Authentication and Authorization Management in Yabi

Andrea Bignamini
IA2-INAF

Scientific Motivation for a Workflow Environment at IA2

GAPS (Global Architecture of Planetary Systems) is a long-term program for the comprehensive characterization of the architectural properties of planetary systems as a function of the hosts' characteristics (mass, metallicity, environment):

- more than 50 INAF and associated scientists in Italy, and from foreign institutes
- more than 10.000 spectra at TNG/HARPS-N since August 2012

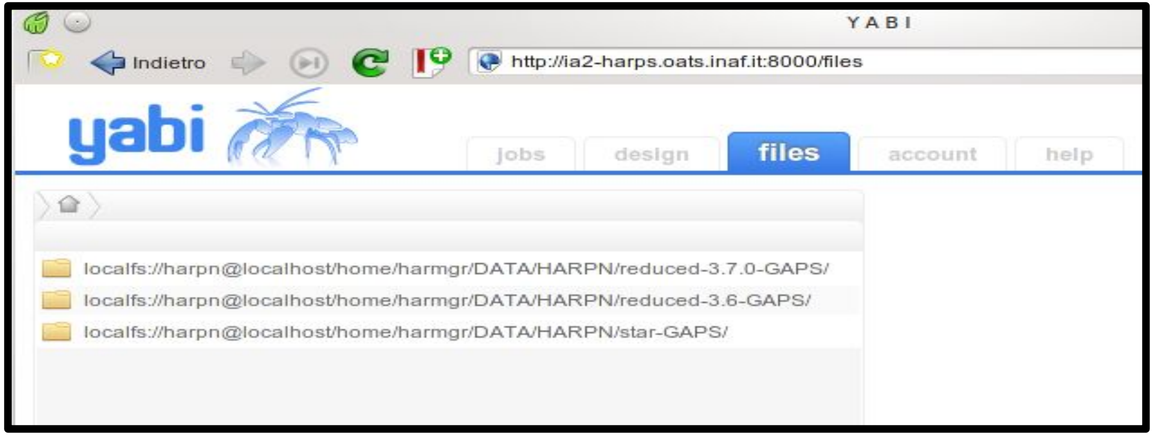
GAPS required a strong interaction with HARPS-N private data

- Customizable re-process of GAPS data with appropriate spectral line mask and options;
- Perform queries on additional metadata content;
- A flexible and collaborative tool to manage additional info about the project and the observations;
- A repository where to access, synchronize, share and search for interesting data.

Yabi

- Yabi is a 3-tier application stack to provide users with an intuitive, easy to use, abstraction of compute and data environments. Developed at the Centre for Comparative Genomics and Murdoch University, Yabi has been deployed across a diverse set of scientific disciplines and high performance computing environments
- Yabi deployed at IA2 allows users to run custom data reduction with the HARPS-N DRS pipeline
- 10 tools, 8 different kind of workflows
- 28 custom masks
- Yabi access extended to all HARPS-N users with the same login credentials of the TNG archive (tools and masks developed by GAPS are only available to GAPS)
- More than 6000 workflows executed since March 2014

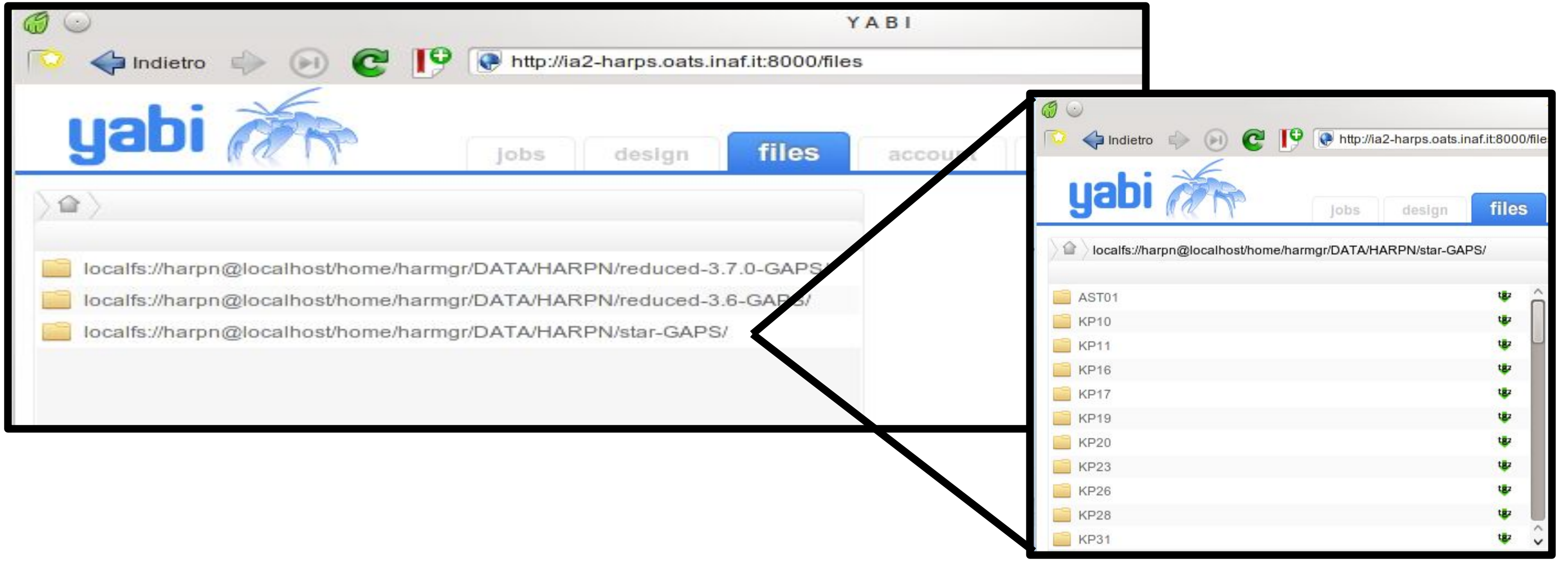
https://ccgapps.com.au/yabi/



localfs://harpn@localhost/home/harmgr/DATA/HARPN/reduced-3.7.0-GAPS/

localfs://harpn@localhost/home/harmgr/DATA/HARPN/reduced-3.6-GAPS/

localfs://harpn@localhost/home/harmgr/DATA/HARPN/star-GAPS/



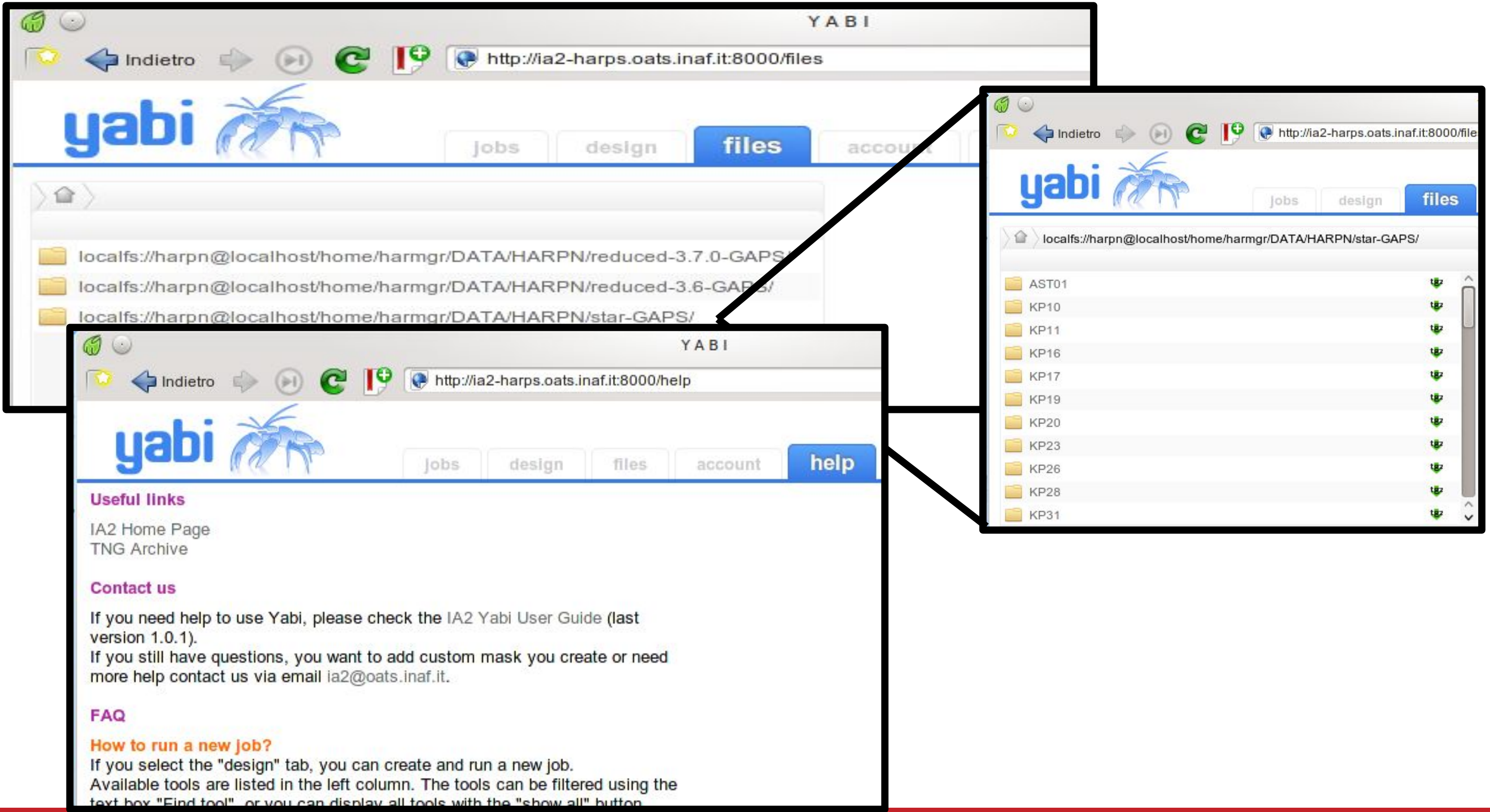
http://ia2-harps.oats.inaf.it:8000/files

localfs://harpn@localhost/home/harmgr/DATA/HARPN/reduced-3.7.0-GAPS/
localfs://harpn@localhost/home/harmgr/DATA/HARPN/reduced-3.6-GAPS/
localfs://harpn@localhost/home/harmgr/DATA/HARPN/star-GAPS/

http://ia2-harps.oats.inaf.it:8000/file

localfs://harpn@localhost/home/harmgr/DATA/HARPN/star-GAPS/

AST01
KP10
KP11
KP16
KP17
KP19
KP20
KP23
KP26
KP28
KP31



http://ia2-harps.oats.inaf.it:8000/files

localfs://harpn@localhost/home/harmgr/DATA/HARPN/reduced-3.7.0-GAPS/
localfs://harpn@localhost/home/harmgr/DATA/HARPN/reduced-3.6-GAPS/
localfs://harpn@localhost/home/harmgr/DATA/HARPN/star-GAPS/

http://ia2-harps.oats.inaf.it:8000/file

localfs://harpn@localhost/home/harmgr/DATA/HARPN/star-GAPS/

AST01
KP10
KP11
KP16
KP17
KP19
KP20
KP23
KP26
KP28
KP31

http://ia2-harps.oats.inaf.it:8000/help

Useful links

IA2 Home Page
TNG Archive

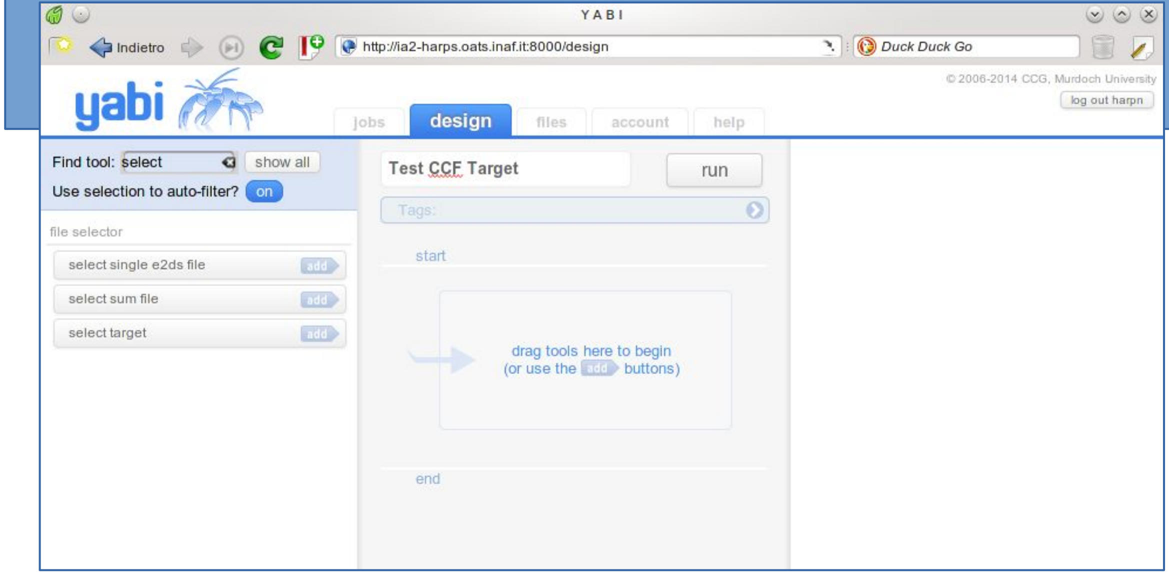
Contact us

If you need help to use Yabi, please check the IA2 Yabi User Guide (last version 1.0.1).
If you still have questions, you want to add custom mask you create or need more help contact us via email ia2@oats.inaf.it.

FAQ

How to run a new job?
If you select the "design" tab, you can create and run a new job.
Available tools are listed in the left column. The tools can be filtered using the text box "Find tool", or you can display all tools with the "show all" button.

- From the “design tab” add a select tool to the workflow (e.g. select target)



http://ia2-harps.oats.inaf.it:8000/design

© 2006-2014 CCG, Murdoch University

Find tool: select show all

Use selection to auto-filter? on

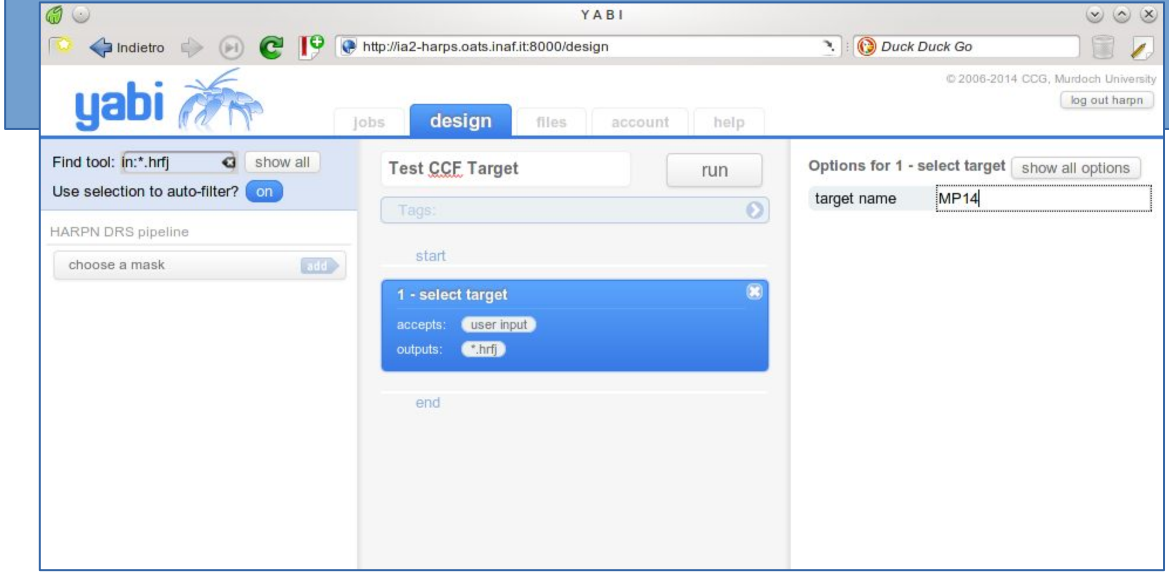
file selector

- select single e2ds file add
- select sum file add
- select target add

Test CCF Target

drag tools here to begin
(or use the add buttons)

- From the “design tab” add a select tool to the workflow (e.g. select target)
- Automatically Yabi suggests the next tool to add: choose a mask



http://ia2-harps.oats.inaf.it:8000/design

Find tool: in:*.hrfj

Use selection to auto-filter? on

HARPN DRS pipeline

choose a mask

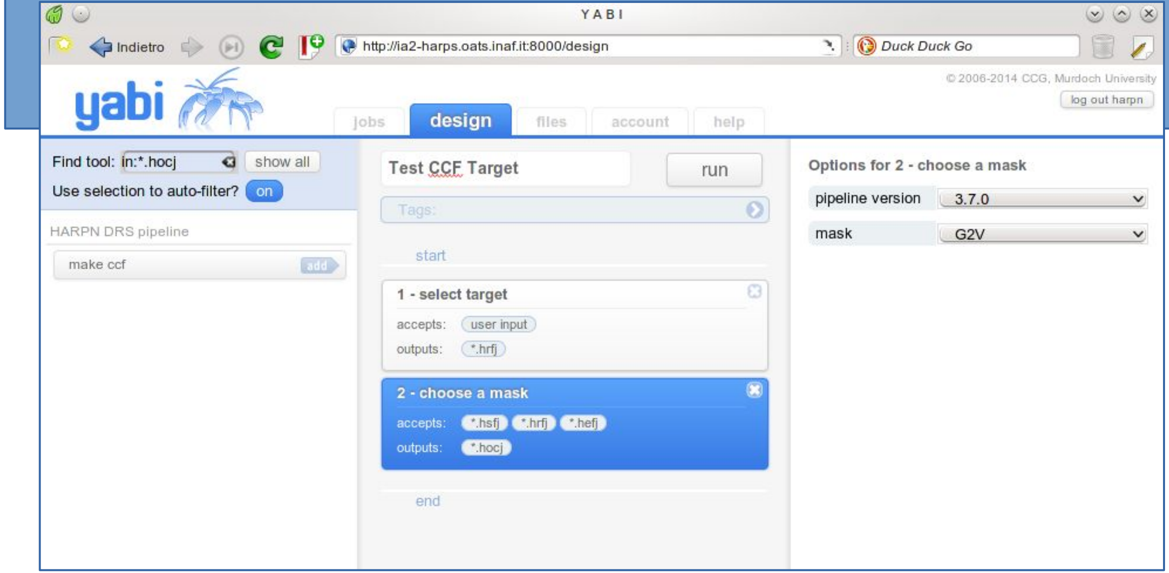
Test CCF Target

1 - select target
accepts: user input
outputs: *.hrfj

Options for 1 - select target

target name MP14

- From the “design tab” add a select tool to the workflow (e.g. select target)
- Automatically Yabi suggests the next tool to add: choose a mask
- Finally, add the tool “make ccf” and set the last input parameters



Find tool: in:*.hocj

Use selection to auto-filter? on

HARPN DRS pipeline

make ccf

Test CCF Target

1 - select target
accepts: user input
outputs: *.hrfj

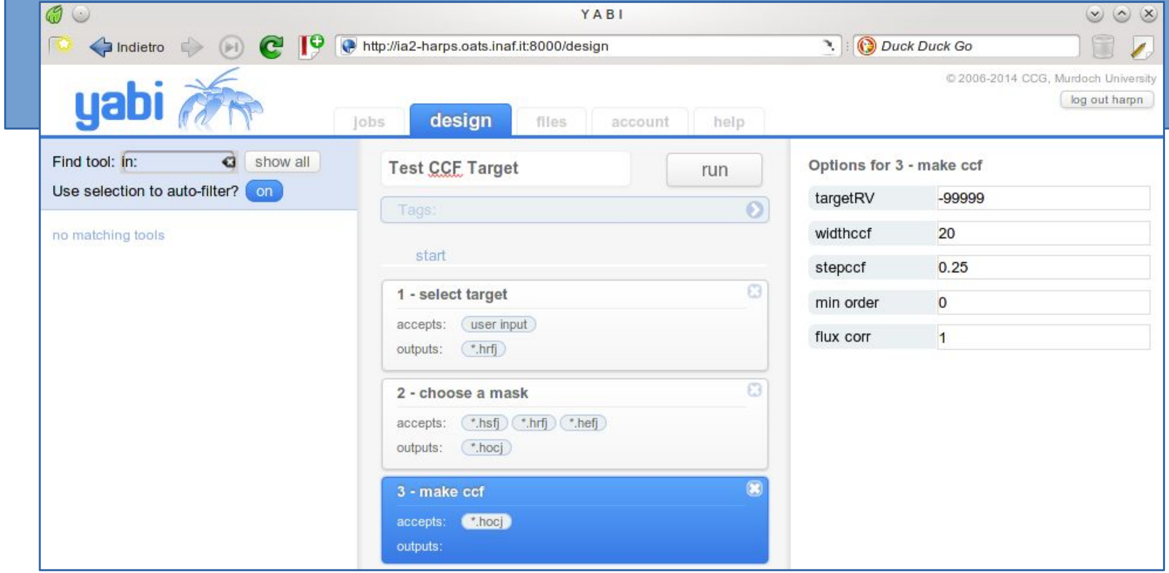
2 - choose a mask
accepts: *.hsfj *.hrfj *.hefj
outputs: *.hocj

Options for 2 - choose a mask

pipeline version: 3.7.0

mask: G2V

- From the “design tab” add a select tool to the workflow (e.g. select target)
- Automatically Yabi suggests the next tool to add: choose a mask
- Finally, add the tool “make ccf” and set the last input parameters



Test CCF Target

1 - select target
accepts: user input
outputs: *.hrfj

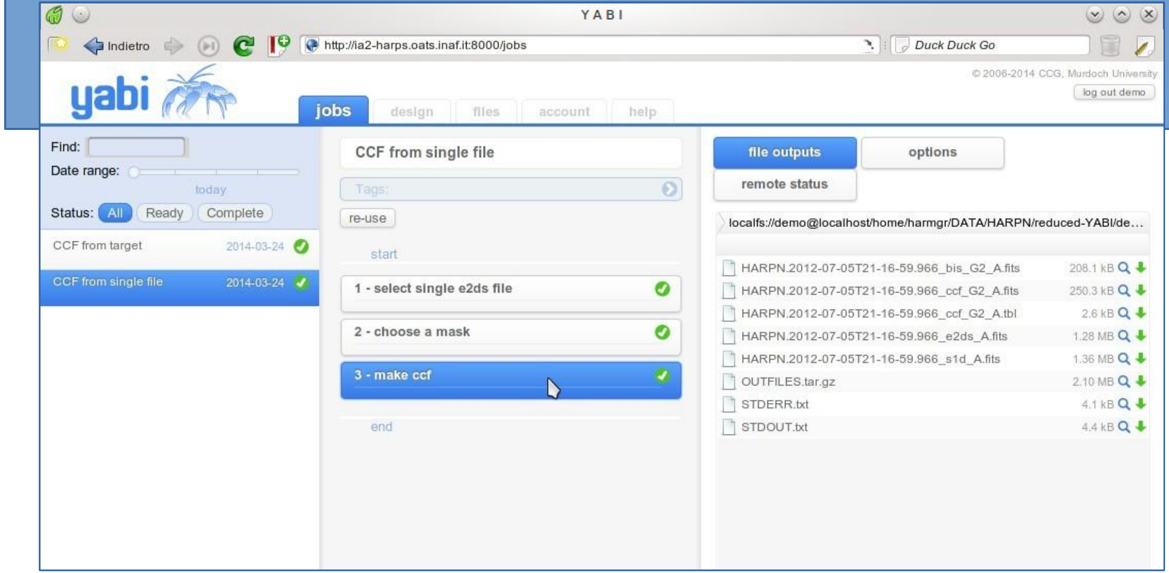
2 - choose a mask
accepts: *.hsfj *.hrfj *.hefj
outputs: *.hocj

3 - make ccf
accepts: *.hocj
outputs:

Options for 3 - make ccf

targetRV	-99999
widthccf	20
stepccf	0.25
min order	0
flux corr	1

- From the “design tab” add a select tool to the workflow (e.g. select target)
- Automatically Yabi suggests the next tool to add: choose a mask
- Finally, add the tool “make ccf” and set the last input parameters



CCF from target	2014-03-24	✓
CCF from single file	2014-03-24	✓

CCF from single file

start

1 - select single e2ds file ✓

2 - choose a mask ✓

3 - make ccf ✓

end

localfs://demo@localhost/home/harmgr/DATA/HARPN/reduced-YABI/de…

HARPN.2012-07-05T21-16-59.966_bis_G2_A.fits	208.1 kB
HARPN.2012-07-05T21-16-59.966_ccf_G2_A.fits	250.3 kB
HARPN.2012-07-05T21-16-59.966_ccf_G2_A.tbl	2.6 kB
HARPN.2012-07-05T21-16-59.966_e2ds_A.fits	1.28 MB
HARPN.2012-07-05T21-16-59.966_s1d_A.fits	1.36 MB
OUTFILES.tar.gz	2.10 MB
STDERR.txt	4.1 kB
STDOUT.txt	4.4 kB

Authorizations in Workflow Environment

- User access to data
  - User data
  - Archive data (remote)
  - Output data
- User access to computational resources
- User access to software (tools)
  - Open Software
  - Licenced software
  - Software developed by users

Backends, Credentials and Backend Credentials

- The Yabi **backend** is a demon that provides execution and file services to the Yabi stack
- The Yabi backend abstracts away the details and complexity of individual protocols and resources to provide a consistent interface to the rest of Yabi
- Any tasks carried out by Yabi are always run as the user not as Yabi
- For each user you need to create a **credential** in order to allow that user to have access to backends (i.e. to execute jobs and workflows)
- To allow a user to use a backend, you need to connect the user credential to the backend through a **backend credential**

Backends, Credentials and Backend Credentials

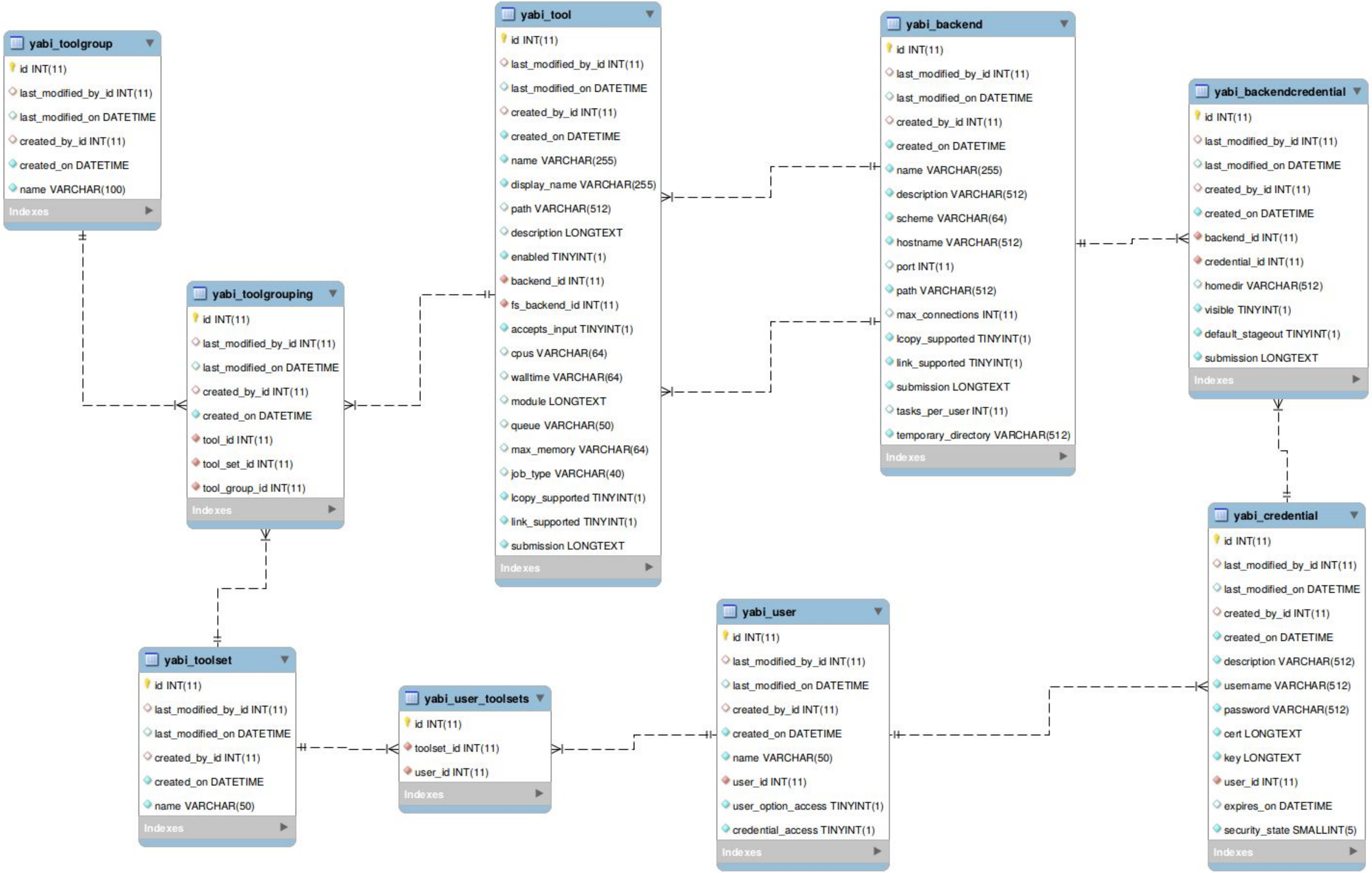
- In Yabi there are 3 types of **Backend**: execution, storage and null (for fileselector tools)
- Several schema and connectors: localex, localfs, PBSPro, SGE, Torque, SSH, SFTP, Slurm, Amazon S3, OpenStack Swift

- A **Credential** is stored encrypted within the Yabi system and belongs to a Yabi user
- In the Credential table all the fields will be needed, depending on the credential type: e.g. ssh key, certificate or user-passwd

- **Backend Credential** is a linking table between a Credential and a Backend
- Backend Credential also defines user’s Default Stageout as a single directory where all the user’s results will be staged out to

Tools, Toolsets, Toolgroups

- **Tools** are the elements the user builds the workflow with
- Yabi can run any command line tool installed on the machine where the tool will be executed
- Adding a new tool
  - A description of the tool and the parameters it accepts
  - Information about how to run the tool on a backend and the file system backend where outputs are saved
- Adding **toolsets** allows grouping of users to determine which tools they have access to
- **Toolgroups** determine how tools are grouped in the user interface
- Each tool in a **toolgroup** is assigned to a toolset, i.e. to a group of users that have access to that tool



yabi_toolgroup
- id INT(11)
- last_modified_by_id INT(11)
- last_modified_on DATETIME
- created_by_id INT(11)
- created_on DATETIME
- name VARCHAR(100)
- Indexes

yabi_tool
- id INT(11)
- last_modified_by_id INT(11)
- last_modified_on DATETIME
- created_by_id INT(11)
- created_on DATETIME
- name VARCHAR(255)
- display_name VARCHAR(255)
- path VARCHAR(512)
- description LONGTEXT
- enabled TINYINT(1)
- backend_id INT(11)
- fs_backend_id INT(11)
- accepts_input TINYINT(1)
- cpus VARCHAR(64)
- walltime VARCHAR(64)
- module LONGTEXT
- queue VARCHAR(50)
- max_memory VARCHAR(64)
- job_type VARCHAR(40)
- lcopy_supported TINYINT(1)
- link_supported TINYINT(1)
- submission LONGTEXT
- Indexes

yabi_backend
- id INT(11)
- last_modified_by_id INT(11)
- last_modified_on DATETIME
- created_by_id INT(11)
- created_on DATETIME
- name VARCHAR(255)
- description VARCHAR(512)
- scheme VARCHAR(64)
- hostname VARCHAR(512)
- port INT(11)
- path VARCHAR(512)
- max_connections INT(11)
- lcopy_supported TINYINT(1)
- link_supported TINYINT(1)
- submission LONGTEXT
- tasks_per_user INT(11)
- temporary_directory VARCHAR(512)
- Indexes

yabi_backendcredential
- id INT(11)
- last_modified_by_id INT(11)
- last_modified_on DATETIME
- created_by_id INT(11)
- created_on DATETIME
- backend_id INT(11)
- credential_id INT(11)
- homedir VARCHAR(512)
- visible TINYINT(1)
- default_stageout TINYINT(1)
- submission LONGTEXT
- Indexes

yabi_toolgrouping
- id INT(11)
- last_modified_by_id INT(11)
- last_modified_on DATETIME
- created_by_id INT(11)
- created_on DATETIME
- tool_id INT(11)
- tool_set_id INT(11)
- tool_group_id INT(11)
- Indexes

yabi_credential
- id INT(11)
- last_modified_by_id INT(11)
- last_modified_on DATETIME
- created_by_id INT(11)
- created_on DATETIME
- description VARCHAR(512)
- username VARCHAR(512)
- password VARCHAR(512)
- cert LONGTEXT
- key LONGTEXT
- user_id INT(11)
- expires_on DATETIME
- security_state SMALLINT(5)
- Indexes

yabi_user
- id INT(11)
- last_modified_by_id INT(11)
- last_modified_on DATETIME
- created_by_id INT(11)
- created_on DATETIME
- name VARCHAR(50)
- user_id INT(11)
- user_option_access TINYINT(1)
- credential_access TINYINT(1)
- Indexes

yabi_toolset
- id INT(11)
- last_modified_by_id INT(11)
- last_modified_on DATETIME
- created_by_id INT(11)
- created_on DATETIME
- name VARCHAR(50)
- Indexes

yabi_user_toolsets
- id INT(11)
- toolset_id INT(11)
- user_id INT(11)
- Indexes

re
er
77

Discussion Topics

- How to allow access to remote data resources?
- How to run workflows on data stored in geographically distributed sites?
- Standardize workflow description with Common Workflow Language (CWL)
- How to provide user access to perform custom data reduction on public data?
- What are the access policies for output data of workflows on public data?
- How to allow user to run their own software? How large this freedom should be?