

A Metadata Framework for Assisting Experiment Planning and Evaluation at Wendelstein 7-X

Anett Spring, Michael Grahl, Marc Lewerentz, Cornelia Klug, Simon Dumke, Torsten Bluhm, Heike Laqua, Martin Grün, Andreas Holtz, Heike Riemann and the W7-X team*

Abstract—Complex experiments such as the stellarator Wendelstein 7-X demand efficient experiment planning, operation and data evaluation. To assist the process organization at W7-X, a metadata framework has been implemented. It annotates a physics program with tags describing its intention, main parameters, boundary conditions, and evaluation results. The approach enables relevant metadata to be collected exactly where information is available and to forward it from experiment planning and execution to the related entry in the central W7-X Logbook. In consequence, these metadata offer possibilities for classifying, searching, filtering etc. both in the database of prepared programs and in the logbook of executed programs. Wherever possible, metadata are generated automatically. Starting from the program’s parameters, the framework allows typical physics or technical quantities or any programmatically extractable information describing the programs intention to be extracted already while editing. During discharge preparation, metadata are retrieved from the related session planning documents and from the actual session environment. Supplementary tags can be added manually from an extendible tag catalog to categorize experiment programs or to provide metadata for not fully integrated diagnostics. All metadata are logged together with status and progress information during execution and are available within the program’s log – where the metadata can be finally completed e.g. to manually qualify the program’s success or by adding programmatically produced metadata using the logbook’s programming interface.

Index Terms—Wendelstein 7-X, Metadata, Software Framework, Data Management, Experiment Evaluation, Data Navigation

I. INTRODUCTION

THE stellarator Wendelstein 7-X successfully went through its first operational phases with evolving complexity [1]. The number of operational components and diagnostic systems fully integrated into the central control and data acquisition system rose to about 30. In addition, about 30 partly integrated diagnostics were in operation. As a result, physics experiments planning became more and more complex. At the same time,

the tight operating regime with up to 60 experiment programs per day made efficient preparation, execution and evaluation essential.

This paper presents the solution implemented at W7-X as an extendible, configurable, highly automatable metadata framework spanning the complete tool chain from experiment configuration to evaluation.

II. W7-X CONTROL CONCEPTS

At W7-X, fast control is implemented as *Segment Control* [2]. Physics experiments (both plasma discharges and technical tests) are set up as *experiment programs* consisting of an arbitrary number of *segments*, each containing *tasks* for each integrated *component* with sets of parameters. An experiment program fully defines the intended behavior of all integrated components over the scheduled time interval. Components comprise both technical components – as heating and gas injection – and physics diagnostics.

Planned *experiment programs* – as parametrization of a static component configuration – are stored in the *ProgramDB*. During the execution of an experiment program, all integrated components are centrally controlled. Data and logs are written to a common archive. Information is afterwards available as *executed program* in a central logbook.

Conceptually, at W7-X all components are integrated and centrally managed by a common workflow and an integrated tool chain – assisting both central experiments and stand-alone operation of components for commissioning or calibration [3]. However in practice, there are several diagnostics under test or provisional operation and not (yet) fully integrated. These *partly integrated components* are controlled by pre-defined triggers from the central segment sequence controller. A crucial disadvantage is that they cannot benefit from the common tool chain – at least not without respective additional interfaces – because e.g. their parameters are not contained in the planned experiment program (but defined externally), their status is unknown to the central controller, and they do not write logs to the central archive.

This work has been carried out within the framework of the EUROfusion Consortium and has received funding from the Euratom research and training programme 2014-2018 and 2019-2020 under grant agreement number 633053. The views and opinions expressed herein do not necessarily reflect those of the European Commission.

Anett Spring and the co-authors are with the Max-Planck-Institut für Plasmaphysik, CoDaC Software Development Group, Wendelsteinstraße 1, 17491 Greifswald, Germany (e-mail: anett.spring@ipp.mpg.de).

* Wendelstein 7-X team as defined in T. Klinger et al., Nuclear Fusion **59** (2019) 112004

III. USE CASE ANALYSIS

A. Metadata definition

Full traceability of the intended experiment programs, environment settings, and the actually executed experiment parameters is essential for efficient operation, experiment evaluation, team-wide data access, and data analysis. The obvious strategy is to enrich both program planning and execution data by relevant context information and to provide these *metadata* for program and data organization where required.

Metadata are referred to as additional information describing the process and environment of the actual experiment data acquisition, such as the intention of an experiment program, the technical and organizational environment, or quality and supplementary experiment relevant information. With this definition, metadata are as relevant as the experiment data and represent an instrument for data interpretation, organization, and scientific queries as increasingly discussed for quite general use cases and very different data types [4], [5], [6] – even for meta-level topics as data provenance itself [7], [8].

While metadata schemata and fast searching are much-discussed research topics – not only in the fusion community – this contribution outlines an integrated framework for metadata collection from diverse sources and the routing along the experiment workflow.

B. Metadata origin within the workflow

Dependent on the type of metadata, the respective information is available at different stages within the experiment workflow. At W7-X, the workflow for preparing, running, and evaluating experiments has been implemented as a complete tool chain consisting of applications, services, and appropriate interfaces. At every stage of this tool chain, it has to be possible to add or enhance metadata. All information is to be kept throughout the tool chain.

1) Experiment program parametrization

Planned experiment programs are defined as a complete set of parameters fully determining the intended behavior of all fast controlled integrated components for any point of experiment time. Subsets of related parameters can be abstracted to a higher-level program description and define metadata.

2) Session preparation

In addition, the experimental environment at W7-X is characterized by supplementary components that are not determined by the program parameter set-up but nonetheless contribute to the experiment's run:

- Components controlled by the operational management, e.g. the main magnet field
- Not fully integrated or external diagnostics

This information is available only at the time of the experiment preparation in the control room. With respect to documentation, also organizational data needs to be linked to experiment programs:

- The related physics proposal(s)
- Experiment organization data such as responsible session leaders, taskforce leader, session organizers etc. according to the control room related roles at W7-X

3) Experiment execution

During the actual experiment run, technical environment information is available and has to be collected:

- Status information about the components
- Program progress status, i.e. information about the technically successful execution of a program and its parts, like successful, aborted, or even skipped segments

4) Experiment evaluation

After a program's execution, both the session leaders and the diagnosticians may evaluate the outcome of the experiment from their specific point of view by a rating or additional notes.

5) Experiment data analysis

Finally, the subsequent thorough data analysis may provide a more detailed characterization of an experiment program.

C. Metadata generation

Metadata as descriptive, characterizing data can be extracted from parameter settings where parameters are complete and evaluation rules are automatable. This is always the preferred solution to assure correct data. However, where full integration is not (yet) possible or individual ratings come into play, it must be possible to add metadata manually.

1) Automatic metadata generation

For fully integrated components, the defined parameters fully describe the component's behavior. Knowing the metadata evaluation rules, the extraction can be automated for the time of program set-up.

2) Programmatic metadata generation

Even for not fully integrated components with their own parametrization and for each kind of programmatic data analysis, metadata might be extracted semi-automatically.

3) Manual metadata generation

Certain metadata such as comments, individual ratings, or any not automatically extractable information can only be entered manually by the user. To assure consistency, user input should be assisted, either by validation or by a catalog of potential metadata entries.

D. Metadata access within the workflow

According the intention of the metadata it has to be available throughout the tool chain:

1) Preparation time

For reasons of a quick overview, metadata have to be apparent while browsing or searching (prepared or even executed) programs and immediately while editing a program.

2) Execution and monitoring

The same applies to the overview while choosing and announcing the next program for the users.

3) Evaluation and documentation

At the end for documentation and searchability, the enhanced set of metadata has to be available within the experiment logbook.

IV. ARCHITECTURE

From the use case analysis with the different metadata origins and ways of creation, it is apparent to implement the metadata handling as a modular, extendible, and configurable framework – consistently spanning the whole tool chain, both for central and stand-alone operation. Metadata are added where arising, made persistent, and provided for the requesting applications and services within the tool chain (fig 1). This approach differs from others (e.g. in [9]) where the metadata enhancement is done at the stage of the documentation only.

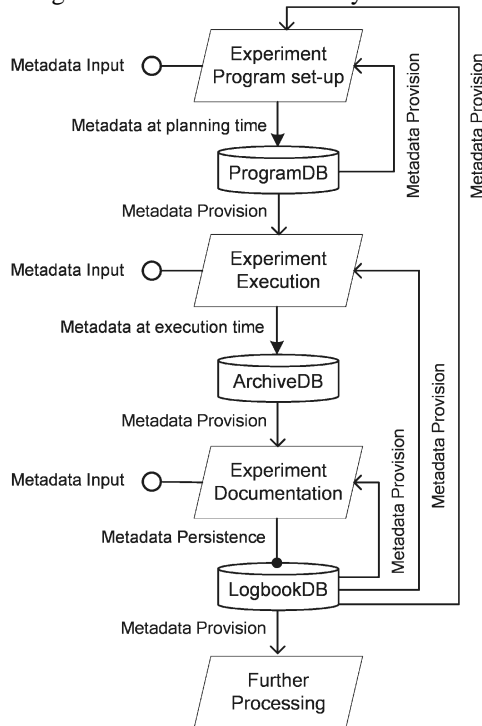


Figure 1 Architecture overview of metadata enrichment and routing along the experiment workflow.

The architecture is designed to route metadata across the lifecycle of an experiment program where metadata are persisted in two intermediate steps: what has been planned (*ProgramDB*) and what has been executed (*ArchiveDB*). After experiment execution, all collected metadata are persistent in the *W7-X Logbook* database to be further enriched. In turn, all metadata can be accessed by the other stages of the workflow: both in the central documentation and in the planning tools as guidance for next programs to be set up based on executed programs.

Interfaces are implemented for programmatic or manual input from extendible sources. The fact that not all metadata can be retrieved automatically and the existence of not fully integrated diagnostics demand for flexible interfaces for diverse information sources.

V. WORKFLOW FROM THE USERS' POINT OF VIEW

Figure 2 illustrates the intended metadata collection and

routing through the workflow from the users' point of view: Starting with the automated creation out of a program's parameters within the experiment program editor *Xedit* [10], the preparation of the next experiment run by the segment control user interface *Xcontrol* [11], and finally the documentation of an experiment run in the central *W7-X Logbook* [12].

At each stage, tags are consistently used as overview and organization means e.g. for sorting or filtering. Deliberately, all manual input of (not automatable) metadata is optional for the users. The intention is not to hinder the workflow by any hard restrictions.

A. *Xedit* – *W7-X Experiment Program Editor*

Xedit [10] allows both generating experiment programs from the scratch and – the mostly used way – to start from already available programs by selecting one from the *ProgramDB* listing and adapting its parameters for explorative experimenting. While editing, metadata are generated automatically using the ComponentModel framework (see chapter VI.B). Additionally, manual input of a categorization tag and comments are possible. The metadata are displayed as tags in the user interface. They are stored with the experiment program and are available in the *Selection* view for a fast visual capture of a program's content. For programs that have already been executed, additional information is available from the *Logbook* (see chapter VI.E).

B. *Xcontrol* – *W7-X Segment Control User Interface*

Xcontrol [11] uses the same *ProgramDB Selection* view to pick the next experiment program to be executed. Before execution of a program, it enhances the program's metadata by session related metadata i.a. from the *ProposalDB* (see chapter VI.C). Again, a dialog for manual user input is available showing a list of additionally configurable tags predefined in a catalog (see chapter VI.A). During execution, *Xcontrol* converts runtime information from the control system like status and progress into metadata (see chapter VI.D) – provided for live monitoring and logging.

C. *W7-X Logbook* – *Experiment Documentation*

The *W7-X Logbook* as a web-based application [12] is the central point where all collected metadata of an experiment program are assembled, displayed, and can be used for navigating experiment data (see chapter VI.F). Here, metadata are – manually or programmatically – further enriched (at this stage primarily resulting from experiment evaluation or data analysis) and in turn provided for *Xedit* or other programmatic processing.

Manual metadata input is done via dialogs assisted by a list of available tags (see chapter VI.A). A programming interface allows programmatic metadata insertion (see chapter VI.E).

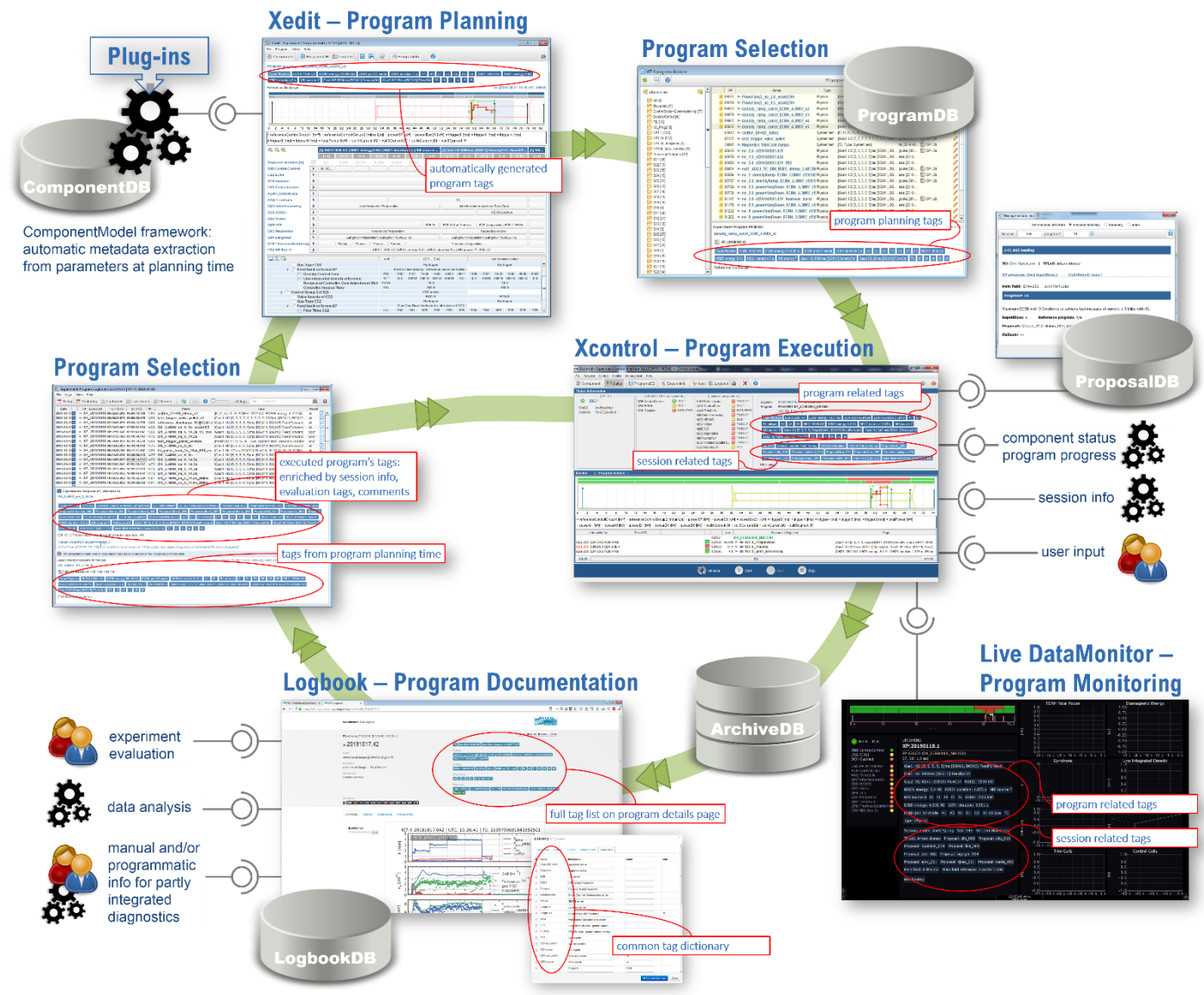


Figure 2 Overview on the metadata interfaces, processing, cross-tool routing and availability

VI. DESIGN AND IMPLEMENTATION DETAILS

Following, some of the implementation specifics of the intended metadata workflow will be described in more detail.

A. Structured metadata format and dictionary

For programmatic processing, metadata are held in a consistent format. The design provides a generic *tag* container that can accommodate various types of metadata: text, Integer / Double values with physical units, Boolean values. Additionally, the container includes process information to assist the different views, e.g. for displaying metadata in graphical user interfaces, for providing predefined selection lists for interactive user input, or for assisting programmatic examination. Attributes hold the related component, the metadata scope (program or experiment related), and the generation mode (automatic or to be configured manually).

A central dictionary comprises all predefined tags, organized by categories (fig 3).

The tag catalog is available to the applications and services throughout the tool chain thus ensuring metadata consistency along the workflow.

```

▼ Catalog:
  name: "TagDictionary"
  ▼ description: "Catalog of predefined tags for W7-X experiment programs"
  ▼ category:
    ▼ 0:
      name: "Experiment"
      description: "Experiment program description"
      ▼ tags:
        ▼ 0:
          name: "Type"
          scope: "program"
          mode: "config"
          value: "Physics"
          unit: null
          description: "Program type"
          ▶ selection: [...]
        ▶ 1: {...}
      ...
    ▼ 1:
      name: "Heating"
      description: "Heating settings"
      ▼ tags:
        ▼ 0:
          name: "ECRH"
          component: "CBG ECRH"
          scope: "program"
          mode: "auto"
          value: 0
          unit: "kW"
          description: "Max. power all ECRH gyrotrons"
          selection: null
        ▶ 1: {...}
      ...

```

Figure 3 W7-X tag catalog (excerpt) as dictionary for pre-defined tags

B. Component Model Framework for automated metadata extraction

In [13] a configurable, extendible framework for generating views on experiment program parameters has been introduced. For now, it is in use for generation of parameter preview curves, and for defining and checking dynamic parameter constraints immediately while editing. At edit runtime, it provides rich data objects by ‘*translating*’ the actual parameters into the static component configurations. These *component model* objects are created per task, per segment, and for the overall program. Special *generators* construct different *views* on these component model objects: the above-mentioned graphical curves or limit checks.

For the automated creation of metadata while setting up a program’s parameters in *Xedit*, a new component model view has been introduced. This includes (1) the configuration of the generator and the actual evaluation rules for the metadata extraction from parameters, and (2) the implementation of the evaluation functions. The first is defined with the static component configuration persisting in a *ComponentsDB*. The second can be standard functions as mean, max, min, integral values etc. or might be specific for a component. New functions can be easily introduced to *Xedit* via plugin technique.

This configurable approach enables typical physics or technical quantities – generally speaking any programmatically

extractable information – to be derived from the program parameters instantly while editing. Furthermore, the implemented automation ensures the creation of a set of mandatory metadata for each experiment program, comprising information about intended heating, fueling, and magnetic field correction control schemes, the integrated diagnostics’ data acquisition settings, and the like.

C. Session related metadata interfaces

In *Xcontrol*, an authorized session leader selects the experiment program to be run next. Besides the available program metadata, further metadata are retrieved from the environment (as the authorized session leader) or by user dialogs, e.g. to complete session information by comments or add tags for not fully integrated components. For the latter, a list of predefined tags (to be filled with respective values) is retrieved from the central tag catalog.

At W7-X, the *ProposalDB* holds the scientific proposals with the physics intention of the planned experiments as well as the pre-planned experiment sessions with additional information such as environmental or organizational settings. In the course of selecting the experiment program to be executed next, this information is collected via the *ProposalDB*’s application programming interface (API) and linked to the program to be executed using the introduced metadata containers. Again, it is assured that mandatory metadata are collected for a most complete and correct documentation possible. At this stage, this includes e.g. the main magnet field’s settings, additional (not fully integrated) diagnostics’ operation, related physics proposals, responsible session leader, etc.

By pressing the button to start a program, *Xcontrol* logs both the complete program parameters and the collected metadata to the central *ArchiveDB*.

D. Segment control metadata interface

The experiment control framework at W7-X comprises not only the concerted operation of the technical and data acquisition systems but also the automatic documentation by logging status and progress information into the central *ArchiveDB* [14]. *Xcontrol*, as control interface and logging instance collects this information by communicating with the central segment sequence controller: this includes the integrated components’ status, the actual timeline, and the technical success of each program part.

All metadata are available for live display on the *DataMonitor* [15] and automatically logged to the *ArchiveDB*.

E. Documentation and evaluation interfaces

The central W7-X experiment and components *Logbook* automatically builds the logbook entries from the above-mentioned program logs. Metadata from the planned program are provided consistently as tags on the logbook pages. Status and technical success are equivalently handled and displayed. Session leaders and diagnosticians can enrich the metadata by additional tags or comments. This is where the physics program evaluation is documented or where every external information that cannot be extracted automatically is added manually or programmatically.

The enhanced information is made persistent in the *LogbookDB*. A customized API using JSON as data interchange format [16] provides (1) detailed queries on executed programs by metadata attributes and (2) programmatic metadata writing – as interface for the not fully integrated diagnostics and for data analysis processes.

F. Data-metadata relation for navigation

A crucial issue with large collections of acquired experiment data is their efficient exploitation. Effective use of the introduced metadata framework empowers the *Logbook* to find and navigate relevant data by using the implemented link between metadata and data.

At the steady state capable W7-X, all acquired data are indexed by timestamps and address in the *ArchiveDB*. While editing, metadata are linked to the prepared experiment program and its segments and tasks in the *ProgramDB*. At execution time, they are explicitly linked to the time interval of the executed program and its parts – thus uniquely relating to the acquired data within these time intervals. On this basis, the *Logbook* serves as a starting point for data navigation by providing direct links to measurement data via the WebAPI [17] of the central *ArchiveDB*.

VII. USAGE EXPERIENCE AND FURTHER DEVELOPMENTS

A. Exemplary metadata usage

A crucial point of agile developments is the close collaboration with the users. To follow this rule, led to many improvements and adaptations of the metadata framework and its interfaces during the last W7-X experimental campaign.

For the session leaders, the main task is to keep track of experiment types and their outcome. The multitude of physics proposals – each requiring a series of experiment runs – and the one-team approach at W7-X – not allocating dedicated experiments for guest scientists – demand for combining experiments with matching intentions. A strict but agile session organization distributes related programs across different experiment days and at the same time has to offer alternative schedules immediately in the event of outages of components or diagnostics. For this, appropriate program metadata together with the subsequent success rating in the *Logbook* provide a quick overview on satisfied or still pending proposals (both for session leaders and for proponents) and assist a responsive session planning by filtering for tag values:

- physical key values of the planned program (intended ECRH and NBI power, fueling scheme, impurities and pellet injection, plasma density control scheme, intended plasma duration, magnetic field corrections etc.) as extracted from the programs parameters in *Xedit*,
- category (physics, test, conditioning, etc.) as added with program setup in *Xedit*,
- main static magnetic field configuration or actions from not fully integrated diagnostics from the session environment in *Xcontrol*,
- related proposals from the session detailing imported by *Xcontrol*, and

- the program evaluation tag from the session leader entered in the *Logbook*.

Similarly, the diagnosticians use metadata to screen experiment programs with special properties, such as high-density plasmas, dedicated heating modes, impurity entries, or machine condition (e.g. after wall conditioning or boronization). In return, they assist the team by providing key metadata from their diagnostics.

B. Status and outlook

The presented metadata framework is reliably in operation and intensely used by session leaders and physicists during the last W7-X operation phases for improved experimentation. It supports the communication within the team of about 500 scientists and technicians and significantly increases the utilization of the limited experiment time.

So far, the expected use cases are covered. At the same time, the users' experience creates new requirements, which can be met by the configurable and extendible implementation.

The actual types of required metadata are determined by the intended use cases and user groups. They are to be enhanced over the W7-X lifecycle. Hence, the central tag catalog has been expanded in co-operation with the users. Based on user input regarding the semantic usage of the tagging, the visualization within the tools is to be adapted, e.g. to visualize differences between the intended and actually executed program.

As the underlying architecture provides extensibility, further metadata sources and applications can be attached. The more components will be fully integrated, the more automation can be implemented, and the more valuable are the overall metadata. On the other hand, the increasing number of components demands for improvements on the user interfaces.

Already under evaluation is the systematic data enrichment by automated data analyses. First examples of programmatic post-experiment data evaluations are already implemented by users, where results are inserted as metadata (for the present, as pictorial comments with diagrams of the key plasma parameters) and thus made available for the team colleagues. It is envisaged to enhance the metadata model to store and provide more complex data structures such as time-value sequences.

Since analysis processes have to be triggered by the availability of data, the concept for an eventing system is under discussion. Together with a common data domain model, which is under development with the physicists, metadata from analyses can considerably enhance data navigation. Ideas for using metadata for complex and use-case driven navigations such as in [18] are to be explored.

On the future development plan at W7-X is the implementation of a framework for scientific queries on the increasing experimental data basis. The metadata framework will be a main building block of a resulting scientific query database.

VIII. ACKNOWLEDGEMENTS

The authors like to thank the W7-X team for fruitful requirements discussions and in-depth application evaluation.

REFERENCES

- [1] H.-S. Bosch et al., Status of W7-X and the way to steady state operation, SOFE Jacksonville/FL 2019
- [2] H. Laqua et al., Experiences with the Segment Control system at Wendelstein 7-X operation, Fusion Engineering and Design 123 (2017), pp. 588-592.
- [3] A. Spring et al., Establishing the Wendelstein 7-X steady state plasma control and data acquisition system during the first operation phase, Fusion Engineering and Design 123 (2017), pp. 579-583.
- [4] P. Pinoli et al., Metadata management for scientific databases, Information Systems 81 (2019), pp. 1-20.
- [5] A. Polyakov et al., Toolkit for intensive work with metadata in specialized information systems, Procedia Computer Science 119 (2017), pp. 59-64.
- [6] J. Stillerman et al., Data catalog project—A browsable, searchable, metadata system, Fusion Engineering and Design 112 (2016), pp. 995-998.
- [7] I. Lupelli et al., Provenance metadata gathering and cataloguing of EFIT++ code execution, Fusion Engineering and Design 96–97 (2015), pp. 835-839.
- [8] G. Abla et al., The MPO system for automated workflow documentation, Fusion Engineering and Design 112 (2016), pp. 1007-1013.
- [9] A. Krämer-Flecken et al., Experiment management system—A way towards a transparent Tokamak, Fusion Engineering and Design 83 (2008), pp. 375-381.
- [10] A. Spring et al., A W7-X experiment program editor—A usage driven development, Fusion Engineering and Design 87 (2012), pp.1954-1957.
- [11] A. Spring et al., User control interface for W7-X plasma operation, Fusion Engineering and Design 82 (2007) 1002–1007.
- [12] M. Grahl et al., The new W7-X logbook—A software for effective experiment documentation and collaborative research at Wendelstein 7-X, Fusion Engineering and Design, In press, Available online 26 February 2019, <https://doi.org/10.1016/j.fusengdes.2019.02.052>.
- [13] M. Lewerentz et al., Experiment planning using high-level component models at W7-X, Fusion Engineering and Design 87 (2012), pp. 1949-1953.
- [14] Ch. Hennig, et al., ArchiveDB—Scientific and technical data archive for Wendelstein 7-X, Fusion Engineering and Design 112 (2016), pp. 984-990.
- [15] S. Dumke et al., Next generation web based live data monitoring for W7-X, Fusion Engineering and Design 129 (2018), pp. 16-23.
- [16] M. Grahl et al., W7-X Logbook REST API for processing metadata and experiment data enrichment at the Wendelstein 7-X stellarator, 12th IAEA Technical Meeting on Control, Data Acquisition and Remote Participation for Fusion Research, Korea 2019, to be published in Fusion Engineering and Design.
- [17] M. Grahl et al., W7-X ArchiveDB WEB API: a web service for the experiment data archive of Wendelstein 7-X, Fusion Engineering and Design 123 (2017), pp. 1015-1019.
- [18] J. Stillerman et al., Scientific data management with navigational metadata, Fusion Engineering and Design 128 (2018), pp. 113-116.