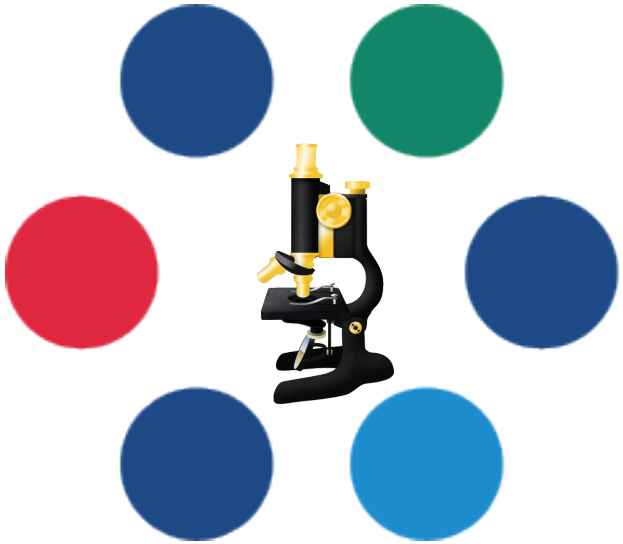


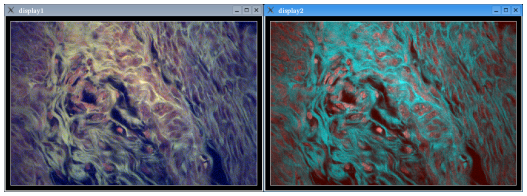
OME and ImageJ

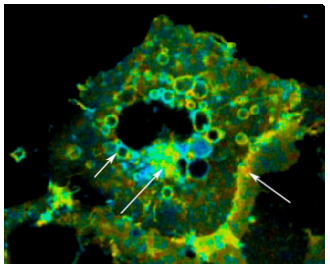
Kevin Eliceiri
eliceiri@wisc.edu
LOCI, www.loci.wisc.edu

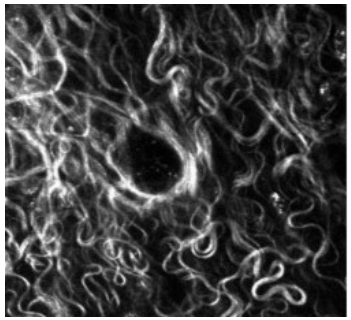
Mission of LOCI:

- New optical instrumentation to facilitate studies of the dynamics of living specimens.
- Better software for capture and visualization of dynamic, 3-D biological events
- Been OME Development partner since 2002
- Image Informatics for multidimensional data
  - spatial and temporal
  - spectral and lifetime dimensions
  - polarization

Our data:







Fluorescence,
Spectra, Lifetime
(λ,τ)

Time-Lapse
(t)

Dynamics

Chemistry

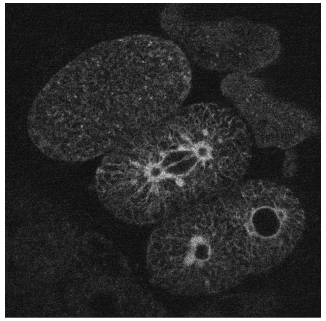
Optical Microscopy

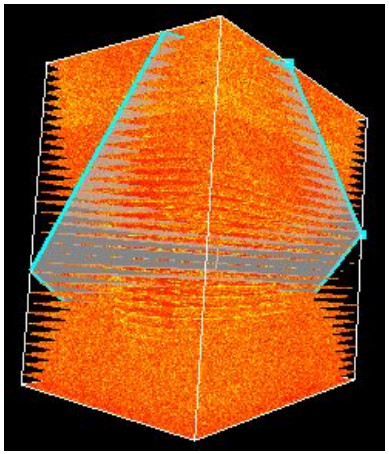
Physics

Structure

Phase, Polarization,
Scatter, Harmonics
(ϕ, θ,…)

Space
(x, y, z)





Supporting Technologies

Hardware
Acquisition software
Analysis and Visualization
Data Management

Overview of our OME efforts

- **Specific OMERO linked applications**
  - BK Cho in Murphy lab on Omero.searcher
  - Forward Project for data dissemination

- Originated **OME-TIFF** in 2004
  - Now fully integrated into OMER0
  - Used by many open and commercial tools
  - Our current focus on robust tools to read and write OME-TIFF

- **Originated Bio-Formats in 2005**
  - partnership with OME and Glencoe
  - Over 120 formats
  - Over 30,000 installations
  - Recent development focus on SCIFIO

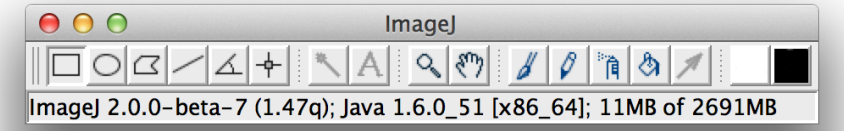
- **XML Schema Improvements for Acquisition**
  - Our WiscScan software and now MicroManager
  - Fused Data models with "OME-TIFF"

- **Interoperability with other tools**—Cell Profiler, Knime, etc

- **ImageJ 2.0** (ImageJDev.org)

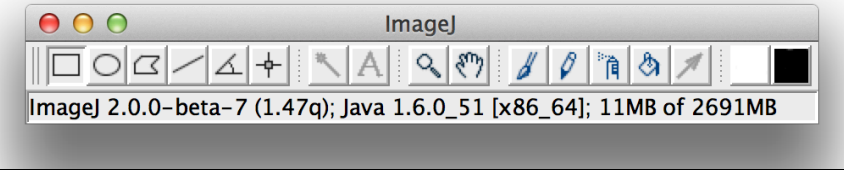
What is ImageJ2?

- ✓ A standalone application



What is ImageJ2?

- ✓ A standalone application

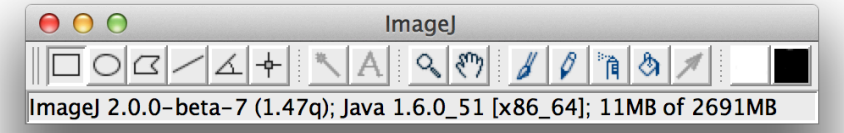


- ✓ A reusable library

```
/** Loads and displays a dataset using the ImageJ API. */
public void loadAndDisplay(final File file) throws Exception {
  // create the ImageJ application context with all available services
  final ImageJ ij = new ImageJ();

  // load the dataset
  final Dataset dataset = ij.io().loadDataset(file.getAbsolutePath());

  // display the dataset
  ij.ui().show(dataset);
}
```

What is ImageJ2?

✓ A standalone application



✓ A reusable library

```
/** Loads and displays a dataset using the ImageJ API. */
public void loadAndDisplay(final File file) throws Exception {
  // create the ImageJ application context with all available services
  final ImageJ ij = new ImageJ();

  // load the dataset
  final Dataset dataset = ij.io().loadDataset(file.getAbsolutePath());

  // display the dataset
  ij.ui().show(dataset);
}
```

✓ An extensible collection of services & plugins

SciJava

ImageJ2 Guiding Principles

- Preserve backwards compatibility
- Maintain good performance
- Support N-dimensional imaging
- Use common input and output for data
- Minimize complexity
    - Introduce dependencies only when benefits outweigh disadvantages
- Employ modern software development practices

Benefits of ImageJ2

- What Will ImageJ 2.0 Do for Me?
  - Work with existing plugins and macros
  - Work with new plugins and scripts
  - Handle larger, more complex datasets
  - Multidimensional visualization tools
  - Easier to link with other software
  - Easier plugin management

ImageJ2 + OME

- OMERO is powerful server-side imaging software
- ImageJ is powerful client-side imaging software
- Both paradigms are valuable to scientists
- Many operations are common to client & server
  - E.g.: Bio-Formats is used for both
- Goal is to identify more areas for code sharing:
  - Big images (e.g., tiling with mipmaps)
  - Image rendering and thumbnails
  - ROIs, Many others
  - Joint Hackathons

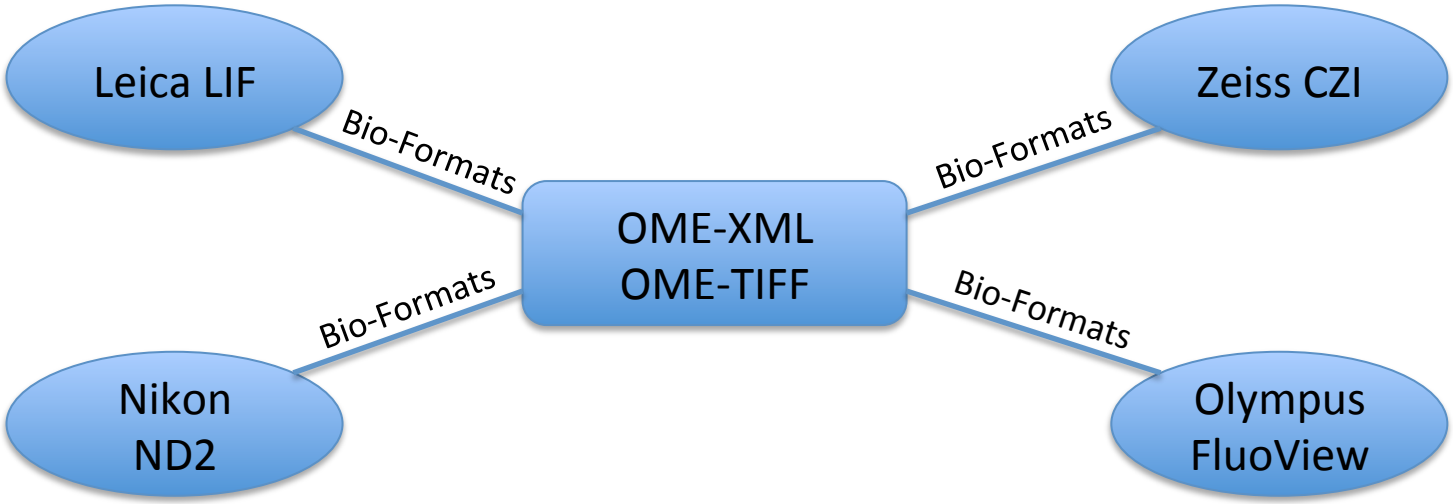
What is ImageJ2?

- ✓ A framework for image processing routines

What is ImageJ2?

✓ A framework for image processing routines

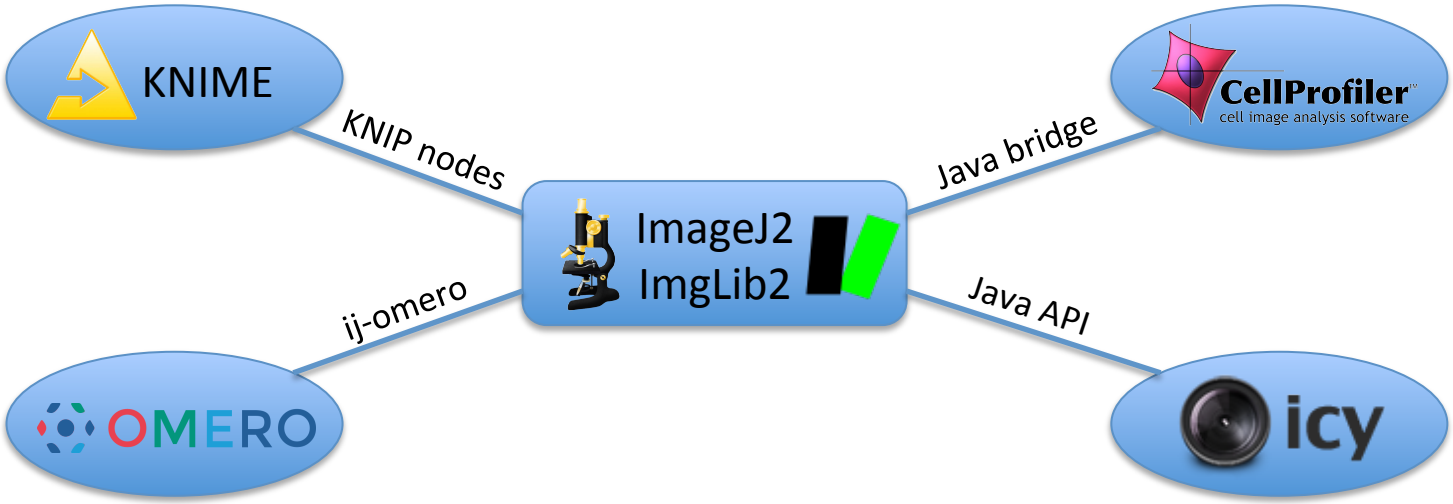
- Bio-Formats and OME-XML provide an interchange format for microscopy data



What is ImageJ2?

✓ A framework for image processing routines

- ImageJ2 and ImgLib2 provide a platform for sharing image processing routines



ImgLib2

- "Write once, run anywhere"

Pixel types

1-bit binary
8-bit signed integer
8-bit unsigned integer
12-bit unsigned integer
16-bit signed integer
16-bit unsigned integer
32-bit signed integer
32-bit unsigned integer
64-bit signed integer
32-bit single precision floating point
64-bit double precision floating point
Complex numbers
Arbitrary precision arithmetic
Even non-numeric types
Fully extensible

Data source

Files on disk
Remote URLs
Local or remote database
(OMERO, CATMAID, etc.)
Fully extensible

Internal representation

Array: Single primitive array
Planar: One primitive array per plane
Cell: Primitive arrays as N-D blocks
SCIFIO: Cells filled on demand from source
Write modified cells to disk cache
Fully extensible

Much more

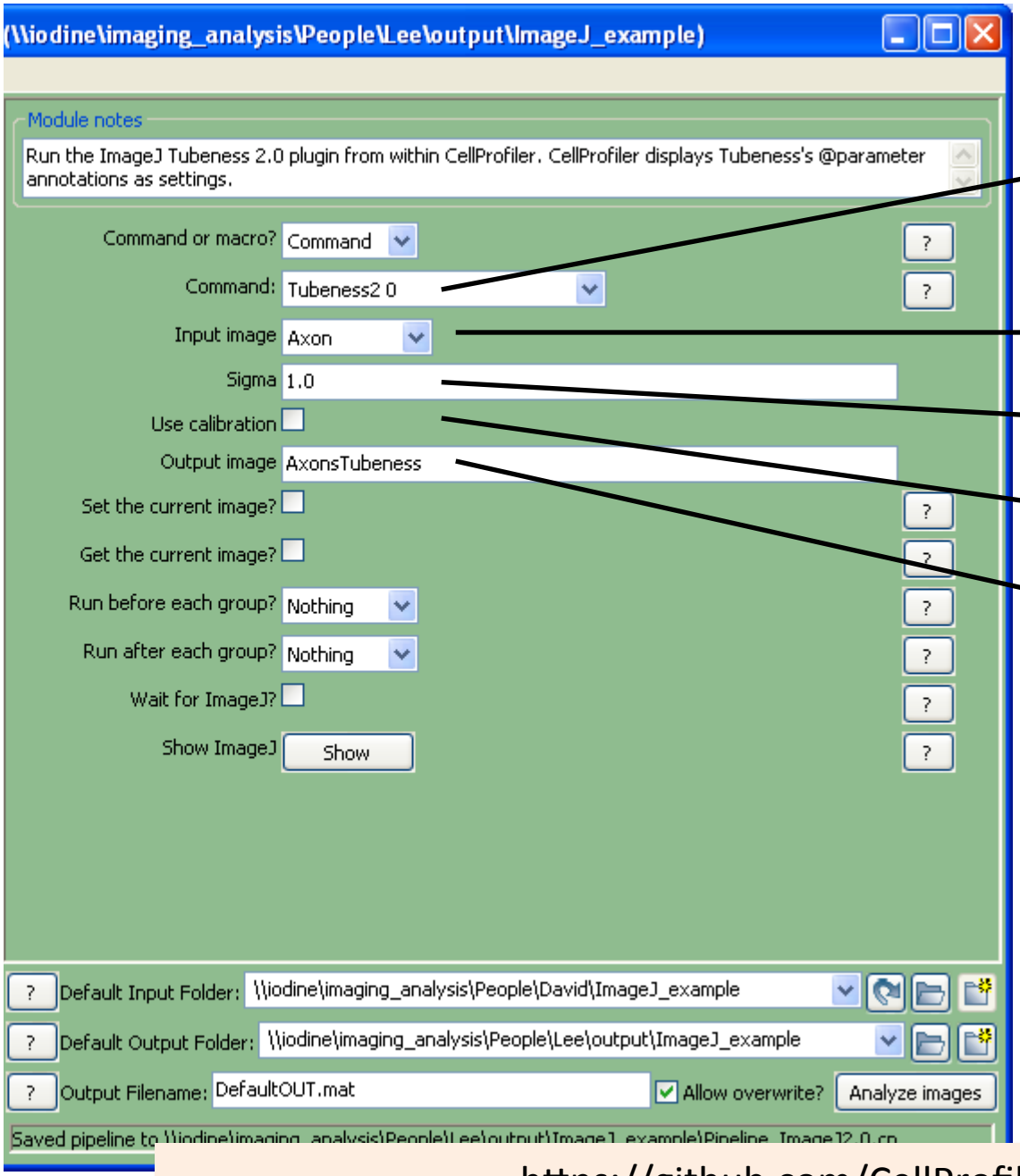
Generative functions
Out of bounds strategies
Compose views on images
Recursive images
Sparse images
OPS: image-wise operations
Image data projections
Resampling & transformation
Transparent type conversion
Metadata attachments
Interface-driven design
Highly compile-time safe
Performance optimizations
Exploit Java's JIT compiler

https://github.com/imagej/imglib

ImageJ2 Commands

- “Write once, run anywhere”
- Operations with typed inputs and outputs
- Maps well to other paradigms
  - KNIME nodes
  - CellProfiler modules
  - OMERO scripts

https://github.com/imagej/imagej



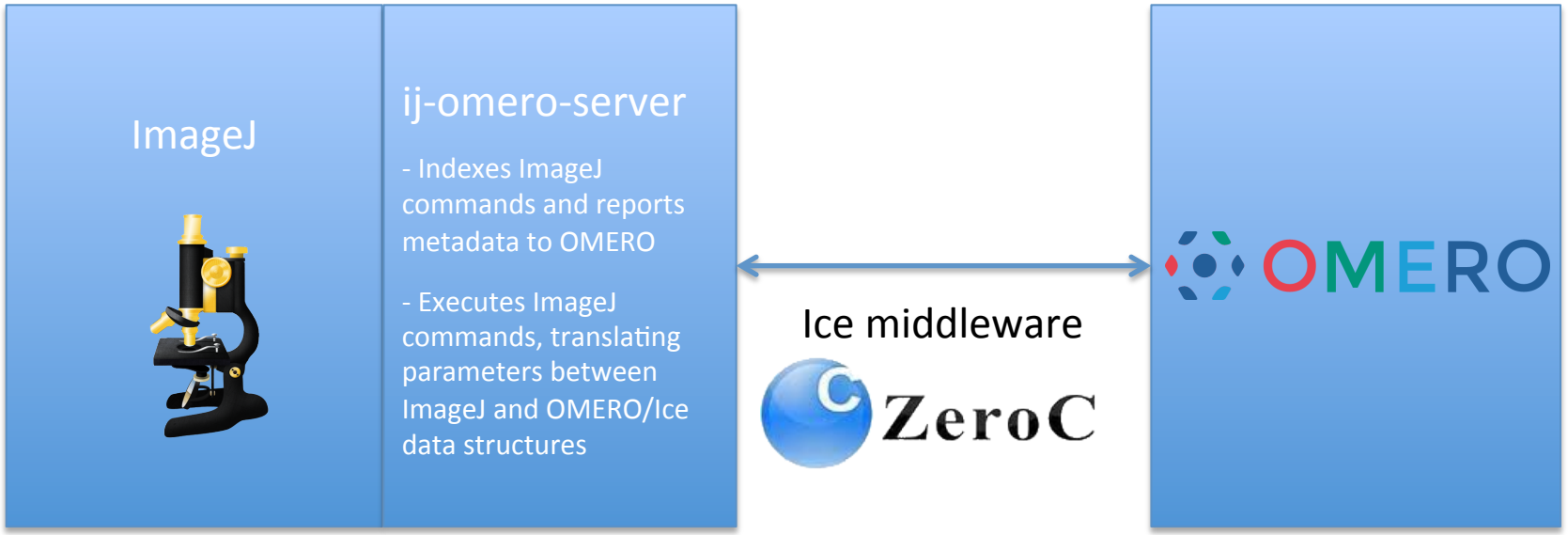

```
@Plugin(type=Command.class)
public class Tubeness2_0
  extends ContextCommand
{

  @Parameter(label="Input image")
  private Dataset original;

  @Parameter(label="Sigma")
  private double sigma = 1.0;

  @Parameter(label="Use calibration")
  private boolean useCalibration;

  @Parameter(label="Output image",
    ioType=ItemIO.OUTPUT)
  private Dataset result;
```

CellProfiler™
cell image analysis software

https://github.com/CellProfiler/CellProfiler

ImageJ on the server side

- In progress: execute ImageJ commands as OMERO scripts
- Drop ImageJ onto the server, gain access to its commands
- All headless ImageJ2 commands + many ImageJ1 plugins
- Working prototype later this summer



https://github.com/imagej/imagej-omero

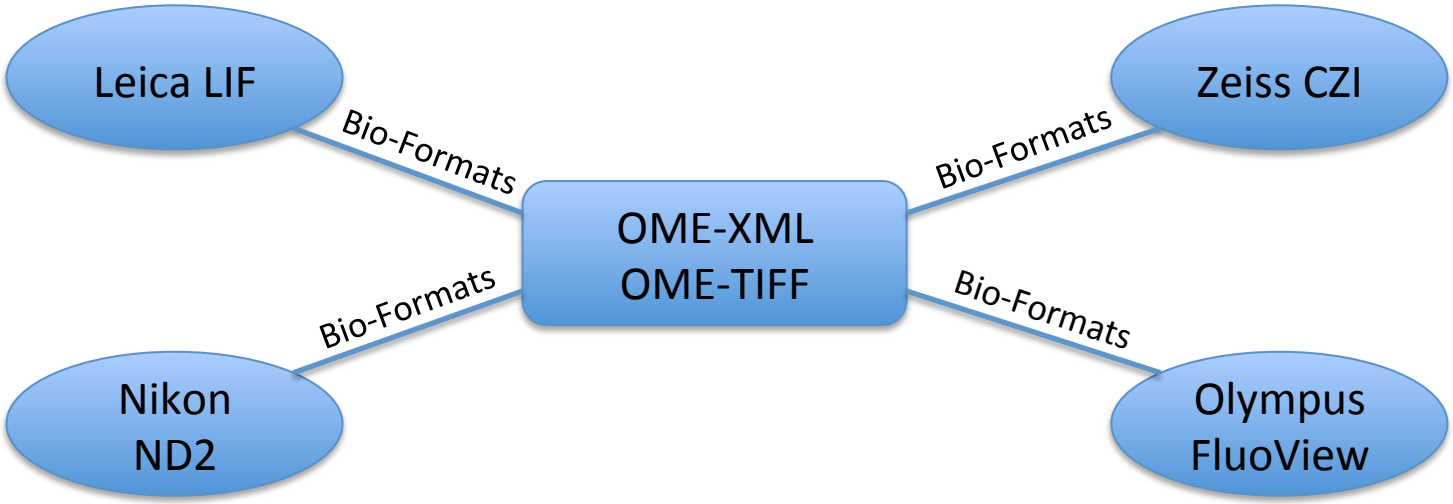
http://trac.openmicroscopy.org.uk/ome/ticket/918

What is SCIFIO?

- ✓ A framework for scientific image I/O

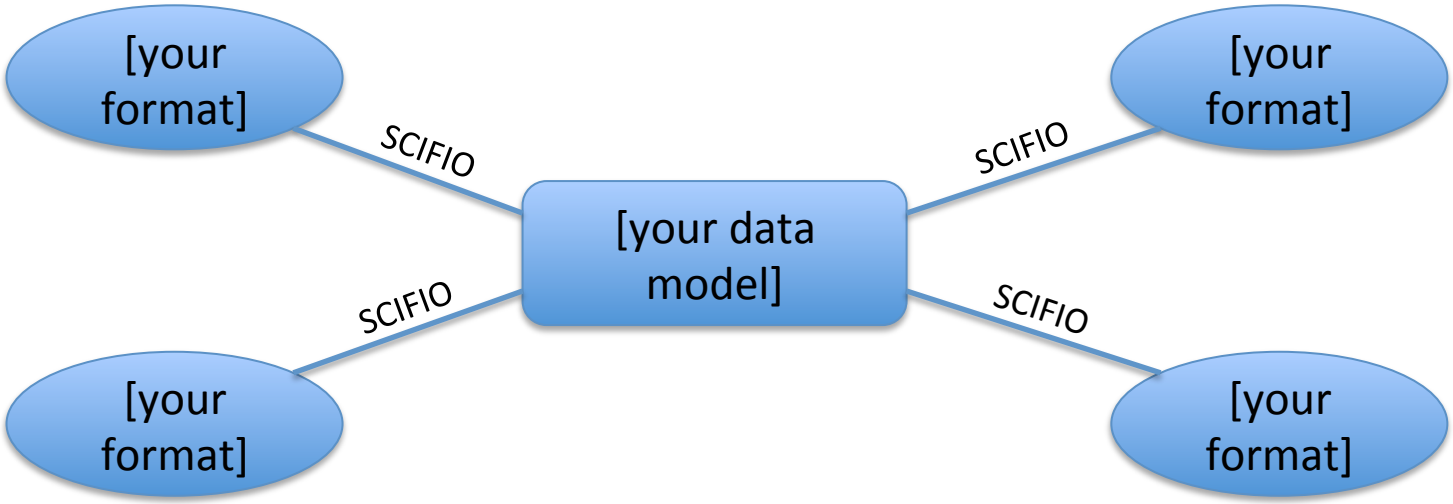
What is SCIFIO?

- ✓ A framework for scientific image I/O
- Bio-Formats and OME-XML provide an interchange format for microscopy data



What is SCIFIO?

- ✓ A framework for scientific image I/O
- SCIFIO provides an exchange framework for scientific image data



SCIFIO

- Core is BSD licensed
- Supports 26 open image formats so far
- In use as image I/O provider for ImageJ2
- Backwards compatible with Bio-Formats 4.x
- For more about SCIFIO, see the poster

https://github.com/scifio/scifio

SciJava Software Stack

- ImageJ2's image I/O: SCIFIO
- ImageJ2's data model: ImgLib2
- ImageJ2's plugin framework: SciJava Common

https://github.com/scijava/scijava-common

Fiji Is Just ImageJ

- Distribution of ImageJ with useful plugins
- Focuses on life sciences analysis problems
- Ships the latest Bio-Formats 4.4.x release
- Also ships with ImageJ2 including SCIFIO
- BF 4.4 & SCIFIO do not conflict with each other
- IJ1 uses Bio-Formats; IJ2 uses SCIFIO
- Either can be used from scripts, macros, etc.

https://github.com/fiji/fiji

ImageJ Hackathon-Madison