SKA1 SDP Performance Prototype Platform (P3-ALaSKA) Prototyping Report

Document Number..... SKA-TEL-SDP-0000151
Document Type..... REP
Revision..... 02
Author..... J. Taylor
Date..... 2019-03-15
Document Classification..... UNRESTRICTED
Status.....Released

Name	Designation	Affiliation	Signature	
Authored by:				
John Taylor	SDP Team	University of Cambridge	John Taylor (Mar 14, 2019)	
			Date:	
Owned by:				
John Taylor	SDP Team	University of Cambridge	John Taylor (Mar 14, 2019)	
			Date:	
Approved by:				
Jeremy Coles	SDP Project Manager	University of Cambridge	Jeremy Coles Jeremy Coles (Mar 14, 2019)	
			Date:	
Released by:				
Paul Alexander	SDP Project Lead	University of Cambridge	Paul Alexander Paul Alexander (Mar 15, 2019)	
			Date:	

DOCUMENT HISTORY

Revision	Date Of Issue	Engineering Change Number	Comments
01	2018-10-31		Prepared for M21, SDP CDR review
02	2019-03-15	ECP-SDP-190001	Prepared for M22 SDP Closeout CDR OARs addressed in this document SDPCDR-98 Unreadable Diagram SDPCDR-99 Give due credit when quoting/ copying bulk text from external source SDPCDR-115 Figure-2

DOCUMENT SOFTWARE

	Package	Version	Filename
Word processor	Google Docs		SKA-TEL-SKO-0000000-01_GenDocTemplate
Block diagrams			
Google docs Add-ons	Cross Reference Table of contents List of figures		Used for figure & table numbering and references. Used for heading numbering. Used to generate list of figures and tables

ORGANISATION DETAILS

Name	SDP Consortium
Lead Organisation	The Chancellor, Masters and Scholars of the University of Cambridge The Old Schools Trinity Lane Cambridge CB1 ITN United Kingdom
Website	www.ska-sdp.org

Table of Contents

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

API	Application Processor Interface
ARL	Algorithmic Reference Library
C&C	Component and Connector
CERN	European Organization for Nuclear Research
CDR	Critical Design Review

CLI	Command Line Interface
COTS	Commercial off the shelf
EF	Execution Framework
ELK	Elasticsearch, Logstash and Kibana
Gb/s	Gigabits per second
GPU	Graphics Processor Unit
HTE	High Throughput Ethernet
I/O	Input/Output
JNI	Java Native Interface
LLN	Low Latency Network
P3	Performance Prototype Platform
RADI	Redundant Array of Inexpensive Disks
RDMA	Remote Direct Memory Access
RoCE	RDMA Over Converged Ethernet
SSH	Secure Socket Shell
SDP	Science Data Processor
SIP	SDP Integration Prototype
SKA	Square Kilometre Array
SKAO	SKA Office

1 Introduction

This document provides a report of the prototyping work performed on the P3-AlaSKA Performance Prototype Platform to help better understand and evolve the architecture of the SDP Operational System in relation to its hardware needs, and how that hardware can be abstracted via a set of services, called Platform Services.

Platform Services is a key part of the Science Data Processor (SDP) architecture. How Platform Services fits into the wider SDP Architecture is detailed in the SDP Operation System C&C [AD1]. The details of how Platform Services connects to the rest of the system is covered in the Platform Services C&C [AD1] and the details of how the different components of the Platform interact are covered in the Platform Services Module View [AD1].

The document is structured as an overview of the P3 environment and prototyping and references specific documents which provide more detail on components and modules of the SDP Architecture and the respective prototyping activities. In particular these are the following memos:

- P3-Alaska OpenStack Prototyping [RD1]
- P3-AlaSKA Container Orchestration and Compute Provisioning Interface [RD2]
- P3-AlaSKA Monitoring and Logging [RD3]

These memos follow a convention of “Motivation and Scope”, “Architectural Context” and “Important Results and Lessons Learned”.

In addition to the Architecture work, additional work on the Buffer component, the use of P3-AlaSKA by the SIP team and Execution Framework team are described in the following, respectively:

- Buffer Modelling and Prototyping [RD4]
- SDP Integration Prototype Report [RD9]
- Execution Frameworks Prototyping Report [RD10]

Further information is also available on potential implementations of Platform Services in the following memos:

- Cloud Native Applications on the SDP Architecture [RD5]
- Monitoring and Logging for the SDP Architecture [RD6]
- Apache Kafka for and SDP log-based Architecture [RD7]

2 Motivation and scope

The Performance Prototype Platform (P3-AlaSKA) provides a prototyping framework for use by the SDP consortium. The compute, storage and networking is realised by a bare metal OpenStack, multi-tenanted solution to produce a flexible software-defined infrastructure that can easily scale to accommodate particular prototyping activities. A number of execution environments are catered for, as described in the Execution Framework Prototyping Report [RD10], such as docker-swarm (ostensibly to support SIP activities as described in [RD9]), Spark-as-a-service and Slurm-as-a-service for DASK and traditional MPI workloads. In addition to these platforms, tenants can bring their own environments for testing purposes.

P3 is also being used to prototype OpenStack itself in the guise of a potential candidate for a number of Platform Services components by investigating core OpenStack services for infrastructure (eg. Ironic, Ansible) as well as Monitoring and Logging (OpenStack Monasca), container orchestration (OpenStack Magnum) and higher level shared services (e.g. File-system-as-a-service through OpenStack Manila and object storage for Buffer provisioning) and packaged execution environments such OpenStack Sahara for Spark-as-a-service. These services are inherent to OpenStack

Another key aspect of prototyping has been working with both CERN and the wider OpenStack community (via the OpenStack Scientific SIG) to build on experiences with OpenStack and related technologies to deliver useful science. This is particularly relevant with respect to the similarity of the CERN OpenStack Cloud servicing the Tier-0 aspects of the Large Hadron Collider and the proposed SKA regional centres. This continues to be on-going work but is described herein for context.

This report focuses on the aspects of the support of Platform Services in the adoption of OpenStack and other open-source packages in the realisation of the SDP Architecture. The use of the system for activities, particularly in respect of support of the System Integration Prototype are described elsewhere.

3 Performance Prototype Platform

The diagram below (Figure 1) provides a schematic representation of the P3 Hardware used for much of the prototyping work summarised here. While P3 does not reflect the scale of the proposed SDP, it does represent a facsimile of a Compute Rack and as such is useful in drawing comparisons between potential elements of the SDP hardware and non-domain software.

Thus, in relation to the SDP Product Breakdown Structure [AD3], the following mapping of P3 components to SDP hardware and software is drawn:

- 100G Infiniband. The Low-Latency Network (LLN) used for both communication and I/O to the Buffer.
- 25G Ethernet. The High Throughput Ethernet (HTE) network used for both communication (specifically on ingest) and I/O to the Buffer and Long Term Storage
- High Storage nodes used as disaggregated Storage Nodes.
- High Memory nodes. For testing applications.
- GPU nodes. For testing applications.
- Compute Nodes. For testing applications as well as supporting hyper-converged storage.
- Ceph Cluster. for Storage Provisioning testing and /home

3.1 Hardware

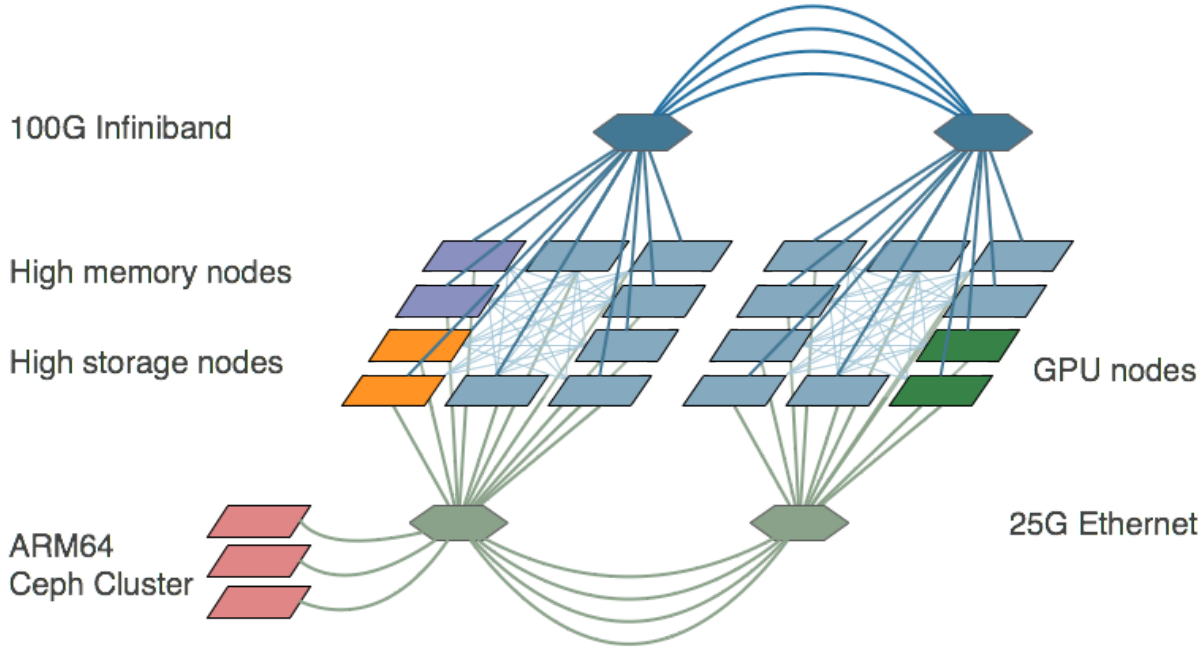

Figure 1: Schematic Representation of P3

3.2 Software

The P3 hardware is provisioned, managed and controlled by OpenStack which is a set of open source software and tools for creating private and public clouds. OpenStack software controls large pools of compute, networking and storage throughout a data centre, managed through a dashboard or via the OpenStack API. A “map” identifying core OpenStack services and their function can be found here and their relation to other projects such as Kolla-Ansible and Bifrost.

The decision was made to partition the P3-ALaSKA system into two separate compute systems, each with its own OpenStack control plane: namely P3 itself together with a staging and development cluster called *Alt-1*. The *Alt-1* system comprises of a single OpenStack controller host and two bare metal compute hosts. The *Production* system (viz. P3 as described above) comprises an OpenStack controller host, a monitoring host, and the remaining compute and storage hosts.

At the heart of the OpenStack software used on P3 is OpenStack Kayobe [RD14]. Of particular note here, is that Kayobe enables deployment of containerised OpenStack to bare metal, a critical aspect of realising performance within an otherwise Cloud software environment which would be impacted by the overheads of virtualisation. Containers offer a compelling solution for isolating OpenStack services, but running the control plane on an orchestrator such as Kubernetes or Docker Swarm adds significant complexity and operational overheads.

Much of the development of Kayobe has been in support of the anticipated functionality and performance of the SKA (hence OpenStack *â la* SKA). Further information on OpenStack Kayobe can be found in [RD14].

The P3-Alaska system is a managed service operated by the University of Cambridge Information Services and supported and operated by the University. P3-AlaSKA is used by SIP, NZA, ICRAR, LOFAR and the University of Cambridge.

3.3 SDP Architecture

3.3.1 Architectural Context

The Platform C&C View [AD1], duplicated for context here in Figure 3, sets the discussion for the Prototyping activities covered here and potential implementations of specific models as represent by the Platform Service Module and Dependency View [AD1].

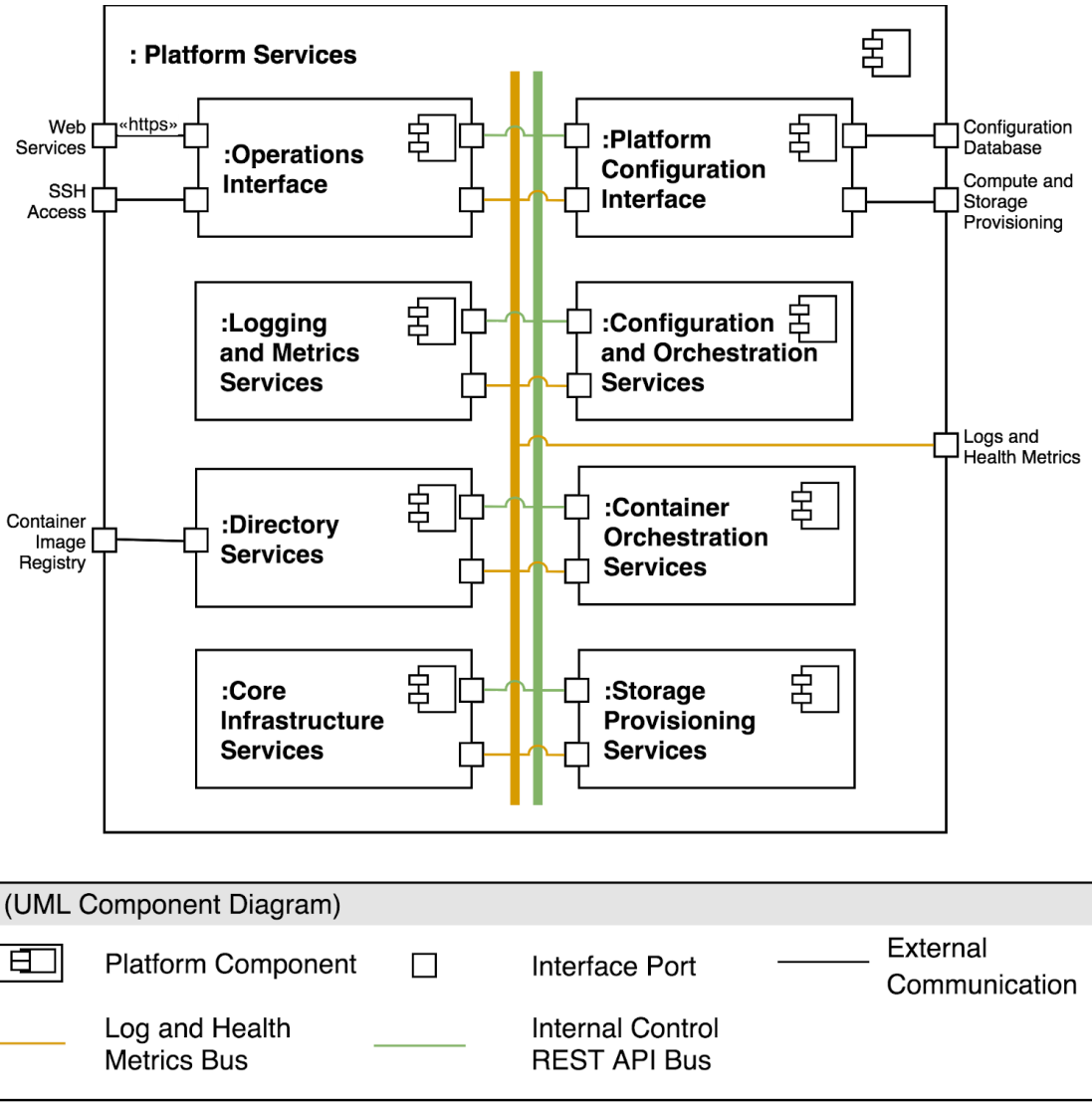

Figure 2: Platform Services C&C View Primary Representation

3.3.2 Architectural Impact

The main architectural impact of the P3-AlaSKA work can be grouped into the following aspects in respect of the Element Catalogue of the Platform C&C View. Conclusions and major findings have been summarised from the various P3-AlaSKA prototyping memos, which describe the work in detail and also discuss some of the difficulties overcome and lessons learned. These are summarised in Table 1 with the appropriate reference:

Component	Conclusion	Reference
Core Infrastructure Services	→ OpenStack private cloud exposing bare metal resources, and multiple networks, including 100Gb/s Infiniband and 25Gb/s Ethernet (with RoCE) together with 10Gb/s management network → P3-AlaSKA up and running and well used by the SIP team and others	P3-Alaska OpenStack Prototyping [RD1]

	→ OpenStack Kayobe makes use of the widely adopted OpenStack Kolla Containers and OpenStack Kolla-Ansible orchestration scripts to deploy OpenStack and do the initial configuration of the network to help discover new nodes → Upgrade from Ocata to Pike and Pike to Queens did not disrupt the workloads running on the system, shows the sustainability of the platform → Hardware all exposed using OpenStack Ironic's bare metal provisioning → OpenStack Neutron's Generic Switch driver used to automate network changes → SoftIron Ceph cluster used to provide OpenStack required storage	
Configuration and Orchestration Services	→ OpenStack Kayobe makes use of OpenStack Kolla and Kolla-Ansible to provision and upgrade OpenStack services. → Heat + Ansible for appliances, such as the Docker Swarm cluster created for SIP as a prototype for the Compute Provisioning Interface → Ansible scripts for provisioning storage systems on the OpenStack provisioned hardware, such as Gluster, Ceph and BeeGFS (with some preliminary work on Lustre)	P3-AlaSKA Container Orchestration and Compute Provisioning Interface [RD2]
Container Orchestration Services	→ OpenStack Magnum has been used to create Docker Swarm and Kubernetes Container Orchestration Engine Clusters → Collaborated with CERN on how they make use of Magnum	P3-AlaSKA Container Orchestration and Compute Provisioning Interface [RD2]
Logging and Metrics Service	→ OpenStack Monasca builds on Apache Kafka, ELK stack , Grafana and InfluxDB to provide multi-tenant Logging and Metrics → Integrating workload specific metrics into Monasca using statsd → Investigated deploying Prometheus with Grafana into tenant environments as an alternative to Monasca	P3-AlaSKA Monitoring and Logging [RD3] Monitoring and Logging for the SDP Architecture [RD6] Apache Kafka for and SDP log-based Architecture {RD7]
User Interface Services	→ SSH gateway to directly access the (mostly) isolated environment → OpenStack Horizon to inspect the current state of the Core Infrastructure Service → Grafana as a view into logging and metrics data → Kibana for deeper log analysis → Rundeck to deliver "operations as a service"	P3-Alaska OpenStack Prototyping [RD1]

Storage Provisioning Interface	→ OpenStack Manila providing on-demand access to unique namespace, prototyped with OpenStack → Integrated with existing Kayobe and kolla-ansible work to get Manila installed → Experimental work around Manila access to pre-defined GlusterFS volumes for cluster file systems → System worked well to provide on-demand access to SoftIron based Ceph cluster → Collaborated with CERN on how they make use of OpenStack Manila and Kubernetes → Ansible also used to automate the creation of parallel file systems needed by the SIP and Buffer prototype efforts	P3-AlaSKA Container Orchestration and Compute Provisioning Interface [RD3] Ansible modules to mount Manila provided volumes [RD8]
Compute Provisioning Interface	→ Expose the hardware to the SDP Operational System. → Used RunDeck triggered Ansible scripts to create a Container Orchestration Engine Cluster (created using OpenStack Magnum) and mounting all the storage that is created via either Manila or Ansible (as part of the Storage Provisioning Interface) → Ansible used to create packaged version of various filesystems	P3-AlaSKA Container Orchestration and Compute Provisioning Interface [RD2]

Table 1: Main conclusions from P3-AlaSKA Prototyping Work with References to Further Reading

3.3.3 Lessons Learned

The individual memos referred to in the Introduction have more detailed information on specific "Lessons Learned" but in general the following aspects are worth highlighting here.

- All development of OpenStack and related open-source software has been performed in the context of upstream ensuring that no technical debt is incurred. This resulted in the last OpenStack Queens release having no extant Kayobe patches. Further development should continue in this vain.
- The Scientific SIG [RD12] has proven an invaluable tool in discussions around shared experiences in the application of OpenStack cloud to the Scientific and High Performance and Throughput Computing application.
- The collaboration with CERN and other organisations such as STFC in the UK[RD13] has ensured that developments have followed "Best Practices" in terms of integration and development around particular OpenStack projects, notably the Compute Storage Interface (CSI) for Kubernetes .
- The decomposition of the P3-AlaSKA system into a Production and smaller Development Cluster allowed testing of various new deployments before going live.
- Regular meetings between the users (notably SIP) of P3-AlaSKA and its Operators ensures particular requirements can be raised ahead of time.
- Monitoring has been prototyped using both Monasca and Prometheus and the better implementation will require further prototyping.

3.4 Risks Addressed

The main risks being addressed relating to the platform:

- SDPRISK-398: Insufficient detail in the SDP platform-middleware interaction
    - Exposure was:High; Residual Exposure is: None
- SDPRISK-406: Requirements for QA Metrics and their control poorly defined at system level
    - Exposure was: Medium; Residual Exposure is: None
- SDPRISK-335: Incomplete interface description to LMC
    - Exposure was: Medium; Residual Exposure is: None

Main risk relating to the buffer:

- SDPRISK-363: Buffer hardware and software does not meet performance requirements
    - Exposure]was: High; Residual Exposure is: Medium
- SDPRISK-400: Hardware specification do not reflect actual needs
    - Exposure was: High; Residual Exposure is: Medium

On top of those, there are two key aspects that are being considered:

- Interface to Platform Services and the use of OpenStack as described in SDP Operational System C&C View [AD1]
- Use of containerisation, and how it affects performance versus the flexibility of containerised application delivery.

4 Collaboration with CERN

Through connections in the OpenStack community, and in particular the OpenStack Scientific SIG, [RD12], a dialogue at a technical level began between CERN and representatives of the SKA, in particular the SDP consortium, led by the P3-AlaSKA team. This association was formalised by the CERN-SKA executive agreement signed in July 2017 and is motivated in part by the experience of the CERN community in relation to SKA Regional Centres and in part the opportunity to collaborate with CERN Openlab.

As a result, representatives of the CERN OpenStack technical team met with a group working on OpenStack-related activities with SDP prototyping in summer 2017, and a number of key common objectives were shared and prioritised. The areas of interest for CERN and for SDP were outlined, and common areas where knowledge could be shared or mutual assistance provided were identified, namely:

- Containerised workloads on bare metal
- Preemptible (spot) instances
- Scalable bare metal deployments
- Application hot storage
- Ceph and RDMA
- Inter-cloud federation

As a result of this collaboration, progress in several of the identified areas has been made. A number of presentations have been made at OpenStack conferences:

- "Future Science on Future OpenStack": Stig Telfer (StackHPC) and Belmiro Moreira (CERN), OpenStack Sydney Summit, November 2017
- "Containers on Bare Metal and Preemptible Instances at CERN and SKA": John Garbutt (StackHPC) and Belmiro Moreira (CERN), OpenStack Vancouver Summit, May 2018
- "Science Demonstrations: Preemptible Instances at CERN and Bare Metal Containers for HPC at SKA": John Garbutt (StackHPC), Belmiro Moreira (CERN), Theodoros Tsioutsias (CERN), to be presented at OpenStack Berlin Summit, November 2018.

A follow-up meeting was held in July 2018 at CERN. The areas of discussion included:

- Deployment of software RAID on bare metal
- Content management databases for infrastructure hardware inventory management
- Running containerised workloads on bare metal
- Storage orchestration for containerised workloads
- Preemptible (spot) instances
- Nova Cells in operation
- Management of GPUs in an OpenStack environment
- OpenStack and HPC use cases
- Experiences of using Ceph at MeerKAT
- Experiences of using Ceph for the Human Brain Project
- Experiences of using Ceph on the P3-ALaSKA SDP prototype
- Software-defined networking using Tungsten Fabric

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.

[AD1] SKA-TEL-SDP-0000013 Rev 07 SDP Architecture

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.

[RD1] SKA-TEL-SDP-0000166 SDP Memo 069 P3-Alaska OpenStack Prototyping
[RD2] SKA-TEL-SDP-0000167 SDP Memo 070 P3-AlaSKA Container Orchestration and Compute/Storage Provisioning Interface
[RD3] SKA-TEL-SDP-0000165 SDP Memo 068 P3-AlaSKA Monitoring and Logging
[RD4] SKA-TEL-SDP-0000126 SDP Memo 045 Buffer Modelling and Prototyping
[RD5] SKA-TEL-SDP-0000131 SDP Memo 051 Cloud Native Applications on the SDP Architecture
[RD6] SKA-TEL-SDP-0000132 SDP Memo 053 Monitoring and Logging for the SDP Architecture
[RD7] SKA-TEL-SDP-0000163 SDP Memo 052 Apache Kafka for and SDP log-based Architecture
[RD8] https://github.com/stackhpc/ansible-role-os-manila-mount
[RD9] SKA-TEL-SDP-0000137 SK1 SDP Integration Prototype (SIP) Report
[RD10] SKA-TEL-SDP-0000117 Execution Frameworks Prototyping Report
[RD11] http://openstack.org
[RD12] OpenStack Scientific SIG
[RD13] STFC-UK UKT0 Community
[RD14] https://kayobe.readthedocs.io/en/latest/

5.3 Other Material

The following links provide more information on P3-AlaSKA itself and some of the prototyping activities it supports.

- Overview of P3
  - https://confluence.ska-sdp.org/display/WBS/Performance+Prototype+Platform
- P3-Alaska Configuration
  - https://github.com/SKA-ScienceDataProcessor/alaska-kayobe-config
- P3-Appliances (ansible)
  - https://github.com/SKA-ScienceDataProcessor/p3-appliances
- P3-Webcasts
  - https://confluence.ska-sdp.org/display/WSC/2017-05-31+SDP+Performance+Prototype+Platform
  - https://confluence.ska-sdp.org/display/WSC/2018-03-28+Update+on+P3-ALaSKA
- Monitoring and Logging as a Service Presentation

    - https://confluence.ska-sdp.org/download/attachments/259621034/A%20Holistic%20Approach%20to%20Monitoring.pdf?version=1&modificationDate=1522222313000&api=v2
- P3 – One Year On
    - https://confluence.ska-sdp.org/download/attachments/259621034/2018-03-28%20P3-ALaSKA%20update%20-%20Stig%20Telfer.pdf?version=1&modificationDate=1522179895000&api=v2