

### HPC-Cloud-Big Data Workflows and Research Data Management: The LEXIS Approach

M. Hayek<sup>(1)</sup>, M. Golasowski<sup>(2)</sup>, K. Slaninová<sup>(2)</sup>, M. Levrier<sup>(3)</sup>, A. Scionti<sup>(4)</sup>, D. Magarielli<sup>(5)</sup>, S. Louise<sup>(6)</sup>, Antonio Parodi<sup>(7)</sup>, P. Harsh<sup>(8)</sup>, F.I. Apopei<sup>(9)</sup>, E. Danováro<sup>(10)</sup>, M. Derquennes<sup>(11)</sup>, F. Donnat<sup>(12)</sup>, L. Ganne<sup>(3)</sup>, R.J. García-Hernández<sup>(1)</sup>, S. Hachinger<sup>(1)</sup>, J. Munke<sup>(1)</sup>, Andrea Parodi<sup>(7)</sup>, G. Vitali<sup>(4)</sup>, P. Viviani<sup>(4)</sup>, O. Terzo<sup>(4)</sup>, J. Martinovič<sup>(2)</sup>

(1) Leibniz Supercomputing Centre, Garching b. M. (D); (2) IT4Innovations, VSB – Technical University of Ostrava, Ostrava (CZ); (3) ATOS, Paris (F); (4) Advanced Computing and Applications, LINKS Foundation, Turin (IT); (5) Avio Aero, Rivalta di T. (IT); (6) CEA LIST, Gif-sur-Yvette (F); (7) CIMA Foundation, Savona (IT); (8) Cyclops Labs, Zurich (CH); (9) Teseo, Turin (IT); (10) ECMWF, Reading (UK); (11) EURAXENT, Cambridge(UK); (12) Outpost24, Antibes (F)



This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825532.

START DATE  
January 1st, 2019

DURATION  
36 months

Funding  
€12.2 M

## LEXIS DISTRIBUTED DATA INFRASTRUCTURE (DDI) AND ORCHESTRATION

### OBJECTIVES

- Provide researchers and industry with a user friendly platform to execute optimized simulation workflows on Cloud/HPC systems.
- Federate internationally competitive supercomputing & data centers (currently IT4I/CZ, LRZ/DE, ECMWF/UK, ICHEC/IRL), taking advantage of advanced orchestration and data management techniques.
- Efficient data management in an European HPC/Cloud Context: FAIR<sup>[1]</sup> Research Data (Findable, Accessible, Interoperable, Reproducible).

### ORCHESTRATION

- The LEXIS Orchestration System orchestrates Hybrid HPC/Cloud applications on the distributed LEXIS platform (cf. Figure 1).
- Applications to be deployed are modeled using TOSCA<sup>[2]</sup> (Topology and Orchestration Specification for Cloud Applications).
- The orchestrator back-end is based on the Yorc solution<sup>[3]</sup>, and the front-end is based on Alien4Cloud<sup>[4]</sup>, developed by Atos.
- A plugin extending the orchestrator to support the management of HPC resources through the middleware HEAppE<sup>[5]</sup>, developed by IT4I, is used.

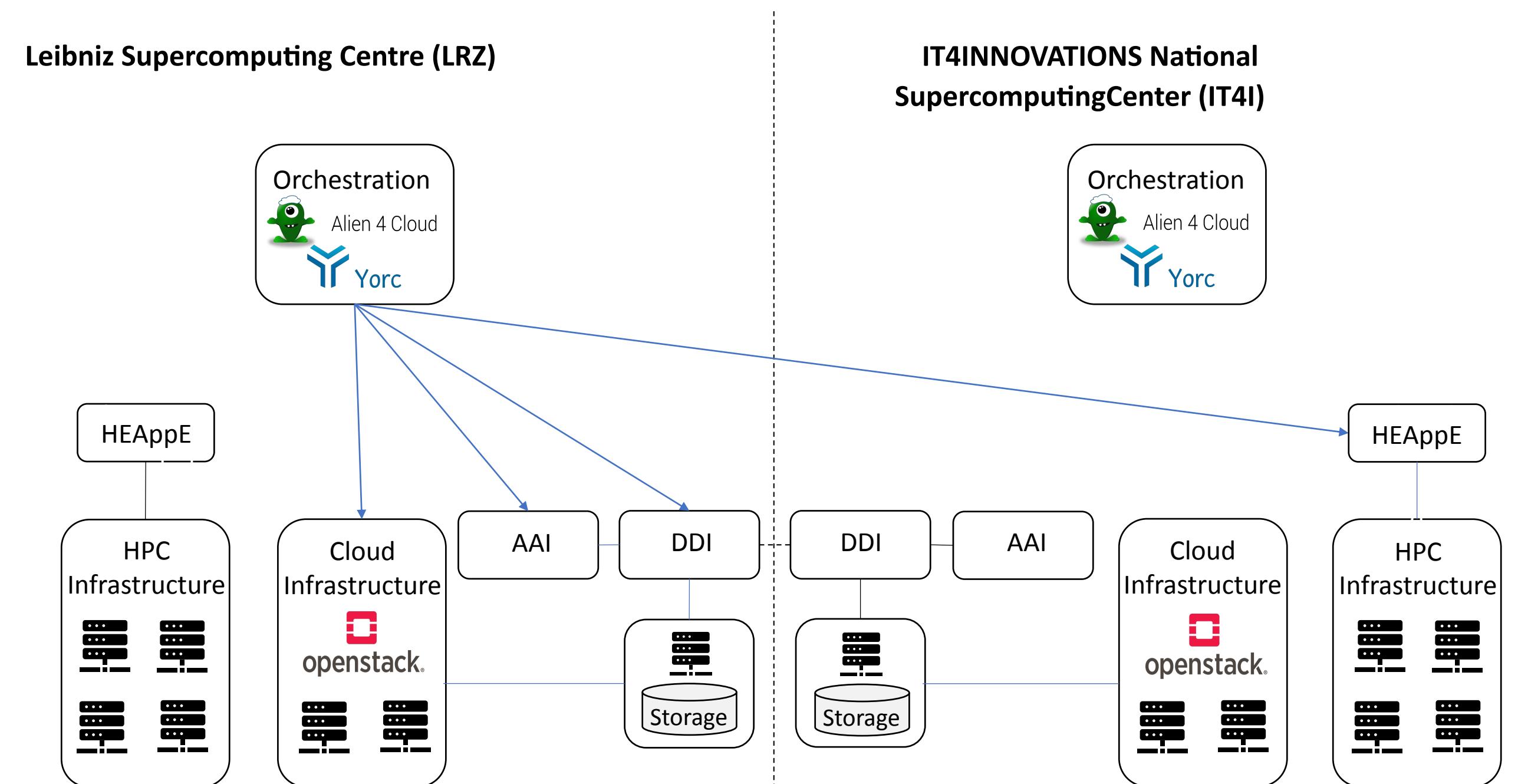


Figure 1: Usage of Distributed LEXIS infrastructure by applications; workflows can selectively or redundantly use Cloud/HPC/Data components (blue arrows for an example workflow, cf. [6]).

## DISTRIBUTED DATA INFRASTRUCTURE

- As part of the LEXIS platform, a federated iRODS<sup>[7]</sup> (Integrated Rule-Oriented Data System) was deployed. It provides unified access to files on different backend systems distributed at LRZ and IT4I.
- On top of iRODS, EUDAT<sup>[8]</sup> B2HANDLE and B2SAFE were deployed for Persistent Identifier (PID) management and data replication. EUDAT B2STAGE is used for data transfer to HPC and GridFTP systems.
- Django/Swagger-based RESTful APIs were developed for data staging between LEXIS storage systems, and for metadata assignment and listing, in order to adhere to the FAIR principles.
- The DDI is connected to the LEXIS Authentication and Authorization Infrastructure (AAI) via OpenID Connect<sup>[9]</sup> and provides appropriate access management.



## LEXIS DATA STAGING API AND BENCHMARKING OF THE LEXIS DDI DATA STAGING REST API

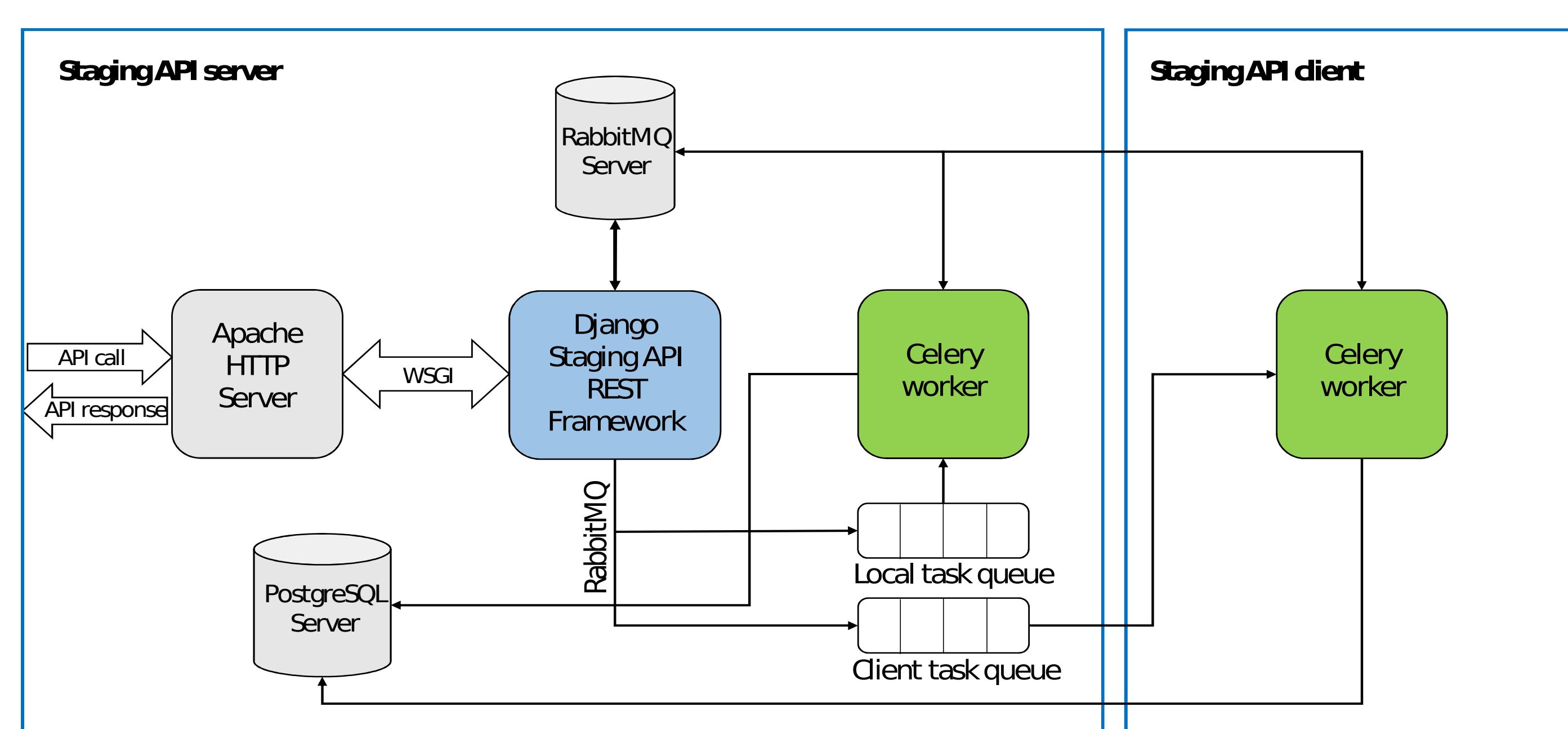


Figure 2: Architecture of the Data Staging API.

- Within LEXIS Workflows, the orchestrator needs to move data between source/target systems (DDI, Cloud and HPC file systems). To this purpose, it gives the Staging API simple dataset/source/target specs.
- The Staging API uses the LEXIS AAI and HEAppE to authenticate users' requests for access to the needed data and computing infrastructures.
- The API is asynchronous and uses a queueing system with Celery<sup>[10]</sup> and RabbitMQ<sup>[11]</sup> to handle tasks. A status query endpoint is provided.
- Different requests are organized into 8 different classes based on the source/target combination. This decides the queue to handle the request and the staging mechanism to be used (SCP, GridFTP, ...).

### REFERENCES

- [1] Wilkinson M.D. et.al., 2016, Scientific Data, 3, 160018 [2] Topology and Orchestration Spec. for Cloud Applications, 2013, docs.oasis-open.org/tosca/TOSCA [3] yorc.readthedocs.io [4] alien4cloud.github.io [5] heappe.eu [6] sc20.supercomputing.org/proceedings/tech\_poster/tech\_poster\_pages/rpost120.html [7] irods.org [8] eudat.eu [9] openid.net/specs/openid-connect-core-1\_0.html [10] www.celeryproject.org [11] www.rabbitmq.com

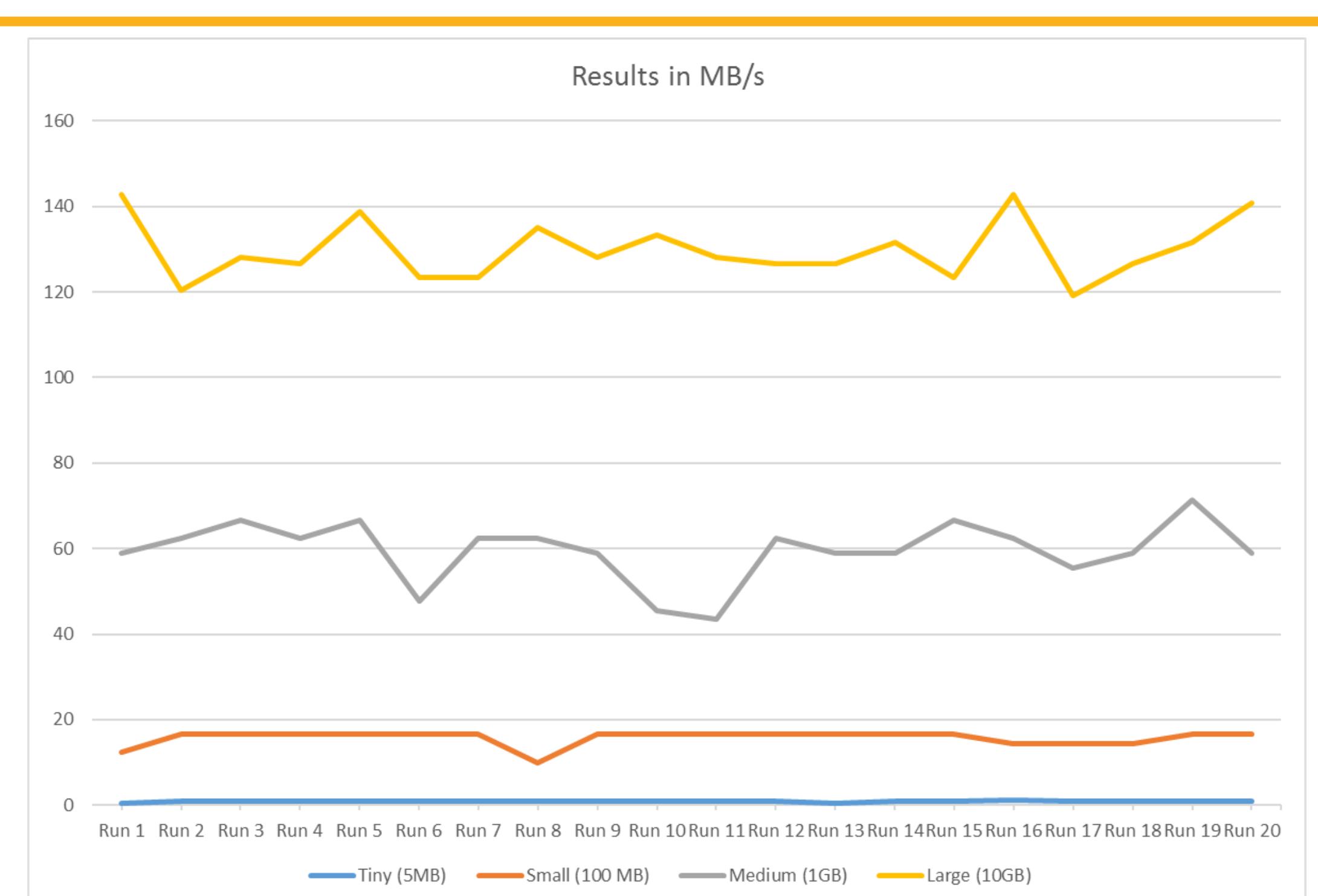


Figure 3: Throughput within the LEXIS DDI using B2SAFE (MB/s from LRZ to IT4I) - depending on file size within a fixed-size data set. Each test was executed 20 times (x-axis).

- To optimise performance of the DDI and of the LEXIS platform as a whole, speed tests were executed on different levels, from the basic DDI system/iRODS federation to the entire LEXIS platform.
- DDI transfers were tested with different control methods (iRODS CLI, EUDAT B2SAFE, iRODS Python client), file sizes (5MB, 100MB, 1GB, 10GB), and directions. Figure 3 shows example results. Generally, a huge advantage in throughput is seen when dealing with larger files.
- Therefore, a Compression API was set up to bundle datasets with many small files before transfer. This significantly accelerates staging.

### CONTACTS

Website: [www.lexis-project.eu](http://www.lexis-project.eu)

Mohamad Hayek: [mohamad.hayek@lrz.de](mailto:mohamad.hayek@lrz.de)

Martin Golasowski: [martin.golasowski@vsb.cz](mailto:martin.golasowski@vsb.cz)

LEXIS WP3 lead: S. Hachinger - [stephan.hachinger@lrz.de](mailto:stephan.hachinger@lrz.de)