Available on CMS information server

CMS IN 2001-xxx

CMS Internal Note

The content of this note is intended for CMS internal use and distribution only

March 13, 2001

CMS Software and Computing Tasks and Deliverables

The CPT Project

Please address comments to:
`Lucas.Taylor@cern.ch`

Draft
Version 0.11

Abstract

This document describes the task breakdown and deliverables of the CMS CPT project, where:

"C" denotes "Core Computing and Software" (CCS);
"P" denotes "Physics Reconstruction and Selection" (PRS); and
"T" denotes "Trigger and Data Aquisition Sustems (TriDAS) but, in the context of CPT, covers only the online software components of the TriDAS project.

Cross-project activities, notably the SPROM (Simulation PROject Management) and RPROM (Reconstruction PROject Management) groups are included.

It is intended to form *a starting point for discussions* that will ultimately lead to:

- a complete task breakdown structure for all CPT activities;
- descriptions of the main deliverables of each task;
- the definition of an organisational sub-structure for the CPT activities with L2 managers and clear reporting lines and channels of communication between the CCS, PRS, TriDAS and detector sub-projects;
- a schedule with well-defined milestones;
- a baseline estimate of the resources required to complete the project according to the required scope and on schedule.

Executive Summary

Fill this in later

blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit

List of Tasks

List of Deliverables

1 Computing Centres *(Martti Pimiä)*

This is supposed to cover the fundamental computing and software infrastructure required for any collaboration. Many, but certainly not all, of the services covered are closely related to those provided in a general way at CERN by the CERN/IT division.

1.1 T0/T1 Centre at CERN *(Coordinator name)*

The T0/T1 at Cern is to support all raw data storage, the first reconstruction pass and adequate connectivity to the outside T1 and directly supported T2 centres. In addition, user support for the Cern-resident community has to be provided. It will provide about 1/3 of the total required computing resources in CMS.

And some more *blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit*

Deliverable 1.1−A	**T0/T1 Liaison**
Description:	*Good contact has to be maintained with Cern/IT.*
Responsible:	CCS group, Cern-based community
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	On-going.
Constraints:	Common T0/T1 for all four experiments. No direct authority.

Deliverable 1.1−B	**T0/T1 Configuration**
Description:	To fit the CMS computing model, the resources of the T0/T1 will need proper configuration, optimisation, tuning... (CPUs, disk servers, mass storage, network topology, etc...)
Responsible:	Responsible person
Client/Sponsor:	Production groups, Online daq.
Completion date:	On-going.
Constraints:	Common T0/T1 for all four experiments. No direct authority.

Deliverable 1.1−C	**T0 Distributed Computing**
Description:	Provide connectivity and services to T1 RCs and attached T2s. Coordinate with outside centres.
Responsible:	*Responsible person*
Client/Sponsor:	All outside centres.
Completion date:	Ongoing.
Constraints:	WAN bandwidth, No direct authority.

Deliverable 1.1−D	**T0 Online connectivity**
Description:	Connection of CMS Online DAQ with T0.
Responsible:	*Responsible person*
Client/Sponsor:	Online, CCS
Completion date:	Fully functional in 2005. Prototypes before then.
Constraints:	Online system design.

Deliverable 1.1−E	**T0/T1 Needs, monitoring and trouble-shooting**
Description:	CMS needs have to be specified, usage of facilities has to be monitored, problems have to be investigated.
Responsible:	*Responsible person*
Client/Sponsor:	Production groups, CCS, Online DAQ.
Completion date:	Started already, fully functional in 2005.
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 1.1−F	**...more tasks...**
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

1.2 Tier 1 Regional Computing Centres *(Coordinator name)*

Each CMS Tier I Regional Center will provide deliverables to the experiment in several different categories. These include providing computing resources, staff to manage the computing, staff to provide software support, staff that will collaborate on development of CMS software products, and staff to share in the R&D and design of CMS computing systems. These contributions will be divided between deliverables to the experiment at large and meeting the needs of the local physics community. While the precise amount of the different contributions will vary from one regional center to another, they will be present to some degree in each. We give figures here for a typical center.

T1 RCs are supposed to provide a full service, 24×7, with an excellent connectivity to the T0/T1 at CERN and its attached T2s. It is optimised for scheduled tasks.

Deliverable 1.2−A	*Computing resources*
Description:	The Tier I center will provide CPU cycles, online disk storage, mass storage, I/O bandwidth between the CPU and disk/mass storage, and network bandwidth between the center and CERN and between the center and supported Tier II and local CMS institutes. Typical quantities of these resources are 150-200K SI95 CPU cycles, 300TB of disk, 1-2PB of mass storage, and 1Gb/sec network connections. A substantial portion of the computing resources are assigned to perform some portion of tha standard CMS processing, including simulation, reconstruction, event selection, physics object creation, and physics group analysis, as well as storage for canonical data samples. These computing tasks are predictable in their demands, and they will be shared among all the CMS centers to ensure all tasks are covered somewhere. These jobs will act as background activities in the center, but enough resources must be reserved for these tasks to ensure timely completion. A fraction of the computing resources and storage will be available for allocation to CMS wide physics groups working on the data samples stored at that particular center. The center must be able to provide accounts for all CMS physicists working on those data samples independent of geographic location. Finally, some remaining portion of the resources will be reserved for the needs of local physics communities served by this center. This includes needs of associated Tier II and local centers which depend on the Tier I to provide data samples and computing cycles. The centers will also need to provide testing and develolpment facilities for CMS physics software development. The assignment of tasks and allocation of resources between these three portions will be negotiated on a regular basis between the center and the CMS CCS project.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	Sufficient staff must be in place at the center to operate the facility and ensure resource usage is consistent with priorities set by the experiment. This includes staff for tasks including operations, system administration, network operation, database administration, security and storage management. Video conferencing and other remote cyber presence activities must be supported. Some of this infrastructure support may share resources with management of existing facilities, but there will be typically 10 FTEs providing CMS operational support at each Tier I center.

Deliverable 1.2−B	**Software Support**
Description:	The center must provide personnel to aid in the installation and distribution of CMS software products, both at the center itself and in associated Tier II and local centers. Consulting support and help desk facilities should be provided. CMS distributed information resources must be provided and supported. Training in the use of CMS software should be provided.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	These activities will require of order 10 FTEs at each center.

Deliverable 1.2−C	**Software Development**
Description:	Each Tier I center will provide a pool of software developers who will collaborate on the worldwide development of CMS software products, including working with physicists developing physics software. The software professinals at a particular center may take on a specific development task, on their own or in collaboration with other centers, as well as providing ongoing consulting and code reviews for physics software.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	Each center shouold provide a minimum of 5 FTEs for such tasks.

Deliverable 1.2−D	**Computing System Research and Development**
Description:	Staff will also participate in the design of new CMS computing systems and evaluation of new technology, and collaborate on advanced computing R&D needed for future CMS computing systems. The needs of the regional centers must be included in architectural decisions, and the centers need to participate in developing solutionsto the unprecedented computing challenges presented by the volume and complexity of the CMS data samples.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	Typically 5 FTEs at each center will participate in these activities.

1.3 Tier 2 Regional Computing Centres (*Coordinator name*)

Typically, these are Institute / local community servers, providing attended services only during working hours (local time), and run un- attended otherwise. Optimised for analysis/chaotic access, MC work in spare time. ...and more... *blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit*

Deliverable 1.3−A	*Name of deliverable here.*
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

1.4 Wide Area Networks (*Coordinator name*)

blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit

Deliverable 1.4−A	*Name of deliverable here.*
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

2 General CMS Computing and Software Services *(Werner Jank)*

This is supposed to cover the fundamental computing and software infrastructure required for any collaboration. Many, but certainly not all, of the services covered are closely related to those provided in a general way at CERN by the CERN/IT division.

2.1 General Computing Facilities *(Coordinator name)*

....by which I mean the CMS clusters, work group servers,..

Deliverable 2.1 – A	**CMS Computing Environment**
Description:	hepix, setenv, ...
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 2.1 – B	**Monitoring, statistics, usage, performance, errors logging,..**
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 2.1 – C	**Job submission, queue setup + configuration, ...**
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 2.1 – D	**Data (pre)staging, distribution, replication, ...**
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 2.1 – E	**Data location search, browsing (meta data), ...**
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 2.1 – F	*...and more deliverable...*
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

2.2 Operating System Support and System Administration *(Coordinator name)*

....of the CMS clusters, work group servers,..

Deliverable 2.2 – A	*xxx*
Description:	hepix, setenv, ...
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

2.3 Local Area Network *(Coordinator name)*

....of the CMS clusters, work group servers,..

Deliverable 2.3 – A	*xxx*
Description:	hepix, setenv, ...
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

2.4 Information Systems *(Coordinator name)*

This item covers the construction, operation, and maintenance of the collaboration information systems.

Deliverable 2.4 – A	**Deployment and Operation of WWW Server**
Description:	Deployment and operation of several kind of www-servers. Outreach (cmsinfo) Workhorse (cmsdoc) Special, secure (FB, MB, SC, ..) Project home-page support, web-based download, cgi-scripts, servelets, ftp server, ...
Responsible:	*Responsible person*
Client/Sponsor:	Collaboration and general public.
Completion date:	On-going.
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 2.4 – B	**Deployment and Maintenance of Collaboration Database System**
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 2.4 – C	**System for Storing Documents (Technical Notes, etc.)**
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 2.4–D	Development and Maintenance of CCS WWW pages
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 2.4–E	Development and Maintenance of PRS WWW pages
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 2.4–F	Development and Maintenance of Online software WWW pages
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

2.5 Communication Systems *(Coordinator name)*

This item covers the construction, operation, and maintenance of the collaboration information systems.

Deliverable 2.5–A	E-mail systems including Lists
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 2.5–B	News System
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 2.5–C	Calendar System
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 2.5–D	Collaborative Working Tools
Description:	This task includes (possibly) development and (definitely) support of network-based collaborative working tools such as videoconferencing, which are invaluable to the highly distributed members of any HEP collaboration.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

2.6 Problem reporting system *(Responsible person)*

Problem Resolution ensures that problems reported are actually resolved in a well understood manner, and that trends or problematic areas are recognised. The work necessary to ensure that all discovered problems are analysed and fixed will be identified. Reports will be created when problems are found in a piece of software or an activity, and a mechanism of identifying and acting on trends in problems will be established.

Deliverable 2.6–A	Problem reporting system
Description:	Establish and deploy a problem reporting system, that allows to ensure problems can be reported, described, noted, analysed, re-directed, traced, etc..
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	Useful, once project areas and base-lined products or processes are available.

Deliverable 2.6–B	Maintained action list
Description:	Take note, prioritise problems, initiate actions, and track them to closure.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	Should be there now, independent of a formal problem reporting.

Deliverable 2.6–C	Closed problem
Description:	Review the solutions to problems as they come in, and distribute the corresponding software, documentation, etc..
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	Assumes that we have any problems.

Deliverable 2.6–D	Trend analysis
Description:	Collect and analyse fault occurrences, frequency, correlation with coding habits, etc., and feed results back into the relevant processes.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	Assumes the above is in active use for some time.

3 Architecture, Frameworks and Toolkits *(Vincenzo Innocente)*

This task covers the implementation, support and documentation of the software foundation on which the applications are built on. The three parts of the foundation are the architecture, the frameworks and the toolkits.

The architecture describes the model or blueprint of the key components of the system: what they are, how they behave, how they interact and so forth. The model is implemented through a main framework and several auxiliary frameworks for specific tasks. Examples of the latter are frameworks for geometry database, visualisation or calibrations. Finally, toolkits define a common language for all the software as highly reusable classes for containers, physics quantities, algorithms and so forth.

3.1 Software Architecture *(Coordinator name)*

This item covers the definition and evolution of the model for the software systems. It defines the key components, including what they are, how they behave, how they interact with each other, what their design is, and how they are partitioned physically. The item also includes documentation on how the architecture was derived: the most important use cases and requirements, analyses of the problem domain, and key decisions made.

Deliverable 3.1−A	**Tools to Create and Manage Architecture Document Views**
Description:	Tools to document the many aspects of the software architecture. They provide a means to generate several consistent documents from a pool of information describing the architecture. That is, each produced document is a view to some aspect of the architecture. The tools will guarantee that different documents are consistent when they have drawn from the same pool originally. They also hyperlink the generated documents as one web.
Responsible:	*Responsible person*
Client/Sponsor:	*Architecture document authors, PRS groups?*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 3.1−B	**Top-Level Architectural Description**
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	It should be published as a CMS internal note.

Deliverable 3.1−C	**Top-Level Core Framework Description**
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	It should be published as a CMS internal note.

Deliverable 3.1−D	**Top-Level Auxiliary Framework Descriptions**
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	It should be published as a CMS internal note.

Deliverable 3.1−E	**Background Document X (General model, Database, Online, Grid, Detector description, Visualisation, Analysis, ...)**
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	It should be published as a CMS internal note.

3.2 Software Framework *(Coordinator name)*

The foundation of the CMS software is a professionally engineered framework into which physics modules may be inserted. the framework should support standard operations performed by physicists, including: accessing the CMS data store; selection and classification of events; the ability to apply calibration algorithms; invocation of standard or custom-built reconstruction algorithms; interactive visualisation and physics analysis systems with presentation-quality output; and creation of user-defined transient or persistent objects. The user of the framework will be able to choose what specific actions to perform and which specific algorithms to use even at run-time or interactively.

Deliverable 3.2−A	**Framework Design Document**
Description:	*Elaborate on purpose and functionality.* NB: This is similar to the "Description" items in the architecture section. Essentially the "Description" and "Design" documents are two different things: the former is a top level design, this one here is a complete design. Whether we want to specify that we want both I (= Lassi) am not sure, getting even one of them would be fabulous.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	It should be published as a CMS internal note.

Deliverable 3.2−B	**Framework for use by Reconstruction Software**
Description:	The current version of the "CMS Analysis and Reconstruction Framework" known as CARF, provides the user with an intuitive and flexible means of performing analysis and reconstruction tasks. CARF uses an ODBMS to store objects persistently and will provide classes and methods to shield end-users from details of the ODBMS storage and access mechanisms, mostly in terms of C++ smart-references and smart-iterators.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	Already delivered.
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 3.2−C	**Common Framework for use by both Simulation and Reconstruction Software**
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 3.2–D	**Creation of an Independent Releasable Unit for the Framework**
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

3.3 Auxiliary Software Frameworks (*Coordinator name*)

Placeholder for other frameworks we will need.

Deliverable 3.3–A	*Auxiliary Framework for xxx*
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 3.3–B	*Auxiliary Framework for yyy*
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

3.4 Toolkits (*Coordinator name*)

The toolkits needed by the project.

Deliverable 3.4–A	**Basic class libraries**
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 3.4–B	*Specific toolkit xxx (repeat for e.g. XML, G4, ...)*
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

3.5 Integration of Frameworks and Grid Services (*Coordinator name*)

blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit

Deliverable 3.5–A	*Name of deliverable here.*
Description:	Support and deliver tools and infrastructure to allow efficient use of CMS worldwide computing resources.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 3.5–B	*Name of deliverable here.*
Description:	Tools will be created for database replication and synchronisation between production sites, remote job submission, system monitoring, and load balancing.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

3.6 Interactive Graphics Toolkits (*Coordinator name*)

For the moment, this means the "Interactive Graphical User Analysis (IGUANA)" software. The IGUANA software project addresses interactive visualisation software needs for three domains:

- graphical user interfaces (GUI's);
- interactive detector and event visualisation; and
- interactive data analysis and presentation;

for use in a variety of areas such as offline simulation and reconstruction, data analysis, and test beams. Tasks include the assessment of use-cases and requirements and the evaluation, integration, adaptation, verification, deployment, and support in the CMS environment of visualisation software from HEP, academia, the public domain, and the commercial sector.

Pre-existing software is exploited as much as possible to optimise the use of the resources available. Therefore, a significant component of this task will be the assessment of available tools and their adaptation, extension, deployment, and support in the CMS environment.

Deliverable 3.6–A	**Document describing the use cases and scenarios for interactive analysis**
Description:	In order to ensure delivery of anything useful from IGUANA it is imperative to re-evaluate the mission and priorities of IGUANA, following the delivery of the "Functional Prototype" software in October 2000 (CMS Note IN 2000/052). The use-cases and scenarios should come from the target IGUANA clients who ultimately should include the whole collaboration. For this deliverable it is assumed that the use-cases and scenarios will be collected by the Cafétask force from a variety of sources especially people involved in the CCS, RPROM, SPROM, and the PRS activities.
Responsible:	Cafétask force.
Client/Sponsor:	IGUANA developers.
Completion date:	April 2001 *(check this with Café!)*
Constraints:	The document describing the use cases and scenarios should be approved by their sponsors and the CPT project management. It should be published as a CMS internal note.

Deliverable 3.6–B	**Requirements document derived from on an analysis of the use-cases and scenarios**
Description:	The requirements should be specified as derived from the use-cases and scenarios. Additional assumptions and constraints, if any, should be clearly described. The document should arrange the requirements according to natural sub-tasks of IGUANA and ensure that the boundaries between IGUANA and other projects are clear.
Responsible:	IGUANA coordinator.
Client/Sponsor:	IGUANA developers.
Completion date:	Approximately two months after the use-cases and scenarios have been completed and agreed upon by their respective sponsors, i.e. June 2001.
Constraints:	The requirements document should be approved by the use-case sponsors and the CPT project management. It should be published as a CMS internal note.

Deliverable 3.6–C	**IGUANA Repository using the CMS-standard Configuration Management System**
Description:	This item consists of a coherent Software Infrastructure including a software repository, multi-platform build, release, distribution, and documentation systems, and regular public versioned releases of both software and documentation. All aspects of this infrastructure should exploit common CMS tools and procedures as much as possible.
Responsible:	IGUANA coordinator.
Client/Sponsor:	IGUANA developers.
Completion date:	Already established. Support is an ongoing activity.
Constraints:	This is absolutely required. The maintenance of this infrastructure requires ongoing effort from qualified personnel.

Deliverable 3.6–D	**Evaluation of Technologies and Choice of a Baseline Set of Products**
Description:	The evaluations should be performed taking into account the most critical requirements. Naturally they will exploit work already carried out for the IGUANA "Functional Prototype" milestone which was completed in October 2000.
Responsible:	IGUANA coordinator.
Client/Sponsor:	IGUANA developers.
Completion date:	Approximately *???* months after the completion of the requirements document.
Constraints:	The evaluation results and the baseline choice of technologies should be documented in a CMS internal note. The baseline choice should be approved by the CPT project management.

Deliverable 3.6–E	**Integration, Maintenance and Upgrades of non-CMS Software used by IGUANA**
Description:	The philosophy of IGUANA is to exploit non-CMS software wherever possible, including Anaphe, HEP-wide, public-domain, and commercial components (incidentally these number almost fifty for all CMS projects). While this strategy saves development time it does imply the need for non-trivial work to integrate and upgrade the external software.
Responsible:	IGUANA coordinator.
Client/Sponsor:	IGUANA developers and users.
Completion date:	There is already a full suite integrated and working. Support is continuously required as products evolve is ongoing.
Constraints:	The task is also required for other CMS projects (which use e.g. the same compilers, database, etc.) and should therefore probably be centrally coordinated.

Deliverable 3.6–F	**Generic Software for Event Display**
Description:	*Elaborate on purpose and functionality.*
Responsible:	IGUANA coordinator.
Client/Sponsor:	RPROM and PRS groups.
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 3.6–G	**Detector Description Browsing and Visualisation**
Description:	*Elaborate on purpose and functionality.*
Responsible:	IGUANA Coordinator.
Client/Sponsor:	Detector description, SPROM, RPROM, and PRS groups.
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	It is assumed that all the detector-specific work is carried out by the sponsors.

Deliverable 3.6–H	**Event Collection Browsing and Manipulation Tools**
Description:	This task provides, in conjunction with the CMS framework, interactive tools which facilitate the users interaction and manipulation of event collections. In particular, they should support: browsing of meta-data associated to event collections; the creation of user collections; the creation of deep or shallow copies; and support for miscellaneous database administration tasks such as cloning, cleaning and deleting of events/collections.
Responsible:	IGUANA coordinator.
Client/Sponsor:	Users of CMS event collections, particularly the RPROM and PRS groups and probably the production teams.
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 3.6–I	**Deployment of a GUI builder**
Description:	Downloading of a pre-existing GUI builder (e.g. QtArchitect), building it on standard CMS platforms, and installing it in a generally accessible place.
Responsible:	IGUANA coordinator.
Client/Sponsor:	IGUANA developers and possibly developers of graphics applications which use the same GUI tools as IGUANA.
Completion date:	We should just do it...it only takes a day or so...
Constraints:	None.

3.7 Detector Description *(Coordinator name)*

This task aims to provide an environment for creating, manipulating, and using the parameters describing the CMS detector in a consistent manner. In particular, it covers the geometrical description of the detector elements at various levels (full engineering detail, full GEANT detail, fast simulation, trigger tower geometries, etc.), associated material properties, magnetic field map, etc. The Detector Description sub-system will serve a number of clients, including OSCAR, the Fast Simulation, ORCA, the Calibration, and the User Analysis Environment. Deliver tools to encapsulate detector description at the various degrees of precision required for CMS software tasks.

Provide an environment for creating manipulating, and using parameters describing the CMS detector in a consistent manner.

Geometry description of the detector elements at various levels (full engineering detail, full GEANT detail, fast simulation, trigger tower geometries, etc.), associated material properties, and magnetic field map.

The sub-system will serve a number of clients including OSCAR, Fast Simulation, Calibration, and User Analysis Environment.

Deliverable 3.7−A	Central Repository of Detector Description Data
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 3.7−B	Interfaces to Engineering Descriptions of Data
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 3.7−C	Interfaces to Simulation, Reconstruction, and Analysis Software
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 3.7−D	Magnetic Field Description
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 3.7−E	GUI Widgets pool
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

3.8 Technology Tracking and Evaluation *(Coordinator name)*

Subtask Description here...what it aims to achieve and key functionality... should be easy to check if achieved or not Should include written overviews on the selected and critical points.

Deliverable 3.8−A	Computing and Networking Performance
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 3.8−B	Databases and Mass Storage Technologies
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 3.8−C	Graphics Hardware and Software
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 3.8−D	OS's and compilers
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 3.8−E	Yet unknown products if it's relevant
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

4 Software Users and Developers Environment *(Stephan Wynhoff Ad-interim)*

This item covers development and support for the CMS software and environment including the associated code and configuration management systems, version control mechanisms, change control and the code release, distribution, and build procedures. An important aspect of this task is the porting, technical verification, and support of CMS and external code on multiple systems.

4.1 Configuration Management *(Responsible person)*

The goal of configuration management is to establish and maintain the integrity of the software and other work-products of our projects and processes. A strategy for configuration management will be defined, all items to be put under configuration management will be identified, agreed on and base-lined, modifications and releases will be controlled, the status of work-products will be noted and made retrievable, completeness and consistency will be ensured, and storage and retrieval will be controlled.

Deliverable 4.1 – A	**Configuration management strategy**
Description:	Write up a configuration management strategy, which includes all activities of configuration management, and a schedule for performing them.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 4.1 – B	**Configuration management system**
Description:	Establish a system for configuration management, that includes configuration item storage and retrival, versioning, code and parameter organisation, build of libraries and executables, any standards and procedures to be used, and the supporting tools.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Support all supported platforms and architectures; Support a distributed development model.*

Deliverable 4.1 – C	**Problem reporting system**
Description:	Establish and deploy a problem reporting system, that allows to ensure problems can be reported, described, noted, analysed, re-directed, traced, etc., and is linked in with configuration management.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	Useful, once project areas and base-lined products or processes are available.

Deliverable 4.1 – D	**Configuration description**
Description:	Write and maintain an up to date list of all configuration items, including their eventual decomposition into lower level configuration items, who is responsible for each item, and when it is to be placed under configuration management.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 4.1 – E	**Configuration history**
Description:	Ensure the maintenance of each configuration item in sufficient detail that previous base-lines can be recovered, when required.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 4.1 – F	**Configuration status report**
Description:	Make available the status of each configuration item, like external library, or some package library and headers, and their relationships in the current or base-lined system state.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 4.1 – G	**Released and distributed software**
Description:	Control and execute the storage, handling, releasing/base-lining and distribution of the items under configuration management; use a sub-set of the system tests for acceptance testing of configuration items at a defined level of granularity.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 4.1 – H	**Preparation of releases and installation at CERN**
Description:	This task covers the preparation and installation of the CMS software in the central release area in various combinations of optimised, shared, debug libraries. It requires familiarity with external software, coordination and collection of consistent subsystem/project developers tags. The complexity of the software multiplies by the number of various components, subsystems, and projects that are used in a single system or the system relies on. The installation and validation of the final release.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	Ongoing
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 4.1 – I	**Integration testing**
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 4.1 – J	**Automatic test procedure (using examples) for new releases**
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 4.1–K	**Release installation tool**
Description:	a la TCL/TK interface on ASIS to facilitate scram setup during the release.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 4.1–L	**Nightly builds**
Description:	As a part of the release process the nightly builds with the proper log book keeping, building test programs and run them to detect critical points as soon as possible.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	The developers of the CMS software would be obliged to provide those tests for their code.

Deliverable 4.1–M	**Header File Checking Tool**
Description:	A tool (e.g. integrated with scram build command) to go over every header file, and for each one produce a .cxx file that includes just that header (and Architecture.h?). It would be used to check whether that header is stand-alone. Output could be of the form: X/Y/interface/Z.h: PASS in the event of success, or: X/Y/interface/Z.h: FAIL <compiler error messages> in the event of failure.
Responsible:	*Responsible person*
Client/Sponsor:	All software developers.
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

4.2 External Software Support (*Coordinator name*)

This task covers the support associated to the use of non-CMS (HEP, public-domain, and commercial) software in the CMS environment. Examples of such software include: database software; graphics packages; documentation systems; design and code management tools; parsers; GUI builder; and so on.

Deliverable 4.2–A	**Software Repository for non-CMS software using the CMS-standard Configuration Management System**
Description:	This item consists of a coherent Software Infrastructure for handling non-CMS software. It includes a software repository, multi-platform build, release, distribution, and documentation systems, and regular public versioned releases of both software and documentation. All aspects of this infrastructure should exploit common CMS tools and procedures as much as possible. In some cases the build, release and distribution mechanisms of other projects (e.g. CERN/IT) will be used but the configuration information is still required.
Responsible:	*Name of deliverable here.*
Client/Sponsor:	*Responsible person*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	The maintenance of this infrastructure requires ongoing effort from qualified personnel.

4.3 Software Performance and Optimisation (*Coordinator name*)

Subtask Description here...what it aims to achieve and key functionality... should be easy to check if achieved or not *blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit*.

Deliverable 4.3–A	**Software profiling system**
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 4.3–B	**Benchmark definition**
Description:	Define benchmarks for measurements to be taken. This can for example be performance measures done in "typical" running conditions, or the time it takes to do a concrete software release.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 4.3–C	**Benchmark report**
Description:	Perform the benchmark on processes and products at pre-defined check-points, and use normal improvement practices to narrow any performance gap.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

4.4 User Applications and Support (*Coordinator name*)

Subtask Description here...what it aims to achieve and key functionality... should be easy to check if achieved or not *blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit*.

Deliverable 4.4–A	**Training**
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 4.4–B	**Example Programs**
Description:	Presentation examples with external packages (ANAPHE).
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 4.4−C	**Helpdesk**
Description:	Presentation examples with external packages (ANAPHE).
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

4.5 User Documentation *(Coordinator name)*

The purpose of documentation is to develop and maintain documents that record information produced by a process or activity. It includes producing a strategy for identification and maintenance of the documents to be produced during the life-cycle of the software, identification of the standards to be used for the development of the documentation, the identification of all documents to be written in a project or when performing a process, a specification review and eventual approval of the contents and purpose of these documents, the verification that documents are produced and maintained in accordance with the specified standards/templates. Note that this is very multifaceted, and can concern many areas. If documentation is not to diverge, focus must be given by very clearly defining the documents to be produced, with an eye on their maintainability.

Deliverable 4.5−A	**Documentation policy**
Description:	Define and agree on what should be documented where and when during the entire life-cycle of the software.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	Needs more than rudimentary understanding of our needs for documentation.

Deliverable 4.5−B	**Document templates**
Description:	Produce a set of document templates for the agreed on documents.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	Documentation policy and documentation infrastructure available.

Deliverable 4.5−C	**Documentation infrastructure**
Description:	Establish infrastructure and standard procedures for writing changing and maintaining documentation, including source code, design, or architecture documents.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	Documentation policy available.

Deliverable 4.5−D	**Verify documentation**
Description:	Review the documentation before releasing to general public.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	Documentation available.

Deliverable 4.5−E	**Distribute documentation**
Description:	Distribute the documentation, and verify availability.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	Documentation released.

Deliverable 4.5−F	**Up to date documentation**
Description:	Maintain documentation according to the defined procedures.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	Software changes in ways relevant for the documentation.

5 Software Process and Quality *(Johannes Peter Wellisch)*

This is basically the scope of GPI....we should think about what name is clearest to the largest number of people... *Task Description here*...what it aims to achieve and key functionality... should be easy to check if achieved or not *blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit* .

5.1 Software Process *(Responsible person)*

This item covers the process associated with the definition, design, development,documentation, integration, verification, deployment, and maintenance of the CMS Software with the aim of improving efficiency and quality. The pragmatic "Cyclic Life Cycle" model has been adopted by CMS that emphasises continuous improvement following ISO/IEC 15504 (SPICE). Items which have been implemented include the structure of the ORCA software repository and the strategy for ORCA development, release, and testing and the CMS coding rules, guidelines, and style. Processes currently under construction and only partly deployed include the more comprehensive assessment of user requirements, checking of software dependencies between software sub-systems and packages, automated checking of coding rule and style conformance, and more comprehensive integration test suites and examples. Processes to be implemented more rigorously in future include: more formal and uniform software requirements analysis and design procedures; documentation of code design, implementation, and usage; and problem reporting, tracking and resolution mechanisms. Deliver tools and methodologies to encourage and support efficient software management and quality assurance. The task here in more general terms is to establish a set of organisational processes for all software life-cycle activities, based on organisational alignment with the goals of our experiment, to assess in how far the different areas are indeed relevant to CMS and contributing to our success, and to improve the important processes continuously to the level where effort and gain are judged to be balanced in view of our goal of taking data doing great physics.

Deliverable 5.1 – A	**Mission statements**
Description:	Define for each process instance invoked the goals the are to be achieved by following the process, and their relation to the goals of CMS as a whole. Include also examples of consequences of non-compliance
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 5.1 – B	**Process scopes**
Description:	Identify for each process instance invoked the activities, roles, authorities and responsibilities. These then are to be ratified and enforced by project management.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 5.1 – C	**Process descriptions**
Description:	Define and document the processes performed in CMS, including for each process instance invoked the input and output work products, entry and exit criteria, check-points, external interfaces with related processes that produce inputs and/or need the results, internal dependencies between the process activities, and metrics and measures that allow to demonstrate the achievement of the process.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 5.1 – D	**Tailoring guidelines**
Description:	Establish tailoring guidelines that describe how to optimise a specific process for individual sub-projects. Note that active but restricted contribution of the process/environment team to development at all levels is necessary to ensure the validity of the approach.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 5.1 – E	**Performance expectation document**
Description:	Establish a document describing the performance expectations when using the process defined in the process description.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 5.1 – F	**Deployed process**
Description:	Deploy the defined and ratified processes throughout CMS, as far as applicable, and verify deployment at established check-points using assessment methods.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 5.1 – G	**Maintained process definition**
Description:	Maintain the process description documents.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 5.1 – H	**Improvement opportunity report (which would typically be a presentation)**
Description:	Identify possibilities for improvement, and define the scope of the improvement activity.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 5.1 – I	**Assessment record**
Description:	Define the method to be used for process assessments, and the goals of the assessment. Include the assessment inputs and the plan for the assessment. Collect relevant information and validate this information. Identify strength and weaknesses. Maintain, report, and exploit the assessment results
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	Improvement activity is scoped.

Deliverable 5.1 – J	**Impact analysis**
Description:	Define the priorities of improvement, and define measures for the impact of the improvement. Deploy the new process as prototype in one area, and confirm the improvement.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 5.1 – K	**New deployed process**
Description:	Update the process definition, and deploy the new process across the board.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

5.2 Measurement, Quality Management, and Quality Assurance (*Responsible person*)

The goal of the measurement process is to collect and analyse data related to the software architecture, design, implementation, testing, etc. in order to support the management of the individual activities. It also allows to objectively demonstrate the quality of the software. It includes the definition of an appropriate set of measurements, and the collection and analysis of the corresponding data. It also includes the distribution of these results to all people involved or interested.

The purpose of this activity is to provide assurance that the design, software, documentation, and other work products and processes comply with the specified requirements and stick to their established plans. This includes producing implementing and maintaining a strategy for performing the quality assurance activities, and recording evidence of their performance. problems/bugs or non-conformance will be found, and the correct application of any standards, procedures as well as fulfilment of requirements will be verified objectively. To be unbiased, the quality assurance activity must act independently from the direct responsible for developing the software.

The goal of Quality Management is to make sure that the software satisfies the people using it. It needs the establishment of a focus on monitoring the quality of products and processes. Quality goals will be written down based on what we need, including implicit and explicitly stated requirements, for a set of check-points within the project life-cycle. An overall strategy will be defined to achieve these stated goals, and identified quality control and assurance activities will be performed and their performance will be confirmed. The actual state will be monitored against the goals, and appropriate action will be taken then the goals are not achieved.

Deliverable 5.2 – A	**Quality assurance: Strategy**
Description:	Write and maintain a paper specifying the scope of quality assurance and the responsibilities for quality.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	Basic technological decisions have been taken.

Deliverable 5.2 – B	**Quality assurance: standards**
Description:	Establish a set of standard quality criteria for the relevant processes and work products. Establish for each item in the quality standard fit-criteria.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	Needs the policy statement.

Deliverable 5.2 – C	**Quality assurance: record**
Description:	Perform a set of tests that establish at the needed level of confidence that the processes and work-products (software, designs, documentation, etc.) are according to the agreed-on quality standards.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	Needs the Quality standards at least in rudimentary form.

Deliverable 5.2 – D	**Quality assurance: Closed problem report**
Description:	Report, analyse, correct etc. any problems found, and trace the corresponding actions to closure.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	Quality record available.

Deliverable 5.2 – E	**Quality Management: Quality goals**
Description:	Based on what we need in terms of quality in the various operational environments, establish quality goals for the products throughout the project. Make sure these can be checked in a quantitative manner.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 5.2 – F	**Quality Management: Overall quality strategy**
Description:	Establish an overall strategy at the organisational and project level that allows to achieve the defined goals. Define metrics that will be used to measure the results of the various project activities, and by defining acceptance criteria at a course grained level, that will help to understand if the documented quality goals have been achieved.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	Quality goals identified.

Deliverable 5.2 – G	**Quality Management: Quality activity specification**
Description:	For each goal, identify quality control and assurance activities, which will help to achieve the goals set at the organisational and project level. Note that the hand-shake with quality assurance is important, as these activities have to be integrated into the software development.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	Overall quality strategy available.

Deliverable 5.2−H	**Quality Management: Quality assessment record**
Description:	Assess the quality throughout the project at pre-defined check-points, applying the specified quality metrics to assess if the project is on track to meat the goals specified.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	Quality activity specification available.

Deliverable 5.2−I	**Quality Management: Corrective action**
Description:	In case high level quality goals have not been met, take corrective or preventive action at the project and organisational level, where appropriate.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 5.2−J	**Example Standard: Coding rules and style guidelines**
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 5.2−K	**Example infrastructure: Tool for checking style rules**
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 5.2−L	**Example infrastructure: Tool for determining CMS and external software dependencies and metrics**
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 5.2−M	**Example infrastructure: Tool for analysing and displaying dependency and metric data**
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 5.2−N	**Metrics analysis paper**
Description:	Establish measurable quantities for process and product measurement and make sure data can be and are collected and analysed against the metrics. An example would be essential cyclomatic complexity for code quality, number of pre-releases during release sequence for code organisation, or the result of path coverage analysis for testing.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 5.2−O	**Measurement record**
Description:	Do the measurement on a base-lined work-product or process, and the above described metrics, benchmarks or statistical data. For example coding rule analysis of source code, using our rules standard, or a dependency diagram of released code. Make the measurement data available for decisions where it is relevant.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

5.3 Verification (*Responsible person*)

Verification confirms that piece of software, a process or a project reflects the specified requirements. It includes a verification strategy, verification criteria for the individual software work products, performance of the verification activity, ensuring that identified faults/defects/bugs can and will be found and fixed, and that any results of the activity can/will be made available to all people concerned.

Deliverable 5.3−A	**Verification strategy**
Description:	Provide a document specifying the criteria by which software, documentation, etc. will be verified, and which work-products are to be verified.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	Assumes that we need actually this.

Deliverable 5.3−B	**Verification results**
Description:	Perform verification, recording the results as needed, and specified in the corresponding strategy paper.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	Verification strategy was made available, and we actually want to do this.

Deliverable 5.3−C	**Closed action item**
Description:	Track initiated corrective action to closure, and maintain action list.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	Action list available.

5.4 Validation *(Responsible person)*

Validation or productization confirms that the software is fit for a specific intended use. It includes the production and implementation of a validation strategy for each intended use considered, the identification of validation criteria for each piece of software, the performance of the validation activity, the resolution of the problems identified, providing evidence that the software is fit for a specific use (for example trigger), and that any results of the activity can/will be made available to all people concerned.

Deliverable 5.4–A	**Validation strategy**
Description:	Write down a validation strategy paper, that specifies the use-cases and operational environments and software/system concerned, along with validation criteria and fit-conditions for all items involved.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	Can start now, but will have to turn into an ongoing activity.

Deliverable 5.4–B	**Validated software/system**
Description:	Identify the techniques processes, test-cases etc. where not specified in the validation strategy, and perform the validation tests, ex when validating code for the trigger environment. Record validation results and fault history.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	Validation strategy available.

Deliverable 5.4–C	**List of action items**
Description:	Plan and initiate corrective action where needed, based on validation result analysis.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	Validation results available.

Deliverable 5.4–D	**Closed action item**
Description:	Track corrective actions to closure, and maintain list of action items.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	Tests for system partitions available.

Deliverable 5.4–E	**Documented set of coding rules and style guidelines**
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 5.4–F	**Tool for checking coding rules**
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 5.4–G	**Tool for determining CMS and external software dependencies and metrics**
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 5.4–H	**Tool for analysing and displaying dependency and metric data**
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 5.4–I	**Integration testing**
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 5.4–J	**Automatic test procedure (using examples) for new releases**
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 5.4–K	**Change control procedure and tools**
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 5.4–L	**Metrics analysis paper**
Description:	Establish measurable quantities for process and product measurement and make sure data can be and are collected and analysed against the metrics. An example would be essential cyclomatic complexity for code quality, number of pre-releases during release sequence for code organisation, or the result of path coverage analysis for testing.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 5.4–M	**Measurement record**
Description:	Do the measurement on a base-lined work-product or process, and the above described metrics, benchmarks or statistical data. For example coding rule analysis of source code, using our rules standard, or a dependency diagram of released code. Make the measurement data available for decisions where it is relevant.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 5.4–N	**Benchmark definition**
Description:	Define benchmarks for measurements to be taken. This can for example be performance measures done in "typical" running conditions, or the time it takes to do a concrete software release.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 5.4–O	**Benchmark report**
Description:	Perform the benchmark on processes and products at pre-defined check-points, and use normal improvement practices to narrow any performance gap.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

5.5 Software Re-use (*Coordinator name*)

The purpose is to promote and ease the use of new and/or existing software from both the people and the product perspective. This includes the definition of re-use strategies, and identification of activities and infrastructure needed to facilitate and perform re-use, as well as their establishment and maintenance. This task in strongly coupled with configuration management and system integration.

Deliverable 5.5–A	**Re-use strategy**
Description:	Identify, develop and apply re-usable entities at all organisational levels, with the goal to improve productivity and quality.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 5.5–B	**Re-use library**
Description:	Establish a re-use library for collection, categorisation, management, update, etc. of re-usable entities.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 5.5–C	**Re-use planning**
Description:	Identify and document reusable entities such as code modules, tests, interfaces,components, documents, infrastructure, processes, etc...
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 5.5–D	**Re-usable item**
Description:	Design and develop re-usable entities, and keep them stable and consistent. Maintain consistency stability and standardisation of re-usable entities in the re-use library (like magnetic field between simulation and reconstruction).
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 5.5–E	**Information session**
Description:	Inform potential users about the existence of re-usable entities, including features and possible restrictions or problematic areas.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 5.5–F	**Technology report**
Description:	Evaluation of new technologies, using prototyping, with the goal of finding better arrangements for the main risks, and/or removing technology related constraints from software and system.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	Assumes that the system constraints and architecturally relevant use-cases available; main risks are understood and currently planned arrangements are documented. Link between system and software constraints and technology understood and documented.

Deliverable 5.5–G	**Database access utilities**
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

5.6 System Integration *(Coordinator name)*

This item covers the development of a system integration test strategy for integrating system units consistent with the release strategy, and for testing the system in the different operational environments. It includes the development of a strategy to re-test (regression testing) system aggregates, in case of a change in a component of the system, and identifying aggregates of system units and a sequence and order in which to test them. This will be done consistent with the system architecture and release strategy. It includes the development of tests for the identified system aggregates, that ensure that the given part of the system satisfies requirements, running the tests, and the documentation of the test results. Each test to be run for a system aggregate will be described, indicating the requirements being checked, input data, system components needed to perform the test, and validation criteria.

Deliverable 5.6−A	**System integration test strategy**
Description:	Develop the strategy for integrating system units and for testing the system. This shall be done consistent with the release strategy.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	Assumes that release strategy and system architecture are available.

Deliverable 5.6−B	**System regression test strategy**
Description:	Develop a strategy for re-testing the system or parts of it, should a change be made.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	System integration test strategy available.

Deliverable 5.6−C	**List of system partitions**
Description:	Build a list of aggregates of system units that can be separately tested. Identify the aggregates, and define a sequence in which to test them.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	System integration test strategy, release strategy and system architecture available.

Deliverable 5.6−D	**Tests for system partitions**
Description:	Describe the tests to be run against each partition of the system. Include the requirements to be checked, input data, fit criteria, and a list of system components and supporting infrastructure needed for the tests.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	List of system partitions available.

Deliverable 5.6−E	**Tested system partitions**
Description:	Test each individual system partition and make sure that it satisfies the specified fit criteria. Write up and storage of the results is mandatory.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	Tests for system partitions available.

Deliverable 5.6−F	**System tests**
Description:	Develop and describe the tests to be run on the integrated system. Specify use-cases and/or requirements checked, input data and fit criteria.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	system architecture is available.

Deliverable 5.6−G	**Tested system**
Description:	Integrate the system, and perform above tests on the integrated system. Make sure eventual faults are corrected, and document all results, including fault history.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	Tested system partitions and system tests available.

6 Production Processing and Data Management *(Tony Wildish)*

xxx

6.1 Production Tools *(Coordinator name)*

Delivery and maintainance of the tools required to manage site-specific and worldwide simulation and reconstruction productions.

Deliverable 6.1–A	**Tools for job generation, submission, authentication, ...**
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 6.1–B	**Tools for data distribution, replication, ...**
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 6.1–C	**Tools for recovery after malfunctions, errors, mistakes, ...**
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 6.1–D	**Tools for monitoring, status, statistics of resources, applications, ...**
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 6.1–E	**Tools for integrating results from distributed productions**
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 6.1–F	*...and more deliverables...*
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

6.2 Production Operations *(Coordinator name)*

Operations of worldwide simulation and reconstruction productions.

Deliverable 6.2–A	**Definition of samples to be produced through liaison with PRS**
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 6.2–B	**Coordination, priorities, distribution (parameters, tasks, ...)**
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 6.2–C	**Validated Samples of Events**
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 6.2–D	**Coordination, priorities, distribution (parameters, tasks, ...)**
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

6.3 Integration of Production Tools and Grid Services *(Coordinator name)*

How related to e.g. production and computing tasks??? Deliver GRID tools and infrastructure to allow efficient use of the CMS worldwide resources. Collaborate with recognised GRID projects to specify and test GRID tool and applications. Wherever possible tools developed by the recognised grid projects (PPDG, GriPhyN, and DataGrid) will be exploited to avoid duplicating effort. It is assumed that the CMS grid activities are primarily integration tasks. If we have to develop everything ourselves, all bets are off.

Deliverable 6.3–A	*Name of deliverable here.*
Description:	Support and deliver tools and infrastructure to allow efficient use of CMS worldwide computing resources.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 6.3–B	*Name of deliverable here.*
Description:	Tools will be created for database replication and synchronisation between production sites, remote job submission, system monitoring, and load balancing.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

6.4 Database Management Tools (*Coordinator name*)

This item covers generic issues pertaining to the use of an ODBMS for the storage of CMS non-event and event data, including those associated to the operation of a database federation in a heterogeneous distributed environment. In addition, a construction database has been developed and deployed as part of the CRISTAL project. Objects are stored persistently, reusing as much as possible of what provided by the ODBMS itself to store, organise and retrieve data, and to administrate both data and meta-data including schema declaration, schema evolution, object versioning and database replication. Optimisation mechanisms such as caching, compression, and physical clustering of objects will be implemented such that they are transparent to the user. A large fraction of the ODBMS activities are carried out in the context of the CRISTAL, GIOD, MONARC, RD45, and WISDOM projects.

Deliverable 6.4–A	**Publish catalogs and contents (local and WAN) and check coherence**
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 6.4–B	**Tools for resource allocation (disk, server etc)**
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 6.4–C	**Tools for optimising performance**
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 6.4–D	**Tools for facilitating common DB administration tasks**
Description:	Creation of new databases, shallow and deep copies, etc.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 6.4–E	**Tools for DB replication, re-clustering, object extraction,..**
Description:	Creation of new databases, shallow and deep copies, etc.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 6.4–F	**Tools for making DB sanity checks, fixing corrupt DB's,...**
Description:	Creation of new databases, shallow and deep copies, etc.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 6.4–G	**Tools for DB serialisation, format-, version-, manufacturer changes**
Description:	Creation of new databases, shallow and deep copies, etc.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

7 B-Tau / Tracker *(M. Mannelli)*

Task Description here...what it aims to achieve and key functionality... should be easy to check if achieved or not *blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit* .

7.1 Tracker Simulation *(Coordinator name)*

Subtask Description here...what it aims to achieve and key functionality... should be easy to check if achieved or not *blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit* .

Deliverable 7.1 – A	*Name of deliverable here.*
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

7.2 Tracker Reconstruction *(Coordinator name)*

Subtask Description here...what it aims to achieve and key functionality... should be easy to check if achieved or not *blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit* .

Deliverable 7.2 – A	*Name of deliverable here.*
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

7.3 Tracker Alignment *(Coordinator name)*

Subtask Description here...what it aims to achieve and key functionality... should be easy to check if achieved or not *blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit* .

Deliverable 7.3 – A	*Name of deliverable here.*
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

7.4 Tracker Data Handling *(Coordinator name)*

Subtask Description here...what it aims to achieve and key functionality... should be easy to check if achieved or not *blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit* .

Deliverable 7.4 – A	*Name of deliverable here.*
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

7.5 B-Tau Tagging *(Coordinator name)*

Subtask Description here...what it aims to achieve and key functionality... should be easy to check if achieved or not *blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit* .

Deliverable 7.5 – A	*Name of deliverable here.*
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

8 E-Gamma / ECAL *(C. Seez)*

Task Description here...what it aims to achieve and key functionality... should be easy to check if achieved or not *blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit* .

8.1 ECAL Simulation *(Coordinator name)*

Subtask Description here...what it aims to achieve and key functionality... should be easy to check if achieved or not *blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit* .

Deliverable 8.1–A	*Name of deliverable here.*
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

8.2 ECAL Reconstruction *(Coordinator name)*

Subtask Description here...what it aims to achieve and key functionality... should be easy to check if achieved or not *blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit* .

Deliverable 8.2–A	*Name of deliverable here.*
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

8.3 ECAL Test Beam and Pre-Calibration *(Coordinator name)*

Subtask Description here...what it aims to achieve and key functionality... should be easy to check if achieved or not *blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit* .

Deliverable 8.3–A	*Name of deliverable here.*
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

8.4 ECAL Calibration *(Coordinator name)*

Subtask Description here...what it aims to achieve and key functionality... should be easy to check if achieved or not *blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit* .

Deliverable 8.4–A	*Name of deliverable here.*
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

8.5 Electron / Photon Higher Level Triggers *(Coordinator name)*

Subtask Description here...what it aims to achieve and key functionality... should be easy to check if achieved or not *blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit* .

Deliverable 8.5–A	*Name of deliverable here.*
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

9 Jets and Missing Energy / HCAL *(S. Eno)*

Task Description here...what it aims to achieve and key functionality... should be easy to check if achieved or not blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit .

9.1 HCAL Simulation *(S. Banerjee)*

Subtask Description here...what it aims to achieve and key functionality... should be easy to check if achieved or not blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit .

Deliverable 9.1 – A	*Name of deliverable here.*
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

9.2 HCAL Reconstruction and Test Beam *(S. Abdullin)*

Subtask Description here...what it aims to achieve and key functionality... should be easy to check if achieved or not blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit .

Deliverable 9.2 – A	*Name of deliverable here.*
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

9.3 HCAL Calibration *(O. Kodolova)*

Subtask Description here...what it aims to achieve and key functionality... should be easy to check if achieved or not blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit .

Deliverable 9.3 – A	*Name of deliverable here.*
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

9.4 Jet/MET Physics Objects and Higher Level Trigger *(Coordinator name)*

Subtask Description here...what it aims to achieve and key functionality... should be easy to check if achieved or not blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit .

Deliverable 9.4 – A	*Name of deliverable here.*
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

10 Muons *(U. Gasparini)*

Task Description here...what it aims to achieve and key functionality... should be easy to check if achieved or not *blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit* .

10.1 Muon Detector Simulation *(Coordinator name)*

Subtask Description here...what it aims to achieve and key functionality... should be easy to check if achieved or not *blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit* .

Deliverable 10.1 − A	*Name of deliverable here.*
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

10.2 Muon Detector Reconstruction *(Coordinator name)*

Subtask Description here...what it aims to achieve and key functionality... should be easy to check if achieved or not *blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit* .

Deliverable 10.2 − A	*Name of deliverable here.*
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

10.3 Muon Detector Alignment, Calibration, and Databases *(Coordinator name)*

Subtask Description here...what it aims to achieve and key functionality... should be easy to check if achieved or not *blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit* .

Deliverable 10.3 − A	*Name of deliverable here.*
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

10.4 Muon Test Beams and Monitoring *(Coordinator name)*

Subtask Description here...what it aims to achieve and key functionality... should be easy to check if achieved or not *blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit* .

Deliverable 10.4 − A	*Name of deliverable here.*
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

10.5 Muon Physics Objects *(Coordinator name)*

Subtask Description here...what it aims to achieve and key functionality... should be easy to check if achieved or not *blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit* .

Deliverable 10.5 − A	*Name of deliverable here.*
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

11 Online Software Infrastructure *(Coordinator name)*

Task Description here...what it aims to achieve and key functionality... should be easy to check if achieved or not *blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit* .

11.1 Online Operations *(Coordinator name)*

Subtask Description here...what it aims to achieve and key functionality... should be easy to check if achieved or not *blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit* .

Deliverable 11.1 – A	**Deliver and validate tools for CMS online farm operations.**
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 11.1 – B	*Name of deliverable here.*
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

12 Café: CMS Architecture Forum for Evaluation *(James Branson)*

Task Description here...what it aims to achieve and key functionality... should be easy to check if achieved or not *blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit* .

12.1 Analysis of Use Cases and Requirements *(Coordinator name)*

Design, validate and support architectures for CMS projects. *blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit* .

Deliverable 12.1 – A	**Documentation of the Existing CMS Architecture**
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 12.1 – B	**Documentation of Critical Use Cases and Requirements**
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

13 GRID *(Coordinator name)*

Deliver GRID tools and infrastructure to allow efficient use of the CMS worldwide resources....and some more... *blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit* .

13.1 Initial Grid Prototypes *(Coordinator name)*

Collaborate with recognised GRID projects to specify and test GRID tool and applications. Wherever possible tools developed by the recognised grid projects (PPDG, GriPhyN, and DataGrid) will be exploited to avoid duplicating effort. It is assumed that the CMS grid activities are primarily integration tasks. If we have to develop everything ourselves, all bets are off.

13.2 Computing Model Simulation *(Coordinator name)*

Deliver the modelling tools to simulate worldwide simulation, reconstruction, analysis and data movement.

Deliverable 13.2 − A	*Name of deliverable here.*
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

14 SPROM: Simulation PROject Management *(Albert de Roeck)*

Task Description here...what it aims to achieve and key functionality... should be easy to check if achieved or not *blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit* .

14.1 Physics Event Generator Infrastructure *(Coordinator name)*

This item covers the software required to support external physics event generator programs in the CMS environment. In particular, it includes software for the management of input parameters in a consistent fashion and procedures for storing output events with a standard format for subsequent use by, for example, OSCAR and the Fast Simulation software. This task does not cover the software of the generator programs themselves.

Deliverable 14.1 − A	**Validated interfaces for third-party event generator programs.**
Description:	Validated interfaces for third-party event generator programs to allow them to be used in CMS Physics studies.
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 14.1 − B	*Name of deliverable here.*
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

14.2 GEANT3-Based Detailed Detector Simulation *(Coordinator name)*

CMSIM is the Fortran-based CMS simulation package. Since new development has essentially ceased, CMSIM will need to be supported for physics studies until GEANT4 and the corresponding OSCAR simulation software have comparable functionality. Maintain while required the Fortran Simulation of the CMS detector. Does not include manpower for detector code changes.

Deliverable 14.2 − A	*Name of deliverable here.*
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 14.2 − B	*Name of deliverable here.*
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

14.3 GEANT4-Based Detailed Detector Simulation (*Coordinator name*)

This item covers the full simulation of particles traversing the CMS detector using the GEANT4 Toolkit, including the use of an appropriate detector description, tuning and validation of physics processes, and simulation of neutron and other backgrounds. OSCAR will provide simulated signal and background events for ORCA. It will also be used to develop and tune parameters and algorithms of the Fast Simulation software.

Deliverable 14.3 – A	*Name of deliverable here.*
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 14.3 – B	*Name of deliverable here.*
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

14.4 Fast Detector Simulation (*Coordinator name*)

This item covers fast simulation of particles traversing the CMS detector and the response of the detector elements, readout electronics, and triggers. Parameterisations will be verified with OSCAR simulations, test-beam analyses, and ultimately real CMS data. The Fast Simulation should use the common Detector Description and be consistent OSCAR, ORCA, and the Physics Object Reconstruction software to permit flexible the use of varying degrees of simulation detail according to the requirements of the user.

Deliverable 14.4 – A	*Name of deliverable here.*
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 14.4 – B	*Name of deliverable here.*
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

15 RPROM Task: Reconstruction *(Stephan Wynhoff)*

The RPROM task force is responsible for the coordination of the development of CMS reconstruction software between the CCS, PRS and Tridas projects. It provides consistent and tested releases of the basic reconstruction code, the high-level reconstruction code for production and physics analysis and the reconstruction code for the higher level trigger on the online farm.

15.1 Basic Reconstruction Software (ORCA of today) (*Responsible person*)

This is ORCA as we know and love it.

Deliverable 15.1 – A	*ORCA*
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	PRS groups as input of high-level reconstruction code, Tridas to run the HLT
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

15.2 Full Reconstruction Software (Integration with physics code) (*Responsible person*)

Mostly configuration, selection of algos from PRS and integration for official production

Deliverable 15.2 – A	*pORCA*
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	Production centers, physics groups, individual physicists
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

15.3 Online Reconstruction Software (Integration with Tridas) (*Responsible person*)

Mostly configuration and integration on the online farm

Deliverable 15.3 – A	*tORCA*
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	Online farm
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

15.4 → PRS sections: Modelling of Detector Response to Simulated Particles and Event Pileup (*Coordinator name*)

In the LHC environment multiple interactions can take place in a single beam crossing due to high luminosity and short bunch spacing. Most interactions are minimum bias events. To enable the flexible re-use of simulated Monte-Carlo events these effects on the response of detector electronics from overlaying events are simulated during reconstruction. Similarly, the selection of the appropriate number of minimum bias events - corresponding to different luminosities - is implemented in the reconstruction code.

Deliverable 15.4−A	**Pileup handling**
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	All subdetector reconstruction codes need to obtain simulated hits from several events: signal and minbias, in time and out-of-time. Physicists need access to the Monte-Carlo information in a consistent way.
Completion date:	existing
Constraints:	*Describe constraints, assumptions and risks.*

15.5 → PRS sections: Modelling of Electronics Response to Signals from the Detector *(Coordinator name)*

In the real world the readout electronics is responsible for converting the energy deposits from particles in the detector into electronic signals. The simulation of this - e.g. signal shapes - is not provided by GEANT but is part of the reconstruction.

15.6 → PRS sections: Modelling of Detector related and Electronic Noise *(Coordinator name)*

The simulation of detector noise is part of the reconstruction to ensure high flexibility and make studies with different simulated noise levels possible. In a similar way electronic noise is taken into account.

15.7 → PRS sections: Simulation of Level-1 Trigger Decisions *(Coordinator name)*

The simulation of detector noise is part of the reconstruction to ensure high flexibility and make studies with different simulated noise levels possible.

15.8 → PRS sections: Modelling of Time-Dependent Detector Characteristics *(Coordinator name)*

During the operation of the CMS detector many time-dependent changes will occur. Examples are the HV status of different subdetectors, time dependend noise levels, "dead" or "hot" readout channels. When comparing reconstructed data to Monte-Carlo such effects should be taken into account where necessary. The reconstruction software should provide the possibility to access information related to the detector status recorded for any given time and take it into account when reconstruction Monte-Carlo events.

15.9 → PRS sections: Full Reconstruction *(Coordinator name)*

There is an immediate need to perform detailed studies of the detector and trigger performance such that the design can be optimised. CMS has committed to perform all such studies using OO software. The ORCA task covers software for the reconstruction of simulated (and ultimately real) events. In the case of simulated events, it also includes the simulation of the detector and electronics response to the GEANT hits, the creation of digitised hits and L1 trigger primitives, and the simulation of the L1 trigger response. For both simulated and real data, it provides software for the reconstruction of tracks segments, clusters, etc. within individual subdetectors and certain entities which combine information from more than one subdetector, such as tracks spanning the whole of the tracker.

15.10 → PRS sections: Fast Reconstruction *(Coordinator name)*

Subtask Description here...what it aims to achieve and key functionality... should be easy to check if achieved or not *blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit* .

15.11 → PRS sections: Physics Object Reconstruction *(Coordinator name)*

Subtask Description here...what it aims to achieve and key functionality... should be easy to check if achieved or not *blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit* .

16 "CPROM" (?) Calibration Project Management *(Coordinator name)*

This task aims to provide the software infrastructure to support the determination, storage and provision of data describing the detector configuration. This includes the engineering description, validation data, alignments, energy calibrations, etc. using data from: test beams; laboratory measurements; survey data; in-situ dedicated systems; slow control systems; and and physics event samples.

16.1 Validation of Physics Data Quality *(Coordinator name)*

This subtask is responsible for validation of event data prior to subsequent reconstruction and analysis.

Deliverable 16.1−A	*Name of deliverable here.*
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 16.1−B	*Name of deliverable here.*
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

16.2 Calibration of Event Data *(Coordinator name)*

This item covers software associated to the calibration of the CMS subdetectors including alignments, energy calibrations, etc. using data from: test beams; laboratory measurements; survey data; in-situ dedicated systems; slow control systems; and and physics event samples. The Calibration clients include ORCA and potentially the OSCAR and Fast Simulation projects, which may ultimately incorporate realistic detector performance data into the simulations.

Deliverable 16.2−A	*Name of deliverable here.*
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 16.2−B	*Name of deliverable here.*
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

16.3 Luminosity Determination *(Coordinator name)*

Subtask Description here...what it aims to achieve and key functionality... should be easy to check if achieved or not *blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit blah blah blah blah yack yack yack yack rabbit rabbit rabbit* .

Deliverable 16.3−A	*Name of deliverable here.*
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

Deliverable 16.3−B	*Name of deliverable here.*
Description:	*Elaborate on purpose and functionality.*
Responsible:	*Responsible person*
Client/Sponsor:	*Who wants the deliverable? As specific as possible.*
Completion date:	*A date, or "before xxx", or "after yyy".*
Constraints:	*Describe constraints, assumptions and risks.*

A Resource Categorisation

Resources are tagged with a number of attributes (fields) to permit filtering, for example within MS Project. A number of fields are defined with the standard form:

X.VALUE

The following fields (i.e. *X*) are defined:

- **R** – Role Description
- **T** – Task Description
- **C** – Country
- **F** – Funding Source
- **I** – Institution

and *VALUE* is a string (e.g. an ackronym of set of initials) describing the field, as described below. Both the field and the value should be specified in upper case. Only the characters [A-Z] and the underscore are permitted.

This scheme makes the the entries more transparently understood and reduces the possibility of incomplete information leading to errors (*i.e.* one can flag missing information as an error by checking each resource has all the associated fields defined. Moreover, the prefix removes the possibility of ambiguities, for example a role having the same descriptive initials as a funding agency.

Multiple filters are defined as a comma delimited list. The conventional order for specifying the filters is:

R.*XX*, **T**.*XXXXX*, **C**.*XX*, **F**.*XXXXX*, **I**.*XXXXX*

Where the *X*'s denote the various values described in the sections below.

All fields should be specified for each resource. If a value which is not known it should be included with the *VALUE* set to **NK**. This ensures that the filter still operates correctly (e.g. gives the correct integral of resources over a category).

A.1 Field R – Role Description

This field defines the professional personnel category. It is assumed that a unique pro-rated cost may be assigned to each category. The values defined are as follows:

- **PM** – Project Management (including Project Manager, Resource Manager, Computing Manager)

- **SA** – Software Architect
- **SP** – Software Programmer
- **SS** – Software Support (including Librarian, Web Master, Production personnel, Toolsmith,...)

- **DA** – Database Architect
- **DP** – Database Programmer

- **PC** – Physicist Coordinator
- **PP** – Physicist Programmer

- **SM** – System Manager

- **OC** – Outside Consultant

For example, to specify that a person is working as a Software Architect, one would specify:

R.SA

where **R** denotes the role field.

A.2 Field T – Task Description

This field defines the task to which the resource is allocated. In general, tasks should correspond to Level 3 sub-projects. The following tasks are defined:

- **CARF** - CMS Analysis and Reconstruction Framework
- **CMSCAN** - CMS scan program (event visualisation)
- **CMSIM** - CMS simulation (Fortran-based with GEANT3)
- **DOCUMENTATION** - Software Documentation
- **MAG_FIELD** - Magnetic Field
- **INFO_SYSTEM** - Information System (World Wide Web, etc.)
- **ODBMS** - Object Database Management System
- **ORCA** - Object Reconstruction for CMS Analysis
- **OSCAR** - Object Simulation for CMS Analysis and Reconstruction (using GEANT4)
- **QA** - Quality Assurance
- **SW_PROCESS** - Software Process
- **SW_SUPPORT** - Software Support
- **SYS_MAN** - System Management
- **TEST_BEAM** - Test Beam

For example, to specify that a person is working on the ORCA sub-project, one would specify:

T.ORCA

where **T** denotes the task field.

A.3 Field C – Country Code

This field defines the country provided the corresponding resource. The value should be specified according to the two-letter ISO 3166 standard country coding scheme which, for completeness, is reproduced in Table 1. We define an additional "country":

- **CE** - CERN

since in many respects CERN may be treated as such. For example, when summing contributions by country it is clearer to have CERN contributions as a single entry rather than to make any attempt to split into individual member state contributions.

For example, to define a resource provided by the United Kingdom one would specify:

C.UK

where **C** is used to denote the country field.

A.4 Field F – Funding Source

This field denotes the funding source. This field is distinct from the country and institute fileds since there is not always a straightforward mapping from these to the funding source (both countires and individual institutes can have multiple sources of funding).

- **CE_CMS** - CERN CMS Group
- **CE_FELLOW** - CERN Fellow
- **CE_IT** - CERN IT Division
- **CE_PA** - CERN Project Associate
- **CH_NSF** - Swiss "National Science Foundation"
- **COMMON_PROJECT** - CMS Common Project
- **IT_INFN** - Italian INFN
- **US_DOE_BASE** - US Department of Energy (base programme)
- **US_DOE_PROJ** - US Department of Energy (project)
- **US_NSF_BASE** - US National Science Foundation (base programme)
- **US_NSF_PROJ** - US National Science Foundation (project)

Funding sources should in general be prefixed with the two-letter country code, shown in Table 1, followed by an underscore. Although this might appear to be redundant information it greatly reduces the possibility of filtering

Table 1: ISO 3166 standard country codes, alphabetically by country name.

AF - Afghanistan
AL - Albania
DZ - Algeria
AS - American Samoa
AD - Andorra
AO - Angola
Av - Anguilla
AQ - Antarctica
AG - Antigua and Barbuda
AR - Argentina
AM - Armenia
AA - Aruba
AU - Australia
AT - Austria
AZ - Azerbaijan
BF - Bahamas
BH - Bahrain
BB - Barbados
BD - Bangladesh
BY - Belarus
BE - Belgium
BZ - Belize
BJ - Benin
BM - Bermuda
BS - Bahamas
BT - Bhutan
BW - Botswana
BO - Bolivia
BA - Bosnia and Herzegovina
BV - Bouvet Island
BR - Brazil
IO - British Indian Ocean Territory
BN - Brunei Darussalam
BG - Bulgaria
BF - Burkina Faso
BI - Burundi
KH - Cambodia (from I'net)
CB - Cambodia (from CIA)
CM - Cameroon
CA - Canada
CV - Cape Verde
KY - Cayman Islands
CF - Central African Republic
TD - Chad
CL - Chile
CN - China
CX - Christmas Island
CC - Cocos (Keeling) Islands
CO - Colombia
KM - Comoros
CG - Congo
CD - Congo, Democratic Republic
CK - Cook Islands
CR - Costa Rica
CI - Cote D'Ivoire (Ivory Coast)
HR - Croatia (Hrvatska)
CU - Cuba
CY - Cyprus
CZ - Czech Republic
CS - Czechoslovakia (former)
DK - Denmark
DJ - Djibouti
DM - Dominica
DO - Dominican Republic
TP - East Timor
EC - Ecuador
EG - Egypt
SV - El Salvador
GQ - Equatorial Guinea
ER - Eritrea
EE - Estonia
ET - Ethiopia
FK - Falkland Islands
FO - Faroe Islands
FJ - Fiji
FI - Finland
FR - France
FX - France, Metropolitan
GF - French Guiana
PF - French Polynesia
TF - French Southern Territories
MK - F.Y.R.O.M. (Macedonia)
GA - Gabon
GM - Gambia
GE - Georgia
DE - Germany
GH - Ghana
GI - Gibraltar
GB - Great Britain (UK)
GR - Greece
GL - Greenland
GD - Grenada
GP - Guadeloupe
GU - Guam
GT - Guatemala
GN - Guinea
GW - Guinea-Bissau
GY - Guyana
HT - Haiti
HM - Heard and McDonald Islands
HN - Honduras
HK - Hong Kong
HU - Hungary
IS - Iceland
IN - India
ID - Indonesia
IR - Iran
IQ - Iraq
IE - Ireland
IL - Israel
IT - Italy
JM - Jamaica
JP - Japan
JO - Jordan
KZ - Kazakhstan
KE - Kenya
KI - Kiribati
KP - Korea (North)
KR - Korea (South)
KW - Kuwait
KG - Kyrgyzstan
LA - Laos
LV - Latvia
LB - Lebanon
LI - Liechtenstein
LR - Liberia
LY - Libya
LS - Lesotho
LT - Lithuania
LU - Luxembourg
MO - Macau
MG - Madagascar
MW - Malawi
MY - Malaysia
MV - Maldives
ML - Mali
MT - Malta
MH - Marshall Islands
MQ - Martinique
MR - Mauritania
MU - Mauritius
YT - Mayotte
MX - Mexico
FM - Micronesia
MC - Monaco
MD - Moldova
MA - Morocco
MN - Mongolia
MS - Montserrat
MZ - Mozambique
MM - Myanmar
NA - Namibia
NR - Nauru
NP - Nepal
NL - Netherlands
AN - Netherlands Antilles
NT - Neutral Zone
NC - New Caledonia
NZ - New Zealand
NI - Nicaragua
NE - Niger
NG - Nigeria
NU - Niue
NF - Norfolk Island
MP - N. Mariana Islands
NO - Norway
OM - Oman
PK - Pakistan
PW - Palau
PA - Panama
PG - Papua New Guinea
PY - Paraguay
PE - Peru
PH - Philippines
PN - Pitcairn
PL - Poland
PT - Portugal
PR - Puerto Rico
QA - Qatar
RE - Reunion
RO - Romania
RU - Russian Federation
RW - Rwanda
GS - S. Georgia and S. Sandwich Isls.
KN - Saint Kitts and Nevis
LC - Saint Lucia
VC - Saint Vincent and the Grenadines
WS - Samoa
SM - San Marino
ST - Sao Tome and Principe
SA - Saudi Arabia
SN - Senegal
SC - Seychelles
SL - Sierra Leone
SG - Singapore
SI - Slovenia
SK - Slovak Republic
Sb - Solomon Islands
SO - Somalia
ZA - South Africa
ES - Spain
LK - Sri Lanka
SH - St. Helena
PM - St. Pierre and Miquelon
SD - Sudan
SR - Suriname
SJ - Svalbard and Jan Mayen Islands
SZ - Swaziland
SE - Sweden
CH - Switzerland
SY - Syria
TW - Taiwan
TJ - Tajikistan
TZ - Tanzania
TH - Thailand
TG - Togo
TK - Tokelau
TO - Tonga
TT - Trinidad and Tobago
TN - Tunisia
TR - Turkey
TM - Turkmenistan
TC - Turks and Caicos Islands
TV - Tuvalu
UG - Uganda
UA - Ukraine
AE - United Arab Emirates
UK - United Kingdom
US - United States
UM - US Minor Outlying Islands
UY - Uruguay
SU - USSR (former)
UZ - Uzbekistan
VU - Vanuatu
VA - Vatican City State
VE - Venezuela
VN - Viet Nam
VG - Virgin Islands (British)
VI - Virgin Islands (U.S.)
WF - Wallis and Futuna Islands
EH - Western Sahara
YE - Yemen
YU - Yugoslavia
ZM - Zambia
ZR - Zaire (see: CD)
ZW - Zimbabwe

errors in cases where several countries have the same ackronyms (e.g. both the US and Switzerland have an "NSF").

For example, to specify that a position is funded by the US National Science Foundation from the base programme, one would specify:

F.US_NSF_BASE

A.5 Field I – Institute

This field denotes the institute providing the corresponding resource. The institute is supplied in addition to the country code and funding source. This provides a finer grain of filtering which enables institutional responsibilities to be identified and specific institutional costs, such as overhead rates and fringe benefits, to be treated correctly.

Rather than give an exhaustive list of institutes we rather give general rules for defining values.

- Use abbreviations only in exceptional cases where they are well known throughout CMS.
  E.g. use **CERN** for CERN but use **NORTHEASTERN** for Northeastern University, rather than **NU** or **NEU**.
- Do not include words such as "University".
  E.g. use **BRISTOL** for the University of Bristol.
- Before defining a new institute value, check it does not already exist.

For example, to specify that a person is working for CERN one would specify:

I.CERN

where **I** denotes the institute category.

References