Scientific Information Management in Collaborative Experimentation Environments

Kaletas, E.C.

Publication date
2004

Appendix D

Glossary

Activity. Special type of experiment components (Chapter 4). Activities may represent laboratory activities, instrumentations, or computational processes. Examples of activities are, respectively, hybridization, array scanning, and array image analysis. See also *experiment components, physical entity, data element.*

Administrator. User type in a VL (Chapter 2). Administrators are responsible for the tasks related to the proper management and operation of the VL, such as resource management, infrastructure maintenance, and user management.

Collaboration. A set of provided functionality related evaluation criteria for related work (Chapter 3). Collaboration functionality includes sharing of resources (hardware, software, data/information resources), supporting cooperative working environments (e.g. video-conferencing, chatboxes, whiteboards), and organizing, managing and controlling collaboration activities (functionality provided by virtual organizations). See also *cooperative working environments.*

Collaborative Experimentation Environment (CEE). A type of support environment for scientific experimentations (Chapter 1). CEE refers to virtual laboratory in its broadest sense, with an emphasis on supporting joint multi-disciplinary projects and collaboration, specifically information sharing among organizations and scientists. It is an integrated solution and support environment that addresses every aspect of experimentation and that supports scientists during the entire life cycle of experiments.

Computation-oriented systems. A class of related work (Chapter 3). Related systems in this group approach experimentation from a computational point of view, and consider experiments as *jobs*. Some of these systems allow scientists to compose jobs by connecting a number of software components to each other and execute the jobs, while others offer a pre-defined set of software components that perform a specific task in a domain. See also *context-oriented systems, instrumentation-oriented systems, metadata-oriented systems.*

Computational process. Special type of activity (Chapter 4). A computational process represents a computational activity in an experiment. There is a software entity corresponding to each computational process (e.g. ScanAlyze for array image analysis). See also *experiment components, activity, software entity.*

Computational processing. One of the three components of the proposed experiment model (Chapter 4). The computational processes involved in a scientific experiment are collectively called as the experiment's computational processing. The computational processing of an experiment is composed of a number of software entities, connected to each other forming a data flow graph. Software entities composing the experiment's computational processing can be executed on one or more computers. Data model for the computational processing of an experiment covers both static information (i.e. descriptions of the software entities) and dynamic information (i.e. run-time information about the processes). See also *experiment procedure*, *experiment context*, *software entity.*

Context-oriented systems. A class of related work (Chapter 3). The main focus of context-oriented systems in experimentation is the experiment design; in other words, the meaning of the experiment. These systems support the scientists during experiment modelling; that is, when defining the goal of the experiment, formulating the goal as an experimental procedure, defining experiment conditions and variables, and selecting the techniques, tools to use. See also *computation-oriented systems*, *instrumentation-oriented systems*, *metadata-oriented systems.*

Context element. Part of an experiment context (Chapter 4). An experiment context consists of context elements. A context element is an instantiation of its corresponding procedure element. Therefore, at the experiment level, a context element provides the description of the experiment component represented by its corresponding procedure element. At the data level, every context element has a corresponding object in the application database. See also *experiment context*, *procedure element*, *experiment components.*

Cooperative working environment. A provided functionality related evaluation criterion for related work (Chapter 3). Cooperative working environments provide special type of collaboration functionality for real-time collaboration activities (e.g. joint sessions, video-conferencing, chatboxes, whiteboards, virtual reality environments, etc.). See also *collaboration.*

Data element. Special type of experiment components (Chapter 4). Data elements correspond to both raw data generated by instruments and used as input to computational processes (e.g. array image), and to processed data and/or information generated by computational processes (e.g. array measurement). See also *experiment components*, *physical entity*, *activity.*

Domain expert. User type in a VL (Chapter 2). Domain expert is a scientist who has extensive knowledge and experience on a given e-science domain and on the experiments being performed in that domain. See also *Scientist.*

Enablers. Related work (Chapter 3). Existing or emerging technologies, standards, and paradigms that enable the development of supporting ICT infrastructures (e.g. information models and standards, distributed information management systems, etc.). Enablers provide certain functionality, mechanisms, facilities, models, and methodologies that can be applied by ICT infrastructure developers to address the requirements for the support infrastructure.

Experiment components. Part of the general structure of scientific experiments (Chapter 4). A scientific experiment consists of a number of experiment components. There are three kinds of experiment components, namely physical entities, activities, and data elements. See also *physical entity*, *activity*, *data element.*

Experiment context. One of the three components of the proposed experiment model (Chapter 4). An experiment context is an instantiation of an experiment procedure. It describes the accomplishment of a particular experiment by providing descriptions of each component involved in the experiment, in other words, by providing the context for each experiment component. See also *experiment procedure*, *computational processing*, *experiment components*.

Experiment procedure. One of the three components of the proposed experiment model (Chapter 4). An experiment procedure defines a particular way of accomplishing an experiment of a certain type. It defines how to make a certain type of experiment to solve a specific scientific problem by providing a complete, step-by-step definition of the experiment. The complete definition of an experiment includes, among others, the components involved in the experiment, their order, and the control variables and parameters as well as the protocols to be applied during the experiment. See also *experiment context*, *computational processing*, *experiment components*.

Generic descriptive elements. Special type of data elements (Chapter 4). Generic descriptive elements represent the components that are common to all experiments and provide descriptive information about them. Such descriptive information does not change from one experiment to another. Examples of generic descriptive elements are clone, gene, organism, scanner, and image analysis program. See also *experiment components*, *data element*.

ICT developer. Developer of the base VL infrastructure (Chapter 2). ICT developers develop different functionalities provided by the base VL infrastructure, e.g. storage of data or monitoring the experiment execution. ICT developers are not users of the VL.

Instrumentation-oriented systems. A class of related work (Chapter 3). Related systems in this group mainly address issues related to instrumentation. These systems provide the necessary functionality for off-line or on-line operation of a real instrument, or for virtual operation of a simulated instrument. In case of off-line operation, the instrument is operated by a technician/operator, and mechanisms are provided for communication between the scientist and the operator, for instance to set the parameter values. See also *computation-oriented systems*, *context-oriented systems*, *metadata-oriented systems*.

Metadata-oriented systems. A class of related work (Chapter 3). Metadata-oriented systems mainly focus on the management of descriptions of the data sets generated by experiments. These systems provide a metadata catalogue containing, for instance, information about the owner, information about the physical location of the data sets, or in some cases information about the contents of the data sets. The commonly provided functionality is querying the metadata catalogue. See also *computation-oriented systems*, *context-oriented systems*, *instrumentation-oriented systems*.

Module. A self-contained executable program in VLAM-G (Chapter 5). Module is the term used in the VLAM-G for 'software entity'. See also *software entity*.

Physical entity. Special type of experiment components (Chapter 4). Physical entities are used during laboratory activities or during instrumentation (e.g. microarray, hybridized array, sample, mRNA probe). See also *experiment components*, *activity*, *data element*.

Procedure element. Part of an experiment procedure (Chapter 4). An experiment procedure consists of procedure elements. At the experiment level, a procedure element represents an experiment component (i.e. a physical entity, an activity, or a data element). At the data level, every procedure element has a corresponding data type (i.e. a class) in the application database schema. See also *experiment procedure*, *experiment components*.

Process Data Flow (PDF). A tool for modelling experiments and experiment-related information (Chapter 4). A PDF is a comprehensive, component-by-component definition of an experiment in a domain. A PDF defines every component involved in the experiment, the properties of each component, and the relationships among the components. A PDF is used as a tool for modelling experiments. See also *experiment components*.

Process Flow Template (PFT). One of the three components of the VLAM-G experiment model (Chapter 5). PFT is the term introduced in the VLAM-G referring to the 'experiment procedure'. See also *experiment procedure*, *study*, *topology*.

Resource. General description (all chapters). Any resource that is needed and used during a scientific experiment in VL, such as computing, storage and networking resources, instruments, data/information, and software.

Reuse policy. Policy type for managing experiment contexts (Chapter 4). Reuse policy is used to define the rules for sharing objects, such as defining the set of objects that are related to a shared object (i.e. an object copied to another context) and hence that also need to be shared, or how to handle sharing of information about instruments or software for which only a single object exists (singleton objects). See also *sharing and reusing objects*, *semantically related objects*.

Scientist. User type in a VL (Chapter 2). A scientist is typically associated with an e-science domain (e.g. molecular biology). Scientists are the actual users of the VL.

Semantically related objects. Related to policies for managing experiment contexts (Chapter 4). Given an object, related objects that provide extra (descriptive) information about that object are called semantically related objects. For example, hardware/software tools and the protocol used to perform the activity, extra information provided as properties, and the comments of other users (if any) comprise the semantically related objects for an activity. See also *sharing and reusing objects*.

Sharing and reusing objects. Related to policies for managing experiment contexts (Chapter 4). Users may need to *share* experiment-related data/information when working on an experiment context. Information to share can be either at the level of an entire experiment context or at the level of objects in a context. For example, if a user wants to perform an analysis experiment on some sample which was prepared in another experiment in another project, in order to use the (description of the) sample in her/his experiment, the user must first copy the sample object from the original experiment context to her/his context which is under a different project than the origin context. In some cases, only a single copy of an object can exist. This is usually the case when the objects are created by domain experts or administrators and can not be updated by ordinary users. Hence, such objects can not be copied, but they must be *reused*. That is, when working on an experiment context, instead of creating a copy of an object and including the copy in the context, objects in the context are directly linked to the original object through relationships. This way, only a single copy of the object is preserved, while that object is reused by many users in different experiments. When sharing/reusing an existing object, all its semantically related objects are shared/reused. See also *semantically related objects*.

Software entity. A self-contained executable program (Chapter 4). Software entities are mainly characterized by the tasks that they perform, their input and output data types, and run-time requirements. Every software entity has a corresponding executable. Software entities can be attached to each other to define a computational processing. See also *computational processing.*

Study. One of the three components of the VLAM-G experiment model (Chapter 5). Study is the term introduced in the VLAM-G referring to the 'experiment context'. See also *experiment context, process flow template, topology.*

Tool developer. User type in a VL (Chapter 2). Tool developers develop support tools that are used in a VL environment, e.g. tools for data analysis.

Topology. One of the three components of the VLAM-G experiment model (Chapter 5). Topology is the term introduced in the VLAM-G referring to the 'computational processing of an experiment'. See also *computational processing, process flow template, study.*

Update policy. Policy type for managing experiment contexts (Chapter 4). Update policy defines what to do in case when a shared object (i.e. an object copied to another context) is updated. See also *sharing and reusing objects, semantically related objects.*

Virtual organization. A temporary alliance of enterprizes that come together to share resources, skills, or core competencies towards reaching a common goal (Chapters 3 and 7).