PDR.14 DEVELOPMENT PLAN (PDR TO CDR)

Document number	SKA-TEL-SDP-0000053
Context	MGT
Revision	01
Author	SDP Consortium
Release Date	2014-02-09
Document Classification	Unrestricted
Status	Draft

Name	Designation	Affiliation
Ian Cooper	Acting Project Manager	ICRAR
Signature & Date:	**Signature:** Ian Cooper Ian Cooper (Feb 10, 2015) **Email:** ian.cooper@uwa.edu.au	

Name	Designation	Affiliation
Paul Alexander	SDP Lead	University of Cambridge
Signature & Date:	**Signature:** Paul Alexander Paul Alexander (Feb 10, 2015) **Email:** pa@mrao.cam.ac.uk	

Version	Date of Issue	Prepared by	Comments
0.1	2015-02-09	V. Allan	

ORGANISATION DETAILS

Name	Science Data Processor Consortium

1 Table of Contents

2 List of Figures

3 List of Tables

4 Introduction

In this document we describe the programme structure, processes, management and planning that will be used to evolve and iterate the Preliminary Design Review (PDR) documentation Pack into a fully proven and costed Critical Design Review (CDR) suite of documentation, complete with a Procurement Document Set.

This document focuses on the overview of the development process in order to provide clarity and ease of comprehension. Furthermore in order to minimise repetition and reduce the risk of inconsistency, this document relies upon the suite of documentation that addresses the topics mentioned in greater detail. These include:

Document Title	Document Number
Assumptions & Non Conformances	SKA-TEL-SDP-0000014
Prototyping Plan	SKA-TEL-SDP-0000054
High Level Risk Register	SKA-TEL-SDP-0000052
Glossary of Terms	SKA-TEL-SDP-0000056

This document is arranged into eight major sections: one for each element of the product tree, and two sections for non-product elements of the project (Systems Engineering, and Management).

Our general plan to progress to CDR is to develop each element of the product tree to the level of detail required for procurement; we have provided detailed plans and descriptions only for those elements that are particularly high-risk, or are otherwise particularly important. For example, we have included descriptions for tasks to develop some elements that require a high level of coordination across existing work packages.

The SDP Consortium has adopted an approach where 'assumptions and non-conformances' are clearly stated (and justified where possible), instead of highlighting the gaps in the requirements and the associated knock-on effects on the completeness of the PDR submission. In this way the PDR Documents can be perceived to be complete, but they are based upon the stated assumptions and non-conformances. The risk to this approach is that some of the assumptions and non-conformances will be rejected; however we believe that this approach will allow a more comprehensive PDR at this point in time. We found that using the Baseline Design in isolation from any other design parameters resulted in extremely high data rates and data volumes, which when extrapolated exceeded not only the power cap, but also the Rough Order of Magnitude (ROM) cost for the SDP. Therefore, with the knowledge of the rebaselining activities being performed, we have tried to make realistic assumptions in key documents in order to move the SDP Element architecture/design and costings into a more realistic SDP ROM. The Assumption and Non-Conformances are clearly summarised in the document Assumptions & Non Conformances: SKA-TEL-SDP-0000014 [AD01].

5 References

5.1 Applicable Documents

The following documents are applicable to the extent stated herein. In the event of conflict between the contents of the applicable documents and this document, **the applicable documents** shall take precedence.

Reference Number	**Reference**
[AD01]	SKA-TEL-SDP-0000014 Assumptions & Non Conformances
[AD02]	SKA-TEL-SDP-0000054 Prototyping Plan
[AD03]	SKA-TEL-SDP-0000052 High Level Risk Register
[AD04]	SKA-TEL-SDP-0000056 Glossary of Terms
[AD05]	SKA-TEL-SDP-0000049 Compliance Matrix
[AD06]	SKA-TEL-SDP-0000042 Preliminary Software Engineering Plan
[AD07]	SKA-TEL-SDP-0000050 Preliminary ILS Plan
[AD08]	SKA-TEL-SDP-0000018 Compute Platform Element Sub-system Design
[AD09]	SKA-TEL-SDP-0000027 Pipelines Element Sub-system Design
[AD10]	SKA-TEL-SDP-0000021 Compute Platform - Improving Sensor Network Robustness
[AD11]	SKA-TEL-SDP-0000026 LMC Element Sub-system Design
[AD12]	SKA-TEL-SDP-0000028 Ingest Pipeline
[AD13]	SKA-TEL-SDP-0000025 Data Delivery Element Sub-system Design
[AD14]	SKA-TEL.SADT.SE-TEL.SDP.SE-ICD-001

5.2 Reference Documents

The following documents are referenced in this document. In the event of conflict between the contents of the referenced documents and this document, **this document** shall take precedence.

Reference Number	**Reference**
[RD01]	SKA SDP Element Industry Plan
[RD02]	Hyperion Initiative: https://hyperionproject.llnl.gov/index.php

6 MGT: SDP Management Strategy

6.1 Rationale and Strategy

The SDP Consortium will continue to utilise a System Engineering (SE) approach to the design process but with some flexibility e.g. Agile System Engineering, which uses an "Iterative and Incremental Design" approach. The system engineering approach covers Requirements & Interface Management together with a structured Prototyping Plan that will be managed under an agile methodology.

Prototyping is a key element of the design approach and we distinguish between horizontal prototyping, which aims to produce a shallow prototype across as much of the system as possible, and vertical prototyping, which focuses on the detailed performance specific components. Our aim is to perform as much horizontal prototyping as is required to define, and validate, the element architecture and perform vertical prototyping of the critical elements only. This approach applies to both the software and hardware aspects of the design. Our key prototyping projects are described in more detail in our Prototyping Plan [AD02]. Note that although individual work packages are leading certain prototyping tasks, other work packages are often involved in completing our prototypes. Our prototyping work is primarily aimed at demonstrating feasibility and hence de-risking critical areas of our architecture.

For the hardware aspects we have the additional design consideration that there will inevitably be evolution of the available technology between the completion of this design and procurement. We are working closely with industry, and our current interactions are describe in more detail below, and in our Industry Plan [RD01]; this will allow more detailed evaluations of our hardware and software options as we progress towards CDR.

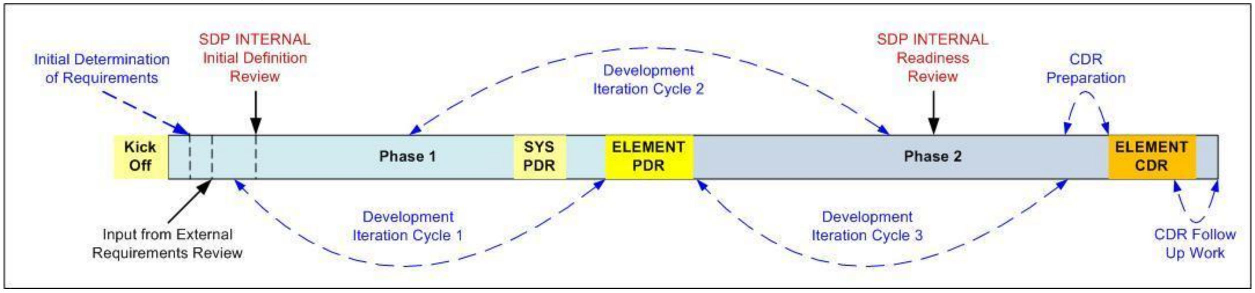
Since we delivered our RfP document set, DELIV has been formally included as part of the SDP. Hence our Consortium work packages define a scope which considers the overall design through to data delivery to the end-user astronomers. For our delivery of data to astronomers, we assume a standard tiered structure of the form pioneered by CERN. In our professional judgement it is essential that a full technical analysis of the data delivery, archive and analysis is now performed so that technically informed decisions can be made about, for example, regional data centres and to obtain proper costs for the SKA project in its broadest context.

6.2 Critical System Requirements and Technologies

Schedule

The SDP development process utilises an agile & iterative system engineering approach which increases the fidelity of the architecture/design with each iteration and culminating with iteration 5 ready for CDR. In addition to this, each of the sub-elements have their own iterative processes which are aligned with the overall SDP milestones. The JIRA Agile Tracking System will be used to manage the project schedule. As the WBS activities are extracted and allocated to individuals the start and stop dates allow us to automatically generate a GANTT chart and to monitor progress.

6.3 SDP Project Timeline

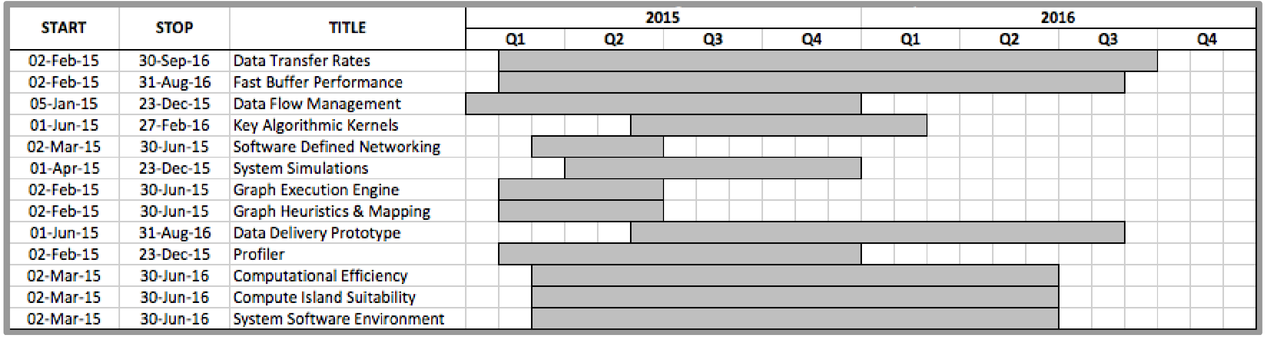


Initial Determination of Requirements
SDP INTERNAL Initial Definition Review
Development Iteration Cycle 2
SDP INTERNAL Readiness Review
CDR Preparation
Kick Off
Phase 1
SYS PDR
ELEMENT PDR
Phase 2
ELEMENT CDR
Input from External Requirements Review
Development Iteration Cycle 1
Development Iteration Cycle 3
CDR Follow Up Work

FIGURE 1: SDP PROJECT TIMELINE

The interim SDP Phase Two Milestones between PDR and CDR are intended to be "snap-shots" of the SDP architecture/design development process (which relies heavily upon the progress of the prototyping) and are used as a way of tracking progress (without adding additional overheads that could defocus the development activities).

Prototype Plan - Draft Top Level Schedule



START	STOP	TITLE
02-Feb-15	30-Sep-16	Data Transfer Rates
02-Feb-15	31-Aug-16	Fast Buffer Performance
05-Jan-15	23-Dec-15	Data Flow Management
01-Jun-15	27-Feb-16	Key Algorithmic Kernels
02-Mar-15	30-Jun-15	Software Defined Networking
01-Apr-15	23-Dec-15	System Simulations
02-Feb-15	30-Jun-15	Graph Execution Engine
02-Feb-15	30-Jun-15	Graph Heuristics & Mapping
01-Jun-15	31-Aug-16	Data Delivery Prototype
02-Feb-15	23-Dec-15	Profiler
02-Mar-15	30-Jun-16	Computational Efficiency
02-Mar-15	30-Jun-16	Compute Island Suitability
02-Mar-15	30-Jun-16	System Software Environment

FIGURE 2: PROTOTYPE PLAN – DRAFT TOP LEVEL SCHEDULE

In developing the Prototype Plan [AD02] we have allowed the individual activity milestones to evolve in isolation from each other; however following the PDR review and rebaselining activities, the revised prototype milestones will be grouped at logical timescales to form the SDP programme milestones leading up to the CDR. The initial plan uses a number of Prototype Status Reviews as the methodology for tracking progress. These suggested milestones do not match those listed in our RfP; we will explicitly reconsider our Phase 2 milestones after rebaselining of SKA1 is completed.

Delta-PDR

We expect to present a delta-PDR after the rebaselining of the SKA Observatory is completed. At the moment, some elements of our costs and architecture can only progress to a level suitable for PDR once rebaselining is completed. At delta-PDR we intend to review our product tree, especially the software elements; our architecture; our work breakdown structure; our milestones; our costs; and our construction plan. The changes we expect to be able to make to the product tree and architecture at that stage will inform our other reviews. Depending on the rest of the project, this will either be a formal submission to SKAO, or an internal milestone; we will have to review our work after rebaselining.

At this stage, we anticipate that our Phase 2 Milestones will be aligned to our prototyping, as described above; we expect to have four milestones before CDR submission, at roughly four-monthly intervals, with an earned value of 1.5M€ per milestone.

6.4 Risk Analysis

The High Level Risks have been identified both for the Pre-Construction and Construction timeframes. These are contained in the High Level Risk Register (HLRR) [AD03].

For the Pre-construction timeframe (PDR to CDR) each of the major risks have been identified that relate to the architecture/design verification activities and form the input into the prototyping plan. We will continue our risk evaluation towards CDR - see task T-13 below.

6.5 Software System Prototyping

Software prototyping will be a mix of horizontal and vertical prototyping. Horizontal prototyping will be aimed at prototyping as much of a complete software system as is required for the delivery of the detailed designs for CDR. A key element of the overall design of the software system is that it will need to continually evolve during the operation of the SKA, not only as the telescope matures from SKA1 to SKA2, but also as new requirements and methodology emerge – these statements are based on experience of existing facilities and acknowledgement of the inherent evolving requirements of any world-class science facility. A key design aim for the software architecture is to accommodate this anticipated evolution and at the same time to minimise long-term software development costs by good architectural design from the outset. The system-level prototyping will be a key element in these design considerations.

The system software prototype will:

Be used to test and verify the system decomposition and specification of internal interfaces, which emerge from the architectural analysis

Test models for scalability to SKA1 and give consideration to scaling of the software system to SKA2.

Prototype the system architecture to test for the required flexibility in architectural design.

Prototype and test the design models for loose versus tight coupling between software and hardware.

The complexity of the system prototype will be minimised. Emphasis will be placed on the system decomposition and interfaces rather than performance. Components within the overall system will initially be (and may remain) wrapped versions of existing components conforming to the SKA interface definitions, or highly simplified implementation of components.

The prototype will be run on existing HPC infrastructures using facilities available within consortium partners (Cambridge HPCS, iVEC, Hartree, CHPC, etc). Later iterations will, if required, be integrated in part on the system hardware prototypes developed within our Open Architecture Laboratory as well as standard HPC facilities.

Vertical software prototyping will be aimed at prototyping and testing component performance and compliance. Our approach to architectural analysis and prototyping, with emphasis on interfaces, will enable vertical prototyping of elements to be done independently of one another, provided they conform to the overall system internal interface definitions. There will inevitably be interdependencies especially between high-level components and those components forming parts of the common system software and common pipeline software frameworks. In these cases the vertical prototyping work will also verify and test the requirements for these common frameworks.

The key software prototyping tasks are listed below, and a fuller discussion of those is available in the Prototyping Plan [AD02].

6.6 Management

The Project Management of the SDP element shall primarily comply with the documented SKA management prescriptions, however some customisation of the general principles is required to support the SDP consortium structure and resource commitments. The full Management details are found in SKA-TEL.OFF.MGT-SKO-MP-002.

6.7 Progress Reporting

There are a number of levels of progress reporting within the SDP Project Management:

- Monthly Reports from the Project Manager, System Engineer, System Scientist, Quality Assurance Engineer & Task Leaders are shared across the consortium and externally to the SKAO via the Confluence wiki.
- Fortnightly Telecons between the SDP and SKAO Project Managers.
- Weekly Management Meeting with Task Leaders (where the deliverables, prototype status and system engineering is addressed and the minutes are shared via Confluence wiki)
- Periodic (typically 2-3 times per year) Face-to-Face meeting held in consortium members' countries and web-conferences, which SKAO representatives are invited to attend.

6.8 Budget

The SDP consortium is fully self-funded for the Pre-Construction work with approx 2500 person months of resource. The SDP consortium organisations assume a percentage of Earned Value per SDP Project deliverable milestone, based upon their percentage of the consortium's total resource. A linear resource distribution is also assumed throughout the SDP Project, however this will vary slightly prior to each milestone deliverable, as consortium wide reviews and consistency checking is required to bridge many time zones.

6.9 Progress Verification

The progress of the SDP consortium towards delivering a costed design pack for the CDR can be directly measured by the reduction of the risks involved in defining the architecture/design of the SDP. We have recorded our known risks and are using our prototype plan to mitigate them, however during the prototyping activities the granularity of the risks will be expanded and our list of risks will become more targeted. Thus we can expect the number of risks to increase during 2015, whilst, at the same time, some of the known risks are retired or their severity is reduced. In 2016 the retired risks should exceed the evolved risks and the architecture/design will enter a state with any remaining risks fully understood. Therefore the true measure of the progress will be obtained by the number of retired or ameliorated risks (due to the prototyping activities) and not by the number of risks themselves.

6.10Industry Engagement

A key aspect of our industry engagement in general, and the Open Architecture Lab in particular, is the creation of a vehicle for the IT supply chain to effectively interact with, and contribute towards, the development of SKA IT systems. The Lab will design and host a number of SKA data processing test beds as part of the SDP pre-construction work-package execution. These test beds will utilise products and know-how provided by industrial partners. The approach to hardware prototyping is based on a similar model to that adopted by Lawrence Livermore in their Hyperion initiative [RD02]. The main elements of this approach are integrated work with industry partners, an emphasis on determining an appropriate scalable element for prototyping and an emphasis on system-level components of the software stack.

As with the software prototype activities being managed under a distributed approach, the consortium members will utilise their local industry partners to assist with their areas of activity. A list of current industrial engagements is provided below. In addition to these specific engagements, a wider dissemination activity will be planned to increase awareness of the SDP.

In the context of prototyping work for the SKA the general approach is to prototype and evaluate heterogeneous architectures for streaming data processing. The emphasis will be on architectures and the associated software stack – a key aim will be to enable the SDP element to be designed so that the SKA can take advantage of developments within the ICT industry both between the end of the detailed design stage and procurement, but also going forward with a long-term upgrade and refresh policy/architecture for the SKA compute infrastructure. Key considerations will be energy efficient architectures, storage systems, role of accelerators, memory hierarchies, interconnects, scalable software stacks and file systems / object stores. For a more detailed discussion, see our Industry Plan [RD01].

The lab will create an evaluation and prototyping test-bed for new hardware and software technologies to address some of the issues with Petascale I/O technology scaling for SKA1 and future capacity to SKA2. This will involve examining processor, memory, networking, storage, visualisation, and other technologies, bearing in mind our need to allow for future technology refresh, expansion, and upgrades. Because of this, and because of the long lifespan of the project, we will be examining Open source software stacks.

Cambridge University have already issued a number of contracts to industry to work on our hardware and software:

- Design and prototyping of a dataflow system and domain specific language for the SKA Science Data Processor: ~£430k awarded to Braam Research LLC, CO, USA. The principal of Braam

Research LLC is Peter J Braam, the creator of the Lustre Parallel filesystem project and well-known computer scientist.

- Delivery of an energy efficient data-centric compute engine (SKA contribution £250k). Awarded to DELL and NVIDIA.
- Investigation and design of integrating software RAID into our data-driven architecture (£89k), awarded to Calsoft.
- Development of efficient kernels on accelerator hardware (£360k) awarded to NVidia corporation. Work commenced in January 2015.
- Intel – new joint software lab in Cambridge, investigating the use of accelerated kernels on Phi architecture and investigating ou overall system architecture.
- DELL – Use of Cambridge/DELL solutions centre to provide system architecture commentary on SDP design.
- Nvidia are funding and located one NVIDIA engineer in Cambridge for SKA work in addition to the funded contract and will provide system architecture commentary on SDP design.
- Ongoing research and industrial interaction with ARM to understand the planned ARM ecosystem for servers and the software and interconnect ecosystem developing around ARM.
- Seagate Systems UK (formerly Xyratex) – active engagement and advice on storage architectures, system design and approaches to data-centric workflows.
- SGI – active engagement on system architecture, resilience, memory hierarchies and storage systems
- AWS - active engagement via ICRAR into cloud-based solution architectures and data management

Microsoft Research – involvement in data-driven architectures and support for Haskell as a possible basis for a DSL

We have also had advanced technical meetings and discussions with:

- Bull - system architecture, cooling and energy efficiency
- Maxeler– involvement via their Hartree system
- Oracle – system architecture commentary on SDP design
- Mellanox – leading switch design and manufacture: interaction on SDP and LFAA
- Cray - system architecture and interconnects
- DDN - storage systems
- HP - general discussions
- Canonical and Red Hat - active discussions on OpenStack Proof of Concept.
- SoftIron UK - active discussions on ARM Server Deployment
- AMD - technical meetings on roadmap

ICRAR located in Perth, Australia also has a number of on-going collaborations with industry:

- Thoughtworks: Reviewed the Pre-Construction project requirements and provided the background and high-level rationale for an approach to deliver the Critical Design Requirements for the SDP pre-construction, which included recommendations & configurations for an environment to support the development of the CDR documents for the SKA SDP.
- Systemic: Have started work in defining the Data Life Cycle Model using SysML.
- DDN: Have performed feasibility studies upon object storage and virtualisation equipment.
- Catalyst IT: Are being considered for a study into the CEPH file system.

The Space Advisory Company is a South African company providing various system engineering related services to the SDP, which includes providing the System Engineer for the consortium who is part of the core Management Team. The company focuses in the standard system engineering areas (requirements management, traceability, etc), but has also been key in developing the costing model and conversion of the performance model to a more programmatic format. We anticipate continued strong involvement from the company. The involvement is largely enabled through the Financial Assistance Programme being run by SKA SA.

The DOME project (www.dome-exascale.nl) funds the ASTRON & IBM Center for Exascale Technology, a research centre, located in Dwingeloo. At this centre ASTRON and IBM (also via its Zürich Research Laboratory) jointly carry out fundamental research into technologies needed to develop the SKA. In addition, the design efforts of ASTRON in the SDP consortium, including leading the Data and Pipeline work packages, are funded through Dome. The collaboration is supported by grants from the Netherlands Ministry of Economic Affairs and the Province of Drenthe, but IBM is also investing in technology development for the SKA, which will be of direct benefit to the project.

Dome is active in three areas. Green Computing addresses technologies to reduce radically the power needed to do computationally intensive work on extremely large amounts of data. Nano-photonics looks at what is needed to drastically improve data transport performance over longer distances and inside computing machines. Data & Streaming focuses on technologies to process data on-the-fly and store data with high efficiency for later use. The project also provides information at system level by performing a quantitative analysis of the data-flow design of SKA and other radio telescopes.

A Users Platform forms part of the DOME project and reaches out to companies (in particular SMEs) but also universities and research institutes to take part in research programmes in areas of common interest.

Companies with interest in the DOME user platform include Dysi-SI (looking into large-scale multi-tier storage technologies and architecture analysis), Transfer-DSW and Sintecs (microserver architectures), and Datacentre Groningen (focusing on energy efficiency (green computing) and data security).

In the course of the LOFAR project, ASTRON developed a very successful form of Public-Private Partnership that allows companies of all sizes - from start-ups to Small-Medium Sized Companies (SME's) and even major multinationals - to become involved in the development of technology for large research infrastructure projects. A modest subsidy (usually between 25 and 35% of the real costs) stimulates industry to perform research and development early-on in the design process. While ASTRON benefits from this collaboration by having access to state-of-the-art industrial processes, knowledge is transferred back to the companies and strengthens their ability to compete in public procurements, often allowing them to win contracts for construction. This approach also results in a network of well-equipped companies that can efficiently and effectively collaborate in high-tech projects. This approach has been adopted in the SKA-Noord Nederland project (SKA-NN 2009-2013, supported by regional funds in the North of the Netherlands), which in part focused on technology and production of complex signal processing systems, and on advanced real-time data processing techniques.

6.11 Highlighted Tasks

In this section and in the highlighted tasks sections throughout the document we highlight important tasks in our development plan that are relevant to the work area in question. Some tasks affect multiple work areas (and are listed repeatedly in this document)[1].

T-1: We will be developing our operational experience at scale. We will be building on operational experience at precursor and pathfinder instruments, and our data centre operational expertise. We will consider offering internships at our HPC labs, or at US national labs. This affects component C.1 The work will be done by the MGT, PROT.OAL and ARCH.OPS teams.

T-2: We will investigate the trade-offs between standardisation of compute islands and compute island components, and the flexibility that compute islands of various technology capabilities (such as having compute nodes with different numbers or types of GPU on them, procured in partial upgrade cycles) or with different configurations. We will work on the following sub-tasks towards CDR; however, we may not have complete plans before the Construction phase begins.

T-2.1: Investigate multiple configurations of self-contained compute islands. This affects component C.1.1. The work will be undertaken by the COMP package.

T-2.2: Provide an analysis of the benefits of standardisation, and why it is preferred from an operational point of view. This affects component C.1.1. The work will be undertaken by the ARCH.OPS workpackage.

T-2.3: Provide an analysis of the benefits of partial upgrades allowing for a natural migration of data in science archive. This task along with T-2.2 will allow the management team to present options for the operational management of the data centres. This affects components C.1.1 and C.4. The work will be done by the MGT and ARCH.OPS packages.

T-2.4: Multiple Compute Island grades make LMC/scheduler task more difficult. This affects components C.6, C.1. The work for this will be led by COMP and involve LMC.

T-2.5: We recognise that the optimal combination of standardisation vs. flexibility vs. optimisation may be different per telescope, and will provide preliminary analyses to reflect this. This affects component C.1.1. The work will be done by the ARCH.OPS workpackage.

T-3: Further Operations Management plans will need to be prepared, covering, amongst other items: service levels, system maintenance process and lifecycle management. This affects component C.2.2.3. The work will be undertaken by MGT, PROT.OAL and ARCH.OPS.

T-4: We will perform a Technology Readiness Level (TRL) analysis. This is not something that is suitable for us to do for PDR, because, for example, SDP compute nodes do not exist yet in the form that will likely be available for deployment in 2022 (which implies a very low TRL), but by 2022, we intend to deploy COTS hardware, which implies a very high TRL. An initial analysis will be provided early in our workplan for CDR, and we will continue updating the TRL analysis as we get more information about the likely shape of hardware in 2022.

T-67: MGT will also be co-ordinating the work required to produce our CDR documentation set.

[1] The task numbering scheme is arbitrary – it reflects the process by which tasks were identified and not their priority status. Some numbers are missing completely as duplicated tasks were identified and merged.

T1 - SDPR-M8-P33/34 - Handling large data.

6.12 Management Risks

SDPR-M8-01 - Rebaselining documentation is incomplete.
SDPR-M8-02 - Rebaselining documentation is late.
SDPR-M8-03 - Workflow deliverables need aligning.
SDPR-M8-04 - Workflow drains resources.
SDPR-M8-05 - SDP Assumption mis-aligned with rebaselining requirements.
SDPR-M8-06 - Prototyping activities have to be changed to align with rebaseling requirements.
SDPR-M8-26/28/31 - SDP Consortium has insufficient resources.
SDPR-M8-27 - SDP Consortium has insufficient experience.
SDPR-M8-29 - Prototyping activities exceed the schedule

6.13 CDR Core Documentation Deliverables

CDR Doc Ref	CDR Document Title
SDP.CDR.01	Statement of Compliance with Requirements
SDP.CDR.02	Procurement Document Set: Architecture Design Document System & Sub-system Requirements & Specification for System & Sub-system Interface Control Document (ICD) Statement of Work (SOW)
SDP.CDR.03	Element Sub-system Design Report
SDP.CDR.04	Requirements Analysis & Allocations
SDP.CDR.05	Parametric Models & Analysis Report
SDP.CDR.06	Software Engineering Plan
SDP.CDR.07	Detailed Cost Breakdown Analysis
SDP.CDR.08	Construction & Implementation Plan
SDP.CDR.09	Configurations Items List
SDP.CDR.10	Element O&M Plan
SDP.CDR.11	Qualification & Acceptance Plan
SDP.CDR.12	Detailed Level Risk Register
SDP.CDR.13	Integrated Logistics Plan
SDP.CDR.14	Training Plan

TABLE 1: CDR CORE DOCUMENTATION DELIVERABLES

7 SE: SDP Systems Engineering Strategy and Workplan

SE tasks cut across the Consortium work packages; most, if not all, teams will be involved in SE tasks at some point between now and CDR.

7.1 Highlighted Tasks

T-5: Develop the Product Tree, increasing the detail at which we describe our components. This is a high priority, as many other items depend on a more detailed component breakdown for their progress.

T-6: Develop the Configuration Items list as the Product tree becomes more detailed. See PDR.10 [AD05].

T-7: Investigate Configuration Management Tools, in order to provide a smooth transition into the Construction phase. See PDR.10 [AD05].

T-8: Develop Configuration Management Plans. See PDR.06 [AD06].

T-9: Develop Software Engineering Plans, once the software components are more precisely specified. See PDR.06 [AD06].

T-10: Develop requirements around calibration in collaboration with PIP and SKAO, as the SKAO provides more information on their calibration strategy. This affects component C.4.1.4.

T-11: Develop requirements around the EoR pipeline in collaboration with the PIP package. This affects component C.4.

T-12: Perform an availability analysis for our systems hardware (see PDR.11) [AD07]. This affects component C.1. The work will be undertaken by the SE, COMP and PROT.OAL packages.

T-13: Continue risk management.

T-14: We will perform a FMECA analysis, starting with the top-level elements of the product tree, and proceeding to lower-level elements if time permits. (See also T-12).

T-15: Continue maturation of requirements in general and the functions and product tree.

T-16: Mature L3 ICDs (ICDs between SDP and other consortia) and develop L4 ICDs (ICDs within SDP).

T-17: We will develop a training plan for SDP operation.

T-18: Create the Pre-construction V&Q plan:

- **T-18.1:** Create verification requirement for each requirement
- **T-18.2**: Create Test plan (could be in the detail of the Prototyping plan)
- **T-18.3**: Execute test plan (prototyping activity)
- **T-18.4**: Measure compliance against requirements
- **T-18.5**: Create documents for CDR

T-19: Construction phase V&Q: Create Test plan.

T-32: We will provide a more detailed analysis of the issues involved in scaling to SKA2 size as we approach CDR. This will affect components C.1, C.2, C.3 and C.6 in particular. Work will be led by the COMP workpackage.

7.2 High-level Risks

SDPR-M8-P25 - SDP External ICDs
SDPR-M7-G21 - efficiency of whole system integration. This work will partially be addressed by T-38.

8 C.1 Hardware Compute Platform

8.1 Components

Number	Component Name	Description
C.1	Hardware Compute Platform	All hardware components within the SKA Science Data Processor
C.1.1	Compute Island	The basic replicable unit of the SKA Science Data Processor
C.1.1.1	Compute Island Management	Dedicated hardware and software to facilitate efficient management of the compute island hardware
C.1.1.2	Compute Node	Basic replicable unit of the compute island
C.1.2	Buffer	Hardware to buffer intermediate data to facilitate iterative processing
C.1.3	SDP Infrastructure	Local cooling, rack space and local power distribution. Interfaces with the data centre infrastructure resources.
C.1.3.1	Racks	The racks to house the science data processor equipment. Includes integrated water-cooling if applicable.
C.1.4	Hierarchical Storage	Hierarchical storage system for science data
C.1.4.1	Medium Performance Buffer	Part of the Hierarchical storage.
C.1.4.2	Long Term Storage	Long term storage equipment for science products
C.1.5	Interconnect System	The collection of data transport networks within the Science Data processor
C.1.5.1	Low-latency network core switch	Hardware to interconnect the internal low-latency networks present within each compute island
C.1.5.2	Management Network	Dedicated network to manage and control the various hardware resources within the SDP. Connects the management compute

		island resources as well.
C.1.5.3	Data Transport Network	The data transport network responsible for the bulk data transport into and out of the compute islands
C.1.6	Delivery Platform Hardware	Dedicated hardware for data delivery
C.1.7	LMC Hardware	Dedicated hardware for Local Monitoring and control

TABLE 2: COMPONENTS OF THE HARDWARE COMPUTE PLATFORM

8.2 Highlighted Tasks

Tasks are ordered numerically, not by importance. This is not a complete list of tasks needed to develop all components to an appropriate level for CDR.

T-1: We will be developing our operational experience at scale. We will be building on operational experience at precursor and pathfinder instruments, and our data centre operational expertise. We will consider offering internships at our HPC labs, or at US national labs. This affects component C.1 The work will be done by the MGT, PROT.OAL and ARCH.OPS teams.

T-2: We will investigate the trade-offs between standardisation of compute islands and compute island components, and the flexibility that compute islands of various technology capabilities (such as having compute nodes configured with different numbers or types of GPU, procured in partial upgrade cycles) or with different configurations offer. We will work on the following sub-tasks towards CDR; however, we may not have complete plans before the Construction phase begins.

T-2.1: Investigate multiple configurations of self-contained compute islands. This affects component C.1.1. The work will be undertaken by the COMP package.

T-2.2: Provide an analysis of the benefits of standardisation, and why it is preferred from an operational point of view. This affects component C.1.1. The work will be undertaken by the ARCH.OPS workpackage.

T-2.3: Provide an analysis of the benefits of partial upgrades allowing for a natural migration of data in science archive. This task along with T-2.2 will allow the management team to present options for the operational management of the data centres. This affects components C.1.1 and C.4. The work will be done by the MGT and ARCH.OPS packages.

T-2.4: Heterogeneous Compute Island Configurations make LMC/scheduler task more difficult. This affects components C.6, C.1. The work for this will be led by COMP and involve LMC.

T-2.5: We recognise that the optimal combination of standardisation vs. flexibility vs. optimisation may be different per telescope, and will provide preliminary analyses to reflect this. This affects component C.1.1. The work will be done by the ARCH.OPS workpackage.

T-12: Perform an availability analysis for systems hardware (see PDR.11 [AD07]). See also T-25.This affects component C.1. The work will be undertaken by the SE, COMP and PROT.OAL packages.

T-20: We will investigate data ingress and egress for the SDP system; can they co-exist in the various networks identified without loss of performance and/or data? See PDR.02.01 [AD08]. This affects component C.1.5. The work will be done by the COMP, DELIV, PIP and DATA packages, with COMP leading.

T-21: We will provide an analysis of the required size of the compute islands interconnect. See PDR.02.01 - Data transfer rates section [AD08] and Prototyping plan section 4.2 [AD02]. This affects components C.1.5.2, C1.5.3. The work will be undertaken by PIP, COMP and DATA.

T-22: Verify "near" real-time data ingress rate and its impact on the type of switches deployed - Top-of-Rack versus Enterprise Core Switches. This affects component C.1.5.1. The work will be undertaken by COMP.

T-23: We will prototype 100GbE de-multiplexed to 40/25/10GbE performance and behaviour (since SaDT is mandating 100GbE see [AD14]) as this may impact occupancy (achieved line-rate) and/or it may cause buffering issues. This affects components C.1.5.3, C 3.5, C.4.1.3, C.6.7.4. Work will be undertaken by COMP, DATA , LMC, and PIP, with COMP leading.

T-24: We will perform an analysis of computational efficiency on cutting edge hardware. This will allow us to keep track of technology and prepare for future hardware. We will pay particular attention to the following probable limits:

- Island size has an upper bound defined by the interconnect. See T-23.
- Island size has a lower bound defined by storage capacity per node.
- SDP system size has an upper bound defined by the bulk data network. See T-25.
- There may be super-linear scaling with the number of compute Islands; if necessary, we will investigate whether it can be mitigated by increasing island size.

This affects components C.1, C.4.2. The work will be undertaken by COMP, PIP and DATA

T-25: Bulk data network reliability/resilience will be investigated, with particular attention to the following points:

T-25.1: A single switch failure may cause a sizable chunk of observational data to be lost. This is a deliberate design choice, but with careful analysis and design we may limit the impact of this loss.

T-25.2: Is there a configuration possible where a switch loss will cause minimal loss in science data?

T-25.3: Is part of the switch stack important enough to duplicate?

This affects component C.1.5.3. This work will be undertaken by COMP and DATA, with COMP leading.

T-26: Data Distribution Schemes & Sizing of the Computing Platform. We aim to construct a few test designs (Compute / Data Islands, Interconnects together with Data Distribution Schemes) and prototype / benchmark them. This work affects components C.1.1, C.1.5, C.4. The work will be undertaken by COMP, PIP, DATA, LMC, with DATA leading.

T-27: We need to do some prototyping for our hierarchical storage, to ensure we can manage the transition of data from the processing phases into the archive. This is hardware prototyping primarily; though we will want to run tests later with our Data Objects, to test for unexpected behaviour. See also T-69. This affects components C.1.4, C.3.3, C.5. Work will be undertaken by the DELIV, DATA and COMP packages, with DELIV leading.

T-28: We will investigate storage solutions addressing pseudo-real-time buffering of the visibility buffer and archival, as described in section 4.3 of the prototyping plan [AD02]. We intend to investigate enterprise-level disks such as SAS, commodity disks such as SATA, DRAM and Non-volatile (NVRAM) storage, and next-generation and current Parallel File Systems such as Lustre, and object storage solutions such as SWIFT and CEPH. This will affect component C.1.4. The work will be undertaken by PROT.OAL and COMP.

T-29: Further network investigations focussing on high performance networks addressing bulk-data transport and low-latency interconnect, investigating Infiniband and other “proprietary” networking, High Speed Ethernet and Software Defined Networks. The last of these is discussed in more detail in section 4.6 of the prototyping plan [AD02]. This will affect components C.1.5 C.3.5. Work will be undertaken by COMP and PROT.OAL.

T-30: We will examine energy efficient computing and machine cooling, in particular examining low-power microprocessor platforms. This affects component C.1. The work will be undertaken by PROT.OAL and COMP.

T-32: We will provide a more detailed analysis of the issues involved in scaling to SKA2 size as we approach CDR. This will affect components C.1, C.2, C.3 and C.6 in particular. Work will be led by the COMP workpackage.

T-39: We need to investigate our data reordering requirements and then devise prototypes to meet those requirements. It is clear from the Pipelines design document (PDR.02.05 sub-element design PIP [AD09]) that we will need to reorder our data for more efficient processing; it is not yet clear where the balance in our system between the cost of reordering data and the cost of processing is (, see PDR.02.01, [AD08]).
We will perform a cost-benefit analysis on these transposes:

- Is there a potentially less optimal way to do flagging on data ordered differently?
- What is the (science) impact of such a change?
- What is the cost impact of such a change?
- Can we define a data flow that is more optimal?

This affects components C.3.4, C.4, C.6.7, and work will need to be undertaken by the DATA, PIP, COMP and LMC packages, with PIP leading.

T-56: We will prototype candidate compute architectures addressing computational kernels and imaging pipelines. See section 4.5 of the prototyping plan [AD02]. We will examine many-core accelerators (such as GPGPU, FPGA and Xeon Phi), Arithmetic Processing Units comprising a CPU and GPU in one package, and low-power SoC technologies (such as ARM, Atom). This work will be undertaken by the ARCH.SWE and PROT.OAL teams.

T-65: We will investigate the potential to combine the Ingress Bulk Data Transport and Low Latency networks into a single network. This will consider QoS (Quality-of-Service) for different traffic patterns together with exploring alternative topologies. This work will also take into consideration the work on Baseline Dependent Averaging (T-40), and the SDN (T-37). This affects components C.1.5.3, C.3.4, C.4, C.6.7. Work will be undertaken by the DATA, PIP, COMP and LMC packages, with COMP leading.

T-69: We will test how our Data Objects interact with possible hardware and software solutions explored in T-28. This affects components C.1.4, C.3.3, C.5. Work will be undertaken by the DELIV, DATA and COMP packages, with DELIV leading.

8.3 High-level Risks

System simulations: SDPR-M8-P21, SDPR-M8-P22. T-38 is designed to help mitigate or provide more accurate scoring for these risks.

Parametric model verification: SDPR-M8-P23 SDPR-M8-P24. T-57 is designed to help mitigate or provide more accurate scoring for these risks.

Data transfer rates: SDPR-M8-P09, SDPR-M8-P10. T-29 is designed to help mitigate or provide more accurate scoring for these risks.

Fast buffer performance: SDPR-M8-P11, SDPR-M8-P12. T-28 is designed to help mitigate or provide more accurate scoring for these risks.

SDPR-M8-30 GPU Performance. T-24 is designed to help mitigate or provide more accurate scoring for this risk.

9 C.2 Software Compute Platform

9.1 Components

Number	Component Name	Description
C.2	Software Compute Platform	Operating system(s) running on SDP hardware
C.2.1	Compute OS software	Software that provides services to other software components beyond those available from the operating system.
C.2.2	Middleware	Software that provides communication services to other software components
C.2.2.1	Messaging Layer	Software component that handles the generation, collection, aggregation and analysis of system logs.
C.2.2.2	Logging system	Software systems to deploy, maintain and control the hardware in the Science data processor
C.2.2.3	Platform Management System	Software tools designed to facilitate system optimisation efforts
C.2.2.4	System Optimisation Tools	Software that automates the movement of data across various storage tiers
C.2.3	Hierarchical Storage Management software	The suite of software components that support and facilitate software development
C.2.4	Application Development Environment and SDK	Software component responsible for hardware allocation and hardware requirements estimates
C.2.5	Scheduler	

TABLE 3: COMPONENTS OF THE SOFTWARE COMPUTE PLATFORM

9.2 Highlighted Tasks

Tasks are ordered numerically, not by importance. This is not a complete list of tasks needed to develop all components to an appropriate level for CDR.

T-3: Further Operations Management plans will need to be prepared, covering, amongst other items: service levels, system maintenance process and lifecycle management. This affects component C.2.2.3. The work will be undertaken by MGT, PROT.OAL and ARCH.OPS.

T-28: We will investigate storage solutions addressing pseudo-real-time buffering of the visibility buffer and archival, as described in section 4.3 of the prototyping plan [AD02]. We intend to investigate enterprise-level disks such as SAS, commodity disks such as SATA, DRAM and Non-volatile (NVRAM) storage, and next-generation and current Parallel File Systems such as Lustre, and object storage solutions such as SWIFT and CEPH. This will affect component C.1.4. The work will be undertaken by PROT.OAL and COMP.

T-31: We will analyse system optimisation, showing the system can run efficiently. See PDR.02.01, [AD08]. This affects components C1, C.2.2.4. The work will be undertaken by COMP.

T-32: We will provide a more detailed analysis of the issues involved in scaling to SKA2 size as we approach CDR. This will affect components C.1, C.2, C.3 and C.6 in particular. Work will be led by the COMP workpackage.

T-33: We will perform an assessment of the suitability of OpenStack for deployment, including an assessment of whether and how it can integrate with containers. This will affect component C.2.2.3. The work will be undertaken by PROT.OAL.

T-34: We will prototype the LMC workflow and our scheduling. This affects components C.2.5, C.6.2. The work will be done by the PROT.OAL and LMC packages.

T-35: As we move to CDR and beyond, we will want to investigate Exascale system software - how to design, test and develop it. This affects component C.2.2.3. The work will be done by PROT.OAL and COMP.

T-36: We will investigate whether we can manage our science archive functions in the same shared filesystem. This is discussed in section 4.3 of the prototyping plan [AD02]. This affects component C.2.3. The work will be undertaken by DELIV, COMP and DATA, with DATA leading.

T-37: We intend to do prototyping around Software Defined Networks (see PDR.02.01.03 [AD10]). This is described in more detail in section 4.6 of the Prototyping Plan. [AD02]. This affects components C.1.5, C.3.5. The work will be undertaken by the COMP, PIP and DATA packages, with COMP leading.

T-38: Perform full system simulations, moving data through the system, and accepting and handling errors (see section 4.7 of the Prototyping Plan [AD02]). This affects component C.2.2. The work will be undertaken by PROT.OAL and ARCH.SWE.

T-39: We need to investigate our data reordering requirements and then devise prototypes to meet those requirements. It is clear from the Pipelines design document [AD09] that we will need to reorder our data for more efficient processing; it is not yet clear where the balance in our system between the cost of reordering data and the cost of processing is (see PDR.02.01 [AD08].

We will perform a cost-benefit analysis on these transposes:

- Is there a potentially less optimal way to do flagging on data ordered differently?
- What is the science impact of such a change?
- What is the cost impact of such a change?
- Can we define a data flow that is more optimal?

This affects components C.3.4, C.4, C.6.7, and work will need to be undertaken by the DATA, PIP, COMP and LMC packages, with PIP leading.

T-58: Error handling is an important part of such a large system, in that we do not want to swamp the system with errors, but we do want to draw important errors to the attention of system administrators and telescope operators. See PDR.02.04 [AD11] for further discussion. This affects components C.6.3, C.6.4, C.6.5, C.6.6. This work will be undertaken by the LMC and COMP workpackages.

9.3 High-level Risks

Software Defined networks: SDPR-M8-P19, SDPR-M8-P20, SDPR-M7-G20. T-37 is designed to help mitigate or provide more accurate scoring for these risks.

10 C.3 Data Layer

10.1 Components

Number	Name	Description
C.3	Data Layer	The software system responsible for data persistence, data base and data life-cycle services as well as the pipeline framework
C.3.1	Data Manager	Distributed software component that deals with the creation, monitoring and termination of Data Objects.
C.3.1.1	Data Manager interface to Pipeline Components	An interface for controlling processing components, defining input, output and for capturing status information.
C.3.2	Data Life Cycle Manager	Software component that implements a rule driven system for data movement, persistence and release based on hardware parameters and policies.
C.3.3	Science Archive Software	Software component that implements the database and query capabilities required for users to discover and retrieve released science data products.
C.3.4	Local Database Services	Generic database services for the SDP, including the Science Archive, the LMC and potentially source catalogues.
C.3.5	Ingest Data from CSP into Data Layer	High speed data interface between the CSP and SDP.

TABLE 4: COMPONENTS OF THE DATA LAYER

10.2 Highlighted Tasks

Tasks are ordered numerically, not by importance. This is not a complete list of tasks needed to develop all components to an appropriate level for CDR.

T-23: We will prototype 100GbE de-multiplexed to 40/25/10GbE performance and behaviour (since SaDT is mandating 100GbE - see [AD14]) as this may impact occupancy (achieved line-rate) and/or it may cause buffering issues. This affects components C.1.5.3, C 3.5, C.4.1.3, C.6.7.4. Work will be undertaken by COMP, DATA, LMC, and PIP, with COMP leading.

T-27: We need to do some prototyping for our hierarchical storage, to ensure we can manage the transition of data from the processing phases into the archive. This is hardware prototyping primarily; though we will want to run tests later with our Data Objects, to test for unexpected behaviour. This affects components C.1.4, C.3.3, C.5. Work will be undertaken by the DELIV, DATA and COMP packages, with DELIV leading.

T-37: We intend to do prototyping around Software Defined Networks (see PDR.02.01.03 [AD10]). This is describe in more detail in section 4.6 of the Prototyping Plan. [AD02]. This affects components C.1.5, C.3.5. The work will be undertaken by the COMP, PIP and DATA packages, with COMP leading.

T-39: We need to investigate our data reordering requirements and then devise prototypes to meet those requirements. It is clear from the Pipelines design document [AD09] that we will need to reorder our data for more efficient processing; it is not yet clear where the balance in our system between the cost of reordering data and the cost of processing is (see PDR02.01 [AD08]).
We will perform a cost-benefit analysis on these transposes:

- Is there a potentially less optimal way to do flagging on data ordered differently?
- What is the (science) impact of such a change?
- What is the cost impact of such a change?
- Can we define a data flow that is more optimal?

This affects components C.3.4, C.4, C.6.7, and work will need to be undertaken by the DATA, PIP, COMP and LMC packages, with PIP leading.

T-40: We need to implement Baseline Dependent Averaging in software (see PDR.02.05.01 [AD12]). This will involve work on the data structures involved, and the pipelines. This work will be closely linked with T-37 - our software-defined networking work. This affects components C.3.5, C.4, C.6.7.4. Work will be undertaken by the DATA, LMC and PIP packages, with PIP leading.

T-41: We will investigate how certain kinds of failures during processing affect our image quality - what is our fault tolerance for various processing steps (for example gather steps in FFTs)? How can we detect and report these failures? This will interact with our concept of precious and non-precious data, where we will put a higher priority on preventing, or detecting and resolving, problems with precious data. This affects components C3.1.1, C.3.2, C.4.1.6.3. Work will be undertaken by the DATA, PIP and LMC packages, with PIP leading.

T-42: Prototyping of the data flow management system and data flow life-cycle management system: see section 4.4 of [AD02]. This will involve prototyping the deployment of physical graphs, the messaging system used by Data Objects, and the system used to persist Data Object state. This will use the work from T-39. This affects component C.3.1. This will be worked on by the DATA, LMC, PIP and COMP packages, with DATA leading.

T-43: We will identify existing graph execution engines and evaluate them using realistic workflows. See s.4.8 of [AD02] for more detail. This affects component C.3.1. The work will be undertaken by the DATA, LMC and PIP workpackages, with DATA leading.

T-44: We will identify and evaluate existing systems that can support the mapping between the logical graphs and the physical graphs. It will also explore potential technologies for the generation of the logical graphs. This is described more fully in section 4.9 of [AD02]. This affects component C.3.1. The work will be undertaken by the DATA and LMC workpackages, with DATA leading.

T-50: We will develop a profiler of IO and Compute behaviour for Data Objects, as described in section 4.12 of the Prototyping Plan [AD02]. This affects component C.3.1. This work will be done by DATA and PIP, led by DATA.

T-59: We will conduct an analysis of our Telescope Model management (global and local), as noted in [AD11]. (pdr.02.04) This affects component C.6.1. This work will be led by LMC, and supported by the PIP and DATA packages.

T-65: We will investigate the potential to combine the Ingress Bulk Data Transport and Low Latency networks into a single network. This will consider QoS (Quality-of-Service) for different traffic patterns together with exploring alternative topologies. This work will also take into consideration the work on Baseline Dependent Averaging (T-40), and the SDN (T-37). This affects components C.1.5.3, C.3.4, C.4, C.6.7. Work will be undertaken by the DATA, PIP, COMP and LMC packages, with COMP leading.

T-67: Verify that the Pipelines components can be made stateless. This affects component C.4. The work will be undertaken by DATA and PIP, and led by PIP.

T-69: We will test how our Data Objects interact with possible hardware and software solutions explored in T-28. This affects components C.1.4, C.3.3, C.5. Work will be undertaken by the DELIV, DATA and COMP packages, with DELIV leading.

10.3 High-level Risks

Tasks T-50, T-42 and T-39 are designed to help mitigate or provide more accurate scoring for the following risks: SDPR-M8-P07, SDPR-M8-P08, SDPR-M7-G03.

SDPR-M7-G17 (M8-P32) - Efficiency (25%) unobtainable due to reduced Data Flow Manager in SKA1. T-42 addresses this risk.

SDPR-M7-G18 - Resilience impacted due to reduced Data Flow Manager in SKA1. T-42 is designed to help mitigate or provide more accurate scoring.

T-43 is designed to help mitigate or provide more accurate scoring for the following risks: SDPR-M8-P13, SDPR-M8-P14

T.44 is designed to help mitigate or provide more accurate scoring for the following risks: SDPR-M8-P15, SDPR-M8-P16

T-39 is designed to help mitigate or provide more accurate scoring for the following risks: SDPR-M8-P22

11 C.4 Pipeline Components

11.1Components

Number	Component Name	Description
C.4	Pipeline Components	Software components that, once executed one after the other, form a Data Processing Pipeline
C.4.1	Processing Library	Software Library of processing components and supporting software
C.4.1.1	Non-imaging processing components	Software components for processing of voltage domain time series data
C.4.1.2	RM Synthesis Component	Software Component for RM Synthesis of Image Cubes
C.4.1.3	Ingest Components	Software Components for the Ingest Pipeline
C.4.1.3.1	Flagging Component	Software Component for Flagging of RFI
C.4.1.3.2	Demixing Component	Software Component for removal of bright sources outside the FoV
C.4.1.4	Calibration components	Software Components for performing the Calibration on visibility data
C.4.1.5	Source Finding Components	Software Components for Finding Sources from Image Cubes
C.4.1.6	Imaging Components	Software Components for transforming visibility data into image cubes
C.4.1.6.1	Gridding	Software Component for putting visibility data on a regular grid
C.4.1.6.2	Deconvolution	Software Component for decovolving Image Cubes
C.4.1.6.3	FFT	Software Component for Fast Fourier Transformation
C.4.2	Algorithmic Software	Numerical and algorithmic software libraries which are used by one or more Processing Components but which present a conventional interface, e.g., a C-like interface, rather than a dataflow interface.
C.4.3	Sky Model Use and Creation	The Sky Model contains the (parameterized) known

		sky.
C.4.4	QA components	Software processing components for producing quality assurance summaries from the observed data, science-ready results and intermediate pipeline processing stages, and for calculating the quantitative performance metrics .
C.4.4.1	Global Sky Model QA	Quality Assurance of the Sky Model components

TABLE 5: COMPONENTS OF THE PIPELINE

11.2 Highlighted Tasks

Tasks are ordered numerically, not by importance. This is not a complete list of tasks needed to develop all components to an appropriate level for CDR.

T-10: Develop requirements around calibration in collaboration with SE and SKAO, as the SKAO provides more information on their calibration strategy. This affects component C.4.1.4

T-11: Develop requirements around the EoR pipeline in collaboration with the SE package. This affects component C.4.

T-23: We will prototype 100GbE de-multiplexed to 40/25/10GbE performance and behaviour (since SaDT is mandating 100GbE - see [AD14]) as this may impact occupancy (achieved line-rate) and/or it may cause buffering issues. This affects components C.1.5.3, C 3.5, C.4.1.3, and C.6.7.4. Work will be undertaken by COMP, DATA, LMC, and PIP, with COMP leading.

T-26: Data Distribution Schemes & Sizing of the Computing Platform. We aim to construct a few test designs (Compute / Data Islands, Interconnects together with Data Distribution Schemes) and prototype / benchmark them. This work affects components C.1.1, C.1.5, C.4. The work will be undertaken by COMP, PIP, DATA & LMC, with DATA leading.

T-32: We will provide a more detailed analysis of the issues involved in scaling to SKA2 size as we approach CDR. This will affect components C.1, C.2, C.3 and C.6 in particular. Work will be led by the COMP workpackage.

T-39: We need to investigate our data reordering requirements and then devise prototypes to meet those requirements. It is clear from the Pipelines design document [AD09] that we will need to reorder our data for more efficient processing; it is not yet clear where the balance in our system between the cost of reordering data and the cost of processing is (see PDR.02.01 [AD08]).
We will perform a cost-benefit analysis on these transposes:

- Is there a potentially less optimal way to do flagging on data ordered differently?
- What is the (science) impact of such a change?
- What is the cost impact of such a change?
- Can we define a data flow that is more optimal?

This affects components C.3.4, C.4, C.6.7, and work will need to be undertaken by the DATA, PIP, COMP and LMC packages, with PIP leading.

T-40: We need to implement Baseline Dependent Averaging in software (see PDR.02.05.01 [AD12]). This will involve work on the data structures involved, and the pipelines. This work will be closely linked with T-37 - our software-defined networking work. This affects components C.3.5, C.4, C.6.7.4. Work will be undertaken by the DATA, LMC and PIP packages, with PIP leading.

T-41: We will investigate how certain kinds of failures during processing affect our image quality - what is our fault tolerance for various processing steps (for example gather steps in FFTs)? How can we detect and report these failures? This will interact with our concept of precious and non-precious data, where we will put a higher priority on preventing, or detecting and resolving, problems with precious data. This affects components C3.1.1, C.3.2, C.4.1.6.3. Work will be undertaken by the DATA, PIP and LMC packages, with PIP leading.

T-45: We will investigate the requirements around postage stamp creation. See [AD09] (PDR.02.05, p.20). This affects component C.4. The work will be undertaken by the PIP and DELIV packages, with PIP leading.

T-46: Investigate the use of Sparse Fourier Transforms (sFFTs) for the slow transients pipeline (see PDR.02.05 p.24 [AD09]). This affects components C.4, C.6.7.3. The work for this will be done by PIP and LMC, with PIP leading.

T-47: We will perform analysis that will allow us to advise on deconvolution algorithm selection, see PDR.02.05 p.42-3 [AD09]. This affects component C.4.1.6.2. The work will be led by PIP, and supported by COMP.

T-48: We will perform analysis of Discrete Fourier Transform algorithms. This will affect component C.4.1.6.3. The work will be led by PIP, and supported by COMP.

T-49: Slow transient latency requirements are currently not well-defined. We will investigate this, including how we will communicate with TM. This affects components C.4 and C.6.7.3. The work for this will be undertaken by LMC and PIP.

T-53: We will in general be performing benchmarking of the efficiency of implementations of algorithms on certain types of hardware. This affects components C.4, C.4.2. The work will be undertaken by PIP, COMP and DATA, with PIP leading.

T-54: We will be undertaking a detailed design of all pipelines components. (This may be an analysis of existing components from LOFAR, ASKAP and other instruments, with a demonstration that they scale to SKA1 levels. This affects component C4. The work will be done by PIP.

T-55: We will investigate how much calibration will be done for the Fast Imaging Pipeline and CSP beamformer feedback, in collaboration with CSP and TM. This will affect component C.4.1.4. The work will be led by PIP and supported by LMC.

T-56: We will prototype candidate compute architectures addressing computational kernels and imaging pipelines. See section 4.5 of the prototyping plan [AD02]. We will examine many-core accelerators (such as GPGPU, FPGA and Xeon Phi), Arithmetic Processing Units comprising a CPU and GPU in one package, and low-power SoC technologies (such as ARM, Atom). This work will be undertaken by the ARCH.SWE and PROT.OAL teams.

T-57: We are in the process of verifying our parametric model. See section 4.1 of the prototyping plan [AD02]. This affects components C.1, C.4. The work is being done by ARCH.SWE.

T-65: We will investigate the potential to combine the Ingress Bulk Data Transport and Low Latency networks into a single network. This will consider QoS (Quality-of-Service) for different traffic patterns together with exploring alternative topologies. This work will also take into consideration the work on Baseline Dependent Averaging (T-40), and the SDN (T-37). This affects components C.1.5.3, C.3.4, C.4, C.6.7. Work will be undertaken by the DATA, PIP, COMP and LMC packages, with COMP leading.

T-67: Verify that the Pipelines components can be made stateless. This affects component C.4. The work will be undertaken by DATA and PIP, and led by PIP.

11.3 High-level Risks

System Simulations: SDPR-M8-P21, SDPR-M8-P22. T-67 and T-38 will address these risks.
Key algorithms: SDPR-M8-P17, SDPR-M8-P18. T-53 addresses this risk.
Parallel environments: SDPR-M7-G21. T-53 addresses this risk.

12 C.5 Data Delivery Platform

12.1Components

Number	Component Name	Description
C.5	Data Delivery Platform	A software stack whose purpose is to enable users to search for and access data products. It includes services to tansport data to specified remote sites and to perform remote visualisation of data products. This will be deployed at SDP sites and be available to deploy other Regional Centres.
C.5.2	Tiered Data Transfer Service	Service for managing the movement of data objects to specified remote sites (e.g., Regionial Centres)
C.5.2.1	LMC interface for data scheduling & reporting	Interface for LMC to request data movement using the Tiered Data Transfer Service, reporting back information from the data delivery services and for these services to be able to access Authn and AuthZ information
C.5.3	User Portal	Web based platform hosting user tools and services related to the data delivery subsystem
C.5.3.1	Astronomer/ Telescope Operator/ Public Interfaces	Web GUI and API interfaces to the data delivery subsystems
C.5.4	Data Discovery Service	Service to enable searching for data objects based on information describing the objects
C.5.4.1	IVOA Models and Services	
C.5.5	Data Visualisation Service	Service that enables remote visualisation of data products
C.5.6	Regional Centre Interface	The interface to move data to the Regional Centres

TABLE 6: COMPONENTS OF THE DATA DELIVERY PLATFORM

12.2 Highlighted Tasks

Tasks are ordered numerically, not by importance. This is not a complete list of tasks needed to develop all components to an appropriate level for CDR.

T-45: We will investigate the requirements around postage stamp creation. See [AD09] (PDR.02.05, p.20). This affects components C.4, C.5. The work will be undertaken by the PIP and DELIV packages, with PIP leading.

T-60: We will investigate what data visualisation services we need to provide, where we should provide them (at the SDP and/or at RCs), and how much visualisation software should be provided by the SDP itself. See PDR.02.03 [AD13]. This affects component C.5.5. The work will be undertaken by DELIV.

T-61: We will be testing for where there could be limitations for handling Radio Astronomy data with the IVOA tools and services currently available. This affects component C.5.4.1. The work will be undertaken by DELIV.

T-62: We will prototype Portal / Web interfaces to SDP data and metadata. This affects component C.5.3.1. This work will be undertaken by DELIV.

T-63: Once Authentication and Authorisation policies are determined by SKAO, we will be prototyping relevant schemes for A&A for DELIV. This affects component C.5. The work will be undertaken by DELIV.

T-64: We will investigate the different tools that are currently available for transferring and managing the transfer of data on wide area networks and prototype them. See section 4.11 of the prototyping plan [AD02]. The work will be undertaken by DELIV.

T-66: We will provide an evaluation of how Regional Centres for the SDP will provide us with better data delivery services. This will be informed by the work on T-64. This affects component C.5. The work will be undertaken by DELIV.

T-69: We will test how our Data Objects interact with possible hardware and software solutions explored in T-28. This affects components C.1.4, C.3.3, C.5. Work will be undertaken by the DELIV, DATA and COMP packages, with DELIV leading.

12.3 High-level Risks

SDPR-M8-P37 Data delivery prototyping takes longer than expected. No tasks address this risk.

13 C.6 Local Monitoring and Control

13.1Components

Number	Component Name	Description
C.6	Local Monitoring and Control	Provides bridge between external monitor and control requirements and internal sub-elements.
C.6.1	Local Telescope Model	Storage of meta-data relevant to current observation. Wide variety of data is stored from static configuration information through to components of the Local Sky Model.
C.6.2	Data Flow Manager (LMC)	Responsible for constructing physically realisable data flow graphs for deployment into the data layer.
C.6.3	QA monitoring	Aggregates and interprets lower level QA data arriving from the various pipelines.
C.6.4	User Interfaces	Visual environment for presenting QA data. Allows user interaction with the metric calculations.
C.6.5	Master Controller and error handling	Single instance central controller for all instantiated SDP capabilities. Main point of contact for TM.
C.6.6	Event monitoring and logging	Logging, alarm and event handling services. Includes aggregation and granular control.
C.6.7	Data Flow Models	Model describing the sequence in which Pipeline Components are to be executed
C.6.7.1	Non-imaging pipeline	Voltage domain time series processing
C.6.7.1.1	Pulsar Timing Post Processing	Timing of known Pulsars
C.6.7.1.2	Pulsar Search Post Processing	Post-processing of Pulsar Search candidates
C.6.7.1.3	Non-imaging transient post processing	Single Pulse / Fast Transients
C.6.7.2	Continuum Imaging Pipeline	Continuum Imaging Pipeline
C.6.7.3	Fast Imaging (Slow Transients)	Real-time continuum imaging for slow transient detection
C.6.7.4	Ingest Pipeline	Data Reception and pre-processing

C.6.7.5	Science Analysis Pipeline	Science Analysis pipelines
C.6.7.5.1	Postage Stamp Source Detection	Source Finding component
C.6.7.5.2	RM Synthesis	RM-Synthesis pipeline
C.7.6.7	Calibration Pipeline	Calibration pipeline for EoR processing
C.6.7.7	Real-time calibration pipeline	Real-time calibration pipeline for feedback to CSP
C.6.7.8	Spectral Line Imaging Pipeline	Spectral Line Imaging Pipeline

TABLE 7: COMPONENTS OF LOCAL MONITORING AND CONTROL

13.2 Highlighted Tasks

Tasks are ordered numerically, not by importance. This is not a complete list of tasks needed to develop all components to an appropriate level for CDR.

T-2: We will investigate the trade-offs between standardisation of compute islands and compute island components, and the flexibility that compute islands of various technology capabilities (such as having compute nodes configured with different numbers or types of GPU, procured in partial upgrade cycles) or with different configurations offer. We will work on the following sub-tasks towards CDR; however, we many not have complete plans before the Construction phase begins.

T-2.1: Investigate multiple configurations of self-contained compute islands. This affects component C.1.1. The work will be undertaken by the COMP package.

T-2.2: Provide an analysis of the benefits of standardisation, and why it is preferred from an operational point of view. This affects component C.1.1. The work will be undertaken by the ARCH.OPS workpackage.

T-2.3: Provide an analysis of the benefits of partial upgrades allowing for a natural migration of data in science archive. This task along with T-2.2 will allow the management team to present options for the operational management of the data centres. This affects components C.1.1 and C.4. The work will be done by the MGT and ARCH.OPS packages.

T-2.4: Heterogeneous Compute Island Configurations make LMC/scheduler task more difficult. This affects components C.6, C.1. The work for this will be led by COMP and involve LMC.

T-2.5: We recognise that the optimal combination of standardisation vs. flexibility vs. optimisation may be different per telescope, and will provide preliminary analyses to reflect this. This affects component C.1.1. The work will be done by the ARCH.OPS workpackage.

T-23: We will prototype 100GbE de-multiplexed to 40/25/10GbE performance and behaviour (since SaDT is mandating 100GbE see [AD14]) as this may impact occupancy (achieved line-rate) and/or it may cause buffering issues. This affects components C.1.5.3, C 3.5, C.4.1.3, C.6.7.4. Work will be undertaken by COMP, DATA, LMC, and PIP, with COMP leading.

T-32: We will provide a more detailed analysis of the issues involved in scaling to SKA2 size as we approach CDR. This will affect components C.1, C.2, C.3 and C.6 in particular. Work will be led by the COMP workpackage.

T-34: We will want to prototype the LMC workflow and our scheduling. This affects components C.2.5, C.6.2. The work will be done by the PROT.OAL and LMC packages.

T-39: We need to investigate our data reordering requirements and then devise prototypes to meet those requirements. It is clear from the Pipelines design document [AD09] that we will need to reorder our data for more efficient processing; it is not yet clear where the balance in our system between the cost of reordering data and the cost of processing is (see PDR.02.01 [AD08]).
We will perform a cost-benefit analysis on these transposes:

- Is there a potentially less optimal way to do flagging on data ordered differently?
- What is the (science) impact of such a change?
- What is the cost impact of such a change?
- Can we define a data flow that is more optimal?

This affects components C.3.4, C.4, C.6.7, and work will need to be undertaken by the DATA, PIP, COMP and LMC packages, with PIP leading.

T-40: We need to implement Baseline Dependent Averaging in software (see PDR.02.05.01 [AD12]). This will involve work on the data structures involved, and the pipelines. This work will be closely linked with T-37 - our software-defined networking work. This affects components C.3.5, C.4, C.6.7.4. Work will be undertaken by the DATA, LMC and PIP packages, with PIP leading.

T-46: Investigate the use of Sparse Fourier Transforms (sFFTs) for the slow transients pipeline See PDR.02.05 p.24 [AD09] This affects components C.4, C.6.7.3. The work for this will be done by PIP and LMC, with PIP leading.

T-49: Slow transient latency requirements are currently not well-defined. We will investigate this, including how we will communicate with TM. This affects components C.4 and C.6.7.3. The work for this will be undertaken by LMC and PIP.

T-58: Error handling is an important part of such a large system, in that we do not want to swamp the system with errors, but we do want to draw important errors to the attention of system administrators and telescope operators. See PDR.02.04 [AD11] for further discussion. This affects components C.6.3,C.6.4, C.6.5, C.6.6. This work will be undertaken by the LMC and COMP workpackages.

T-59: We will conduct an analysis of our Telescope Model management (global and local), as noted in [AD11]. (PDR.02.04) This affects component C.6.1. This work will be led by LMC, and supported by the PIP and DATA packages.

T-65: We will investigate the potential to combine the Ingress Bulk Data Transport and Low Latency networks into a single network. This will consider QoS (Quality-of-Service) for different traffic patterns together with exploring alternative topologies. This work will also take into consideration the work on Baseline Dependent Averaging (T-40), and the SDN (T-37). This affects components C.1.5.3, C.3.4, C.4, C.6.7. Work will be undertaken by the DATA, PIP, COMP and LMC packages, with COMP leading.

13.3 High-level Risks

There are no high-level risks associated with C.6; the data flow risks are covered by the work done for the C.3 component.

SCIENCE DATA PROCESSOR

PDR.14 DEVELOPMENT PLAN (PDR TO CDR)

Document number……SKA-TEL-SDP-0000053
Context……MGT
Revision……01
Author……SDP Consortium
Release Date……2014-02-09
Document Classification……Unrestricted
Status……Draft

Name	Designation	Affiliation
Signature & Date:		

Name	Designation	Affiliation
Signature & Date:		

Document No: SKA-TEL-SDP-0000053
Revision: 01
Release Date: 2015-02-09

Unrestricted
Author: SDP
Page 1 of 38

PDR14 Development Plan (1)

EchoSign Document History February 10, 2015

Created:	February 09, 2015
By:	Verity Allan (vla22@mrao.cam.ac.uk)
Status:	SIGNED
Transaction ID:	XJEZW8Z4CXX3U77

“PDR14 Development Plan (1)” History

Document created by Verity Allan (vla22@mrao.cam.ac.uk)

February 09, 2015 - 4:50 PM GMT - IP address: 131.111.185.15

Document emailed to Ian Cooper (ian.cooper@uwa.edu.au) for signature

February 09, 2015 - 4:51 PM GMT

Document viewed by Ian Cooper (ian.cooper@uwa.edu.au)

February 09, 2015 - 10:12 PM GMT - IP address: 101.186.41.147

Document e-signed by Ian Cooper (ian.cooper@uwa.edu.au)

Signature Date: February 09, 2015 - 10:13 PM GMT - Time Source: server - IP address: 101.186.41.147

Document emailed to Paul Alexander (pa@mrao.cam.ac.uk) for signature

February 09, 2015 - 10:13 PM GMT

Document viewed by Paul Alexander (pa@mrao.cam.ac.uk)

February 10, 2015 - 8:49 AM GMT - IP address: 131.111.185.15

Document e-signed by Paul Alexander (pa@mrao.cam.ac.uk)

Signature Date: February 10, 2015 - 8:49 AM GMT - Time Source: server - IP address: 131.111.185.15

Signed document emailed to Paul Alexander (pa@mrao.cam.ac.uk), Ian Cooper (ian.cooper@uwa.edu.au) and Verity Allan (vla22@mrao.cam.ac.uk)

February 10, 2015 - 8:49 AM GMT