	
Adjust modules: + - ^ V	
Start Test Mode	
	Welcome to CellProfiler

In the above example, you would drag-and-drop the FLEX files into the file list panel.

In the **Metadata** module, enable metadata extraction in order to obtain metadata from these files. The key step here is to obtain the metadata tags necessary to do two things:

- Distinguish the stacks from each other. This information is contained as the file itself, that is, each file represents a different stack.
- For each stack, distinguish the slices from each other, ensuring proper ordering. This information is usually contained in the image file's internal metadata, in contrast to Example #1 described above.

To accomplish this, do the following:

- Select "Extract from image file headers" as the metadata extraction method. In this case, CellProfiler will extract the requisite information from the metadata stored in the image headers.
- Click the "Update metadata" button. A progress bar will appear showing the time elapsed; depending on the number of files present, this step may take a while to complete.
- Click the "Update" button below the divider.

The resulting table should show the various metadata contained in the file. In this case, the relevant information is contained in the *C* and *Series* columns. In the figure shown, the *C* column shows three unique values for the channels represented, numbered from 0 to 2. The *Series* column shows 8 values for the slices collected in each stack, numbered from 0 to 7.

Module settings								
Extr Metadata extr	act metadata? • Yes • No	e file bear	ders 🔻					
Extract	metadata from All images	-	•]					
	Update metadat	а						
	Add another ext	traction me	ethod					
(u. t. c.)			-	-			_	
Update	Path / URL	Series	Frame	C	ChannelName	ColorFormat	Т	Z
1	C:\Users\mbraExample1.flex	None	None	None	None	None	None	None
2	C:\Users\mbraExample1.flex	0	0	0	Exp1Cam1	monochrome	0	0
3	C:\Users\mbraExample1.flex	0	1	1	Exp2Cam2	monochrome	0	0
4	C:\Users\mbraExample1.flex	0	2	2	Exp3Cam3	monochrome	0	0
5	C:\Users\mbraExample1.flex	1	0	0	Exp1Cam1	monochrome	0	0
6	C:\Users\mbraExample1.flex	1	1	1	Exp2Cam2	monochrome	0	0
7	C:\Users\mbraExample1.flex	1	2	2	Exp3Cam3	monochrome	0	0
8	C:\Users\mbraExample1.flex	2	0	0	Exp1Cam1	monochrome	0	0
9	C:\Users\mbraExample1.flex	2	1	1	Exp2Cam2	monochrome	0	0
10	C:\Users\mbraExample1.flex	2	2	2	Exp3Cam3	monochrome	0	0
11	C:\Users\mbraExample1.flex	3	0	0	Exp1Cam1	monochrome	0	0
12	C:\Users\mbraExample1.flex	3	1	1	Exp2Cam2	monochrome	0	0
13	C:\Users\mbraExample1.flex	3	2	2	Exp3Cam3	monochrome	0	0
14	C:\Users\mbraExample1.flex	4	0	0	Exp1Cam1	monochrome	0	0
15	C:\Users\mbraExample1.flex	4	1	1	Exp2Cam2	monochrome	0	0
16	C:\Users\mbraExample1.flex	4	2	2	Exp3Cam3	monochrome	0	0
17	C:\Users\mbraExample1.flex	5	0	0	Exp1Cam1	monochrome	0	0
18	C:\Users\mbraExample1.flex	5	1	1	Exp2Cam2	monochrome	0	0
19	C:\Users\mbraExample1.flex	5	2	2	Exp3Cam3	monochrome	0	0

In the **NamesAndTypes** module, assign the channel(s) to a name of your choice. If there are multiple channels, you will need to do this for each channel. For this above example, you could do the following:

- Select "Assign images matching rules".
- Make a new rule [Metadata] [Does] [Have C matching] [0].
- Click the 🖃 button to the right of the rule to add another set of rules underneath.
- Add the rule [Image][Is][Stack frame]. This combination tells CellProfiler not to treat the image as a single file, but rather as a series of frames.
- Name the image DAPI.
- Click the "Add another image" button to define a second image with a set of rules.
- Make a new rule [Metadata][Does][Have C matching][1]
- Click the 🖃 button to the right of the rule to add another set of rules underneath.
- Add the rule [Image][Is][Stack frame].
- Name the second image *GFP*.

Module settings	
Assign a name to Images matching rules 🗢	?
Select the rule criteria Match All 🔹 of the following rules	?
Metadata Does Have C matching 0 -+	
Image Is Stack frame Is Image Image Is Image Image Is Im	
Name to assign these images DAPI	?
Select the image type Grayscale image	?
Set intensity range from Image metadata 👻	?
Select the rule criteria Match All 🚽 of the following rules	?
Metadata Does Have C matching 1 -+	
Image Is Stack frame Is Image Image Is Image Image Is Im	
Name to assign these images GFP	?
Select the image type Grayscale image	?
Set intensity range from Image metadata	?
Remove this image	-

- Click the "Add another image" button to define a third image with a set of rules.
- Make a new rule [Metadata][Does][Have C matching][2]
- Click the 🖃 button to the right of the rule to add another set of rule options.
- Add the rule [Image][Is][Stack frame].
- Name the third image *TxRed*.
- In the "Image set matching method" setting, select "Metadata".
- Select "FileLocation" for the *DAPI*, *GFP*, and *TxRed* channels. The *FileLocation* metadata tag identifies the individual stack, and selecting this parameter ensures that the channels are first matched within each stack, rather than across stacks.
- Click the 🖃 button to the right to add another row, and select "Series" for each channel.
- Click the "Update" button below the divider to confirm that the proper image slices are listed and matched across the channels. The corresponding *FileLocation* and *Series* for each channel should now be matched to each other.

Image set ma	atching method	Metadata 🔻				?
M	atch metadata	DAPI FileLocation Series	GF FileLocati Series	P TxRed ion ▼FileLocation ▼ + - ↓ ▼Series ▼ + - ℃		Ш
Update		DAPI		GFP	TxRed	
Dictures/Stack	Example 1. flex			Example 1. flex	Example 1. flex	-
Pictures/Stack	Example 1. flex			Example 1. flex	Example 1. flex	-
victures/Stack	Example 1.flex			Example 1. flex	Example 1. flex	-
Pictures/Stack	Example 1.flex			Example1.flex	Example 1. flex	
Pictures/Stack	Example 1.flex			Example 1. flex	Example 1. flex	
Pictures/Stack	Example1.flex			Example 1. flex	Example 1. flex	
Pictures/Stack	Example1.flex			Example 1. flex	Example 1. flex	
Pictures/Stack	Example1.flex			Example1.flex	Example 1. flex	
Pictures/Stack	Example2.flex			Example2.flex	Example2.flex	
Pictures/Stack	Example2.flex			Example2.flex	Example2.flex	_
Pictures/Stack	Example2.flex			Example2.flex	Example2.flex	_
Pictures/Stack	Example2.flex			Example2.flex	Example2.flex	_
Pictures/Stack	Example2.flex			Example2.flex	Example2.flex	_
Pictures/Stack	Example2.flex			Example 2. flex	Example2.flex	
Pictures/Stack	Example 2. flex			Example 2. flex	Example 2. flex	
Pictures/Stack	Example2.nex			Example2.nex	Example2.nex	

In the **Groups** module, select the metadata that defines a distinct image stack. For the example above, do the following:

- Select "FileLocation" as the metadata category.
- The tables below this setting will update themselves, and you should be able to visually confirm that each of the two image stacks are defined as a group, with 8 slices' worth of images.

you want to group your images?	● Yes 💿 No							?
Metadata category	FileLocation	•						?
	Add another m	etadata item						
Grouping list		Group: FileLo	cation	Count				?
	1	file:///C:/UsExar	mple1.flex 8					
	2	file:///C:/UsExar	mple2.flex 8					
Image sets		Group number	Group inde	×	Group: FileLocation	Path: DAPI	Â	?
	1	1	1	file	///C:/UsExample1.flex	C:\Users\mbraes\StackData		
	2	1	2	file	///C:/UsExample1.flex	C:\Users\mbraes\StackData		
	3	1	3	file	///C:/UsExample1.flex	C:\Users\mbraes\StackData		
	4	1	4	file	///C:/UsExample1.flex	C:\Users\mbraes\StackData		
	5	1	5	file	///C:/UsExample1.flex	C:\Users\mbraes\StackData		
	6	1	6	file	///C:/UsExample1.flex	C:\Users\mbraes\StackData	Ξ	
	7	1	7	file	///C:/UsExample1.flex	C:\Users\mbraes\StackData		
	8	1	8	file	///C:/UsExample1.flex	C:\Users\mbraes\StackData		
	9	2	1	file	///C:/UsExample2.flex	C: Users (mbraes (StackData		
	10	2	2	file	///C:/UsExample2.flex	C: Users (mbraes (StackData		
	11	2	3	fie	///C:/UsExample2.flex	C: Users (mbraes (StackData		
	12	2	7 C	file	///C:/UsExample2.flex	C: Users moraes (StackDate		
	14	2	6	file	///C·/Us Example? flex	C: Users/mbra_es/StackDate		
	15	2	7	file	///C:/UsExample2 flex	C:\Users\mbraes\StackDatz	-	
					III all agreen mercample 2 merc	The second friend and the for the for the forthe forth		

Example #3: More advanced image sequences consisting of a single file

As another example, you have a set of Z-stacks in the following format:

- The stacks are in Zeiss' CZI format.
- Each stack consists of a number of slices with 4 channels (DAPI, GFP, Texas Red and Cy3) at each slice.
- One stack has 9 slices, two stacks have 7 slices and the fourth has 12 slices. Even though the stacks were collected with differing numbers of slices, the pipeline to be constructed is intended to analyze all stacks in the same manner.
- Each slice is in grayscale format.

In this case, the procedure to set up the input modules to handle these files is as follows (please note that this procedure is basically identical whether the file is for a time-lapse assay or a Z-stack assay):

In the **Images** module, drag-and-drop your folders of images into the File list panel. If necessary, set your rules accordingly in order to filter out any files that are not part of a Z-stack sequence.

In the above example, you would drag-and-drop the CZI files into the file list panel. In this case, the default "Images only" filter is sufficient to capture the necessary files.



In the **Metadata** module, enable metadata extraction in order to obtain metadata from these files. The key step here is to obtain the metadata tags necessary to do two things:

- Distinguish the stacks from each other. This information is contained as the file itself, that is, each file represents a different stack.
- For each stack, distinguish the slices from each other. This information is usually contained in the image's internal metadata.

To accomplish this, do the following:

- Select "Extract from image file headers" as the metadata extraction method. In this case, CellProfiler will extract the requisite information from the metadata stored in the image headers.
- Click the "Update metadata" button. A progress bar will appear showing the time elapsed; depending on the number of files present, this step may take a while.
 - -Module settings ? Extract metadata? • Yes • No Metadata extraction method Extract from image file headers 🔹 ? Ξ Extract metadata from All images • ? ? Update metadata Add another extraction method ÷ Path / URL Update Series Frame C ChannelName ColorFormat SizeC SizeT SizeZ т z 1 9 = 1 C:\Users\mbra...pleStack1.czi None None None None Planar 4 None None 0 2 C:\Users\mbra...pleStack1.czi 0 0 Ch2-T1 monochrome None None None 0 0 C:\Users\mbra...pleStack1.czi 0 3 1 Ch3-T2 monochrome None None None 0 0 1 4 C:\Users\mbra...pleStack1.czi 0 2 2 Ch2-T3 monochrome None None None 0 0 3 5 C:\Users\mbra...pleStack1.czi 0 3 Ch1-T4 monochrome None None None 0 0 C:\Users\mbra...pleStack1.czi 0 4 0 Ch2-T1 6 monochrome None None 0 None 1 C:\Users\mbra...pleStack1.czi 0 5 1 7 Ch3-T2 monochrome None None None 0 1 C:\Users\mbra...pleStack1.czi 0 Ch2-T3 None 0 8 6 2 monochrome None None 1 C:\Users\mbra...pleStack1.czi 0 7 monochrome None 9 3 Ch1-T4 None None 0 1 Ch2-T1 10 C:\Users\mbra...pleStack1.czi 0 0 None None None 0 8 monochrome 2 C:\Users\mbra...pleStack1.czi 0 1 Ch3-T2 monochrome None None 0 11 9 2 monochrome None None C:\Users\mbra...pleStack1.czi 0 12 10 2 Ch2-T3 None 0 2 C:\Users\mbra...pleStack1.czi 0 3 13 Ch1-T4 None 0 11 monochrome None None 2 0 14 C:\Users\mbra...pleStack1.czi 0 12 Ch2-T1 monochrome None None None 0 3 None 15 C:\Users\mbra...pleStack1.czi 0 13 1 Ch3-T2 monochrome None None 0 3 C:\Users\mbra...pleStack1.czi 0 16 14 2 Ch2-T3 monochrome None None None 0 3 C:\Users\mbra...pleStack1.czi 0 Ch1-T4 17 15 3 monochrome None None None 0 3 C:\Users\mbra...pleStack1.czi 0 16 0 Ch2-T1 18 monochrome None None None 0 4 17 1____ 19 C:\Users\mbra...pleStack1.czi 0 Ch3-T2 monochrome None None None 0 4 20 C:\Users\mbra...pleStack1.czi 0 18 2 Ch2-T3 monochrome None None None 0 4
- Click the "Update" button below the divider.

The resulting table should show the various metadata contained in the file. In this case, the relevant information is contained in the *C* and *Z* columns. In the figure shown, the *C* column shows four unique values for the channels represented, numbered from 0 to 3. The *Z* column shows nine values for the slices represented from the first stack, numbered from 0 to 8, followed by the slices for other stacks.

Of note in this case, for each file there is a single row summarizing this information. For the first stack shown, the *sizeC* column reports a value of 4 and *sizeZ* column shows a value of 9. You may need to scroll down the table to see this summary for the other stacks.

In the **NamesAndTypes** module, assign the channel(s) to a name of your choice. If there are multiple channels, you will need to do this for each channel. For this example, you could do the following:

- Select "Assign images matching rules".
- Make a new rule [Metadata] [Does] [Have C matching] [0].
- Click the 🖃 button to the right of the rule to add another set of rule options.
- Add the rule [Image][Is][Stack frame].
- Name the image DAPI.
- Click the "Add another image" button to define a second image with a set of rules.
- Make a new rule [Metadata][Does][Have C matching][1]

- Click the 🖃 button to the right of the rule to add another set of rule options.
- Add the rule [Image][Is][Stack frame].
- Name the second image GFP.

Module settings	
Assign a name to Images matching rules 👻	
Select the rule criteria Match All 👻 of the following rules	
Metadata ▼ Does ▼ Have C matching ▼ 0 - +	
Image Is Stack frame Image Image	E
Name to assign these images DAPI ?	
Select the image type Grayscale image ?	
Set intensity range from Image metadata 👻	
Select the rule criteria Match All 🔹 of the following rules	
Metadata Does Have C matching 1 -+	
Image ▼ Is ▼ Stack frame ▼ - +	
Name to assign these images GFP ?	
Select the image type Grayscale image 👻	
Set intensity range from Image metadata 👻	
Remove this image	-

- Click the "Add another image" button to define a third image with a set of rules.
- Make a new rule [Metadata][Does][Have C matching][2]
- Click the 🖃 button to the right of the rule to add another set of rule options.
- Add the rule [Image][Is][Stack frame].
- Name the third image *TxRed*.
- Click the "Add another image" button to define a fourth image with set of rules.
- Make a new rule [Metadata] [Does] [Have C matching] [3].
- Click the 🖃 button to the right of the rule to add another set of rule options.
- Add the rule [Image][Is][Stack frame].
- Name the fourth image *Cy3*.
- In the "Image set matching method" setting, select "Metadata".
- Select "FileLocation" for the DAPI, GFP, TxRed and Cy3 channels. The FileLocation metadata tag identifies the individual stack, and selecting this parameter insures that the channels are first matched within each stack, rather than across stacks.
- Click the 🖃 button to the right to add another row, and select "Z" for each channel.
- Click the "Update" button below the divider to confirm the channel matching. The corresponding *FileLocation* and *Z* for each channel should now be matched to each other.

Image set ma	tching method	Metadata 🔻						?
Ma	atch metadata	Cy3 FileLocation	DAPI	GFP FileLocation	TxRed ▼ FileLocation	▼ + -) (ł)	
		Z	Z	Z	▼Z	• + -	Û	ш
ļ								
Update		Cy3	DAPI		GFP		TxRed	<u> </u>
tures/StackDa	ExampleStack:	1.czi	ExampleStack1.czi		ExampleStack1.czi	Î	ExampleStack1.czi	
tures/StackDa	ExampleStack:	1.czi	ExampleStack1.czi		ExampleStack1.czi	E	ExampleStack1.czi	
tures/StackDa	ExampleStack:	1.czi	ExampleStack1.czi		ExampleStack1.czi	E	ExampleStack1.czi	
tures/StackDa	ExampleStack:	1.czi	ExampleStack1.czi		ExampleStack1.czi	E	ExampleStack1.czi	
tures/StackDa	ExampleStack:	1.czi	ExampleStack1.czi		ExampleStack1.czi	E	ExampleStack1.czi	
tures/StackDa	ExampleStack:	1.czi	ExampleStack1.czi		ExampleStack1.czi	E	ExampleStack1.czi	E
tures/StackD	ExampleStack:	1.czi	ExampleStack1.czi		ExampleStack1.czi		ExampleStack1.czi	
tures/StackD	ExampleStack:	1.czi	ExampleStack1.czi		ExampleStack1.czi		ExampleStack1.czi	
tures/StackDa	ExampleStack1	1.czi	ExampleStack1.czi		ExampleStack1.czi	E	ExampleStack1.czi	
tures/StackDa	ExampleStack2	2.czi	ExampleStack2.czi		ExampleStack2.czi	6	ExampleStack2.czi	
tures/StackDa	ExampleStack2	2.czi	ExampleStack2.czi		ExampleStack2.czi	6	ExampleStack2.czi	
tures/StackDa	ExampleStack2	2.czi	ExampleStack2.czi		ExampleStack2.czi		ExampleStack2.czi	
tures/StackDa	ExampleStack2	2.czi	ExampleStack2.czi		ExampleStack2.czi		ExampleStack2.czi	
tures/StackD	ExampleStack2	2.czi	ExampleStack2.czi		ExampleStack2.czi		ExampleStack2.czi	
tures/StackD	ExampleStack2	2.czi	ExampleStack2.czi		ExampleStack2.czi		ExampleStack2.czi	
tures/StackD	ExampleStack2	2.czi	ExampleStack2.czi		ExampleStack2.czi		ExampleStack2.czi	
tures/StackD	ExampleStack:	S.CZI	ExampleStack3.czi		ExampleStack3.czi	t	ExampleStack3.czi	
tures/StackD	ExampleStack:	S.CZI	ExampleStack3.czi		ExampleStack3.czi		ExampleStack3.czi	
tures/StackD	ExampleStack:	D.CZI	ExampleStack3.czi		ExampleStack3.czi		ExampleStack3.czi	
tures/StackD	ExampleStack:	5.CZI	ExampleStack3.czi		ExampleStack3.czi		ExampleStack3.czi	
tures/StackD	ExampleStack:	0.021	ExampleStack3.czl		ExampleStack3.czi		ExampleStack3.czi	

In the **Groups** module, select the metadata that defines a distinct image stack. For the example above, do the following:

- Select "FileLocation" as the metadata category.
- The tables below this setting will update themselves, and you should be able to visually confirm that each of the four image stacks are defined as a group, with 9, 7, 7 and 12 slices' worth of images.

Module settings									
D		Ъ						?	
Do you want to group your images?	Yes No								
Metadata category	FileLocation	-							
2 /									
	Add another m	etadata item							
Grouping list				-					
		Group: FileLo	cation	Coun	t				
	1	file:///C:/UspleS	tack1.czi	9			Ξ		
	2	file:///C:/UspleS	tack2.czi	7					
	3	file:///C:/UspleS	tack3.czi	7					
	4	file:///C:/UspleS	tack4.czi	12			Ŧ	1	
Image sets									
indge beta		Group number	Group in	Idex	Group: FileLocation	Path: Cy3			-
	1	1	1	f	ile:///C:/UspleStack1.czi	C:\Users\mbraes\StackData			-
	2	1	2	f	ile:///C:/UspleStack1.czi	C:\Users\mbraes\StackData			
	3	1	3	f	ile:///C:/UspleStack1.czi	C:\Users\mbraes\StackData	E		
	4	1	4	f	ile:///C:/UspleStack1.czi	C:\Users\mbraes\StackData			
	5	1	5	f	ile:///C:/UspleStack1.czi	C:\Users\mbraes\StackData			
	6	1	6	f	ile:///C:/UspleStack1.czi	C:\Users\mbraes\StackData			
	7	1	7	f	ile:///C:/UspleStack1.czi	C:\Users\mbraes\StackData			
	8	1	8	t	ile:///C:/UspleStack1.czi	C:\Users\mbraes\StackData:			
	9	1	9	1	ne:///C:/UspleStack1.czi	C: Users (mbraes (StackData)			
	10	2	1	4	ile:///C:/UspleStack2.czi	C: Users (mbraes (StackData)			
	12	2	2	4	ile:///C:/UspleStack2.czi	C: Users (IIDIaes (StackData)			
	13	2	4	f	ile:///C:/UspleStack2.czi	C:\Users\mbraes\StackData			
	14	2	5	f	ile:///C:/UspleStack2.czi	C:\Users\mbraes\StackData			
	15	2	6	f	ile:///C:/UspleStack2.czi	C:\Users\mbraes\StackData	Ŧ		
	•					•		n	
								1	T

Without this step, CellProfiler would not know where one stack ends and the next one begins, and would process the slices in all stacks together as if they were constituents of only one stack.