Large-scale EXecution for Industry & Society

Deliverable D2.6

Infrastructure validation and assessment report

Co-funded by the Horizon 2020 Framework Programme of the European Union
Grant Agreement Number 825532
ICT-11-2018-2019 (IA - Innovation Action)

DELIVERABLE ID | TITLE	D2.6 | Infrastructure validation and assessment report
RESPONSIBLE AUTHOR	Johannes Munke (LRZ)
WORKPACKAGE ID | TITLE	WP2 | LEXIS Requirement Definition and Architecture Design
WORKPACKAGE LEADER	Atos
DATE OF DELIVERY (CONTRACTUAL)	31/12/2021 (M36)
DATE OF DELIVERY (SUBMITTED)	02/01/2022
VERSION | STATUS	V1.0 Final
TYPE OF DELIVERABLE	R (Report)
DISSEMINATION LEVEL	PU (Public)
AUTHORS (PARTNER)	LRZ; LINKS; IT4I; O24; AVIOAERO; CIMA; CEA; ECMWF; EURAXENT et al.
INTERNAL REVIEW	Andrea Parodi (CIMA); Piyush Harsh (CYC)

Project Coordinator: Dr. Jan Martinovič – IT4Innovations, VSB – Technical University of Ostrava
E-mail: jan.martinovic@vsb.cz, **Phone:** +420 597 329 598, **Web:** https://lexis-project.eu

DOCUMENT VERSION

VERSION	MODIFICATION(S)	DATE	AUTHOR(S)
0.1	Defined table of contents Coarse definition of Section content	01/11/2021	Johannes Munke, Stephan Hachinger (LRZ)

0.2	Added Section 3.1.2
	Added Executive Summary
	Added Section 3.2.2
	Added Section 3.1.3
	Added Section 3.1.1
	Added Sections 3.0 and 3.2.3
	Added Section 3.3
	Added Sections 2.0, 2.1 and 2.2
		10/11/2021
		14/11/2021
		17/11/2021
		17/11/2021
		22/11/2021
		22/11/2021
		07/12/2021
		07/12/2021
			Frédéric Donnat (O24);
			Stephan Hachinger (LRZ);
			Stéphane Louise (CEA);
			Martin Golasowski (IT4I);
			Giacomo Vitali (LINKS);
			Stephan Hachinger (LRZ);
			Marc Derquennes (EURAXENT);
			Johannes Munke (LRZ)

0.3	Added Sections 1, 4 and 5, Final checks and clean up for review	20/12/2021	Stephan Hachinger, Johannes Munke (LRZ)
0.4	Added Section 3.2.1 Applied suggestions of reviewers and final clean up	22/12/2021 23/12/2021	Stephan Hachinger (LRZ); Donato Magarielli (Avio Aero); Johannes Munke (LRZ)
0.5	Corrections from LRZ	28/12/2021	Stephan Hachinger (LRZ)
0.6	Final read and corrections	29/12/2021	Stephan Hachinger (LRZ); Jan Martinovič (IT4I)
1.0	Final check of the deliverable	02/01/2022	Kateřina Slaninová (IT4I)

GLOSSARY

ACRONYM	DESCRIPTION
AAI	Authentication and Authorisation Infrastructure
ALIEN4CLOUD	Application LIfecycle ENablement for Cloud (https://alien4cloud.org)
API	Application Programming Interface
AWS	Amazon Web Services (for Virtual Machine hosting and more)
BB	Burst Buffer
CI	Continuous Integration
CD	Continuous Deployment
D	Deliverable
DAM	Dynamic Allocator Module
DDI	Distributed Data Infrastructure
FAIR	Findable, accessible, interoperable, reusable – the current state-of-the-art paradigm for Research Data Management.
FPGA	Field-Programmable Grid Array (in the context of LEXIS, we mean accelerator cards, possibly with network connectivity, equipped with FPGAs)
GPFS	General Parallel File System (by IBM)
GPU	Graphics Processing Unit
HA	High Availability
HEAppE	High-End Application Execution Middleware (https://heappe.eu)
HPC	High-Performance Computing
HTTP	Hypertext Transfer Protocol
HW	Hardware
IAAS	Infrastructure-as-a-Service, usual denomination for a cloud-service on which entire Virtual Machines can be deployed by the user
IAM	Identity and Access Management
IOT	Internet of Things
IRODS	Integrated Rule-Oriented Data System (https://irods.org/)
M	Month
MODULE (LEXIS MODULE)	Logical/functional entity within the LEXIS Platform, bundling a system or software component which is released, updated and deployed as a whole. Thus, LEXIS modules are the entities relevant for release management and versioning. The actual modules defined as of 2021 are laid out in the present deliverable.
MS	Milestone
NFS	Network File System
NVME	NVM (non-volatile Memory) Express, usually used as interface to SSDs

NVMEOF	NVMe-Over-Fabrics
NVRAM	Non-Volatile Random Access Memory
PID	Persistent Identifier
RDMA	Remote Direct Memory Access
ROCE	RDMA over Converged Ethernet
REST	Representational State Transfer
SCP	Secure Copy
SIEM	System Information and Event Management
SR-IOV	Single Root I/O Virtualisation
SSD	Solid State Drive
SSH	Secure Shell
SSHFS	Secure Shell File System – a mechanism to mount filesystems of remote servers contacted via SSH (https://github.com/libfuse/sshfs)
TOSCA	Topology and Orchestration Specification for Cloud Applications (by OASIS, http://docs.oasis-open.org/tosca/TOSCA/v1.0/os/TOSCA-v1.0-os.html)
TRL	Technology Readiness Level (https://ec.europa.eu/research/participants/data/ref/h2020/wp/2014_2015/annexes/h2020-wp1415-annex-g-trl_en.pdf)
VM	Virtual Machine
VPN	Virtual Private Network
V100	NVIDIA Tesla V100 Graphics Card with Volta GV100 GPU
WCDA	Weather and Climate Data API
WP	Work Package
YORC	Ystia Orchestrator (https://ystia.github.io/)

TABLE OF PARTNERS

ACRONYM	PARTNER
Avio Aero	GE AVIO SRL
Atos	BULL SAS
AWI	ALFRED WEGENER INSTITUT HELMHOLTZ ZENTRUM FUR POLAR UND MEERESFORSCHUNG
BLABS	BAYNCORE LABS LIMITED
CEA	COMMISSARIAT A L ENERGIE ATOMIQUE ET AUX ENERGIES ALTERNATIVES
CIMA	CENTRO INTERNAZIONALE IN MONITORAGGIO AMBIENTALE - FONDAZIONE CIMA
CYC	CYCLOPS LABS GMBH
ECMWF	EUROPEAN CENTRE FOR MEDIUM-RANGE WEATHER FORECASTS
EURAXENT	MARC DERQUENNES
GFZ	HELMHOLTZ ZENTRUM POTSDAM DEUTSCHESGEOFORSCHUNGSZENTRUM GFZ
ICHEC	NATIONAL UNIVERSITY OF IRELAND GALWAY / Irish Centre for High-End Computing
IT4I	VYSOKA SKOLA BANSKA - TECHNICKA UNIVERZITA OSTRAVA / IT4Innovations National Supercomputing Centre
ITHACA	ASSOCIAZIONE ITHACA
LINKS	FONDAZIONE LINKS / ISTITUTO SUPERIORE MARIO BOELLA ISMB
LRZ	BAYERISCHE AKADEMIE DER WISSENSCHAFTEN / Leibniz Rechenzentrum der BAdW
NUM	NUMTECH
O24	OUTPOST 24 FRANCE
TESEO	TESEO SPA TECNOLOGIE E SISTEMI ELETTRONICI ED OTTICI

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

EXECUTIVE SUMMARY

Following up on earlier infrastructure- and technology-oriented Deliverables D2.1-D2.4 [1, 2, 3, 4], this Deliverable D2.6 describes the final status of the LEXIS Platform, based on quality control / release management and validation activities. It accompanies Deliverable D2.5 [5], which assesses the co-designed LEXIS architecture.

While the deliverable builds upon the project results up to Month 30 (M30), and presents links to other deliverables where appropriate, it mainly reflects the outcomes of an intensive period of infrastructure assessment, validation and consolidation from M30 to M36. Release management and status of the LEXIS modules - forming the LEXIS Platform and mainly encapsulating software components - have undergone several reviews, leading to a well-synchronized understanding of the platform among co-design and Pilot work packages. All missing documentation on the release management, technical and user-view side has been contributed, and platform components have been successfully deployed at one more computing centre (ICHEC). A multi-stage discussion and resolution of remaining integration issues has been accompanied by an internal validation procedure, by scenario-based user-view tests of the platform, a security assessment, and extensive performance tests. At the same time, validation by Pilot workflows reached its peak intensity, stress-testing the platform and triggering further refinements. More and more LEXIS Open Call workflows are run as well and used to validate the LEXIS Platform from an absolutely independent end-user point of view. All in all, despite all difficulties the project has reported and faced, validation evidences a matured platform reaching a TRL near to 8, as promised in the project proposal. The platform components and documentation, as produced during the project, are being disseminated as Open Source software and openly accessible documentation.

Position of the deliverable in the whole project context

Deliverable D2.6 (together with Deliverable D2.5 [5]) shall allow the project consortium to certify the achievement of Milestone MS8 “Final integration and LEXIS technologies validation”, for which - besides this report - a full integration of data system and orchestration system is required as well as benchmarking and bottleneck-removal efforts on the platform. The latter efforts focus in particular on the Distributed Data Infrastructure, whose optimization is further described in Deliverable D3.6 (referenced here where appropriate). Deliverable D2.6 [5] is the outcome of the work of Task 2.3 “Infrastructure Set-up, Roll-outs, Validation and Benchmarking” and Task 2.2 “Co-Design and Lessons Learned”. It builds upon earlier deliverables resulting from these two Tasks (as described at the beginning of this summary), in particular on Deliverable D2.4 [4], but adds the validation and assessment aspects as a completely new theme.

Description of the deliverable

The outline of the deliverable follows roughly the timeline of our technical management, assessment and validation efforts in particular within the last year and with a focus on the last six months. It begins with an introduction (Section 1) and a summary of the release management concept and documented release status of the LEXIS modules (Section 2). The largest part of the deliverable then focuses on the validation (Section 3). We distinguish (i) “internal” validation procedures, as defined within the co-design work-packages, and respective results (Section 3.1), (ii) validation procedures and results by Pilots (WP5-WP7; Section 3.2), and (iii) validation results obtained within the LEXIS Open Call and with extension of the platform to ICHEC (Section 3.3). Section 4 discusses our efforts to openly disseminate documentation and software/source-code packages needed to join the LEXIS Platform and to operate it. Section 5 summarizes the deliverable. Main contributors in writing have been LRZ, LINKS, O24, IT4I, Avio Aero, CEA, CIMA and ECMWF.

1 INTRODUCTION

With the LEXIS project progressing, the LEXIS Platform has been validated, and platform management concepts for sustained operation have been introduced. The present deliverable reports on our final validation and assessment of the infrastructure deployed within the LEXIS H2020 project. It builds upon Deliverable D2.4 [4], where the roll-out status of LEXIS technology and infrastructure has been reported in the framework of a consistent release management for the first time. Besides delivering an update to Deliverable D2.4 [4], allowing for an assessment of the release and deployment status (Section 2), we concentrate on discussing the validation of the LEXIS Platform (Section 3). In the efforts described here, we focus on technical/functional aspects and validation results, complementing the assessment within Deliverable D2.5 [5] which focuses on co-design and architecture. We report on tests internally designed within the technical work packages (Section 3.1), platform validation by the LEXIS Pilots (Section 3.2), and on the inclusion of further use cases (within the LEXIS Open Call) as well as a new platform partner (ICHEC), which is putting the platform to the test (Section 3.3). The deliverable closes with a discussion of LEXIS software dissemination on Github/Zenodo (Section 4), allowing for an assessment of the LEXIS Open Source strategy, and with a summary (Section 5).

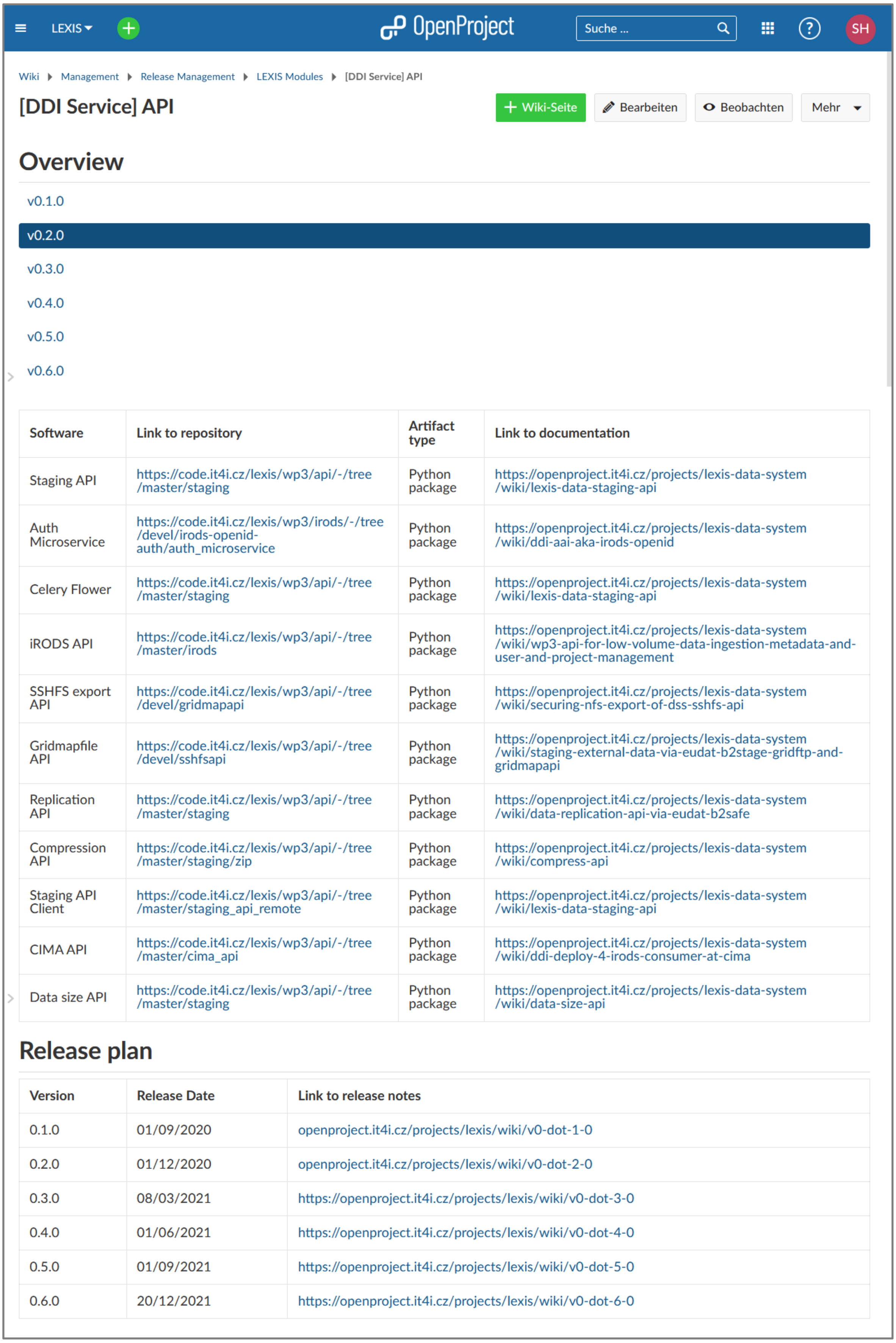
2 DEPLOYED INFRASTRUCTURE AND RELEASE MANAGEMENT ASSESSMENT

To ensure a decent quality of the LEXIS components, a release management concept was introduced in the beginnings of the project. The concept and associated guidelines were introduced in Deliverable D2.4 [4]. These guidelines defining the versioning concept and the branching model are used during the development process. Also, the templates for the release documentation are provided, which consist of a release plan, changelog and links to the corresponding code repositories.

The following Section 2.1 shows the release management status of an exemplary selected module in detail. An overview of the complete status of the LEXIS release management is given in Section 2.2. This overview, based on an availability check for release documentation for all submodules, shows that the LEXIS release management concept is currently implemented and working.

2.1 EXAMPLE OF RELEASE STATUS

One central element of the LEXIS release management is the OpenProject instance of the project. As already mentioned in the description of the release management concept in Deliverable D2.4 [4], every LEXIS submodule has its own release management page on this instance. The complete list of modules can be found in Table 1 of the following Section 2.2. The following screenshots (Figure 1 and Figure 2) are showing the documentation of the current development state of the LEXIS submodule “API” of the LEXIS module “DDI” (Distributed Data Infrastructure). Figure 1 gives an overview of versions and locations of the repositories and the documentation of the artefacts related to the release, while Figure 2 shows a changelog containing all changes which lead to version 0.6.0 of the DDI APIs.



Wiki ▸ Management ▸ Release Management ▸ LEXIS Modules ▸ [DDI Service] API

[DDI Service] API

Overview

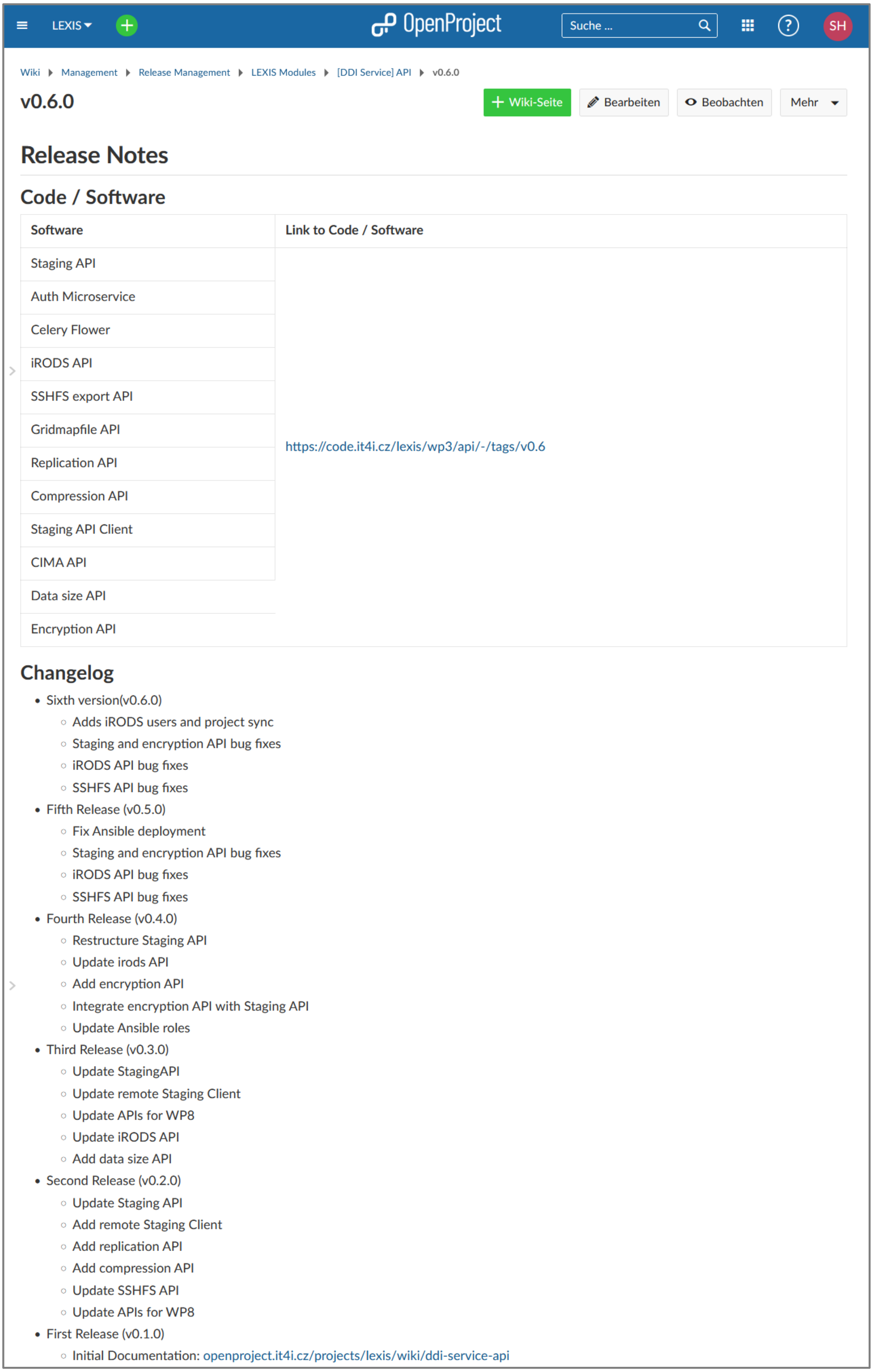
- v0.1.0
- v0.2.0
- v0.3.0
- v0.4.0
- v0.5.0
- v0.6.0

Software	Link to repository	Artifact type	Link to documentation
Staging API	https://code.it4i.cz/lexis/wp3/api/-/tree/master/staging	Python package	https://openproject.it4i.cz/projects/lexis-data-system/wiki/lexis-data-staging-api
Auth Microservice	https://code.it4i.cz/lexis/wp3/irods/-/tree/devel/irods-openid-auth/auth_microservice	Python package	https://openproject.it4i.cz/projects/lexis-data-system/wiki/ddi-aai-aka-irods-openid
Celery Flower	https://code.it4i.cz/lexis/wp3/api/-/tree/master/staging	Python package	https://openproject.it4i.cz/projects/lexis-data-system/wiki/lexis-data-staging-api
iRODS API	https://code.it4i.cz/lexis/wp3/api/-/tree/master/irods	Python package	https://openproject.it4i.cz/projects/lexis-data-system/wiki/wp3-api-for-low-volume-data-ingestion-metadata-and-user-and-project-management
SSHFS export API	https://code.it4i.cz/lexis/wp3/api/-/tree/devel/gridmapapi	Python package	https://openproject.it4i.cz/projects/lexis-data-system/wiki/securing-nfs-export-of-dss-sshfs-api
Gridmapfile API	https://code.it4i.cz/lexis/wp3/api/-/tree/devel/sshfsapi	Python package	https://openproject.it4i.cz/projects/lexis-data-system/wiki/staging-external-data-via-eudat-b2stage-gridftp-and-gridmapapi
Replication API	https://code.it4i.cz/lexis/wp3/api/-/tree/master/staging	Python package	https://openproject.it4i.cz/projects/lexis-data-system/wiki/data-replication-api-via-eudat-b2safe
Compression API	https://code.it4i.cz/lexis/wp3/api/-/tree/master/staging/zip	Python package	https://openproject.it4i.cz/projects/lexis-data-system/wiki/compress-api
Staging API Client	https://code.it4i.cz/lexis/wp3/api/-/tree/master/staging_api_remote	Python package	https://openproject.it4i.cz/projects/lexis-data-system/wiki/lexis-data-staging-api
CIMA API	https://code.it4i.cz/lexis/wp3/api/-/tree/master/cima_api	Python package	https://openproject.it4i.cz/projects/lexis-data-system/wiki/ddi-deploy-4-irods-consumer-at-cima
Data size API	https://code.it4i.cz/lexis/wp3/api/-/tree/master/staging	Python package	https://openproject.it4i.cz/projects/lexis-data-system/wiki/data-size-api

Release plan

Version	Release Date	Link to release notes
0.1.0	01/09/2020	openproject.it4i.cz/projects/lexis/wiki/v0-dot-1-0
0.2.0	01/12/2020	openproject.it4i.cz/projects/lexis/wiki/v0-dot-2-0
0.3.0	08/03/2021	https://openproject.it4i.cz/projects/lexis/wiki/v0-dot-3-0
0.4.0	01/06/2021	https://openproject.it4i.cz/projects/lexis/wiki/v0-dot-4-0
0.5.0	01/09/2021	https://openproject.it4i.cz/projects/lexis/wiki/v0-dot-5-0
0.6.0	20/12/2021	https://openproject.it4i.cz/projects/lexis/wiki/v0-dot-6-0

Figure 1: Overview OpenProject page of the "API" submodule of the LEXIS module "DDI"


LEXIS
OpenProject
Suche ...
SH
Wiki ▸ Management ▸ Release Management ▸ LEXIS Modules ▸ [DDI Service] API ▸ v0.6.0
+ Wiki-Seite
Bearbeiten
Beobachten
Mehr

v0.6.0

Release Notes

Code / Software

Software	Link to Code / Software
Staging API	https://code.it4i.cz/lexis/wp3/api/-/tags/v0.6
Auth Microservice	
Celery Flower	
iRODS API	
SSHFS export API	
Gridmapfile API	
Replication API	
Compression API	
Staging API Client	
CIMA API	
Data size API	
Encryption API	

Changelog

- Sixth version(v0.6.0)
  - Adds iRODS users and project sync
  - Staging and encryption API bug fixes
  - iRODS API bug fixes
  - SSHFS API bug fixes
- Fifth Release (v0.5.0)
  - Fix Ansible deployment
  - Staging and encryption API bug fixes
  - iRODS API bug fixes
  - SSHFS API bug fixes
- Fourth Release (v0.4.0)
  - Restructure Staging API
  - Update irods API
  - Add encryption API
  - Integrate encryption API with Staging API
  - Update Ansible roles
- Third Release (v0.3.0)
  - Update StagingAPI
  - Update remote Staging Client
  - Update APIs for WP8
  - Update iRODS API
  - Add data size API
- Second Release (v0.2.0)
  - Update Staging API
  - Add remote Staging Client
  - Add replication API
  - Add compression API
  - Update SSHFS API
  - Update APIs for WP8
- First Release (v0.1.0)
  - Initial Documentation: openproject.it4i.cz/projects/lexis/wiki/ddi-service-api

Figure 2: OpenProject page of the Version 0.6.0 of the "API" submodule of the LEXIS module "DDI".

As evident from Figure 1 and Figure 2, the development process was following the guidelines mentioned above and introduced in Deliverable D2.4 [4]. During the development process, six versions have been released so far. The related code from the table will be made public according to the process described in Section 4 and Deliverable D9.10 [6].

2.2 RELEASE MANAGEMENT STATUS

Table 1 shows an updated listing of all LEXIS modules and their submodules from Deliverable D2.4 [4]. A diagram which reflects the architecture of LEXIS with its modules is included in Deliverable D2.5 [5]. Table 1 also indicates whether at least one release with documentation has been made for each submodule. This information is based on a check at the beginning of December, 2021. The availability of a release documentation for practically all submodules reflects the fact that the release management concept is working and followed.

MODULE	SUBMODULE	DESCRIPTION	Avail. docs
HPC (WP4)	USAGE COLLECTORS	HPC usage collector for Billing system via HEAppE	Yes
	HEAPPE INSTANCES	Middleware solution for providing HPC capabilities	Yes
	CLUSTERS	Release actions within the module “Clusters” represent all actions to make HPC clusters in the LEXIS centres accessible via the LEXIS Platform	Yes
LEXIS BACKEND SERVICES (WP8)	API	Unified API for the portal, proxying requests to the lower-level APIs (e.g. DDI APIs) and the UserOrg-Service of the LEXIS Platform	Yes
	USERORG-SERVICE	Go Module for the creation, control and account of users, organizations, projects and resources	Yes
	SERVICE INTERFACE - DATASETS	Go Module providing interface to the WP3 Datasets	Yes
	SERVICE INTERFACE - A4C	Go Module providing interface to the A4C Workflows	Yes
	SERVICE INTERFACE - APPROVAL SYSTEM	Go Module providing interface to the Approval Service	Yes
LEXIS FRONTEND (WP8)	DATASETS	Module for datasets management (upload, view, etc.) and DDI	Yes
	LEXIS PORTAL BACKEND	Go Module to serve the portal frontend and keep up part of the interaction with the users	Yes
	AAI CONNECTOR	Connector to KeyCloak and local AAI	Yes
	WORKFLOWS	Module for workflow management (definition, execution, monitoring, etc.)	Yes
	RESOURCES AND BILLING	Module for resource usage visualization and billing	Yes
	USER AND USERGROUP MANAGING	Users and organizations management in cooperation with security settings.	Yes

ORCHESTRATOR SERVICE (WP4)	YORC DYNAMIC ORCHESTRATION PLUGIN	Yorc plugin interacting with the Dynamic Allocation Module to get dynamically the locations of infrastructure resources where to create during a workflow execution	**Yes**
	YORC OIDC CLIENT	Library of functions to interact with OpenID Connect, bundled in Yorc LEXIS plugins	**Yes**
	API	API exposed to the portal to interact w/ orchestrator	**Being made**
	YORC	Component provides main orchestration functionalities and it interacts w/ A4C and HEAppE middleware	**Yes**
	DYNAMIC ALLOCATOR MODULE	This component provides functionalities for dynamic placement	**Yes**
	ALIEN4CLOUD (A4C)	Frontend of the Yorc orchestration engine, exposing interface and API to the portal	**Yes**
	A4C GO CLIENT	Go client for Alien4Cloud REST API used by the Portal	**Yes**
	YORC PLUGIN - HEAppE	Yorc plugin developed so that Yorc can use the HEAppE API to manage HPC resources	**Yes**
	YORC PLUGIN - DDI	Yorc plugin developed so that Yorc can use the DDI API to manage Data transfers	**Yes**
	TOSCA TEMPLATES	TOSCA templates for common workflows in LEXIS, for easy uptake by users	**Being made**
AAI SERVICES (WP4)	MONITORING	Monitoring using syslog send to syslog-server or standard monitoring system from HPC centres	**Yes**
	KEYCLOAK/KEYCLOAK API	Identity and Access Management for LEXIS	**Yes**
	KEYCLOAK LIBRARY	Keycloak Library extension to Gocloak Library	**Yes**
ACCOUNTING AND BILLING SERVICE (WP8)	CYCLOPS SYSTEM	Accounting and billing framework for LEXIS services	**Yes**
APPROVAL SERVICE (WP4)	HEAppE´s MANAGE & STATUS REPORT	Module for managing HEAppE instances	**Yes**
	API	API for providing HPC projects information to LEXIS portal	**Yes**
	Approval	Module for management of HPC computational projects	**Yes**
CLOUD (WP2)	BURST BUFFER	Atos SBB/SBF products deployed with data node machines at IT4I and LRZ	**Yes**
	FPGA	FPGA card deployed in IT4I Burst Buffer no. 2 with drivers	**Yes**

DDI SERVICE (WP3)	OPENSTACK USAGE COLLECTORS	OpenStack Cyclops' collectors	**Yes**
	IRODS	iRODS zone deployment at IT4I and LRZ and tests	**Yes**
	APIs	APIs, e.g. for data staging between iRODS and different computing systems	**Yes**
	WCDA (API)	API for curated weather & climate data and metadata	**Yes**
	PERFORMANCE MONITORING	Automated scripts measuring DDI performance	**Yes**
MONITORING (WP3)	SYSTEM TESTS	DDI tests	**Yes**
	MONITORING CORE	Prometheus/Grafana	**Yes**

Table 1: LEXIS modules (first column) and their submodules (second column), a short description and an indication about the presence of release documentation as of 12/2021.

3 VALIDATION OF THE PLATFORM

In parallel to the development of a consistent release scheme, the LEXIS Platform has been validated in terms of functionality and reliability. In order to reach Milestone MS8 of the LEXIS project, various components have been validated and benchmarked, and seen optimization in order to remove bottlenecks or add missing functionalities. Thus, the platform has been made future-proof for sustained operation after the LEXIS H2020 project.

Validation has proceeded in three scopes: with internal test and validation procedures (Section 3.1), the co-design work packages (WP2, 3 and 4) have aimed at understanding whether usage scenarios are well implemented (Section 3.1.1), whether further security hardening is needed (Section 3.1.2), and whether synthetic benchmarks indicate decent speed (Section 3.1.3). In parallel, the LEXIS Pilots (WP5, 6, 7) have validated the platform running and optimizing their workflows (Section 3.2). Finally, feedback from the LEXIS Open Call users (cf. also Deliverable D9.12 [7]) has been received to validate the platform, to identify and resolve issues important from an absolute end-user point of view, while with ICHEC also an additional computing centre has been successfully added to the platform (Section 3.3).

3.1 INTERNAL VALIDATION

3.1.1 Feature-based platform tests

LEXIS is a platform for the execution of complex workflows and data management tasks, however with a clear target to be easy to use thanks to a user-friendly portal and to specific APIs, while also implementing an efficient and state-of-the-art AAI with a single-sign-on mechanism. After the implementation phase, the system needed to be properly tested and validated before allowing public access to it, by properly verifying all relevant features. Despite all challenges, we have succeeded in providing the necessary functionality to on-board the LEXIS Open Call partners and begin supplying them with our LEXIS services. We have then focused our ambition on making the current set of features reliable. Further functionality (such as a more automated process for organization creation - which currently works via LEXIS support) can be added in future development cycles on the platform.

The features/functionalities tested are described in the following paragraphs. For each feature, a test-procedure table is given. For Feature A, the verification is reflected by LEXIS Portal screenshots in order to illustrate the process.

Feature A - Creation of users and rights assignment

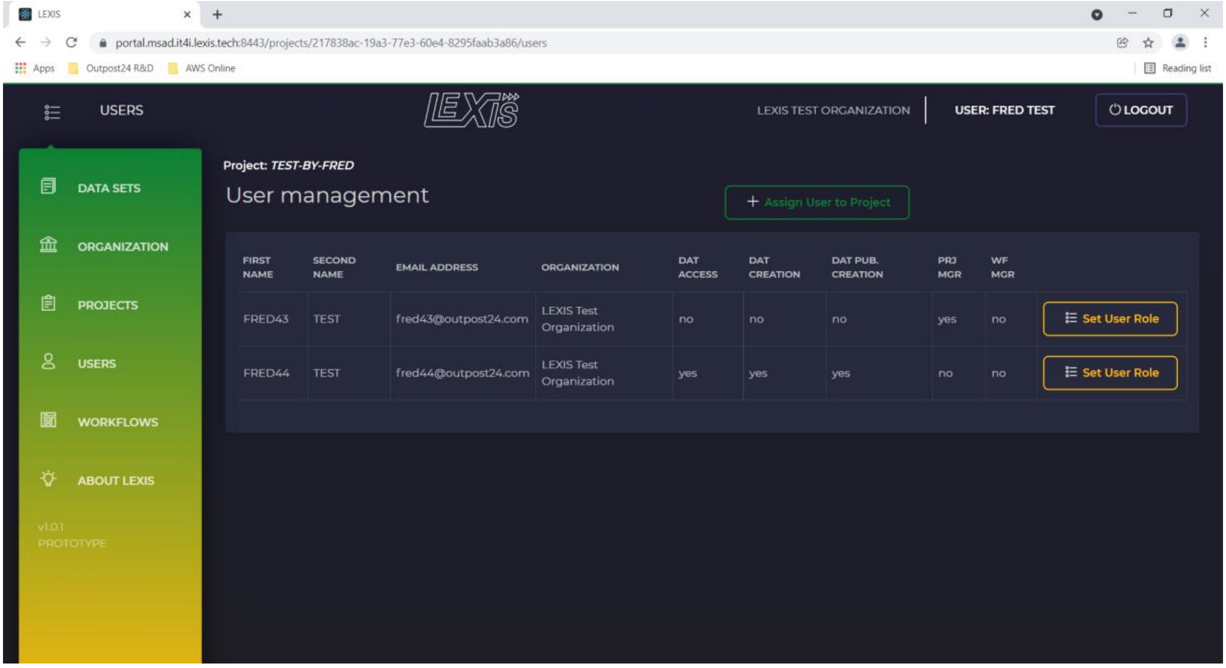
Table 2 reports the procedure for testing Feature A.

Description	**The feature consists in creating a user, adding her/him to a project, and giving her/him different permissions. We have tested in particular the assignment of project management, dataset management and workflow management rights.**
Requirements	• An existing user being a manager of an organization, • A LEXIS project.
Step 1	The Organization Manager creates a user in the LEXIS Portal.
Step 2	The Organization Manager adds the previously created user to a project, and gives her/him the desirable permissions.
Step 3	The Organization Manager is able to see the user and his/her permissions.
Step 4	The newly created user is able to act on the project according to her/his rights, by

	- Logging into LEXIS Portal, - Verifying whether (s)he can access the project modification, dataset management and workflow management functionalities corresponding to the rights assignment.

Table 2: Verification of Feature A - feature description and steps for verification.

Figure 3 shows the Organization Manager view after user creation and role assignment. It illustrates the successful verification of Step 3.



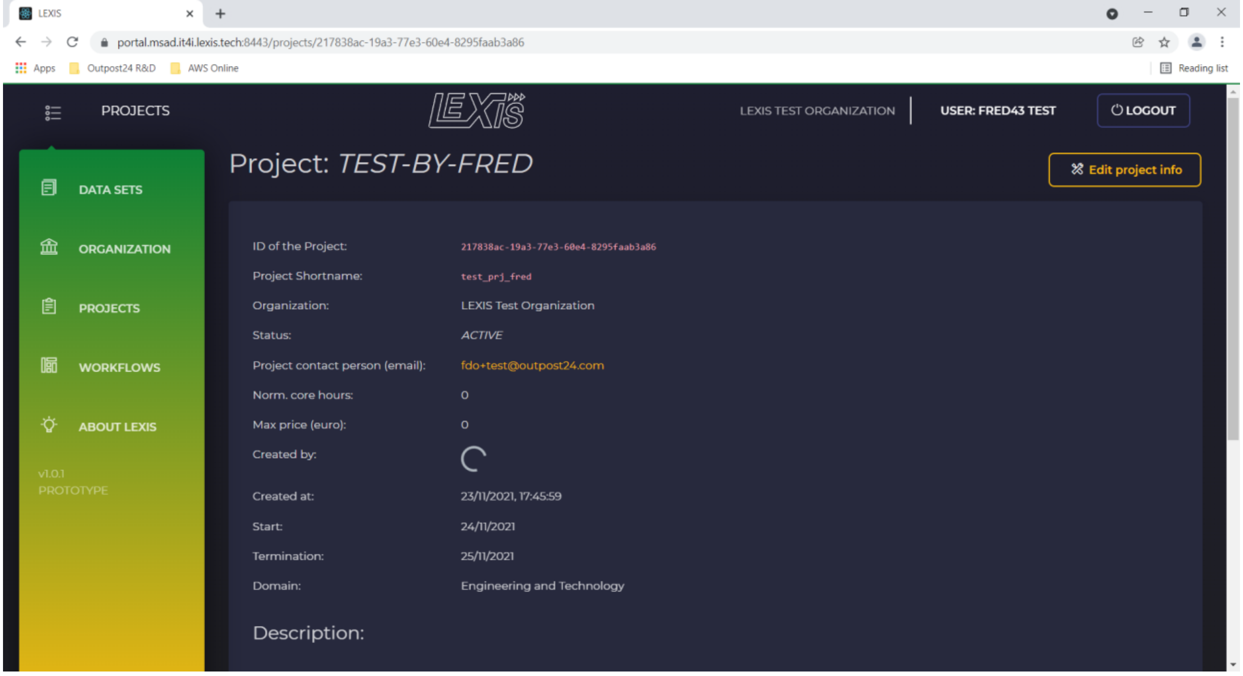
Project: *TEST-BY-FRED*

User management

FIRST NAME	SECOND NAME	EMAIL ADDRESS	ORGANIZATION	DAT ACCESS	DAT CREATION	DAT PUB. CREATION	PRJ MGR	WF MGR	
FRED43	TEST	fred43@outpost24.com	LEXIS Test Organization	no	no	no	yes	no	Set User Role
FRED44	TEST	fred44@outpost24.com	LEXIS Test Organization	yes	yes	yes	no	no	Set User Role

Figure 3: Verification of Feature A - Screenshot of Organization Manager view in portal after Step 3

Figure 4 illustrates that the user “FRED43” can successfully access project information and the button for modification/editing is active (Step 4).



Project: *TEST-BY-FRED*

ID of the Project: 217838ac-19a3-77e3-60e4-8295faab3a86
Project Shortname: test_prj_fred
Organization: LEXIS Test Organization
Status: *ACTIVE*
Project contact person (email): fdo+test@outpost24.com
Norm. core hours: 0
Max price (euro): 0
Created by:
Created at: 23/11/2021, 17:45:59
Start: 24/11/2021
Termination: 25/11/2021
Domain: Engineering and Technology

Description:

Figure 4: Verification of Feature A - Screenshot of user’s view as a Project Manager who can edit project information

When accessing the portal as a Data or Organization Manager without project management rights (e.g. user “FRED44” as shown in Figure 3), project details can neither be accessed nor modified. This has been verified together with the data-management access for respective test user accounts.

Feature B - Project creation and assignment of resources at computing/data centres

Table 3 details the procedure for verifying Feature B.

Description	**The feature/functionality comprises the creation of a LEXIS project by an Organization Manager, as well as the request and correct assignment of the needed computing and storage resources.**
Requirement	• An existing user being Organization Manager within an organization.
Step 1	The Organization Manager user creates a new project “prj1” in the portal (getting also Project Manager and Data Manager rights on his new project).
Step 2	The manager verifies that the portal’s dataset management interface, showing her/his space within the LEXIS Distributed Data Infrastructure, is accessible for her/his LEXIS Project.
Step 3	• The manager asks for an assignment of HPC/Cloud-Computing resources (pre-existing allocations) to the LEXIS Project, • The LEXIS team with the manager verify the correct assignment and creation of the HEAppE instances needed to access the HPC/Cloud-Computing resources.

Table 3: Verification of Feature B - feature description and steps for verification

The described steps have been executed, in particular, for various LEXIS Open Call projects, and therefore Feature B has been successfully verified.

Feature C - Data management on the LEXIS DDI through the portal

Table 4 illustrates Feature C and the procedure for its verification.

Description	**The feature consists of basic data management within a project.**
Requirement	• A user with data access and data creation rights.
Step 1	User creates two datasets within the DDI for input data and application container for a simple one-step workflow as an example.
Step 2	User uploads • Input data for the workflow to one of the datasets, • A container image to the other data set.

Table 4: Verification of Feature C - feature description and steps for verification

The full procedure has been executed by uploading several images and input data to the DDI within the scope of the LEXIS Pilot and the LEXIS Open Call participants. It follows that Feature C has been correctly verified, also by checking whether the uploaded data are accessible for download.

Feature D - Workflow execution via the LEXIS Portal

Full workflow execution initiated by a user - constituting a test of a majority of the capabilities of the platform - has been defined as Feature D, for which the verification procedure is detailed in Table 5.

Description	**The feature tested here is the execution of an application in the federated LEXIS environment by instantiating a workflow execution from an available workflow.**

Requirement	- A user with Workflow Manager rights (implying Project and Data Manager rights).
Step 1	User logs into the portal and instantiates a new workflow from one of the implemented templates, filling in all mandatory parameters.
Step 2	User starts a workflow execution of the newly-created workflow.
Step 3	User downloads output data from the newly-created workflow.

Table 5: Verification of Feature D - feature description and steps for verification

The steps have been successfully performed for several available workflow templates, one example being the WP7 as described in Section 3.2.3. The workflows have been executed, with progress being monitored via the portal.

Verification result

When testing the essential set of features illustrated above, glitches have been removed and the platform with all its front- and backend components has been successfully verified in its basic functionality.

3.1.2 Security assessment

Based on the co-design documents on LEXIS architecture, in particular the ones defining the security architecture (Deliverables D4.5 [8] and D4.7 [9]), security assessment procedures have been defined to verify the actual implementation. This ensures that the LEXIS Platform architecture complies with a “Zero Trust Architecture” in terms of deployment and that all security concepts such as “Defence in Depth” or “Network Perimeter Security” are really put in place. The implementation of the security policies has matured with the platform, and is described in the following paragraphs.

Assessment of software components

In Deliverable D4.5 [8] and Deliverable D4.7 [9], the security status of several components has been assessed based on available documentation and a write-up of security-relevant parts of their concept. Several significantly exposed or important systems have then undergone auditing and tests (code audits, running audits, etc.) in particular:

- LEXIS AAI,
- LEXIS Portal,
- LEXIS APIs from LEXIS Portal, LEXIS Orchestrator, LEXIS DDI,
- IT4I HEAppE, and
- LEXIS DAM.

As a more concrete example, the code of the DAM has been internally audited with several Open Source tools such as Pylama [10] which encompasses several other Open Source tools for static code and style analysis such as Pylint [11], Pycodestyle [12] and Pyflakes [13]. The dynamic analysis of the code source has been made using some scripting in order to access APIs (trying to abuse the API by injecting some fault or trying to bypass some permission, etc.). This allowed us to detect and remove some flaws in the code that could lead to security issues.

The LEXIS-internal development code repository (IT4I gitlab instance) is also monitored for establishing security best practices and avoid common security flaws such as leak of sensitive secrets (SSH Keys, API tokens, password, etc.). In case of leaks, data has been removed from the repository and the secret has be updated on the LEXIS Platform. This repository monitoring has also been an important prerequisite for publishing official code of most modules on GitHUB, as intended for LEXIS (cf. Section 4). As side note it is important to mention that source-code management platform such as GitLab conveniently embed security components such as secret detection [14].

Regular tests to assess general infrastructure security

From an infrastructure standpoint, the firewalls within the platform are checked regularly and the logs are audited. There is no full automation, as of now, in order to detect port scanning or tries to abuse the network perimeter, but all logs from the platform are sent to a system information and event management (SIEM) tool (Elasticsearch [15]) for complete monitoring of the infrastructure.

The infrastructure is also regularly assessed from a vulnerability standpoint with several Open Source tools such as OpenVAS [16] that are placed inside the security perimeter.

Regular tests to assess portal, API and platform security

We have automated some important tests to run them on a continuous basis against our Development and Production systems. As an example, we have created scripts to check the access management on the LEXIS Portal API that try to:

- Create/update/delete users belonging to a specific organization without having proper permissions, simulating a malicious user trying to perform management tasks on a 3rd-party organization; also, important details such as a modification of users' email addresses were tested as this concerns the destination of password-reset messages,
- Create/update/delete permissions for a specific user on a specific project or resource, simulating a malicious user trying to gain access to a specific project or dataset (changing her/his own permission or adding another fake user).

The same approach was used for all the APIs, focusing on attempts to bypass the authentication or authorization measures put in place. With appropriate and double-checked measures in LEXIS, we have thus made sure that if one control is failing then a second layer of control will block the malicious user, validating also the Zero Trust Architecture.

Security audit based on external-attacker perspective

Clearly, all these concepts - in particular the audits and tests already successfully carried out, have already helped us to make the LEXIS Platform secure and to validate its consistent security concept. After setting up both the security best practices for code development and code deployment (including the tools themselves such as GitLab), and the security best practices for infrastructure security (including all security tools and equipment such as firewalls, scanners, monitoring, etc.), we have finally conducted an audit based on an external-attacker perspective.

This audit assessed the entire LEXIS infrastructure by attempting to consider attack surfaces and identify weaknesses that could lead to security breach, allowing a malicious attacker to get access on LEXIS Platform. We used tools such as DNSdumpster [17], Sublist3r [18] and Amass [19] to discover servers/applications running in the LEXIS infrastructure and domain. Once all the potential targets were identified, we then ran some automated vulnerability scanning using a collection of popular Open Source tools [20] and the Outpost24 vulnerability scanner [21]. In addition to automated vulnerability scanning from the outside, we are planning to audit the infrastructure by direct (human) simulation of an external attacker. This audit will again focus on attempts to first discover all LEXIS infrastructure components, then on finding weaknesses in the system and finally on gaining access privileges.

Results

While the detailed results of our security enhancement efforts are obviously not to be published, we can summarize here that we have identified and fixed several significant vulnerabilities – including vulnerabilities in 3rd-party/community software – within the LEXIS infrastructure. After all our iterations made for improvement, we are confident to offer an infrastructure with solid security status.

3.1.3 Performance tests

Performance tests were conducted during the validation phase to get an overview of the performance limits of the platform. As the performance characteristics of individual computing clusters within the infrastructure are well known, we have focused on network bandwidth tests and on tests of data-transfer speeds within the DDI. The results of these tests predict the time spent between the actual execution of workflow steps. Using these predictions together with the computing-cluster characteristics, the DAM has all necessary information for an optimum placement of workflow steps under the constraints given by the user.

To begin with, we repeatedly tested network bandwidths between the LEXIS compute sites with the iperf3 tool [22]. The latest tests have been performed between the infrastructure VMs located in IT4I, LRZ and ICHEC. Each test generated a TCP stream for 60 seconds using four parallel threads in both directions, repeated three times. Results of the tests are presented in Table 6; all speeds are measured in Gbit per seconds. IT4I and LRZ used VMs deployed on on-premises VMware clusters, while ICHEC provided an AWS instance for the tests.

SOURCE	IT4I	LRZ	ICHEC
IT4I	-	1.14 Gbps	0.95 Gbps
LRZ	1.37 Gbps	-	1.17 Gbps
ICHEC	2.09 Gbps	1.14Gbps	-

Table 6: Measured bandwidths between federated sites within the LEXIS infrastructure.

The tests show that a convenient WAN bandwidth of at least 1 Gbps can be expected between all sites. In particular - as the ICHEC machine is deployed on AWS - it turns out that a similar bandwidth can be expected to commercial computing centres.

The performance reported here for the network reflects decent data-transfer speeds within the LEXIS DDI of roughly 100 MByte/s, as long as no large number of small files are copied, which requires repeated writes into the DDI/iRODS metadata catalogue (iCAT). These findings are reported in detail in Deliverable D3.6 [23], and are part of a cascade of DDI speed tests further discussed in that deliverable.

3.2 VALIDATION BY PILOTS

Validation by pilots has focused on functional key requirements as presented in Deliverable D2.5 [5], Section 3. The aspects listed there in tabular form are thus among the most important points covered below.

3.2.1 WP5

WP5 has validated the LEXIS Platform with aeronautical computational fluid dynamics applications, simulating a low-pressure turbine in the "Turbomachinery Use Case" and rotating parts/gearboxes in the “Rotating Parts Use Case”. The focus of the first use case and the related validation efforts have been on computational performance gained for an industrial turbomachinery application by secure and effective usage of HPC systems, with very remarkable boosts by GPU usage, as well as live tracking of the simulation processes by checkpointing and visualisation, with the final aim of efficiently examining complex fluid dynamic behaviour in aeronautical engine critical components such as turbines. Instead, the research and implementation activities in the second use case have been focused on supporting the execution of the adopted CFD application pre-processor and solver, optimizing their set-up as well, with the final target of developing and assessing a newly designed CFD methodology aimed at studying complex flow fields in mechanical parts rotating at high speed. All in all, WP5 workflows have validated the following aspects of the platform, as mentioned in Deliverable D2.5 [5]:

- HPC and Cloud-Computing tasks including Long-Term HEAppE jobs,
- Checkpointing and failover,
- Security including data encryption and replication,
- Visualisation,
- Orchestration of workflows on the federated platform.

Below, we further expand on some of these aspects. WP5 deliverables have from the very beginning had a strong focus on validating the platform for industrial usage (while the other work packages follow an approach focused a bit more on scientific reporting). Thus, the sections below contain many pointers to WP5 deliverables describing the validation further - with very encouraging results.

Validation of workflow building blocks: HPC runs with performance evaluation and Cloud jobs

Within Deliverable D5.4 [24], the benefits in particular of running TRAF (main code for the Turbomachinery Use Case, cf. Deliverable D5.4 [24]) and nanoFluidX (main code for the Rotating Parts Use Case, cf. Deliverable D5.4 [24]) on the HPC systems available through LEXIS have been evaluated. This updates earlier results from Deliverable D5.1 [25] and Deliverable D5.2 [26]. It turns out that speed gains running TRAF and nanoFluidX on HPC systems are significant, in particular with GPUs (with more difficulties with FPGAs as of now), and the simulations can aid the industrial design processes in Aeronautics, as expected. More specifically, in the Turbomachinery Use Case, besides the analysis of the benefits of HW-acceleration in terms of computational time reduction, the assessment of the adopted GPU-equipped HPC solutions has been carried out from a cost perspective as well, providing beneficial outcomes also from that point of view. One specific focus of validation has been on HPC runs exceeding queue-time limits, which have been successfully implemented with HEAppE controlling restarts of the code when necessary. The Computing-Cloud parts of the LEXIS infrastructure were successfully tested for WP5 jobs in preprocessing (cf. Deliverable D5.4 [24]).

Validation of checkpointing and failover capability on LEXIS infrastructure

During turbomachinery simulation runs, TRAF writes a series of checkpointing files, to allow for a monitoring of the progressing simulation, and even more importantly, to allow for restarts/failovers in case of failure. Thanks to optimisation of the LEXIS DDI (cf. Deliverable D3.6 [23]), these files can now be conveniently managed within the LEXIS infrastructure (cf. Deliverable D5.4 [24]). Thus, a failover to another computing centre is possible.

Security prerequisites; data encryption and data safety by replication

The compliance of LEXIS infrastructure with internal security regulations of Avio Aero has been validated by means of Avio Aero security checklists (confidential material) filled in appropriately by data-centre partners within LEXIS. Accommodating requirements by Avio Aero, the LEXIS platform in its final state offers data encryption at rest as well as data replication for additional protection against data loss. These features have been tested by WP5.

Validation of visualisation features

Remote visualisation techniques have been used to post-process data from the Turbomachinery Use Case, but also to check data when pre-processing it for the Rotating Parts Use Case. Also, live monitoring of application process in the Turbomachinery Use Case has been successfully implemented with simple status visualisation. All these results are laid out within Deliverable D5.4 [24].

Validation of LEXIS orchestration system for running WP5 workflows on federated platform

The LEXIS orchestration system and federated platform has successfully been used to run WP5 workflows, as also showcased in a screenshot included in a recent conference publication on the LEXIS Platform [27].

3.2.2 WP6

Within WP6, validation has first focused on the ability of the LEXIS Platform to execute individual building blocks of the workflow, which are more diverse than within other Pilots (as already laid out in Deliverable D6.1 [28]). In particular, we have been considering database workloads, extreme-IO workloads and HPC workloads, all being optimized to the LEXIS Platform. Thus, we have validated the following aspects:

- HPC and Cloud-Computing jobs,
- Optimized data handling,
- We have assessed the ability of the federated platform to run deadline-based urgent-computing workloads.

Validation of workflow building blocks on LEXIS systems: HPC runs with TsunAWI and Cloud jobs

For execution on HPC clusters, TsunAWI [29] underwent major optimizations, facilitated by the possibility of using the LEXIS Platform. This serves as an important blueprint for onboarding more customers onto the LEXIS Platform - in particular customers with lacking experience on large HPC machines (supercomputers), for which code optimization projects in the surrounding of the LEXIS Platform may be conceptualized. The TsunAWI code was optimized with respect to single node execution, with a switch to single-precision arithmetic (where appropriate) providing a reduction of compute time by 30%. Also, MPI communication was implemented to obtain excellent weak and strong scalability. Thus, running TsunAWI on the LEXIS Platform for large setups, e.g. an ocean basin or the whole Indonesian Archipelago with inundation simulation along all coasts, is reduced from several hours to just a few minutes. For example, the LEXIS test case "Coquimbo fine" for the Chile tsunami 2015 could be executed on IT4I's KAROLINA supercomputer within 3:23 min on 20 nodes, while it takes more than 50 minutes on a single comparable server. Other parts within the WP6 workflows, such as the damage assessment computation, are ideal workloads for putting the LEXIS cloud infrastructure at test.

Validation of data handling on the LEXIS platform: database workload and I/O optimisation; DDI functionality

Within WP6 workflows (cf. Deliverable D6.1 [28]), a PostGIS database plays a central role in rapid assessment of expected damage by earthquakes and tsunamis. This database has been substantially optimized internally, but - probably even more significant in the LEXIS Platform context - it was explored on which LEXIS systems the database can be run. For rapid execution of database queries, in-memory databases or - alternatively - storage on NVMe (M.2) disks or in NVRAM are an attractive option. The LEXIS Platform offers such storage on virtual machines (in particular IT4I virtualization nodes) via the ATOS Smart Bunch of Flash solution (cf. e.g. [30]), running on the LEXIS Data Nodes / Burst Buffer Servers at IT4I and LRZ. This solution exports NVMe/NVRAM memory within these servers as storage volumes to external machines via NVMEoF. It can thus be said that the LEXIS Platform offers an optimum solution for hosting WP6 databases. As it appears essential to some customers (including WP6) to permanently host such a database, persistent over different workflows, such a service might complement the workflow orchestration offers of LEXIS. Within WP6, an informal practical solution was found.

The Data Nodes already mentioned can be used in a SBB (Smart Burst Buffer, cf. e.g. [30]) mode in order to buffer output data and pre-fetch input data. The buffer is flushed in the background, while the workflow continues. As described in more detail in Deliverable D3.6 [23], this setup was tested with the TsunAWI [29] post-processing routine for the final data reduction and output step. Increases in throughput and decreases in latency with respect to traditional storage backends are notable especially if these backends are of limited performance. Thus, buffering can be a longer-term strategy for keeping the time constrains of WP6 workflows (cf. last paragraph of this Section). The burst-buffer space is handled by I/O interception using pre-loaded libraries, so the burst buffer is transparent to the workflow code. Only a different orchestrator configuration must be provided to indicate the directory to be optimized and basic configuration parameters (storage address, etc.).

Besides these application-specific optimisations, WP6 has begun to test the federated DDI with test datasets.

Assessment of LEXIS orchestration system for running WP6 workflows on federated platform

To implement urgent computing, for earthquake and tsunami warnings in particular, on HPC centre/cloud infrastructure, deadlines are an important instrument in workflow control. Usually, results have to be obtained within a certain number of minutes by authorities in order to be efficiently used by civil protection departments. Thus, a very relevant model in this context is to try start crucial prediction steps in parallel with different accuracy on different computing facilities, in order to guarantee that at least one of these parallel steps be executed in time. Depending on the actual use case, the best result obtained in time is then selected or the first result obtained is chosen; the other results are discarded. From the orchestrator side, this requires an extension of the usual concept of job timeouts, which are anyway commonly implemented to cancel job submissions when some jobs are stalled.

With acceptable tweaks, it is possible to create deadlines in the LEXIS Orchestration System, such that WP6 workflows can be conveniently run.

3.2.3 WP7

Within WP7's work, the platform has been validated with complex workflows involving HPC and Cloud-Computing job in an "input data pre-processing - weather simulation - application models (forest fire hazard prediction, etc.)" scenario. Multiple data sources to be used for the weather simulations (as initial/boundary data and as weather station data to be assimilated) have been incorporated in the data flow. Producing large amounts of data (cf. e.g. [31]), the use case has tested almost all aspects of the LEXIS platform, including:

- Intensive interplay of (almost prototypical) HPC and Cloud tasks,
- Data handling using WCDA and DDI,
- Orchestration of workflows on the platform, making strong use of federation.

WP7 work has included a particularly strong focus on verification of the overall performance and reliability of the platform.

Validation of LEXIS infrastructure with first mixed HPC/Cloud-Computing workflows

In 2020, first versions of a WP7 workflow without the assimilation of current weather station data have been orchestrated on the LEXIS Platform. As shown in a poster [31] on SC20, the workflow validated the basic technical capabilities of the LEXIS Platform to orchestrate the required jobs and move the data accordingly. Based on weather simulations with WRF (after appropriate data pre-processing), the workflow yielded forest fire risk predictions with the help of the model RISICO. Aspects not yet validated have been assimilation of large amounts of data, including data from the TESEO smart gateway, and a multi-user triggering and configuration of workflows via the portal. The workflow shown on SC20 is reproduced in Figure 5.

Validation of data handling via WCDA and DDI

The workflows run within WP7 have successfully validated the principal capability of the platform to use data from many sources (including the TESEO smart gateway or similar, IoT-style sources) and to manage data within the workflows with the help of the WCDA (Weather and Climate Data API) and the DDI (Distributed Data Infrastructure). While no principal capabilities have been found missing, a few issues have been successfully uncovered and are being resolved, as a sign of the platform developing towards TRL 8 in 2021. From a functional point of view, the LEXIS data system thus covers the expected functionalities (only the rights management has not been validated in detail by WP7, as it is of minor importance here - with exception of workflows including commercial models such as ADMS, cf. [32]).

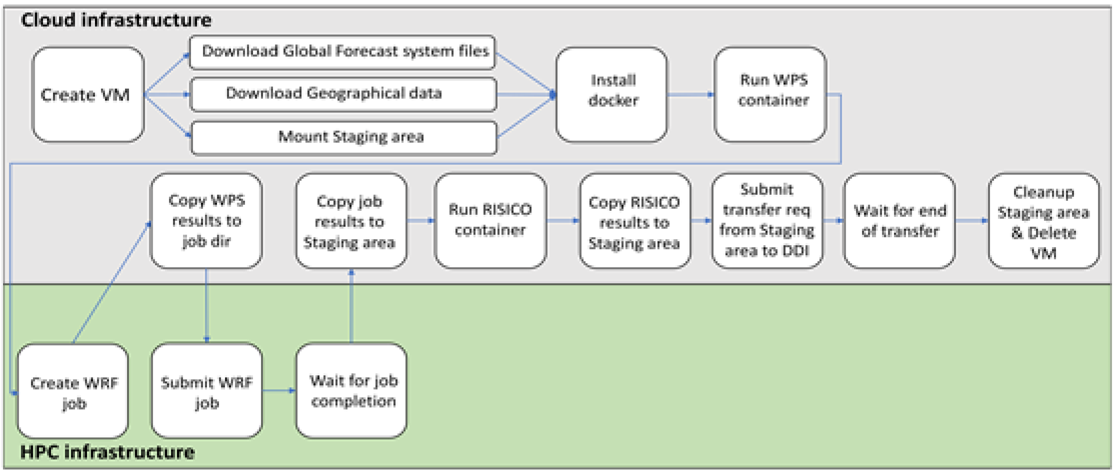

Cloud infrastructure
Create VM
Download Global Forecast system files
Download Geographical data
Mount Staging area
Install docker
Run WPS container
Copy WPS results to job dir
Copy job results to Staging area
Run RISICO container
Copy RISICO results to Staging area
Submit transfer req from Staging area to DDI
Wait for end of transfer
Cleanup Staging area & Delete VM
Create WRF job
Submit WRF job
Wait for job completion
HPC infrastructure

Figure 5: Weather and Climate Large Scale Pilot Workflow - one of the first versions, as shown on SC20

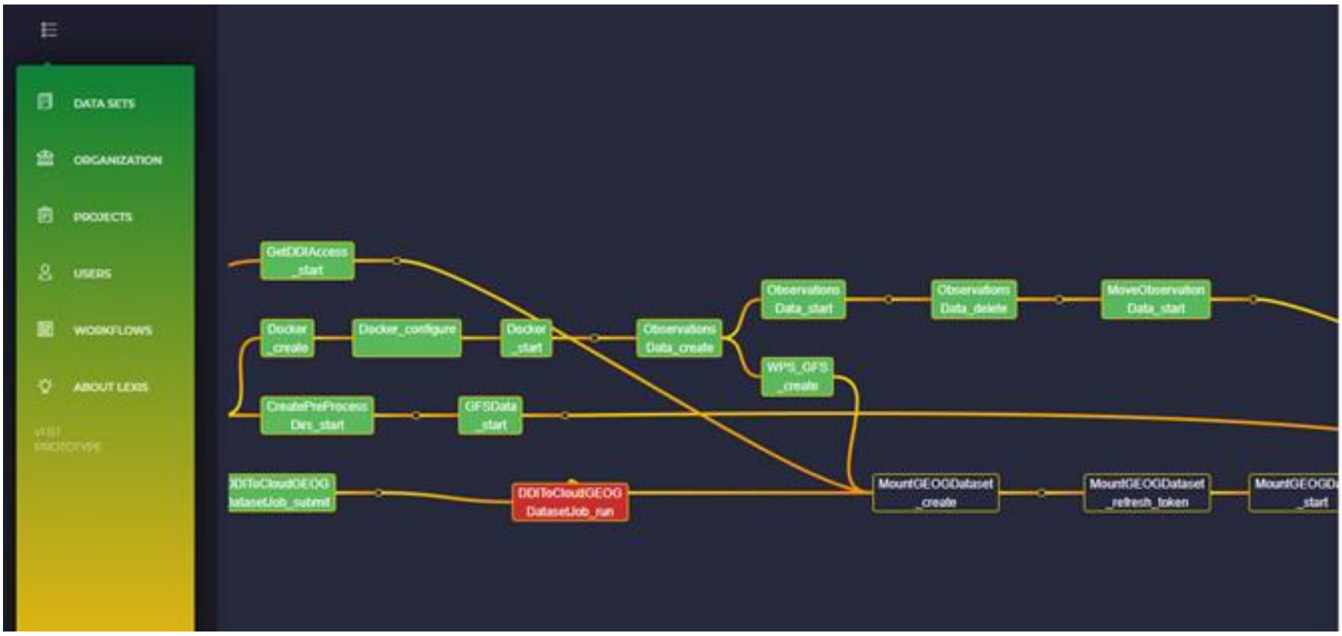


Figure 6: Part of an extended WP7 workflow in progress on the LEXIS Portal with an error discovered by the orchestrator in the step marked in red, and the green steps concluded

Validation of federated orchestration capabilities, reliability and speed of the platform

The workflows as described above have since been extended to include large-scale data assimilation from weather stations. Regarding application models, RISICO has been complemented by CONTINUUM (hydrological simulation), ADMS (particle dispersion) and ERDS (exceptional rainfall prediction). The status of workflows as of mid-2021 (with information on the modelling codes mentioned) has been described in a book chapter [32] and in various conference talks (e.g. at ISGC 2021).

After a period of workflows being executed by WP4 specialists and under preliminary shared accounts, WP7 has begun to start runs via the portal under proper users (multi-user scheme), where some manual rights adjustments have still been necessary (the problems necessitating this have, however, largely been eliminated in late 2021). Figure 6 shows a part of a complex, current WP7 workflow in progress on the LEXIS Portal workflow display.

Platform reliability has been identified - as envisaged within the project plan - as a major item to work on since summer 2021. Reliability issues with DDI and AAI system have thus been resolved. With the current experience, the platform is well usable; some further improvement on the reliability side can push it beyond TRL 8.

Execution speeds of reference workflows have been compared with execution speeds on a CIMA operational cluster for a few cases. It turns out that, despite the complexity of the LEXIS distributed computing and data infrastructure, execution speeds are not suffering any catastrophic slow-down. Time is lost in particular when data transfers happen. The respective staging processes have been somewhat optimized already (cf. Deliverable D3.6 [23]), and will be further optimized in the future. Thus, it is expected that the strengths of the distributed computing paradigm behind LEXIS (optimization of execution locations, automatic choice of free and capable resources, etc.) will pay out in the future.

3.3 LEXIS OPEN CALL VALIDATION AND EXTENSION OF THE PLATFORM TO ICHEC

As described in the previous Sections, large-scale “internal” platform validation efforts have been made in 2020/21. For a sustainable platform, however, more is needed (and has been done within LEXIS): a complementary validation from the perspective of end users – focused on general usability, and from the perspective of additional computing centres inclined to join the platform.

LEXIS Open Call and validation from end users’ perspective

As presented in Deliverable D9.12 [7], the LEXIS Open Call gives an opportunity, on top of other validations previously described, to get lessons learned, feedback from users and detection of bugs. As described also in Deliverable D2.5 [5], we have identified several actions to take after confronting the platform to real third-party users.

The Project Managers set-up on LEXIS side to support and monitor each applicant have a unique position to collect feedback on the user experience and to detect needs for debugging, all along each project life, and to report to the teams developing the LEXIS Platform. This process has been very efficient, e.g., for the LEXIS Portal, where the user experience is central to the development of this central component of the LEXIS Platform.

On a weekly basis, all Project Managers have been analysing the information collected and brought it to discussion in the scope of WP2/co-design. The beginning usage of the platform by “real life users” has helped us to remove obstacles and to verify that LEXIS actually lives up to what is expected: a platform which experts and non-experts can use to run their workflows of various kinds on mixed HPC/Cloud-Computing infrastructure.

Within the Pilot use cases, some functionalities of the platform have naturally not been be fully stress-tested, as these use cases were gradually on-boarded between 2019 and 2021 onto the growing platform and its data, workflow and user management (including the LEXIS identity management / AAI). In particular, lots of input data for the Pilots has been residing on the platform for a longer time already, and users, organisations and computational projects have been existing within the LEXIS platform and identity management. As expected, the test by new use cases built up from zero has thus uncovered e.g. some glitches within user and project creation mechanisms, and with the upload of datasets which are very large or consisting of several thousand files. The related problem reports and lessons learned have been helping us to significantly stabilise and improve the platform.

The LEXIS Open Call projects are currently ongoing, and at the end of each project, an interview of applicants and a questionnaire will serve to further validate very detailed aspects of LEXIS usability. An outlook on the impact of this LEXIS Open Call validation process is given in multiple deliverables including Deliverable D2.5 [5], Deliverable D8.4 [33], and Deliverable D9.12 [7].

Integration of new computing centres into the platform

With ICHEC joining the project consortium and the LEXIS platform, it has been demonstrated that the essential components of LEXIS, in particular the multitude of DDI components, can be successfully deployed at further sites, extending the LEXIS platform. Besides generating this very positive overall result, the on-boarding experience has helped us to further refine installation recipes and concepts.

4 DISSEMINATION OF RELEASES

LEXIS release management, as described in Section 2 of this Deliverable, and in earlier Deliverables referenced therein, goes hand in hand with a publication strategy for LEXIS software. LEXIS (sub-)module releases, having passed the necessary quality checks, are being delivered to GitHub, within a dedicated LEXIS group (in a few special, cases, such as YORC which has relevance outside LEXIS, original software repositories are clearly used and kept). This publicly showcases the openness of the platform and the feasibility of onboarding more sites.

In order to actively support the FAIR principles (findable, accessible, interoperable, reusable) for research data [34], including research software, LEXIS (sub-)module releases on GitHub are assigned Digital Object Identifiers (DOIs) via the Zenodo-Github DOI mechanism [35]. In addition, the GitHub page (https://github.com/lexis-project), and the Zenodo LEXIS community (https://zenodo.org/communities/lexis) will contain a platform-release data product, reflecting the current and further major platform releases. This product includes the most recent version of the LEXIS Platform/module scheme (cf. also Deliverable D2.5 [5]) and DOI-based links to the relevant LEXIS (sub-)module releases.

The current status and planning of LEXIS (sub-)module uploads and LEXIS software dissemination is further reflected in an extensive table within Deliverable D9.10 [6], pointing to the relevant repositories.

5 SUMMARY

With this deliverable, we have assessed the release status and the results of validation of the LEXIS Platform, as a basis for declaring MS8 (“Final integration and LEXIS technologies validation”) achieved. We have focused on release management and release/deployment status, on validation of the platform by technicians (WP2-4), Pilots (WP5-7) and external test customers (LEXIS Open Call). Furthermore, we have touched upon our efforts to extend the platform to more sites (including ICHEC) and to disseminate LEXIS software in collaboration with WP9. The release management and assessment work presented has been the result of Task 2.3 - while practically all other WP2 tasks and WP3-8 have generated the results and systems which have been the basis for completing Task 2.3.

From a project management perspective, the LEXIS Platform can be regarded as fully integrated (in particular, also the data system and the orchestration solution interoperate well), and a benchmarking/validation effort on the platform has made it possible to eliminate or mitigate problems and bottlenecks. An assessment of the co-designed architecture is carried out in Deliverable D2.5 [5].

The LEXIS Platform has been organized into functional modules (cf. Deliverable D2.4 [4] and the present deliverable), which correspond to code/documentation repositories and release documentation for each module. This systematic approach, developed within the co-design effort, has allowed us to have a consistent release documentation of the platform, including change logs and release plans. The current release status has been illustrated in this deliverable.

Internal validation has shown that functionalities are overall available as expected; the concrete execution of validation scenarios had uncovered some problems which were resolved, e.g. by corrections in the role-based access control model of the platform and/or rights implementation. Security assessment and speed tests have given positive and encouraging results.

In the scope of the Pilot use cases, remarkable and very pleasant improvements in workflow and/or workflow-step (code) execution have been achieved with respect to the pre-LEXIS status. All in all, the LEXIS Platform with its large accumulated computing power can be considered very useful for all workflows from WP5-7. Given this result, in later 2021 we have been concentrating on improving reliability (removal e.g. of principal points of failure identified in distributed data and compute infrastructure) and resilience. This corresponds to the platform status advancing near to TRL 8 over 2021, in alignment with the project plan. A principal and expected bottleneck in workflow execution speed are data transfers, where optimization has made us achieve significant speed gains and will continue to do so (cf. Deliverable D3.6 [23]). Apart from this fact, workflows execute well and are able to take advantage of the distributed infrastructure with its accumulated computing power.

The LEXIS Open Call users have been contributing very valuable further feedback, validating the LEXIS user interface (portal) and basic infrastructure functionality “from scratch”. Owing to this extensive testing, some more glitches have been eliminated and the users are now evaluating the platform further running their workflows. This important step towards a sustainable operation of the platform and extension of the user base pleasantly concludes our work within the LEXIS H2020 project run-time. We are convinced that it has well prepared the LEXIS platform for the future.

REFERENCES

[1] LEXIS Deliverable, *D2.1 Pilots needs / Infrastructure Evaluation Report.*

[2] LEXIS Deliverable, *D2.2 Key parts LEXIS Technology Deployed on Existing Infrastructure and Key Technologies Specification.*

[3] LEXIS Deliverable, *D2.3 Report of LEXIS Technology Deployment - Intermediate Co-Design.*

[4] LEXIS Deliverable, *D2.4 Report of LEXIS Technology Deployment - Updated Test-Beds Infrastructure.*

[5] LEXIS Deliverable, *D2.5 Final Assessment of the Co-Designed LEXIS Architecture.*

[6] LEXIS Deliverable, *D9.10 Impact KPI and Metrics Achievements Report and Plan - final version.*

[7] LEXIS Deliverable, *D9.12 Open Call Framework and Stakholders Engagement on Targeted Large-Scale Pilots - final.*

[8] LEXIS Deliverable, *D4.5 Definition of Mechanisms for Securing Federated Infrastructures.*

[9] LEXIS Deliverable, *D4.7 Centralized AAI: Coverage of All Significant Systems.*

[10] “Pylama,” [Online]. Available: https://github.com/klen/pylama. [Accessed 22.12.2021].

[11] “Pylint,” [Online]. Available: https://pylint.org/. [Accessed 22.12.2021].

[12] “Pycodestyle,” [Online]. Available: https://pycodestyle.pycqa.org/en/latest/. [Accessed 22.12.2021].

[13] “Pyflakes,” [Online]. Available: https://github.com/PyCQA/pyflakes. [Accessed 22.12.2021].

[14] "GitLab Secret Detection," [Online]. Available: https://docs.gitlab.com/ee/user/application_security/secret_detection/. [Accessed 22.12.2021].

[15] “Elasticsearch Guide,” [Online]. Available: https://www.elastic.co/guide/en/elasticsearch/reference/current/index.html. [Accessed 23.12.2021].

[16] "OpenVAS," [Online]. Available: https://www.openvas.org/). [Accessed 22.12.2021].

[17] "DNSdumpster," [Online]. Available: https://dnsdumpster.com/. [Accessed 22.12.2021].

[18] "Sublist3r," [Online]. Available: https://github.com/aboul3la/Sublist3r. [Accessed 22.12.2021].

[19] "Amass," [Online]. Available: https://github.com/OWASP/Amass. [Accessed 22.12.2021].

[20] "Pentest-Tools," [Online]. Available: https://github.com/blackhatethicalhacking/Pentest-Tools. [Accessed 22.12.2021].

[21] "Outpost24," [Online]. Available: https://outpost24.com/. [Accessed 22.12.2021].

[22] "ipref3," [Online]. Available: http://software.es.net/iperf/. [Accessed 22.12.2021].

[23] LEXIS Deliverable, *D3.6 Data Flow Optimisation and Data System Core.*

[24] LEXIS Deliverable, *D5.4 Avio Aero use cases: Critical Review to Highlight Benefits and Limits by Operating on Advanced HPC Solutions.*

[25] LEXIS Deliverable, *D5.1 Turbomachinery Use Case: Analysis of Results Run on State-of-Art HPC System.*

[26] LEXIS Deliverable, *D5.2 Rotating Parts Use Case: Analysis of Results Run on State-of-Art HPC System.*

[27] Hachinger, S., et al., "HPC-Cloud-Big Data Convergent Architectures and Research Data Management: The LEXIS Approach," in *Challenges in High Performance Data Analytics: Combining Approaches in HPC, HTC, Big Data and AI, International Symposium on Grids & Clouds 2021 (ISGC 2021), Proceedings of Science, 378, 004*, Taipei, Taiwan, 2021.

[28] LEXIS Deliverable, *D6.1 Baseline scenarios and requirements.*

[29] N. Rakowsky, A. Androsov, A. Fuchs, S. Harig, A. Immerz, S. Danilov, W. Hiller and J. Schröter, "Operational tsunami modelling with TsunAWI – recent developments and applications," *Natural Hazards and Earth System Sciences, 13,* pp. 1629-1642, 2013.

[30] Munke, J., et al., "Data System and Data Management in a Federation of HPC/Cloud Centres (Ch. 4)," in *HPC, Big Data, and AI Convergence Towards Exascale*, Boca Raton FL (USA), CRC Press / Taylor & Francis, 2021, in production (ISBN 9781032009841).

[31] Hayek, M., et al., "Orchestration of a Forecasting Chain for Forest Fire Prevention Using the LEXIS Cloud/HPC Platform," in *SC20, The International Conference for High Performance Computing, Networking, Analysis and Storage - Research Posters*, online conference, 2020. Available: http://sc20.supercomputing.org/proceedings/tech_poster/tech_poster_pages/rpost120.html. [Accessed 23 12 2021].

[32] Parodi, A., et al., „Exploitation of Multiple Model Layers Within LEXIS Weather and Climate Pilot: An HPC-Based Approach (Ch. 8)," v *HPC, Big Data, and AI Convergence Towards Exascale*, Boca Raton FL (USA), CRC Press / Taylor & Francis, 2021, in production (ISBN 9781032009841).

[33] LEXIS Deliverable, *D8.4 Roadmap for Further Development of the LEXIS Portal.*

[34] Wilkinson, M., et al., "The FAIR Guiding Principles for scientific data management and stewardship," *Sci Data 3, 160018,* 2016.

[35] M. Potter and T. Smith, "Software Sustainability Institute Blog - Making code citable with Zenodo and GitHub," [Online]. Available: https://doi.org/10.5281/zenodo.45042. [Accessed 23.12.2021].