Canonical Workflow Framework for Research
CWFR

- Position Paper –
Version 2

The CWFR Group
Editors: Alex Hardisty, Peter Wittenburg
December 2020

With this paper we want to describe the motivation and basic ideas behind CWFR. Two working meetings were held to discuss the CWFR concept and to relate it with other work around "workflows" that has already been done. We intend to further develop this paper dependent on the growing insights based on the discussions and interactions we are planning to organise.

1. Motivation for CWFR

Modern trends in data science (increasing volumes and complexity[1]) indicate clearly that automated workflows or workflow fragments will make data practices in the data labs[2] much more efficient. There will always be exceptions where manual steps will be necessary to look at specific data, but deep insights into research processes [1] indicate that there are many recurring patterns across disciplines and silos that lend themselves for automation, although the individual researchers might not be aware of these patterns. This trend was anticipated earlier and many IT specialists worked on technical workflow frameworks over many years which led to a sequence of fashions starting with languages such as BPEL, and the first-generation languages of tools like Kepler, Taverna, KNIME, etc. This was followed by the emergence of a more open 'Common Workflow Language' and implementations of the latest generations of tools such as Tarvados, Toil, Pegasus, REANA, IPython, Galaxis, Jupyter, etc[3].

Yet, these technical workflow frameworks are not as widely used in daily practices in the data labs as one might expect. There are several reasons for this gap such as lack of awareness, lack of experts, under-resourced departments, lack of trust in stable technologies, etc. What we can observe is that the bulk of scientific work is still done employing individually conceived procedures based on manual activities. Often that includes the use of short term 'one-off' scripts automating small fragments of research workflows or can involve the use of research tools that include some workflow steps and user-oriented workflow fragments serving specialist needs. In a few advanced labs, mixed teams of researchers and IT specialists are working on more substantial and sustainable workflows that, however, are not meant to be generic or reusable in other research environments. An increasing number of young researchers are making use of Jupyter-like frameworks to implement workflow or script fragments for their analytics to support interoperability and reproducibility which in turn requires provenance information.

---

[1] Often the 4 Vs are mentioned: Volume, velocity, variety and veracity in this context

[2] We are using the generic term „data lab" to point to all institutions, departments, or laboratories that generate data with the help of observations, experiments, analyses and simulations, that manage data or that are processing and analyzing data.

[3] We do not include references to all technologies, since they are easy to find on the web.

Thus, we can identify two paradoxes related to workflows:

1. There is a huge gap between the large investments in IT-based workflow technology on the one hand and the degree of usage of workflow frameworks in the data labs on the other hand.
2. The individual data scientists believe that their work is unique, but one can clearly observe recurring patterns in data generation, management and analysis.

Having analysed data practices, we can identify a third paradox.

3. Many researchers are aware of the FAIR principles and see their relevance for making data practices more efficient. However, currently they are not ready to change their research practices in the data labs fabrics (see Figure 1). They shift FAIRness to the end stage of a project – the publication step which is mostly linked with a classical scientific publication.

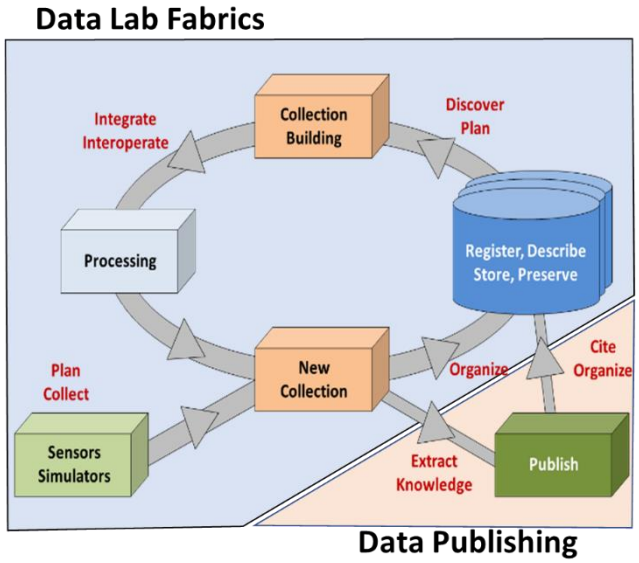


Figure 1. This diagram became the basic one in the RDA Data Fabric group indicating the work in the data labs. New and already existing data, stored in distributed repositories, are organised into collections and subject of some processing resulting in new data which again will be organised according to some principles and stored in repositories. This process is being continued until sufficient evidence for some theory has been achieved which allows publishing results. Only some data will be associated with such publications. It is known that more than 90% of the data being created will stay in the repositories and not be made available in the classical publication sense.

The implications of this attitude are that most data being created and managed in the lab's data fabrics will not be made FAIR and that making data FAIR at the end of a project is by factors more costly and tedious than doing it "by design" [2].

It is obvious that there are gaps between what principles request and technologies offer on the one hand and what researchers in the data labs are doing on the other hand. There are several reasons for this discrepancy. (1) Researchers rightly refuse to accept new methods if the tools are not ready and sustained, and if new inefficiencies and disruptions can be expected. That said, researchers are interested in excellent tools if they help them carry out their research as smoothly and as timely as possible to publish their results in highly competitive scenarios. (2) In general, researchers want to be as independent as possible. Any technology that requires help of scarce experts will only be accepted by some researchers working with advanced tools. (3) Researchers want to demonstrate that their work is special and determined by their individual approaches. This creates psychological barriers to accept standardised workflow approaches.

As indicated above, there are other factors that hamper adoption of workflows such as lack of awareness and lack of funding to hire experts.

2. Concept of CWFR

To describe CWFR we first address the basic ingredients of CWFR, then the required "glue" that has the potential to bind things together and, finally, the benefits that could be achieved.

2.1 Ingredients of CWFR

The above-mentioned gaps were the motivation to start brainstorming about CWFR which must be guided by studying the researchers' practices and not be determined by IT considerations. We therefore see CWFR as a new layer on top of workflow technologies. This layer can evolve over time according to new needs independent of the changes of the underlying workflow technologies. CWFR is thus not about re-inventing the wheel at the technology level but adding another layer which is closer to the researchers' practices. This implies that CWFR components should ideally be reusable independent of whether people use the Common Workflow Language, Jupyter notebooks, Taverna or any other orchestration/description technology which exists or will emerge.

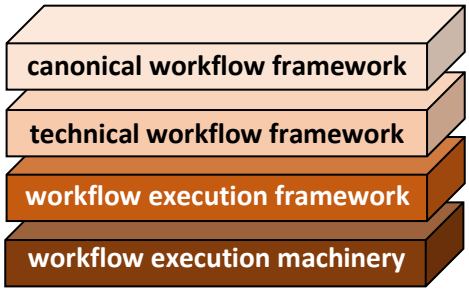
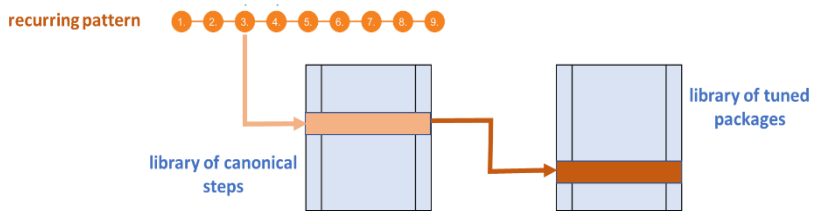


Figure 2. This diagram indicates that CWFR is a layer that is conceptually close to the practices of researchers in the labs and that is on top of the stack of workflow technologies. Ideally, the existing technical workflow frameworks would only to be adapted slightly.

For CWFR we can identify three elements (see figure 3):

1. patterns of recurring canonical steps in research activities;
2. libraries of canonical steps; and,
3. packages per canonical step.

Figure 3. This diagram shows the 3 basic elements of CWFR: (1) recurring patterns, (2) library of canonical steps and (3) libraries of specialised packages per step.



Each canonical step may be associated with packages that are specific for a certain context. An example could be the format for an ethical review request which will vary in some detail between institutions. Thus, in addition to a library of canonical steps there will be libraries of specialised packages. These three elements or sub-layers of CWFR are indicated in figure 3.

An analysis of experiments with humans in some labs revealed that recurring patterns of atomic canonical steps could be found. This is described in Figure 4. It needs to be noted that

(1) not all steps are being carried out by all experimental labs, i.e. in some labs some steps might be skipped,

(2) the apparent linear sequence in reality is subject to many iterations, i.e. researchers may go back to an earlier step and change specifications and start again, and

(3) for some steps there will already be excellent and specialised software, such as for experiment execution, which needs to be embedded.

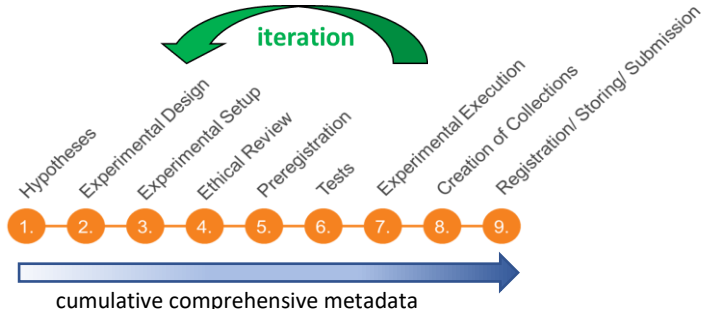


Figure 4. This diagram shows the recurrent pattern occurring in experiments with human subjects as they occur in several labs across some disciplines. Nine canonical steps could be identified and not all of them are being used in all labs.

Of course, there are different recurring patterns in the many research disciplines and workflow practices and there will be many different canonical steps (but not infinite). In addition to analysing the experimental workflow pattern as indicated above, colleagues in the CWFR group analysed a typical workflow for machine learning experiments[4] and in two working meeting sessions 17 colleagues presented workflow patterns that they use[5]. All this work indicates the diversity of workflow applications and underscores the need for an analysis of the components that can be reused across patterns at the beginning of this CWFR initiative. At the same time, some functions such as "resolve a PID, analyse its Kernel Values, register a PID, read metadata, analyse metadata on certain references, add provenance to metadata and others" already appear to be widely reused.

2.2 Interoperable Glue in CWFR

Workflows can be described as a set of activities A_1 to A_k all contributing in some form to state k (see Figure 5). We call the cumulative description achieved at state k the CWFR State Digital Object at state k, or in short the CWFR-DO_k. It captures all relevant information aggregated throughout all states or captures references to information that has been generated throughout these steps. CWFR-DOs are true FAIR-DOs (FDO) since they are identified by a Handle and have some metadata the categories of which have been specified and are registered in an open registry. Further, their content must be typed attributes to allow machine actionability.

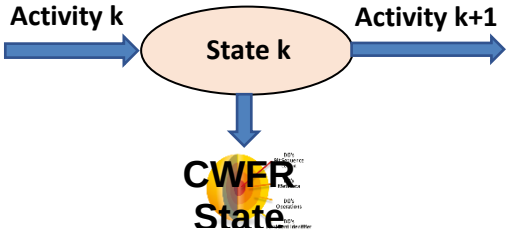


Figure 5. This figure indicates at an abstract level that workflows consist of sequences of activities leading to states that are fully described by a state Digital Object.

CWFR-DOs are the interoperable glue for CWFR workflows. Since we do not know the scope of canonical steps the CWFR-DO must be structured and flexible, i.e. if someone registers a new canonical step it is necessary to define the attributes required for this step and to organise CWFR-DO in a way that a chunk of attributes can be added without interfering with other descriptions (see Figure 6). PIDs would be used to reference to each of the canonical steps. A set of typed attributes allowing machines to process and reuse information at later steps are added at each step in the workflow process.

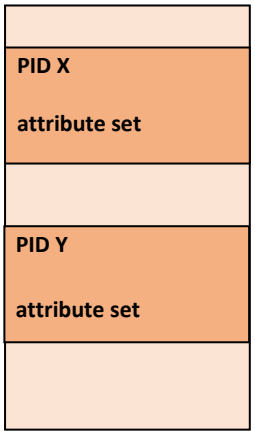


Figure 6. This figure indicates a possible structure of the CWFR-DO catering for chunks of attributes inserted by the different activities.

Let us take as an example a request for an ethical review. During the early steps, information such as "researcher name, department, research intentions, experimental design, experimental setup" has been entered in a typed way. Obviously, this kind of information needs to be presented to the ethical review board, i.e. a mapper must be integrated that pulls out the relevant information from the already made descriptions and inserts it in the right way into the form needed. The action at that step would include a mapper, sending the request form and waiting for an answer. Since the information in CWFR-DO is strictly typed, mappers could be easily formulated in a declarative way without programming capabilities. Other actions could also be supported in a way understandable by laypersons.

Since all CWFR-DOs will have a PID and are organised in a project registry, it would be easy for researchers to go back to an earlier state and redo the actions (see Figure 6). Therefore, all relevant information would be available to the users. With proper editors available, one can assume that with a few changes to the CWFR-DO the researcher can redo his workflow orchestration task.

[4] https://osf.io/umhy5/
[5] https://osf.io/9ut4p/

It is obvious that the success of CWFR will depend on a well-chosen approach to implement CWFR-DO. This will require further hard work and inspecting what has already been done.

It should be stressed again that workflows, workflow templates as well as their components and elements such as CWFR-DO should be findable, accessible, interoperable, and reusable (FAIR). For this purpose, any research stage reflected by a workflow template or implemented in a workflow should be described as formally as possible with semantic metadata, use knowledge-based languages, be accessible for reuse within the workflow or without it. CWFR-DO need to be first-class citizens on the Internet.

2.3 Benefits of CWFR

The benefits of CWFR are significant:

- Researchers would be able to plug relevant canonical actions and tuned packages into their workflows.
- Researchers could perform iterations of work steps without the need to enter in all information again and therefore enormously increase efficiency through CWFR.
- Researchers could use a machinery that automatically generates FAIR DO, thus FAIR-compliant data, without needing to bother with technicalities.
- Researchers could create stable references without having to collect and archive detailed provenance information allowing for future replication of entire workflows by themselves.

3. Feasibility of CWFR

Creating the CWFR landscape seems to be a huge task, so questions about its feasibility are appreciated[6]. Reasons for assuming feasibility are:

- The number of canonical steps across a wide domain of research disciplines is quite limited.
- The set of specialised software to collect (for example) subjects and to run the experiments is limited making it easy to define the parameters that are required and thus to define the set of attributes needed for the corresponding canonical steps.
- The set of mappers that transform attribute values within CWFR-DO to packages is limited.
- The set of DO related commands (e.g., resolve PID, analyse kernel attributes, register PID, etc.) is standard and can easily be expanded.

Once such a CWFR workflow is implemented, the achievement would be great since many labs across disciplines would benefit.

We should not underestimate, however, the effort needed to create simple software to help the researcher in various steps, such as text editors allowing to inspect and modify the content of CWFR-DOs. Also, this aspect will require further analysis.

4. Related Work to CWFR

During the two working meetings it became obvious that much work has already been done in the area of technical workflow support and that in some advanced centres and IT departments experts are implementing workflows with larger scope to facilitate data-driven research.

[6] For the concrete case presented in Figure 3, it was obvious that the task was doable given the many code snippets which are already around and which would have to be integrated.

4.1 Spectrum of Workflows

The spectrum of these efforts is wide and covers the whole bandwidth between the two extremes illustrated in the diagram below (see Figure 7). At one extreme, “automated, structured, repeatable workflows” executing in a custom virtual research environment (science gateway) offer the possibility to execute tens of thousands of workflow runs with varying data and/or parameter sets, keeping track of all the results. On the other end of the spectrum, workflows are more fragmented, bespoke and ‘personal’, often used as part of simple and more manual pipelines for cleaning, preparing and

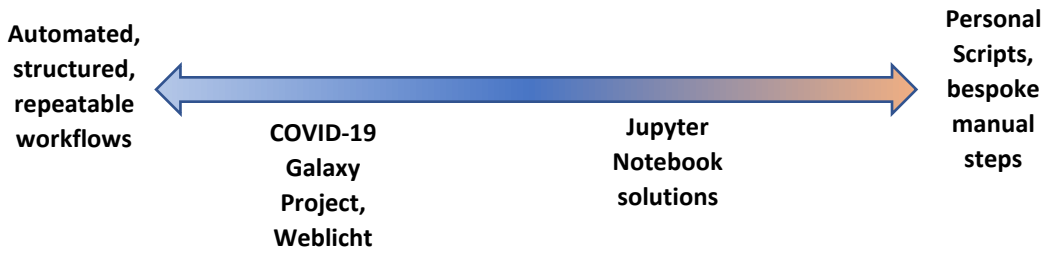


Figure 7. This figure indicates the spectrum of methods being used to realise research workflows and how some known technologies fit into this spectrum.

analysing data. Solutions such as the COVID-19 Galaxy Project and CLARIN’s “Weblicht” framework are close to the left pole (see diagram) while Jupyter notebook solutions are often closer to the right pole.

Despite the provisioning over the years of more elegant and easier to use workflow frameworks, there will always be researchers who will rely on individual solutions driven by their specialised needs.

4.2 Fashions of Workflow Technology

Simple pipeline constructions implemented with the help of some scripting language offered by operating systems have a long tradition. Originally, these were used mainly for repeating system management operations or to organise and execute computations at HPC systems. In 2003, BPEL, one of the first workflow languages, was launched originating from businesses which wanted to structure complex and repeated business processes. Also, in 2003, Triana and Taverna were started as open source projects to create workflow frameworks in the scientific domain. Several other typical workflow languages and many frameworks with different foci were designed and implemented, for example, Kepler, Kmime, Dispel. In 2014 work on the Common Workflow Language (CWL) began with a focus on supporting portability and thus reproducibility across different computational environments. CWL is being used now by added layer workflow frameworks and became relatively popular in the research domain. Most of these languages require, however, software programming skills to exploit them effectively. Galaxy, on the contrary, is offering a framework that can be controlled via APIs and programming skills but also offers user-interfaces that enables every researcher without programming skills to compose and run workflows if the appropriate components (tools) have been integrated beforehand.

Motivated by the success of the Python language, also in 2014, a parallel attempt was made with Jupyter notebooks to develop an interactive framework to support data science and scientific computing allowing to embed code written in a variety of different programming languages. Jupyter therefore supports the concept that much excellent software is already around to do complex calculations in different fields such as ab initio calculations in natural sciences or highly optimised machine learning packages being used across research domains. Jupyter allows embedding such codes into workflows. It is a popular choice being applied now by an increasing number of young researchers in particular.

It should be noted that various other specialised software packages exist that include some workflow stages combined with discipline-oriented functions to support researchers in their data-intensive

work. One of these software packages including a distinct workflow engine is iRODS, a comprehensive package to support data management and processing. In the realm of RDA Practical Policy, a broad set of workflow fragments (practical policies) was analysed and implemented.

4.3 Digital Object technology

The concept of Digital Objects has been defined by RDA Data Foundation & Technology (DFT) [3] which is widely based in the conceptual work of Kahn and Wilenski [4]. The strength of the DO concept is in its persistent identification, abstraction (a DO can include all kinds of content), its persistent binding (binding all relevant information relevant to access and use the encoded content) and its encapsulation capacities (DO types can be related with procedures). DOs therefore are robust "atoms", being first-class elements on the Internet independent of technology (see Figure 8). Recently, the FAIR principles were assigned to DOs which resulted in the concept of FAIR DOs (FDO)[7]. DOs are mostly FAIR already[8], however, DOs do not make any statements about the metadata provided by the communities while FAIR requires "machine actionability" of metadata as well which is dependent on changes by the research communities. A set of specifications and software packages are around to support DO work such as (1) the DO Interface protocol (DOIP) which exists as specification and as software development kit, (2) the Handle registration and resolution software being used worldwide since 20 years, (3) the Data Type Registry to register Kernel metadata which is associated with Handles, and (4) a configurable software for managing digital objects and as a showcase for the use of DOIP. (F)DOs would be therefore an excellent basis for implementing CWFR State DOs.

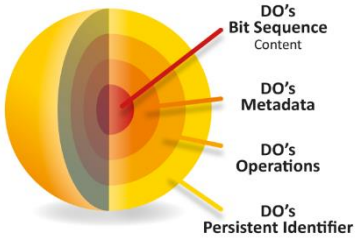


Figure 8. This figure is now often being used to indicate the stable atomic nature of FDOs, their abstraction capacity, their binding role, and their potential for encapsulation.

There is a potential problem with DOs. As the internal bit string is updated, or the metadata is updated (automatically or manually), or the operations (methods) are changed, we face the identity problem; there are substantial implications for curation and provenance in keeping track of the state of the DO as a whole with each update of its constituent parts. In this state it is not atomic. The problem increases if a DO is an integration of other DOs.

4.4 CWFR Digital Objects

In 2.2 above we introduced the concept of CWFR DOs to incorporate all information of a workflow up to an actual state, allowing us to look back at what has exactly been done, to start execution again after a while and to replicate a given workflow in multiple computing environments. This aspect of workflow technology is not new. Here we want to introduce two example mechanisms (UIMA and Research Objects) that give inspiration for arranging canonical workflow steps and recording all state information.

In 2013, IBM announced its comprehensive 'Watson' technology to deal with all kinds of data/information/knowledge processing applications, which is now being used for many different types of complex information systems. One of the core elements the Watson experts needed was a construct that would allow the many different modules to interact. UIMA (Unstructured Information Management Architecture) was designed exactly as this core element for interaction. It is highly structured, flexible, and requires typed attributes. UIMA has been standardised (OASIS) and became an open source Apache project[9].

---

[7] Announcement from German EU presidency.

[8] https://github.com/GEDE-RDA-Europe/GEDE/tree/master/FAIR%20Digital%20Objects

[9] https://uima.apache.org/

The concept of Research Objects (RO) emerged from the eScience work and is a method for the identification, aggregation, and exchange of scholarly information on the Web. ROs is a way to associate all related resources about a scientific investigation so that they can be shared using a single identifier. This includes the possibility to replicate research work. ROs mainly rely on the Web technology stack, and is being used in a number of research projects. Recently, the concept of RO-Crates has been introduced to establish a lightweight approach to packaging research data with their metadata making use of schema.org and JSON-LD.

ROs work well if they are frozen to represent the state of the research at a given time and a new RO is initiated for subsequent work; this assists in overcoming the problem with DOs outlined above.

A merging of the earlier concepts of UIMA and/or RO with the concept of FDO could be beneficial to implement CWFR State DOs.

5. Insights in CWFR Use Cases

During the two working meetings 17 use cases from different research fields were presented. These covered a wide spectrum of workflow solutions. In this section we give a summary in the form of observations of these use cases knowing that they are just a small selection of efforts currently being invested.

- Experts in many research domains are working on workflow use cases, yet often addressing workflow fragments covering specific steps in order to limit complexity.
- Jupyter notebooks are very popular in natural sciences in particular, and young students in particular exhibit facility with technical matters such as software development.
- There is a proliferation of smart workflow tools with different foci servicing special needs and different groups of users.
  Note: Here, https://s.apache.org/existing-workflow-systems more than 280 workflow tools are listed.
- Workflow experiments are covering the whole data life cycle from creation and processing to publication, long-term archival and (re)use.
- The number of training courses in universities about workflow technologies is increasing. However, employing institutions lag in providing guidelines for improving the FAIRness and efficiency of data-driven projects, which would increase participation and the prospects for real change.
- Good practices such as DO, RO and UIMA do not play a role in the presented use cases.
- Excellent research tools are indeed being used in many institutes partly including a few workflow steps. It will be a necessity that CWFR offers possibilities to embed them.

References

[1] K. Jeffery, P. Wittenburg, L. Lannom, G. Strawn, C. Biniossek, D. Betz, C. Blanchi. Not Ready for Convergence in Data Infrastructures. Data Intelligence. 2021. Vol.1.
[2] N. Beagrie: Keeping Research Data Safe -JISC Research Data Digital Preservation Costs Study; APA Conference Budapest, 2008
[3] G. Berg-Cross, R. Ritz, P. Wittenburg: RDA DFT Core Terms and Model; https://b2share.eudat.eu/records/10d52078bf6e416588ab18e2069fc636
[4] R. Kahn, R. Wilenski: A framework for distributed digital object services; International Journal on Digital Libraries (2006) 6(2): 115–123DOI 10.1007/s00799-005-0128-x

Further Reading

Common motifs in scientific workflows: An empirical analysis, https://doi.org/10.1016/j.future.2013.09.018, D Garijo, P Alper, K Belhajjame, O Corcho, Y Gil, C Goble, Future Generation Computer Systems 36, 338-351 - identifies some patterns from examples
Abstract, link, publish, exploit: An end to end framework for workflow sharing, https://doi.org/10.1016/j.future.2017.01.008 - looks at abstractions and instantiation
Andrio, P., Hospital, A., Conejero, J. *et al.* BioExcel Building Blocks, a software library for interoperable biomolecular simulation workflows. *Sci Data* **6,** 169 (2019). https://doi.org/10.1038/s41597-019-0177-4 - libraries and building blocks

Appendix. CWFR Group

Stan Ahalt	RENCI
Sadaf Alam	CSCS
Ivonne Anders	DKRZ
Carsten Baldauf	FHI
Dirk Betz	U Cologne
Claudia Biniossek	U Cologne
Christophe Blanchi	DONA
Damien Boulanger	CNRS
Alexander Czmiel	BBAW
Romain David	ERINHA
Carole Goble	U Manchester
Christian Grimm	DFN
Volker Gülzow	DESY
Ingemar Häggström	EISCAT
Alex Hardisty	U Cardiff
Margareta Hellström	U Lund
Felix Henninger	U Mannheim
Martin Jäkel	ZHAW
Keith Jeffery	KEITHGJEFFERYCONSULTANTS
Reinhold Kliegl	U Potsdam
Dimitris Koureas	NATURALIS
Larry Lannom	CNRI
Thomas Lauer	U Erfurt
JianHui Lee	CNIC
Daniel Mallmann	FZ Juelich
Katja Marciniak	BBAW
Ralph Müller-Pfefferkorn	TU Dresden
Christian Ohmann	ECRIN
Per Öster	CSC
Limor Peer	U Yale
Peters	DKRZ
Nici Pfeiffer	OSF
Beth Plale	U Indiana
Ulrich Schwardmann	GWDG
Nikolay Skvortsov	RAS
Alessandro Spinuso	KNMI
Rainer Stotzka	KIT
George Strawn	BRDI
Dieter van Uytvanck	CLARIN
Philipp Wieder	GWDG
Peter Wittenburg	MPCDF
Martin Zünkeler	KAIROS
Carlo Maria Zwölf	Observ. Paris