

Oxford Update

Hierarchical Data, Tagging and Search

Douglas Russell

douglas.russell@bioch.ox.ac.uk

June, 2013



DEPARTMENT OF BIOCHEMISTRY
UNIVERSITY OF OXFORD
A decorative graphic consisting of two wavy lines, one red and one blue, crossing each other.



wellcome trust
Strategic Award

Micron
OXFORD

Table of contents

Update

- Processing Services
- Tagging Prototype
- C++ Gateway

HierarchicalData

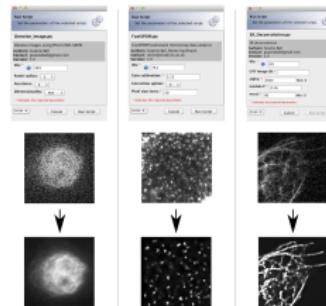
- Filesystem Hierarchy
- Important Information
- Tag Navigation
- Nested Common Search Criteria
- In Practice

Update

- ▶ Installation of large scale OMERO in production
- ▶ Proliferation and promotion of use in Davis lab and beyond
- ▶ Feedback to Dundee
- ▶ Processing Services
- ▶ Tagging Prototype
- ▶ C++ Gateway

Processing Services

- ▶ Denoising
- ▶ Deconvolution
- ▶ Localisation Microscopy
- ▶ Matlab Analysis



Tagging Prototype

The screenshot shows a Windows application window titled "MainWindow". On the left is a tree view of project files:

- 1 testper... 702
- dtest1 1054
- 2 TestProj1 518
- TestD... 703
- dtest1 1054
- 3 proj2 603
- 4 DPWRPr... 1152
- Course 1301

On the right is a table with columns: Filename, Extension, R3D, Hello, TagSet1, D3D, Tag2, and dpwr-test1. The table contains four rows of data:

		Filename	Extension	R3D	Hello	TagSet1	D3D	Tag2	dpwr-test1
1	2	leme070710_grk6cgv3_sqdGFP01_05_R3D.dv		<input checked="" type="checkbox"/> R3D	<input checked="" type="checkbox"/> Hello	<input checked="" type="checkbox"/> TagSet1	<input type="checkbox"/> D3D	<input type="checkbox"/> Tag2	<input type="checkbox"/> dpwr-t...
2	3	leme020710_GLScy3_sqdGFP01_2_R3D_D3D.dv		<input checked="" type="checkbox"/> R3D	<input type="checkbox"/> Hello	<input type="checkbox"/> TagSet1	<input checked="" type="checkbox"/> D3D	<input type="checkbox"/> Tag2	<input type="checkbox"/> dpwr-t...
3	4	leme020710_GLScy3_sqdGFP01_2_R3D.dv		<input checked="" type="checkbox"/> R3D	<input type="checkbox"/> Hello	<input type="checkbox"/> TagSet1	<input type="checkbox"/> D3D	<input checked="" type="checkbox"/> Tag2	<input type="checkbox"/> dpwr-t...
		t-SI-Stack_2.dv		<input type="checkbox"/> R3D	<input type="checkbox"/> Hello	<input type="checkbox"/> TagSet1	<input type="checkbox"/> D3D	<input type="checkbox"/> Tag2	<input checked="" type="checkbox"/> dpwr-t...

Figure: Prototype Demonstrating Auto-Tagging

C++ Gateway

- ▶ Friendlier C++ interface wrapping for OMERO Clients
- ▶ Based on and mirroring the Python Gateway

Hierarchical Data, Tagging and Search

- ▶ Very common request for OMERO to have more levels than Project and Dataset
- ▶ Driven by very reasonable need to organise data
- ▶ Why and how are tags a solution to this?
- ▶ Can tags be an improvement on this?

A screenshot of a terminal window titled '-/tree-structures:bash'. The window displays a hierarchical file structure. The root directory is 'tree-structures'. Inside 'tree-structures' are four subdirectories: '2010', '2011', '2012', and '2013'. Each year directory contains a 'Collaborations' folder, which in turn contains 'John', 'Lisa', and 'Pete'. In '2013', there is also a 'February' folder containing a 'DeltaVision' folder with 'Denoised' and 'File1_Info1_Info2_Info3_Denoised.dv' files. Another 'DeltaVision' folder is present in '2013' under 'January'. The terminal shows the command 'tree -C date-first/Data/' was run. At the bottom, the prompt 'dpwrussell@dpwrwk:~/tree-structures\$' is visible, along with several tabs at the bottom of the window.

```
dpwrussell@dpwrwk:~/tree-structures$ tree -C date-first/Data/
date-first/Data/
├── 2010
│   └── Collaborations
│       ├── John
│       ├── Lisa
│       └── Pete
├── 2011
│   └── Collaborations
│       ├── John
│       └── Pete
├── 2012
│   └── Collaborations
│       ├── John
│       └── Pete
└── 2013
    ├── Collaborations
    │   ├── Lisa
    │   ├── Pete
    │   └── DeltaVision
    │       ├── Denoised
    │       │   └── File1_Info1_Info2_Info3_Denoised.dv
    │       └── File1_Info1_Info2_Info3.dv
    └── February
        └── DeltaVision
            ├── Denoised
            │   └── File1234_Info1_Info_2_Info10.Denoised.dv
            └── File1234_Info1_Info_2_Info10.dv
    └── January
        └── DeltaVision

24 directories, 4 files

dpwrussell@dpwrwk:~/tree-structures$
```

Figure: Filesystem Hierarchy

A screenshot of a terminal window titled '-/tree-structures:bash'. The command run is 'dpwRussell@dpwrwk:~/tree-structures\$ tree -C collab-first/Data/'. The output shows a hierarchical file structure:

```
collab-first/Data/
├── 2010
├── 2011
├── 2012
└── 2013
    ├── February
    │   └── DeltaVision
    │       └── Denoised
    │           └── File1234_Info1_Info_2_Info10_Denoised.dv
    ├── January
    └── Collaborations
        ├── John
        │   ├── 2010
        │   ├── 2011
        │   ├── 2012
        └── Lisa
            ├── 2010
            ├── 2011
            └── 2013
        └── Pete
            ├── 2010
            ├── 2011
            ├── 2012
            └── 2013
                └── DeltaVision
                    └── Denoised
                        └── File1_Info1_Info2_Info3_Denoised.dv
                            └── File1_Info1_Info2_Info3.dv
```

23 directories, 4 files

The terminal window has tabs at the bottom: '/Documents/OME/Presentations/Paris2013 : vi' and '/Documents/OME/Presentations/Paris2013 : okular'.

Figure: Alternate Filesystem Hierarchy

The Important Information

- ▶ E.g. '**File1_Info1_Info2_Info3_Denoised.dv**'
- ▶ From directory structure: **2013, Collaborations, Pete, DeltaVision, Denoised**
- ▶ From filename: **Info1, Info2, Info3**

Using Tags Instead

- ▶ Assign tags for all important information
- ▶ The only difference in navigating tagged data as opposed to a filesystem is greater choice of path to take



Figure: All Tags, Searchable



Figure: Selecting Starting Point

OMERO Tag Search

Results

2013

Tags

Search

Collaborations DeltaVision Denoised February Info1 Info2 Info3 Info10

January Lisa Pete

Click

When number of results is below some threshold...

Tags intersecting with selected Tags diminishes...

Image

File1_Info1_Info2_Info3_Denoised.dv

File1_Info1_Info2_Info3.dv

File1234_Info1_Info2_Info10_Denoised.dv

File1234_Info1_Info2_Info10.dv

The screenshot shows the OMERO Tag Search interface. On the left, under 'Results', there is a single item labeled '2013'. On the right, under 'Tags', there is a search bar and a grid of tags: Collaborations, DeltaVision, Denoised, February, Info1, Info2, Info3, Info10, January, Lisa, and Pete. A 'Click' button is located below the tags. Below the tags, a message says 'Tags intersecting with selected Tags diminishes...'. On the far left, a list of files is shown, starting with 'Image' and including 'File1_Info1_Info2_Info3_Denoised.dv', 'File1_Info1_Info2_Info3.dv', 'File1234_Info1_Info2_Info10_Denoised.dv', and 'File1234_Info1_Info2_Info10.dv'. A scroll bar is visible on the right side of this list.

Figure: Initial Tag Intersection and Results

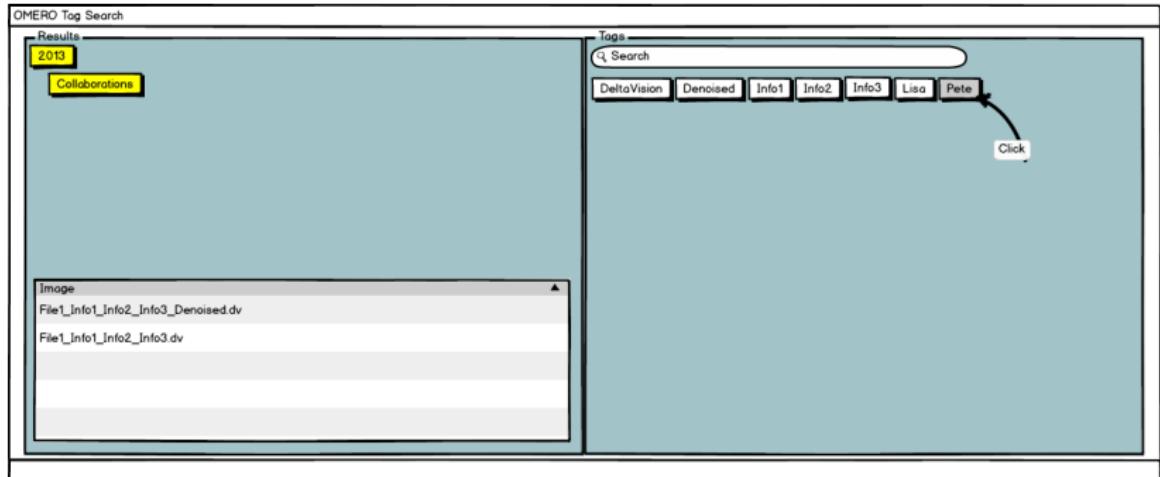


Figure: Reduced Tag Intersection and Results



Figure: Further Reduced Tag Intersection and Results



Figure: Further Reduced Tag Intersection and Results



Figure: Exact Replica of Filesystem Hierarchy

Nested Common Search Criteria

- ▶ What if the criteria of interest is not at the top level of the filesystem tree?
- ▶ Difficult to search in a filesystem.
- ▶ What about with tags?
- ▶ E.g. All Denoised data from 2013

A screenshot of a terminal window titled '-/tree-structures:bash'. The window displays a hierarchical file structure. The root directory is '2010', which contains 'Collaborations' and three sub-directories for 'John', 'Lisa', and 'Pete'. Below '2010' are '2011', '2012', and '2013'. '2013' contains 'Collaborations' and two sub-directories: 'February' and 'January'. 'February' contains 'DeltaVision' and 'Denoised' sub-directories, with files 'File1_Info1_Info2_Info3_Denoised.dv' and 'File1_Info1_Info2_Info3.dv'. 'January' contains 'DeltaVision' sub-directory. The command 'tree -C date-first/Data/' was run to generate this output. The terminal also shows the prompt 'dpwrussell@dpwrwk:~/tree-structures\$'.

```
dpwrussell@dpwrwk:~/tree-structures$ tree -C date-first/Data/
date-first/Data/
├── 2010
│   └── Collaborations
│       ├── John
│       ├── Lisa
│       └── Pete
├── 2011
└── 2012
└── 2013
    ├── Collaborations
    ├── February
    │   └── DeltaVision
    │       └── Denoised
    │           ├── File1_Info1_Info2_Info3_Denoised.dv
    │           └── File1_Info1_Info2_Info3.dv
    └── January
        └── DeltaVision
24 directories, 4 files
dpwrussell@dpwrwk:~/tree-structures$
```

Figure: Filesystem Hierarchy

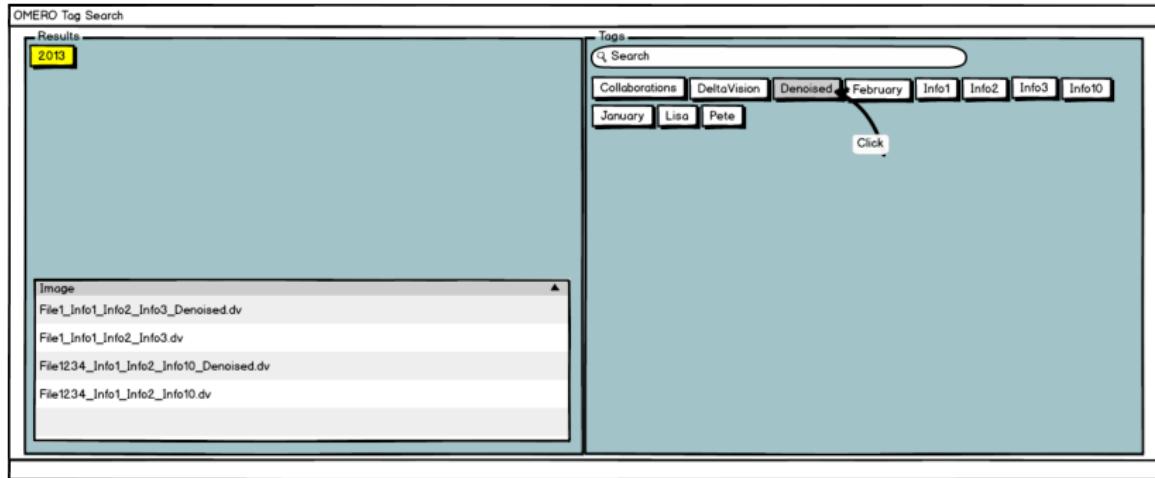


Figure: '2013' Starting Point

OMERO Tag Search

Results

2013

Denoised

Tags

Search

Collaborations DeltaVision February Info1 Info2 Info3 Info10 January

Pete

Image

File1_Info1_Info2_Info3_Denoised.dv

File1234_Info1_Info2_Info10_Denoised.dv

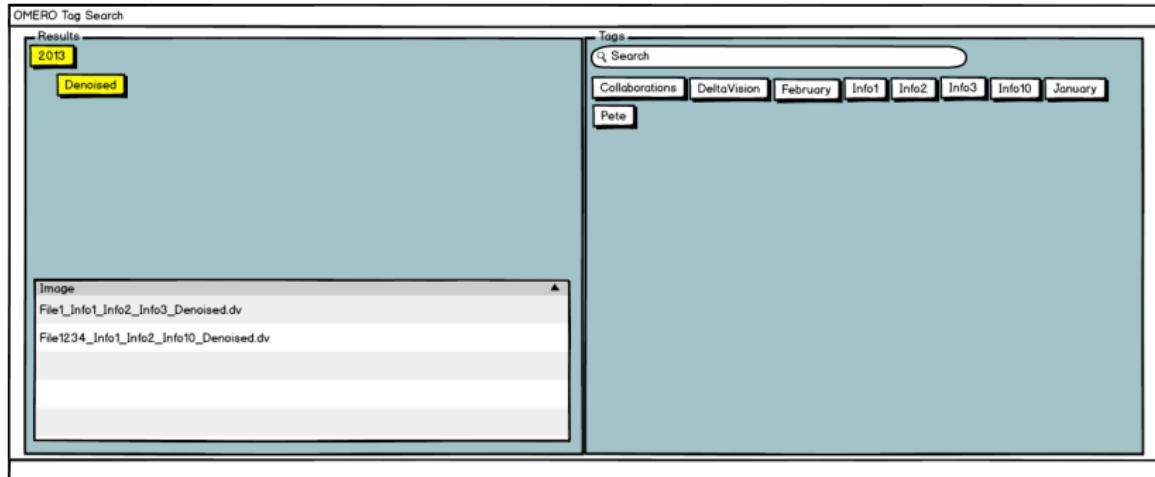
The screenshot shows the OMERO Tag Search interface. In the top left, there's a search bar with the placeholder 'Search'. Below it is a grid of tags: Collaborations, DeltaVision, February, Info1, Info2, Info3, Info10, and January. To the right of the grid is a button labeled 'Pete'. On the far left, under 'Results', are two yellow-highlighted tags: '2013' and 'Denoised'. A large central panel displays a list of files: 'File1_Info1_Info2_Info3_Denoised.dv' and 'File1234_Info1_Info2_Info10_Denoised.dv'. Below this list is a small 'Image' thumbnail.

Figure: Results matching '2013' and 'Denoised'

In Practice

- ▶ Mockups based on design discussion with Will, Jean-Marie, Gus and others
- ▶ For implementation in OMERO Clients
- ▶ Search needs some work