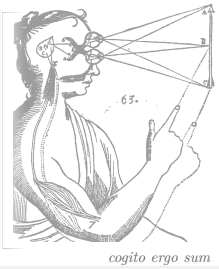
CCN Center for Cognitive Neuroscience at Dartmouth



cbbs center for behavioral brain sciences

DataLad – decentralized data distribution for consumption and sharing of scientific datasets

Yaroslav O. Halchenko[1], Michael Hanke[2]

[1] Dartmouth College, Hanover, NH, USA
[2]Otto-von-Guericke University, Magdeburg, Germany

NIH, Bethesda, Aug 2017

http://datalad.org

http://www.pymvpa.org

http://NeuroDebian.net

http://duecredit.org

Acknowledgments

centerforopenneuroscience.org/whoweare#michael_hanke_

Center for Open Neuroscience

Centroids

Collaborators

- Michael Hanke
- Nikolaas N. Oosterhof
- Matthew Brett
- Joey Hess
- Benjamin Poldrack

Emeritus

Collaborating projects

Partners

Michael Hanke

University of Magdeburg, Germany

Formerly a visiting post-doctoral research at Dr.Haxby's lab, now a J.-Prof., one of the first Psychoinformaticians, official Debian developer, member of INCF neuroimaging task force -- he is an old-time collaborator and a lead of PyMVPA, NeuroDebian, DataLad and other projects.

Joey Hess

Independent Guru

Joey's own introduction *"I'm Joey Hess and I write programs"* conceals his paramount role in establishing the core of the Debian distribution (debhelper, debian-installer, debconf, pristine-tar, etc.) and his work on variety of other software projects, such as git-annex which we rely upon in the DataLad project.

Benjamin Poldrack

University of Magdeburg, Germany

Works on the DataLad project.

Interns (Dartmouth)

Debanjum

Gergana

Acknowledgments

Houston, we've got a problem...

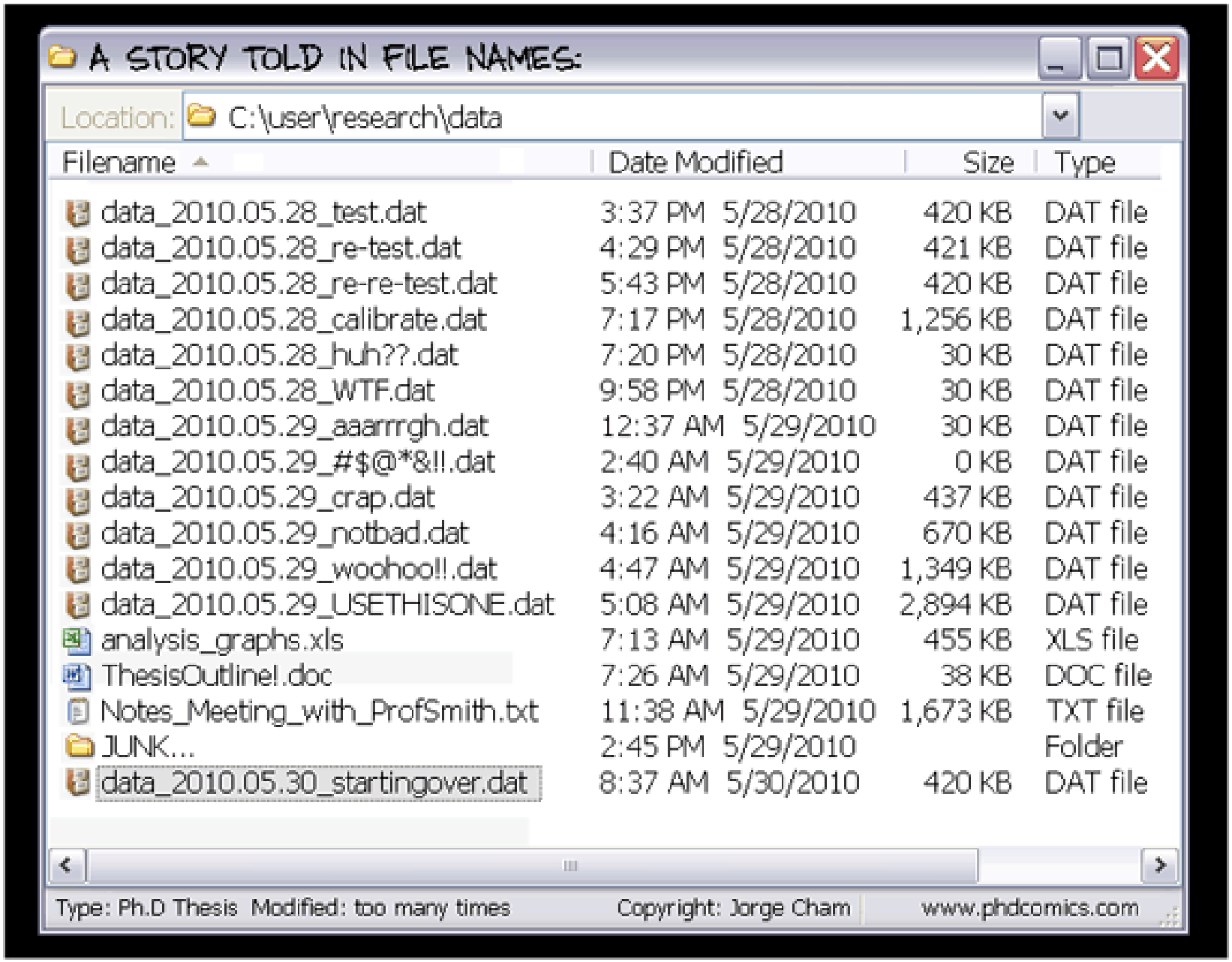
Data is a 2nd-class citizen within software platforms

Why?

- tarballs are **inefficient** distribution format
- **absent versioning** of data

derived and/or curated data does change!

Why?


A STORY TOLD IN FILE NAMES:
Location: C:\user\research\data
Filename | Date Modified | Size | Type
data_2010.05.28_test.dat 3:37 PM 5/28/2010 420 KB DAT file
data_2010.05.28_re-test.dat 4:29 PM 5/28/2010 421 KB DAT file
data_2010.05.28_re-re-test.dat 5:43 PM 5/28/2010 420 KB DAT file
data_2010.05.28_calibrate.dat 7:17 PM 5/28/2010 1,256 KB DAT file
data_2010.05.28_huh??.dat 7:20 PM 5/28/2010 30 KB DAT file
data_2010.05.28_WTF.dat 9:58 PM 5/28/2010 30 KB DAT file
data_2010.05.29_aaarrrgh.dat 12:37 AM 5/29/2010 30 KB DAT file
data_2010.05.29_#$@*&!!.dat 2:40 AM 5/29/2010 0 KB DAT file
data_2010.05.29_crap.dat 3:22 AM 5/29/2010 437 KB DAT file
data_2010.05.29_notbad.dat 4:16 AM 5/29/2010 670 KB DAT file
data_2010.05.29_woohoo!!.dat 4:47 AM 5/29/2010 1,349 KB DAT file
data_2010.05.29_USETHISONE.dat 5:08 AM 5/29/2010 2,894 KB DAT file
analysis_graphs.xls 7:13 AM 5/29/2010 455 KB XLS file
ThesisOutline!.doc 7:26 AM 5/29/2010 38 KB DOC file
Notes_Meeting_with_ProfSmith.txt 11:38 AM 5/29/2010 1,673 KB TXT file
JUNK... 2:45 PM 5/29/2010 Folder
data_2010.05.30_startingover.dat 8:37 AM 5/30/2010 420 KB DAT file
Type: Ph.D Thesis Modified: too many times
Copyright: Jorge Cham
www.phdcomics.com

ge!

Why?

- tarballs are **inefficient** distribution format
- **absent versioning** of data

derived and/or curated data does change!

- code version control systems are **inadequate** for data

duplication, monolithic storage, etc.

- **absent generic data distributions**

no efficient ways to install and upgrade

- **cacophony** of authorization schemes, interfaces, protocols
- **absent data testing**

*data can and **does** have bugs (see e.g. Halchenko, 2012; Rohlfing, 2013)*

- **difficulty to share** new or derivative data

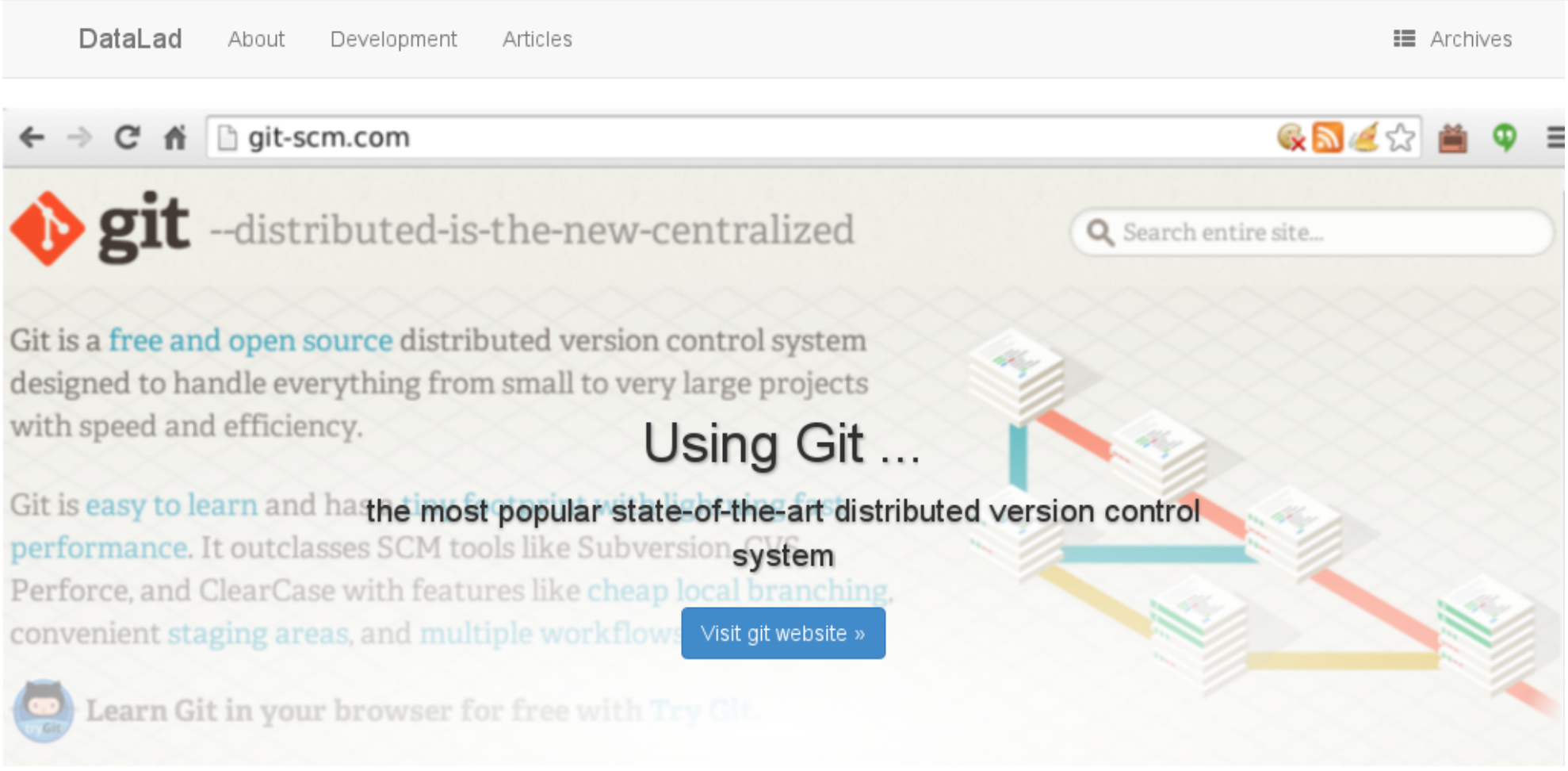
shareable? some is not! where to host? how to "link" back?

DataLad's goal

DataLad's goal

Managing data should be as easy as managing code and software

Welcome datalad.org



DataLad aims to provide access to scientific data available from various sources (e.g. lab or consortium web-sites such as Human connectome; data sharing portals such as OpenFMRI and CRCNS) through a single convenient interface and integrated with your software package managers (such as APT in Debian). Although initially targeting neuroimaging and neuroscience data in general, it will not be limited by the domain and we would welcome a wide range of contributions.

DataLad demo1 - from search to get

DataLad demo1 - from search to get

```
File  Edit  View  Search  Terminal  Help
2 14868.....................................:Fri 21 Oct 2016 09:57:10 PM EDT:.
(git)hopa:~/datalad[master]
$> datalad search Haxby
labs/haxby
labs/haxby/raiders
openfmri/ds000105
2 14871.....................................:Fri 21 Oct 2016 09:57:54 PM EDT:.
(git)hopa:~/datalad[master]
$>
```

DataLad demo1 - from search to get

```
File  Edit  View  Search  Terminal  Help
2 14868.....................................:Fri 21 Oct 2016 09:57:10 PM EDT:.
(git)hopa:~/datalad[master]
$> datalad search Haxby
labs/haxby
labs/haxby/raiders
openfmri/ds000105
2 14871.....................................:Fri 21 Oct 2016 09:57:54 PM EDT:.
(git)hopa:~/datalad[master]
$> datalad search Haxby | xargs datalad install
3 installed items are available at
<Dataset path=/home/yoh/datalad/labs/haxby>
<Dataset path=/home/yoh/datalad/labs/haxby/raiders>
<Dataset path=/home/yoh/datalad/openfmri/ds000105>
2 14872.....................................:Fri 21 Oct 2016 09:58:14 PM EDT:.
(git)hopa:~/datalad[master]
$>
```

DataLad demo1 - from search to get

```
File  Edit  View  Search  Terminal  Help
(git)hopa:~/datalad[master]
$> datalad search Haxby
labs/haxby
labs/haxby/raiders
openfmri/ds000105
2 14871.....................................:Fri 21 Oct 2016 09:57:54 PM EDT:.
(git)hopa:~/datalad[master]
$> datalad search Haxby | xargs datalad install
3 installed items are available at
<Dataset path=/home/yoh/datalad/labs/haxby>
<Dataset path=/home/yoh/datalad/labs/haxby/raiders>
<Dataset path=/home/yoh/datalad/openfmri/ds000105>
2 14872.....................................:Fri 21 Oct 2016 09:58:14 PM EDT:.
(git)hopa:~/datalad[master]
$> cd /home/yoh/datalad/labs/haxby/raiders
README.md                   qa/             sub002/  sub007/  task_key.txt
TODO.md                     scan_key.txt    sub003/  sub008/
aligned/                    scripts/        sub004/  sub009/
dataset_description.json    stimulus/       sub005/  sub010/
masks/                      sub001/         sub006/  sub011/

2 14873.....................................:Fri 21 Oct 2016 09:58:30 PM EDT:.
hopa:~/datalad/labs/haxby/raiders
$>
```

DataLad demo1 - from search to get

```
File  Edit  View  Search  Terminal  Help
2 14880.....................................:Fri 21 Oct 2016 10:00:20 PM EDT:.
hopa:~/datalad/labs/haxby/raiders
$> ls sub001/anatomy/
highres004.PAR@                highres008.PAR@                scout001.PAR@
highres004.REC@                highres008.REC@                scout001.REC@
highres004.nii.gz@             highres008.nii.gz@             scout005.PAR@
highres004_defaced.nii.gz@     highres008_defaced.nii.gz@     scout005.REC@
highres004_defacemask.nii.gz@  highres008_defacemask.nii.gz@
2 14880.....................................:Fri 21 Oct 2016 10:00:29 PM EDT:.
hopa:~/datalad/labs/haxby/raiders
$>
```

DataLad demo1 - from search to get

File Edit View Search Terminal Help

```
2 14880.....................................:Fri 21 Oct 2016 10:00:20 PM EDT:.
hopa:~/datalad/labs/haxby/raiders
$> ls sub001/anatomy/
highres004.PAR@                highres008.PAR@                scout001.PAR@
highres004.REC@                highres008.REC@                scout001.REC@
highres004.nii.gz@             highres008.nii.gz@             scout005.PAR@
highres004_defaced.nii.gz@     highres008_defaced.nii.gz@     scout005.REC@
highres004_defacemask.nii.gz@  highres008_defacemask.nii.gz@
2 14880.....................................:Fri 21 Oct 2016 10:00:29 PM EDT:.
hopa:~/datalad/labs/haxby/raiders
$> datalad get -J4 sub*/anatomy/*nii.gz
2016-10-21 22:00:46,030 [INFO   ] Getting 58 items of dataset <Dataset path=/hom
e/yoh/datalad/labs/haxby/raiders> ...
Total (15 ok, 12 failed out of 58) 23%|  | 57.0M/253M [00:17<01:04, 3.06MB/s]
sub005/anat .. aced.nii.gz:  98%|▉| 7.33M/7.50M [00:08<00:00, 726KB/s]
sub005/anat .. aced.nii.gz:  71%|▋| 5.08M/7.12M [00:06<00:03, 624KB/s]
sub006/anat .. aced.nii.gz:  75%|▊| 4.24M/5.66M [00:04<00:01, 858KB/s]
sub007/anat .. aced.nii.gz:  21%|▏| 1.49M/6.99M [00:00<00:00, 6.18MB/s]
```

DataLad demo1 - from search to get

```
File  Edit  View  Search  Terminal  Help
2 14881.....................................:Fri 21 Oct 2016 10:01:25 PM EDT:.
hopa:~/datalad/labs/haxby/raiders
$> ls sub001/anatomy/
highres004.PAR@                highres008.PAR@                scout001.PAR@
highres004.REC@                highres008.REC@                scout001.REC@
highres004.nii.gz@             highres008.nii.gz@             scout005.PAR@
highres004_defaced.nii.gz@     highres008_defaced.nii.gz@     scout005.REC@
highres004_defacemask.nii.gz@  highres008_defacemask.nii.gz@
2 14882.....................................:Fri 21 Oct 2016 10:01:44 PM EDT:.
hopa:~/datalad/labs/haxby/raiders
$>
```

How: Foundation #1 – Git is

- a **version control system** initially developed to manage Linux project code
- **distributed** - content is available across all copies of the repository while allowing for aggregation of individual differences
- a backbone of **GitHub** and other *social coding* portals
- **very efficient** for managing textual information (code, text, configuration, *etc.*)
- **inefficient** for storing data

How: Foundation #2 – Git-annex

- is **built on top of Git**
- provides **access to data content** from variety of sources: HTTP, FTP, Webdav, bittorent, RSYNC, Amazon S3, *etc.*
- allows for **custom extensions** to get access to offload data: Dropbox, Google Drive, Box.com (will demo later) *etc.*
- features optional Dropbox-like **synchronization** facility via *git-annex assistant*

How: Foundation #2 – Git-annex

- is **built on top of Git**
- provides **access to data content** from variety of sources: HTTP, FTP, Webdav, bittorent, RSYNC, Amazon S3, *etc.*
- allows for **custom extensions** to get access to offload data: Dropbox, Google Drive, Box.com (will demo later) *etc.*
- features optional Dropbox-like **synchronization** facility via *git-annex assistant*

Both Git and git-annex largely work on a single repository level

How: Foundation #2 – Git-annex

- is **built on top of Git**
- provides **access to data content** from variety of sources: HTTP, FTP, Webdav, bittorent, RSYNC, Amazon S3, *etc.*
- allows for **custom extensions** to get access to offload data: Dropbox, Google Drive, Box.com (will demo later) *etc.*
- features optional Dropbox-like **synchronization** facility via *git-annex assistant*

Both Git and git-annex largely work on a single repository level
TBs of scientific data are out there in separate custom portals

How #1+2=#3: DataLad

- comes with **command line and Python** interfaces

How #1+2=#3: DataLad

- comes with **command line and Python** interfaces
- supports both **git and git/annex** repositories

How #1+2=#3: DataLad

- comes with **command line and Python** interfaces
- supports both **git and git/annex** repositories
- manages **multiple repositories** organized into “super-datasets” using standard git sub-modules mechanism

How #1+2=#3: DataLad

- comes with **command line and Python** interfaces
- supports both **git and git/annex** repositories
- manages **multiple repositories** organized into “super-datasets” using standard git sub-modules mechanism
- is **scalable** since data stays with original data providers

How #1+2=#3: DataLad

- comes with **command line and Python** interfaces
- supports both **git and git/annex** repositories
- manages **multiple repositories** organized into “super-datasets” using standard git sub-modules mechanism
- is **scalable** since data stays with original data providers
- **unifies access** to data regardless of its origin (custom portals with authentication, S3, *etc.*) or serialization (*e.g.*, tarballs)

How #1+2=#3: DataLad

- comes with **command line and Python** interfaces
- supports both **git and git/annex** repositories
- manages **multiple repositories** organized into “super-datasets” using standard git sub-modules mechanism
- is **scalable** since data stays with original data providers
- **unifies access** to data regardless of its origin (custom portals with authentication, S3, *etc.*) or serialization (*e.g.*, tarballs)
- aggregates datasets’ **meta-data** and allows for quick **search**

How #1+2=#3: DataLad

- comes with **command line and Python** interfaces
- supports both **git and git/annex** repositories
- manages **multiple repositories** organized into “super-datasets” using standard git sub-modules mechanism
- is **scalable** since data stays with original data providers
- **unifies access** to data regardless of its origin (custom portals with authentication, S3, *etc.*) or serialization (*e.g.*, tarballs)
- aggregates datasets’ **meta-data** and allows for quick **search**
- can **publish** original or derived datasets publicly (a web server, WiP: github) or for internal use (e.g. via ssh), while possibly keeping data available from elsewhere

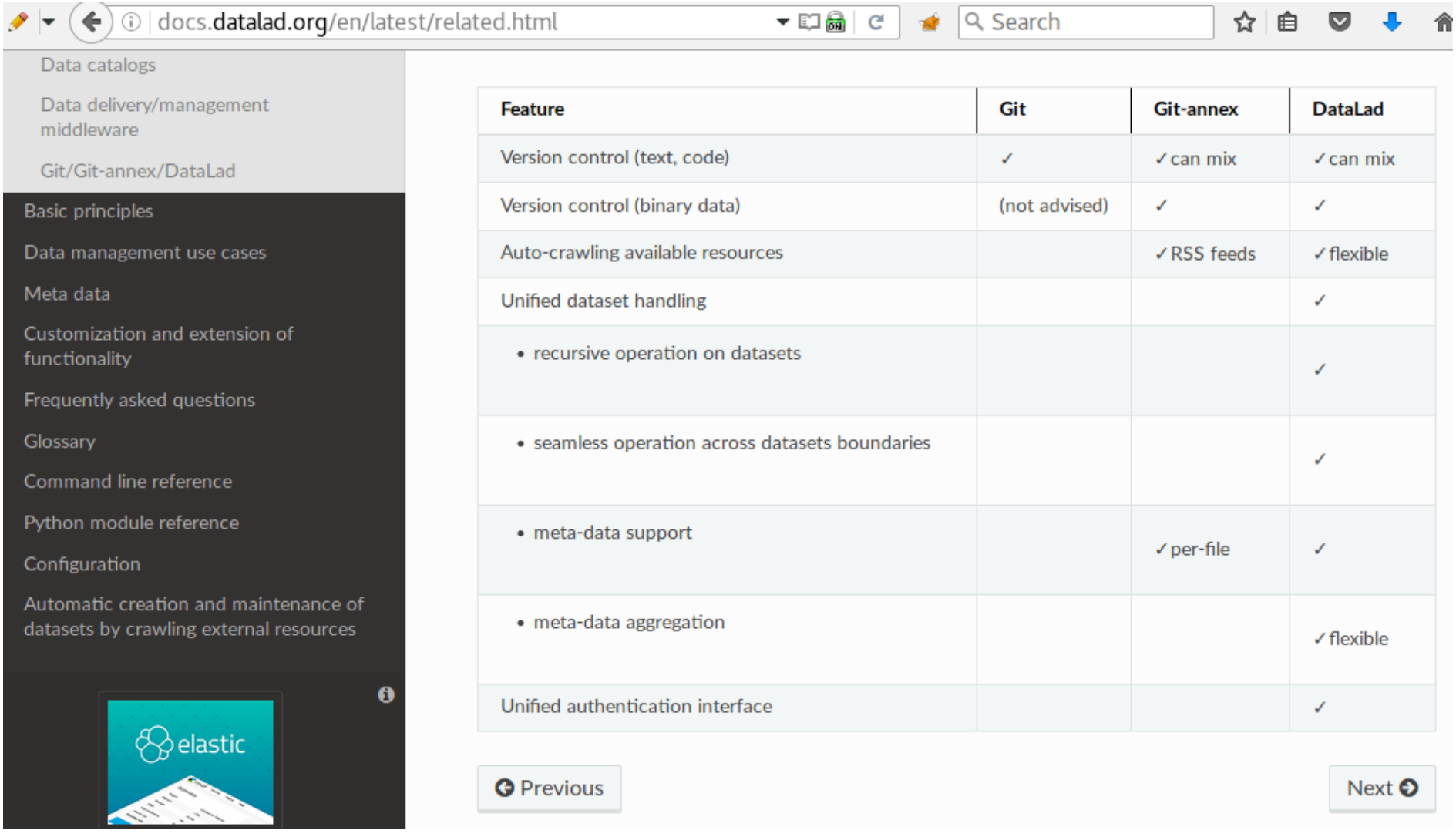
How #1+2=#3: DataLad

- comes with **command line and Python** interfaces
- supports both **git and git/annex** repositories
- manages **multiple repositories** organized into “super-datasets” using standard git sub-modules mechanism
- is **scalable** since data stays with original data providers
- **unifies access** to data regardless of its origin (custom portals with authentication, S3, *etc.*) or serialization (*e.g.*, tarballs)
- aggregates datasets’ **meta-data** and allows for quick **search**
- can **publish** original or derived datasets publicly (a web server, WiP: github) or for internal use (e.g. via ssh), while possibly keeping data available from elsewhere
- can **export** datasets (tarballs, WiP: ISA-TAB)

How #1+2=#3: DataLad

- comes with **command line and Python** interfaces
- supports both **git and git/annex** repositories
- manages **multiple repositories** organized into “super-datasets” using standard git sub-modules mechanism
- is **scalable** since data stays with original data providers
- **unifies access** to data regardless of its origin (custom portals with authentication, S3, *etc.*) or serialization (*e.g.*, tarballs)
- aggregates datasets’ **meta-data** and allows for quick **search**
- can **publish** original or derived datasets publicly (a web server, WiP: github) or for internal use (e.g. via ssh), while possibly keeping data available from elsewhere
- can **export** datasets (tarballs, WiP: ISA-TAB)
- can **crawl** external online data sources, and update git/annex repositories upon changes

Git vs Git-annex vs DataLad



Feature	Git	Git-annex	DataLad
Version control (text, code)	✓	✓can mix	✓can mix
Version control (binary data)	(not advised)	✓	✓
Auto-crawling available resources		✓RSS feeds	✓flexible
Unified dataset handling			✓
• recursive operation on datasets			✓
• seamless operation across datasets boundaries			✓
• meta-data support		✓per-file	✓
• meta-data aggregation			✓flexible
Unified authentication interface			✓

OpenfMRI ds000001: website

https://openfmri.org/dataset/ds000001/

Revision: 2.0.0 Date Set: May 24, 2016, 7:26 p.m.

Notes:

- Converted to BIDS standard.

Data Associated with Revision:

- Raw data on AWS

Revision: 1.1.0 Date Set: Feb. 18, 2016, 8:28 p.m.

Notes:

Updated orientation information in NIFTI headers for better left-right determination.

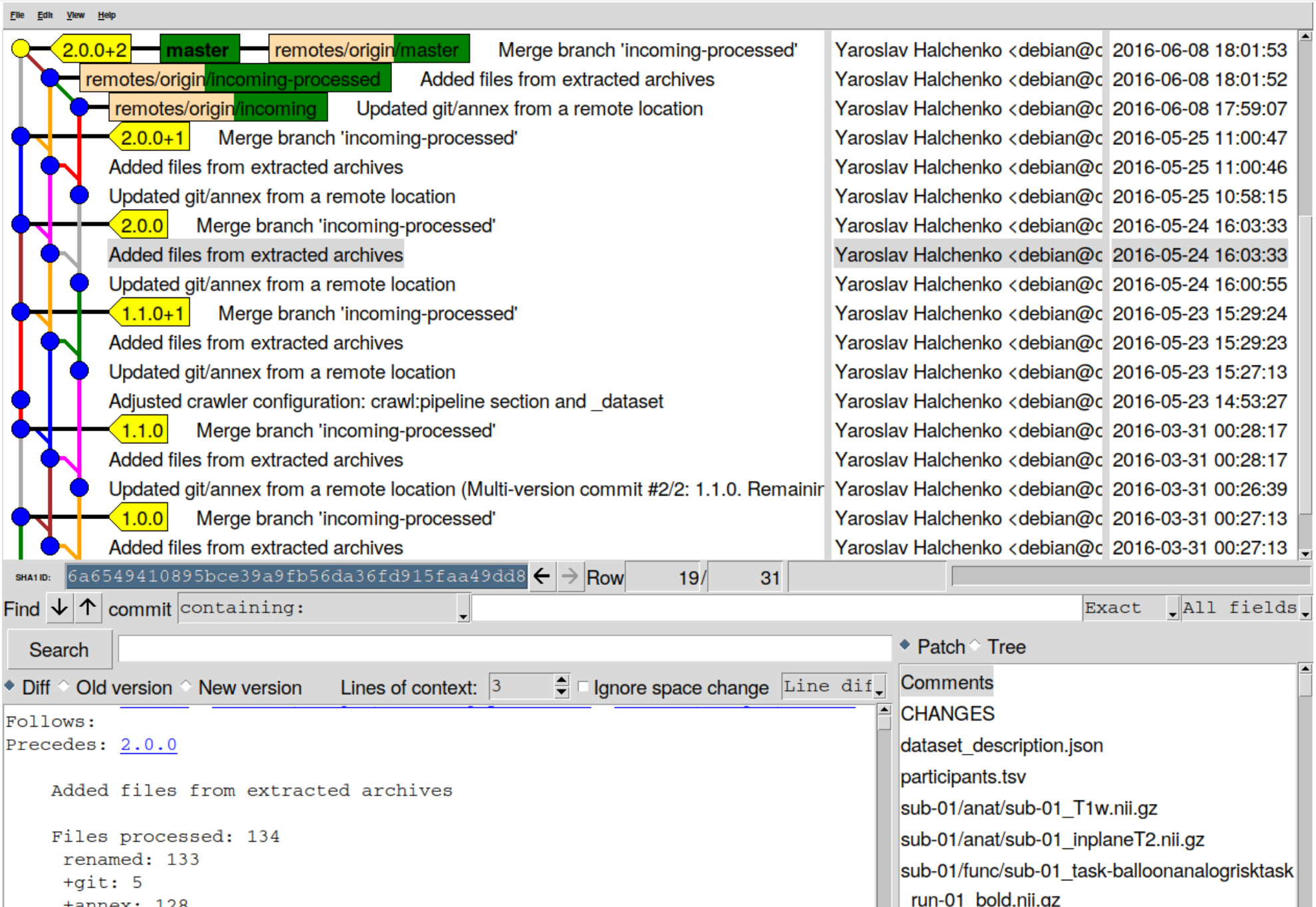
Data Associated with Revision:

- Raw data checksums
- Raw data on AWS

Revision: 1.0.0 Date Set: July 10, 2012, 8:28 p.m.

Data Associated with Revision:

OpenfMRI ds000001: crawled version (gitk)



OpenfMRI: crawled super-dataset

DataLad

To install this dataset in your current directory use

```
datalad install ///openfmri/
```

To install with all subdatasets and all data

```
datalad install -r -g ///openfmri/
```

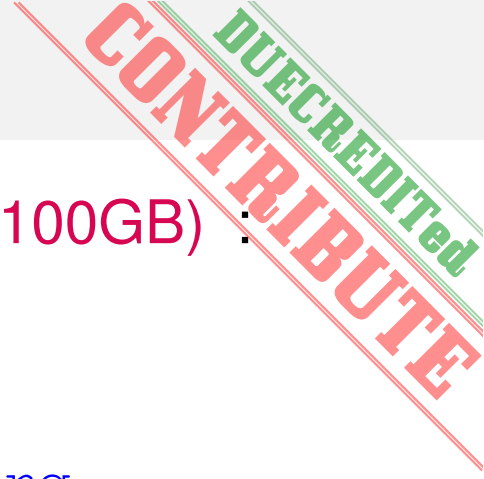
For more information about DataLad and installation instructions visit datalad.org

datasets.datalad.org / openfmri /

Search:

Name	Last Modified	Size	Description
./	2016-10-11 00:27:05	717.8 kB/1.3 TB	OpenfMRI (http://openfmri.org)
../	2016-09-22 15:31:00		
ds000001/ @2.0.1	2016-09-19 19:18:54	2.8 kB/2.4 GB	Balloon Analog Risk Task
ds000002/ @2.0.1	2016-10-07 22:43:10	2.4 kB/2.9 GB	Classification learning
ds000003/ @2.0.1	2016-10-07 22:43:13	2.1 kB/413.5 MB	Rhyme judgment
ds000005/ @2.0.0+3-2-gbed9245	2016-09-07 16:26:20	2.4 kB/1.9 GB	Mixed-gambles task
ds000006/ @2.0.0+2-2-g8b1d65a	2016-09-07 16:26:20	2.4 kB/4.8 GB	Living-nonliving decision with plain or mirror-reversed text

DataLad WiP



Our growing "distribution" (>10TB, hosting locally only 100GB) :

- http://datasets.datalad.org

Covered :

- Neuroimaging: http://openfmri.org, http://crcns.org, *etc.*
- Other neuro-data (from kaggle, INDI, FC etc.)
- Even some music (podcast radio): github.com/datalad/ratholeradio-archive

Coming :

- More neuroimaging data (HCP, XNAT-support, etc)
- Extended meta-data support
- ... *(have interesting data provider? File github issue: github.com/datalad/datalad)* ...
- Integration: NeuroDebian; OSF, Zenodo

Straight from the oven : MRI DICOM → DataLad BIDS

https://github.com/nipy/heudiconv/pull/32

Summary: DataLad ...

- helps to manage and share available and your own data via a simple (command line or Python) interface
- helps with
  - authentication
  - crawling of websites with data resources
  - getting data from archives
  - publishing your new or derived data
- uses pure git/git-annex repositories under – power users can stay in power, and everything is version controlled
- makes meta-data *useful* to normal humans
- is ready for you to start using it, documentation is growing: www.datalad.org

Summary: DataLad ...

- helps to manage and share available and your own data via a simple (command line or Python) interface
- helps with
  - authentication
  - crawling of websites with data resources
  - getting data from archives
  - publishing your new or derived data
- uses pure git/git-annex repositories under – power users can stay in power, and everything is version controlled
- makes meta-data *useful* to normal humans
- is ready for you to start using it, documentation is growing: www.datalad.org

Managing data can be similar to managing code and software


Brain Download:
iz compltes.
ICANHASCHEEZBURGER.COM

Thank you!

For more information visit

Website: datalad.org

Github: github.com/datalad

Docs: docs.datalad.org

Twitter: @datalad (I am @yarikoptic, Michael is @hanke)

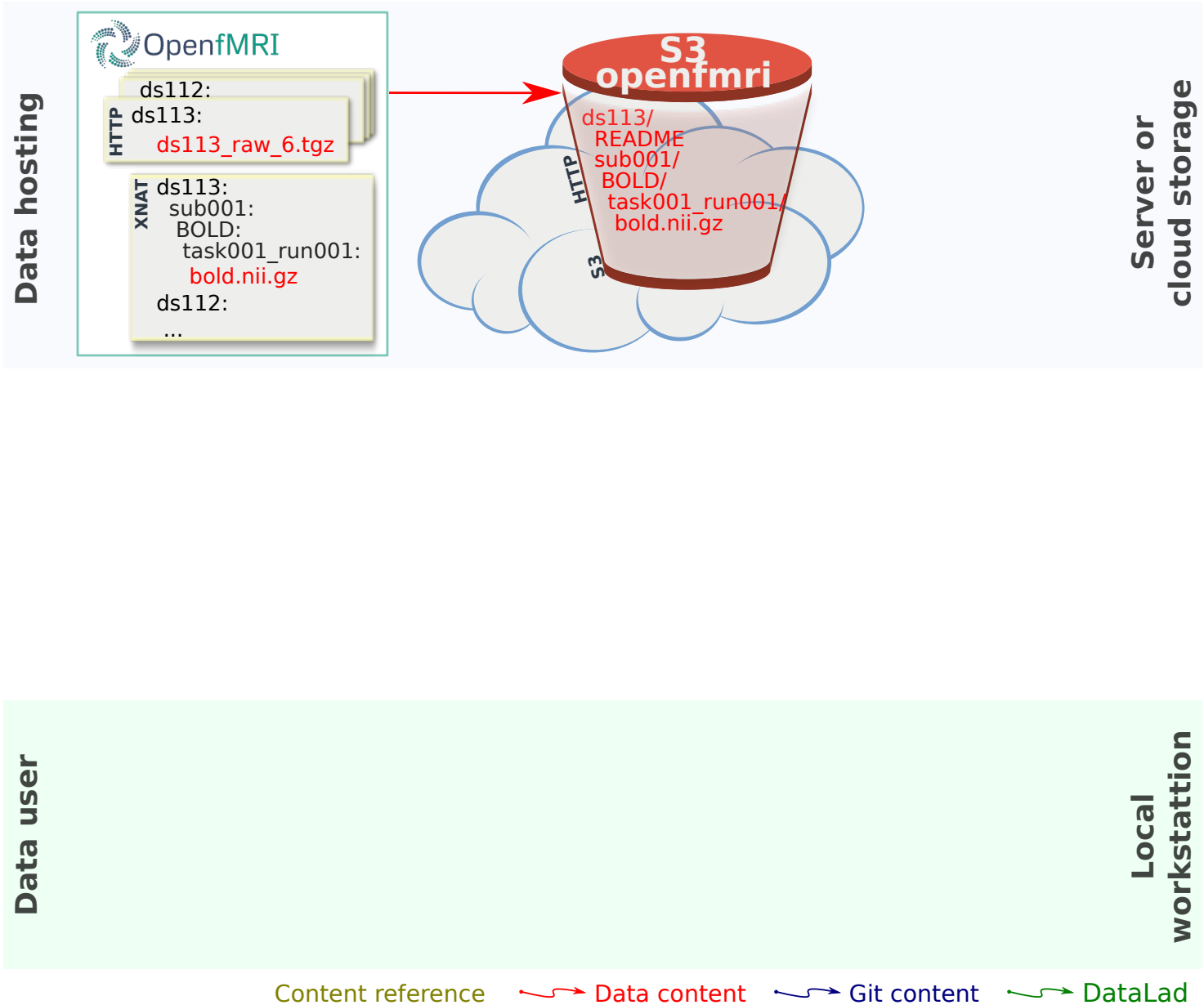
Earth: Moore Hall 415, Dartmouth College, NH, USA

References

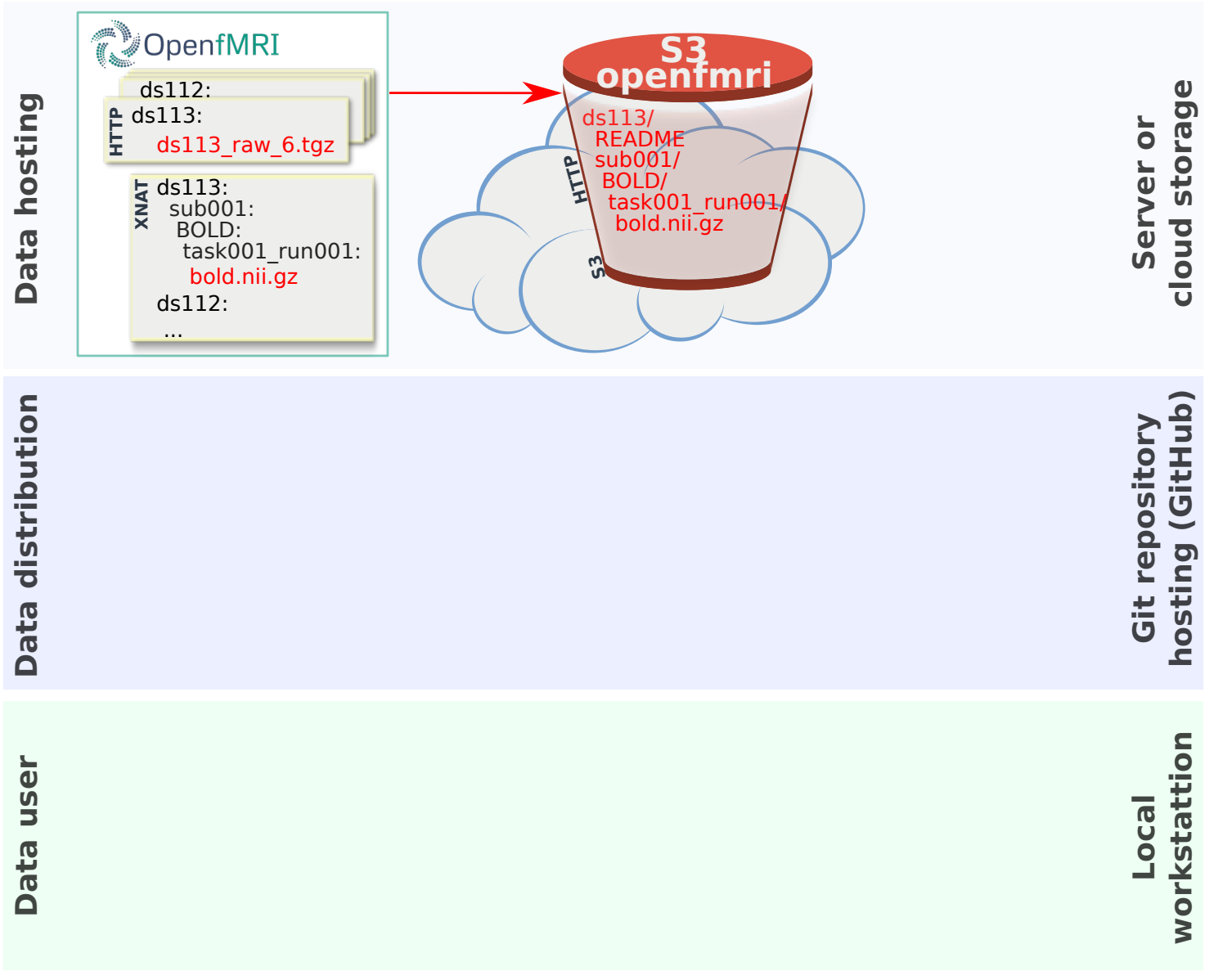
Halchenko, Y. O. (2012). Incorrect probabilities in Harvard-Oxford-sub left hemisphere. [Retrived 11-Mar-2013].

Rohlfing, T. (2013). Incorrect icbm-dti-81 atlas orientation and white matter labels. *Frontiers in Neuroscience*, 7(4).

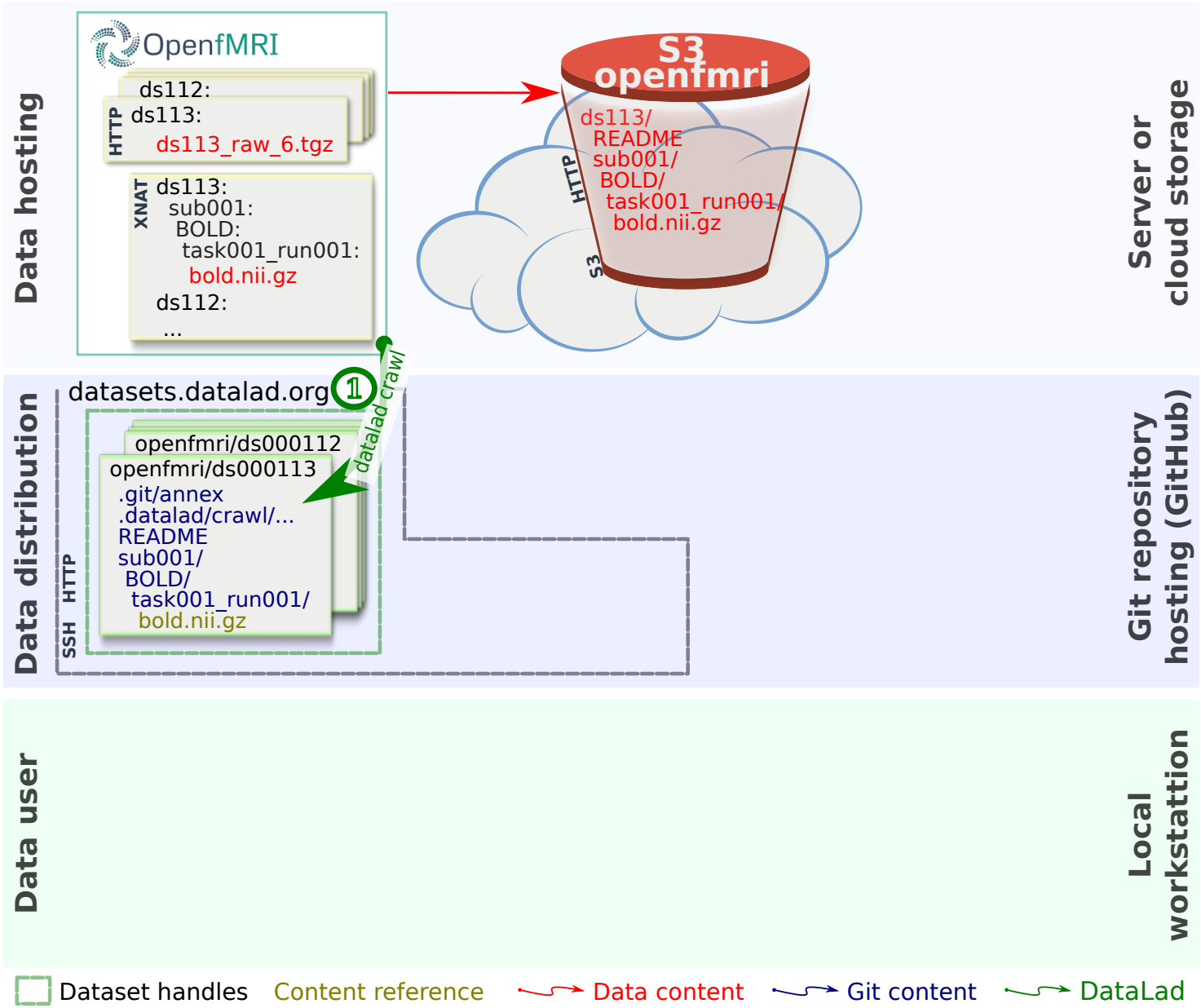
DataLad data distribution: Data life cycle



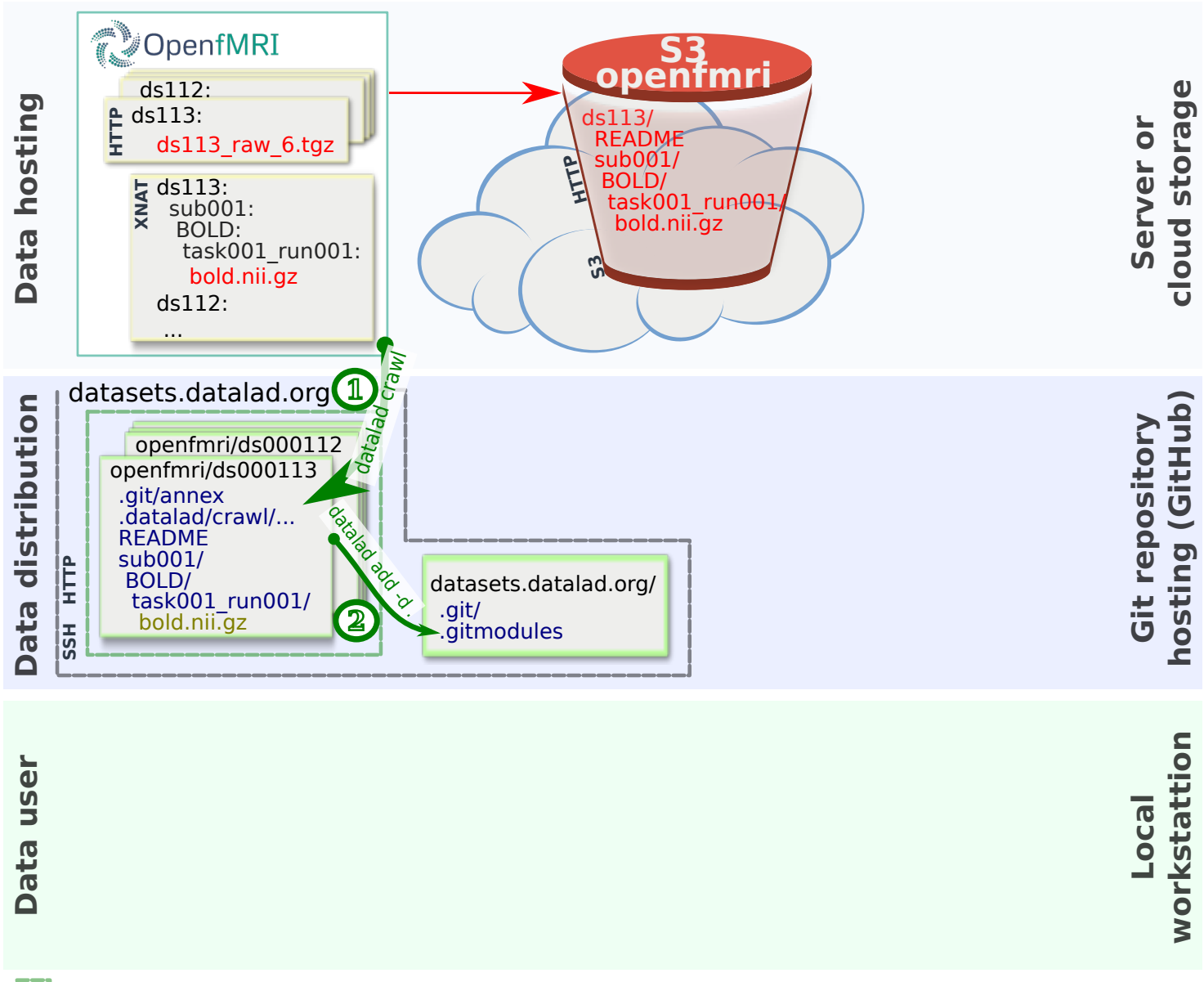
DataLad data distribution: Data life cycle



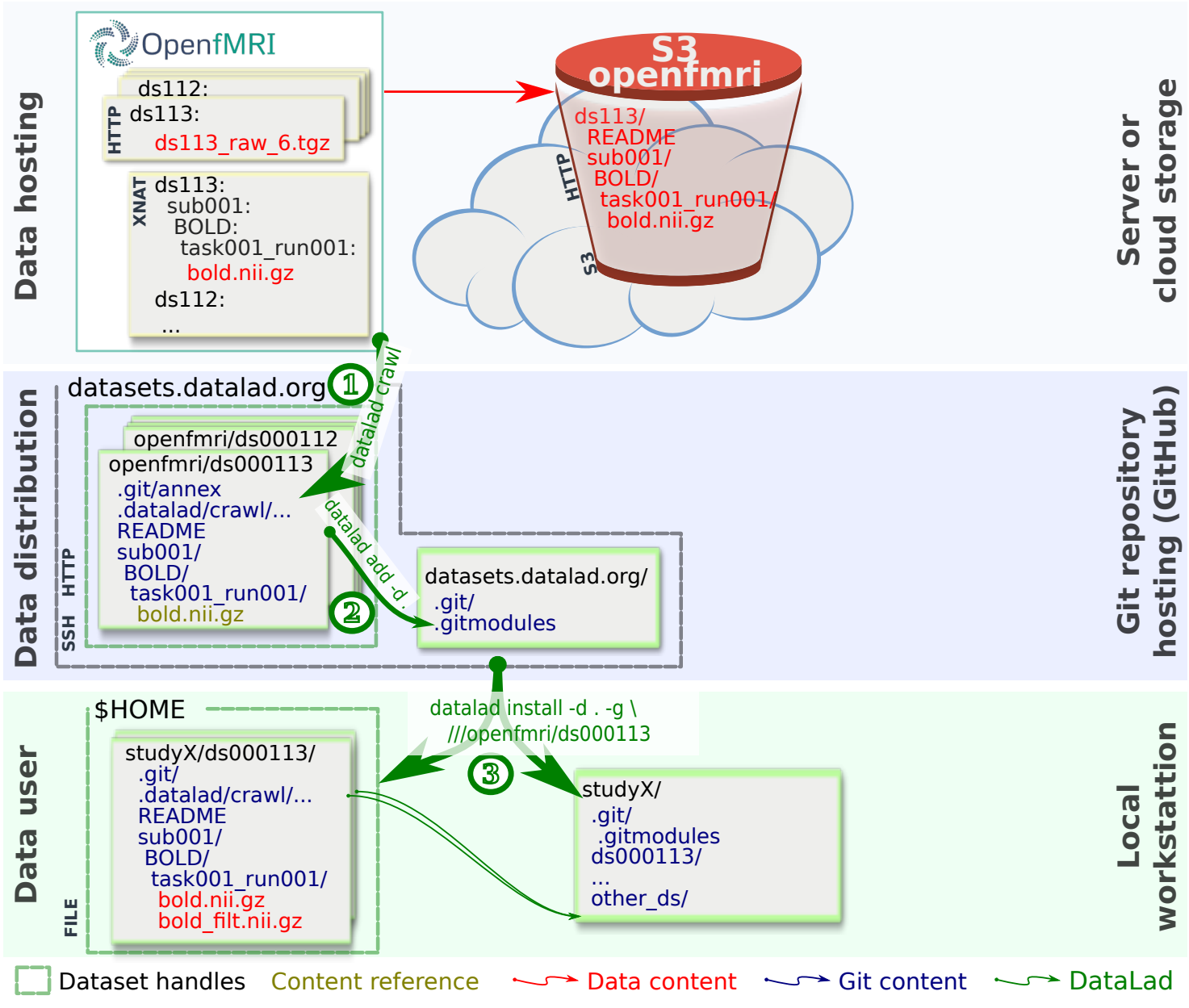
DataLad data distribution: Data life cycle



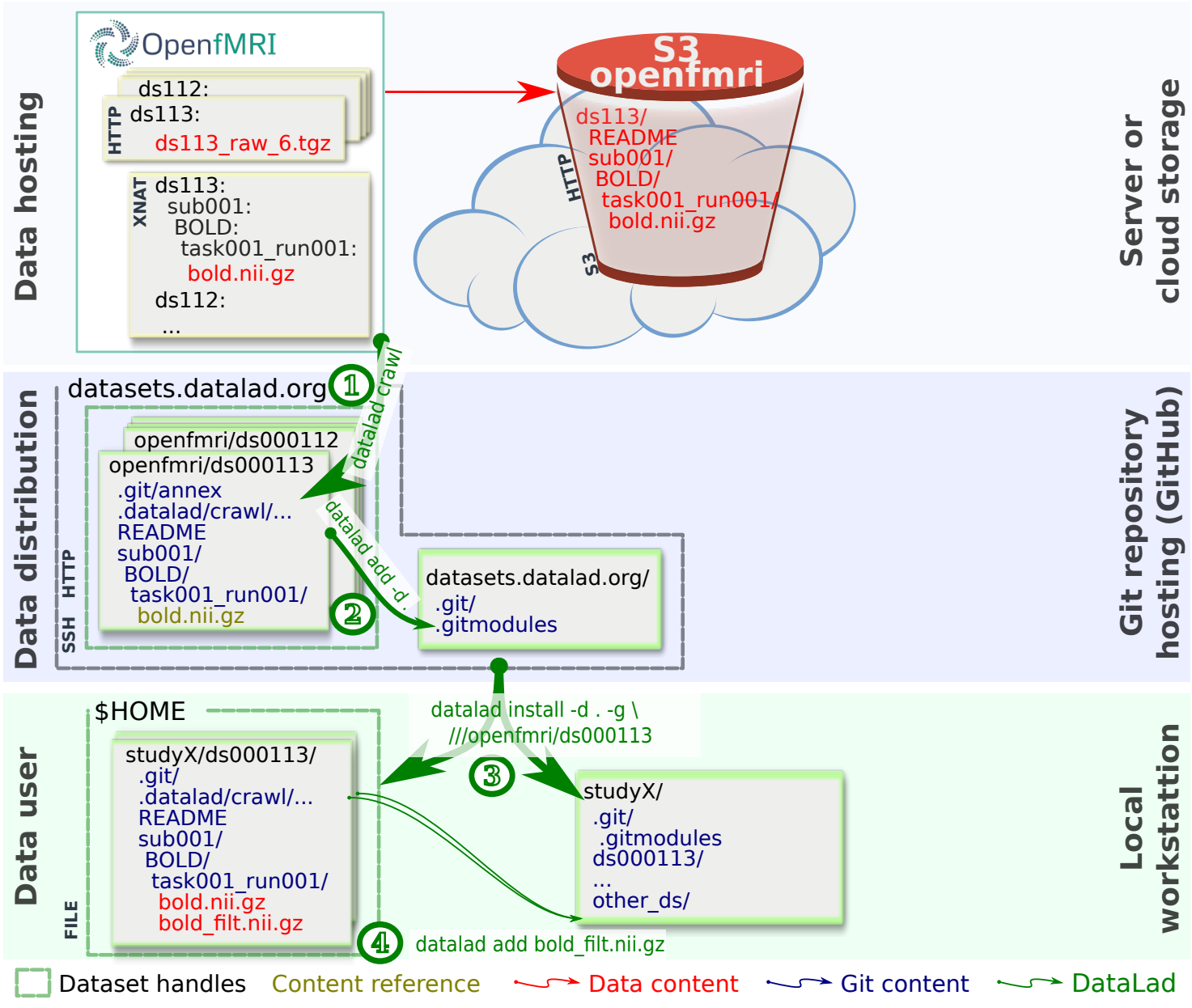
DataLad data distribution: Data life cycle



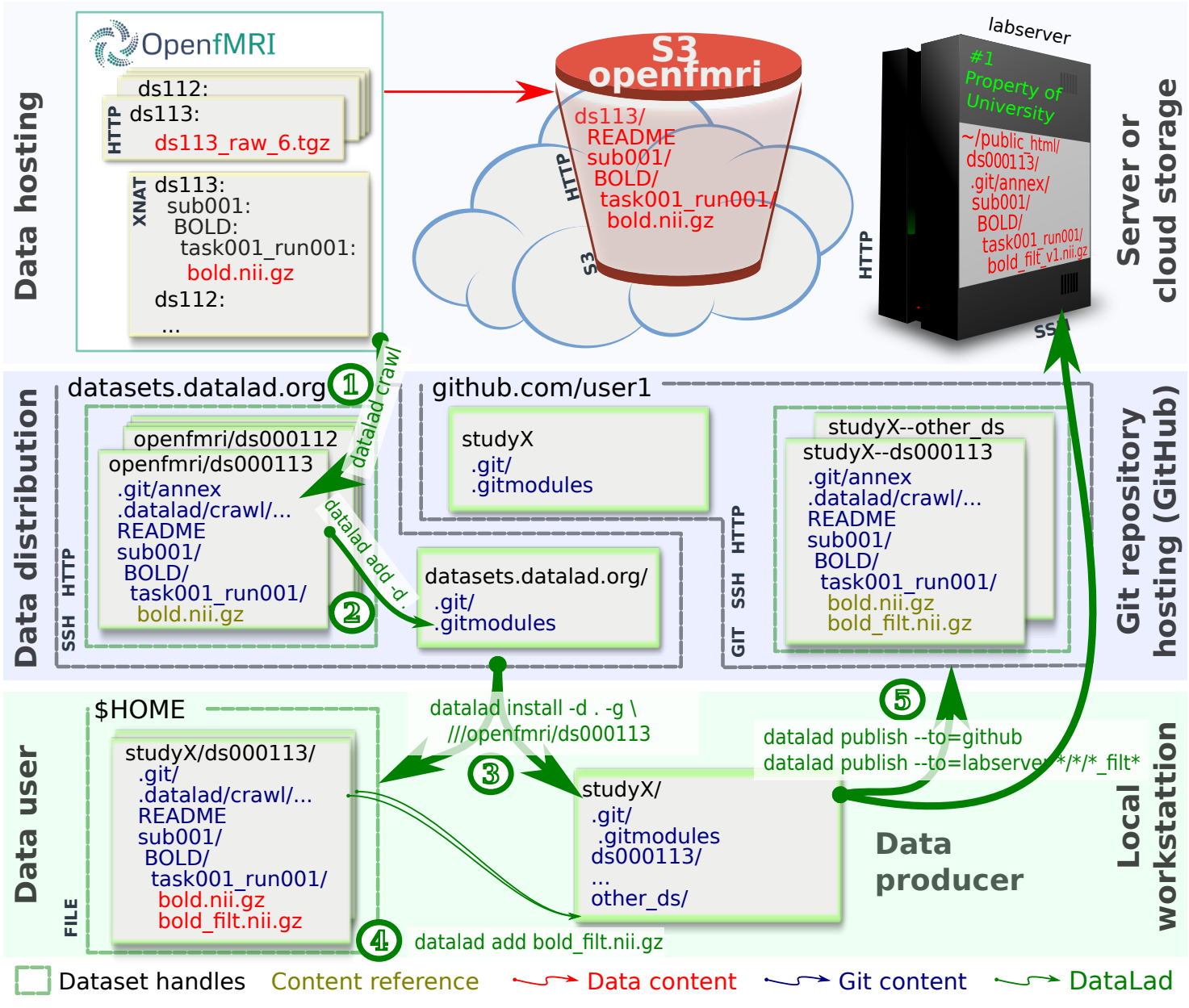
DataLad data distribution: Data life cycle



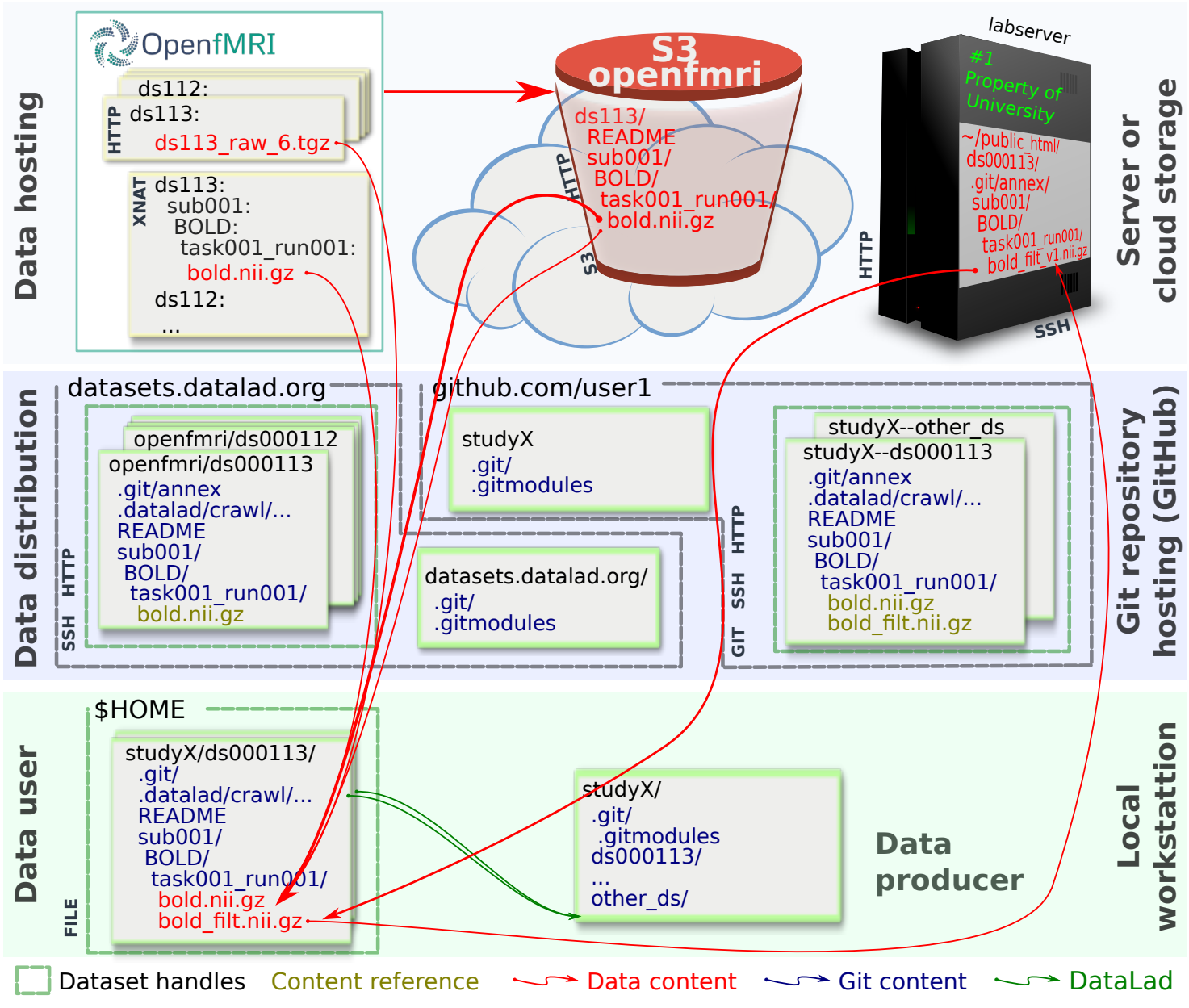
DataLad data distribution: Data life cycle



DataLad data distribution: Data life cycle



DataLad data distribution: Data life cycle



AutomagicIO: automatically fetch necessary files

Given Python code which accesses files within annex repository (example from PyMVPA):

using a 12 dof linear transformation.

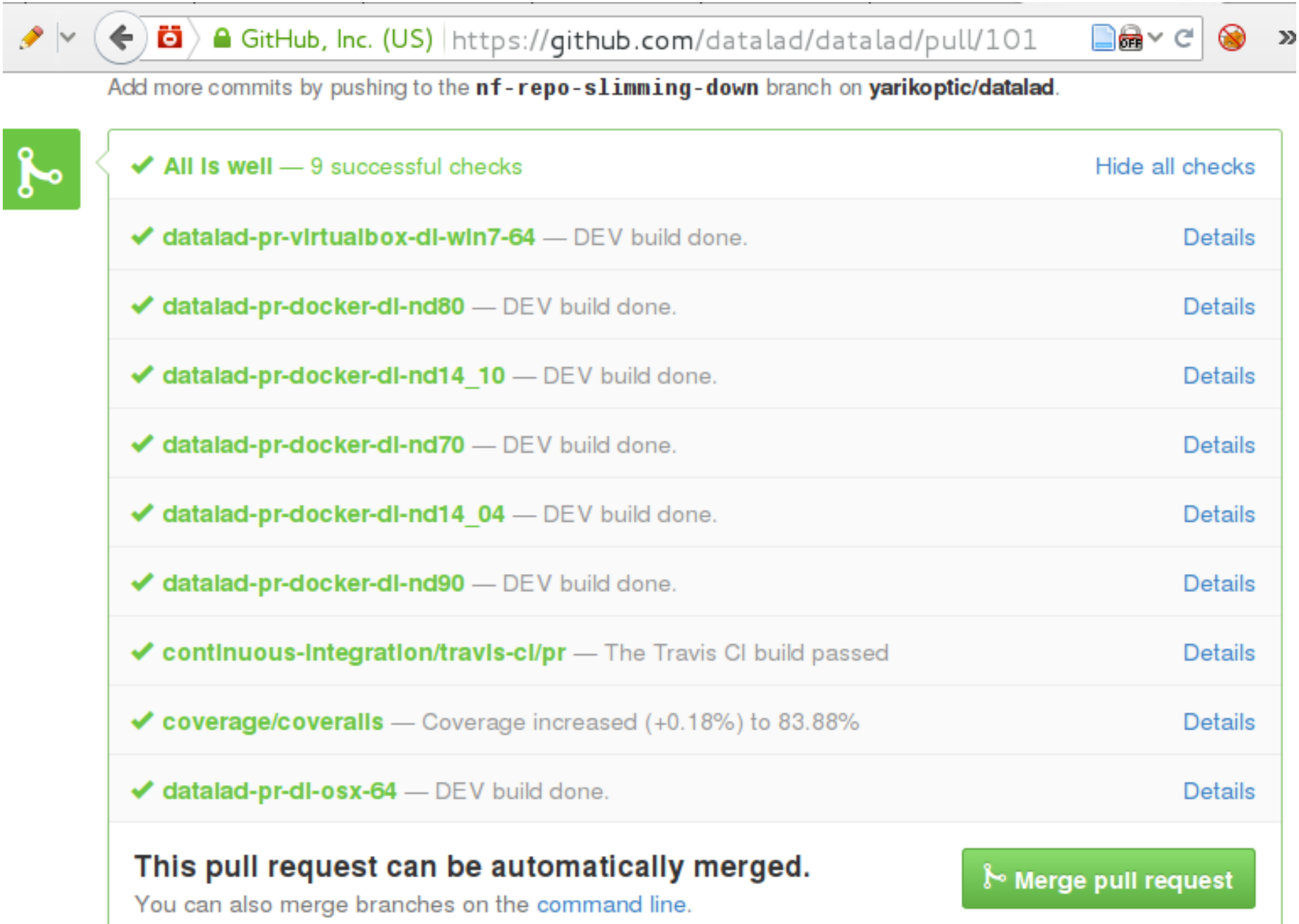
```
verbose(1, "Loading data...")
filepath = os.path.join(cfg.get('location', 'tutorial data'),
                        'hyperalignment_tutorial_data.hdf5.gz')
ds_all = h5load(filepath)
# zscore all datasets individually
_ = [zscore(ds) for ds in ds_all]
# inject the subject ID into all datasets
for i,sd in enumerate(ds_all):
    sd.sa['subject'] = np.repeat(i, len(sd))
# number of subjects
nsubjs = len(ds_all)
# number of categories
ncats = len(ds_all[0].UT)
# number of run
nruns = len(ds_all[0].UC)
verbose(2, "%d subjects" % len(ds_all))
verbose(2, "Per-subject dataset: %i samples with %i features" % ds_all[0].shape)
verbose(2, "Stimulus categories: %s" % ', '.join(ds_all[0].UT))
```

AutomagicIO: automatically fetch necessary files

DataLad can automatically fetch necessary load whenever specific file is requested:

```
2 5329.....................................:Thu 23 Jun 2016 12:39:11 PM CEST:.
(git)hopa:/tmp/PyMVPA[master]
$> datalad install -s http://data.pymvpa.org/datasets/tutorial_data /tmp/tutorial_data
2016-06-23 12:39:13,771 [INFO   ] Installing /tmp/tutorial_data (install.py:353)
1 installed item is available at
<Dataset path=/tmp/tutorial_data>
2 5329.....................................:Thu 23 Jun 2016 12:39:13 PM CEST:.
(git)hopa:/tmp/PyMVPA[master]
$> MVPA_LOCATION_TUTORIAL_DATA=/tmp/tutorial_data python -m datalad doc/examples/hyperalignment.
py
 Loading data...
2016-06-23 12:39:19,746 [INFO   ] File /tmp/tutorial_data/hyperalignment_tutorial_data.hdf5.gz h
as no content -- retrieving (auto.py:164)
/tmp/tutorial_data/.git 100%[==============================>]  15.04M  --.-KB/s    in 0.02s
  10 subjects
  Per-subject dataset: 56 samples with 3509 features
  Stimulus categories: Chair, DogFace, FemaleFace, House, MaleFace, MonkeyFace, Shoe
 Performing classification analyses...
  within-subject... done in 1.2 seconds
  between-subject (anatomically aligned)...done in 0.6 seconds
  between-subject (hyperaligned)...done in 3.3 seconds
 Average classification accuracies:
  within-subject: 0.57 +/-0.063
  between-subject (anatomically aligned): 0.42 +/-0.035
  between-subject (hyperaligned): 0.62 +/-0.050
```

DataLad's testing



ReproNim: Reproducible Basics

www.reproducibleimaging.org/module-reproducible-basics

Schedule

09:00	Command line/shell	Why and how does using the command line/shell efficiently increase reproducibility of neuroimaging studies? How can we assure that our scripts do the right thing?
12:00	Version control systems	How do version control systems help reproducibility, and which systems should be used?
16:10	Package managers and distributions	How can we establish and control computation environments using available package managers and distributions?
18:10	'Right to share'	What are the legal aspects to be kept in mind to ease sharing and reproducibility?
21:10	Other day-to-day reproducible practices	How does reproducibility help in fixing bugs? What can you do to be ready to share your studies and have them be reproducible?