XMM-Newton SAS Virtual Development Environments

Aitor Ibarra & José Marcos on behalf of XMM-Newton SAS Team

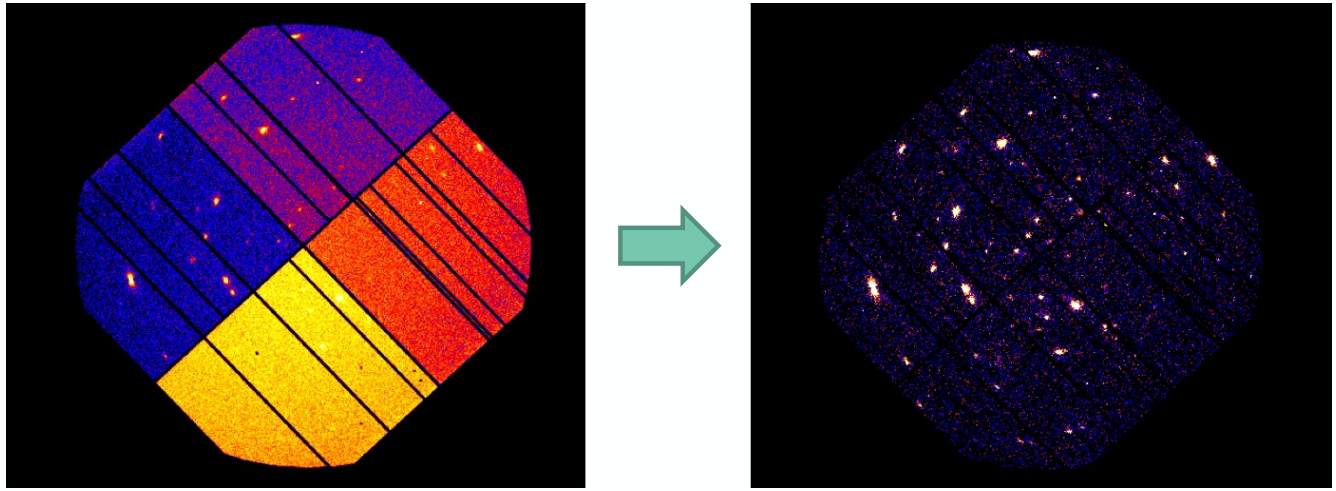
XMM2ATHENA - 26/02/2024

Overview

1. XMM-Newton Science Analysis System (SAS) Introduction
2. SAS building process
    1. Infrastructure-as-Code
3. SAS code maintenance
    1. Platform-as-Code
4. SAS Virtual Development Environments
5. SAS on cloud infrastructures: Datalabs
6. What next…

XMM-Newton Science Analysis System (SAS) Introduction

- The XMM-Newton Scientific Analysis System (SAS) is a freely distributed suite of programs for dealing with data from all XMM-Newton instruments.
- SAS is a collection of tasks (C/C++ & Fortran-[77,90]), scripts (perl & python) and libraries, specifically designed to reduce and analyze data collected by the XMM-Newton instruments.
- SAS is able to convert the XMM-Newton data from L0.5 (raw) to L3 (science products)
  - React quickly to new developments in calibration
  - Applies calibrations to raw data



  - Optimally screen/filter the data

SAS development process

- SAS code → high dependency on the C/C++/F77/F90 compiler → introduces high dependency on the Operating System
- Third-party libraries makes the building process a bit more complex.
  - HEASOFT → used in harness tests → recommendation build from source code
  - Perl → evolves with time → version dependency
  - TexLive….
- Daily builds based on ftp and tgz
  - Geographically Distributed Code Development.
  - Continuous Integration System BEFORE the Concept was invented!!!
  - Central Code Repository.
  - SAS developers authenticated via gpg key-ring.
- Too complex to maintain this infrastructure and knowledge transfer.

ENVIRONMENT

From configuring build environments manually:

SAS development process: (Dockerized SAS builder)

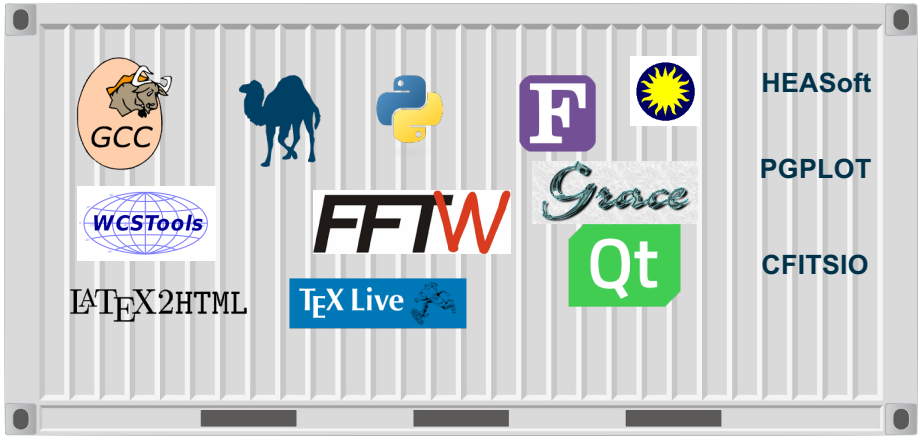
- Hardware virtualization
  - The first step to reduce costs.
  - But still too manual process.
- Infrastructure as Code
  - New paradigm to encapsulate and automatize the building process.
- Dockerization of the SAS building process:
  - Not easy because of the complexity
- Jenkins as SAS builder orchestrator

Infrastructure as Code:
- Dockerfiles
- Bitbucket code repository
- Nexus repository to store third-party libraries versions and SAS docker images

ENVIRONMENT

From configuring build environments manually:

To Docker containers with same tools and configurations:



SAS code maintenance:

- SAS development takes place on developer's infrastructure
  - Coordination with IT team → Different projects – Different needs.
  - Infrastructure set-up effort is high.
  - Switching between Operating Systems (Linux/Mac OS) to maintain the code is not easy.
  - Switching between Operating Systems to work (Windows/Linux) tedious and costly.
  - Big organizations tendencies make this process even more complex → ESA365.

- SAS Virtual Development Environment
  - Docker deployment → Platform-as-Code (PaC)
  - Easy to integrate in Cloud Platforms.
  - Easy to integrate with common code repositories (Gitlab – Bitbucket)

IaC + PaC

- Centralized setup and maintenance.
- Less HW dependency.
- Simpler backup environment.
- Security Improvement.
- Access to many environments per user.

SAS Virtual Development Environments

Where are we now?

- Working on a stable prototype for SAS developers:
  - Users account management not easy….
  - User Authentication via ESA cosmos account (the same as the XMM-Newton Archive!!)
    - Still working how to set-up accessibility to different tools are repositories.
  - Working on SAS Bitbucket migration.
  - Writing documentation for SAS developers.
- Accessible via:
  - Local Visual Studio Code (testing other IDEs)

SAS virtual development environments

SAS Virtual Development Environment

SAS Technology Stack

Next Steps…

ATLASSIAN Bitbucket

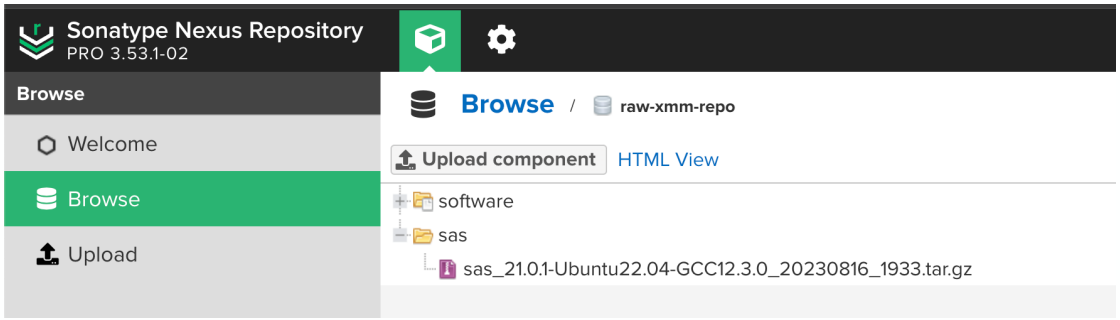
GitLab

kubernetes

SAS Virtual Development Environments

Steps to set-up SAS VDE

1.- Authenticate, Download and Run SAS Docker builder Image from ESA Nexus repository.
2.- Download or have access to CCF repository in your host machine.



#> docker pull scidockreg.esac.esa.int:61900/xmm/sasbuilder/ubuntu-22.04/gcc-11.3.0:1.0.0-3

#> mkdir /pathToCCF/ccf
#> rsync -v -a --delete --delete-after --force --include='*.CCF' --exclude='*/' sasdev-xmm.esac.esa.int::XMM_CCF /pathToCCF/

SAS Virtual Development Environments

Steps to set-up SAS VDE

3.- Run de SAS builder image
4.- Build SAS

#> ./docker_run_it.sh -o ubuntu-22.04 -g gcc-11.3.0

#> sasbuilder/bin/buildsas "" "11.3.0" devtrack dev

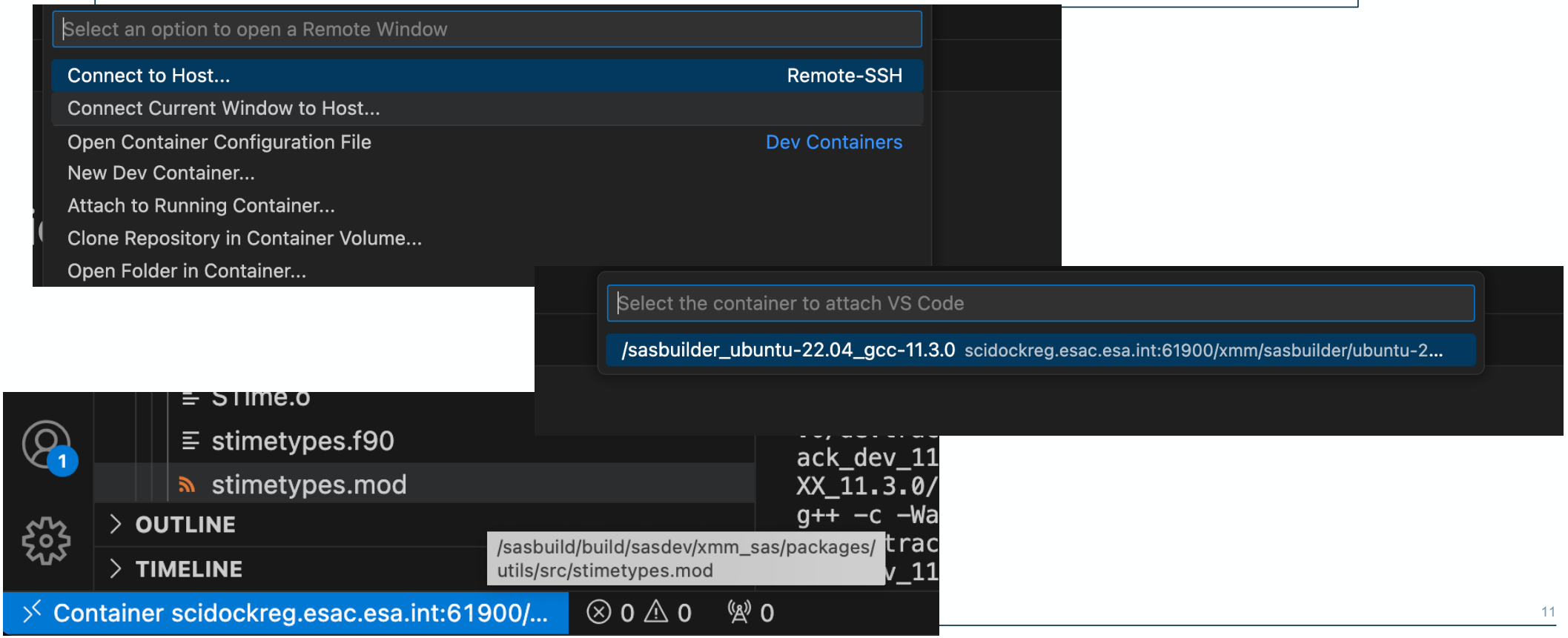
```
aibarra@xmml110:~/SAS_DOCKER/SASDEV$ ./docker_run_it.sh -o ubuntu-22.04 -g gcc-11.3.0
Current UID:GID : 5159:100
Using group users with gid 100
Generating locales (this might take a while)...
  en_US.UTF-8... done
Generation complete.
. /sasbuild/local/docker_ubuntu_22_04/GNU_CC_CXX_11.3.

                          Workstation docker_ubun

                        Linux Ubuntu22.04 x86_64


[sasbuild@docker_ubuntu_22_04 ~]$
```

```
sasbuild@go-vm5f03f64e:~/docker_dev/wa04_sas_builder_docker$ ./docker_run_it.sh -o ubuntu-22.04 -g gcc-11.3.0
Current UID:GID : 5153:200
Creating group sas with gid 200
Generating locales (this might take a while)...
  en_US.UTF-8... done
Generation complete.
. /sasbuild/local/docker_ubuntu_22_04/GNU_CC_CXX_11.3.0/headas/architecture/headas-init.sh


                                Workstation docker_ubuntu_22_04

                              Linux Ubuntu22.04 x86_64


[sasbuild@docker_ubuntu_22_04 ~]$ sasbuilder/bin/buildsas "" "11.3.0" devtrack dev

Warning: buildsas: Setting default builder configuration
```

SAS Virtual Development Environments

Steps to set-up SAS VDE

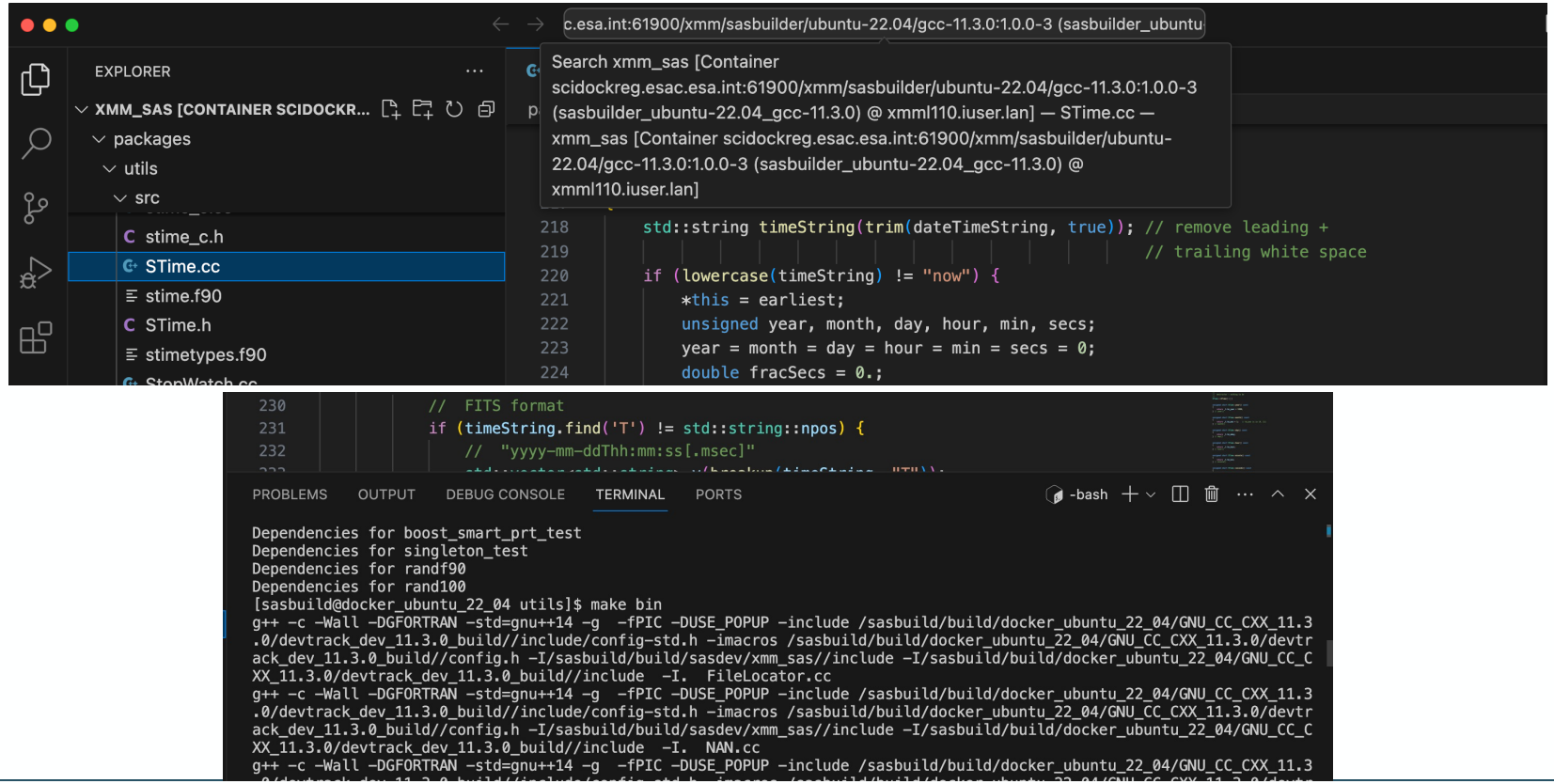
5.- Start coding!!!



SAS Virtual Development Environments

Steps to set-up SAS VDE

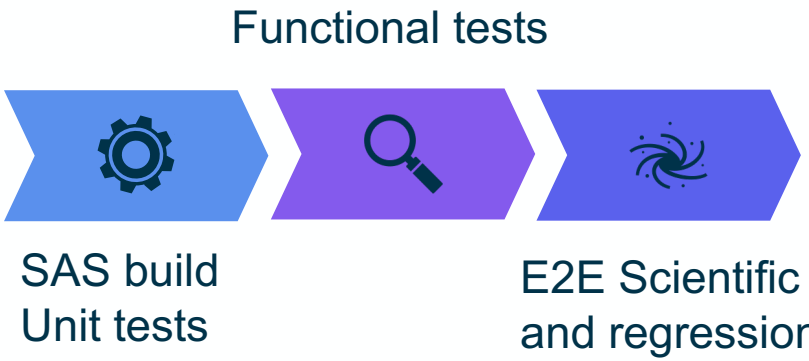
5.- Start coding!!!

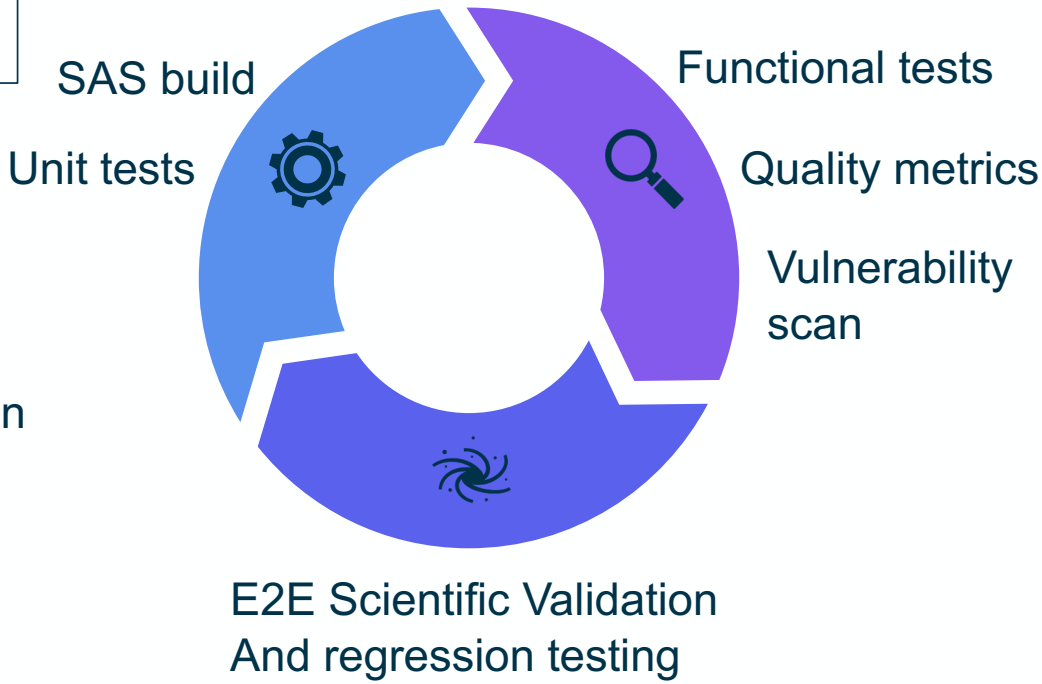


SAS building process based on dockers

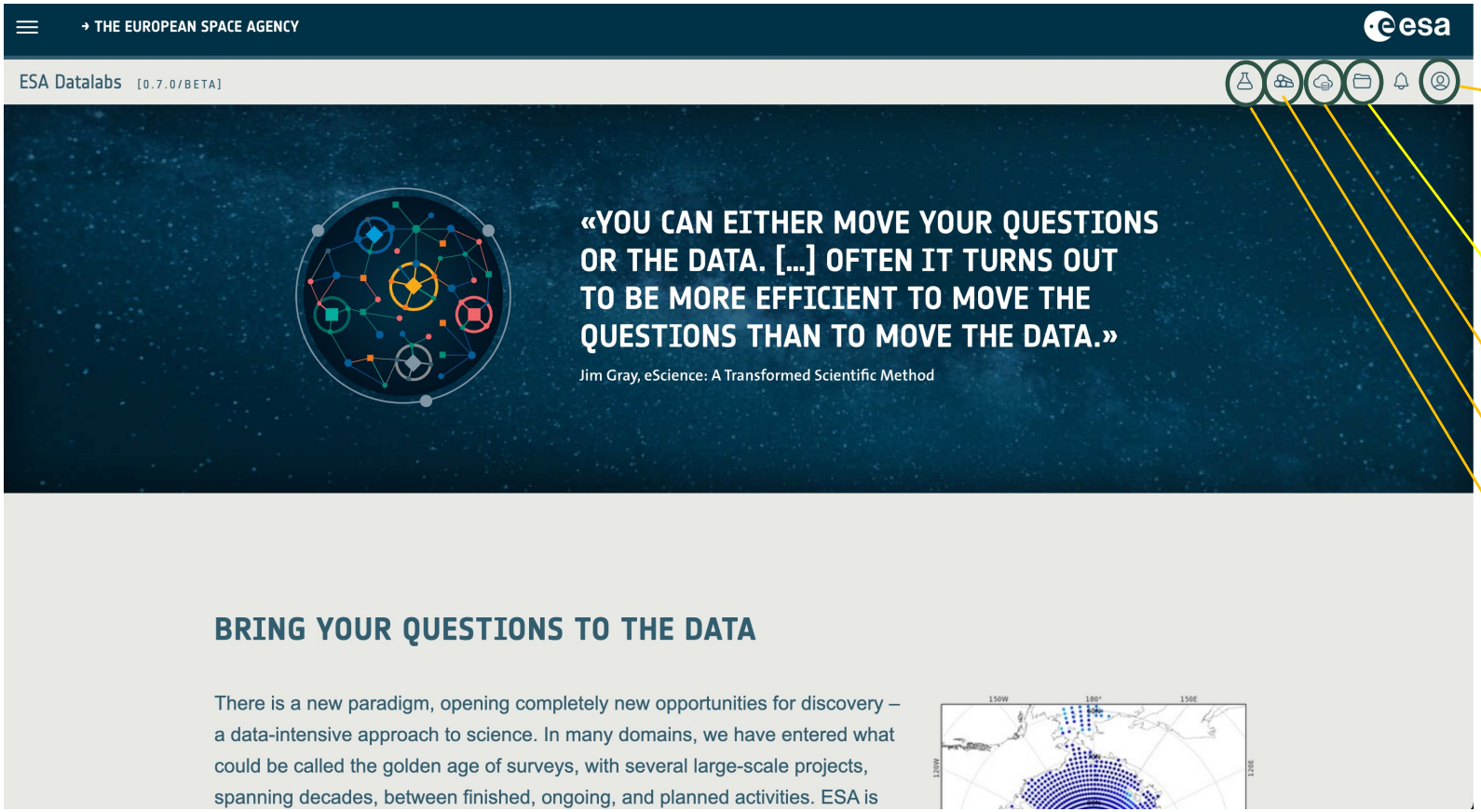
Goal

- **Avoid hardware dependencies**
- **To have a bit more agile Continuous Integration System**
- **Get ready to move the integration system to a cloud platform**





SAS in Datalabs



Registration is subject to moderation

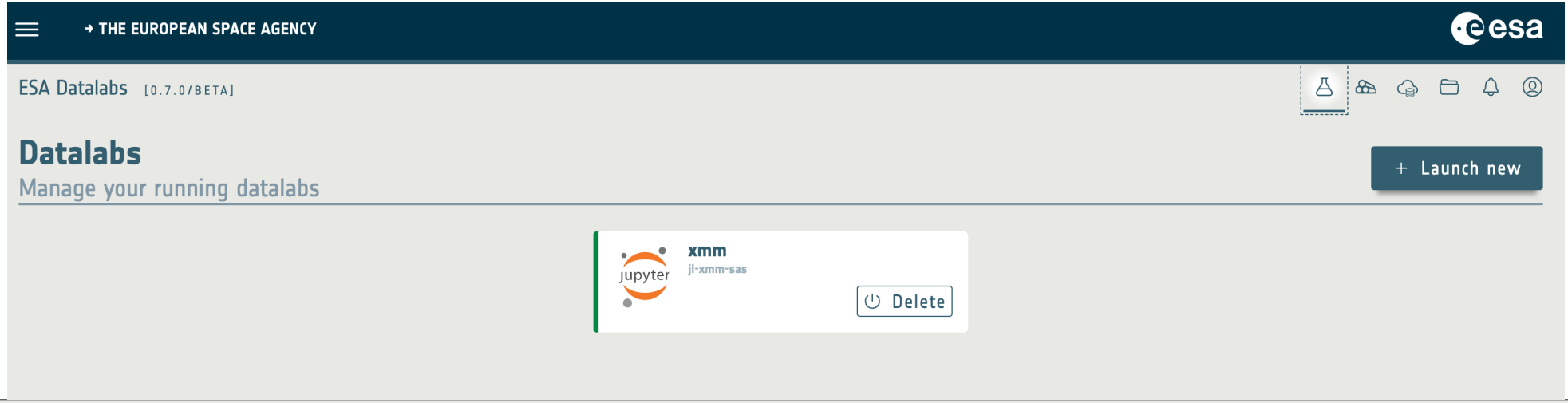
Personal and team workspace

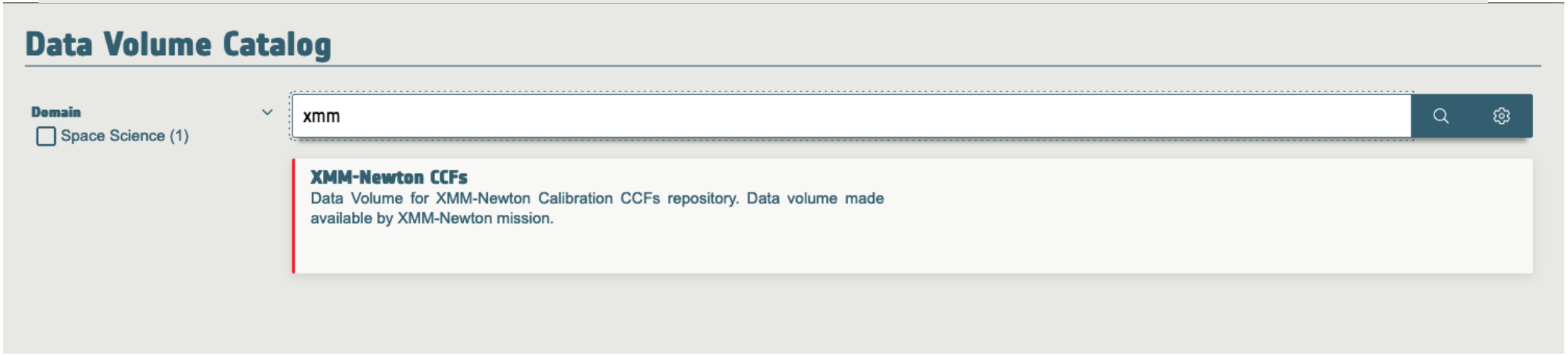
Data Volumes

Pipelines

Datalabs

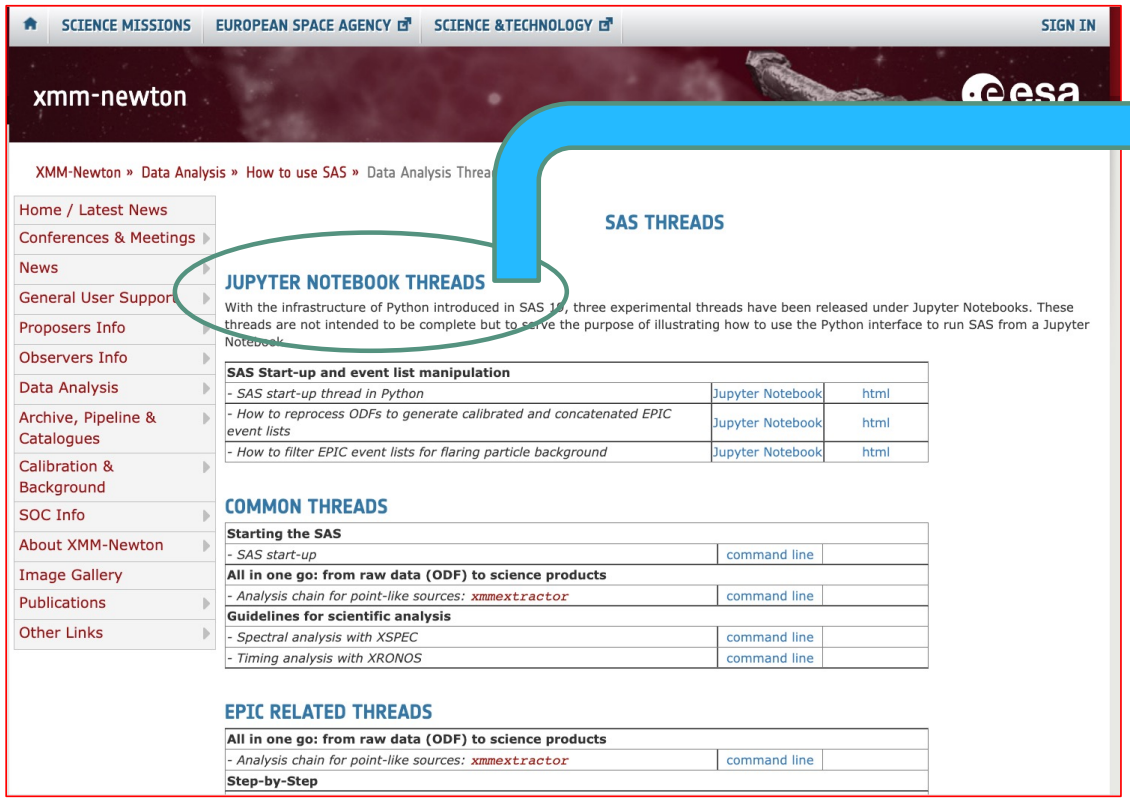
SAS in Datalabs

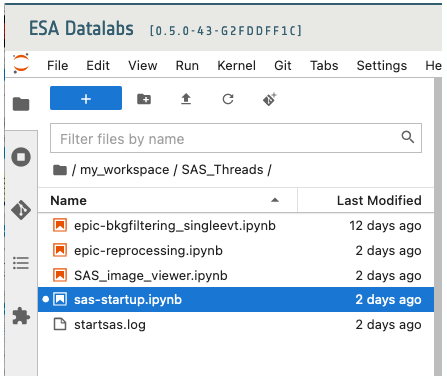


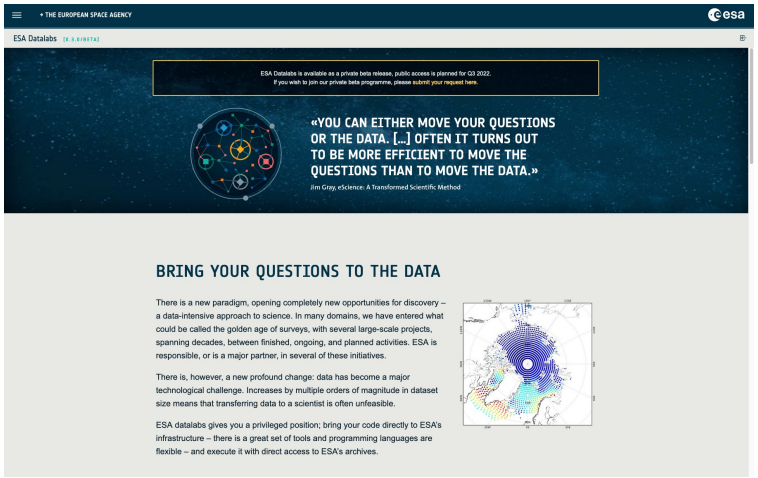


SAS evolution along the years: DL4SAS + Threads

- **To help users to analyse XMM-Newton data, a set of Data Analysis Threads are provided to the community.**

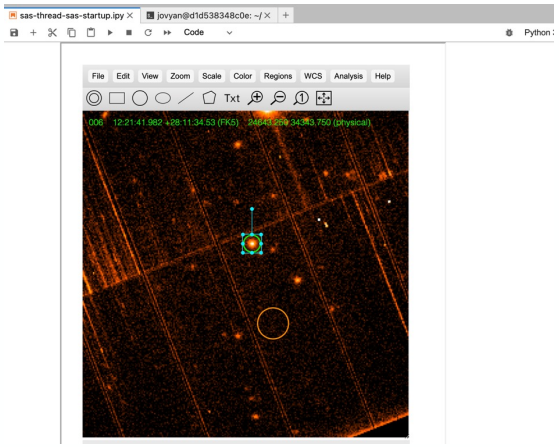


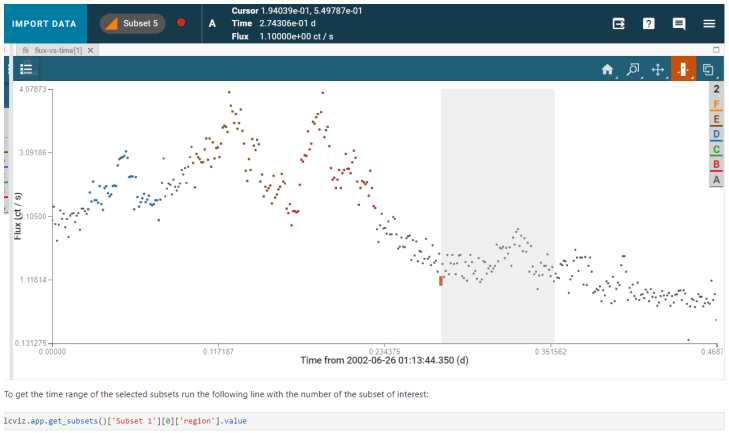




SAS on cloud infrastructures: Datalabs/SciServer??

- SAS is a docker can be used in any cloud platform.
- Scientific cloud platforms (Datalabs and SciServer) uses Jupyter Lab as user interface.
- SAS can be adapted to thes platforms as long as we provide to user the basic functionalities to work with XMM-Newton data:
  - Image visualization
  - Light-curve visualization
  - Interactivity with these two functionalities
- Currently working on how to add interactivity in SAS Jupyter Lab environments using:
  - jpyjs9
  - lcviz





Future Work

- Improve the SAS DevOps infrastructure
  - Fully automatic SAS building deployment indifferent cloud environments
- Improve/automatize SAS Virtual Development Environments
- Improve SAS & Datalabs (cloud) usage
  - Improve the SAS python infrastructure to help users to create their own scripts.
  - Interactivity
    - Source a background regions
    - Good Time Interval selection
- More things to come…