A Real-time Control Computer for the E-ELT

Document: GF-PDR-02

Management plan and work packages definition

Version 1.0

18th January 2016

Change Record

Version	Date	Author(s)	Remarks
0.1	11 Jan 2016	D. Gratadour	Initial version
1.0	18 Jan 2016	D. Gratadour	Final version

Applicable Documents (AD)

These are the Green Flash PDR documents

No.	Title	Reference	Issue	Date
AD01	Introduction	GF-PDR-01		
AD02	Management plan and WP definition	GF-PDR-02		
AD03	Requirements Specification	GF-PDR-03		
AD04	System Architecture	GF-PDR-04		
AD05	Distributed GPUs for real-time HPC	GF-PDR-05		
AD06	FPGA Solution for hard real-time	GF-PDR-06		
AD07	Interconnect Strategy	GF-PDR-07		
AD08	Interface Control Document	GF-PDR-08		
AD09	Supervisor Strategy	GF-PDR-09		

Reference Documents (RD)

These are documents external to the Green Flash project

No.	Title	Reference	Issue	Date

Acronyms and abbreviations

Table 1 Acronyms and Abbreviations

AD	Applicable Document
AO	Adaptive Optics
CANARY	Durham/LESIA on-sky AO demonstrator
CPU	Central Processing Unit
CUDA	NVIDIA GPU based software development language
DARC	Durham AO Real-time Controller
DDS	Data Distribution Service
DM	Deformable Mirror
DRAGON	Durham laboratory-based AO demonstrator bench
ELT	Extremely Large Telescope
E-ELT	European ELT
ESO	European Souther Observatory
FPGA	Field Programmable Gate Array
GPU	Graphics Processing Unit
GUI	Graphical User Interface
HLS	High Level Synthesis
HPC	High Performance Computing
MIC	Many Integrated Core
MVM	Matrix-Vector Multiplication
NIC	Network Interface Controller
PCIe	Peripheral Component Interconnect express
RD	Reference Document
RTC	Real-Time Control
RTL	Register Transfer Level
SIMD	Single Instruction Multiple Data
SPARTA	ESO VLT AO Real-time Control System
SHERE	VLT Planet finder instrument
UDP	User Datagram Protocol
UK ATC	United Kingdom Astronomical Technology Centre
VLT	Very Large Telescope
WFS	Wave-Front Sensor
WP	Work Package

Table of Contents

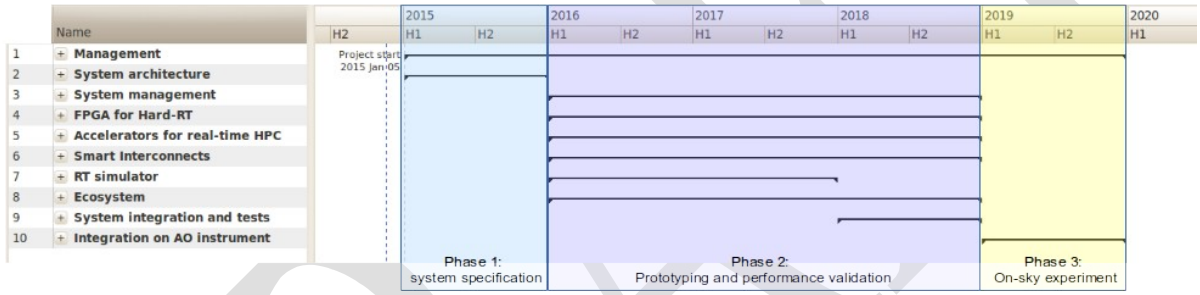
1 Scope

This document describes the management plan for Green FLASH and provide a comprehensive list of work packages descriptions.

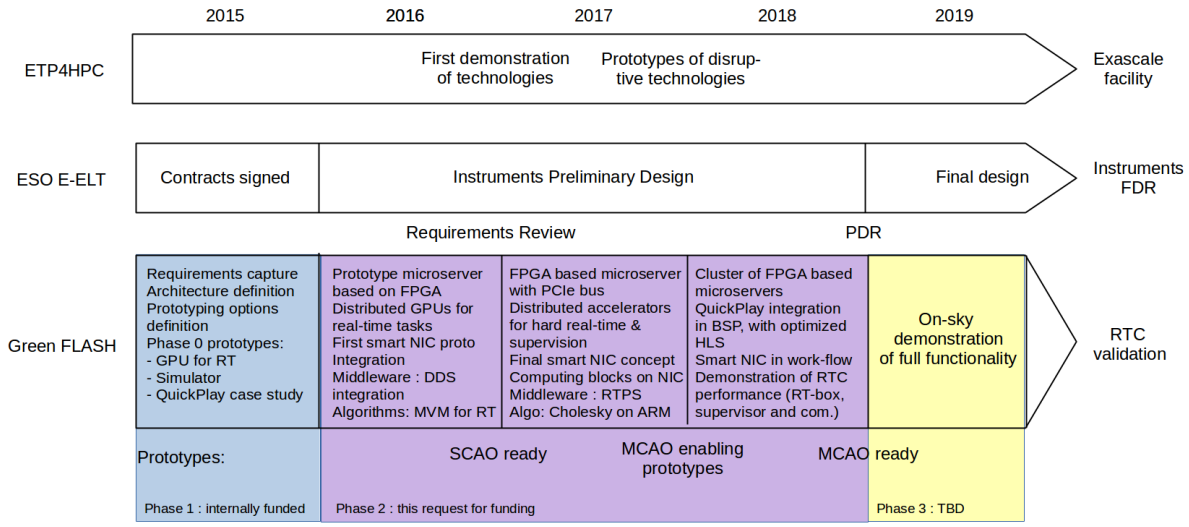
2 Management plan

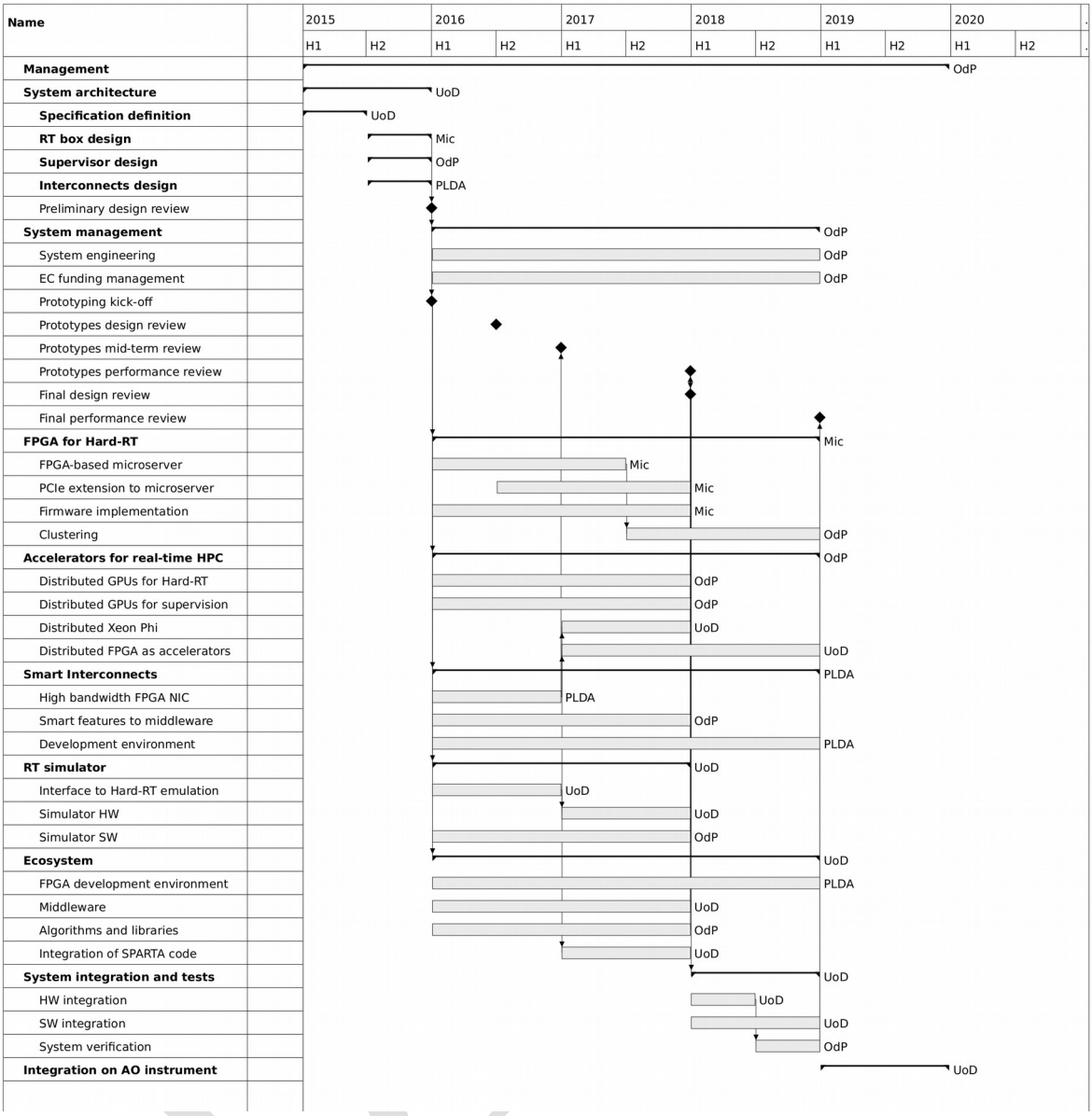
2.1 Project timeline

The following table provides a comprehensive of work packages and tasks as well as corresponding staff effort in person.month and start and end dates.The Green Flash project is broken down into 3 phases depicted in the figure below. In the following, we mostly describe the implementation of the second phase. However, for more clarity, we also show how it interacts with the other phases of the project.



The figure below shows the convergence between the Green FLASH schedule and technology demonstrations, the ETP4HPC program and the E-ELT instrumentation phase B.





The above figure is a detailed Gantt diagram, representing the timing of the main WP and their main tasks as well as the main milestones mentioned above. For each WP and tasks the lead partner, responsible for the corresponding deliverables, is displayed to the right. Some details are represented for the system architecture WP to show the interaction between the output of phase 1 and phase 2 prototyping.

The activities in the Green Flash project have been broken down into 10 work packages (WP):

1. Management
   System architecture*
2. System management
3. FPGA for hard real-time

4. Accelerators for real-time HPC
5. Smart interconnects
6. Real-time simulator
7. Ecosystem
8. System integration and tests
   Integration on AO instrument*

In the previous list the WP marked with a * are not funded by EC and have been or will be funded through other means. Each of these main WP contains several tasks, each being led by one of the partners and with possible contributions from the other partners.

2.2 Work plan

The following table provides a comprehensive of work packages and tasks as well as corresponding staff effort in person.month and start and end dates.

WP	WP Title	Lead	Lead name	p.m	Start	End
1	**Management**	**1**	**OdP**		**01/15**[1]	**12/19**
1.1	Project management and budget	1	OdP	(6)	01/15	12/19
1.2	Dissemination / valuation strategy	1	OdP	(12)	01/15	12/19
1.3	Energy efficiency	3	Mic	5	01/15	12/19
1.4	Quality insurance	4	PLDA	6	01/15	12/19
1.5	Cost reduction	3	Mic	7	01/15	12/19
2	**System management**	**1**	**OdP**		**M1**	**M36**
2.1	System architecture	2	UoD	(6)	M1	M6
2.2	System engineering	1	OdP	6 (3)	M1	M36
2.3	EC funding management	1	OdP	(3)	M1	M36
3	**FPGA solutions for hard Real-time**	**3**	**Mic**		**M12**	**M48**
3.1	Prototype FPGA-based microserver	3	Mic	23	M12	M30
3.2	PCIe extension of microserver	3	Mic	16	M18	M36
3.3	Firmware implementation	3	Mic	38	M12	M30

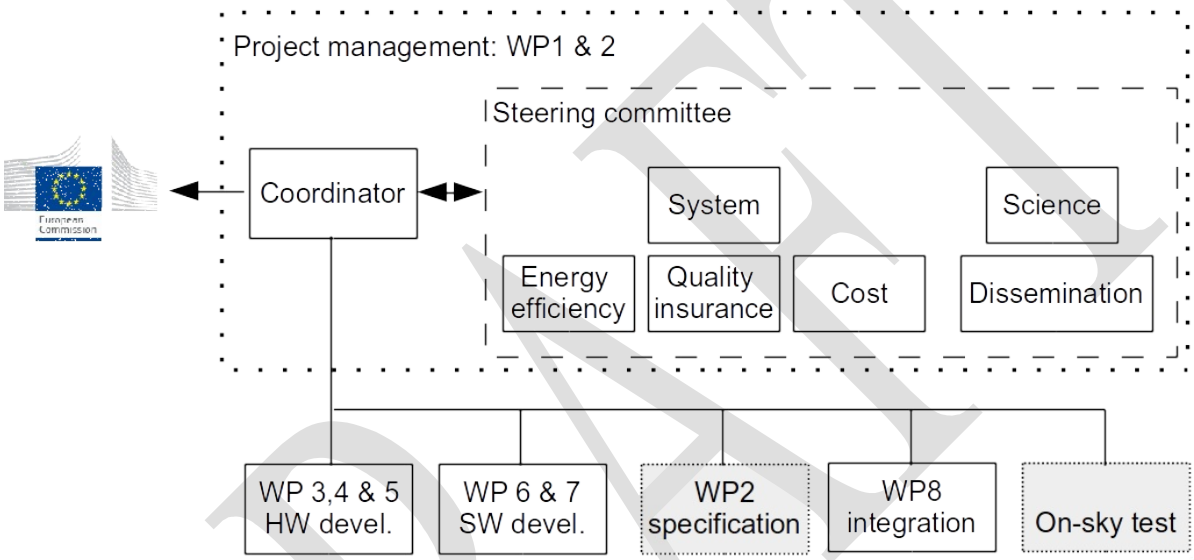
1 WP1 start and end dates take into account phase 1 and phase 3 of the overall project

3.4	Clustering microservers	1	OdP	25 (6)	M18	M36
4	**Accelerators for real-time HPC**	**1**	**OdP**		**M1**	**M36**
4.1	Distributed GPUs for real-time HPC	1	OdP	25 (6)	M1	M24
4.2	Energy efficient linear algebra with GPUs	1	OdP	20 (4)	M1	M24
4.3	Intel Xeon Phi as an alternative solution	2	UoD	10	M12	M24
4.4	FPGAs accelerators as alternative solution	2	UoD	20	M12	M36
5	**Smart interconnects**	**4**	**PLDA**		**M1**	**M36**
5.1	High bandwidth FPGA NIC	4	PLDA	10	M1	M12
5.2	Smart features to middleware	1	OdP	20 (10)	M1	M24
5.3	Development environment and IPs	4	PLDA	61 (5)	M1	M36
6	**Real-time simulator**	**2**	**UoD**		**M1**	**M24**
6.1	Interface to hard real-time emulation	2	UoD	10	M1	M12
6.2	Simulator hardware	2	UoD	6 (2)	M12	M24
6.3	Simulator software	1	OdP	20 (4)	M1	M24
7	**Ecosystem**	**2**	**UoD**		**M1**	**M36**
7.1	FPGA development environment	4	PLDA	17	M1	M36
7.2	Middleware	2	UoD	12	M1	M24
7.3	Algorithms and libraries	1	OdP	20 (6)	M1	M24
7.4	Integration of SPARTA	2	UoD	12	M12	M24
8	**System integration and tests**	**2**	**UoD**		**M36**	**M48**
8.1	Hardware integration	2	UoD	6	M24	M30
8.2	Software integration	2	UoD	9	M24	M30

8.3	System verification	1	OdP	10 (5)	M30	M36
Total person.months				**414 (+72)**		

2.3 Management structure

The management structure is depicted in the figure below. The project coordinator, responsible for the interaction with EC, manage the activities in the various WP in constant dialogue with the project Steering Committee addressing both system level issues and science and dissemination matters.



The management activities, both at the project and system levels, are covered in WP1 and WP2.

The Coordinator (OdP) is responsible for the management of the project, including maintaining the schedule, monitoring progress with respect to milestones and deliverables, and resolving any issue that threaten the project progress. The Coordinator is the single point- of-contact with the European Commission and takes responsibility for the provision of progress reports and other documents at fixed deadlines.

The Coordinator will be supported by the Observatoire de Paris Contract Department (SRCV), which has experience in handling European (Commission) research projects of this type. The support will cover contract and financial administration, including the preparation of cost statements, monitoring milestones and the receipt of deliverables, and the processing and reimbursement of partner institute financial claims.

A WP leader will be assigned to each WP, who reports to the Coordinator and oversees the activities in the WP. The WP leader is responsible for ensuring the work is kept to schedule and the deliverables are submitted to the Coordinator on time.

A group of experts from each project partner will constitute a Steering Committee, which will aid the Coordinator in all aspects of project and system management, along the course of the project. Meetings of the Steering Committee (project meetings) will be held so as to review the status of the

project in general, and in particular the status of deliverables, outstanding/completed action items and the assignment of new ones, and to discuss any proposed changes to the project schedule or work plan.

Each core aspect of the project is assigned an expert: energy efficiency (task 1.3, resp. Microgate) , cost reduction (task 1.5, resp. Microgate) and quality insurance (task 1.4, resp. PLDA) in coordination with the system engineer (task 3.1, resp. OdP) for system management on one side and compliance with science requirements and dissemination strategies (task 1.2, resp. UoD) for the scientific exploitation of the prototypes on the other side. The Steering Committee meetings will be held at intervals of 3 – 9 months and as required by circumstances, making use of video- and tele-conferencing as appropriate. Any participant in the project, may attend the meetings as appropriate and practical.

2.4 Risk mitigation

Description of risk	Work package(s) involved	Proposed risk-mitigation measures
Custom hardware logic does not meet expected performance once integrated with other components	WP4	Accurate device selection already in the early design phase, on base of a design based on accurate timing model. Develop accurate timing models already in the early design phase
On-board processor does not capable of performing all expected tasks as expected	WP4, WP5	Evaluate carefully in the early design phase the on-board processor performance Accurate real-time kernel selection and configuration
Design optimization of QuickPlay less performant than expected	WP4, WP6, WP8	Dimension properly the logic in order to leave more freedom to the automated workflow
Accelerator hardware unable to meet computational and latency requirements.	WP5, WP6	We will investigate multiple accelerator options: GPU, MIC, FPGA and down-select the most appropriate
No effort available to port software to emerging and future hardware through the lifetime of the project.	WP5, WP7, WP8	We will investigate the use of OpenCL (and other techniques) to largely abstract the application software from the hardware.

2.5 Interaction with external institutions / partners

As discussed in a document AD09, a collaboration with the MORSE project as allowed us to developed an optimized pipelined approach for the computation of the *reconstructor* matrix[2]. The goal of the MORSE project is to design linear algebra methods that achieve the fastest possible time to an accurate solution on large-scale multicore systems with hardware accelerators, using all the processing power that future high end homogeneous and heterogeneous systems can make available. It relies on the use of a dynamic run time system to schedule computational tasks simultaneously on various compute devices and a data flow programming model based on the use of direct acyclic graphs for an efficient scheduling in which the tasks are executed out-of-order and scheduled according to a critical path for the execution. This efficient approach has allowed us build a new implementation that outperforms asymptotically previous state-of-the-art implementations up to 13-fold speedup.

The interaction with ESO and E-ELT first light instruments consortia is a crucial input to all this research and development program. On one hand, it is essential to follow the ESO standards evolution and integrate them in our approach to guarantee the compatibility with the final solution. On the other hand, the precise instruments specifications will evolve during the preliminary design studies, scheduled to start concurrently with the green Flash project, with a possible impact on the general requirements in terms of bandwidth, computing power or algorithms. The good match between the schedules of these design studies and green Flash, will give us the opportunity to organize regular checkpoints with these consortia to monitor the possible evolution of these requirements and integrate them in our road map. The output of the green Flash project could serve as the starting point of the final design studies for the AO modules RTC.

2.6 Staff effort

	WP1	WP3	WP4	WP5	WP6	WP7	WP8	WP9	**Total person.months per participant**
1 OdP	**(12)**	**6 (6)**	6 (6)	**50 (10)**	20 (15)	10 (6)	20 (6)	10 (5)	122 (+ 66)
2 UoD	(6)			25		**26**	**24**	**15**	90 (+6)
3 Mic	12		**96**						108
4 PLDA	6				**71**		17		94
Total person.months	18 (+18)	6 (+6)	102 (+6)	75 (+10)	91 (+15)	36 (+6)	61 (+6)	25 (+5)	414 (+72)

In the table above, the figures for OdP and UoD between parenthesis are supported by internal funding and the rest of the figures are relevant to EC funding. Concerning the totals,

2 A. Charara et al., Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis 2014, DOI: 10.1109/SC.2014.27

the additional contribution from OdP and UoD is also displayed between parenthesis.

The overall staff effort for the prototyping phase of the green Flash project is estimated at 486 person.months of which 414 are funded by EC.

3 Work packages description

3.1 WP1: Management

Objectives

This WP in conjunction with WP3 (dedicated to the prototyping phase), will provide the management structure for the project during its full length including the prototyping and validation phase for which we require funding in this proposal and the preliminary study, described in WP2, and the system integration on telescope not described in this document. Beyond the overall project management / coordination, the dissemination strategy management is addressed here as well as energy efficiency, quality insurance and cost reduction.

Tasks description

- *Task 1.1: Project management and budget*
  Overall project coordination/management through a steering committee
- *Task 1.2: Dissemination strategy*
  Define and implement the dissemination strategy
- *Task 1.3: Energy efficiency*
  Define an energy efficient strategy and enforce it (benchmarks, measurements)
- *Task 1.4: Quality insurance*
  Management of the development process (traceability, documentation, standards, version control). For the various Green Flash project components
- *Task 1.5: Cost reduction*
  Define a road map for producing cost effective end-products

3.2 WP2: System Management

Objectives

This WP will provide the system management structure for the prototyping phase of the project. The main objectives are: requirements capture and specifications definition, global system engineering, management and coordination of the prototyping activities and EC funding management. The system engineer (task 2.2), who coordinates the prototyping activities and monitors progress through mid-term and final performance reviews is part of the project steering committee. The design will include details of all hardware and software options to be investigated. The system specification and architecture options will be discussed and validated during a preliminary design review (PDR).

Tasks description

- *Task 2.1: System architecture*
  This sub WP will collect the system requirements for the prototype real-time system for the E-ELT. These requirements will be based on input from ESO
- *Task 2.2: System engineering*

Coordinate and manage the prototyping activities. In this task, the system engineer coordinates the efforts led in the various prototyping WP including final integration and performance analysis.

- *Task 2.3: EC funding management*
  Provide a communication channel with EC and monitor budget

3.3 WP3: FPGA solutions for hard real-time

Objectives

This WP provides a concept study for a stackable, energy efficient microserver for data-intensive applications based on a high cell density FPGA harboring an ARM HPS. It involves the prototyping of a main board with Ethernet / Infiniband links and PCIe local interconnect and the production of several boards to be clustered. The performance in terms of communication bandwidth and compute throughput will be assessed for the AO application on a single board and on the small scale cluster.

Tasks description

- *Task 3.1: Prototype microserver based on FPGA*
  Develop a prototype microserver board based on FPGA with ARM HPS. This activity aims to develop a prototype of a high throughput computational and communication board dedicated to hard real-time control, data pre/post processing and communication with acquisition and control units.
- *Task 3.2: Extension of microserver with PCIe bus*
  Extend the prototype of microserver board to include PCIe root complex
- *Task 3.3: Firmware implementation*
  Develop IPs + drivers for the microserver boards

3.4 WP4: Accelerators for real-time HPC

Objectives

This WP aims at assessing solutions relying on hardware accelerators in a distributed memory configuration to address both the RT-box and the supervisor module designs of the AO RTC. The performance of these solutions, based on GPUs, Intel MIC and FPGA will evaluated in terms of determinism for the real-box performance and overall throughput for the supervision process. This WP has strong ties with WP6 on smart interconnects since in such distributed configuration, the achievable performance strongly depends on the ability to enable low latency intranodes communications. The output of this WP is a series of small scale prototypes relying on these various accelerator technologies on which both RT-box and supervisor strategies will be evaluated. This performance assessment will be used for the down selection of technologies during the final design review of the AO RTC prototype.

Tasks description

- *Task 4.1: Distributed GPUs for real-time HPC*
  In this task we propose to investigate the level of performance determinism and the scalability of GPU based clusters targeting the E-ELT first light AO RT box specification.

- *Task 4.2: Supervisor module based on GPUs*
  The goal of this task is to address the most compute intensive task of the supervisor module: the computation of the MMSE tomographic reconstructor matrix and more generally the achievable performance of .large matrix (up to 100k x 100k) inversion in a pipeline
- *Task 4.3: Xeon Phis as alternative solution*
  This task will investigate the use of the Xeon Phi as an alternative acceleration technology for the hard real-time data processing in the RTC prototype.
- *Task 4.4: FPGAs as alternative solution*
  A set of standards and benchmarks will be developed for the comparison of FPGA based systems with other technologies such as GPUs.

3.5 Smart interconnects

Objectives

The goal of this WP is to provide a comprehensive study of a smart interconnect concept, in the context of the AO application, including hardware, firmware, middleware and development environment considerations. Through this integrated approach, we propose to provide a tailored interconnect solution for the AO RTC supporting the various standard communication protocols in the system (internodes, with sensors, etc …) in a unified approach and including the ability to integrate computing blocks at the NIC level. In this WP several NIC prototypes will be delivered and their interoperability with mainstream middleware (including the integration of these features in the development environment) will be addressed.

Tasks description

- *Task 5.1: High bandwidth FPGA NIC*
  Develop a common architecture and several versions of a high bandwidth NIC (10 Gbe and infiniband).
- *Task 5.2: Smart features to middleware*
  The goal of this task is to define and test a strategy for the implementation of the smart interconnect features at the level of the middleware. This includes communication features for middlewares oriented towards data broadcasting (DDS) and system monitoring (CORBA) but also compute capabilities for middleware such as MPI or more generally new programming models.
- *Task 5.3: Development environment and IPs*
  Develop the QuickPlay ecosystem to add smart features and open the way to further developments

3.6 WP6: Real-time simulator

Objectives

The real-time simulator is a novel aspect of the RTC design and probably one of the most challenging. It draws together both the HPC requirements of the hard real-time data pipeline and those of accelerating large-scale simulations. The simulator is not a part of the hard real-time system but must interface to it and be sufficiently versatile to provide either full or partial simulation of any or all aspects of the RTC. The primary objective is to provide the ability to run the RTC with various (or all) of the elements of an AO system absent. This is essential for testing of the RTC during the AO development phase where various large-scale hardware items may not be available. Such a simulator is also critical to optimising the RTC and to understanding its performance. The HPC hardware used

for the simulator should allow it to run at a speed as close as possible to that of the actual RTC. Simulation code for AO systems must simulate guide stars, the turbulence of the atmosphere, the wave-front sensor cameras and the deformable mirrors. Such code is normally run on HPC accelerated systems but has never yet been integrated as a part of an AO system RTC or required to perform at real-time rates.

Tasks description

- *Task 6.1: Interface to hard real-time emulation*
  The actual hard real-time interfaces to both sensor cameras and to corrector deformable mirrors must be accurately emulated. This sub-work package will require substantial input from the design of the actual interconnects within the data pipeline.
- *Task 6.2: Simulator hardware*
  The goal of the data pipeline hardware within the RTC is to meet the requirements of the system in throughput, latency and jitter. Ideally, the simulation hardware would be capable of providing the same performance when part or all of the system is being simulated.
- *Task 6.3: Simulator software*
  The goal of this task is to provide a realistic an optimized software bundle for the real-time simulator

3.7 WP7: Ecosystem

Objectives

The tasks in this WP will supervise the developments of the essential blocks of the Green Flash ecosystem, namely FPGA development environment, middleware, algorithms and SPARTA components. While specific aspects of these topics will be addressed locally on other WP (as for instance FPGA development environment in WP6), the goal here is to manage this activities globally during the course of the prototyping phase to produce a coherent, comprehensive and expandable ecosystem fitting the needs of the Green Flash various components.

Some critical aspects will addressed such as the required evolutions in the middleware to handle efficiently data streams at the ELT scale or integration of new control and supervision schemes through standard libraries.

Tasks description

- *Task 7.1: FPGA development environment*
  Oversight of the use of QuickPlay in the project and coordinate developments. The main goal is to develop, expand and maintain the development environment for FPGA to meet project needs
- *Task 7.2: Middleware*
  The proposed RTC will be a heterogeneous distributed system and is fundamentally a system for handling large amounts of data very fast including real-time algorithmic manipulation. The middleware that is selected must provide all of the functionality that is required across the entire RTC.
- *Task 7.3: Algorithms and libraries*
  This task concentrates on the definition of the various algorithms and libraries to be used in the AO RTC pipeline, including the real-time box and the supervisor module.
- *Task 7.4: Integration of SPARTA*

The overall design of the prototype RTC will draw on past experience with such systems at Durham and at Paris observatory. As well as guiding the overall design on this basis, this work-package will also investigate the re-use of design ideas and code written for SPARTA.

3.8 WP8: System integration and tests

Objectives

The prototype RTC will be composed of a large number of both hardware and software elements developed by each of the 4 partners. This work-package will provide a plan for the management of the full integration, testing and verification of all of these parts. It will then involve the execution of this plan leading to a fully operational RTC prototype.

A system verification document will be produced that specifies a full set of system tests and a set of benchmarks that will be used to test the performance of the system against its specification and requirements. This system verification will make extensive use of the real-time simulator. However, we will also test the RTC using actual wave-front sensor cameras to generate pixel data, using phase-screen emulation for turbulence and using actual deformable mirrors. This testing will make extensive use of the AO infrastructure available at CfAI in Durham and at Paris Observatory.

Tasks description

- *Task 8.1: Hardware integration*
  The various hardware elements of the RTC will be individually tested under the relevant work package. This work package will provide an integration and testing plan well before the start of the integration phase. Whilst system verification is performed under a separate sub-work-package, the system hardware will be fully tested during the integration phase.
- *Task 8.2: Software integration*
  The overall software for the system will be very diverse; from FPGA IP cores through GPU based code to libraries of dedicated algorithmic code. This sub-work-package will draw together all of this software in parallel with the integration of the system hardware. All software packages will be delivered for integration with complete internal testing reports.
- *Task 8.3: System verification*
  The goal of this task is to lead the system verification phase of the RTC prototype using the RT simulator in various modes (high, medium and low precision).

3.9 Interfaces between work packages

In the figure below, we show how the various WP will interact along the course of the prototyping phase of the project and how the main responsabilities are distributed among the partners. WP1 the project and system management WP is not represented in this figure. WP2 is represented as an input to the PDR, i.e. the prototyping phase kick-off. For the sake of clarity, the interactions between individual tasks are not been represented everywhere. They are detailed in the following. The critical WP interactions are represented in red.

As shown in this figure WP5: smart interconnect and WP7: ecosystem are central to the project. They interact together and with most of the WP. Indeed, this is where the core components, common to all architecture designs are developed.

Inside WP5, the various tasks are strongly inter-related as detailed in the WP description. Moreover, between this WP and WP7, we expect a strong two-ways interaction between task 5.2 and task 7.2 on middleware developments and task 5.3 and task 7.1 on FPGA development environment. Additionally, the developments in WP5 around the communication and processing blocks for FPGA designs (mainly in task 5.3) should feed WP3 so as to ensure that these blocks can be integrated in the microserver internal design.

There is also a strong interaction between WP5 and WP4, as smart interconnects are expected to boost the accelerators clusters performance. Here again a strong two-ways interaction is expected, tasks in WP5 being fed by requirements derived from proposed architectures in WP5.

Moreover, the developments in WP7 inter-relate with the core prototyping activities in WP3 and WP4. The integration of the FPGA development environment (task 7.1) in the micorserver ecosystem (WP3) is critical to leverage the FPGA capacities at the core of the microserver by providing full flexibility in building tailored designs. Task 7.1 is thus linked to tasks 3.1, 3.2 and 3.3 through a two-ways interaction, feedback from the board hardware design as well as specific components in the firmware feeding the definition of the development road map in task 7.1. Additionally, the ability to port the rest of the software ecosystem (middleware, libraries and SPARTA components) is also an important aspect of the project and task 7.2, 7.3 and 7.4 inter-relate as well with WP3.

Finally, we also expect a strong interaction between tasks 7.2 and 7.3 and WP4. The prototypes developed in WP4 will be the testbeds for the performance assessment of middleware solutions and linear algebra algorithms and libraries. Moreover, the output from this performance assessment as well as the architectures of these prototypes and the evolution of their core components will continuously feed the development strategy led in WP7.