This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 777533.

PROviding Computing solutions for ExaScale ChallengeS

D8.2	Validation Report of PROCESS production prototype		
Project:	PROCESS H2020 – 777533	**Start / Duration:**	01 November 2017 36 Months
Dissemination[1]:	Public	**Nature[2]:**	**R**
Due Date:	31 July 2020	**Work Package:**	**WP 8**
Filename[3]	PROCESS_D8.2_ValidationReportPROCESSProduction Prototype_v1.0.docx		

ABSTRACT

This deliverable describes the validation of the PROCESS production prototype which is described in detail in D8.3. This deliverable is the successor to D4.5 in which the 2nd prototype was validated, which lead to an update of the use case requirements and PROCESS architecture (described in D2.3) and performance model (described in D3.3).

[1] PU = Public; CO = Confidential, only for members of the Consortium (including the EC services).
[2] R = Report; R+O = Report plus Other. Note: all "O" deliverables must be accompanied by a deliverable report.
[3] eg DX.Y_name to the deliverable_v0xx. v1 corresponds to the final release submitted to the EC.

Deliverable Contributors:	Name	Organization	Role / Title
Deliverable Leader[4]	Maassen, J.	NLESC	Deliverable coordinator
Contributing Authors[5]	Madougou, S.	NLESC	Writer
	Graziani, M	HES-SO	Writer
	Balázs. S.	LSY	Writer
	Schmidt, J.	LMU	Writer
Reviewer(s)[6]	Belloum, A.	UvA	Reviewer
	Bobák, M.	UISAV	Reviewer
Final review and approval	Höb, M.	LMU	Coordinator

Document History

Release	Date	Reasons for Change	Status[7]	Distribution
0.1	18.05.2020	Initial structure of document	Draft	Workgroup
0.2	11.06.2020	Abstract + introduction added	Draft	Workgroup
0.3	10.07.2020	Performance section complete	Draft	Workgroup
0.4	13.07.2020	Version ready for review	Draft	Consortium
0.5	17.07.2020	Processed review	Draft	Consortium
0.6	23.07.2020	Processed second review	Draft	Consortium
1.0	28.07.2020	Final Version	Released	Public

4 Person from the lead beneficiary that is responsible for the deliverable.
5 Person(s) from contributing partners for the deliverable.
6 Typically, person(s) with appropriate expertise to assess the deliverable quality.
7 Status = “Draft”; “In Review”; “Released”.

Table of Contents

Executive Summary

In this deliverable we validate the first production prototype of the PROCESS services by checking if these services meet the functional and performance requirements of the use cases (UCs), as described in D2.3. This production prototype is released simultaneously in deliverable D8.3. We will focus on the open issues that have been resolved since D4.5 and identify if there are any open issues left which need resolving before the final release of the services in D8.4.

Our validation shows that the first production prototype meets most of the functional UC requirements. Only UC#1 and UC#2 have a small number of open issues which have not yet been completely fulfilled. For UC#1, supporting distributed learning using OpenMPI and Hovorod has proven challenging, as the test infrastructure currently does not support the combination of singularity containers with OpenMPI. In addition, the GPUs available in the test environment are too outdated to efficiently run the use case. Effort are ongoing to solve both issues. For UC#2, the integration between the use case specific user interface and the IEE is progressing, but not completely finished.

When validating the performance of the PROCESS services, our platform model indicates that the overhead introduced is minimal and does not lead to any significant bottlenecks when running the use case applications, provided that sufficient resources are available. However, the scheduling model does indicate that the local scheduling system used from the PROCESS services may not scale to the number of machines required for exascale supercomputers. As this is a general problem faced by all vendors and users of such machines, we expect these issues to be solved once more exascale size machines become available.

The lack of DTN support in the PROCESS testbed is a problem, as the current internet based data transfers do not provide the necessary performance needed to transfer the large data volumes needed for UC#2, even after further optimization of the services. Our experiments and data transfer model clearly show that the current long-range network connections used by PROCESS to transfer data between sites are insufficient to support the data transfers volumes necessary. These results clearly illustrate the need for an efficient DTN network between European compute centres. To resolve this issue, we have been in contact with SURFnet (the Dutch NREN), Geant (the European e-Infrastructure consortium) and the various administrators of the PROCESS testbed sites in an attempt to set up an experimental DTN infrastructure for PROCESS. Unfortunately, we did not manage to convince all PROCESS site administrators to support us in this. At the moment it seems unlikely that we will be able to do so within the lifetime of this project.

List of Figures

List of Tables

1 Introduction

Deliverable D4.5 (released in M27), described the validation of the use cases (UCs) against the 2nd service prototype and presented the final PROCESS architecture. At the time, both the services and UC codes were still under development, and therefore only preliminary validation results were presented. This validation lead to an update of the use case requirements and PROCESS architecture, described in detail in D2.3 (released in M30). The services were then adapted to satisfy the new requirements, and the performance model was updated, as described in D3.3 (released in M32).

In D8.3 (released simultaneously with this deliverable), we release the first packaged versions of the PROCESS services. Below, we will extend the validation performed in D4.5 to this first packaged release. The main goal of this validation is to ensure that the PROCESS solutions meet the requirements of the use cases and user communities, notably by deploying and running the UC codes on the PROCESS platform.

As in D4.5 we validate the production prototype by checking whether all functional requirements of the UCs have been met in this production prototype and by assessing the scalability and overhead of the PROCESS platform with the performance models described in D3.3. The technology readiness level (TRL) of each of the components and their interactions will be evaluated in D8.3.

We will focus on the open issues that have been resolved since D4.5 and identify if there are any open issues left which need resolving before the final release of the services in D8.4.

2 Validation of functional use case requirements

As of D4.5, most PROCESS components and services were integrated and running as part of the 2nd platform prototype. The requirements for use case 3 and 5 had been fully met, while use cases 1, 2, and 4 had a small number of requirements still open.

In this section, we look back to the unmet requirements of use cases 1, 2 and 4 in the second prototype and check whether they are now met in the first production prototype.

2.1 Use case 1

We report in Table 1 the unmet requirements for use case 1 as reported in D4.5 (page 7) and D2.3 (page 6) and report their status in the production prototype.

Requirement	Motivation	Status in D4.5/D2.3	Status in production prototype
Support GPU usage in containers	Required for efficient single node training	In progress	Fulfilled
Support of distributed libraries for parallel GPU training: OpenMPI and Horovod	Required for distributed training using Tensorflow and Keras	In progress	In progress

Table 1: Use case 1 requirement status.

In the following, we report on the status of these requirements in the production prototype and we give details on their validation and results:

- The support for GPU usage in containers is currently being addressed. The production prototype will be comprehensive of the network training module (CamNet software layer II) with GPU access for training and integration with the IEE. The development of a Layer III container which will support network interpretability and visualization is still in progress.
- The support of distributed libraries for parallel GPU training is a major technical challenge. Hovorod requires OpenMPI to run, but unfortunately the integration of OpenMPI and Singularity is currently not supported on the HPC infrastructure site. In addition, the GPUs available at the site are too outdated to efficiently run the use case. Effort are ongoing to solve both issues. The distributed learning solution is implemented, however, on the local site of HESSO, where the need for Singularity containers can be easily bypassed.

2.2 Use case 2

We report in Table 2 the unmet requirements for use case 2 as reported in D4.5 (pages 10 and 11) and D2.3 (page 6 and 7) and report their status in the production prototype.

Requirement	**Motivation**	**Status in D4.5/D2.3**	**Status in production prototype**
UI requirements	A use case specific UI is used based on an EOSC-hub. Backend integration with IEE is needed to access the full PROCESS services	Fulfilled. Integration with IEE is in progress	In progress
Data staging and transfer	Needed to transfer data from the archive in Amsterdam to the compute sites	Fulfilled but optimization needed	Fulfilled
DTN infra	Needed to improve the efficiency of data transfer	Not available	Dropped
Multisite execution	Needed to enable processing of independent datasets on multiple sites	In progress	In progress

Table 2: Use case 2 requirement status

In the following, we report on the status of these requirements in the production prototype and we give details on their validation and results:

- UI requirements: These where met in D4.5 by extending the existing EOSC-hub LOFAR Web UI. The integration of this UI with the IEE through a REST API is being implemented and tested.
- Data staging and transfer: This requirement was fulfilled since D4.5. Further optimization was performed to improve the data transfer performance. However, to reach an acceptable data transfer rate for the large data volumes needed by UC#2, our measurements indicated that a dedicated DTN infrastructure would be most suitable (as described in the next requirement).
- DTN infra: Dedicated data transfer nodes connected to high-bandwidth long-distance network infrastructure is needed to transfer the large data volumes needed by UC#2. Unfortunately, our proposal to arrange this was not accepted by all site admins of PROCESS consortium. We have therefore dropped this requirement, as it is unlikely to be fulfilled during the scope of the project and was not a requirement originally envisioned at the start of the PROCES project. Other transfer optimisations suitable for regular internet connections have been implemented instead.
- Multisite execution: This is already possible, as the user can select different computing sites for different pipeline executions. However, currently one cannot select different datasets in the IEE frontend. This selection is possible from UC#2 frontend but the integration with IEE is not fully completed yet.

2.3 Use case 4

We report in Table 3 the unmet requirements for use case 4 as reported in D2.3 (page 9 and 12) and report their status in the production prototype.

Requirement	**Motivation**	**Status in D2.3**	**Status in production prototype**
Provide support for HDFS	Required to access local storage infrastructure	In progress	Fulfilled

Table 3: Use case 4 requirement status.

In the following, we report on the status of these requirements in the production prototype and we give details on their validation and results:

- Provide support for HDFS: The support was provided by the underlying infrastructure in the UiSAV testbed. The application container was modified accordingly to be able to access this HDFS file system. The data for training the models are read from HDFS into data frames.

2.4 Use case 5

As stated in D4.5, all requirements for use case 5 have been met.

3 Validation of platform performance

In deliverable D3.1, D3.2 and D3.3, extensive measurements were performed based on micro benchmarks and use case scenarios to evaluate the performance of the PROCESS platform and to develop a performance model suitable to extrapolate its performance to exascale applications. As explained in D3.3, some variables in our performance model are determined by external factors, such as the scheduling delay at the PROCESS compute sites, long-distance networking performance between the PROCESS sites when staging data, or individual performance of the use case applications.

For that reason, three models have been built: one for PROCESS platform overhead, one for the scheduling overhead, and one for data staging and transfer. The platform overhead model will be used to validate the choices made in PROCESS architecture by allowing us to estimate the overhead at the exascale level. The scheduling overhead and data staging and transfer models will be used to identify external bottlenecks which are beyond the direct control of PROCESS. Below we will use these models to validate the performance of the PROCESS architecture, up to 100K containers and 1 exabyte of data. For details we refer to Section 5 in D3.3 (pages 28-30).

3.1 Platform overhead

In Figure 1(a), we recall the model output for the platform overhead model built from our measurements and described in detail in D3.3 (page 34). We use this model to extrapolate what would be this overhead as we approach an exascale size machine. As of June 2020, the #1 supercomputer in the Top500[8] runs at approximately 0.4 exaflops using an estimated 100K nodes. Figure 1(b) therefore shows the extrapolation of our platform overhead when running 100K containers. This figure is using log scale due to the high container count. As the figure shows, with 100K containers the overhead is estimated to be about 1200 seconds or 20 minutes. This is small when we put it in perspective with the large container count and often long execution times per container (in the order of days for UC#2).

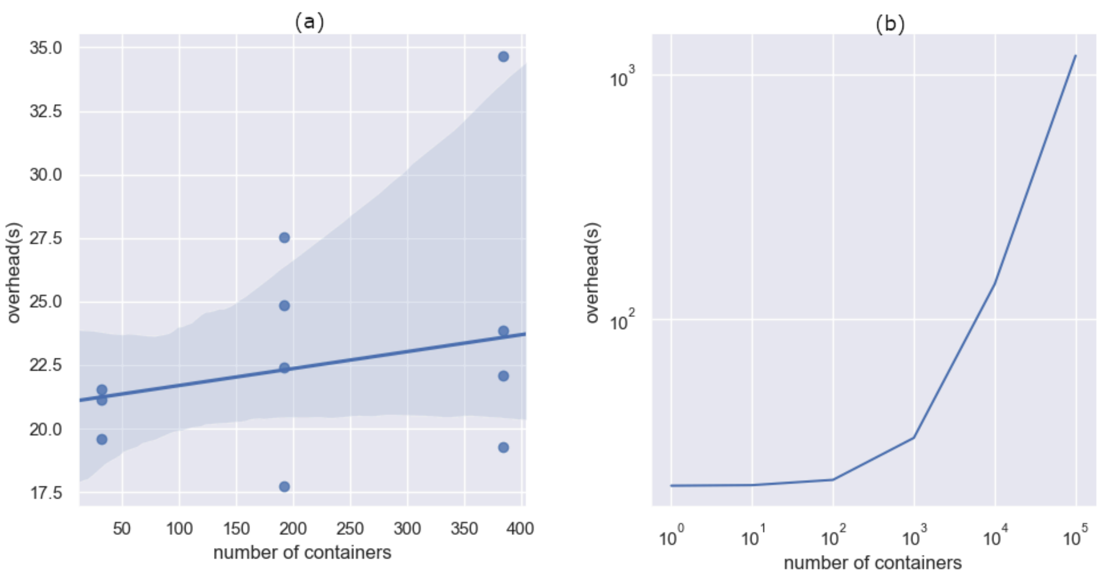

Figure 1: Platform overhead estimation at exascale machine size

8 https://www.top500.org/lists/top500/2020/06/

3.2 Scheduling overhead

Similarly, to the platform overhead, a model for the scheduling overhead is described in D3.3 (page 35). Figure 2(a) and Figure 2(b) show scheduling overhead estimations produced by this model. This overhead is caused by the local scheduling infrastructure on the machine (such as SLURM) and therefore external to the PROCESS services.

The projections obtained from the model for 100K containers is shown in Figure 2(b) at log scale. In contrast to the platform overhead, the scheduling overhead is estimated to be quite large at very high container count. It is in the same order as the container count. Therefore, we expect that exascale data processing on such large number of nodes will present some challenges to current scheduling techniques and to require new approaches to scheduling.

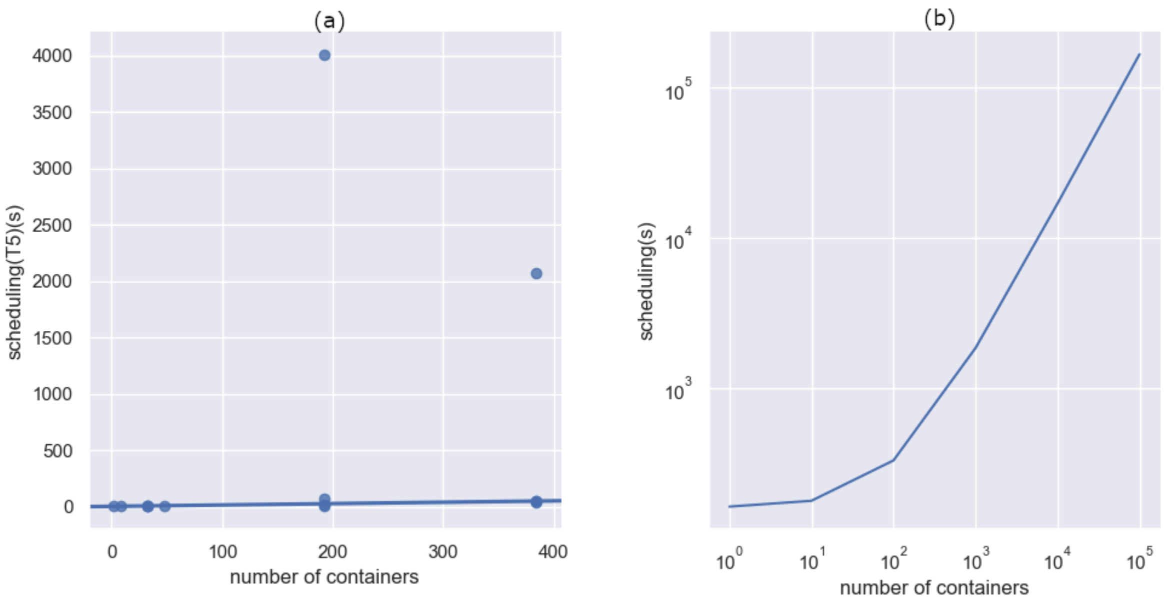

Figure 2: Scheduling overhead estimation at the exascale level

3.3 Data transfer overhead

In deliverable D3.3, we showed that the staging (in and out) is independent of the number of containers (page 36), but instead directly depends on the data size involved and performance of the network infrastructure available between test locations. The linear model of the relationship is illustrated in Figure 3(a), shown below, for a small stage-in in the context of IEE.

Using this model, hypothetical transfer times are projected for sizes up to an exabyte. As expected, because of the very low transfer rates in our testing environment, the stage-in time projections are extremely large for large datasets. For example, the stage-in of the 7 PB of data currently in LOFAR LTA would take about 7 years. Transferring a full exabyte would take several centuries. To solve these data transfer issues a dedicated Data Transfer Network (DTN) infrastructure would be needed. Experiments by Geant in 2018[9] have shown that 100Gbit DTN networks are feasible on a European and even global scale. At such transfer rates it would take less than a minute to transfer a single 16TB LOFAR observation.

---

[9] https://www.slideshare.net/JISC/data-transfer-experiments-over-100g-paths

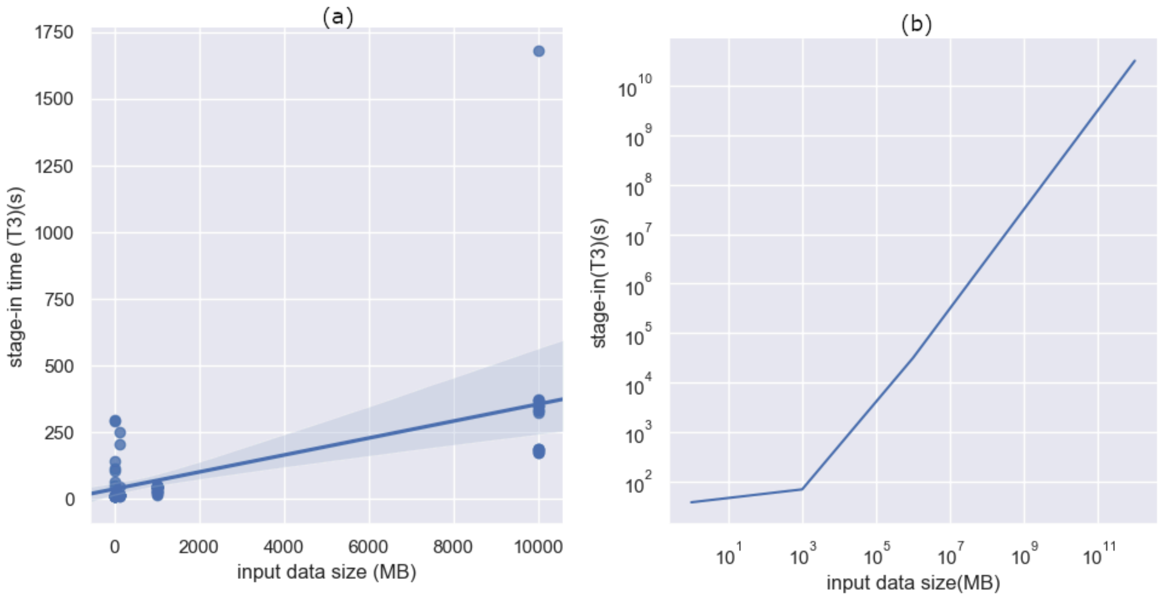

Figure 3: Data transfer estimation at the exabyte level

4 Conclusion

The first production prototype of the PROCESS services can meet most of the use case requirements. Only use cases 1 and 2 have a small number of open issues which have not yet been completely fulfilled. For use case 1, supporting distributed learning using OpenMPI and Hovorod has proven challenging, as the test infrastructure does currently not support this. For use case 2, the integration between the use case specific user interface and the IEE is progressing, but not completely finished. The lack of DTN support in the PROCESS testbed is a problem, as the current internet based data transfers do not provide the necessary performance needed to transfer the large data volumes required or UC#2, even after further optimization of the services.

When validating the performance of the PROCESS services, our platform model indicates that the overhead introduced is minimal and does not lead to any significant bottlenecks when running the use case applications, provided that sufficient resources are available. However, the scheduling model does indicate that the local scheduling system used from the PROCESS services may not scale to the number of machines required for exascale supercomputers. As this is a general problem faced by all vendors and users of such machines, we expect these issues to be solved once more exascale size machines become available.

Our data transfer model clearly shows that the current long-range network connections used by PROCESS to transfer data between sites are insufficient to support the data transfers volumes necessary for some of our use cases. Both our model results and UC#2 experiments clearly illustrate the need for an efficient DTN network between European compute centres. We have been in contact with SURFnet (the Dutch NREN), Geant (the European e-Infrastructure consortium) and the various administrators of the PROCESS testbed sites in an attempt to set up an experimental DTN infrastructure for our PROCESS use case experiments. Unfortunately, we did not manage to convince all PROCESS site administrators to support us in this. At the moment it seems unlikely that we will be able to do so within the lifetime of this project.