Wf4Ever: Advanced Workflow Preservation Technologies for Enhanced Science

STREP FP7-ICT-2007-6 270192

Objective ICT-2009.4.1 b) – "Advanced preservation scenarios"

D2.1: Workflow Lifecycle Management Initial Requirements

Deliverable Co-ordinator: Sean Bechhofer

Deliverable Co-ordinating Institution: University of Manchester (UNIMAN)

Other Authors: Stian Soiland-Reyes, Khalid Belhajjame, Jiten Bhagat

This deliverable provides initial requirements for Workflow Lifecycle Management ("Research Objects").

Document Identifier:	Wf4Ever/2010/D2.1/v1.0	Date due:	May 31, 2011
Class Deliverable:	Wf4Ever 270192	Submission date:	May 31, 2011
Project start date:	December 1, 2010	Version:	v1.0
Project duration:	3 years	State:	Final
		Distribution:	Public

i. Wf4Ever Consortium

This document is a part of the Wf4Ever research project funded by the IST Programme of the Commission of the European Communities by the grant number FP7-ICT-2007-6 270192. The following partners are involved in the project:

Intelligent Software Components S.A. Edificio Testa Avda. del Partenón 16-18, 1º, 7ª Campo de las Naciones, 28042 Madrid Spain Contact person: Dr. Jose Manuel Gómez-Pérez E-mail address: jmgomez@isoco.com	**University of Manchester** Department of Computer Science, University of Manchester, Oxford Road Manchester, M13 9PL United Kingdom Contact person: Professor Carole Goble E-mail address: carole.goble@manchester.ac.uk
Universidad Politécnica de Madrid Departamento de Inteligencia Artificial Facultad de Informática, UPM 28660 Boadilla del Monte, Madrid Spain Contact person: Dr. Oscar Corcho E-mail address: ocorcho@fi.upm.es	**University of Oxford** Department of Zoology University of Oxford South Parks Road, Oxford OX1 3PS United Kingdom Contact person: Dr. Jun Zhao / Professor David De Roure E-mail address: {jun.zhao@zoo.ox.ac.uk, david.deroure@oerc.ox.ac.uk}
Poznań Supercomputing and Networking Center Network Services Department Poznań Supercomputing and Networking Center Z. Noskowskiego 12/14, 61-704 Poznan Poland Contact person: Dr. Raúl Palma de León E-mail address: rpalma@man.poznan.pl	**Instituto de Astrófísica de Andalucía** Dpto. Astronomía Extragaláctica Instituto Astrofísica Andalucía Glorieta de la Astronomía s/n 18008 Granada, Spain Contact person: Dr. Lourdes Verdes-Montenegro E-mail address: lourdes@iaa.es
Leiden University Medical Centre Department of Human Genetics Leiden University Medical Centre Albinusdreef 2, 2333 ZA Leiden The Netherlands Contact person: Dr. Marco Roos E-mail address: M.Roos1@uva.nl	

ii. Change Log

Version	Date	Amended by	Changes
0.1	2010-12-01	Sean Bechhofer	Created Document/Initial Draft
0.2	2011-05-10	Sean Bechhofer	Lifecycle States, User Roles
0.3	2011-05-13	Stian Soiland-Reyes	Added user and technical requirements
0.4	2011-05-18	Khalid Belhajjame	Added text on technical requirements
0.5	2011-05-18	Khalid Belhajjame	Summary of the use cases
0.6	2011-05-19	Sean Bechhofer	SoA, Introduction
0.7	2011-05-24	Jiten Bhagat	Minor corrections
0.8	2011-05-24	Stian Soiland-Reyes, Jiten Bhagat	Updates to the user and tech requirements; updated numbering system of requirements
0.9	2011-05-24	Sean Bechhofer	Minor edits
0.10	2011-05-25	Jiten Bhagat	Fleshed out tech requirements; completed mapping with user requirements
0.11	2011-05-25	Stian Soiland-Reyes	Updated the “distributed nature” of ROs section
1.0	2001-05-31	Sean Bechhofer	Final edits

iii. Executive Summary

This document presents initial requirements for workflow lifecycle management features, including an analysis of the representational and lifecycle management needs for Research Objects, the needs for workflow validation and reproducibility to address the workflow decay issues, and the needs for workflow abstraction to aid in indexing workflows, classifying them, comparing them with each other, and explaining their execution and behaviour in a way closer to the actual conceptualization of problems by scientists. Requirements are gathered through the use of Wf4Ever case studies.

The initial list of requirements provides a starting point for design and implementation work. Requirements will change over time, however, and this should be considered a *living document* that will be revisited regularly throughout the life of the project, taking into account the continuous software development and integration model that we apply.

This document (D2.1) provides an overview of context and terms used in deliverables D3.1 and D4.1

iv. Table of contents

v. List of Figures

1. Introduction

Changes are occurring in the ways in which research is conducted. Within wholly digital environments, methods such as scientific workflows, research protocols, standard operating procedures and algorithms for analysis or simulation are used to manipulate and produce data. Experimental or observational data and scientific models are typically “born digital” with no physical counterpart. Shifts in dissemination mechanisms are thus leading towards increasing use of electronic publication methods. Traditional paper publications are, in the main linear and human (rather than machine) readable. A simple move from paper-based to electronic publication, however, does not necessarily make a scientific output decomposable. Nor does it guarantee that outputs, results or methods are reusable.

Studies continue to show that research in all fields is increasingly collaborative [olson]. Most scientific and engineering domains would benefit from being able to “borrow strength” from the outputs of other research, not only in information to reason over but also in data to incorporate in the modelling task at hand. Scientific practice is based on publication of results being associated with provenance to aid interpretation and trust, and description of methods to support reproducibility. However, simply publishing data out of context fails to: 1) reflect the research methodology; and 2) respect the rights and reputation of the researcher.

There is thus a need for a framework that facilitates the reuse and exchange of digital knowledge and allows the representation of this contextual information.

The Wf4Ever project proposes the use of Research Objects (ROs) in order to address the issues identified above. ROs are semantically rich aggregations of resources that provide a layer of structure on top of information delivered as, for example, Linked Data. An RO provides a container for a principled aggregation of resources, produced and consumed by common services and shareable within and across organisational boundaries. An RO bundles together essential information relating to experiments and investigations. This includes not only the data used, and methods employed to produce and analyse that data, but also annotations describing data, methods and the people involved in the investigation. ROs have the potential to support reproducible science, allowing the validation of results, and to encourage reuse of existing methods, promoting more efficient use of resources.

In the context of Wf4Ever we focus on those Research Objects that encapsulate scientific workflows, and consider how the application of an RO approach can support the preservation of those workflows and the results obtained through their execution. Research Objects allow capturing of workflow dependencies to scripts, services and datasets which are necessary for executing the workflow, we can therefore consider the task of preserving a workflow to be a specialisation of the more general task of preserving a Research Object.

Motivations and high level discussions of the features and lifecycle of Research Objects have been described elsewhere [bechhofer, bechhofer2]. In this document, we provide a deeper analysis of the user and technical requirements placed on ROs by the two Wf4Ever user case studies [See D5.1 and D6.1].

2. Deliverable Roadmap

Work packages WP2, WP3 and WP4 are concerned with the definition and implementation of models to support workflow preservation, largely through the definition of Research Objects, supporting the bundling of workflows (or other less formal descriptions of processes) with additional information regarding input/output data, provenance, sharing, access and so on. The key intention is to support reproducibility of computational science, while ensuring the quality of that reproduction.

A collection of Deliverables describe initial requirements that Wf4Ever will use in order to inform development throughout the project. Although these documents can be considered separately, they share aspects and can be considered as a single document presenting overall requirements for Wf4Ever.

These initial deliverables describe the user domains used as an initial focus for the project (Genomics and Astronomy) along with initial requirements gathered from an analysis of use cases taken from those domains.

The following documents are included in this initial set:

- **D2.1** Workflow Lifecycle Management Initial Requirements
- **D3.1** Workflow Evolution, Sharing and Collaboration Initial Requirements
- **D4.1** Workflow Integrity and Authenticity Maintenance Initial Requirements
- **D5.1** Astronomy Workflow Preservation Requirements
- **D6.1** Genomics Workflow Preservation Requirements

Although they are separate documents, there is significant shared content between them, in particular D2.1, D3.1 and D4.1 all draw from the domain descriptions given in D5.1 and D6.1 and discuss technical requirements. The requirements documents also share commonalities in the methodologies applied to the requirements extraction process. As discussed in the introduction to this document, the notion of Research Objects form the core of our approach to workflow or method preservation and sharing.

This "roadmap" section provides an overview of the materials contained in each document.

D2.1 Workflow Lifecycle Management Initial Requirements (this document) discusses representational and lifecycle needs for Research Objects (information objects intended to provide encapsulations that support the preservation and reuse of workflows with. The document also presents context and terms of reference that are used in the other technical requirements documents, including an identification of *dimensions* of reuse, the user *roles* that are involved in the various processes, and an overview of the methodology used to extract requirements from the user domains. D2.1 should thus be considered as a “master document” for the set of technical requirements.

D3.1 Workflow Evolution, Sharing and Collaboration Initial Requirements provides an analysis of: i) the versioning and evolution needs of Research Objects and the relationships between workflows and their related resources (datasets, services, etc.), ii) the needs for personalised recommendations, and iii) the collaboration spheres concept for sharing and reuse.

D4.1 Workflow Integrity and Authenticity Maintenance Initial Requirements provides requirements for the support of workflow integrity and authenticity maintenance features. This analysis allows understanding of the needs of provenance information for maintaining and evaluating the integrity and authenticity of Research Objects in the Wf4Ever preservation system, and will define the requirements for methods and tools addressing the computation and evaluation of authenticity and integrity.

D5.1 Astronomy Workflow Preservation Requirements characterises the domain of Astrophysical workflow use and requirements for preservation, introducing a number of workflow golden exemplars with use cases and user roles which will drive work during the project.

D6.1 Genomics Workflow Preservation Requirements characterises the domain of Genomics workflow use and requirements for preservation, and introduces workflow golden exemplars, use cases and user roles.

Within this document, we consider the lifecycle of Research Objects. This takes the form of a consideration of the various states that Research Objects may occur in, the transitions that occur between those states, and the roles of the users who interact with the objects. We also consider requirements on the representations or models that will be used for Research Objects.

Note that this document (and the other documents in this collection of deliverables) should be considered as *initial* requirements for the project. They do not attempt to define the models used, and we expect that additional requirements will be identified throughout the course of the project.

This Document (**D2.1**) provides:

- An overall description of the intention and purpose of Research Objects;
- A description of User Roles.
- A discussion of types or dimensions of reuse;
- A description of Research Object lifecycle states;
- Presentation of initial User and Technical Requirements.

3. Methodology

The process of extracting requirements from the user domain descriptions has been performed using a similar methodology to that as used for documents *Workflow Evolution, Sharing, and Collaboration [Deliverable 3.1]* and *Workflow Integrity and Authenticity Maintenance Initial Requirements [Deliverable 4.1]*.

The methodology can be summarized as follows:

- User scenarios: we begin by studying and analyzing some user scenarios or golden exemplars from our target users' domains of activity.
- Isolate user requirements: from the scenarios and exemplars, we distill a set of user requirements.
- User review: we review the distilled requirements with the users
- Project technical requirements: extract technical requirements from the user requirements
- Classify technical requirements: organize the technical requirements into different categories (dimensions)

The intent is that this above process will be iterated as required. In the early stages, we do not try to capture all possible requirements, but rather to focus on those areas that the users perceive will have the greatest impact on their day-to-day work. This is done in the full expectation that subsequent development iterations will introduce and analyze new requirements, the details of which may well depend on experience with implementation and user experience with the earlier requirements.

Our analysis of the use cases also makes use of a number of user *roles* that have been identified. These roles help to characterize the various tasks that users which to perform within each use case. User roles are described in Section 6.

A more detailed description of the steps in the process is given below.

User scenarios

We begin by gathering user scenarios or golden exemplars from our target users' domains of activity.

At this stage, we ask the users to articulate their goals and requirements in terms that are germane to their day-to-day work, without particular reference to the technologies we aim to develop. To the extent that the users are also technically knowledgeable, this separation of concerns might not be as clear-cut as idealized here, but the important thing is that the users feel they can focus on what they would like to achieve rather than the technology needed to deliver on those goals.

It is also important that the users first capture what is important to them, rather than produce a comprehensive list of everything that they might one day find useful. There will be plenty of opportunity later in the overall process to capture additional requirements as they assume greater importance.

Isolate user requirements

From user-provided materials we distill or infer a set of free-standing user requirements.

This involves a close examination of the user supplied materials, and extracting any information that can be interpreted as a goal or requirement of the user in their day-to-day work, and any benefit they may realize by virtue of having any such requirement satisfied.

Following a common agile development practice for describing "User Stories" (http://en.wikipedia.org/wiki/User_story), we aim to capture details that can be articulated in the form:

"As a (type of user) I want (articulation of requirement) so that (description of ensuing benefit to the user)" This form helps to focus attention on a user's needs rather than the technical means whereby they might be satisfied.

See also: http://blog.mountaingoatsoftware.com/advantages-of-the-as-a-user-i-want-user-story-template. The author of this blog post, Mike Kohn, has been credited as being the originator of this form (http://scrummethodology.com/scrum-user-stories/).

User Review

The user representative is asked to review the distilled requirements, to confirm that they do properly reflect the scenario described, or to clarify any misunderstandings there may be in its interpretation. This review process will typically elicit additional requirements and/or benefits that can be included.

Meetings both via teleconference and face-to-face with subject-matter experts in the astronomy and bioinformatics domains have been useful in order to trigger and refine the extraction of requirements.

Project Technical Requirements

Until this point, the focus has been entirely on articulating and prioritizing user goals and requirements, in principle without reference to the technical mechanisms by which they may be realized.

In this step, the user requirements are assessed in the context of a technical deployment environment, and corresponding technical requirements are proposed whereby the requirements can be satisfied.

The technical requirements should co-evolve with both the emerging user requirements, and with the developing technical context and architecture. As such, they provide a clear point of linkage between the user requirements and the technical implementation, and provide a basis for justifying technical design decisions, and prioritizing feature development, according to user requirements.

4. Use case summary

1. Summary of the Astronomy use case

Astronomy is among the first scientific disciplines to embrace and benefit from early development of web-based technologies enabling cross-linking of resources across archives. Our partners from the Instituto Astrofísica Andalucía have in Deliverable 5.1 identified three use cases that are representative of the experiments that are performed within the astronomy field.

The first use case, titled "*propagation of quantities*", showcases the need to update values that are dependent on other volatile values. Specifically, the user is interested in maintaining the freshness of data values that measure the magnitude, distance and intrinsic luminosities of a set of objects. The process by which the freshness of such values is maintained, is implemented using a workflow. Such a workflow is enacted every time values of variables, on which the magnitude, distance or intrinsic luminosities depend, are updated.

The second use case, "*extraction of galaxy samples*", aims to retrieve a set of 2D images from existing catalogues with the objective of identifying a list of potential objects, e.g., companion galaxies and their hosts, that meet given special distribution criteria in the sky.

The last use case, "*Modeling of 3D data of galaxies*", showcases the need for processing and transferring large volumes of data, which are 3D binary cubes with two spatial dimensions and a third one associated with the velocity of the gas emitting the light captured. Such data are generated and processed using workflows.

The above three use cases elicited many requirements, which we will be reported later in this document. It is however worth mentioning at this stage that the main requirements are: (i) the need for automating the processing of data in the astronomical field using workflows, (ii) the need to document and share such workflows, and (iii) the need to version methods and workflows as well as data values.

2. Summary of the Bioinformatics use case

One of the main issues in biomedical research lies in the study of large datasets, and combinations of thereof, with the objective to understand the mechanisms that explain the onset and the progression of human diseases. In this regard, the department of human genetics at Leiden University Medical Centre, a Wf4Ever partner, investigates the genetic background and molecular mechanisms behind a number of rare and common diseases. We summarize in what follows the three use cases put forward by this partner in Deliverable 6.1.

The first use case, "*Metabolic Syndrome*", aims to mine the relationships between the genotype (genetic code) and the phenotype (disease symptoms). This study involves running *in silico* experiments that are enacted by workflows, but also the analysis of relationships between data used as input to the experiments and the data obtained as a result.

The second use case, "*The role of epiGenetics in Huntington's Disease*", aims to investigate the mechanisms leading to HD phenotypes. As for the previous use case, this requires the design and modeling of experiments that combine different types of data sets and analysis tools.

The third use case, "Toxicogenomics – experience from a novice user", aims to interpret the effect of gene transcription factor on the gene expression in the small intestines from wild type and PPARalpha-null mice. To design the experiment that can be used for this study, the user, who is not familiar with workflow managements systems, attempts to design the experiment using the Taverna workbench. In doing so, existing workflow systems that are stored within the myExperiment[1] repository will be used as component (sub-workflows) within the target experiment.

[1] http://www.myexperiment.org

5. State of the Art

Reproducible Science

Mesirov [mesirov] describes the notion of *Accessible Reproducible Research*, where scientific publications should provide clear enough descriptions of the protocols to enable successful repetition and extension. Mesirov describes a *Reproducible Results System* that facilitates the enactment and publication of reproducible research. Such a system should provide the ability to track the provenance of data, analyses and results, and to package them for redistribution/publication. A key role of the publication is *argumentation*: convincing the reader that the conclusions presented do indeed follow from the evidence presented.

De Roure and Goble [deroure] observe that results are “reinforced by reproducibility”, with traditional scholarly lifecycles focused on the need for *reproducibility*. They also argue for the primacy of method, ensuring that users can then reuse those methods in pursuing reproducibility. While traditional “paper” publication can present intellectual arguments, fostering reinforcement requires inclusion of data, methods and results in our publications, thus supporting reproducibility. A problem with traditional paper publication, as identified by Mons [mons] is that of “Knowledge Burying”. The results of an experiment are written up in a paper which is then published. Rather than explicitly including information in structured forms however, techniques such as text mining are then used to extract the knowledge from that paper, resulting in a loss of that knowledge.

In a paper from the Yale Law School Roundtable on Data and Code Sharing in Computational Science, Stodden et al [yale] also discuss the notion of Reproducible Research. Here they identify *verifiability* as a key factor, with the generation of verifiable knowledge being scientific discovery's central goal. They outline a number of guidelines or recommendations to facilitate the generation of reproducible results. These guidelines largely concern openness in the data publication process, for example the use of open licences and non-proprietary standards. Long term goals identified here include the development of version control systems for data; tools for effective download tracking of code and data in order to support citation and attribution; and the development of standardised terminologies and vocabularies for data description. Mechanisms for citation and attribution (including data citation, e.g. Data Cite[2]are key in providing incentives for scientists to publish data.

The Scientific Knowledge Objects [giunchiglia] of the LiquidPub project describe aggregation structures intended to describe scientific papers, books and journals. The approach explicitly considers the lifecycle of publications in terms of three ``states'': Gas, Liquid and Solid, which represent early, tentative and finalised work respectively.

Groth et al [groth] describe the notion of a “Nano-publication” -- an explicit representation of a *statement* that is made in scientific literature. Such statements may be made in multiple locations, for example in different papers, and validation of that statement can only be done given the context. An example given is the statement that *malaria is transmitted by mosquitos*, which will appear in many places in published literature, each occurrence potentially backed by differing evidence. Each nano-publication is associated with a set of

[2] http://datacite.org/

annotations that refer to the statement and provide a minimum set of (community) agreed annotations that identify authorship, provenance, and so on. These annotations can then be used as the basis for review, citation and indeed further annotation. The Nano-publication model described in [groth] considers a statement to be a *triple* -- a tuple of three concepts, subject, predicate and object -- which fits closely with the Resource Description Framework (RDF) data model, used widely for (meta)data publication (see the discussion on Linked Data below). The proposed implementation uses RDF and Named Graphs. Aggregation of nano-publications will be facilitated by the use of common identifiers (following Linked Data principles), and to support this, the Concept Web Alliance[3] are developing a ConceptWiki[4], providing URIs for biomedical concepts. The nano-publication approach is rather "fine-grain", focusing on single statements along with their provenance.

Benefits of explicit representation are clear. An association with a dataset (or service, or result collection, or instrument) should be more than just a citation or reference to that dataset (or service, or result collection). The association should rather be a *link* to that dataset (or service, or result collection, or instrument) which can be followed or dereferenced explicitly, thereby providing access to the actual resource and thus enactment of the service, query or retrieval of data, and so on.

Linked Data

Providing links, rather than associations, between resources will help foster reproducibility. The term Linked Data is used to refer to a set of best practices for publishing and connecting structured data on the Web. Linked Data explicitly encourages the use of dereferenceable links as discussed above, and the Linked Data "principles" -- use of HTTP URIs for naming, providing useful information when dereferencing URIs, and including links to other URIs -- are intended to foster reuse, linkage and consumption of that data.

Through the use of HTTP URIs and Web infrastructure, Linked Data provides a standardised publishing mechanism for structured data, with "follow your nose" navigation allowing exploration and gathering of external resources. For example, [missier] uses a Linked Data approach to publish provenance information about workflow execution. The use of RDF (and thus associated representation machinery such as RDF Schema and OWL) offers the possibility of inference when retrieving and querying information.

What Linked Data does not explicitly provide, however, is a common model for describing the structure of our ROs and additional aspects that are needed in order to support the scholarly process -- factors such as lifecycle, ownership, versioning and attribution. Linked Data thus says little about how that data might be organised, managed or consumed. Linked Data provides a platform for the sharing and publication of data, but simply publishing data as Linked Data will not be sufficient to support and facilitate its reuse.

Jain et al [jain] also question the value of "vanilla" Linked Data in furthering and supporting the Semantic Web vision. Their concerns focus on how one selects appropriate datasets from the "Linked Data Cloud", a concern about the lack of expressivity used in datasets (thus limiting the use to which reasoning can be usefully employed), and the lack of schema mappings between datasets. The nano-publications of Groth et

[3] http://www.nbic.nl/about-nbic/affiliated-organisations/cwa/introduction/

[4] http://conceptwiki.org/

al [groth] are also looking to add additional shared content on top of the Linked Data approach in terms of minimal annotations.

Preservation and Archiving

The Open Archival Information System (OAIS) reference model [oais] describes "open archival information systems" which are concerned with preserving information for the benefit of a community. The OAIS Functional Model describes a core set of mechanisms which include Ingest, Storage and Access along with Planning, Data Management and Administration. There is also separation of *Submission Information Packages*, the mechanism by which content is submitted for ingest by a Producer; *Archival Information Package*, the version stored by the system; and *Dissemination Information Package*, the version delivered to a Consumer.

OAIS considers three external entities or actors that interact with the system. Producers, Management and Consumers, to characterise those who transfer information to the system for preservation; formulate and enforce high level policies (planning, defining scope, providing "guarantees") and are expected to use the information respectively. OAIS also consider a notion of a *Designated Community*, a subset of consumers that are expected to understand the archived information. The consideration of the Designated Community is important as it provides a context within which the preserved objects are to be used/interpreted, and thus impacts on requirements or features.

Aggregation

The idea of aggregation in a web context has already been addressed by the Open Archives Initiation Object Reuse and Exchange Specification (OAI-ORE, or ORE). ORE defines a data model and a number of concrete serialisations (RDF, Atom and RDFa) that allow for the description of aggregations of Web resources. The key concepts in ORE are the notions of *Aggregation*, which represents an aggregation of a number of resources; and *ResourceMap*, which provides a structural model for describing the elements in the aggregation (*AggregatedResources*) and relationships between them.

The ORE model is agnostic as to the semantics of such aggregations -- examples are given which include aggregations of favourite images from Web sites, the aggregation of a number of different resources to make up a publication in a repository, or multi-page HTML documents linked with ``previous'' and ``next'' links.

ORE provides a description of Resource Map Implementations using RDF, which integrates well with current approaches towards the publication of Linked Data [vandesompel].

Content, Container and Vocabularies

In terms of the conceptual models that can support the scientific process, there is much current interest in the representation of Scientific Discourse and the use of Semantic Web techniques to represent discourse structures. Ontologies such as EXPO[5], OBI[6], ISA[7] MGED[8] and SWAN/SIOC[9] provide vocabularies that

[5] http://expo.sourceforge.net/

[6] http://obi-ontology.org/page/Main_Page

[7] http://isatab.sourceforge.net/

allow the description of experiments and the resources that are used within them. The myExperiment ontology[10] borrows terms from a number of well-known ontologies/schemas, and is used to describe myExperiment content, in particular making use of OAI-ORE in descriptions of myExperiment packs.

The HyPER community[11] is focused on infrastructure to support Hypotheses, Evidence and Relationships. The Semantic Publishing and Referencing (SPAR) Ontologies[12] also provide facilities for describing the component parts of documents and the scholarly publishing process. Functional Requirements for Bibliographic Records[13] (FRBR) is a conceptual model of the bibliographic universe outlined in a 1998 report from the International Federation of Library Associations and Institutions (IFLA). The report uses entity-relationship analysis to "provide a clearly defined, structured framework for relating the data that are recorded in bibliographic records to the needs of the users of those records." The most influential parts of the FRBR report are the definitions of user tasks and bibliographic entities..

In the main, however, this work tends to focus on the details of the relationships between the resources that are being described – what might be termed *content* rather than *container*. It is likely, however, that these vocabularies will be of use within the Research Objects developed by Wf4Ever.

---

[8] http://mged.sourceforge.net/ontologies/index.php

[9] http://www.w3.org/TR/hcls-swansioc/

[10] http://rdf.myexperiment.org/ontologies/

[11] http://hyp-er.wik.is/

[12] http://opencitations.wordpress.com/2010/10/14/introducing-the-semantic-publishing-and-referencing-spar-ontologies/

[13] http://www.ifla.org/VII/s13/frbr/frbr.pdf

6. User Roles

A number of different user roles have been suggested in the use cases D5.1 and D6.1, we have here also included the implicit role of *Creator*. These user roles help to characterise the tasks that users are performing, and in turn allow us to identify technical requirements. These user roles have been observed in both the genomics and astronomy use cases, although it is possible that new roles may later be discovered in other fields and use cases.

User roles are not fixed, and users may play multiple roles or change between roles during the course of an investigation. For instance a *Creator* might become a *Reader* to find relevant workflows which she then *compares* to be able to select the component to reuse. She might *contribute* to the existing workflow by informing the original author of suggested changes to make the workflow more general, *review* the reused research object to determine if the method she's embedding is sane, and finally self-*publish* her new research object on her blog. In other scenarios each of the roles may be played by different people or organisations.

Creator

A creator is a scientist conducting an investigation who wishes to collect together resources as a Research Object that can then be reused or repurposed. This may be for personal re-use (the scientist may not yet wish to *publish*).

Contributor

A contributor is a scientist who provides materials/methods/data that may be used within a Research Object, but who is not necessarily creating a complete Research Object.

Collaborator

A collaborator is a scientist who provides materials/methods/data that may be used within a Research Object, but who may not even be aware of the fact that she is actually contributing content to a Research Object. Collaborators may be naïve in terms of their understanding or experience of workflows or the Research Objects approach.

Reader

The scientist is looking for related works, state of the art, in her field of research. She skims the titles and abstracts of publications or existing materials, sometimes delving into the content of the Research Object and may be interested in re-use or comparison. Users are likely to begin as *readers* and will gain new roles as they become more familiar with RO research techniques, evolving to play the role of *comparators*, *re-users*, *publishers*, and *evaluators*.

Comparator

A comparator will be looking for an RO which is similar to those with which she is currently working. She will want to know if the work has been already published as a RO, and if there are ROs that execute similar tasks to those present and needed in her own RO. She may be more interested in the workflows or methods

contained in the RO and less in metadata, data, authorship and publications related. A comparator may come from outside of the domain for which the original RO has been developed – for example workflows for statistical tasks in biology may also prove useful for an astronomer. Once a suitable RO has been found, the comparator may than take on the role of re-user. A goal for comparison may not necessarily be to use the workflows/methods as a basis for work, but rather to know or understand what “the competition” is doing.

Re-User

A re-user is a scientist who knows the underlying methods encapsulated in an RO (for example a Taverna workflow), and how to extract and replace modules from such methods/workflows and insert them into her own. A re-user will often have played the role of comparator in order to obtain the RO, at other times a colleague may have played the role of comparator and the re-user simply uses the RO identified for her. Like the comparator, a re-user will be primarily interested in the workflow/methods and less in “DC style” metadata.

Publisher

A user who wants to publish an enhanced publication "beyond the pdf", a Research Object, disseminating results or methods to the community. Note that here publishing can be taken to encompass a wide range of activities, not just traditional publishing routes. Thus publishing could also include the embedding of an RO in a blog post, the upload of an RO to a service such as myExperiment, or upload to an institutional repository. Publication could be undertaken by the main “author” of the publication, or could be done by another party acting on behalf of the author, such as a traditional journal.

Evaluator/Reviewer

An evaluator or reviewer takes a published RO and evaluates or reviews the content. This could involve a validation or confirmation of the results presented (thus potentially requiring execution of the methods encapsulated in the RO). Alternatively, review could involve an evaluation of the underling methods of the RO, along with suggestions for improvement in scientific or technical terms.

7. User requirements

These requirements have been distilled from the use cases as summarized in section 4 using the methodology described in section 3.

The user requirements are here organized according to the *user roles* as described in section 6. It is important to recognize that a research object user would often be floating between roles. For instance a *Re-user* might take on the role of *Reader* in order to find a workflow, before resuming as a *Re-user*. A *Creator* may become a *Publisher* (for instance posting on a blog), or the *Publisher* might be a separate entity (say a journal). In this overview, requirements which can be applicable for several user roles are written under what is thought to be their primary user role, which the other roles assume if needed.

As a *Creator of Research Objects* ...		
	I want to...	**so that ...**
UR1.1	create workflows	I can automate and streamline aspects of my investigation
UR1.2	collect data	I can conduct an investigation
UR1.3	aggregate existing resources	I can conveniently access related resources from a single place
UR1.4		I can be sure that I have a matching collection of resources
UR1.5	reference data stored elsewhere	I can aggregate data that is larger/more complex/restricted
UR1.6	describe the relationships between aggregated resources	other researchers can see how the resources fit together
UR1.7	describe the relationships between aggregated resources	I can facilitate the automation of processing of aggregated resources
UR1.8	be recognised as the creator of an RO	I get credit
UR1.9	assign a persistent URL to an RO	I can include the link in my book
UR1.10	record which web services were used by workflow	I can track web service changes
UR1.11		I can give citations to external resources used
UR1.12	embed other's publications	I can later find related reference material/citations
UR1.13		I can get information when designing my experiment

UR1.14	record notes while designing workflow	I can later pick up my thoughts around a part of workflow
UR1.15		I can disseminate reasoning behind my design decisions
UR1.16	annotate experimental results using semantic models	I can find/show links to other, relevant research objects
As a *Contributor to Research Objects* ...		
	I want to...	**so that ...**
UR2.1	provide a workflow	it can be incorporated or used in an investigation
UR2.2		other researchers can review the processing performed
UR2.3		other researchers can repeat the processing performed
UR2.4	provide new or updated data/results	investigations are up to date
UR2.5	modify contents	I can fix a known error with a workflow or investigation
UR2.6	be credited for my contributions	I get credit
UR2.7	have access to RO being created by another researcher	I can contribute to shaping the RO before it's public
As a Collaborator of Research Objects …		
	I want to…	**so that …**
UR3.1	provide content	it can be incorporated or used in an investigation
UR3.2		other researchers can review the processing performed
UR3.3		other researchers can repeat the processing performed
As a *Reader of Research Objects* ...		
	I want to...	**so that ...**
UR4.1	find relevant materials	I can understand the field
UR4.2	browse an overview	I can determine whether there is something useful

		for me
UR4.3	survey the field	check whether something has been done before
UR4.4	examine the relationships between resources	I can understand the relationships between resources
UR4.5	access data	I can look at it and use it for my own purposes
UR4.6	access metadata	I can see where data/methods came from
UR4.7	follow the steps taken	I can understand the investigative process or method
UR4.8	find workflow by purpose	I can investigate different approaches to the same problem
As a *Reviewer/Evaluator of Research Objects* ...		
	I want to...	**so that ...**
UR5.1	rerun an investigation	I can validate that the results are as given
UR5.2	examine the relationships between resources	I can validate those relationships
UR5.3	access data	I can validate the data used
UR5.4	check if external data has changed	I can determine if results are still valid
UR5.5	follow the steps taken	I can validate the investigative process and identify any problems
UR5.6	examine the resources	I can determine the source of those resources
UR5.7	rate content	I can recommend materials to colleagues
As a *Comparator of Research Objects* ...		
	I want to ...	**so that ...**
UR6.1	compare an RO with others	I can determine whether the investigation is novel
UR6.2		I can understand the differences between investigations
UR6.3		I can consider reusing it in the future
As a *Re-User of Research Objects* ...		
	I want to ...	**so that ...**
UR7.1	build a new workflow based on an existing one	I can do something new with less effort

UR7.2	build a new workflow based on an existing one	I can use an existing, known, validated methodology
UR7.3	build a workflow using components/parts of another workflow	I don't have to investigate how to use a service
UR7.4	run an existing workflow with new data	I can get new results by using existing procedures
UR7.5	rerun parts of a workflow	I can avoid re-running long-processing parts of workflow when only some of the data has changed
UR7.6	use results from an existing investigation as input to a new one	I can build on existing results
UR7.7	use data from an existing investigation as input to a new one	I can build on existing data
UR7.8	see versions of a workflow	I can use the latest working version
UR7.9		I can better understand a workflow by understanding how it has evolved
UR7.10		I can see how the latest version of a workflow differs from an earlier version I may have used
UR7.11	extract content	I can reuse that content for other investigations
As a *Publisher of Research Objects* ...		
	I want to...	**so that ...**
UR8.1	publish an RO	it is available for others to see or use
UR8.2	provide references to ROs	they can be cited (leading to credit)
UR8.3	be able to advertise an RO	It reaches its target audience
UR8.4	restrict access to parts of RO	publication complies with license restrictions
UR8.5		data owners are happy

8. Dimensions

Research Objects are intended to support the sharing, publishing and preservation of research results, methods and workflows. This preservation can then support the reuse (where reuse can be interpreted in a number of ways) or validation of the information or content of the ROs. We identified the following dimensions based on the extracted user requirements in section 5. Example requirements are here shown in italics.

- **Repeat** - include sufficient information for others to rerun the investigation at a later date. *As a Re-user I want to run an existing workflow with new data.*
- **Reproduce** - include sufficient information for an independent investigator to reproduce the results, e.g. obtain the same results. *As a Reviewer I want to rerun an investigation to validate the result.*
- **Replay** - provide a a comprehensive record of what has happened, without necessarily including the means to perform the investigation again. *As a Reviewer I want to follow the steps taken.*
- **Live/Refreshable/Notifiable** - provide dynamic links to content, updating with ease when something changes. *As a Re-user I want to see versions of a workflow to use the latest working version.*
- **Component Reuse** - deconstructing an investigation in order to reuse components or pieces. Reassembly of deconstructed aggregations. *As a Re-user I want to build a new workflow based on an existing one*
- **Reliability** - verification and validation of the components, along with measures of trust in the data, results and methods. *As a Reviewer I want to follow the steps taken to validate the investigative process and identify any problems*
- **Justification** - why and how particular decisions were made. *As a Creator I want to describe the relationships between aggregated resources*
- **Resilience** - coping with change/loss/errors. *As a Re-user I want to run an existing workflow with new data.*
- **Cross Boundary** - Objects reused across different research communities. *As a Reader I want to find relevant materials so that I can understand the field*
- **Discovery** - The ability to find/discover/retrieve ROs. Mechanisms for exposure/publication of ROs. *As a Reader I want to survey the field to check if something has been done before*
- **Reference** - means of identifying ROs. *As a Publisher I want to provide references to ROs so that they can be cited*
- **History** - Providing roll-back to retrace steps, fix errors, diagnose errors. *As a Re-user I want to see versions of a workflow to understand how it has evolved.*

9. Research Object Lifecycle

Research Objects are intended to support the process of scientific investigation. Within the use cases we have identified a number of abstract states that ROs can transition into and out of as part of their evolution in time. Each of these particular states can be characterized in terms of the properties that we expect the objects in that state to exhibit; the potential transitions to other states that are possible; and roles that users may have when interacting with objects in those states.

The basic lifecycle states identified include *Live Objects*, *Publication Objects* and *Archived Objects*. These are an initial identification of states based on our current experience – we expect that this characterization may be extended further during the course of the project.

Live Objects (LO) represent a work in progress. They are thus *mutable* as the content or state of their resources may change, leading to the need for version management. Live objects are potentially under the control of multiple owners and may fall under mixed stewardship, raising issues of security and access control.

Publication Objects (PO) are intended as a record of past activity, ready to be disseminated as a whole. This is in line with our key motivation for Research Objects, namely to support "rich publication" by moving from traditional paper based (linear) dissemination mechanisms, to aggregations of related and interlinked pieces of information. POs are *immutable*, and their multiple successive versions are considered as distinct objects. They must be citeable, and credit and attribution are central aspects of the publication process as they are key to providing rewards, and thus incentives, for scientific publication. As an example, myExperiment packs can be viewed as an embryonic form of Publication Objects, where Workflow specifications are collected along with results obtained or papers along with presentational materials. POs may also make use of ontologies for the representation of the rhetorical or argumentation structure in the publication. Proposals such as DataCite[14] will have a role to play here, and we expect that POs will be citeable via mechanisms such as DOIs.

Archived Objects (AO) encapsulate aggregations that represent a point of a Research Object's life where it has either been deprecated, or has reached a version that the author prescribes to be stable and meaningful and is appropriate for publication or long term preservation. AOs are therefore immutable, with no further changes or versions allowed. For example, an AO may represent a historical record for resources used in an experiment which has concluded, or has been abandoned.

One key differentiation between an archived and published object is that an archived object comes with some expectation or guarantee as to the preservation of the object. Publication objects are citeable and can be referenced, but do not necessarily come with guarantees as to the *preservation* of their contents/state. For example, an RO could be published through an embedding in a blog site. Archived Objects in contrast, have an expectation for the preservation of their content.

[14] http://www.datacite.org/

With this simple state classification, we can describe the lifetime of a Research Object in terms of its evolution from LO, to either PO or AO (the “terminal states”), while at the same time multiple versions of a LO may be created, each evolving independently into POs or AOs. *Figure 1* below shows states along with possible transitions between them. We also identify where particular user roles (Creator, Contributor, Publisher, Evaluator/Reviewer, Reader) may interact with objects in particular states.

Different user roles interact with ROs in different lifecycle states. Creators and Contributors will use objects in a Live state, as they develop and construct ROs. Publishers move Live objects to the Published state, where they are then available for Evaluators, Reviewers, Comparators and Readers. A Re-user will transition an RO from a Published state to Live.

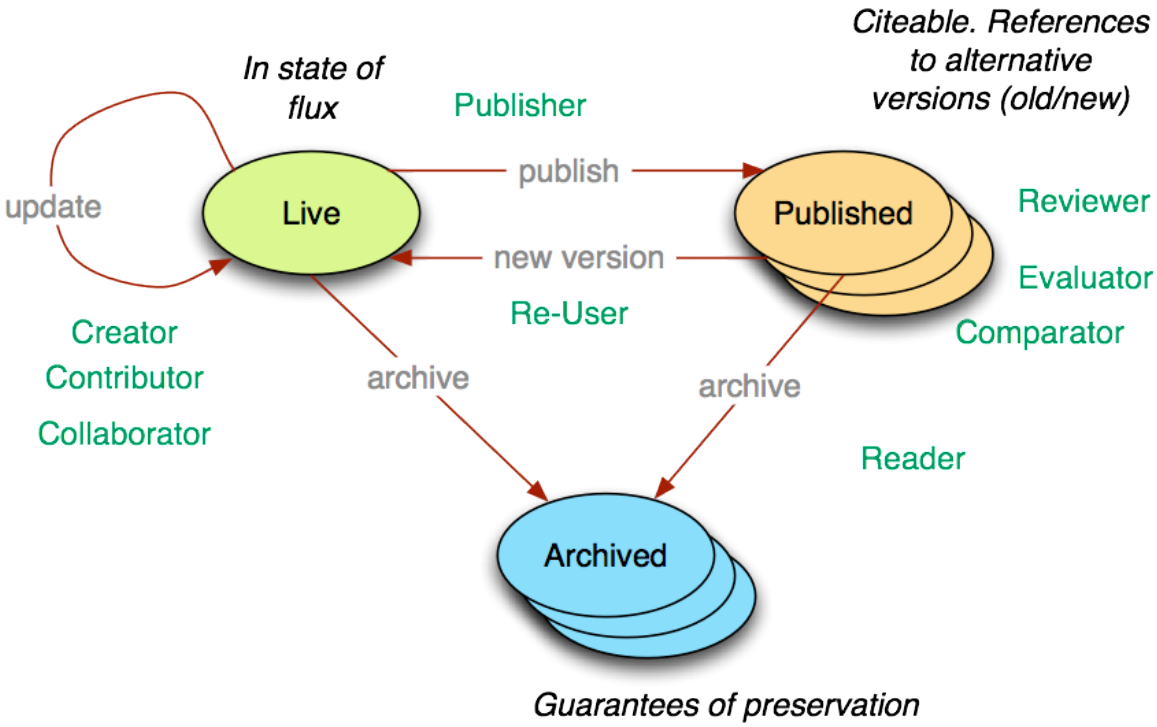

Figure 1 Lifecycle states, transitions and associated user roles.

10. Technical requirements

Based on elicited user requirements, we have extracted these technical requirements that should be supported to enable research objects managements throughout the life cycle presented earlier. It should be noted that some technical requirements are duplicated but different user requirement contexts, and thus may have a subtly different effect on the technical architecture and solution.

***Creator of Research Objects* ...**		
	User Requirement	**Technical Requirement**
TR1.1a	to create workflows	System for workflow creation.
TR1.1b		Workflows as content items in ROs
TR1.2a	to collect data	Embedding data as content items in ROs.
TR1.3a	to aggregate existing resources to access them from a single place	Linking to existing resources within ROs.
TR1.3b		Embedding existing resources as content items in ROs.
TR1.4a	to aggregate existing resources to be sure I have a matching collection	Metadata about aggregated resources
TR1.5a	to reference data stored elsewhere	URIs/linking to public and private data
TR1.5b		Metadata on accessing private data (e.g.: server access, authentication).
TR1.6a	to describe the relationships between aggregated resources to see how resources fit together	Aggregation structures with annotated relationships.
TR1.6b		Vocabularies for representing relationships.
TR 1.7a	to describe the relationships between aggregated resources to facilitate the automation of processing	Rich annotated links (e.g.: roles, timestamps) between resources.
TR1.8a	to be recognised as the creator of an RO	Mechanism for author identification.
TR1.8b		Creator/author annotations.
TR1.9a	assign a persistent URL to an RO	Support persistent URL services.
TR1.9b		Provide immutable URLs which will work throughout

		evolutions of the system.
TR1.10a	record which web services were used by workflow to track changes	Harvest/pull metadata about web services.
		Mechanisms for generating checksums, and recording timestamps and versions of web services.
TR1.11a	record which web services were used to give citations	Uniquely identify web services.
TR1.11b		Resolve publications/ROs for used web services.
TR1.12a	embed publications to find related reference material/citations	Resolve publications/ROs for embedded papers.
TR1.13a	embed publications to get information when designing an experiment	Embed PDFs/other ROs.
TR1.14a	record notes for later reading	Workflow tool to provide easy way to annotate parts of workflow.
TR1.15a	record notes to show reasoning	Mechanisms for publication “clean-up”.
TR1.16a	annotate experimental results using semantic models	Support for vocabularies and semantic models.
TR1.16b		Mechanisms for rich annotation of items in an RO.
Contributor to Research Objects ...		
	User Requirement	**Technical Requirement**
TR2.1a	to provide a workflow, for reuse	Mechanism for publishing workflow-centric ROs.
TR2.1b		Mechanisms for relating workflows/ROs with others.
TR2.2a	to provide a workflow, for reviewing	Rich annotation for describing workflows/ROs (e.g.: purpose, history).
TR2.3a	to provide a workflow, for others to repeat the processing performed	Capture input data to allow repeat runs.
TR2.3b		Rich annotation for describing workflows/ROs (e.g.: usage, caveats, conditions).
TR2.4a	to provide new or updated data/results	Mechanisms for adding and updating resources in an RO.
TR2.4b		Mechanisms for capturing history of changes to resources in ROs.

TR2.5a	to modify contents	Mechanisms for updating contents of an RO.
TR2.5b		Mechanisms for capturing history of changes to content in ROs.
TR2.6a	to be credited for my contributions	Mechanism for author identification.
TR2.6b		Creator/author annotations for content in ROs.
TR2.7a	to contribute to ROs being created by other researchers/collaborators	Mechanisms for accessing and working with ROs in a “live” state.
TR2.7b		Mechanisms for capturing history of changes to content in ROs.
TR2.7c		Mechanisms for dealing with conflicts in changes made to content within ROs and the ROs themselves.
Reader of Research Objects ...		
	User Requirement	**Technical Requirement**
TR4.1a	to find relevant materials	Searching/browsing of ROs.
TR4.1b		Tagging and other rich annotation capabilities.
TR4.1c		Mechanisms for defining and finding relations between ROs.
TR4.2a	to browse an overview	Interfaces for the presentation of overviews.
TR4.3a	to survey the field	Support for grouping of ROs by field/tag/content, e.g.: "Most popular in field X"
TR4.4a	to examine the relationships between resources	Interfaces for browsing and following links between resources.
TR4.4b		Description of relationships between resources.
TR4.5a	to access data	Mechanisms for storing data or linking to data.
TR4.5b		Ability to download data.
TR4.5c		Ability to request access to data (or forward requests on).

TR4.6a	to access metadata	Mechanisms for storage/retrieval of structured metadata.
TR4.6b		Rich representation of metadata.
TR4.7a	to follow the steps taken	Support for describing the overview of workflow/method.
TR4.7b		Replay workflow execution.
TR4.8a	to find workflows by their purpose	Support classification of workflows based on purpose/domain/problem.
TR4.8b		Interfaces for finding similar workflows based on purpose.
Reviewer/Evaluator of Research Objects ...		
	User Requirement	**Technical Requirement**
TR5.1a	Rerun an investigation to validate	System for workflow creation.
TR5.1b		Capture provenance trace of original execution.
TR5.1c		Capture original inputs and required tools.
TR5.1d		Access to original services or copies of original return values.
TR5.1e		Interfaces for running / “playing” ROs.
TR5.2a	to examine the relationships between resources	Browse annotations on how relationship was made (for instance "dataX producedBy runY of workflowZ using inputA").
TR5.3a	to access data, to validate	Verify data equality/similarity via checksums, timestamps, actual content, etc.
TR5.4a	to check if external data has changed, to determine if results are still valid	Mechanisms for checking updates to external data.
TR5.4b		Notifications for changes detected.
TR5.5a	to follow the steps taken	Support for browsing annotations on steps to validate scientific reasoning.
TR5.5b		Support for comparing steps with known methodologies (abstract workflows)

TR5.6a	to examine the resources and their source	Support for browsing annotations on origin/source of data.
TR5.6b		Mechanisms for following links to verify data equality/similarity/validity.
TR5.7a	to rate and recommend content	Mechanisms for rating and recommending content.
TR5.8b		Capturing rating changes over time as RO evolves.
Comparator of Research Objects ...		
	User Requirement	**Technical Requirement**
TR6.1a	to compare an RO with others, to determine novelty	Mechanisms for finding similar ROs (e.g.: with similar/same data/services/workflows).
TR6.1b		Mechanisms for finding ROs based on scientific field, keywords/tags, methodology.
TR6.1c		Mechanisms for comparing workflow/methodology structure, in particular the abstract workflow.
TR6.2a	to compare an RO with others, to understand differences	Mechanisms for comparing workflow structures.
TR6.2b		Mechanisms for comparing individual data items between ROs.
TR6.2c		Mechanisms for browsing scientific reasoning of relationships between resources.
TR6.2d		Mechanisms for comparing hypothesis of each RO.
Re-User of Research Objects ...		
	User Requirement	**Technical Requirement**
TR7.1a	to build a new workflow based on an existing one to save time	Support for embedding/linking to existing workflows in ROs.
TR7.1b		Support for customising existing workflows, keeping links to original.
TR7.2a	to build a new workflow based on an existing one to use existing methodology	Ability to gather citations to existing RO.
TR7.2b		Support for getting the “latest” version or past versions of workflows/ROs.

TR7.2c		Support for discovering and referencing other people's extensions/uses of chosen workflow.
TR7.3a	to reuse components/parts of other workflows	Support for describing the individual components/parts of workflows.
TR7.3b		Mechanisms for finding components/parts of workflows.
TR7.3c		Identification mechanisms for individual components/parts of workflows.
TR7.4a	to run an existing workflow with new data	Ability to verify that workflow can still run (e.g.: by checking if underlying web services used are still accessible).
TR7.5a	to rerun parts of a workflow	Ability to verify that parts of workflow can still run (e.g.: by checking if underlying web services used are still accessible).
TR7.6a	to use results from an existing investigation as input to a new one, so I can build on existing results	Support for linking/referencing of resources from one RO to another, including retrieval of data.
TR7.6b		Support for linking to newer versions of data from later runs.
TR7.7a	to use data from an existing investigation as input to a new one, so I can build on existing data	Support for linking/referencing of resources, but also following its links back to origin/source
TR7.8a	to see versions of a workflow, so I can use the latest working version	Support for the notion of a 'version' (possibly non-linear).
TR7.8b		Support for forward-links to versions in other repositories/by other users.
TR7.9a	to see versions of workflow, to see how it evolved	Capture history of workflow evolution
TR7.10a	to see versions of workflow, see differences between versions	Support for comparing different versions of workflows
TR7.11a	to extract content, for reuse in other investigations	Support for extracting content in a whole and atomic fashion.

TR7.11b		Capture enough information (such as the source and usage) about content, to enable easy reuse.
Publisher of Research Objects ...		
	User Requirement	**Technical Requirement**
TR8.1a	to publish an RO to make it available	Mechanisms to ensure components of RO are accessible.
TR8.1b		Support for snapshots of RO and its resources.
TR8.1c		Support for including ROs in publication sources.
TR8.2a	to provide references to ROs	Mechanisms for identification of published ROs.
TR8.2b		Mechanisms for citation.
TR8.3a	to be able to advertise an RO	Environment for depositing ROs that has a public presence.
TR8.3b		Mechanisms for sharing and promoting ROs.
TR8.4a	to restrict access to parts of an RO, to comply with license restrictions	Mechanisms for “selective hiding” of content in an RO.
TR8.5a	to restrict access to parts of an RO, to keep data owner happy	Mechanisms for “selective hiding” of content in an RO.

Based on the above technical requirements, we further distil in what follows representational and functional requirements that should be supported to enable effective and efficient managements of research objects.

Research Object Identity: This is perhaps the most basic and important technical requirement that user requirements hinted to. There is a need for a mechanism that allows to uniquely refer to a research object. Rather than trying to invent yet another identification system, we will review and choose among existing identification scheme the one that is best suited for research object identification. Examples of existing identification schemes include URI[15], DOI[16], PURL[17].

Representation: Content-wise, research objects can be seen as bundles of heterogeneous structured and unstructured data sets, e.g., XML documents, images, relational data sources, and methods, in particular workflows, which can be enacted to produce new data sets (results). The representation of research objects

[15] http://www.w3.org/2001/12/URI/

[16] http://www.doi.org/

[17] http://purl.oclc.org/

should cater for the description of those elements as well as for the fact that research objects can be connected to each other using typed associations, e.g., associations used to specify the different versions or states of a given research object.

Versioning: As mentioned earlier, a contributor may wish to add new elements or modify existing ones within a research object. Such operations may yield the creation of a new version of the research object in question. It follows then that there is a need for a mechanism for versioning research objects to support the creation of a research object, maintain information about the different versions of a research object as well as ensuring their integrity. There is a plethora of version management systems, e.g., Subversion[18], Git[19], Mercurial[20]. Such systems are, however, mainly used for software versioning. This raises the question as to whether such systems can be used and/or adapted to fit versioning requirements in the context of research objects. We intend to investigate this question in the following stages of the project.

Distribution: A research object may be composed of multiple elements (data and methods) that are physically distributed, which can be edited and updated independently. We therefore need a mechanism for synchronizing research objects in distributed settings, including checksums, versioning and conflict management. Existing distributed version control systems (DVCS) for source code like Git and Mercurial provide a technical solution to many of these challenges, which might be leveraged for managing distributed research objects.

Research objects themselves might reside in multiple locations, like on a USB stick, on a blog, in a journal and in several RO repositories. This distributed nature raises additional challenges with RO identity and resolvability.

Editing Research Objects: As mentioned in the previous section, users (e.g., creator and contributor) need to be able to edit a research object by aggregating data and methods together. Such users are not necessarily information technology experts. For example, the user requirements reported in this deliverable were elicited by scientists from the life sciences and astronomy. To support such users in editing research objects, there is a need for tools, e.g., a workbench, that allows them to fetch and aggregate existing data, to design methods, e.g. workflows, to enact those methods, to store the results obtained as well as any metadata that the creator may wish to add with the purpose of facilitating research objects discovery and reuse.

Provenance Management: provenance plays a key role in understanding the dependencies between the elements that constitute a research object and the dependencies between the elements of different research objects. For instance, as highlighted in the previous section, to assess the outcome claimed within a research object, evaluators may need to trace back the data that contributed to that outcome, e.g., the evaluator may want to know the input used to produce a given workflow result. Provenance is a key ingredient to other activities, e.g., to understand, compare and debug research objects. Therefore, there is a

[18] http://subversion.tigris.org/

[19] http://git-scm.com/

[20] http://mercurial.selenic.com/

need for collecting provenance of the elements that compose research objects and the traces of methods (e.g., workflows) executions. As well as logging provenance information, support for browsing and querying provenance is required to facilitate the tasks users have at hand.

Browsing and Querying Research Objects: The mechanism through which users can access research objects and express their requirements, as to which research objects are of interest, is of upmost importance in the context of the Wf4ever project. Users must be able to browse research objects using imprecise queries, e.g., keyword queries, as well as precise queries that specify the properties of the research objects to be retrieved, e.g., predicated queries. For example, a reader who is interested in gaining knowledge of specific domain, say Astronomy, will be interested in browsing research objects. In doing so, the user may want to examine the components of a research object exploiting intra-references that aggregates those components. The reader may also explore other research objects by exploiting associations that connects research objects, e.g., to consult previous versions of a research object or to examine the research objects that make use of the research object s/he is examining. On the other hand, a comparator, may be interested in locating research objects with specific properties. An example of a query the comparator may issue is "give me the research objects that use the same data inputs as the research object identified by *ro1*".

Indexing Research Objects: Users may have to query a large population of research objects. For example, to identify the research objects that are similar to a given one, the comparator may need to query all known research objects. Accessing and querying a large population of research objects is likely to give rise to efficiency issues.

Indexing support is a mechanism that can be used to overcome the efficiency issue. As underlined by user requirements, research objects are rich structures that bundles elements of different types and references other research objects. Therefore, the indexing support used to facilitate access to such structures should cater for the richness (and therefore the heterogeneity) of research object in terms of contents.

11. References

[olson] G. Olson, A. Zimmerman, N. Bos, Scientific Collaboration on the Internet, MIT Press, 2008.
[bechhofer] Bechhofer, S., De Roure, D., Gamble, M., Goble, C. and Buchan, I. (2010) Research Objects: Towards Exchange and Reuse of Digital Knowledge. In: The Future of the Web for Collaborative Science (FWCS 2010), April 2010, Raleigh, NC, USA.
[bechhofer2] Bechhofer, S., Ainsworth, J., Bhagat, J., Buchan, I., Couch, P., Cruickshank, D., Delderfield, M., Dunlop, I., Gamble, M., Goble, C., Michaelides, D., Missier, P., Owen, S., Newman, D., De Roure, D. and Sufi, S. (2010) Why Linked Data is Not Enough for Scientists. In: Sixth IEEE e--Science conference (e-Science 2010), December 2010, Brisbane, Australia.
[cohn] Mike Cohn (2008-04-26): *Advantages of the "As a User, I want" user story template* in *Mike Cohn's blog Succeeding with Agile*. http://blog.mountaingoatsoftware.com/advantages-of-the-as-a-user-i-want-user-story-template
[mesirov] J. P. Mesirov, Accessible reproducible research, Science 327 (5964) (2010) 415 – 416.
[deroure] D. De Roure, C. Goble, Anchors in Shifting Sand: the Primacy of Method in the Web of Data, in: Web Science Conference 2010, Raleigh NC, 2010.
[mons] B. Mons, Which gene did you mean?, BMC Bioinformatics 6 (2005) 142.
[yale] Yale Roundtable Participants, Reproducible Research: Addressing the Need for Data and Code Sharing in Computational Science, Computing in Science and Engineering 12 (5) (2010) 8–13. doi:10.1109/MCSE.2010.113.
[giunchiglia] F. Giunchiglia, R. ChenuAbente, Scientific Knowledge Objecst V.1, Tech- nical Report DISI-09-006, University of Trento (January 2009).
[groth] P. Groth, A. Gibson, J. Velterop, The Anatomy of a Nano-publication, Information Services and Use 30 (1) (2010) 51–56. URL http://iospress.metapress.com/index/FTKH21Q50T521WM2.pdf
[klyne] G. Klyne, J. J. Carroll, Resource Description Framework (RDF): Concepts and Abstract Syntax, W3C Recommendation, World Wide Web Consor- tium, http://www.w3.org/TR/owl-guide/ (2004). URL http://www.w3.org/TR/owl-guide/
[oais] Consultative Committee for Space Data Systems, Reference Model for an Open Archival Information System (OAIS), Blue Book CCDS 650.0-B-1, Open Archives Initiative (2002).
[clark] T. Clark, J. S. Luciano, M. S. Marshall, E. Prud'hommeaux, S. Stephens (Eds.), Semantic Web Applications in Scientific Discourse 2009, Vol. 523, CUER Workshop Proceedings, 2009.
[soldatova] L. N. Soldatova, R. D. King, An ontology of scientific experiments., J. of the Royal Society, Interface / the Royal Society 3 (11) (2006) 795–803.
[courtot] M. Courtot, W. Bug, F. Gibson, A. L. Lister, J. Malone, D. Schober, R. Brinkman, A. Ruttenberg, The OWL of Biomedical Investigations, in: OWLED 2008, 2008.
[whetzel] P. L. Whetzel, H. Parkinson, H. C. Causton, L. Fan, J. Fostel, G. Fragoso, L. Game, M. Heiskanen, N. Morrison, P. Rocca-Serra, S.-A. Sansone, C. Taylor, J. White, C. J. Stoeckert, The MGED Ontology: a resource for semantics-based description of microarray experiments, Bioinformatics 22 (7) (2006) 866–873.
[vandesompel] H. V. de Sompel, C. Lagoze, M. Nelson, S. Warner, R. Sanderson, P. John- ston, Adding eScience Assets to the Data Web, in: C. Bizer, T. Heath, T. Berners-Lee, K. Idehen (Eds.), Linked Data on the Web (LDOW2009), 2009.
[missier] P. Missier, S. S. Sahoo, J. Zhao, A. Sheth, C. Goble, Janus: from Workflows to Semantic Provenance and Linked Open Data, in: Procs IPAW 2010, 2010.
[jain] P. Jain, P. Hitzler, P. Yeh, K. Verma, A. Sheth, Linked Data is Merely More Data, in: Linked AI: AAAI Spring Symposium "Linked Data Meets Artificial Intelligence", 2010.