



Document Title

**Common Specification for the
FAIR Accelerator Control System (PSP 2.14.10)**

EDMS Document No

F-CS-C-01e

Doc Date yyyy-mm-dd

2014-03-06

Doc Version number

v3.1

Abstract

The document describes the Common Specification for the Accelerator Control System for the FAIR Accelerator Project. It covers the PSP code 2.14.10.

Table of Contents

1	Purpose and Classification of the Document	4
1.1	Responsibilities.....	4
1.2	Classifications of Requirements	4
2	Scope of the Technical System	6
2.1	System Overview.....	6
2.2	Limits of the System	7
2.3	Basic Requirements for the Control System Conception.....	8
2.4	Architecture	10
3	Technical Specifications	12
3.1	Controls Network	12
3.2	Timing System.....	12
3.3	Equipment Access and Frontend Controller	13
3.4	Device Access Model and Frontend Software	15
3.5	Industrial Control Systems.....	16
3.6	Common Services	16
3.7	Settings Management.....	17
3.8	General Application Concept.....	18
3.9	Control Room Concept	18
3.10	Interlock System	19
3.11	Post Mortem System	20
3.12	Maintenance	20
4	Quality Assurance, Tests and Acceptance	22
4.1	Development Methodology	22
4.2	Quality Assurance at Contractor Site.....	24
4.2.1	Software components.....	24
4.2.2	Hardware components	25
4.3	Quality Assurance at Contracting Body Site	26
4.3.1	Software Components	26
4.3.2	Hardware Components.....	26
4.4	FAT (Factory Acceptance Test).....	27
4.4.1	Software components.....	27
4.4.2	Hardware components	28
4.5	SAT (Site Acceptance Test)	28
4.5.1	Software components.....	29
4.5.2	Hardware components	29
5	Documentation.....	31
5.1	Documentation for Electronics Hardware	31
5.2	Documentation of Software Systems.....	32
5.3	Further Documentation	33
6	Warranty	34
I.	Attached Documents	35
II.	Related Documentation	35
III.	Document Information	35
III.1.	Document History	35

List of Figures

Figure 1: Architecture Overview	10
Figure 2: Mechanical model of a SCU	14
Figure 3: Iterative Software Development	23
Figure 4: Rational Unified Process: Phases and Iterations	23

1 Purpose and Classification of the Document

The purpose of this document is to provide a general overview description of the scope of the accelerator control system and general specifications of the control system and its subsystems. Detailed specifications are given in separate documents. This document does neither cover software interfaces to the control system nor technical interface specifications of equipment connected to or being controlled by control system. These specifications are defined as Technical Guideline documents.

Concerning the organization the **contracting body** must define the technical guidelines, must specify the control system functionality and must approve the delivered products.

The **main contractor** must coordinate development of components, must define the overall control system architecture, must define development guidelines, must define interfaces between components, must integrate components, must set up development and test environments. The main contractor verifies the design proposed by a contractor regarding the integration of the component into the overall control system and the adherence to all stipulations.

A **contractor** is responsible for a single component of the control system. All contractors which contribute to the control system must adhere to all stipulations defined by the contracting body and the main contractor.

1.1 Responsibilities

The responsibilities with respect to changes and modifications of the present document are entirely in the hands of the Accelerator Controls and Electronics Department (CSCO) of the GSI Helmholtz Centre for Heavy Ion Research GmbH (GSI) Darmstadt.

For initial information please contact the administration of the Accelerator Controls and Electronics Department.

Further information on the organizational chart, names of responsible persons and task leaders, as well as the agreed document release and approval procedure is summarized in the organizational note 'Controls Project for FAIR'.

1.2 Classifications of Requirements

The following definitions of requirement classifications are being used throughout the document:

- **"Must"** or **"shall"** or **"is required to"** is used to indicate mandatory requirements, strictly to be followed in order to conform to the standard and from which no deviation is permitted.
- **"Must not"** or **"shall not"** mean that the definition is an absolute prohibition of the specification.
- **"Should"** or **"is recommended"** is used to indicate that among several possibilities one is recommended as particularly suitable, without

Document Title: Common Specification for the Accelerator Control System

mentioning or excluding others or that a certain course of action is preferred but not required.

- "**Should not**" or "**is not recommended**" mean that there may exist valid reasons in particular circumstances when the particular behavior is acceptable or even useful, but the full implications should be understood and the case carefully weighted before implementing any behavior described with this label.
- "**May**", which is equivalent to "**is permitted**", is used to indicate a course of action permissible within the limits of the standard.

2 Scope of the Technical System

2.1 System Overview

The technical system "accelerator control system" comprises the full electronics, hardware and in particular software infrastructure base for all accelerators and beam-lines of FAIR and GSI that is needed to control, commission, run and operate the GSI/FAIR accelerator complex including existing GSI installations.

In detail, the scope of the control system work-package contains:

- All hard- and software of the equipment (field) layer of the control system to interface and control all types of accelerator devices. This includes the necessary control and interface electronics, the front-end controller (FEC) CPU systems as well as the embedded (real-time) front-end control software (PSP code 2.14.10.5 "Front-end systems (Device interfaces + controller)").
- All types of human machine interfaces (HMI) for operators in the control center (HKR) needed to plan, set, monitor, and operate the GSI/FAIR accelerator chain and beam-lines according to the designated operational concept. This includes dedicated applications for beam diagnostics measurement data (PSP code 2.14.10.2.2).
- All integrated functions and system services expected for a state-of-the-art accelerator control system, e.g. settings management, alarm system, archiving system, diagnostic and operation logging, software oscilloscope system, communication middleware systems, user rights management, etc. (PSP code 2.14.10.1 "Core System Software Packages" and PSP code 2.14.10.2 "System Software Packages").
- A high precision time- and event distribution system to synchronously orchestrate accelerator equipment distributed all over the GSI/FAIR site. The timing system includes a central installation (Timing Master), all active communication and transmission components and dedicated timing receiver electronic boards for all relevant equipment. (PSP code 2.14.10.3 "General Machine Timing System" together with the RF-System PSP code 2.14.10.4 "Bunch Timing System").
- All active and passive Ethernet network components needed for the accelerator control system. It is foreseen to provide network access for other users in the FAIR buildings by the general use of VLAN technology (PSP code 2.14.10.9 "Network").
- Full IT hardware base for the accelerator control system. This includes a central installation of rack-based powerful servers, hard-disk arrays, a high-performance database installation for configuration and operational data, backup services, disaster recovery system, and all systems to host general IT services (PSP code 2.14.10.8 "Computing and Controls Infrastructure").
- Main control room installations. This includes operator consoles, fixed displays and dedicated control electronics to be installed in the Main

Document Title: Common Specification for the Accelerator Control System

Control Room ("Hauptkontrollraum") (PSP code 2.14.10.10 "Main Control Room").

- Machine safety systems. The control system covers a beam interlock system as well as protection functions for beam inserts. This covers signal aggregation, processing, and transmission technology but not the signal producing or consuming equipment except these are inherent systems of the control system (e.g. timing system) (PSP code 2.14.10.11.2 "Machine Protection Systems" and PSP code 2.14.10.11.3 "Interlock System").
- Personnel Safety System. The control system covers an industrial class monitoring and controlling system for access of personnel and radiation protection. It covers appropriate I/O systems, signal aggregation, transmission, processing and business logic but not the signal producing or consuming equipment except these are inherent systems of the control system (PSP code 2.14.10.11.1).
- Retrofitting and modification of selected existing controls solutions in the present GSI injector chain accelerator system in order to allow seamless integration in FAIR (PSP code 2.14.10.2.12 "System and GSI device integration").
- Cabling needed for all components of the accelerator control system, e.g. Ethernet network, Timing system network, field-buses, safety and interlock signals, etc. (PSP codes 2.14.10.3.2/3, 2.14.10.4.3, 2.14.10.9.4/5).
- Required infrastructure to house control system technology such as cabinets, racks, crates, etc. (PSP codes 2.14.10.8.8, 2.14.10.8.10).

2.2 Limits of the System

Notwithstanding the definition of the scope defined in the previous section, the control system work-package does not cover the following items or aspects:

- Control systems for physics experiments (users of the particle beams) or any testing installations (e.g. magnet testing) are not covered by the control system work-package.
- The hardware delivery and device class software implementation of/for the front-end systems of beam instrumentation data acquisition systems are not covered. Instead this is part of the respective beam instrumentation work-package(s).
- The implementation of the accelerator physics models for the respective machines and beam-lines is out of the scope of this work-package, but must be provided by machine physics specialists. Nevertheless, the software framework for consistent setting generation of the GSI/FAIR accelerator chain is covered in this work-package (see chapter 3).
- Within the control system a comprehensive database will be operated. Nevertheless, the control system work-package is not responsible and does not take ownership of data not directly related to the control system or its subsystems (e.g. magnet calibration data).

Document Title: Common Specification for the Accelerator Control System

- The technical system does not cover any building or civil construction activities related to the control system, e.g. enlargement or reconstruction of the present GSI control center HKR ("Hauptkontrollraum").
- Analog signal oscilloscopes and other similar measurement equipment for operators or experimenters used in the control center HKR are not in the scope of the control system. Such dedicated systems are not foreseen and will instead be realized by software based control system solutions.
- Equipment and electronics for local control rooms (other than the main control room) are not in the scope of the control system. Remote control rooms (other than on GSI site) are not supported.
- Integration of hardware and software components and third party control and data acquisition systems (e.g. experimental control systems) using other than the defined controls interfaces are not supported.

2.3 Basic Requirements for the Control System Conception

The requirements for the FAIR accelerator control system are in several aspects well beyond the capacity of the existing control system of the present GSI accelerator chain. Since substantial revision of the existing system (leading to significant investments) would have been required in order to enable the system for FAIR, decisions were taken that new functionality shall not be retrofitted to the present control system. Instead, a new control system for FAIR shall be designed and implemented that respects the new functionality needed (e.g. data acquisition performance, safety functions). The existing GSI control system is presently being modernized and obsolete technology replaced in order to allow integration into a new FAIR control system.

The general design concepts are as follows:

- A new control system shall be designed and implemented for the integrated GSI/FAIR beam facility. It extends the existing GSI accelerator control system and enables users to control the full GSI and FAIR accelerator facility in an integrated way.
- The same common accelerator control architecture, infrastructure, hardware and software base shall be used for all FAIR machines and beam-lines and will be moreover applied to existing GSI machines wherever needed for integration.
- The system shall be substantially built on proven principles and solutions of the existing control system and extend them as needed.
- The system architecture shall be based on a strictly modular design with well-defined and designed interfaces. This allows breaking down the project in interconnected work-packages that can be developed and implemented rather independently.
- Proven solutions and complete system building blocks (e.g. software frameworks) from other control systems shall be used where applicable. GSI has set up technical collaborations, in particular with CERN, in order to benefit from synergy effects of framework solutions.

Document Title: Common Specification for the Accelerator Control System

- Standardization: Taking into account the technical role of the control system in the FAIR project, its level of complexity and in particular the high number of internal and external interfaces (e.g. to equipment connected) a strict policy of standardization on all aspects of the control system is essential (concepts, technologies, interfaces, etc.).
- Considering a long life-cycle of the control system, the supplier shall base his concepts and solutions on universal widely available and persistent standards and technical solutions (e.g. Ethernet, high-level programming languages C++/Java, etc.). Industrial components off-the shelf (COTS) shall be used wherever applicable. For electronics parts a second source generally shall be available to minimize procurement risks.
- Open technologies and solutions (e.g. Linux OS, open source software products) shall be used as much as possible. Closed solutions, licensed code and commercial products shall be avoided as much as possible to avoid future dependencies and maintenance service costs.
- Where possible, the FAIR control system will be validated and tested already at the existing GSI machines in order to avoid parallel commissioning of a new control system and new FAIR machines.
- The control system shall be designed not only for the FAIR stages (modules) 0-3 but for the full FAIR accelerator facility, including further accelerator parts that are considered experiments (e.g. FLAIR). The system shall avoid system performance limits and instead consider future extensions (e.g. GSI cw-linac project).
- Implementation during operation: The implementation of the new control system and its deployment at the existing GSI injector chain has to respect the available maintenance periods (e.g. GSI beam shutdown periods). The reliable beam operation of the present GSI injector chain with its present functionality must not be compromised during system development, implementation and commissioning.

The operational concept of the GSI/FAIR accelerator chain imposes the following additional requirements to the conceptual design:

- The FAIR control system must efficiently support a high degree of truly parallel beam operation of the GSI and FAIR accelerator chain. Parallel operation provides maximum integrated beam time and integrated luminosity for each of the different research programs operated in parallel. This requires a highly sophisticated sequence and cycle management.
- The GSI/FAIR accelerator facility shall be operated with a minimal operations team (for normal operation). The control system shall provide adequate means for an integrated efficient machine operation with a high degree of system automation and overview. This includes sophisticated modelling of the beam manipulation along the accelerator chains to calculate good initial settings and efficient tuning and trimming procedures.

2.4 Architecture

The FAIR accelerator control system must be based on loosely coupled components with well defined interfaces. Most of them can be categorized in three levels: equipment layer, middle layer and application layer (see Figure 1: Architecture Overview).

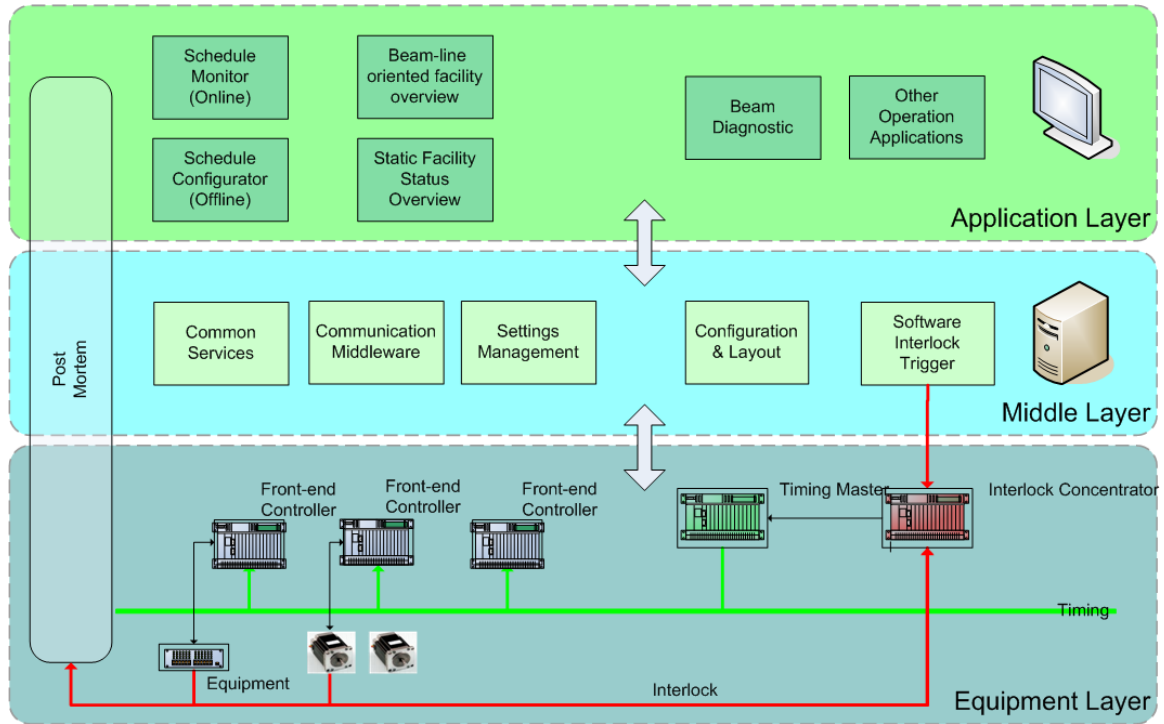


Figure 1: Architecture Overview

The equipment layer consists of hardware (“equipment”) interfaces and their representation (“device”) towards the rest of the control system. The hardware equipment is used to measure sensor data (e.g. for beam diagnostics) and control accelerator components (e.g. actuators of power supply of magnets). The software representation of the equipment (or group of equipment) as devices is done on “Frontend controllers” (FEC). The FECs are mostly Linux based embedded system controllers hosting the controller software (“FESA”). Another crucial part of the equipment layer is the “timing system” which sets up a dedicated real time message network which distributes timing events to synchronize activation of data settings or correlation of sensor data on FECs.

The middle layer must provide services both to the equipment layer and the application layer. These may be dedicated service processes hosted on powerful UNIX Servers as well as logical abstraction layers for inter component communication (middleware). The communication to the equipment and application layer is through the control system network using TCP/IP which generally interfaces and integrates the non real-time subsystems. Some important components of the middle tier are: service for data setting generation (“LSA”), accelerator configuration and layout service, alarm system, archiving, name services, device access middleware.

The application layer combines the applications for operators or other users like e.g. beam diagnostics expert, control system expert, experimenter. Typically,

Document Title: Common Specification for the Accelerator Control System

these are graphical user interface (GUI) applications or command line tools. Functional aspects of applications will be described in the “general application framework” and the “control room concept”.

Machine protection concentrates on detecting machine failures and causing the appropriate reactions. In principle, equipment has to protect itself. Only where this is not completely possible the machine protection system has to assure the protection - which is of course not in the scope of the control system. However, machine protection is supported by control system infrastructure and the interlock and post mortem components. These are orthogonal aspects of the FAIR control system which are specified in dedicated chapters.

To integrate FAIR components not in the domain of this EOI, the control system provides frameworks, e.g. FESA for the beam diagnostic specialists, LSA for the machine physicist, GUI guidelines and standard GUI components.

The technical implementation of software components of the control system must comply with the following non-functional system-wide requirements: All GUI applications must comply with the control system GUI guidelines. Any implementation of inter-process communication must comply with the accelerator control system's middleware and communication guidelines. Any usage of means of data persistence must comply with the accelerator control system's data persistence guidelines. Any implementation must respect constraints defined by the IT infrastructure (CPU performance, Network performance). Therefore, the control system main contractor will provide additional Development Guidelines ([8], [9], [10], [11], [12]) which must be applied in development of software components for the accelerator control system by all contractors to achieve a coherent system architecture and a coherent base of applied software technologies.

Furthermore, the Control System FAIR Technical Guidelines ([2], [3], [4], [5]) and the general FAIR Technical Guideline [6] must be respected.

3 Technical Specifications

While the overall architecture has been described in chapter 2.4, the main components are now being described in more detail. Each chapter contains the 'purpose' of the component, 'design' describes foreseen realizations and 'interfaces' lists known connections between those components.

3.1 Controls Network

Purpose: The controls network will be used for all non real-time communication within the control system. In addition, the controls network must be applied to transport streaming-data of experiments through an interconnection point to the general purpose network.

Design: The accelerator network must be designed as separated VLANs with high priority, based on a physical network with high bandwidth.

10-, 24- or 48-port manageable network switches are projected as edge network-devices. Only one end device must be allowed per port. The edge network switch uplinks (1000Mb/s, 1000Mb/s trunks or 10Gb/s) should be redundant (non-symmetric in bandwidth) connected to concentrators in every building/accelerator via fiber optics.

The concentrator uplinks (10Gb/s, 10Gb/s trunks or 100Gb/s (if available)) should be again redundant (non-symmetric in bandwidth) connected to a couple of core switches, located on a central point via fiber optics.

The existing accelerator network must be attached to the core switches.

Interfaces: 10/100/1000 Mb unshielded twisted pair ports with a RJ45 connector must be available on every outlet (4000 numbers), as default. Each device, which would like to communicate via network, must be known by MAC address. In this way a dynamic attachment of VLAN-IDs to the port is available.

3.2 Timing System

Purpose: A timing system is necessary to trigger and synchronize equipment actions, timed according to the accelerator cycles. The timing system must handle machine cycles and manipulation phases from 20 ms up to the order of several seconds for synchrotrons and up to several hours for storage rings. Furthermore high-precision equipment synchronization with sub-ns precision is needed for RF components. The timing system also must take real time decisions to decide correctly between predefined alternative cycles to be executed.

Design: To separate the two necessary timing domains, the timing system for FAIR must consist of two distinct, yet tightly coupled systems: The general machine timing system (GMT) with sub- μ s precision and the so called bunch phase timing system (see BuTiS Common Specification [7]) designed for

Document Title: Common Specification for the Accelerator Control System

distributed RF synchronization. Unless specially emphasised the term “timing system” is used only for the GMT.

BuTiS: For high-precision RF synchronization BuTiS distributes a 100 kHz ident clock, a 10 MHz reference clock and a 200 MHz clock. Distributed RF receivers can be phase matched to each other and receive a corrected time reference with sub-ns-precision. RF receivers must be interfaced by SCUs (see FAIR common specification “RF system”).

GMT: The FAIR timing network must be a physical separate network in tree topology with one global system timing master atop. It will be based on synchronous Ethernet and PTP (precision time protocol) using GbE (Gigabit Ethernet) and special layer 2 switches. Triggering and synchronization of equipment action shall be realized by broadcasting events to all timing event receivers facility-wide. The synchronization requires accuracy in the low ns range. The distributed time is UTC based and defines the time standard for the complete control system. Timing receivers must be part of the equipment: integrated inside the SCU or implemented as a special timing receiver board, the later mainly for beam diagnostic usage.

Each event must have a unique identifier, an execution time and must carry additional information. Events must be grouped in time windows of about 100 μ s and it must be guaranteed that they reach all receivers in time. To reach highest robustness forward error correction algorithms must be implemented.

The timing master must autonomously orchestrates the complete facilities beam operations in a predefined schedule. Alternatives must be executed depending on real-time decisions based on e.g. interlock information or experiment requests. Higher layers of the control system must be informed about all decisions taken by the timing system.

Interfaces: The timing master must be a logical device inside the control system. The timing master gets input from the interlock system and may generate interlocks itself. It must be able to cause alarms. It must receive its programming from the settings management through the front-end controller software.

3.3 Equipment Access and Frontend Controller

Purpose: At lowest level of the control system, equipment interfaces must be accessed by front-end controllers (FEC) to communicate with the *equipment*. Besides providing set values for the equipment and reading out measured data the front-end controller must also monitor the equipment and is responsible for notifying the control system about a malfunction.

In order to synchronize actions on front-ends, some of the FECs must be equipped with a standardized timing interface.

Design: At the equipment layer, the various actuators, sensors, and data acquisition equipment must be interfaced to the control system through the following types of FECs:

Document Title: Common Specification for the Accelerator Control System

- Scalable Control Units (SCUs) are the standard controllers for FAIR. Most accelerator devices (like power supplies, RF-systems, kickers, etc.) must be interfaced by a SCU. The SCU is a control system network node, connected also to the timing network (wherever necessary), and provides local CPU-power for real-time control, fast data acquisition as well as any specific functionality needed (e.g. state-dependent tolerance band control). It is a processor board based on the Intel x86-architecture with a Linux operating system. Time critical functions must be implemented in FPGA technology. A dedicated parallel bus is defined as an interface to electronics boards of devices, which must be supported by the foreseen slave boards. The SCU provides a device-implemented function generator (FG) for equipment that needs to be controlled by time-dependent functions (ramps). The FG must provide linear and quadratic interpolation at 1 MHz data rate between base points with 24 bit output resolution.



Figure 2: Mechanical model of a SCU

- VME, PCI, PCIe and μ TCA single board computers based on the Intel X86-architecture with a Linux operating system used for high-performance real-time processing and data acquisition. Such systems can employ a large variety of standardized and custom I/O modules (ADC, DAC, binary I/O, Counters, etc.). Typically, timing receiver boards, beam diagnostic systems and interlocks are implemented in this technology.
- FECs (which in this case could be standalone gateway PCs) that communicate with Programmable Logic Controllers (PLCs) which themselves control the industrial equipment. PLCs can be chosen when the process is not synchronized to accelerator timing and when sampling periods are longer than ~ 100 ms. Being highly reliable, cost effective and easy to program via standard high-level languages, PLCs are foreseen e.g. to manage and control vacuum components, machine cryogenics or RF monitoring.

Document Title: Common Specification for the Accelerator Control System

Interfaces: Frontend controllers must access their equipment as described above. FECs which need timing information must be directly connected to the timing system (SCU) or use timing receiver boards connected via the backplane.

Interface to the interlock system must be done directly on equipment level and/or on frontend controller level via software or hardware. FECs must connect to upper layers of the control system via the control system network, as described in the following chapter.

3.4 Device Access Model and Frontend Software

Purpose: In order to integrate various equipments into the control system, the device access model must provide a uniform software interface towards the upper layers. It must allow modelling similar equipments as abstract device classes where the equipments' process variables (sensors and actuators) are represented as properties.

To allow time multiplexed operation of the accelerators, devices must keep several settings and enable fast switching between them triggered by events of the timing system.

Design: Front-end software must be based on the FESA framework which is an environment to design, develop, test and deploy real-time control software for the FECs. It will also be the new standard for the GSI injector chain.

The front-end software itself is split in two logical layers: *device presentation* which implements the device classes as well as the network access, and *equipment control* which interfaces the equipment and does the real-time handling. Both functional layers can be physically implemented on the same hardware platform.

Single devices are represented as software objects of a particular device class towards the upper layer. Combinations of several single device classes into a new integrated class are used to model more abstract devices. The device properties are set and read using synchronous or asynchronous access methods or subscription.

Time multiplexed operations are realized by real-time actions of the front-end software which are triggered by events from the timing system. Based on event information, the corresponding setting for the component is selected to fit the actual beam parameters. Thus, all necessary settings must reside in the FECs.

Front-end software must react on equipment interlocks and other malfunctions with appropriate error handling. Alarming and post mortem actions are essential parts of it (see 3.6 (Common Services), 3.11 (Post Mortem)).

Interfaces: The front-end software must have interfaces to its equipment and to SCADA systems via gateways. It must also have interfaces to the middle layer (using middleware for communications), to the alarm system and to the timing system.

3.5 Industrial Control Systems

Purpose: Some of the technical subsystems of the FAIR facility (e.g. machine cryogenics, vacuum system, facility monitoring) are not time-critical with respect to accelerator timing and highly industrial related. In order to integrate these subsystems in the control system, the components must be able to take part in the alarm system, provide time stamped measurement data (e.g. to correlate vacuum quality to beam intensity, display state of vacuum valves) and must allow access to configured set points. However, industrial control systems must be designed as standalone systems and their functionality must not require any connection to the control system.

Design: The industrial control system for those subsystems must consist of a PLC system infrastructure and a common supervisory control and data acquisition system (SCADA).

Gateway processes must connect the industrial control system to the accelerator control system by representing technical components as logical devices. Write access from the accelerator control system to the industrial control components may be locked by the industrial control system.

The UNICOS framework (unified industrial control system) developed at CERN which is based on the commercial SCADA product PVSS2 is evaluated for usage at FAIR for vacuum and cryogenics.

Interfaces: Device access to industrial control system components must be done by accessing logical devices realized within the FESA Framework to allow control system integration. The equipment interface will be realized by the industrial control systems (e.g. digital/analog I/O, Profibus DP/PA, Profinet, Profisafe, RS232, OPC DA/UA).

3.6 Common Services

Purpose: Being a component of the middle tier, a common service must provide complex functionality for other components of the control system, such as the equipment layer or the application layer.

In addition, some complex and resource intensive high-level applications shall be designed with separated business and presentation tier. The non-visual business part then becomes available to other applications as a service.

Design: Amongst others, the following components must provide common service functionality to the complete stack of control system software.

Alarm System: The alarm system must transport, process and visualize error conditions (malfunctions) of hardware and software. It must also propagate alarm states from producers to consumers. Producers of alarms are logical devices or middle layer services. A consumer of an alarm might be a human operator or software to process the alarm or to trigger reactions to the alarm. The alarm system does not cover the scope of a hardware-based safety system (like e.g. interlock system, machine protection system) but may serve as a possible

Document Title: Common Specification for the Accelerator Control System

interface to the control system for such systems. The alarm system must not be considered as a safety critical system.

Archiving: The archiving service must store data collected and generated by the control system. It must allow to store data on a configurable repetition rate, acquisition triggered e.g. by timing events or on-change. Data is either raw or processed data gathered from the equipment layer, higher level data computed by other services or applications, or generated abstract data like an alarm status. The service must include functionality to retrieve and filter historical data. The archiving service does not cover the administration and storage of the accelerator's configuration or setting data. It does not cover post-mortem storage.

Diagnostic Logging: The diagnostic logging service must provide the functionality to store human readable text entries generated by the complete stack of control system software in a coherent way to allow efficient querying. In first order these text entries are generated for diagnostic and debugging usage. Also this service can be used to create a text-protocol of control system activities for operational usage (see Operational Logging).

Operational Logging: The operational logging service must provide functionality to keep track of the control system activity in form of a text protocol. This service may be realized by retrieving special diagnostic logging entries.

Software Oscilloscope System: It is assumed that distributed oscilloscopes in the FAIR facility already have a vendor specific GUI client application to display measured data. The Software Oscilloscope Service must allow selecting and integrating this output to an operator's console. Configuration of the oscilloscopes and raw data retrieval should be realized where applicable.

3.7 Settings Management

Purpose: To be able to control the complex FAIR machine, a central settings management service is necessary based on a physical model for accelerator optics (twiss, machine layout), parameter space and overall relations (between parameters and between accelerators). The settings management must allow generation of consistent settings, coherent modification of settings as well as hardware exploitation like equipment control.

Design: The settings management system must be a service of the control system's middle layer. It must be realized by using the CERN settings management system LSA (LHC Software Architecture).

Using this framework, the GSI/FAIR accelerators will be set and manipulated as much as possible on the base of high level physical values. The framework itself is modular and allows implementing the physics model of the different accelerators reusing common parts. Standardized APIs allow accessing data in a common way as basis for generic applications for all accelerators.

The settings management allows offline generation of machine settings. Triggered by client applications, the service is able to send these settings to all

Document Title: Common Specification for the Accelerator Control System

involved devices and activate them by programming the timing system. Applications for trimming allow consistently changing these settings.

Interfaces: The settings management presents standard APIs towards applications. It contains persistence functionality in order to keep settings and to allow accessing historical settings. The service communicates with devices using the device access component (see 3.4 “Device Access Model and Frontend Software”).

3.8 General Application Concept

Purpose: Applications for standard operation developed by the controls group or external suppliers (e.g. beam diagnostic group or external institutes) must fit a common application concept.

Design: On the one hand, an application framework and standard components must result in a common “look and feel” of applications used in the control room. Therefore, common application windows, GUI selector elements, error message handling and GUI elements in general are specified. On the other hand, the application concept must fulfill functional specifications resulting from the FAIR operation concept. To take account of the several parallel operation modes and complexity of FAIR the application layer must provide various aspects of operating the complete facility. Such aspects are e.g. a facility status overview which can be grouped physically by accelerator or other components and a beam status information and control which works logically on a production-chain basis (source-target).

Set up and integration of new experiments (i.e. defining additional beam production chains) must be as much as possible supported by tools providing a high degree of automation and must follow predefined workflows. Since it is planned not to use analog signals in the main control room, additional software must be provided to measure and control hardware components using the standard controls network, e.g. the software oscilloscope service. Additionally it must be possible to display continuously measured data from SCU interfaced devices in applications.

Interfaces: Applications must use common interfaces which are provided by the middle layer, e.g. the device access API, common service APIs or interfaces to allow inter-component communication.

3.9 Control Room Concept

Purpose: Operation control of the FAIR complex and the existing GSI accelerator facilities will be located in the existing main control room (HKR) of GSI and additional (2-3) local control rooms.

Design: The IT infrastructure equipment of the control rooms must be connected to the technical control system network. In the main control room, the existing UNILAC console will remain substantially unchanged due to the specific

Document Title: Common Specification for the Accelerator Control System

hardware infrastructure (analog signal connections). Further on, FAIR control must be done at several FAIR operation consoles which are made up by one or many multi-head displays, keyboard and mouse input device. The FAIR consoles must not be connected to any analog signal of the control system. Besides, fixed displays inside and outside of main control room are foreseen to display information of global interest (e.g. scheduling, facility status, beam properties).

A FAIR console must run a preconfigured selection of applications to represent a standard operation environment. However, it must be possible to start other applications or combination of applications on any console (e.g. due to commissioning or new experiment setup). Dedicated accelerators may alternatively be controlled by a local control room. This option must be granted by means of the main control room and may be cancelled there. The control room concept and the application concept must allow limiting the number of FAIR operation staff in standard operation mode to 5. This must be considered by the application design. Generally a remote control is not foreseen, but for technical support remote login must be possible.

3.10 Interlock System

Purpose: The interlock system must concentrate various interlock source signals, process them and trigger the appropriate reactions. In case of interlocks, it must be possible to switch the beam to a pilot beam or disable further beam production of this or all beam production chains, plus eventually dump the beam in an upstream accelerator. It is not the responsibility of the interlock system to guarantee internal safety of any equipment.

The interlock system is part of the machine protection system, which protects the accelerator from damage by mislead beams and e.g. minimize radioactive contamination by unforeseen beam deposition.

Design: Equipment must provide fault conditions – like missing cooling water – as summary interlock information to the interlock system (detailed fault information must be provided additionally to the front-end controller as a status pattern). Furthermore, equipment itself or front-end controllers shall provide interlocks derived from e.g. actual value deviation. Besides faulty equipment, other sources of interlocks can be equipment monitoring beam parameters like beam position, online beam transmission, radiation by beam loss, or beam on collimators. On a higher level, software processes can also be sources to the interlock system. The autonomous beam dump system (part of the machine protection system) must generate interlock signals to inhibit further beam production on the affected beam line and has to trigger post mortem. Generally, interlock signals must change the overall scheduling in such a way that e.g. further beam production will be prevented.

Interlock signals must be reliably transported.

Another aspect of machine safety is to prevent beam destructing diagnostic elements from damage by high intensity beams. Insertion of those elements into the beam must only be allowed in low intensity mode. However, their insertion into the beam line must generate interlock signals to prevent the production of high intensity beams.

Document Title: Common Specification for the Accelerator Control System

Interfaces: The interlock system interfaces equipment, front-end controllers and software processes. All of them can be considered to be sources and targets of interlock information. In order to disable beam production and to trigger post mortem the interlock system must closely interact with the timing system.

3.11 Post Mortem System

Purpose: The Post Mortem System must provide a complete snapshot of the machine for further analysis in case of a major malfunction. The information must include status information, measured values and their evolution in time for a short period before and after the incident. Therefore the Post Mortem System must be built as collection of various diagnostic services. It reflects an orthogonal aspect and imposes requirements on several components of the control system. It must be triggered by various input channels to inform all devices which took part in producing the beam to freeze their data for further analysis. It must be possible to freeze only parts of the facility. The Post Mortem System must include means to collect, correlate and analyze data of several origin and abstraction level.

Design: Logically, the input channels to trigger a post mortem event are the quench detection system, beam loss monitor system and other sources of the interlock system. The distribution of post mortem events must be done via the timing system and the control system network for those components not participating in the timing system. The post mortem event on a front end controller may even be self-triggered.

The front end controller software must be able to store transient data in a circular Post Mortem Buffer of appropriate length. The post mortem event must trigger a freeze of this buffering and cause the front end controller to enter the post mortem state. The format of the post mortem data which is collected by the post mortem service must be coherent across the control system, i.e. it has to be time stamped and self-describing in a generic format. The display and analysis tool must incorporate additional information from the interlock system, the alarm system and operational or diagnostic logging. A selection of collected post mortem data must be archived on a configurable base. After the machine is brought back into full operational state, the post mortem control service must allow resetting the state of the front end controller and its post mortem buffer.

Interfaces: Interfaces to the timing system and the control system middle layer must be used to propagate the post mortem event. The front end software must react on the post mortem event (using appropriate internal ring buffers). The analysis tools must interface the alarm, archiving and logging services to correlate post mortem data with the other sources of information.

3.12 Maintenance

Purpose: Since the control system is a complex system which is assumed to have a lifetime of more than a decade it is essential to focus on good maintainability. To achieve this, methods must be foreseen reaching from diagnostic tools in case of an error on the one hand to reliable and redundant structures for core components on the other hand.

Document Title: Common Specification for the Accelerator Control System

Design: The maintenance aspect is an orthogonal aspect which affects all layers of the control system. For each layer, adequate possibilities for error diagnosis must be provided, e.g. remote login into front-ends with log analysis, test procedures and status monitoring, monitoring of central services and infrastructure components, central diagnostic logging for applications with the possibility to observe inter-process communication.

Means to provide quality of service (reliability, scalability and performance) also contribute to the maintainability of the system. In general, loosely coupled and modular components ensure that necessary changes typically affect only single components of the control system. Additionally, specific hardware and software components must be implemented redundantly.

Since a continuous accelerator operation is expected, a test environment for all components must be established which must support not only the development but mainly error diagnostics and fault correction. During the whole lifetime of the control system this test environment must also contribute to continuous improvement and adaptation to new requirements.

4 Quality Assurance, Tests and Acceptance

The contractor has to fulfill the QA requirements specified in the FAIR General Specification [1]. In addition and substantiation to the FAIR General Specification this chapter applies.

Quality assurance must ensure that development of the control system implements the functionality specified by the contracting body. However, due to the complexity of the control system which mirrors the complexity of the unique FAIR facility, requirements are not fully known in advance or may even change during FAIR construction and operation. The control system integrates many underlying systems and therefore evolves during FAIR development.

The overall complexity of the control system requests for an adequate development process to assure high quality. In addition and substantiation of the FAIR General Specification [1] an incremental and iterative methodology is advised, especially for complex software building blocks and for the system as a whole. However, many components will be produced in strict sequential development and production as described in F-GS-B.

Within the foreseen minimum lifetime of the FAIR accelerator complex of 30 years (see [1]), the computer hardware and software technologies will drastically change, the way the accelerator is operated will change and the accelerator itself will be extended. During the lifetime of FAIR, continuous adaptation and extension of the control system is therefore foreseen.

Thus, the control system architecture and the quality assurance must ensure a high level of maintainability and exchangeability of components. Maintainability is here defined as easiness to fix faults, to increase quality properties, and to adapt to changing environments. Hence, product quality must generally be achieved by adherence to architectural guidelines and development principles as defined by the main contractor.

4.1 Development Methodology

Based on the main requirements an overall solid system architecture is setup and in a first iteration a vertical slice through all layers is built realizing the most crucial functionality. Then, in well-defined iteration cycles, components are replaced by newer versions, implementing more functionality, and additional components are added to the base system. By the iterations, the control system will be extended to the full specified functionality.

In each single iteration cycle, requirements and designed functionality can be refined and adjusted, the functionality is implemented and the achieved components as well as the overall functionality are tested. Iteration holds for the system as a whole as well as for single components.

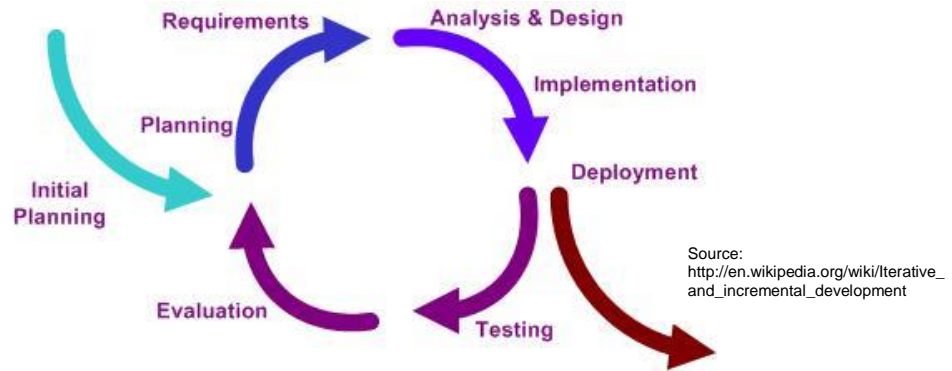


Figure 3: Iterative Software Development

As is demonstrated by the well established Rational Unified Process (RUP), refinement of requirements, analysis and design and testing extend over the whole project phase.

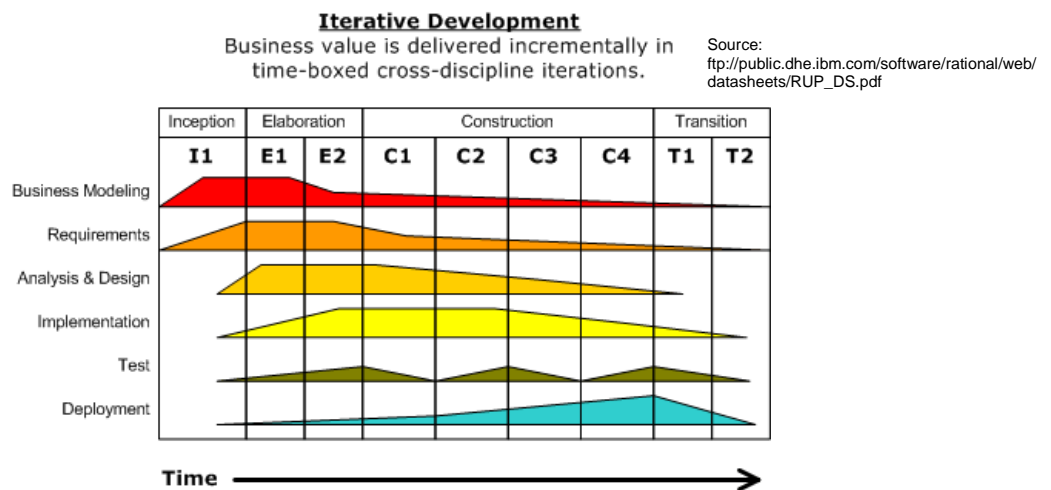


Figure 4: Rational Unified Process: Phases and Iterations

Each iteration cycle must to be accurately planned before it is started. The contracting body defines the functionality to be implemented in this iteration cycle and the tests to be performed to confirm proper working. Testing is, in every iteration cycle, on building block level as well as on the level of the complete system. Each iteration cycle must repeat tests from previous iterations.

Each iteration cycle will be performed in strict phases

- before start of iteration, the contracting body will define the
 - functionality to be implemented, especially correction of defects of former iteration cycle
 - test plan to verify proper functionality
- implementation
- test of functionality of component
 - at contractor site (FAT)
 - at contracting body site (SAT A), if appropriate
- integrate into running control system
- test control system overall functionality (SAT Ba)

- record all deficits found in tests

While incremental iterative development is adequate especially for complex software building blocks and for the system as a whole, for many components strict sequential development and production as described in F-GS-B will be used.

The contracting body must specify for each component the development methodology as well as all quality assurance measures before development is started.

4.2 Quality Assurance at Contractor Site

The contractor is responsible for informing the contracting body about his internal QA system including documentation of developments, tests, work-flows and change requests. This has to be approved by the contracting body before work is allowed to be started by the contractor.

The contractor's QA must respect standards which are aligned to ISO 9001, as demanded by the FAIR General Specification (see [1]). The appropriate measures may vary for each building block of the control system. The contractor does not have to be certified to ISO 9001.

The contractor must adhere to the system architecture and all other control system stipulations which are specified by the main contractor.

The contractor must obtain the approval of the contracting body for all technical concepts and detailed design reports before start of system implementation. The contractor must obtain the approval of the main contractor for the integration into the control system as well as the adherence to guidelines and standards before start of system implementation.

The main contractor must provide full vertical systems for validation and iterative integration tests as soon as available and if necessary. A central system at the contracting body's site for SATs and other systems for FATs at the contractor's sites must be provided. Such systems must be continuously extended and updated.

4.2.1 Software components

The contractor's QA must respect standards which are aligned to the ISO 9001 application ISO-IEC 90003. The contractor does not have to be certified to ISO-IEC 90003.

The contractor must implement an appropriate development process and methodology to respect the fact that detailed requirements for the control system cannot fully be specified at the project start.

The contractor must use a full development environment defined and provided by the main contractor. This development environment must consist at least of:

- Version control for source code and documentation
- Bug/feature tracking system (keeping track of open issues and tasks)
- Automated build tools and automated test tools

Document Title: Common Specification for the Accelerator Control System

- Project web
- Test environments
- Code standards
- Source code standards must be enforced through the use of automated tools.

4.2.2 Hardware components

The contractor's QA must respect standards which are aligned to the ISO 9001. The contractor does not have to be certified to ISO 9001.

The contractor must implement an appropriate development process. This process must establish at least the following:

- Complete documentation during development
- Principles for selection of components (e.g. second source, electronic components for industrial use)
- Design guidelines (reproducibility, operating grade of components, voltage utilization, current carrying capacity, power dissipation and temperature, ability to be tested)
- Worst-case calculations; simulations if reasonable
- Design standards for FPGA-Design (VHDL code guidelines, test benches)
- Test procedures and validation rules (type verification and test, temperature-rise test, routine test)
- Change service and product maintenance

The hardware design process must be organized in several phases. Each phase needs a written acceptance from the contracting body, before the start of the next phase.

For the great number of different hardware components (devices as well as complete control subsystems), which have to be designed for the FAIR control system project it is necessary to define a special quality assurance method for the dedicated hardware component type.

Therefore the first phase is to work out a detailed specification and an agreement for the whole design and production process of the special hardware component type.

The contracting body describes the functional requirements and defines the interfaces. The contractor writes the detailed specification.

This specification must respect the following aspects:

- Product specification sheet
- Development of hardware design and design documentation e.g. schematic representation, block diagram, integrated circuit layout, circuit diagram, cable list, connection diagram, assembly plan etc.
- Specification of needed design tools, standards and file formats, e.g. EPLAN Electric P8 for switching cabinets, EDA tools, VHDL development environment, test environment etc.

Document Title: Common Specification for the Accelerator Control System

- Prototyping, test installations and test environments, inspection certificates, dimensioning rules and calculations
- Approval process for the design phases, e.g. acceptance test of schematic diagrams
- Release and test procedures before starting a new phase
- Description of all required phases e.g. specification and design, production, delivery, commissioning, start up and putting into operation
- Agreement about the contractor's support at site during commissioning, assembly, installation, start up and putting into operation phases

For some components it is necessary to produce prototypes or test installations prior to the (series) production process to get confidence in the quality of the product or to evaluate system performance. In such cases the contracting body will provide specifications for prototype design, production and tests that must be respected by the contractor. Starting the production in these cases is not allowed without written acceptance of the prototype by the contracting body.

4.3 Quality Assurance at Contracting Body Site

The contracting body provides the specifications and by approving all delivered components must ensure that the specified functionality is realized.

The contracting body must approve the overall system architecture proposed by the main contractor and all technical concepts and detailed designs proposed by the contractors.

The contracting body must ensure that all contractors adhere to the technical guidelines provided by the contracting body itself and to all other guidelines provided by the main contractor.

The contracting body will specify for each component the development methodology as well as all quality assurance measures before development is started.

The contracting body must follow the development process.

4.3.1 Software Components

The contracting body must coordinate and direct that the same environment will be used for all contractors contributing to the control system or using its frameworks.

4.3.2 Hardware Components

The contracting body must coordinate the process of connecting equipment to the control system of GSI/FAIR.

4.4 FAT (Factory Acceptance Test)

After finishing one iteration of a component, successful internal function testing and before delivery, the Factory Acceptance Test (FAT) is used to ensure that the component meets the functional and non-functional requirements defined by the contracting body and is ready to be shipped.

For performing the FAT, a test procedure must be established and the contractor must provide the personnel and an adequate testing environment to perform the tests. However the complete GSI/FAIR environment including the accelerator facility itself and also other components of the control system cannot fully be represented in a test environment. Therefore, it is necessary that the FAT for components where it is obligatory to have the complete environment or parts of it present, respects that situation and is handled accordingly. E.g. it can be agreed in written form between the contracting body and the contractor to perform parts of the FAT at the GSI/FAIR site or to loan special HW to the contractor for all necessary FAT tests.

After receiving notification of readiness for the FAT, the contracting body shall decide on a case-by-case basis whether the test shall be carried out in the presence of contracting body's employees or if issuing a test certificate is sufficient. If the FAT will be attended by a responsible person of the contracting body and this attendance requires travelling to the contractor's site, the date of the FAT has to be officially agreed upon in advance.

Results of the FAT including all defects (errors, open issues and non-conformity with both functional and technical specifications) must be documented in an acceptance report and signed by the delegates of the contracting body and the contractor.

In case defects are identified, the contracting body may declare the FAT as passed provided the contractor hands over an agreed detailed schedule for fixing those issues.

A successful FAT is the necessary precondition for any SAT.

4.4.1 Software components

The FAT for software components consists of

- Unit tests for all parts of the component with a reasonable test coverage
- Integration Tests with other components if feasible (e.g. those other components exist in the environment, are already part of the test system provided by the contracting body, or are simulated)

System tests, which really ensure that the component meets the specified functionality, typically can only be performed in the full environment at GSI/FAIR. It can be agreed, that the FAT can take place at the GSI/FAIR site, then already the FAT can cover the full system test. But typically the full system test will be part of the SAT.

4.4.2 Hardware components

After manufacturing/production and before delivery, the equipment or product must be tested at the factory. The contractor must provide adequate testing equipment (compare below) and personnel to perform the tests. A test procedure must be established for each specific product or unit of a series production and must be delivered with the equipment or product including all test reports.

A successful FAT is the necessary precondition for both the start of mass production as well as any SAT.

In order to request a FAT the following conditions have to be fulfilled in advance:

- Detailed specifications and technical descriptions have to be delivered to and approved by the contracting body.
- Significant test results (type test report) of at least one prototype have to be approved by the contracting body. Dependent on the FAT even a small batch of prototypes may be required to perform all specified tests. The contractor is responsible to provide the necessary amount of prototypes.

The FAT for hardware components consists of:

- Functionality test
- Interface test
- Test of electrical and mechanical properties
- Test of assembly and wiring
- Type test (e.g. insulation tests from the power part)
- Test of the compliance with the specified tolerances

The manufacturer is responsible for providing all needed measuring instruments and test equipment for the FATs.

Before a component is transported to the FAIR site it is mandatory to perform all FATs. The results of the FATs must be documented in an acceptance report. Only if the outcome of these tests is accepted by the contracting body the contractor is permitted to start the transport procedure.

If agreed, prototypes used for the FAT can become part of the central test system at the contracting body's site.

4.5 SAT (Site Acceptance Test)

The final acceptance test shall be carried out after the installation of the accelerator complex and when testing without beam has been completed.

However, to be able to test components as early as possible, the Site Acceptance Test (SAT) after each iteration of a component ensures that the component fulfills all user requirements and that it is ready to use within the users environment. Each iteration cycle must repeat tests from previous iterations.

As specified in the FAIR General Specification [1], two different types of SAT exist – namely SAT A, which represents the tests after delivery and SAT B, which includes the integration of the component into the overall control system and if applicable, tests without and with beam.

Document Title: Common Specification for the Accelerator Control System

The SAT will be performed by at least one responsible person of the contracting body. All test results must be documented in a way that they are completely reproducible. Results of the SAT including all defects (errors, open issues and non-conformity with both functional and technical specifications) must be documented in an acceptance report and signed by the delegates of the contracting body and the contractor.

If critical defects exist, they have to be fixed and the SAT has to be repeated. If there is a planned next iteration for this component, only non-critical defects can be settled and retested within the next iteration. After successful tests and after approval of the documentation, the SAT is passed.

Wherever possible, the FAIR control system components will be validated and tested already at the existing GSI machines in order to avoid parallel commissioning of a new control system and new FAIR machines.

The warranty period will start when the product has successfully passed the final acceptance test without beam.

4.5.1 Software components

Necessary preconditions of each SAT are:

- Declaration of readiness for the SAT by the contractor.
- Successful corresponding FATs, all remaining items of the FAT have been settled.
- Delivery of all components.
- Complete documentation.
- Training of the GSI/FAIR staff, if specified in the corresponding detailed specification.

For software components the SAT Part A consists of compiling the software, deploying it into a test system and integrating it into the software environment.

The SAT Part B for software components consists mainly of

- the integration test with all other components of the control system – also those, which were not present for the FAT
- the full system test to assure that the component meets the specified functionality

As far as possible, the component will already be installed and tested within the existing GSI facility to give the opportunity to test the component under production conditions.

4.5.2 Hardware components

Necessary preconditions of each SAT are:

- Declaration of readiness for the SAT by the contractor.

Document Title: Common Specification for the Accelerator Control System

- Successful corresponding FATs and all remaining items of the FAT have been settled. In the case of mass production proof of successful adequate sample tests matching FAT requirements.
- Delivery of the component(s) to the contracting body's site.
- Complete documentation.
- Training of the GSI/FAIR staff, if specified in the corresponding detailed specification.

SAT Part A includes the tests after delivery to the contracting body's site but before the component is integrated in its final installation place.

SAT Aa is the incoming good inspection, e.g. a visual inspection.

SAT Ab includes all tests that are agreed upon with the contractor, e.g. repeating the function tests for random samples.

SAT Part B comprises all tests to be performed at the final installation place (installation in accelerators, beam lines or corresponding supply rooms)

SAT Ba (the test without beam) consists of:

- Verification of the assembly and wiring of cabinets and/or subsystems
- Power on test
- Operation test of hardware components as specified within the release Procedures for this dedicated hardware component type
 - Dry run
 - Functional test of the component itself
 - Functional test of the component in integration with its environment

The SAT will only be accepted when all remaining open issues are considered non-critical by the responsible person of the contracting body. Furthermore a detailed schedule about the fixing of all open issues has to be handed over by the contractor and must be approved by the responsible person of the contracting body before the SAT is passed.

If it is agreed in written form between the contracting body and the contractor a SAT may be passed even before all elements of a series have been delivered. In such case the complete acceptance procedure as described above has to be supplemented accordingly.

5 Documentation

In addition to the general documentation requirements as specified in the FAIR General Specification [1], the contractor must deliver all documentation as described in the following subchapters.

All documentation must be given in English (and German when available or specified as such) as electronic documents.

The contractor must provide an engineering design report that must include at least a conceptual description and a detailed design description.

All technical concepts and design have to be given in form of a design report to the contracting body and must be approved by the responsible coordinating group of the contracting body before start of construction.

5.1 Documentation for Electronics Hardware

The contractor must provide the following obligatory technical documentation and data for electronics hardware:

- Schematic diagrams
- PCB layout (layer separated if pdf, jpeg, tiff etc.)
- CAD/CAE files
- Layout diagram
- Production data (e.g. Gerber)
- Engineering Bill of Materials (BOM, engineering part list; includes second source, alternative parts, technical notes)
- Manufacturing BOM (structural part list, includes all necessary parts for the final product)
- Assembly instructions
- Type test report (prototype approval process)
- Test protocols or test reports of measurements during development
- Test procedures, test specification, test code
- Worst-case measurements and simulation results (if relevant)
- Technical data (includes environmental conditions, electrical and mechanical data, maximum ratings etc.)
- User manual (includes information for intended use, implementation, configuration, operation etc.)
- Maintenance manual
- Functional description (includes design rules, block diagrams, flow charts, worst-case considerations and calculations etc.)

Additional required documents and data for FPGA designs:

- Source code, commented (VHDL is mandatory)
- Test benches, commented (VHDL is mandatory)
- Binaries
- Simulation results
- Timing analyses

Document Title: Common Specification for the Accelerator Control System

- Functional description (includes block diagrams, flow charts, definitions, names and meanings of variables, etc.)
- Work flow description (includes information about design tools, binary generation, revision control etc.)
- Programming and deployment instructions

Additional required in case of product enhancement (change management):

- Release notes (includes reasons for changes and newer requirements, possibly problem report, compatibility, technical changes etc.)
- Product history

Documentation for Hardware components of the industrial control system:

- circuit diagrams, block diagrams
- cable and connection lists
- connection layout, wiring diagram
- terminal diagrams, terminal assignment
- cabinet layout
- field bus/network layout
- system/subsystem layout
- piping and instrumentation diagram (PID)
- interface and connector pin-out
- parts list (BOM)
- PLC source code, commented
- Type test report (prototype approval process)
- Test protocols or test reports
- User manuals for all integrated hardware components

The documents must be delivered (unless otherwise agreed) as well in English as well as in German language in PDF/A-Format according to ISO 19005-1 and EPLAN Electric P8 source files.

For all electrical equipment the supplier has to deliver the test certificates for the high voltage test, insulation and protective conductor test pursuant to the IEC standard.

The engineering and the installation of the electric control unit has to fulfil all current relevant European and IEC standards and norms, especially the German Machinery Directive (98/37/EG and 2006/42/EG).

The supplier has to deliver a "CE Declaration of Conformity" and must sign the components with the CE marking.

The documentation must fully integrate the FAIR nomenclature and ID system information.

5.2 Documentation of Software Systems

The contractor must provide the following obligatory documentation for all software products and subsystems:

Document Title: Common Specification for the Accelerator Control System

- All documents related to the software development process of the component: specification, design documents, architectural overviews
- Implementation related documentation
 - Code documentation (e.g. Doxygen or JavaDoc generated)
 - API description
 - Installation manual
 - Prerequisites of hardware and software (versions of used third party libraries)
 - Configuration manual
 - Further documentation if necessary (e.g. hardware interface, device model)
- User documentation (for HMI) as online documentation (Latex generated HTML or PDF documents) either as static functionality description or workflow-oriented description

Like the software itself, the provided documentation must follow dedicated guidelines and has to be reviewed as well.

Documentation for Software components of the industrial control system:

- All documents related to the software development process of the component: specification, design documents, architectural overviews, functional analysis, instrumentation lists, grafcet templates, logic specification
- Implementation related documentation
 - Code documentation
 - API description
 - Installation manual
 - Prerequisites of hardware and software (versions of used third party libraries)
 - Configuration manual
 - panel and object pictures including generation tool
 - Further documentation if necessary (e.g. hardware interface, device model)
 - every type of source code (e.g. SCADA), commented
- User documentation (integrated into SCADA system) either as static functionality description or workflow-oriented description

Like the software itself, the provided documentation must follow dedicated guidelines and has to be reviewed as well.

5.3 Further Documentation

As generalization of 5.1 and 5.2 the contractor must provide the following obligatory documentation for products and subsystems in any case:

- Complete interface descriptions
- User manuals
- Functional descriptions
- Detailed technical descriptions

Document Title: Common Specification for the Accelerator Control System

- Maintenance manuals

Where applicable further documentation must be delivered for

- Cabinets and racks
- PLC based systems
- Circuit diagrams
- Connection schemes, lists
- Networks, e.g. Ethernet
- Field-busses and signal cables

6 Warranty

The specifications from the FAIR General Specifications [1] apply.

I. Attached Documents

List of abbreviations for controls (Abbreviations_Controls.pdf).

II. Related Documentation

- [1] F-GS-B-0.1e, FAIR General Specifications
- [2] F-TG-C-01e, FAIR Technical Guideline "Ethernet Network Connectivity"
- [3] F-TG-C-02e, FAIR Technical Guideline "Control System Equipment Control Interfaces"
- [4] F-TG-C-03e, FAIR Technical Guideline "Control System Equipment Interlock and Status Signal Interface"
- [5] F-TG-C-04e, FAIR Technical Guideline "Control System Equipment Functional Requirements"
- [6] F-TG-ET-01e, FAIR Technical Guideline "Electrical Design Rules and Regulations"
- [7] F-CS-RF-14e, FAIR Common Specification "BuTiS"
- [8] F-DG-C-03e, FAIR Development Guideline "Software Architecture Guideline"
- [9] F-DG-C-02e, FAIR Development Guideline "GUI Guideline"
- [10] F-DG-C-01e, FAIR Development Guideline "FESA Development Guideline"
- [11] F-DG-C-04e, FAIR Development Guideline "Equipment Integration Guideline"
- [12] F-DG-C-05e, FAIR Development Guideline "Control System Naming Guideline"

III. Document Information

III.1. Document History

Version	Date	Description	Author	Review / Approval
0.1	26. Apr. 2011	First version		
1.0	06. Oct. 2011	Revised changes	R.Bär, J.Fitzek, T.Fleck, G.Fröhlich, L.Hechler, R.Huhmann,	

Document Title: Common Specification for the Accelerator Control System

			U.Krause	
3.0	21. Aug. 2012	Incorporated FAIR review comments	CCT	CCT
3.1	5. Mar. 2014	Inconsistencies with IKC eliminated	R. Bär	