D7.6 Second Report on ICS-D Requirements Definition

Document information Summary

Date	17 September 2018
Document title:	Second Report on ICS-D Requirements Definition
Leader Partner	UKRI (BGS)
Main Author(s):	Keith Jeffery, Matt Harrison, Daniele Bailo, Kuvvet Atakan
Contributing author(s):	Chris Card
Reviewer(s):	-
Approved by:	PMO; Implementation Phase Council
Target audiences:	Project partners, European Commission
Keywords:	EPOS IP; ICS-D requirements.
Deliverable nature:	Key Deliverable; Report
Dissemination level:	Confidential
Delivery date:	M36
Version:	1.2

TABLE OF CONTENTS

Contents

SUMMARY

ICS-D is the distributed component of ICS. ICS-D characteristics are documented in the metadata catalog and ICS-C utilises the user request and the catalog to generate a workflow to be deployed and executed across ICS-D using computing resources in the TCS environment and the e-I (e-Infrastructure) environment external to EPOS. ICS-D may be a distributed computing or sensor platform (whether in the field or in a laboratory) and also may be specialised software services integrated with a distributed computing platform e.g. for analytics, simulation or visualisation.

We specify the updated elements of the characteristic description of ICS-Ds to be included in the catalog and the plan for the integration.

1. Introduction

The purpose of this deliverable is to document the definition of requirements for ICS-D (Distributed ICS: Integrated Core Services) being integrated within the ICS-C (Central ICS) within the EPOS-IP architecture (D7.2).

2. ICS-D

Introduction

The ICS-C of EPOS is the entrypoint and as such consists of a metadata catalog describing EPOS assets (software services, datasets, workflows, facilities, equipment) to interact with the user (through the GUI: graphical user interface) and programmatic access (APIs) together with components to construct workflows from the user request and manage access to the TCSs – all connected by a bus architecture with messaging. Some workflows require more computing resource than that available within the facilities of the relevant TCSs. In this case additional resource is required from the e-Is (e- Infrastructures). To preserve a homogeneous interface shielding the ICS-C portal and its users from the heterogeneity of the e-Is, the concept of ICS-D (Integrated Core Services – Distributed) was created. During the period M18-M36 the concept was extended to include specialised software services distributed over distributed computing resources. Such services may include analytics, simulation, visualisation.

Concept

The ICS-D concept is to extend the ICS-C in a virtualised way with additional computing facilities to support the deployed workflows when their resource requirements exceed those of the relevant TCSs. Additionally, if assets from several TCSs are to be used together it may be convenient to transport them all (after suitable selection and projection of data and selection of software components) to one or more e-Is in order to relieve resource loading on the TCSs and to minimise data transport and latency. This should all be managed by the workflow management component of ICS-C. However, non-functional requirements – restrictions concerning privacy, security, performance, cost, rights etc. – have to be taken into account in the deployment and may preclude the use of ICS-D linked to e-Is for some deployments.

The linkage between ICS-C on the one hand and the e-Is and TCS local computing resources and assets on the other (ICS-D) is the deployment phase. The workflow for the deployment is generated within the ICS-C by interaction with the user. The workflow is checked by the end-user before deployment. However, the detailed content/capability of the assets is not known e.g. the dataset may not contain the relevant information despite its metadata description or the software may not execute as the user expects despite the metadata description. The execution of the deployment is monitored and execution information is returned to the end-user. The workflow may be deployed in one of two ways: (a) directly with no user interaction during execution of the deployment; (b) step-by-step with user interaction (so-called computational steering) between each step. Deployments of type (a) will have better optimisation (for performance) and security but could possibly execute a workflow the components of which do not behave as the user expects. Deployments of type (b) lack optimisation but allow the user to stop the workflow deployment at any step, examine the results and – if not as expected – reorganise the workflow (by changing components) to meet more closely the requirement.

The more recently defined type of ICS-D, namely a specialised software service, is defined in the metadata catalogue of ICS-C like any other software service although the coupling with offered distributed computing resources (or sensors or equipment) is recorded to ensure availability for deployment.

3. Requirements of ICS-D to Enable Integration

Introduction

In order to integrate ICS-Ds with the ICS-C it is necessary to have sufficient information in the catalog – and appropriate software in the ICS-C – to compose the access to the ICS-D into a workflow deployment. Current joint projects between EPOS and various e-Is (such as EGI, PRACE, EUDAT) continue to refine the interface parameters required and documenting them for the design of ICS-D within EPOS.

Requirements

This deliverable concentrates on ICS-D requirements. However, for EPOS to offer an integrated homogenizing environment to the end-user, CES (Computational Earth Science) and VRE (Virtual Research Environment) also have to be considered with the requirements.

ICS-D

The requirements for an ICS-D to be included within the EPOS ICS-C catalog and thus made available include one or more of the following (depending on the type of ICS-D):

- Provide one or more API-type interfaces to be used by ICS-C in constructing deployments;
- Allow HPC (High Performance Computing) /HTC (High Throughput Computing) access;
- Allow run-time access to external data providers;
- Provide storage for data in run-time;
- Provide storage for data and results;
- Provide portability (choices by user to external or internal hardware resources);
- Provide analysis tools (for composition into a workflow or as a service);
- Provide visualization tools (for composition into a workflow or as a service);
- Provide simulation tools (for composition into a workflow or as a service);
- Provide user accounting, authentication and authorization tools linked to those of ICS-C;
- Provide space for multiple users using a single VRE but multiple deployments (at least one per user);
- Provide API for connecting to RI-e-infrastructures at run-time;
- Provide procurement policies (including licencing, access rules, price lists for services) as a local catalog to be harvested by ICS-C;

CES

The requirements from the EPOS CES community need to be covered by ICS-D since CES is a specialisation of ICS-D for a particular community using particular computing (or sensor or laboratory) resources and particular software.

The descriptions provided by the EPOS CES imply a user interaction step-by-step in the process of satisfying the user request – including accesses to RIs and e-Is as well as the ICS-C catalog. An alternative solution is for the user to interact with the ICS-C catalog to generate interactively a complete specification for a workflow which is then assembled (may be implemented manually to start with but later increasingly implemented automatically) and deployed across multiple RIs and e-Is (i.e. facilities of the TCSs and ICS-D). The latter is the preferred solution for the ICS-C and ICS-D interactions.

VRE

Requirements for a VRE focus on a user interface which accesses RIs (which in turn access e-Is), provides researcher communication, administrative and management support plus access to all the assets of the RIs and resource capabilities of the e-Is. The assets of RIs and capabilities of e-RIs are provided by the catalog metadata entries.

Governance

Currently developing a plan for integration of selected ICS-D with the ICS-C is difficult to handle due to the constantly changing landscape of large cloud initiatives in Europe. The EPOS WP7 team is working closely with EOSC (European Open Science Cloud) through the EOSC Pilot and EOSC hub projects and proposals for future projects, including the recently approved ENVRI-FAIR (INFRAEOSC-04). However, the team is also gaining experience of working with PRACE (for supercomputing) and commercial cloud suppliers.

EPOS needs to develop:

(a) a strategy for preferred suppliers of computing (also sensor, laboratory) resources as ICS-Ds;
(b) a strategy for preferred suppliers of specialised software services as ICS-Ds;
(c) clear procurement policies to provide an open and transparent tender process for suppliers to apply as ICS-D.

4. The Integration Plan

The first step is for an ICS-D provider to apply to become an EPOS ICS-D. This process includes a validation to ensure the metadata is appropriate and that the provider has appropriate governance in place including availability and sustainability of the offering.

The second step is for metadata describing/characterizing each ICS-D to be loaded (by harvesting or uploading) into the test system ICS-C metadata catalog, ready for discovery, contextualization and action (such as composing into a workflow or executing) by the end-user performing the validation tests or software assisting the end-user.

The third step is for metadata describing/characterizing each ICS-D to be loaded (by harvesting or uploading) into the production system ICS-C metadata catalog, ready for discovery, contextualization and action (such as composing into a workflow or executing) by the end-user or software assisting the end-user.

For this integration to perform adequately, the ICS-C software must be ready to manage the metadata appropriately and assist the user in executing a service or composing a workflow from various assets and services as required.

Access to ICS-D assets and services is best viewed through the GUI

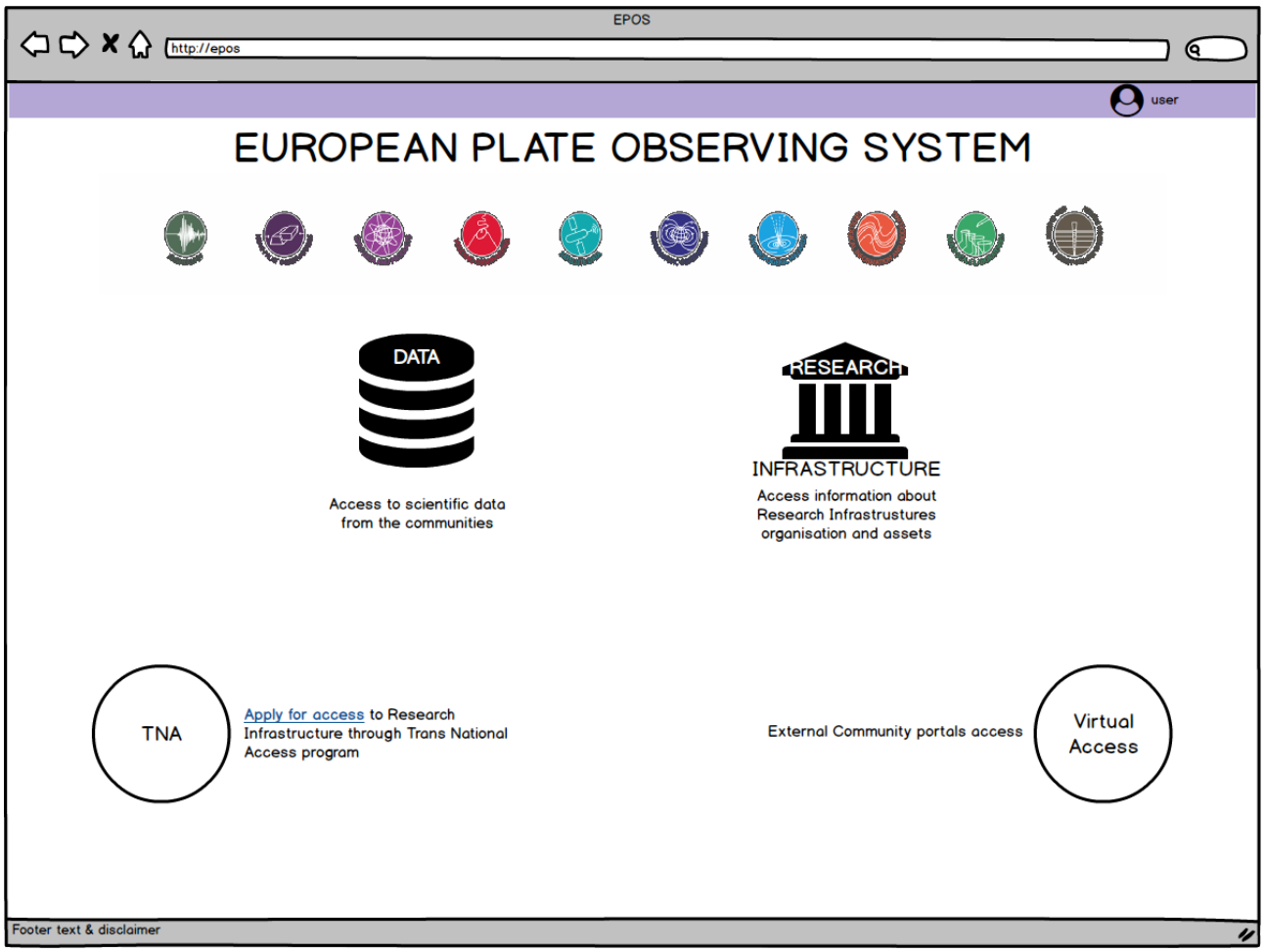

Figure 1 Landing Page

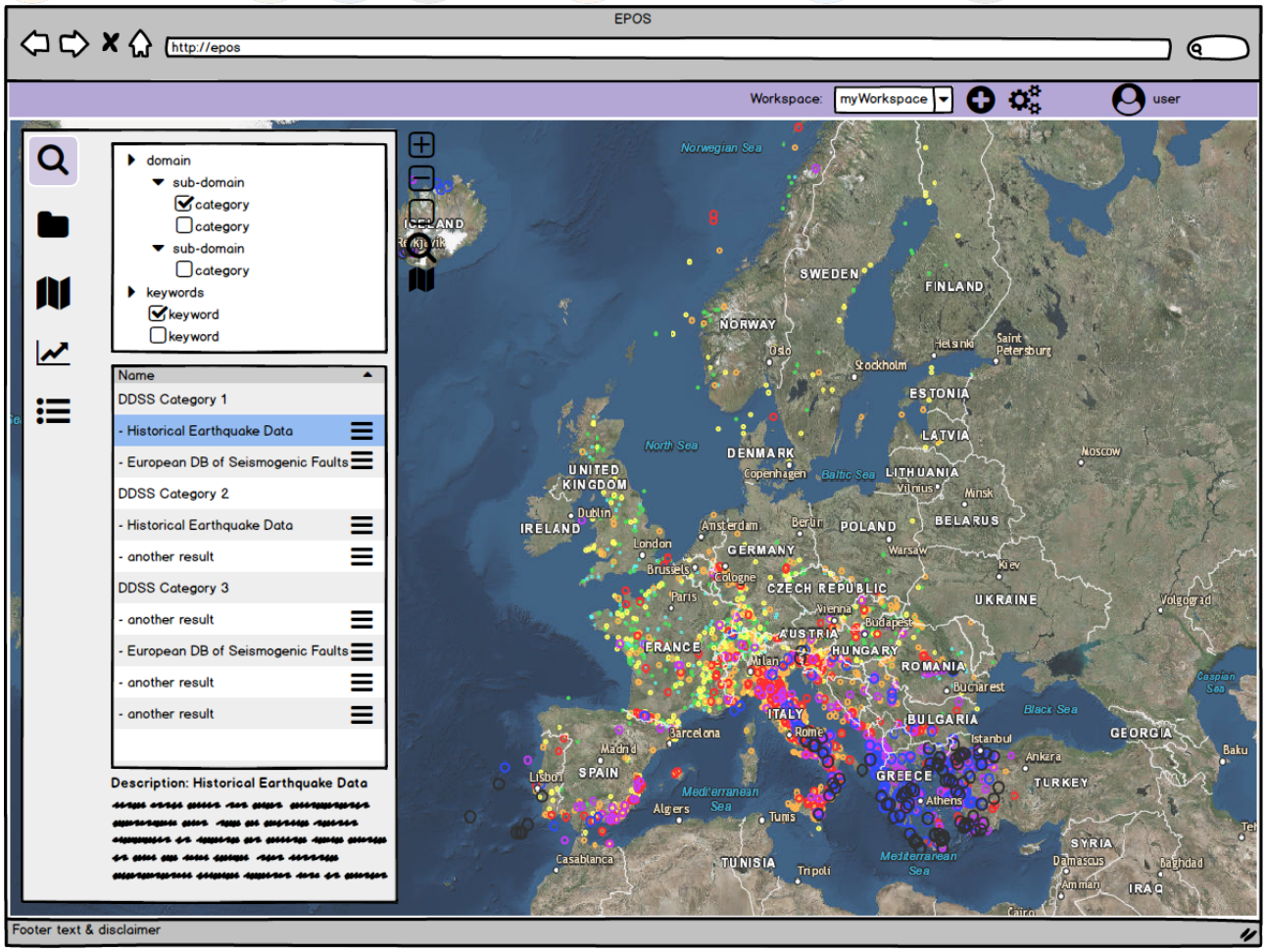

Figure 2 Faceted Search

In general ICS-D will be encapsulated within services offered by the TCSs and recorded as metadata in the ICS-c catalog. However, it is possible to envisage that at some time an end-user may wish to construct a workflow using various services selected from the catalog and a particular ICS-D (e.g. a particular supercomputer facility). This is analogous to use of facility/equipment and we shall find a solution that is congruent across both these aspects as the development of ideas (particularly on governance, legal and financial aspects) on TNA (trans-national access) develop.

Having discovered ICS-D services via the user interface, the ICS needs to provide additional software components that help the user utilise such facilities. To do this the ICS user interface has the concept of workspaces where users compile (almost like a shopping cart) a record of resources they have discovered using the ICS. An example is given below.

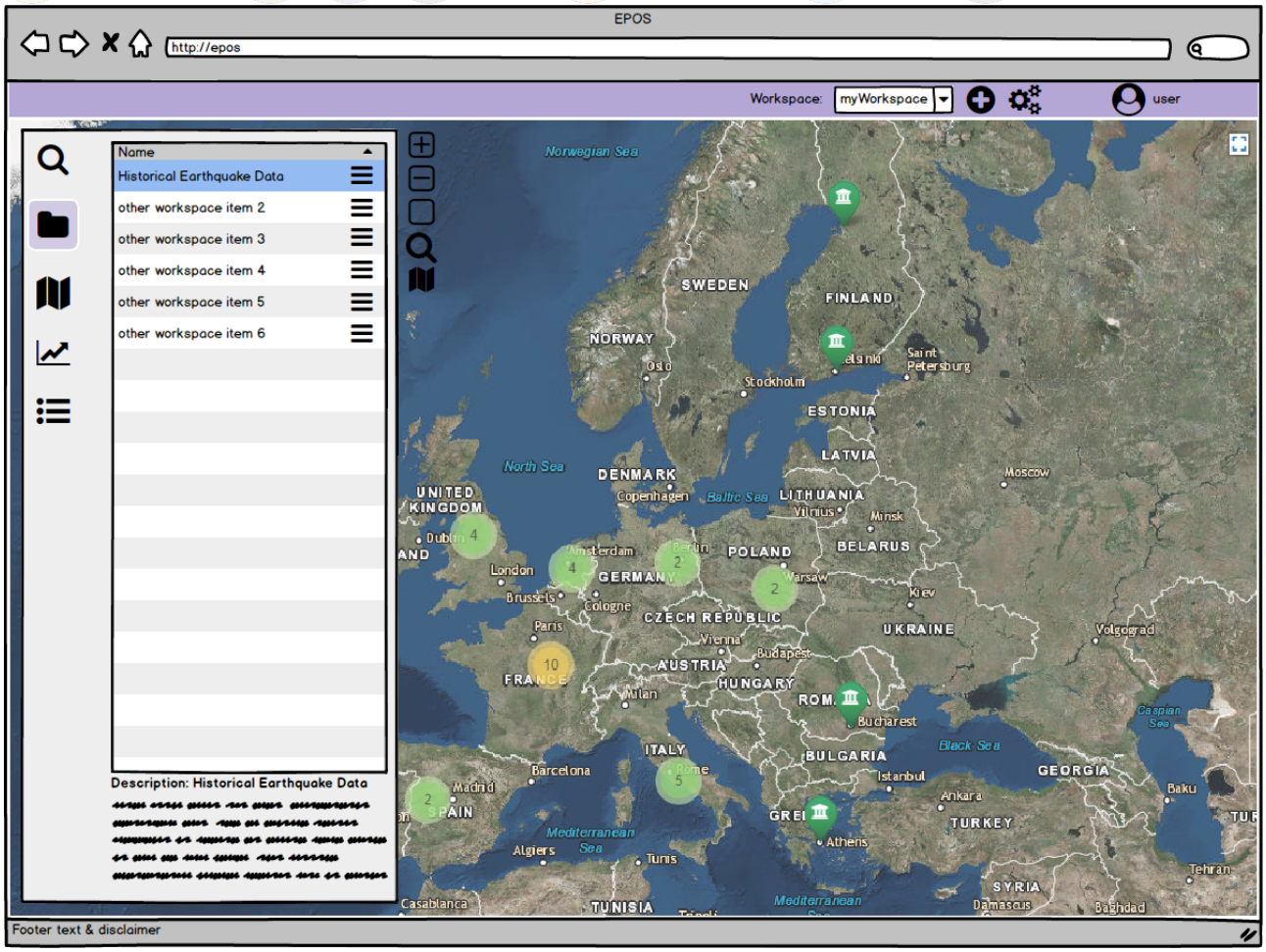

Figure 3 Workspace Contents

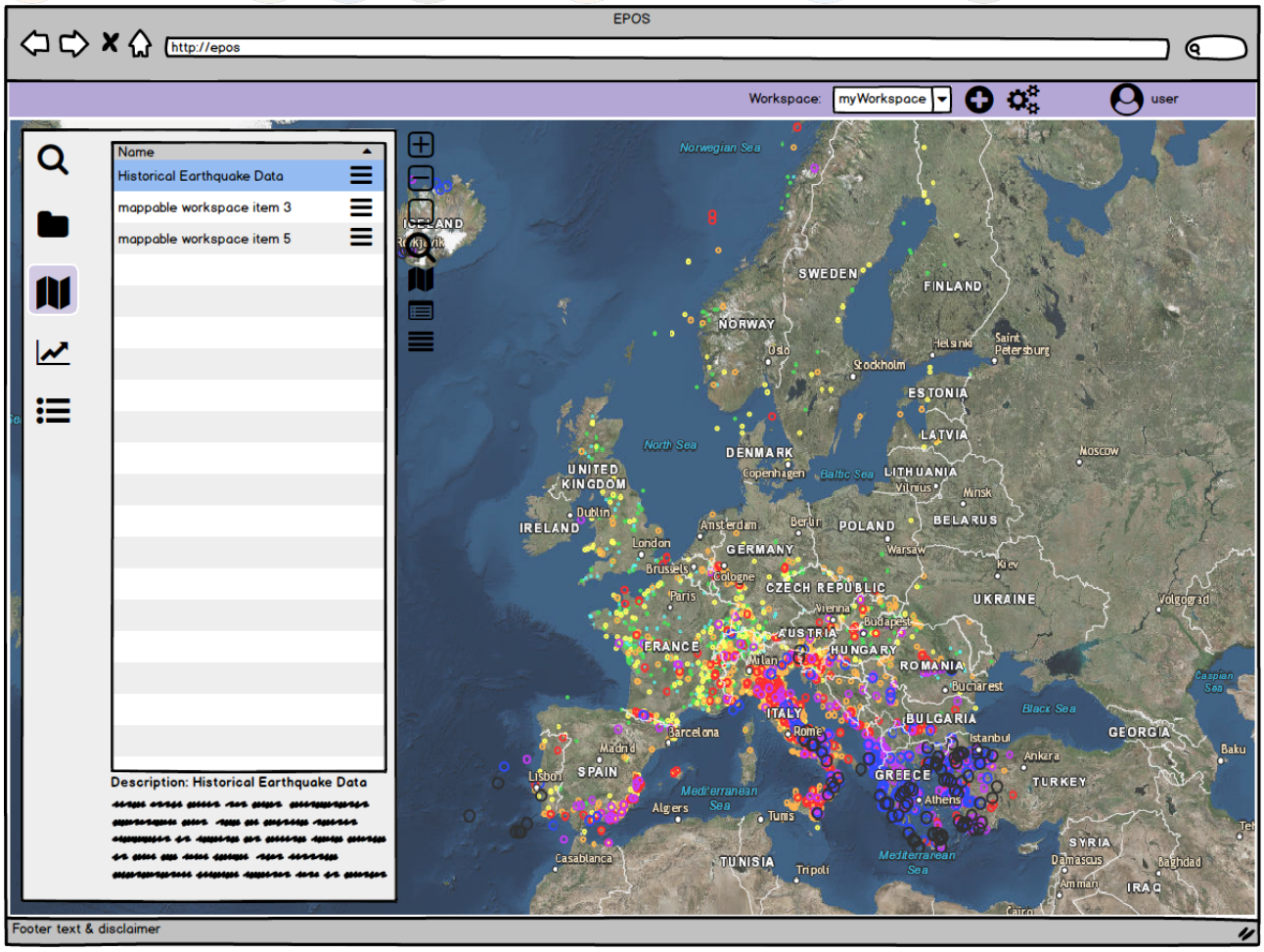

Figure 4 Mappable Workspace Content

Having compiled a list of resources, the ICS then offers a range of actions that are appropriate to the resources gathered together in the workspace. For example, if the workspace consists of a series of mapping data services, then the user is provided with an option to interact with a spatial visualisation of those datasets.

The ICS has the concept of enabling users to build up a processing model utilising the resources they have discovered and captured in their workspace. Such resources might include datasets, computational models, computing resources on which to run the models (as might be provided by an ICS_D node) and data centres which would storage and manage the model outputs. Such a process has been conceptualised in the ICS user interface as shown below.

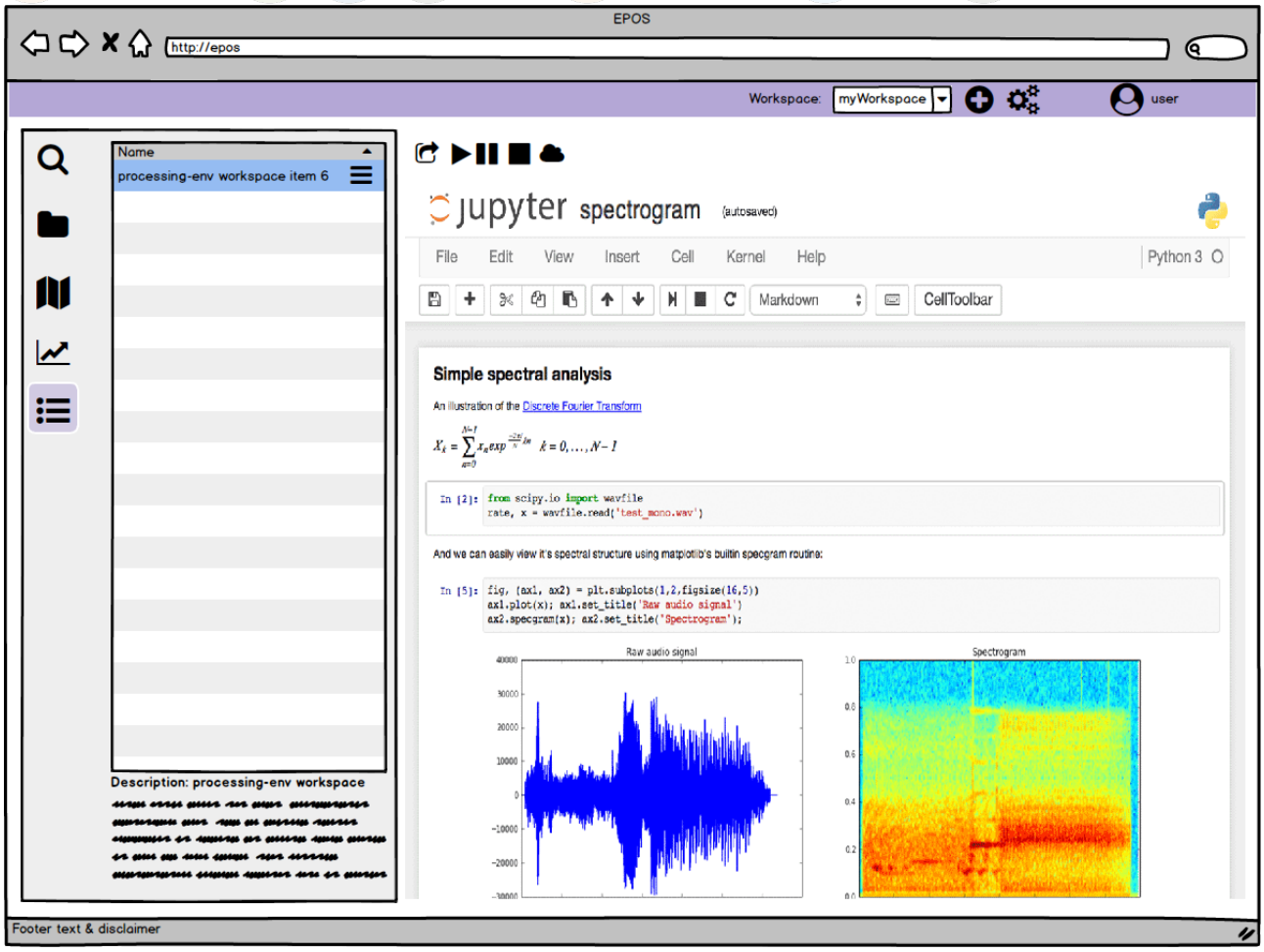

Figure 5 Workspace Processing Environment

This design and initial development needs to be realised within the ICS during the pre-production phase (M36-M48). To do this the following stages are required:

1. ICS-D facilities need to be identified;
2. They need to be appropriately described by CERIF compliant metadata and loaded in to the ICS metadata catalogue;
3. The ICS user search interface needs to be extended to enable the discovery of such facilities.
4. The ICS workflow functionality needs to be provided by the development of software that enables users to combine discovered data and computational processes with ICS-D facilities such as HPC, arrange and schedule usage of the facility and be able to access the results on completion.

This is a key activity for the ICS development team over the next 6 months. Building on the solid foundations already established, the team are excellently position to realise this goal.

5. CONCLUSION

The required metadata elements for ICS-D description have been defined (ANNEX) and are being validated within the ICT team. The ICS-C portal user interface is being further developed for

requirements including ICS-D. Once this connection is tested, the parameters required for the workflow manager will be defined and included in the catalogue by collection from the ICS-Ds.

ANNEX 1 Metadata Description of ICS-D

EPOS-IP		ICS-D	Metadata Elments Required in EPOS ICS-C Metadata Catalog (to be mapped to CERIF entities/attributes)			20180726
PLATFORM					note facility is one or more platforms under one	
	identity	UUID			maybe federated IDs e.g. local ID	cfEquip.cfEquipId
	platformname	name			may be federated / may be multilingual	cfEquip-EquipName
	platform acronym	acronym				cfEquip.cfAcro
	platform description				e.g. model or version	cfEquipDescr
	plaform keywords				from restricted vocabulary	cfEquipKeyw
	eqpt is a platform	classification			platform	cfClass
	platform kind	classification			e.g PaaS,	cfClass
	location	coordinate pair			spatial coordinates	cfEquip_PAddr
	owner	UUID, orgname			UUID/name/role owner/Dtstart/Dtend	cfOrgUnit_Equip
	manager	UUID, persname			UUID/name/role manager/DTstart/DTend	cfPers_Equip

	supplier	UUID, org name			UUID/name/role supplier/Dtstart/Dtend	**cfOrgUnit_Equip**
	[provenance]				derived from history of changes	
	Hardware				specification of capability	**cfEquip_Indic**
		CPU	minCPU	maxCPU	(do we need to distinguish GPU?)	**cfIndic**
						cfIndic_Class
		cores	mincores	maxcores		(repeated)
		RAM	minRAM	maxRAM		(repeated)
		Storage	minSTOR	maxSTOR		(repeated)
		Storage trabsfer	minSTORTRA	maxSTORTRA	mb/sec	(repeated)
	Container	type			linked to supplier	(repeated)
	VM	flavour			linked to supplier	(repeated)

	OS	name	version		linked to supplier	(repeated)
	software languages	name	version		linked to supplier	(repeated)
	data file structures	name	version		linked to supplier. Examples Hadoop	(repeated)
	Access performance				offered by the platform (especially real time/near real time)	
		network	minTRANS	meanTRAns	transfer speeds in mb/sec	(repeated)
		network	minLAT	meanLAT	latency in ms	(repeated)
	Access restrictions				in place for the platform	
		regUSER	userID	userPW		requires discussion with AAAI
		certUSER	certID	certKIND		requires discussion with AAAI
	Cost				costs in euros	
		costCPUsec				(repeated)

		costRAMMBsec				(repeated)
		costSTORMBday				(repeated)
SENSORNETSERVER					note facility is sensornet server plus sensors	
	identity	UUID			maybe federated IDs e.g. local ID	**cfEquip.cfEquipId**
	platformname	name			may be federated / may be multilingual	**cfEquip-EquipName**
	platform acronym	acronym				**cfEquip.cfAcro**
	platform description				e.g. model or version	**cfEquipDescr**
	plaform keywords				from restricted vocabulary	**cfEquipKeyw**
	eqpt is a sensornet server	classification			sensornet server (i.e. edge/fog)	**cfClass**
	platform kind	classification			e.g. 16-bit server	**cfClass**
	location	coordinate pair			spatial coordinates	**cfEquip_PAddr**
	owner	UUID, orgname			UUID/name/role owner/Dtstart/Dtend	**cfOrgUnit_Equip**
	manager	UUID, persname			UUID/name/role manager/DTstart/DTend	**cfPers_Equip**
	supplier	UUID, org name			UUID/name/role supplier/Dtstart/Dtend	**cfOrgUnit_Equip**
	[provenance]				derived from history of changes	
	Hardware				specification of capability	**cfEquip_Indic**

		CPU	minCPU	maxCPU		**cfIndic**
		cores	mincores	maxcores		**cfIndic_Class**
		RAM	minRAMMB	maxRAMMB		(repeated)
		Storage	minSTORMB	maxSTORMB		(repeated)
		Storage trabsfer	minSTORTRA	maxSTORTRA	mb/sec	(repeated)
	OS	name	version			(repeated)
	software languages	name	version			(repeated)
	data file structures	name	version		linked to supplier. Examples csv	(repeated)
	Access performance				offered by the platform (especially real time/near real time)	

		network	minTRANS	meanTRAns	transfer speeds in mb/sec	(repeated)
		network	minLAT	meanLAT	latency in ms	(repeated)
	Access restrictions				in place for the platform	
		regUSER	userID	userPW		requires discussion with AAAI
		certUSER	certID	certKIND		requires discussion with AAAI
	Cost				costs in euros / hour to utilise the sensor network	(repeated)
SENSOR/DETECTOR	identity	UUID			maybe federated IDs e.g. local ID	**cfEquip.cfEquipId**
	platformname	name			may be federated / may be multilingual	**cfEquip-EquipName**
	platform acronym	acronym				**cfEquip.cfAcro**
	platform description				e.g. model or version	**cfEquipDescr**
	plaform keywords				from restricted vocabulary	**cfEquipKeyw**
	eqpt is sensor/detector				sensor/detector	**cfClass**
	platform kind	classification			e.g. kind of seismometer	**cfClass**
	location	coordinate pair			spatial coordinates	**cfEquip_PAddr**
	owner	UUID, orgname			UUID/name/role owner/Dtstart/Dtend	**cfOrgUnit_Equip**
	manager	UUID, persname			UUID/name/role manager/DTstart/DTend	**cfPers_Equip**
	supplier	UUID, org name			UUID/name/role supplier/Dtstart/Dtend	**cfOrgUnit_Equip**

	[provenance]				derived from history of changes	
	Hardware	what parameters are useful?			specification of capability	**cfEquip_Indic**
						cfIndic
						cfIndic_Class
						(repeated)
						(repeated)
					mb/sec	(repeated)
	OS	name	version			(repeated)
	software languages	name	version			(repeated)
	data file structures	name	version		linked to supplier. Examples timeseries 16-bit	(repeated)

	Access performance				offered by the platform (especially real time/near real time)	
		network	minTRANS	meanTRAns	transfer speeds in mb/sec	(repeated)
		network	minLAT	meanLAT	latency in ms	(repeated)
	Access restrictions				in place for the platform	
		regUSER	userID	userPW		requires discussion with AAAI
		certUSER	certID	certKIND		requires discussion with AAAI
	Cost				costs in euros / hour to utilise the sensor	(repeated)
SOFTWARE SERVICE					may be offered bundled with platform or independent	
	identity	UUID			maybe federated IDs e.g. local ID	**cfResProd.cfResProdId**
	product is a software service					**cfClass**
	softwareservicename	name			may be federated / may be multilingual	**cfResProdName.cfName**
	Software service description					**cfResProdDescr**
	software service keywords				from restricted vocabulary	**cfResProdKeyw**
	location	coordinate pair			spatial coordinates	**need cfResProd_Paddr, cfPaddr-GeoBBox or via orgunit of owner**

	owner	UUID, orgname			UUID/name/role owner/Dtstart/Dtend	**cfOrgUnit_ResProd**
	manager	UUID, persname			UUID/name/role manager/DTstart/DTend	**cfPers_ResProd**
	supplier	UUID, org name			UUID/name/role supplier/Dtstart/Dtend	**cfOrgUnit_ResProd**
	licence documentation				a document	**cfResPubl**
	use condition				e.g. CC-BY	**cfClass**
	embargo				handled by cfClass of role linking owner to product and	**cfResProd_cfOrg**
	[provenance]				derived from history of changes	
	Hardware Required				to compare with platform offering	**cfEquip_Indic**
		CPU	minCPU	maxCPU	(do we need to distinguish GPU?)	**cfIndic**
		cores	mincores	maxcores		**cfIndic_Class**
		RAM	minRAMMB	maxRAMMB		(repeated)
		Storage	minSTORMB	maxSTORMB		(repeated)
		Storage trabsfer	minSTORTRA	maxSTORTRA	mb/sec	(repeated)

	Container required	type				(repeated)
	VM required	flavour				(repeated)
	OS required	name	version			(repeated)
	software languages required	name	version			(repeated)

	data file structures	name	version		linked to supplier. Examples Hadoop	(repeated)
	Access performance required				to compare with platform offering	
		network	minTRANS	meanTRAns	transfer speeds in mb/sec	(repeated)
		network	minLAT	meanLAT	latency in ms	(repeated)
	Access information required				to compare with platform offering	
		regUSER	userID	userPW		requires discussion with AAAI
		certUSER	certID	certKIND		requires discussion with AAAI
	Cost limits				to compare with platform offering	
		costCPUsec				(repeated)
		costRAMMBsec				(repeated)
		costSTORMBday				(repeated)
	User Input parameters required					
		[application specific]				**cfPers_Indic**
		name				**cfResProd_Indic**
		type				**cfIndic**

		minvalue	maxvalue	listvalue		cfIndic_Class
	Dataset Metadata required					
		[application specific]				
	identity	UUID				cfResProd.cfResProdId
	product is a schema					cfClass
	schemaname					cfResProdName.cfName
	description					cfResProdDescr
	keywords					cfResProdKeyw
	location	coordinate pair			spatial coordinates	need cfResProd_Paddr, cfPaddr-GeoBBox or via orgunit of owner
	owner	UUID, orgname				cfOrgUnit_ResProd

	manager	UUID, persname				cfPers_ResProd
	supplier	UUID, org name				cfOrgUnit_ResProd
	licence documentation				a document	cfResPubl
	use condition				e.g. CC-BY	cfClass
	embargo				handled by cfClass of role linking owner to product and	cfResProd_cfOrg
	[provenance]				derived from history of changes	
	URL					cfResProdURL
	Access information required				to compare with platform offering	
		regUSER	userID	userPW		requires discussion with AAAI
		certUSER	certID	certKIND		requires discussion with AAAI
	Datasets required					
		[application specific]				
	identity	UUID				cfResProd.cfResProdId
	product is a dataset					cfClass

	schemaname					cfResProdName.cfName
	description					cfResProdDescr
	keywords					cfResProdKeyw
	location	coordinate pair			spatial coordinates	need cfResProd_Paddr, cfPaddr-GeoBBox or via orgunit of owner
	owner	UUID, orgname				cfOrgUnit_ResProd
	manager	UUID, persname				cfPers_ResProd
	supplier	UUID, org name				cfOrgUnit_ResProd
	licence documentation				a document	cfResPubl
	use condition				e.g. CC-BY	cfClass
	embargo				handled by cfClass of role linking owner to product and	cfResProd_cfOrg

	[provenance]				derived from history of changes	
	URL					**cfResProdURL**
	minLAT	meanLAT	maxLAT			**cfResProd_Indic**
						cfIndic
						cfIndic_Class
	size	Mb				**cfResProd_Indic**
						cfIndic
						cfIndic_Class
	medium					**cfClass**
	format					**cfClass**
	selection					**cfClass**
	projection					**cfClass**
WORKFLOW SERVICE					may be offered bundled with platform or independent	
					assumed to include software services with datasets	
	identity	UUID			maybe federated IDs e.g. local ID	**cfResProd.cfResProdId**

	workflowservicename	name			may be federated / may be multilingual	**cfClass**
	workflow service description					**cfResProdDescr**
	workflow service keywords				from restricted vocabulary	**cfResProdKeyw**
	location	coordinate pair			spatial coordinates	**need cfResProd_Paddr, cfPaddr-GeoBBox or via orgunit of owner**
	owner	UUID, orgname			UUID/name/role owner/Dtstart/Dtend	**cfOrgUnit_ResProd**
	manager	UUID, persname			UUID/name/role manager/DTstart/DTend	**cfPers_ResProd**
	supplier	UUID, org name			UUID/name/role supplier/Dtstart/Dtend	**cfOrgUnit_ResProd**
	licence documentation				a document	**cfResPubl**
	use condition				e.g. CC-BY	**cfClass**
	embargo				handled by cfClass of role linking owner to product and	**cfResProd_cfOrg**
	[provenance]				derived from history of changes	
	Hardware Required				to compare with platform offering	**cfEquip_Indic**

		CPU	minCPU	maxCPU	(do we need to distinguish GPU?)	**cfIndic**
		cores	mincores	maxcores		**cfIndic_Class**
		RAM	minRAMMB	maxRAMMB		(repeated)
		Storage	minSTORMB	maxSTORM		(repeated)
		Storage	minSTORTRA	maxSTORT	mb/sec	(repeated)
	Container required	type				(repeated)
	VM required	flavour				(repeated)
	OS required	name	version			(repeated)

	software languages required	name	version			(repeated)
	data file structures	name	version		linked to supplier. Examples Hadoop	(repeated)
	Access performance required				to compare with platform offering	(repeated)
		network	minTRANS	meanTRAns	transfer speeds in mb/sec	(repeated)
		network	minLAT	meanLAT	latency in ms	(repeated)
	Access information required				to compare with platform offering	
		regUSER	userID	userPW		requires discussion with AAAI
		certUSER	certID	certKIND		requires discussion with AAAI
	Cost limits				to compare with platform offering	
		costCPUsec				(repeated)
		costRAMMBsec				(repeated)
		costSTORMBday				(repeated)
	User Input parameters required					

		link to required for each software service				
	Dataset Metadata required					
		link to required for each metadata asset				
	Datasets required					
		link to required for each dataset asset				
		In CATALOG in user profile)			only if security sufficient; maybe confirm each time	
END-USER Deployment parameters						
	ID	UUID			federated IDs based on role and temporal interval	
	Name				do we need name or is ID sufficient?	
	regUSER	userID	userPW			
	certUSER	certID	certKIND			

	[for each workflow/software service]					
	Cost	overrideable overall cost			each set linked to a particular workflow or software service	cfPers_Indic
						cfResProd_Indic
						cfIndic
						cfIndic_Class
	Time	overrideable overall elapsed time			each set linked to a particular workflow or software service	cfPers_Indic
						cfResProd_Indic
						cfIndic
						cfIndic_Class
	[User input parameters]	overrideable			each set linked to a particular workflow or software service	cfPers_Indic
						cfResProd_Indic
						cfIndic
						cfIndic_Class