





## SDP Science Pipeline Management Use Case View

Document number.....SKA-TEL-SDP-0000013  
 Document Type.....REP  
 Revision.....05  
 Author.....M. Ashdown  
 Release Date.....2018-04-25  
 Document Classification.....Unrestricted  
 Status.....Released

Lead Author	Designation	Affiliation
Mark Ashdown	SDP Project Scientist	University of Cambridge
Signature & Date:	 Mark Ashdown (Apr 24, 2018)	

Released by	Designation	Affiliation
Paul Alexander	SDP Project Lead	University of Cambridge
Signature & Date:	 Paul Alexander (Apr 24, 2018)	

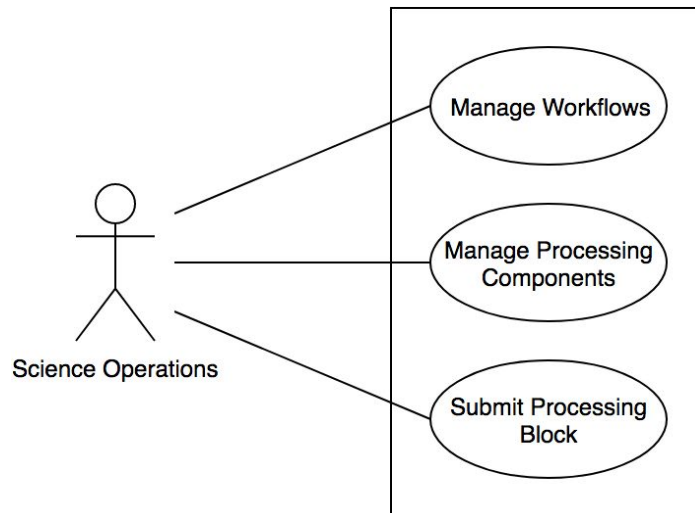
# Table of Contents

<b>1. Primary Representation</b>	<b>3</b>
<b>2. Element Catalogue</b>	<b>3</b>
2.1. Elements and Their Properties	3
2.1.1 Telescope Operator	3
2.1.2 Manage Workflows	3
2.1.3 Manage Processing Components	3
2.1.4 Submit Processing Block	4
2.2. Relations and Their Properties	4
2.3. Element Interfaces	4
2.4. Element Behavior	4
<b>3. Context</b>	<b>5</b>
<b>4. Variability Guide</b>	<b>6</b>
<b>5. Rationale</b>	<b>6</b>
<b>6. Related Views</b>	<b>7</b>
<b>7. References</b>	<b>8</b>
7.1. Applicable Documents	8
7.2. Reference Documents	8
<b>8. Version History</b>	<b>8</b>

## List of Abbreviations

C&C	Component and Connector
SDP	Science Data Processor
SKA	Square Kilometre Array

# 1. Primary Representation



**Figure 1:** UML diagram showing the use cases for managing the SDP science pipelines. Each of the use cases (in ellipses) is linked to the actor carrying them out (the stick person).

## 2. Element Catalogue

### 2.1. Elements and Their Properties

#### 2.1.1 Science Operations

Science Operations users will interact with the SKA system to manage the science pipelines that will be executed by the SDP.

#### 2.1.2 Manage Workflows

Managing workflows consists of actions such as:

- Get an existing workflow;
- Create a new workflow;
- Modify an existing workflow;
- Request the validation of a workflow (this could be automatically triggered by the creation or modification of a workflow);
- Get the status of a workflow (e.g. new, validated, not validated, deprecated);
- Set the status of a workflow (e.g. mark as deprecated).

#### 2.1.3 Manage Processing Components

Managing processing components consists of actions such as:

- Get the source code of an existing processing component;

- Create the source code for a new processing component;
- Modify the source code of an existing processing component;
- Request the validation and deployment of a processing component (this could be automatically triggered by the creation or modification of a processing component);
- Get the status of a processing component (e.g. new, validated, not validated, deployed, deprecated);
- Set the status of a processing component (e.g. mark as deprecated).

#### **2.1.4 Submit Processing Block**

Submitting a processing block consists of actions such as:

- Get the list of validated workflows;
- Get the specification of the parameters of a workflow (e.g. their types and range of acceptable values);
- Set the values of the parameters;
- Check the values of the parameters against the specification;
- Submit the processing block containing the workflow.

### **2.2. Relations and Their Properties**

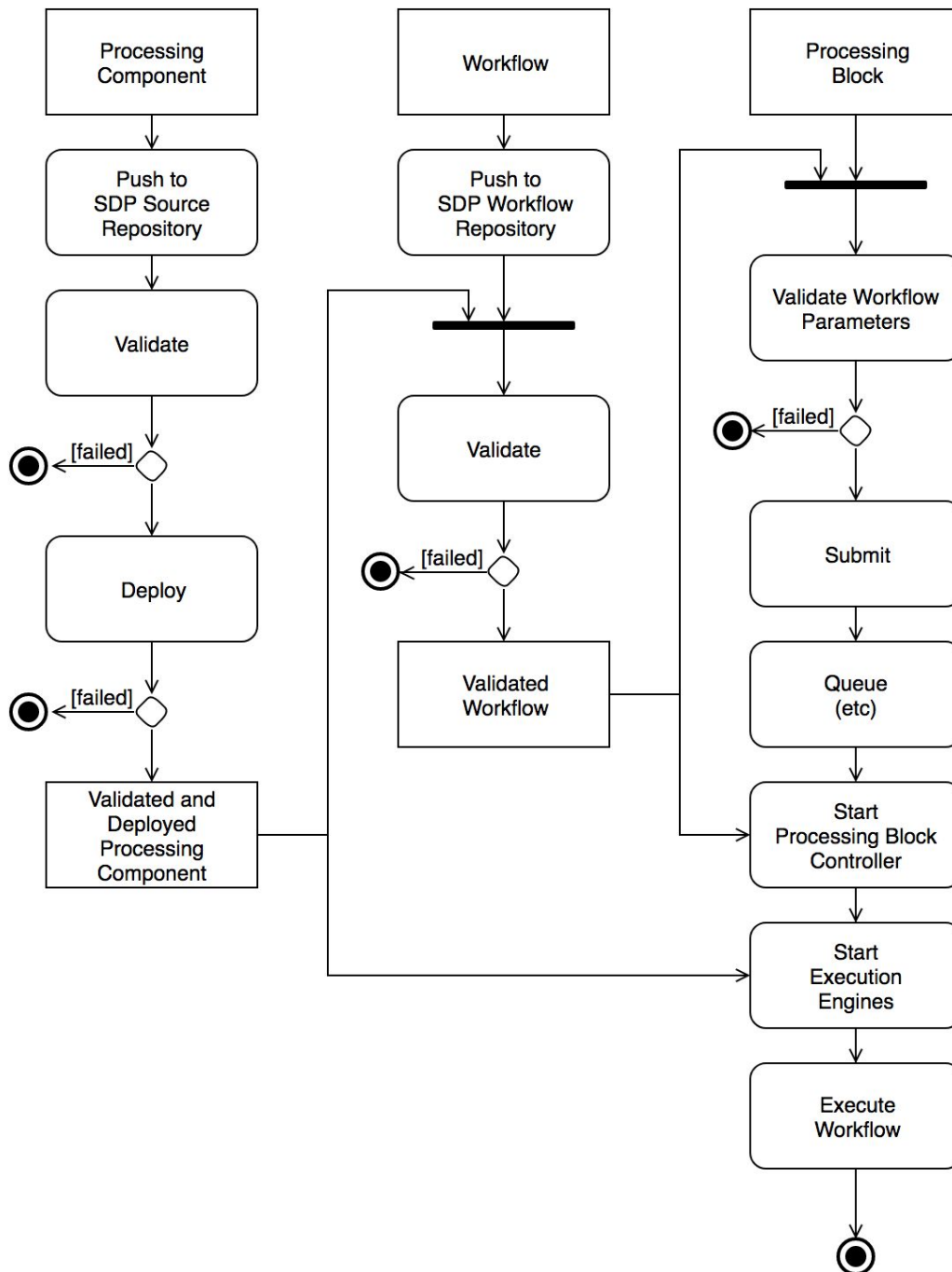
Not applicable.

### **2.3. Element Interfaces**

Not applicable.

### **2.4. Element Behavior**

Figure 2 shows an activity diagram for the most important elements of these use cases: creating or modifying a processing component, creating or modifying a workflow, and submitting a processing block. These three activities are combined in one diagram because the validation of a workflow depends on its processing components being validated and deployed, and the submission of a processing block depends on its workflows being validated. The existence of a workflow repository and a source repository (for processing components) is assumed.



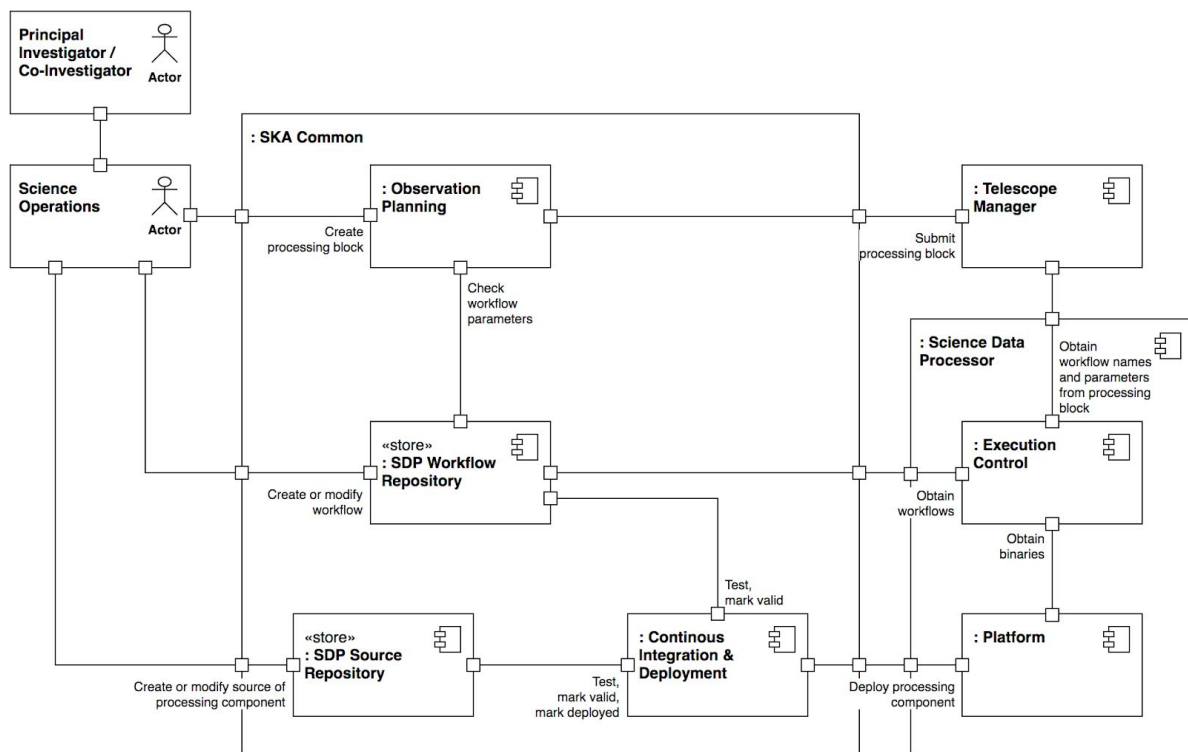
**Figure 2:** Activity diagram for creating or modifying a processing component (left-hand column), creating or modifying a workflow (central column) and submitting a processing block (right-hand column).

### 3. Context

The use cases described in this document focus on the management of the science pipeline workflows that will be executed by the SDP. These use cases have been generated from the point of view of Science Operations users interacting with the system as a whole.

## 4. Variability Guide

There is considerable variability in how the SDP science pipeline management system could be implemented to satisfy the use cases described in this document. Figure 3 shows a C&C view of an example architecture for this system. It makes assumptions about the system-level architecture, since the workflow and source repositories, and the continuous integration and deployment system are placed in SKA Common. The continuous integration and deployment system tests and validates the workflows and processing components, and also passes the processing components to the SDP Platform to be deployed. However it is implemented, the interfaces between the science pipeline management system and the SDP Platform are required.



**Figure 3:** C&C view of an example architecture for the SDP science pipeline management system. The standard notation has been abused to show the Principal Investigator / Co-Investigator and Science Operations actors as components of the system.

## 5. Rationale

This view has been developed as use cases because it has implications for the system-level architecture. The only drivers for the use cases are the Quality Attribute Scenarios.

### 5.1 Modifiability

**Requirement:** SDP\_REQ-809 (New pipeline workflows or algorithms)

The use cases therefore directly address the modifiability of the scientific pipelines by considering the management of the workflows and processing components, including the creation of new ones, or modification of existing ones.

## 6. Related Views

This view is not a decomposition of another view, since there is no overall use case view for SDP. The elements of the system involved in executing the science pipelines, such as Execution Control and the Processing Block Controller, are described in the SDP Operational System C&C View [RD01]. The elements of the system involved in deployment and provisioning of executables is described in the SDP Platform Services C&C View [RD02]. The relationship between workflows, processing components and execution frameworks is described in the SDP System-level Module Decomposition and Dependency View [RD03]. Further details on individual processing components are given in the SDP Processing Component Module View [RD04].

## 7. References

### 7.1. Applicable Documents

There are no applicable documents.

### 7.2. 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.

- [RD01] SKA-TEL-SDP-0000013, SDP Operational System Component and Connector View, Rev 05
- [RD02] SKA-TEL-SDP-0000013, SDP Platform Services Component and Connector View, Rev 05
- [RD03] SKA-TEL-SDP-0000013, SDP System-level Module Decomposition and Dependency View, Rev 05
- [RD04] SKA-TEL-SDP-0000013, SDP Processing Component Module View, Rev 05

## 8. Version History

Version	Date of Issue	Prepared by	Comments
05	2018-04-25	Mark Ashdown	Prepared for M20 Pre-CDR submission