Comprehensible Control for Researchers and Developers facing Data Challenges

Malcolm Atkinson
School of Informatics
University of Edinburgh
Edinburgh, UK
Malcolm.Atkinson@ed.ac.uk

Rosa Filgueira
EPCC
University of Edinburgh
Edinburgh, UK
r.filgueira@epcc.ed.ac.uk

Iraklis Klampanos
Institute of Informatics and Telecommunications
National Centre for Scientific Research Demokritos
Athens, Greece
iaklampanos@iit.demokritos.gr

Antonis Koukourikos
Institute of Informatics and Telecommunications
National Centre for Scientific Research Demokritos
Athens, Greece
kukurik@iit.demokritos.gr

Amrey Krause
EPCC
University of Edinburgh
Edinburgh, UK
a.krause@epcc.ed.ac.uk

Federica Magnoni
Istituto Nazionale di Geofisica
e Vulcanologia
Rome, Italy
federica.magnoni@ingv.it

Christian Pagé
CECI
Université de Toulouse, CNRS, Cerfacs
Toulouse, France
christian.page@cerfacs.fr

Andreas Rietbrock
Geophysical Institute
Karlsruhe Institute of Technology
Karleruhe, Germany
andreas.rietbrock@kit.edu

Alessandro Spinuso
dept. Observation and Data Technologies
Koninklijk Nederlands Meteorologisch Instituut
De Bilt, The Netherlands
alessandro.spinuso@knmi.nl

***Abstract*—The DARE platform enables researchers and their developers to exploit more capabilities to handle complexity and scale in data, computation and collaboration. Today's challenges pose increasing and urgent demands for this combination of capabilities. To meet technical, economic and governance constraints, application communities must use use shared digital infrastructure principally via virtualisation and mapping. This requires precise abstractions that retain their meaning while their implementations and infrastructures change. Giving specialists direct control over these capabilities with detail relevant to each discipline is necessary for adoption. Research agility, improved power and retained return on intellectual investment incentivise that adoption. We report on an architecture for establishing *and sustaining* the necessary optimised mappings and early evaluations of its feasibility with two application communities.**

***Index Terms*—conceptualisation, data-driven science, scientific workflows, provenance, HPC on cloud, Multi-everything CSCW**

I. Introduction

Today's research challenges in long-term multi-disciplinary campaigns (e.g., observing the early universe, ameliorating environmental hazards, scarce-resource conservation) require collaborating communities, sharing data, methods and infrastructure – pooling inputs – thinking together. This presents conceptual challenges to their participants in coping with the complexity and in communicating with each other and with the services they use with sufficient clarity and precision. Organisations and individuals often engage in several such federations, requiring that they retain autonomy.

The urgency of research challenges and exploitation of new possibilities "require rapid innovation which is inhibited when *full alignment* is attempted", Nobel Laureate[1] Venki Ramakrishnan [1] (see page 263). DARE is pioneering an approach where rapid innovation by agile research teams and stable production work can be sustained and share as much as possible. More than two decades of supporting eScience and its predecessors have convinced us that this is necessary. In consequence, we can state clearly the form this should take. We are at the early stages of implementing this strategy.

The digital environment: (computational platforms, software, services, algorithms, libraries of components and data sources), change rapidly compared with a research federation's goals. Therefore, collaborating communities need to explore new opportunities without having to reformulate their methods, change their working practices or be distracted by technical detail. We propose three innovations to enable this:

1) *Extension of collaboration* support from its current main focus on *data* to include *concepts*, *methods* and *collections* with equal status and properly interconnected.
2) *Independent direct control of focused contexts* that are cross-coupled and span from conceptualised domain spaces to sophisticated implementations.
3) *Governance models* determining when and where to adopt innovations or retain production stability.

These empower sustainable and scalable collaborations with new capabilities for production and experimentation.

The DARE project is delivering these innovations by pioneering a high-level platform that will be widely applicable

[1]For discovering (with others) how ribosomes work.

and therefore sustainable. This is co-developed by application experts and their research-developer specialists to tailor their R&D environments exploiting DARE's innovations. During DARE they co-develop with the platform team that builds on prior experience [2] and meets new requirements from environmental scientists [3].

Below, we clarify requirements, present an architecture and describe early experience with two communities who face such challenges: computational seismologists and climate-impact modellers.

II. Clarifying requirements

Challenges, such as mitigating the impact of climate change, or giving reliable information to responders to an environmental-hazard event, require advances on three fronts:

1) using effectively growing volumes of diverse data,
2) exploiting computational power to simulate phenomena and correlate results with observations, and
3) managing increasing complexity of data, models, collaborations and research campaigns.

Research has to move into reliable and effective stable production contexts to meet repeated demands. But communities require agility to meet new emergencies and to compete.

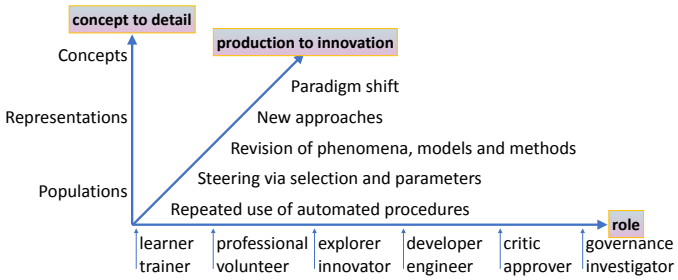

Fig. 1. Individuals and organisations move in these three dimensions.

A. Broadening collaboration support

CSCW research has shown that providing shared information is necessary but not sufficient to aid collaboration; active use and engagement is necessary, which depends on presenting to each collaborator the relevant information, in the right form at the right time, so that it helps *their* work. Research communities are inherently complex, as shown in Fig. 1.

1) Concepts: Research is developed in terms of *concepts* that are in focus, e.g., an observable of phenomena, or an encoding of a statistical comparison. Each expert has their own set of concepts. Collaboration depends on agreeing on sufficient common concepts and on how to interrelate those not fully aligned [4]. DARE supports a nested hierarchy of *contexts* for this purpose – see Fig. 2. Experts discuss, refine and use concepts, such as: `Earthquake`, `Flood` and `ArchiveService`. Agreement on common concepts, their names and properties enables precise communication between humans and with systems across space, time and disciplines. Relationships need to be explicit, e.g., `Earthquake` and `Flood` are kinds of `EnvironmentalHazardEvent` and an `Earthquake` has a `SeismicSource` and `WavePropagationModel`.

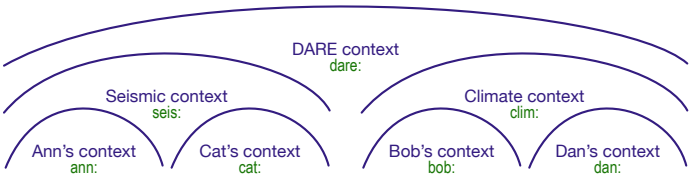

Fig. 2. Work contexts are progressively specialised by tailoring outer contexts.

2) Methods: Communities develop *methods* that are effective. Much education, training, formalisation and R&D is focused on methods; a substantial investment. As far as possible, methods are expressed in terms of concepts independent of technology. Methods cover all aspects of work, e.g., archiving and publishing results, analysing observations, simulating phenomena, validating a new implementation of a method, installing a change to a shared context, etc.

3) Data: The need to share *data* is well established [5]. Data represent everything. However, without their connection with concepts they are hard to understand and many users think in terms of concepts and assume data represent their instances. Without methods data cannot be produced, managed or used. The interrelationship between methods and data is crucial. New insights about how to combine data lead to new methods and expose new concepts.

4) Collections: We differentiate *collections* from data as working practices treat them differently. Users build and manipulate collections, e.g., ensembles of simulation results, relevant observations or examples of new phenomena. Facilitating these practices is key to adoption and it also provides opportunities for improved engineering. Collections may be of anything: concepts, methods, data or collections.

For individuals, groups and communities, we need to support all four aspects of a *context*: *concepts*, *methods*, *data* and *collections* and their interrelationships for their immediate use and for others revisiting or reviewing their work. The shaping of all aspects are equally important. Change may be led in any context by exploring any of the four aspects.

B. Delivering direct control

As users work they build more information in *their* context, e.g., new sets of validated time series, new versions of an analysis process, new methods and new concepts. With validation and agreement innovations may be used more widely. Users apply their expertise, e.g., selecting methods, supplying parameters, monitoring visualisations and taking remedial action when necessary. Their judgement informs their work and hence outcomes. In complex collaborations their work interacts with that of other experts, and extends into new territory. A comprehensive provenance system facilitates this [6]. It enables them to see what is happening, wherever it is happening. If they have moved into new territory this is an intellectual ramp developing their understanding. If there are problems, they dig into detail for diagnostic evidence without

having to understand underlying systems. If there are many stages to be accomplished by multiple individuals, it enables coordination. If resources are to be conserved or regulations honoured, it provides a definitive source of evidence [7]. Provenance acts as a *lingua franca* as it uses standards to hide technology variations. It still needs translating for each reader.

Users work directly in their familiar context. They must feel they are in control and understand what is happening so they can exercise their judgement, e.g., stop a method proceeding further to avoid waste if they judge it will not deliver worthwhile results. However, the platform needs to determine when and where actions take place to avoid data transport costs, use appropriate resources, support users with limited resources and achieve pervasive provenance.

C. Enabling governance

Many stakeholders need to steer work to meet changing priorities, new challenges and revised regulations. They need to balance the need for stability in production services with the need for innovation. They investigate issues and propose remedies. They develop a sustainability model and mix collaboration with competition. As communities mature this governance model develops. The tailoring and provisioning of work contexts enables them to implement decisions. The comprehensive provenance collection enables investigations and planning. As scale and complexity grow, the three-phase (**CRP**) methodology pioneered by Trani [8] should govern the long-term evolution of all aspects of shared contexts:

1) the specialists in a context *agree* the **C**oncepts they need,
2) the **R**epresentations of these are then developed, and
3) the **P**opulations of relevant entities are then assembled.

III. DARE ARCHITECTURE

The platform is an intermediary between the tools, web sites and interaction methods presented to users and underlying ICT facilities. To do this it provides two mechanisms:

1) an API via which tools request, control and monitor actions and inspect provenance trails; and
2) a DARE development kit (DDK) to facilitate the authoring and use of such actions – currently a python library.

This involves three subsystems, as shown in Fig. 3[2]:

1) the *DARE knowledge base* (DKB) that specifies the interpretation of all the terms used by the API and DDK;
2) the *Workflow-as-a-Service* (WaaS) that performs actions encoded in a set of notations by arranging to optimise, enact, monitor and control the encoded methods, setting up or choosing enactment targets; and
3) the *protected pervasive persistent provenance* (P4) system that records user interactions and WaaS enactments.

[2]Details in D2.1 https://zenodo.org/record/2613550#.XJy7fNHgonM

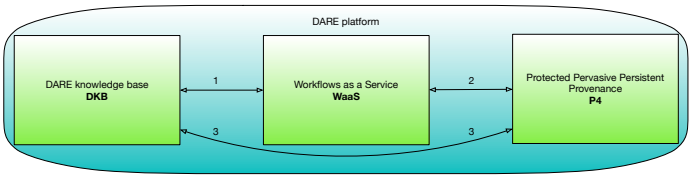

Fig. 3. Each instance of the DARE platform includes three subsystems that combine to deliver the platform's capabilities – see Sec. III-A to Sec. III-D.

A. DARE Knowledge Base (DKB)

The operational use of the DKB as an intermediary and integrator is shown in Fig. 4 with six communication steps.

1) **Request** A user requests some action.
2) **Submit** A tool interprets that action and transmits it to the API.
3) **Translate** Information about each term is provided by the DKB. The platform validates the action, fills in details, chooses targets. It maps actions to targets.
4) **Enactment** Where necessary the target is prepared, e.g., by deploying and initialising services. Actions are delegated to targets and connected with each other and with persistent services. Some targets are external resources. Provenance records are streamed to P4 – see Sec. III-C.
5) **Steering** Users observe and steer their work to judge whether useful results are emerging or to take remedial action. Actions record provenance data as they progress. A steered visualisation informs (remedial) actions.
6) **Finishing** When enactment finishes the new state of the system is reflected in the DKB.

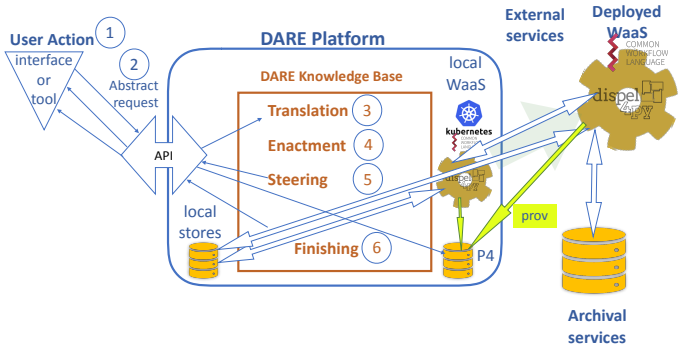

Fig. 4. The platform mediating between users and the evolving digital environment to preserve semantics and deliver new capabilities.

The DKB has these features:

1) universal **time-stamping** to identify a particular version of an entry, so that a time-bounded query reconstructs a previous state. Updates produce a new entry with the same identifier but a later timestamp, chained to inhibit updates to old versions; providing arbitration.

2) **identification** external entities where the full information resides[3] use the external PID. Local entries have a given identifier, e.g., the identifier in a method's script. Lookup without a prefix tries the innermost and then successive outer contexts, up to `dare:` delivering inheritance. Explicit prefixes enable controlled imports; timestamped look-ups get versions.
3) production **properties** are the *common* attributes agreed by the users of a context plus those for platform operation, such as `successor`. When multiple entries describe one logical entity, e.g., an `implementation` of a method, then cross-references are required. Properties may restrict use and mutation.
4) free-form **annotation** attaches a sequence of annotations to an entry. They are used by developers and software[4] to explore and innovate in production contexts. Updates to the `annotations` are not recorded as "official" with a new `timestamp` and `successor`. This differentiates experiments from production while maximising sharing. When agreed they may be promoted to production.

The four aspects of each context have additional properties.

1) **Concept** The agreed names (`labels` and external/internal PIDs) and essential properties of concepts are made explicit. Concepts may inherit from super-concepts, and enumerate their sub-concepts. They use ontologies to make agreements precise and cross-reference representations, e.g., python and OWL classes.
2) **Method** Each method has a set of name-description pairs of inputs and a similar set defining outputs. That name denotes a port or an array of ports. An input description specifies a required concept and optionally required representations, logical or temporal relationships, default values and optionality. Methods reference implementations. Platform instances choose which technologies to support, but should include:
   a) data streaming, e.g., provided by `dispel4py` [9] for production, and
   b) immediate enactment in python for developers and exploration – see Sec. III-B
3) **Data** These entries are the focus of users' work. They permit users to manage their data without it being localised or to refer to data managed by others. Local and external naming is supported, e.g., as for `Dataset` entries in DCAT [10]. Rules for use may be encoded. Methods for **import**, **ingest** and **export** should be engineered for scale, protection and preservation.
4) **Collections** User actions for constructing, using and managing collections are presented locally but implemented optimally. A collection may be related with local or external collections by composition and queries. Standard named collections for each concept are provided, e.g., `xs` is all instances of `X`.

[3]DKB holds minimal information needed or created by users and platform.

[4]As structured and precise as needed, e.g., timestamped using the Web Annotation Vocabulary www.w3.org/TR/annotation-vocab

B. Workflows-as-a-Service (WaaS)

DARE provides researchers and developers with the means to express their computational and data transformation needs in a consistent, self-documenting and high-level manner, close to their current way of working. Data-driven workflows have been identified as the best technology meeting this requirement [2] for the following reasons: (a) they are defined using familiar programming languages, (b) each processing element (node) in the workflow has a clear role and semantics – domain-specific concepts and methods that merit communication, (c) they are independent from implementation decisions and computational contexts, and (d) they offer optimisation that requires no input from their users.

The platform provides cataloguing, optimisation, deployment, monitoring, provenance recording and tracking. Workflow optimisation includes target selection, preparation and deployment. The DKB makes combining these tasks possible. The semantics of methods, components, computational contexts and data, enables optimisers to deploy parts of workflows to different targets, while orchestrating and monitoring their interaction to ensure progress and correct results.

The platform accepts workflows written in python using the dispel4py library [9]. DARE is incorporating the Common Workflow Language (CWL) [11], [12] into WaaS. Users interact with a workflow and the processing elements registry and submit workflows via a Web API.

C. Protected Pervasive Persistent Provenance (P4)

Provenance provides *definitive* evidence of what has happened. P4 extends the VERCE system [2] with improved precision, controls on provenance capture, domain metadata injection, agile use, tools and visualisations with steerable clustering [13], [6]. These extend the W3C-PROV standard[5] and the ProvONE developments[6]. DARE is extending the collection and control mechanisms in P4 to make provenance pervasive and protected, i.e., captured for every significant action, granting researchers control on disclosing and sharing their traces. Provenance streams need to be available promptly for method-steering; this required new mechanisms. Provenance must be preserved and protected until there is no prospect of investigators accessing for any reason. The resulting accumulated provenance becomes a valuable data resource, queried and analysed to improve planning, methods, optimisations and community procedures. This in-depth use is supported by tooling and a Web API that facilitates the formulation of complex recursive traversals across provenance relationships, the inclusion and use of application metadata terms and summarisation.

We distinguish different types of provenance information to capture completely the progress and phases of scientific practices. Recordings captured by execution of a method as retrospective provenance or lineage. These records trace the behaviour of a method's enactment including the mapping

[5]http://www.w3.org/TR/prov-dm/

[6]https://purl.dataone.org/provone-v1-dev

between its logical representation and concrete implementations and its mapping onto computational resources. To resolve incipient ambiguities, we have developed a model (S-PROV) [13] that captures aspects associated with the distribution of the computation, volatile and materialised data-flow and dependencies on the internal state of each process. We consider a set of Interaction Patterns controlled by users' choices as they develop methods and configure work-spaces. These are associated with conceptual sessions that require the deployment and incremental update of software libraries, containerisation and access to development tools, such as Jupyter notebooks. Users benefit from such environments by sharing the combination of documented code and results. However, reproducibility is not guaranteed if the shared information is not supported by a snapshot of the computational context used and an affordable and semi-automated way of reconstructing the same conditions. We analysed these scenarios, identifying implications of managing a session from the user's and system's perspective.

We record the interactions of research-developers and scientists with the platform as (*Interaction patterns*); integrating these with (*lineage*) collected as their methods run and with the DKB; enabling a thorough analysis of the traces. This contributes to the quality of results that can be shared, for instance, as published research objects [14] packaged with extensive and semantically consistent documentation. In section V-C we will provide more technical background.

D. Platform integration, deployment and operation

The three subsystems, see Fig. 3, must maintain consistency as they handle requests via the API, or perform methods via the DDK. These include consistent identification of entities between the DKB and P4, consistent interpretation of actions between the DKB and WaaS, DKB updates matching P4 records and provenance traces in P4 cross-referenced from `Run` entries in DKB. The initial state of an instance of the platform honours these invariants. All activities thereafter, via the user and developer facilities or during the installation of new platform releases must maintain these invariants.

IV. Scientific applications

DARE's two demanding use cases are typical of many applications as a result of our growing wealth of data, increasing computational capabilities, complex studied systems and networked federations [15]. Extensive communities of diverse users address pressing challenges. To do so they demand improved, understandable and controlled shared resources expanding their research capabilities in the three directions: data volumes, computational power and coping with complexity.

A. Computational seismology

Seismology faces the challenge of managing increasing amounts of recorded and simulated data. Harnessing highly accurate and efficient simulation tools combined with exponentially increasing data volumes is becoming a huge challenge. Calculation requirements are moving towards exascale computing (e.g., the ChEESE project[7]), GPU exploitation (e.g. SPECFEM3D) and iCloud resources (EOSC). However, how to support rapid combination, analysis and evaluation of high-fidelity simulations and recorded data is an open question that is crucial for rapid hazard assessment and emergency responses after large earthquakes. Seismological data analysis requires tools that can be easily customised and applied to keep up with the quickly evolving knowledge in the scientific communities. Robust provenance-driven tools are needed to smartly organise storage of data and allow for their exploration, combination and reuse, while promoting error detection and reproducibility of scientific experiments. These needs are crucial to provide reliable and robust (including error margins) estimates of the earthquake impact after a large event to coordinate emergency action and to inform the public. To achieve this goal we have broken down the analysis task into three domain-specific applications with increasing complexity as well as rising computational and storage demands.

In this framework, the rapid assessment of seismic ground motion (RA) is a domain application that represents one of the main issues in seismology embodying all the aforementioned needs. After a large earthquake, we must rapidly simulate the propagation of seismic waves in surrounding areas to characterise the earthquake's impact by estimating ground-motion parameters that are indicative of structural damage and risk. Comparison and integration of synthetic with recorded ground-motion data allow us to improve the characterisation of the ground behaviour in the whole region affected by the earthquake. The theoretical foundations and applicable procedures are well established, with the general high-level steps represented by Fig. 5. For the RA application we focus towards on-demand rapid calculations and easy management of very large simulation results.

The earthquake source plays a key role in strong ground motion. Thus, another crucial application in seismology is the rapid characterisation of Seismic Sources (SS) to evaluate the impact on the radiated seismic energy and ground-motion behaviour. This requires the management of multiple models with very demanding computational and data-intensive calculations, cross-referencing and reuse of pre-calculated large data volumes.

Given the two domain applications (RA and SS), we are finally able to tackle the comprehensive application outlined at the beginning, to statistically characterise the ground-motion parameters and their uncertainties by analysing ensembles of models (ES) sampling variations in seismic source parameters. This will allow us to attribute uncertainties to the strong ground motion estimates, thereby increasing their reliability and robustness leading to improved usability. This stresses again the handling of large multi-model calculations, the recovery and reuse of very large data volumes from multiple methods and the comparison with large recorded data sets.

Drawing on a strong scientific background, these applications are a valuable test-bed for the platform and its ability to

[7]https://cheese-coe.eu

manage the interconnections between data, computations and analysis keeping them transparent to the users – Sec. VI-A.

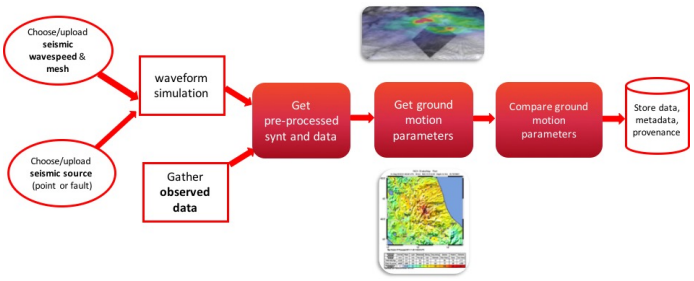

Fig. 5. The Rapid Assessment method analysing the impact of an earthquake.

B. Climate Change Impact Data Analytics

Today scientific researchers in the climate domain have to deal with much larger data volumes. Coupled Model Intercomparison Projects (CMIP) exercises are conducted periodically by the international community to advance climate science and to provide the scientific basis for IPCC reports. Each of these CMIPs generate larger data volumes, because of improved spatial and temporal resolution, the design of larger experiments (more ensemble members, more investigations, more climate modelling centres, etc.). The increase in data from the previous exercise CMIP5 to the current CMIP6 will be from 2 to 30 PB (anticipated) globally. All data are stored by the Earth System Grid Federation (ESGF), an international collaboration that develops, deploys and maintains the infrastructure for the management, dissemination, and analysis of climate data.

Downloading locally all data needed for analysis and post-processing is no longer possible because of the data volumes needed. This potentially restricts the use of climate data only to those who can afford the necessary local infrastructure and network bandwidth. This creates a pressing need for an easier use of climate data, especially for researchers working on climate change impact assessment.

To fulfill this objective, the climate4impact[8] platform and services (C4I) has, since 2009, been developed by European Commission funded projects: IS-ENES, IS-ENES2 and IS-ENES3, targeting climate change impact modellers, impact and adaptation consultants, and other experts using climate change data (see Fig. 6 and [16]). It provides users with harmonised access to climate model data through tailored services. It features static and dynamic documentation, use cases and best-practice examples, an advanced search interface, an authentication and authorisation system integrated with the ESGF and a visualisation interface. A very important feature is that it has on-demand data-processing capabilities. However, for now, all data processing occurs on the C4I-platform server. That is not scalable! Another critical issue is that provenance information – a key factor in assuring evidence quality – is very limited [17].

[8]https://climate4impact.eu/

Fig. 6. The C4I https://climate4impact.eu/ front-end aimed at the climate change impact modelling researchers.

The DARE platform will ameliorate the scalability and provenance problems. It will enable end-users to perform calculations on-demand by providing a transparent back-end to C4I. They can then perform scientifically required analyses even when they require large volumes of input data. The computations will be closer to the data, in a bandwidth sense, as well as being deployed on available computational resources. This will facilitate multi-scenario analyses that must be performed to analyse climate scenarios with estimates of uncertainty and assessment of impact.

V. Prototype DARE platform

The DARE architecture (see Sec. III) is prototyping a socio-technical strategy for achieving and sustaining long-term and challenging objectives. The current implementation [18], outlined below, is a significant step towards those objectives that supports our two use cases.

A. DARE catalogue

The DARE catalogue constitutes the functional implementation of the DKB, focusing for the moment on the interfacing with the enactment system. It is logically divided in three catalogues, handling *Processing Elements* (PEs)[9], *Components*[10], and *Data* respectively, corresponding to three concepts in the `dare:` context.

The *PE* catalogue exposes via the DARE API descriptions of the available PEs that can be used for building more complex workflows. The implementation of each PE is also stored and retrieved via the catalogue's API in order to be used in method graphs that are then submitted for enactment to selected components of the platform or external services. This

[9]Instances of these are actions connected by data streams in methods.
[10]Components are Web or local services.

catalogue holds descriptions of which components PEs and workflow graphs containing them can be mapped to and which mappings to use for each target type. Optimisation parameters and constraints will be added here.

The *components* catalogue holds information on the nature and state of the platform's (or network of platforms) infrastructural assets. It is populated by harvesting metadata from the Kubernetes installation orchestrating each instance of the platform. It gathers information on the location, state, and status of each asset and registers or updates the description of the component in the catalogue. Basic information like the hosting IP and accessible ports are provided by Kubernetes. This also holds for the component's generic status (`stopped`, `running`, `busy`). Information regarding the component's purpose and possible uses would be added by providers when registering their components with the Kubernetes cluster.

The *data* catalogue acts as a metadata registry for the datasets used and produced by the users. Conceptually, we can distinguish between descriptive and functional characteristics of the datasets.The former include information on the owner and creator of the dataset (at the personal and organisational levels), define relevant topics and themes and provide links to other assets, e.g., other datasets or publications. Such information is expected to be used for discovery and linking. The latter are more closely associated with the operation of the platform as they specify access rights, the type and location of the file(s) included in the dataset, as well as temporal and spatial information for the equipment and/or process that produced the dataset.

The data catalogue's uses concepts defined by the DCAT W3C Recommendation[11]. Hence, datasets described within the catalogue are conceptualised as instances of the `dare:Dataset` class, a subclass of `dcat:Dataset`. Additional properties for associating a dataset with the overall operation of the DARE platform (i.e the dataset's creator/contributor and the processes within the platform that led to the creation of the dataset) are included in the catalogue's schema. Taking into account the work on EPOS-DCAT [19], further concepts for scientific equipment (`dare:Equipment`) and facilities (`dare:Facility`) are defined, as generalisations of `epos:Equipment` and `epos:Facility` concepts.

B. DARE prototype workflow provision

We used RA (Sec. IV-A) as a test case for workflow provisioning. This application has five phases (see Fig. 5): (1) select an earthquake that generated an observed wavefield, (2) simulate seismic waveforms for the same earthquake using SPECFEM3D [20], MPI-based software; (3) pre-process both synthetic and real data; (4) calculate the ground motion parameters for both; (5) compare them by creating shake maps.

We built a `dispel4py` workflow representing each part as a streaming pipeline, except for the generation of the synthetic data, since SPECFEM3D is a parallel simulation code. Then, we used CWL to connect the `dispel4py` workflows, to describe semantically their input and output parameters, and to orchestrate their executions using an appropriate mapping to the computing resources available. We created a docker container with SPECFEM3D and MPI to deploy on demand a consistent environment on a cluster for generating synthetic waveforms using local or distributed resources. CWL managed the execution of SPECFEM3D, enabling us to fully implement the RA method. To validate this work we tested every step of RA using a small dataset and the sequential `dispel4py` mapping on a laptop.

Once validated, we registered the `dispel4py` workflows. The method is activated via the DARE API where a web service takes care automatically of the entire execution acting as an intermediary between users' applications and the underlying computing resources. It provisions a computing environment with all the necessary elements (MPI cluster, dispel4py, CWL, SPECFEM3D) on demand using Kubernetes[12]. Once that is available, the service automatically runs and monitors the RA and collects its provenance and results, making them available immediately.

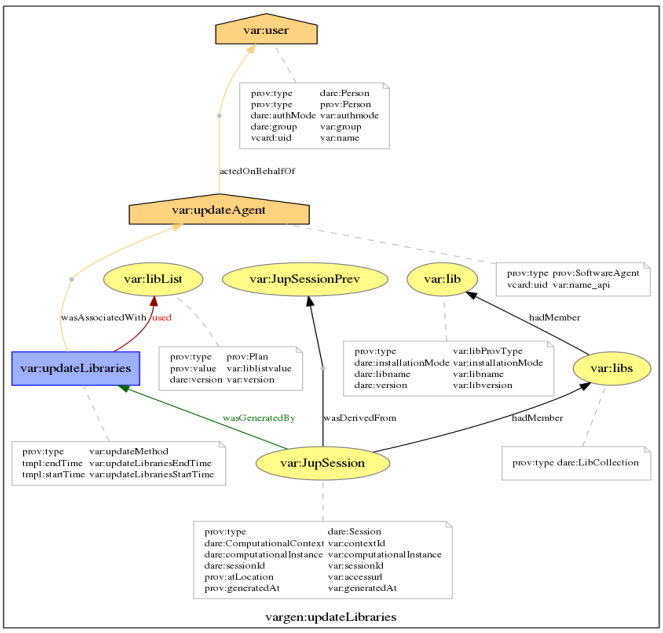

Fig. 7. Interaction Patterns: Provenance Template describing the activity of updating the libraries used during a session. The session consists of a Jupyter controlled environment and links back to its previous state

C. Prototype provenance acquisition and provision

We outline the provenance functionality, described in [6], that enables the acquisition and exploitation of provenance data. We capture users' choices when they customise their computational space, track reproducible progress and share methods and results within or beyond DARE.

[11]https://www.w3.org/TR/vocab-dcat/

[12]kubernetes.io/docs/concepts/overview/what-is-kubernetes

1) Workflows' execution lineage: the execution of a method is described by its initial inputs, its components, their inter-dependencies and their implementation, and the computational resources used. We acquire provenance from different types of systems (CWL and dispel4py). They are mapped to S-PROV [13], in order to be interactively explored and visualised using DARE's S-ProvFlow tools and lineage API[13]. We provide three kinds of exploration functionalities.

Monitoring. The execution of a workflow can be monitored at different levels of detail. From views at conceptual level, considering the classification of the components introduced by the users, to more detailed information about parallel enactment of the workflow. Triggers associated with the occurrence of specific metadata values or value-ranges initiate actions, e.g., `notify_a_user` or `deliver_results` [13].

Lineage queries let users explore a selected data-product bidirectionally, i.e., how was it produced? and how was it used? Users and software can navigate the data-derivation graph in both directions. Queries specify the depth traversed at each step. Combining this with queries on metadata makes a large collection of provenance data very useful. DARE provides drill down to nodes and related datasets.

Discovery. Users can search for workflow executions and datasets adopting concepts and metadata terms from agreed (application-domain) vocabularies and local experimental terms. Descriptions of metadata in the DKB guide the formation of queries and visualisations, e.g., through hints about terms and ranges. These relate to terms and concepts introduced during a user's runs that have been recorded in the S-PROV entities describing parameters, data and components. The same metadata helps users retrieve session snapshots and their dependencies for reproducibility.

2) Interaction patterns: we adopt the Provenance Templates framework [21] to describe and capture provenance information characterising interaction patterns between users and the platform. Templates model provenance relationships describing the creation and update of a session, for instance using containers interfaced by a Jupyter notebook environment. User's choices of adopting new or updated software libraries can be captured preserving the derivation relationships with the previous state of the environment, as shown in Fig. 7. The templates are stored and made available through a dedicated service, the Provenance Template Registry[14], which has been extended to allow its containerised deployment. These templates are associated with an instance of the platform and used to generate provenance documents. The documents can be linked with the lineage through the `session-id`, so the results obtained by accessing the optimised lineage queries offered by the S-ProvFlow system are combined and integrated with session information.

VI. Experimentation

Our approach intends to deliver user benefits, collaboration advantages and sustainability. Such long-term aspects of supporting data-driven science and distributed behaviour are difficult to measure and hidden by transition effects this early. We show that key risks are avoided: e.g., that the incremental adoption model based on conceptualised controllable contexts enables controlled adoption without too much disruption. Our users speak for themselves.

A. Computational seismology experience

The DARE platform supports our work – see Sec. IV-A. Exploiting the modularity and flexibility of the approach, applications are conceived and developed using more general and abstract terms that facilitate user customisation and updates. Continuously evolving scientific approaches are therefore more easily explored and more importantly applied.

The platform takes care of the main issues that impede complex research applications. Computational intensive and complex numerical software, required for our rapid on-demand seismic simulations, are handled automatically. APIs manage the connection between user inputs and DARE components, both for computations and for data-intensive `dispel4py` processing used by our seismological workflows. Transfer and storage of large data volumes are handled transparently, automatically capturing related metadata and provenance information, an essential requirement for managing and validating our applications that use multiple simulations and pre-calculated databases.

For instance, in the RA use case, where different interlinked workflows perform common conceptual tasks over data with similar properties, the possibility offered by DARE to record, query and visualise provenance information can be used to highlight data-reuse dynamics within large collaborative experiments. In Fig. 8, we show a radial diagram that displays runs executed by two users. The right half of the diagram shows interlinked workflows organised into separated radiants, according to their conceptual tasks. These were described by specifying concepts and metadata to contextualise the methods involved. In contrast, the left side, with a poor conceptual characterisation, typical of the early phase of exploration, results in chaotic and harder to visually analyse provenance graphs [13]. The platform overcomes the pre-configured inflexibility of previous facilities [2]. It avoids restrictions; freeing up everything, e.g., the type of input and output, the processing needed and the provenance captured. These are now controllable, enabling experts and exacting users to customise their work in a powerful research environment.

The platform has the potential to expand beyond single applications, facilitating exploration of multiple, diverse seismological issues, and delivering a valid research platform for other communities.

B. Climate-Impact modelling experience

A very important aspect of climate-related research (see Sec. IV-B) is that scenario uncertainty must be evaluated. That means that multi-scenario analyses must be performed by the researchers. Since the on-demand data-processing capabilities of C4I are not currently scalable there is limited capacity to use

[13]https://github.com/KNMI/s-provenance

[14]https://github.com/EnvriPlus-PROV/ProvTemplateCatalog

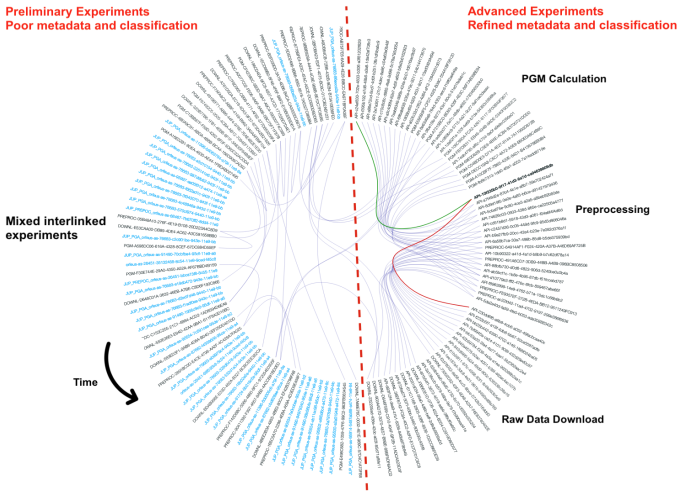

Fig. 8. Radial diagram highlighting data reuse between different workflows of the RA use case. Vertices are workflows execution *ids* colour-coded by user. The edges represent data flows. Red and green edges are visualised upon hovering on the ids to represent data input and output respectively. The runs were selected by interactively querying the provenance archive to find those using a common set of seismic stations. The right half shows the connections between interlinked workflows that were described using DARE functionalities for user and data-driven contextualisation. They were performed by a user who improved the description of the methods over time. Consequently, these are visually better organised yielding a reduction in the complexity of performing results management.

C4I to perform such analyses. An obvious idea is to move the computation submitted via C4I closer to the data (in a network bandwidth sense) where computing resources are available for a specific user. Since C4I is supporting a large heterogeneous base of users, this means that computing resources they have access to are on different architectures [22].

The DARE platform unlocks the potential of C4I by deploying the computation back-end onto heterogeneous systems. It adapts to the target architecture, and is easily deployed thanks to its container-based technology. It makes the development of new services, as well as the support for several different architectures, a much faster and easier process thereby opening up new possibilities. The automated production of provenance information, which is still in its early days in the climate-science domain, is an important feature.

VII. Related work

The need for users to express their Intentions so that are Interpreted correctly over long periods while their target (cloud) platforms evolve with Incentivised adoption (I^3) was identified by Schubert and Jeffery [23] as essential to meet the challenges of modern science, engineering and business. A crucial incentive was identified by Myres *et al.* [24]: *reward users by delivering automation to remove chores and improve productivity*; in their case by supporting collections via catalogues. Our user-tuned contexts deliver this. That incentivises adoption by communities. It will be backed by governance and funders as it prepares material for FAIR archives [5]. Schuler and Kesselman identify the need for researchers to control their research context and enable domain experts to directly describe and re-organise their data [25].

Requirements were detailed for 23 environmental-science communities, including our two, by ENVRI [3]. Its conclusion proposed common provision of high-level services to make them sustainable and to facilitate boundary-crossing collaboration. This motivated and shaped DARE's goals. It is formalised as the ENVRI Reference Model[15] which provides definitions of common functional requirements decomposed into viewpoints progressing from conceptual to technical.

The analysis of ten-years progress in scientific workflows [26] posed the issue of formulating methods for collaborating communities in ways that sustained interpretations across time and contexts as one of the three priority challenges. Conceptualisation is a necessary step towards this goal.

Using shared catalogues as a basis for integration is central to the VRE4EIC project, developing research environments for collaborating research communities [27]. It demonstrates the integration potential. The SWITCH project shows its KB supporting algorithms for enactment-target selection, optimisation, mapping and coping with heterogeneity as well as combining the developers' and orchestration viewpoints [28]. The DCAT recommendation from W3C delivers an extensive precise vocabulary for integrating data catalogues [10], a dominant form of collections. Trani has demonstrated how this can be used in the EPOS context that includes our seismologists [19], using his methodology to engage communities in the maintenance of their core concepts [4].

WaaS, where the computational context is built when necessary, supported a data-intensive medical application [29]. The DRIP system optimises deployment of workflows across resources taking account of costs and reliability [28]. Provenance-template services were prototyped in the ENVRI context [30]; DARE uses and develops these.

VIII. Conclusions and future work

We have clarified the requirements for researchers, engineers and decision makers addressing today's challenges. They need a methodology, research environment and tools to:

1) collaborate across existing boundaries,
2) exploit data effectively but compliantly,
3) use and adapt the increasing computational power, and
4) combine stability with agility.

They face inevitable complexity combining disciplines, in their phenomena and models, in their data and in the interaction between established practices and external forces. They have to take responsibility, and make well-informed decisions. They have to understand their work *context* and perform *actions* that achieve their *intent*. They need protection from extraneous detail, intellectual ramps so they can expand their capacity to act and need tools to analyse what has been done and verify effects.

DARE has developed a logical architecture and an implementation framework that can deliver this. The architecture enables users to control four aspects of their work in harmony.

[15]https://wiki.envri.eu/display/EC/ENVRI+Reference+Model

1) *Concepts* What they are thinking and talking about.
2) *Methods* What they do to anything in their world.
3) *Data* The digital representation of anything.
4) *Collections* Identified and managed populations.

Individuals work in a context tuned to their work, initially a clone of a standard context for their work which they adapt as they work. The architecture must maintain relationships between contexts.

The framework comprises of three subsystems.

1) *Dare Knowledge Base (DKB)* which holds information to deliver the architectural logic, to support abstractions, and for algorithms in the other subsystems.
2) *Workflows as a Service (WaaS)* which validates, optimises and performs actions.
3) *The provenance system (P4)* which collects, preserves and protects a history of what has happened.

The DKB holds or references information, such as the concepts used, their relationships, properties and representations. It maintains relationships for methods such as the concepts they work on and produce, their implementations and the links to provenance records of their use. It may mine statistics from those records for optimisers to use. It tracks the relationships between and current form of contexts. The WaaS handles any means of specifying actions and offers a repertoire of built-in actions. The P4 offers controls to users and developers to specify detail, link with domain data and organise presentations of large histories. Virtualisation and containerisation is key to feasibility and sustainability of the framework. There is currently a prototype with sophisticated versions of WaaS and P4, and a preliminary version of DKB [18]. This is already used by two application communities.

We are investigating the information needed to fully support CMDC harmony, presentation and implementation. The support for contexts has yet to be investigated. Agile rapid development means that in one context its users can remodel their part of the world. Meanwhile, many users pursue production and expect stability, though they make moderate changes in their spaces continuously. But their worlds need to remain connected for all their overlapping interests. This requires a methodology, tools and underpinning information already investigated by Trani [8]. We hope others will join us in helping develop a good approach to making experts reliably self-sufficient in multi-everything communities collaborating to address big challenges. We invite experts from many application domains and a diversity of computer scientists to work together to deliver a sustainable and evolving conceptual platform.

ACKNOWLEDGEMENTS

DARE builds on a succession of prior EU projects: ADMIRE, VERCE, ENVRI, ENVRIplus, and on nationally funded projects. It depends on the coordination of e-Infrastructures by EOSC. The authors also thank Ghita Berrada, Oscar Corcho and Luca Trani for advice on early drafts and the developers and systems teams that have delivered and tested the platform.

REFERENCES

[1] V. Ramakrishnan, Gene Machine, Oneworld Publications, 2018.
[2] M. Atkinson, et al., VERCE delivers a productive e-Science environment for seismology research, in: Proc. IEEE eScience, 2015.
[3] M. P. Atkinson, et al., Deliverable 5.1: A consistent characterisation of ... research infrastructures, Tech. rep., ENVRIplus project (2016).
[4] L. Trani, M. Atkinson, D. Bailo, R. Paciello, R. Filgueira, Establishing core concepts for information-powered collaborations, FGCS 89 (2018) 421–437.
[5] P. Wittenburg, et al., Digital objects as drivers towards convergence in data infrastructures, Tech. rep., GO FAIR Office, Leiden (2018).
[6] A. Spinuso, M. Atkinson, F. Magnoni, Active provenance for Data-Intensive workflows: engaging users and developers, in: Proc. eScience BC2DC workshop, 2019.
[7] R. Zhao, M. Atkinson, Towards a computer-interpretable actionable formal model to encode data governance rules, in: Proc. eScience workshop BC2DC, 2019.
[8] L. Trani, A methodology to sustain common information spaces for research collaborations, Ph.D. thesis, Uni. of Edinburgh (2019). URL https://www.era.lib.ed.ac.uk/handle/1842/36139
[9] R. Filgueira, et al., dispel4py: A Python Framework for Data-Intensive Scientific Computing, International Journal of HPC Applications.
[10] A. G. Beltran, et al., Data Catalog Vocabulary (DCAT) – revised edition., Tech. rep., W3C (2018).
[11] P. Amstutz, M. Crusoe, N. Tijanić, B. Chapman, J. Chilton, M. Heuer, et al., common workflow language, Tech. rep., W3C.
[12] F. Z. Khan, et al., CWLProv–Interoperable retrospective provenance capture and its challenges.
[13] A. Spinuso, Active provenance for data intensive research, PhD Thesis, The University of Edinburgh, 2018.
[14] K. Belhajjame, O. Corcho, D. Garijo, J. Zhao, D. Newman, G. Klyne, K. Page, M. Roos, Workflow-Centric Research Objects : A First Class Citizen in the Scholarly Discourse 1–12.
[15] M. Atkinson, M. Parsons, The digital-data challenge, in: The DATA Bonanza, John Wiley & Sons, Inc., 2013, Ch. 1, pp. 5–13. doi:10.1002/9781118540343.ch1.
[16] C. Pagé, et al., Review CLIMATE4IMPACT services and objectives, EC FP7-IS-ENES2 Project (2016).
[17] C. Pagé, A. Spinuso., Requirements and Test Cases I, EC H2020-DARE Project Deliverable (2018).
[18] I. A. Klampanos, et al., DARE: A Reflective Platform Designed to Enable Agile Data-Driven Research on the Cloud, in: Proc. eScience workshop BC2DC, 2019.
[19] L. Trani, R. Paciello, M. Sbarra, D. Ulbricht, the EPOS IT Team, Representing Core Concepts for solid-Earth sciences with DCAT – the EPOS-DCAT Application Profile, Geophysical Research Abstracts 20.
[20] D. Peter, D. Komatitsch, Y. Luo, et al., Forward and adjoint simulations of seismic wave propagation on fully unstructured hexahedral meshes, Geophys. J. Int. 186 (2011) 721–789.
[21] L. Moreau, B. Batlajery, T. D. Huynh, D. Michaelides, H. Packer, A templating system to generate provenance, IEEE Transactions on Software Engineering PP (99) (2017) 1–1. doi:10.1109/TSE.2017.2659745.
[22] C. Pagé, W. S. D. Cerff, M. Plieger, A. Spinuso, X. Pivan, Ease access to climate simulations for researchers: IS-ENES climate4impact, in: Proc. eScience workshop BC2DC, 2019.
[23] L. Schubert, K. G. Jeffery, New Software Engineering requirements in Clouds and large-scale systems, IEEE Cloud Computing 2 (2015) 48–58.
[24] J. Myers, M. Hedstrom, D. Akmon, et al., Towards sustainable curation and preservation: The SEAD project's data services approach, in: IEEE e-Science 2015, IEEE, 2015, pp. 485–494.
[25] R. E. Schuler, C. Kesselman, A high-level user-oriented framework for database evolution, in: Proc. SSDBM, 2019, pp. 157–168.
[26] M. Atkinson, S. Gesing, J. Montagnat, I. Taylor, Scientific Workflows: Past, Present and Future, FGCS 75 (2017) 216–227.
[27] P. Martin, et al., mapping heterogeneous research infrastructure metadata into a unified catalogue for use in a generic vre, FGCS 101.
[28] P. Stefanic, et al., SWITCH workbench: A novel approach for the development and deployment of time-critical microservice-based cloud-native applications, FGCS 101.
[29] R. F. da Silva, et al., Using Simple PID-inspired Controllers for Online Resilient Resource Management of ... Workflows, FGCS 101.

[30] B. Magagna, et al., Data provenance and tracing for environmental sciences: system design, Deliverable D8.5, ENVRIplus (2018).