ASB CCN#2

Final Presentation Meeting

4 October 2018

AGENDA

1. Introduction to the ASB project and the CCN2 extension
2. Introduction to the ASB technical concepts
3. Technical achievements and demonstration scenario
4. Live demonstration
5. CCN2 deliverables
6. Concluding remarks
7. Future work

THE ASB PROJECT

- **Looking for a way to reduce development time and costs for building a processing facility (PF).**
- **ASB established to investigate the potential for automating the building of PFs.**

  Key challenges:

  - Provide a scalable and dynamically re-configurable platform for processing facilities having variable and complex processing demands such as an Instrument Processing Facilities (IPF).
  - Automate the building of workflows from user specifications making the deployment of the processors in the cloud seamless for users, but under their control.

- **Along the way Principle Investigators and Scientists highlighted the need for support over the development lifecycle as well as on-demand processing in addition to systematic processing.**
  - Need to host on PC, Server, and Cloud
  - Manage and process large data series (Big Data)
  - Complex workflows executable on multiple platforms globally distributed
  - Additional user interfaces

THE ASB CCN2 EXTENSION

- The ASB implementation is driven by two distinct usages: automated processing in IPFs and support for algorithm development.
- In the main ASB project, two use cases reproducing real-life applications have been prepared:
  - The SMOS L1OP use case reproduces a fragment of the SMOS IPF. This ASB prototype application is processing incoming products without user intervention.
  - The PROBA-V n-Daily Compositor use case reproduces the processing chain of the Compositor implemented in the PROBA-V MEP. The resulting application is meant to be executed on-demand.
- The ASB framework has been selected in project PROBA-V MEP Third Party Services (MEP-TPS) for supporting the development, testing and execution of algorithms developed by scientific partners.
- ASB was lacking functionalities identified as needed in discussions with potential users, and in particular with the MEP-TPS partners.

➢ **The features selected for implementation in ASB CCN2 partially fill the gap between the main ASB project and the MEP-TPS project.**

TASKS STATUS

ASB CCN#2 Proposal – ASB-SA-PRO-002-CCN2

Section 1.4.2 – Relationship CCN2, MEP-TPS Project and Follow-on Work

- Complete
- Work in progress
- To be worked on

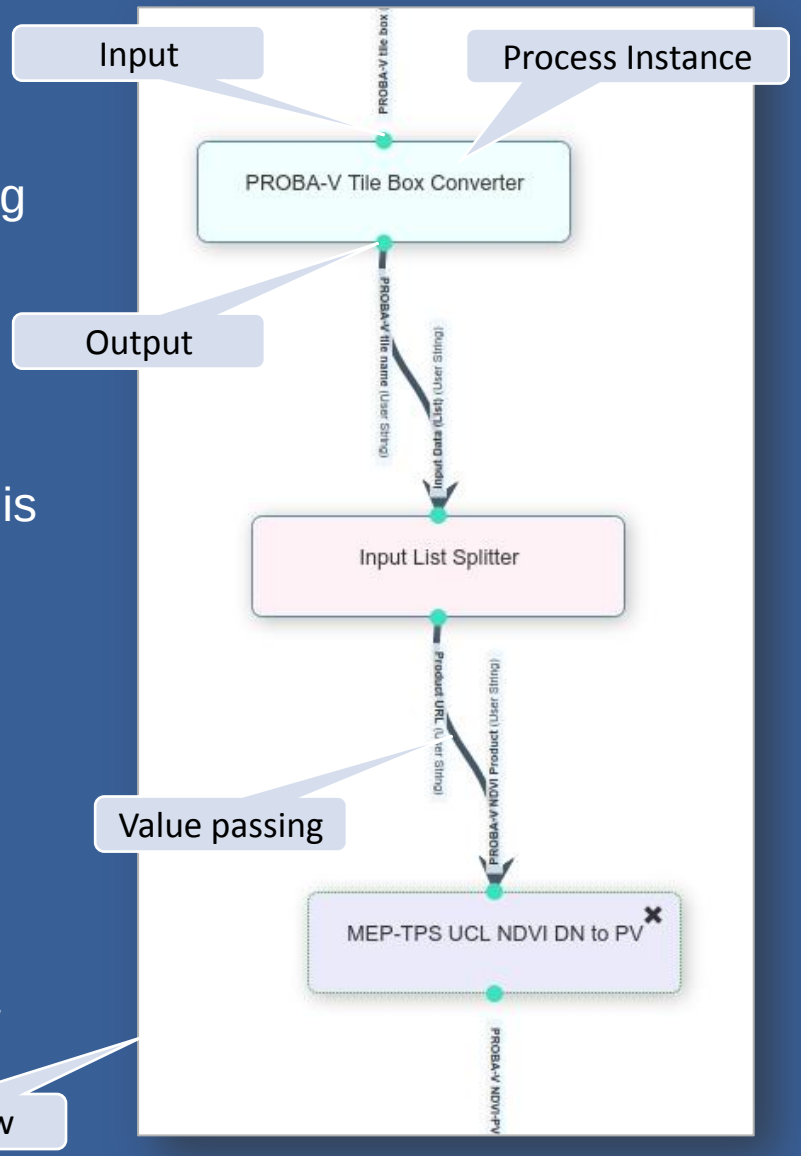
1. ASB CCN#2
   - 1a. Addition of the developer role (= "contributing users" in the MEP-TPS)
   - 1b. Remote execution of algorithms
   - 1c. User selected processor execution

2. MEP-TPS
   - 2a. Processor execution scheduler module
   - 2b. Remote access to generated products
   - 2c. Programmatic processor execution

3. Follow-on work
   - 3a. Upgrade of the developer's interface
   - 3b. Processor monitoring (enh.)
   - 3c. Knowledge elements versioning (enh.)
   - 3d. Automated build and user interface
   - 3e. Parallelising Large Collections of Data
   - 3f. Automate IT resources dynamicity
   - 3g. Data Manager
   - 3h. Processor execution reporting (enh.)
   - 3i. Extreme Real-life conditions and testing

ASB TECHNICAL CONCEPTS

- An **Algorithm** (user provided or built-in) is packaged within a **Docker container** and referred to as a **Process** when imported in ASB.
  - ➢ ASB now includes a **Process Import Tool** for packaging and importing custom algorithms.
- A **Process** has (typed) **input and output parameters**.
  - Input parameters are above and Output parameters are below.
- A **Processor** is an application that is linked to a **Mission**, or Project, and is defined with a **Workflow** of Processes.
  - ➢ ASB includes a graphical **Workflow Editor** for creating workflows interactively, e.g. connecting parameters with drag-n-drop.
- Processors can be executed on-demand or scheduled.
- A Processor can be executed many times with the same or different parameters.
  - ➢ ASB includes interfaces for monitoring and control, reporting and data access.

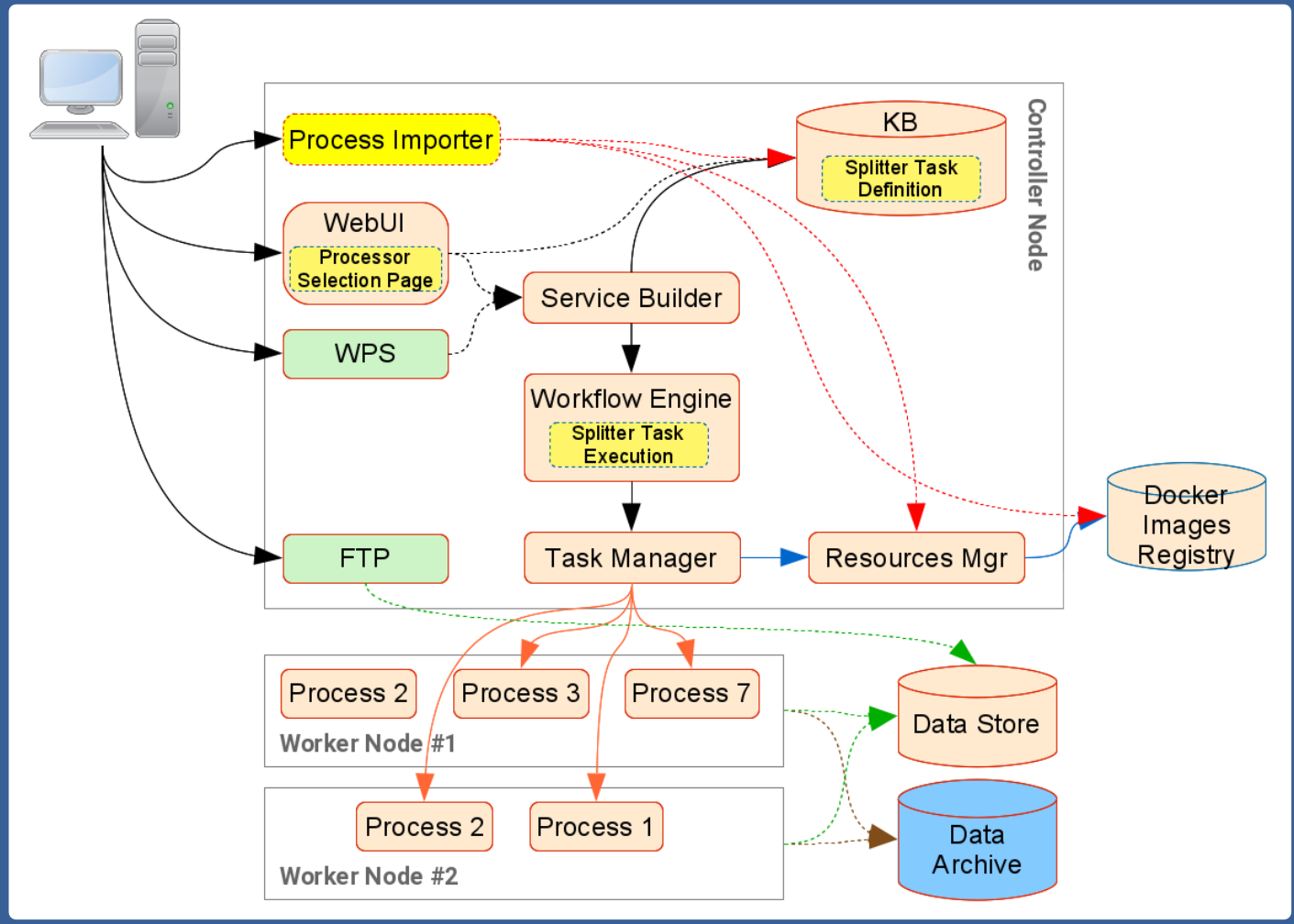


ASB CCN2 TECHNICAL ACHIEVEMENTS

Implemented extensions

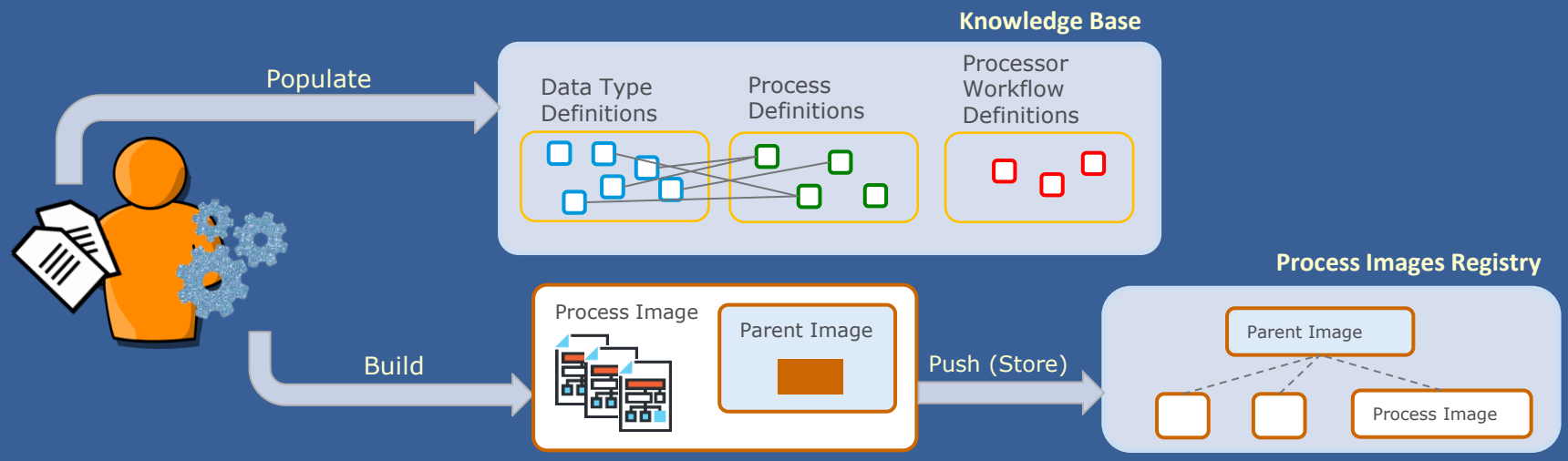
- Process Import Tool
- Parallelisation of large collections of data using Splitter tasks
- Processor selection and execution

The impact of the new features on the overall architecture is highlighted in yellow:

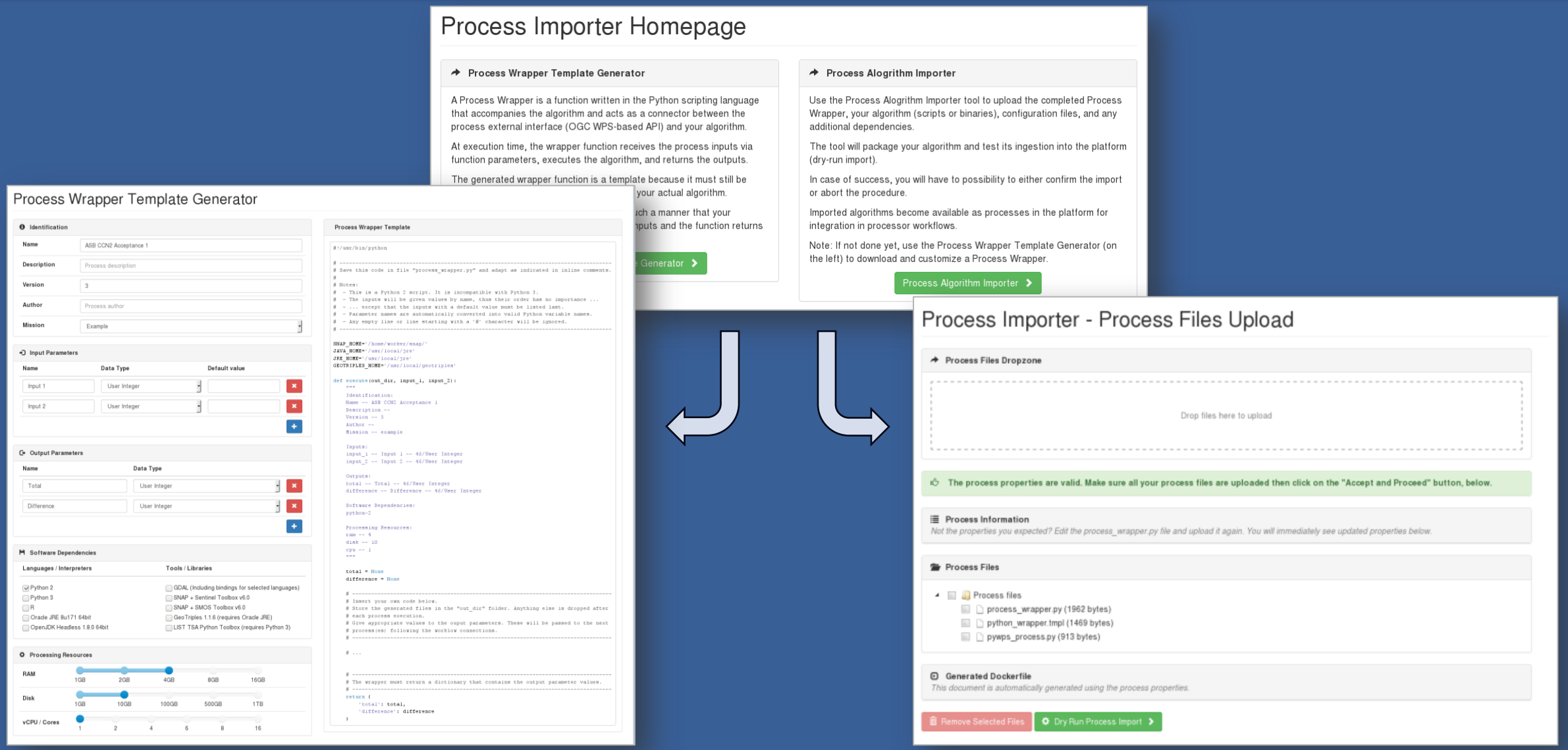


IMPORTING AN ALGORITHM AS A PROCESS

- A user has a processing chain description and the algorithms that implement the different steps of the chain.
- The processing chain description is used to add knowledge into the framework: data types, parameters, processes and processor/workflow definitions.
- The algorithms are packaged within Docker container images and are pushed in a Docker registry.



IMPORTING AN ALGORITHM AS A PROCESS



PROCESSOR WORKFLOW EDITING

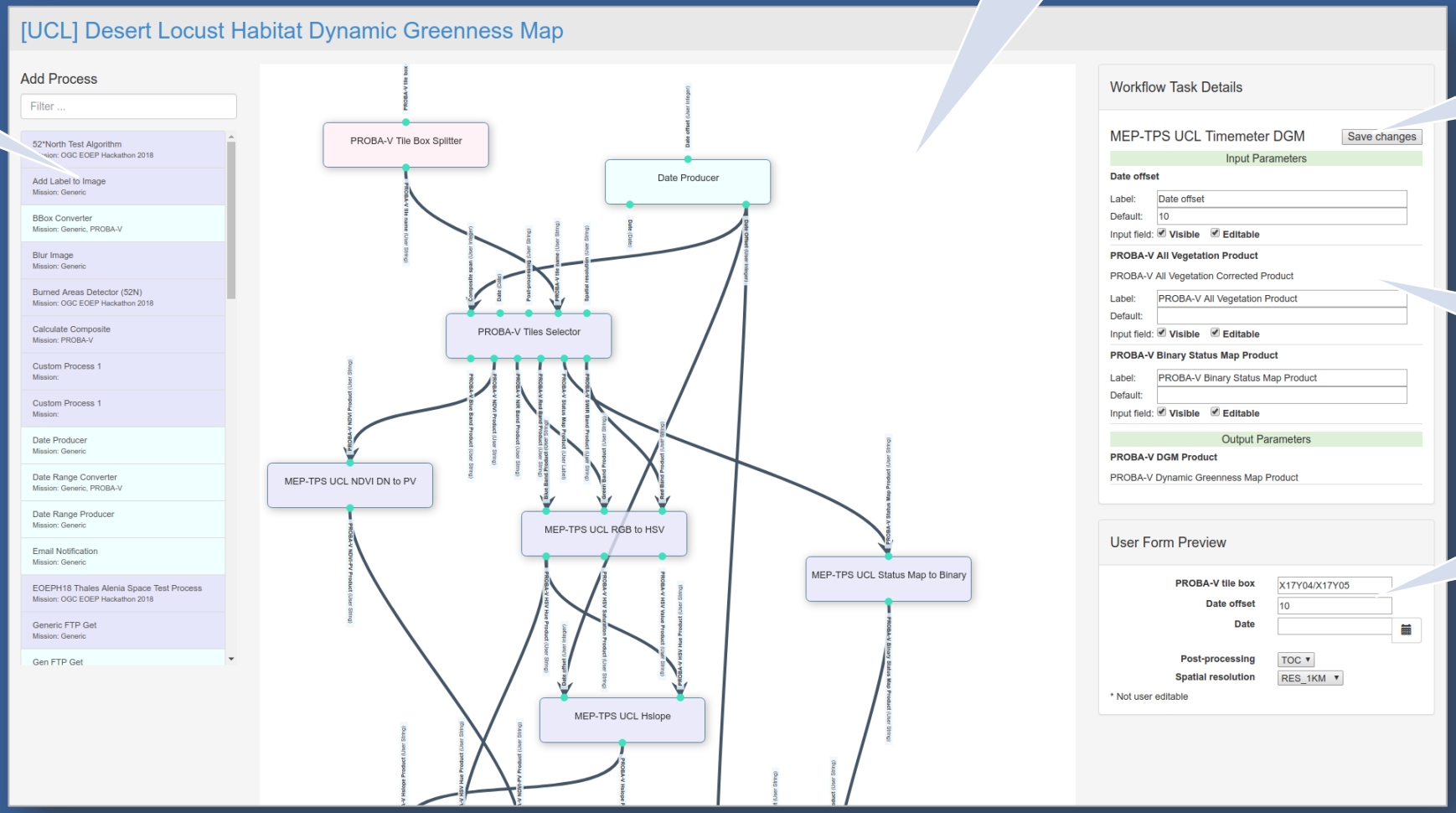
Available Processes

Workflow Drawing Canvas

Selected Process Instance Details

Customizable Input Properties

Parameterization Form Preview



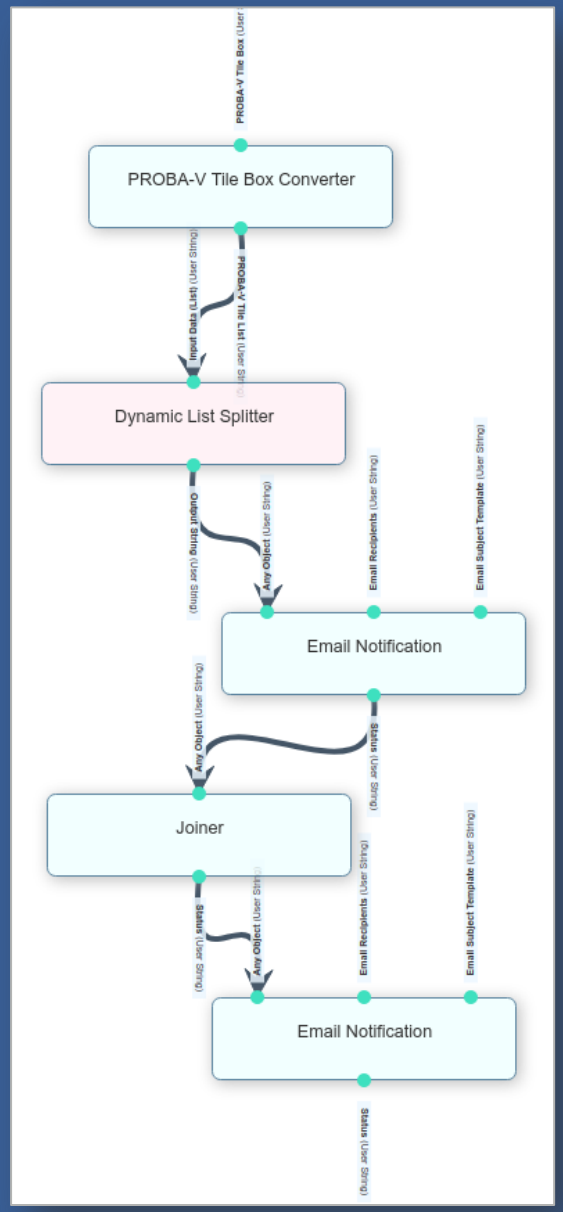
PARALLELISING LARGE COLLECTIONS OF DATA USING SPLITTER TASKS (1/2)

Dynamic Input List Splitter Task

- Receive a list of items as input parameter
- Execute the next tasks in the chain (workflow fragment) once for each item in the list
- Execute workflow fragments in parallel

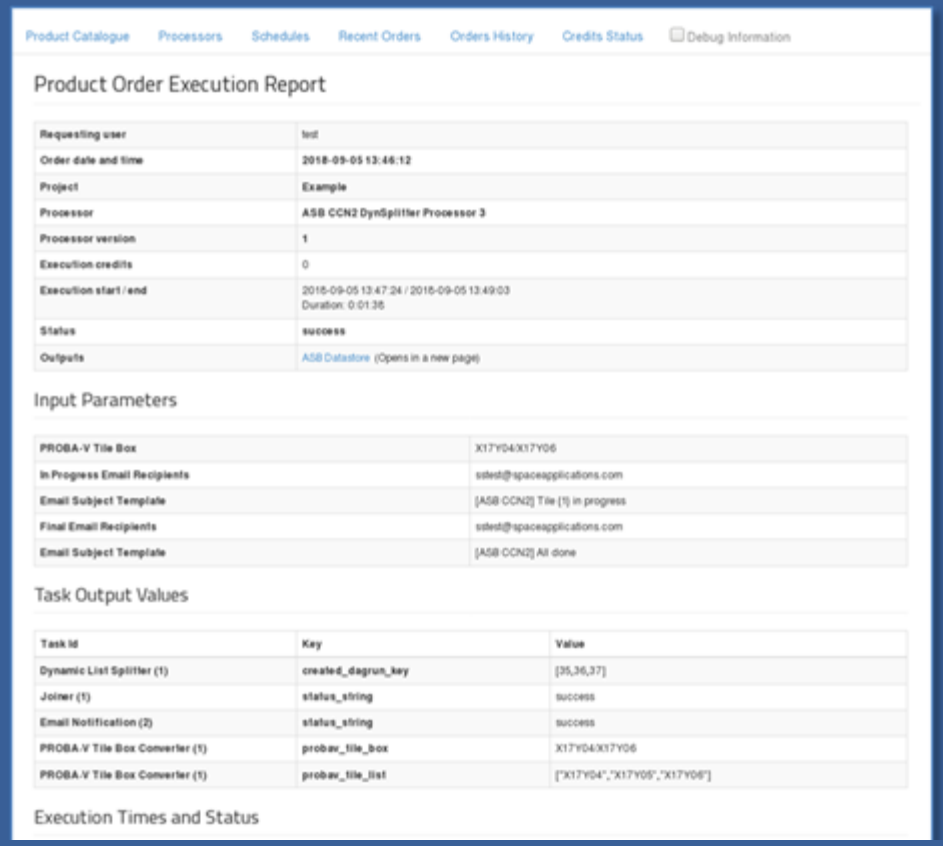
Enhancement

- Added support for dynamically fetched or generated lists
- Allows integrating a Dynamic List Splitter in-between the tasks in a workflow
- In this example:
  - The PROBA-V Tile Box Converter receives a tile box as input (e.g. "X16Y04/X20Y00") and outputs the corresponding list of tiles
  - The Dynamic List Splitter receives the tiles list and executes the next task (Email Notification) once on each tile
  - The Joiner task waits for all the parallel workflow fragments to complete
  - A final Email Notification is issued



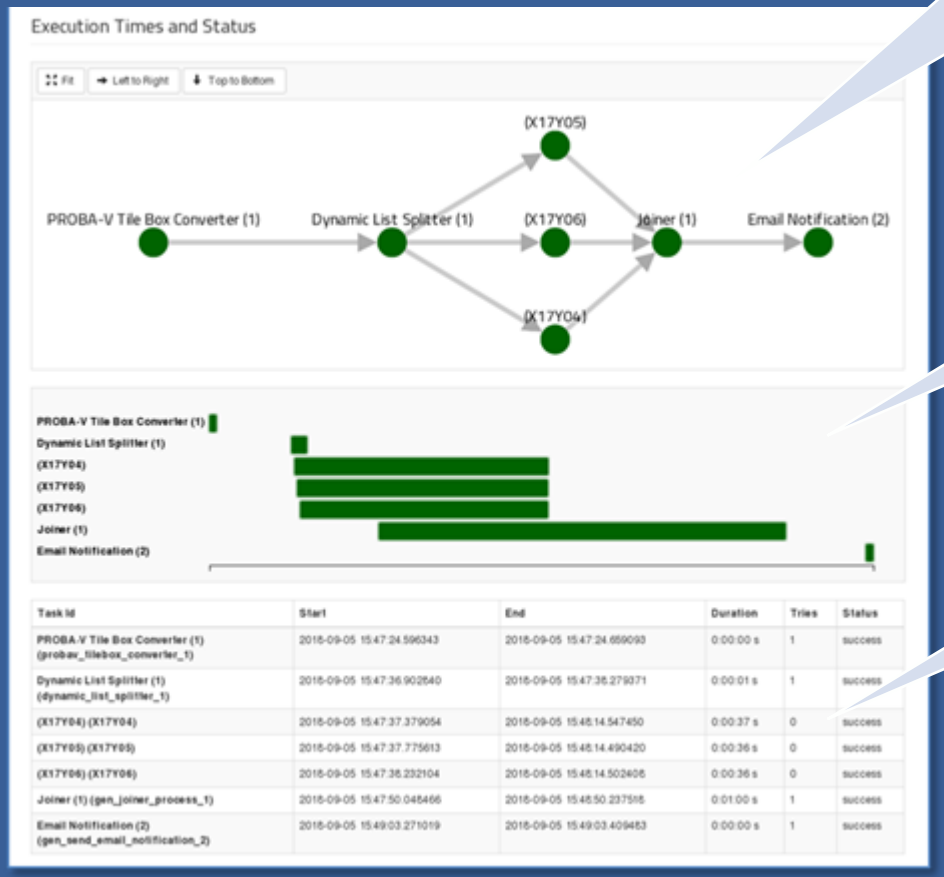
PARALLELISING LARGE COLLECTIONS OF DATA USING SPLITTER TASKS (2/2)

Execution Reports



Top

Bottom



Graphical representation of the workflow including the fragment instances

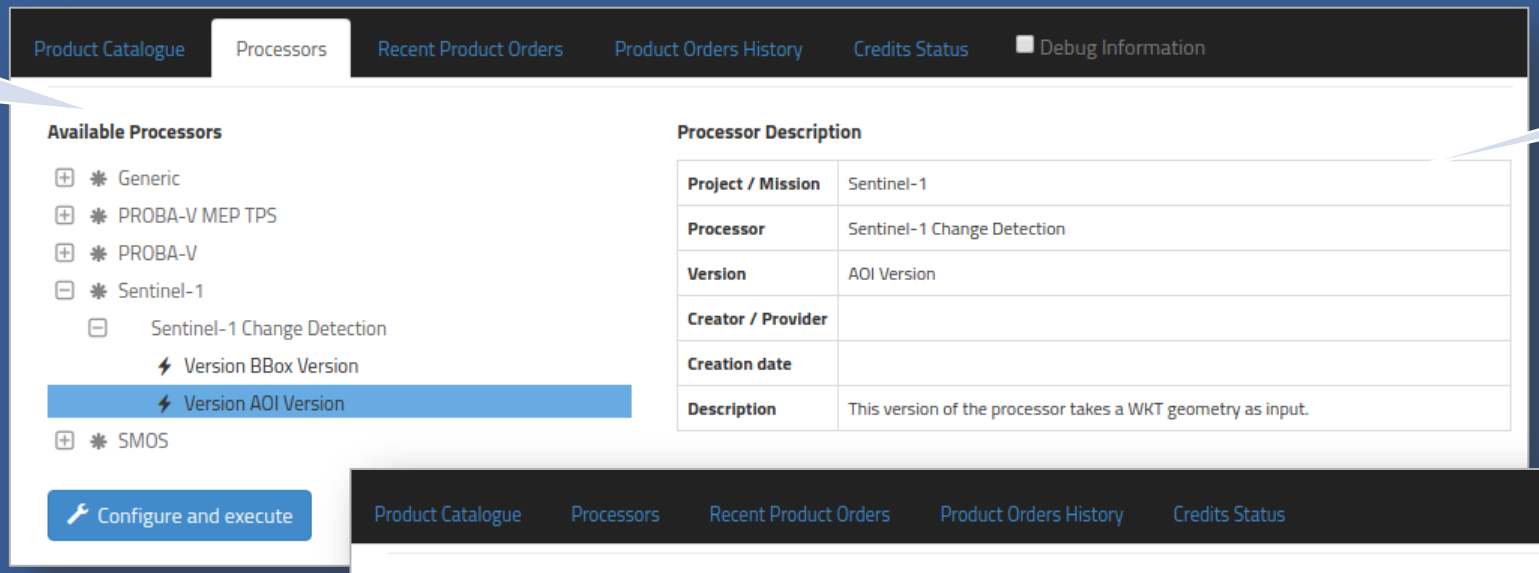
Gantt chart showing the start/end times

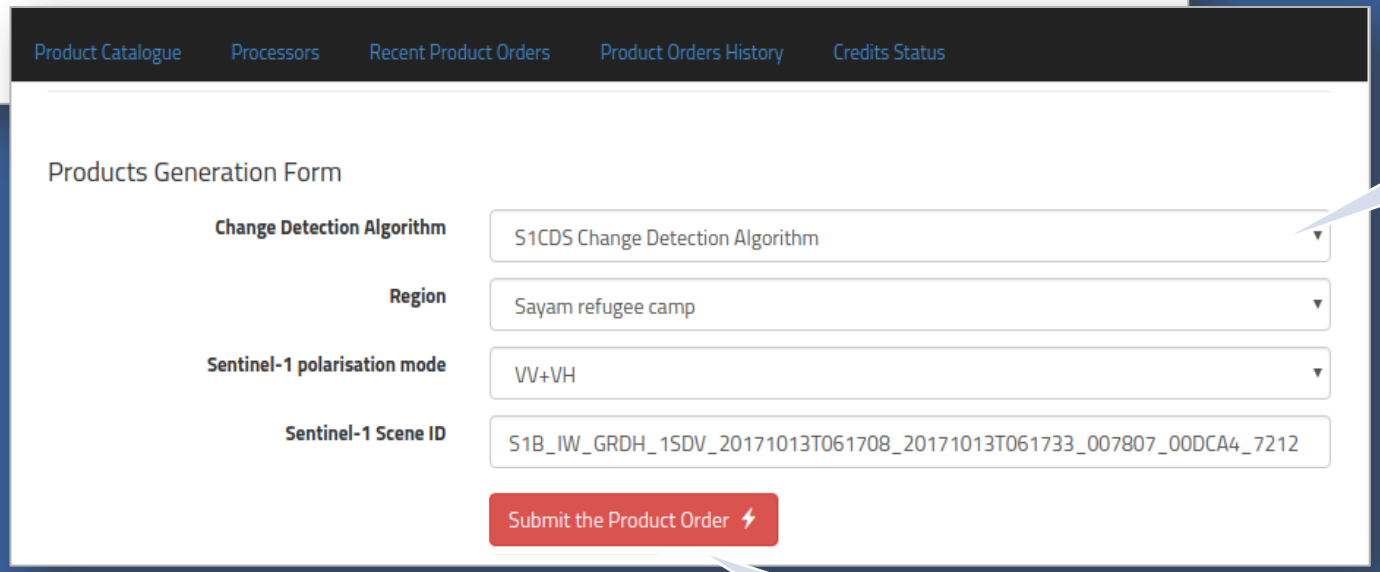
Detailed start/end times of the tasks and fragments

PROCESSOR SELECTION AND EXECUTION

Mission, Processor, Version Tree

Selected Processor Details





Parameterization Form

Submit Button

DEMONSTRATION INFRASTRUCTURE

- **Controller Node** (1)

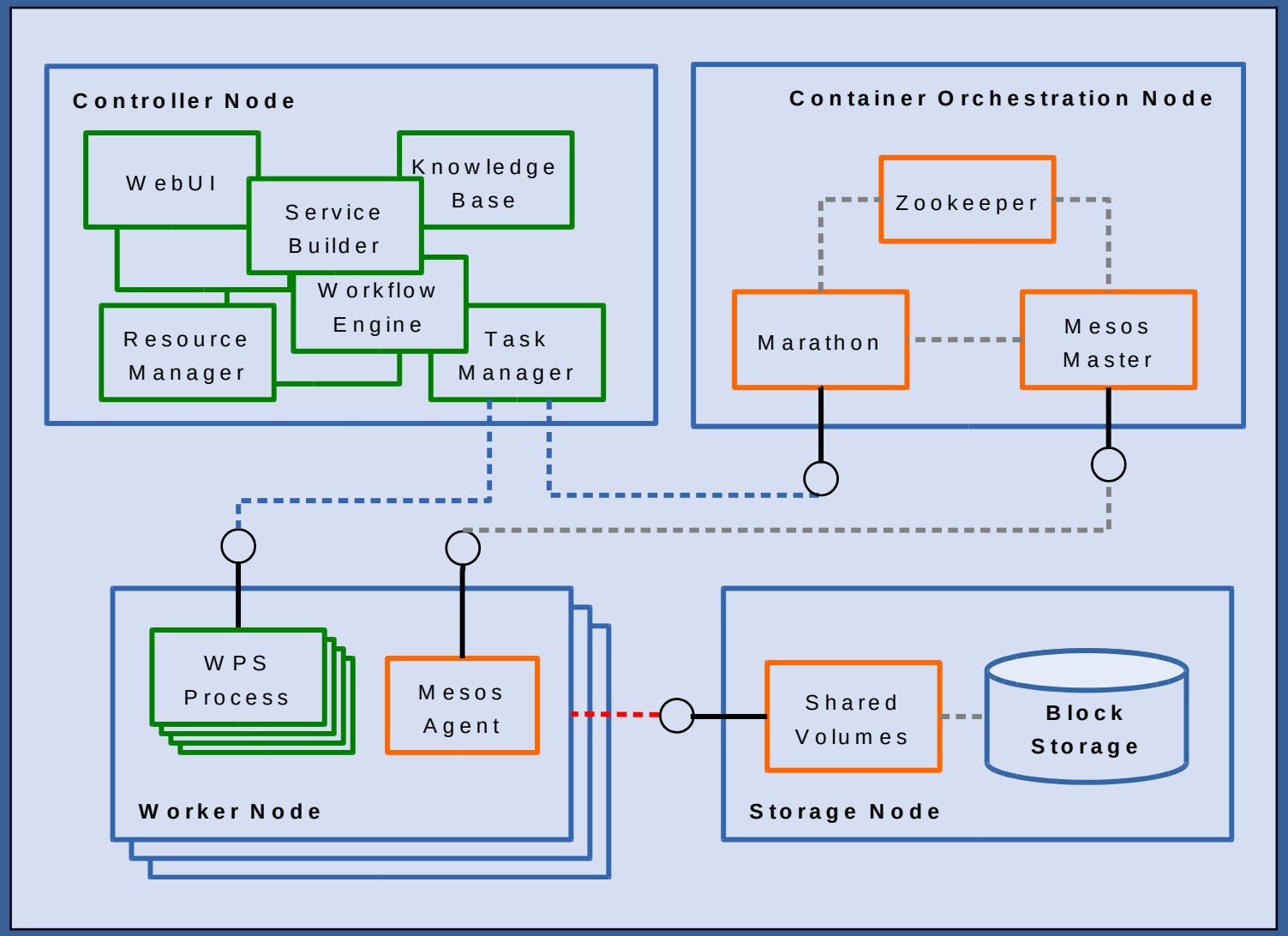
  ASB Core Components
- **Orchestration Node** (1)

  Deployment, execution, un-deployment of Docker Containers
- **Worker Nodes** (3)

  Execution of WPS Processes within the deployed Docker Containers
- **Storage Node** (1)

  Shared access and storage of data files
- **Docker Registry** (1, not represented)

Tested and demonstrated in public cloud environments: CloudSigma, OTC Cloud, Hetzner



DEMONSTRATION SCENARIO

1. **Workflow Editor**
   - Available processes, canvas, parameters customization, form preview
2. **Process Import Tool**
   - Process Wrapper Template Generator
   - Import a custom process wrapper with a call to an R script
3. **Workflow Editor**
   - Creation of a new Processor
   - Insertion of the new Process in the Processor Workflow
4. **End-User Interface**
   - Selection of the new Processor, parameterization, execution
   - Access to the Processor execution report

LIVE DEMONSTRATION

DELIVERABLE: ENHANCEMENTS SPECIFICATION TECHNICAL NOTE FOR ASB CCN2

Collects the material produced within the CCN2 development lifecycle:

1. Use Case Scenarios denoting the user needs
2. Technical Specifications derived from the use case scenarios
3. Updated architecture and design of the new features
4. Software verification and validation test plan, including acceptance test procedures
5. Acceptance test report
6. CCN2 features usage instructions (software user manual)
7. Demonstration plan
8. Software configuration file material (against the main ASB SCF)

Releases delivered at DDR and AR

DELIVERABLE: CONSOLIDATED EVALUATION REPORT

Demonstration Activities

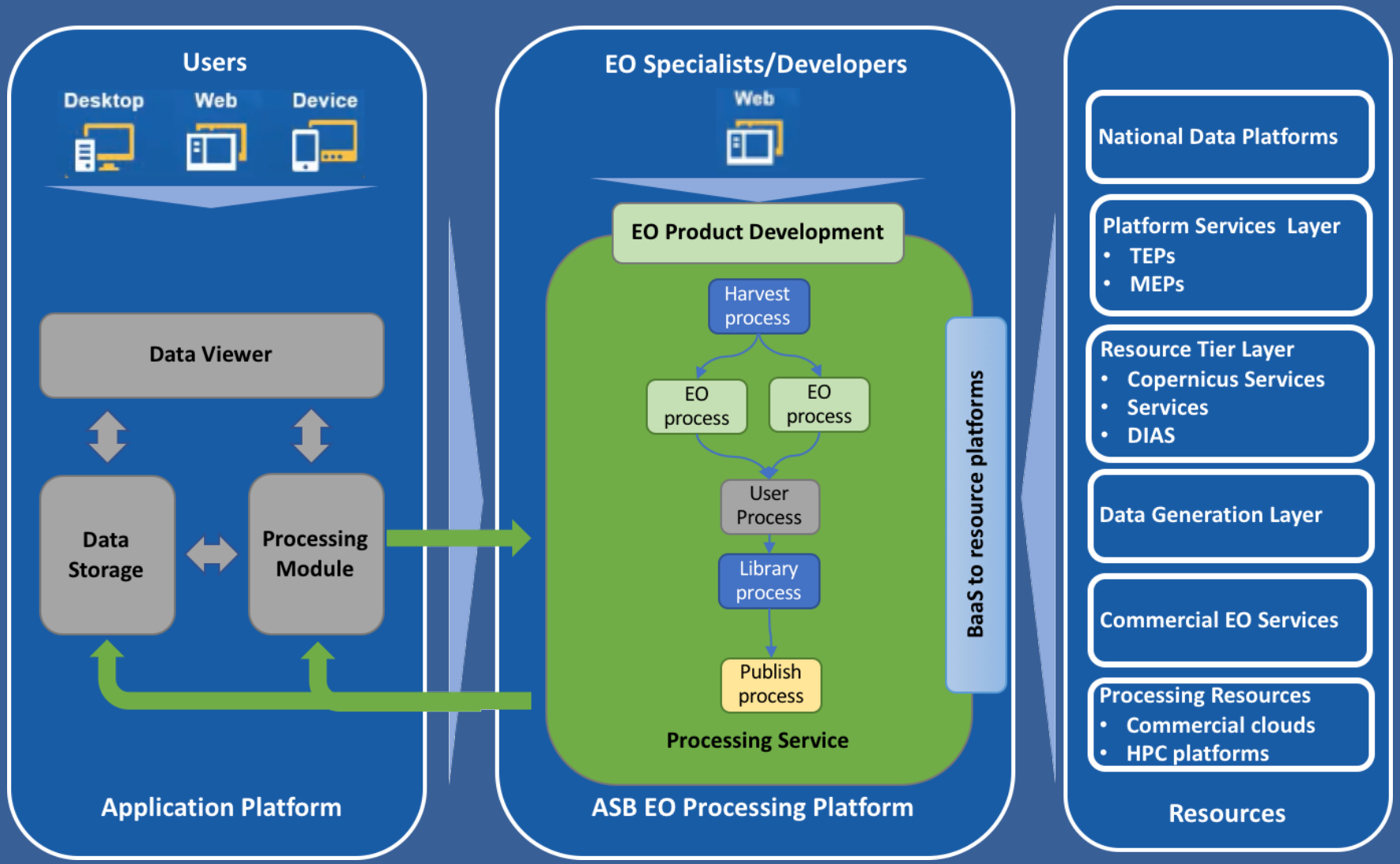
- The key capabilities of the ASB Framework have been presented to technical partners in EC's H2020 funded project EOPEN. This has happened during the first EOPEN Plenary Meeting organized in Athens on September 13 and 14, 2018.
- PROBA-V MEP TPS partner LIST has been invited to attend a demonstration of the new capabilities of the ASB Framework via a teleconference. In particular, it has been shown to LIST how R functions they have implemented in the PROBA-V Time Series Analysis (TSA) toolbox can be imported within ASB and chained within workflows.

Feedback collected in the Consolidated Evaluation Report, delivered before the Final Presentation

CONCLUDING REMARKS

- The goals of CCN2 have been achieved providing a framework for processing complex processing chains for on-demand and automated (unsupervised) processing services.
- We are seeing, after a lot of effort, a pick-up on ASB. There are misconceptions to overcome.
- The ASB Framework has a achieved a major goal of being agnostic to the platform. We have demonstrated that you can easily switch from one platform to another without changing the implemented processing services.
- Low code, no IT knowledge needed to import user modules.
- Users find it easy to run processing services using dynamically generated forms.
- ASB is reaching project status and is being used in services for SatCen and the H2020 EOPEN.
- In EOPEN application users will use processing services transparently from their GIS environment.

FUTURE – EOPEN CONFIGURATION



FUTURE WORK

- Tackle the user "wish list"
  - bulk imports to build a library of modules (LIST PROBA-V Toolbox, SNAP toolboxes…)**
  - improved monitoring and control of process executions
  - segmenting long processing chains for optimizing the packaging and the executions
- Linking or embedding such tools as Jupyter Notebook seems to be the next step to support processing chain development making development work truly seamless
- Document concerns on performance and address them by performing a PDGS IPF realistic example
- Extend with new built-in generic processes in support of building processing chains
- The ASB Framework is ready for use in operational systems application to compare capabilities of ASB to those needed for new missions for PDGS IPFs
- Investigate over the ASB Framework potential with the needs of the Common Architecture
- Demonstrate a use of the Network of EO Resources

** Already investigated direct calls to SNAP functions and use in a processing chain.

Thank you

Bernard Valentin
Bernard.valentin@spaceapplications.com

Leslie Gale
Leslie.gale@spaceapplications.com