



## A6.D4.1 – Security Manager

E. Garcia

<b>Document Number</b>	A6.D4.1
<b>Document Title</b>	Security Manager
<b>Version</b>	1.0
<b>Status</b>	Final
<b>Work Package</b>	WP 6.4
<b>Deliverable Type</b>	Prototype
<b>Contractual Date of Delivery</b>	30 June 2007
<b>Actual Date of Delivery</b>	30 July 2007
<b>Responsible Unit</b>	ATO
<b>Contributors</b>	
<b>Keyword List</b>	S&D Manager, Run-time Framework
<b>Dissemination level</b>	PU

## Change History

<b>Version</b>	<b>Date</b>	<b>Status</b>	<b>Author (Unit)</b>	<b>Description</b>
0.1	30/7/2007	Draft	Enrique J. Garcia (ATO)	Creation of the document
1.0	31/7/2007	Final	Enrique J. Garcia (ATO)	Minor changes to make the document compliant with the template

## **Executive Summary**

Security Manager (S&D Manager) is a module included in Serenity Runtime Framework. S&D Manager is in charge of select the most suitable pattern in real time besides connect different modules of Serenity Runtime Framework architecture. In this deliverable, main actions and design aspects about S&D Manager will be seen in detail.

## Table of Contents

1. Introduction .....	5
1.1. Overview .....	5
1.1.1. General description .....	5
1.1.2. Functionalities .....	5
1.1.3. Input/Output interfaces .....	5
1.2. S&D Manager within SRF Architecture .....	6
2. S&D Manager components .....	7
2.1. Introduction .....	7
2.2. Application Access .....	7
2.3. Monitor Service Access .....	8
2.4. Context Manager Access.....	9
2.5. S&D Query.....	10
2.6. Console Access.....	10
2.7. Manager.....	11

# 1. Introduction

---

## 1.1. Overview

### 1.1.1. *General description*

The S&D Manager is the heart of the SRF architecture:

- This component evaluates the current system context, the application requirements and the available patterns and identifies the suitable patterns for deployment.
- It is also responsible for the activation and deactivation of implementations and redeeming actions when monitoring rules have been violated.
- The S&D Manager will also translate S&D configurations fed from users (through the console) into system patterns that need to be activated and also responsible for interacting with users (through the console) when human intervention is needed.

### 1.1.2. *Functionalities*

The S&D Manager will implement the logic of S&D patterns by combining application requirement, available S&D patterns and current system context in order to choose the appropriate implementation that needs to be invoked. The S&D Manager will be the component responsible for activating and deactivating pattern implementations and will also be accountable for taking necessary actions (based on the monitoring rules) when informed by the Monitor Service of a violation. The S&D Manager will also be able to run system patterns in order to conform to S&D configurations and will be able to interact with end-users through the console in order to provide information about the framework's status, potential violations or allow the user to make changes to the framework's configuration.

### 1.1.3. *Input/Output interfaces*

- Input will be provided to the S&D Manager from the application with either requirements or exact implementation chosen in order for the S&D Manager to either choose the appropriate implementation or initiate the implementation.
- The S&D Query will provide to the S&D Manager with the patterns that satisfy the request done by the application.
- Input will also come from the Monitor Service and the Event Manager concerning either events of violations of monitoring rules.
- Input may also come from the Context Manager following a request for context information.
- Finally, input may come from the Console, which will be in the form of changes to the S&D configurations or input from the end user.
- Direct output of the S&D Manager is the implementation executable, which the Manager is able to activate and deactivate.
- Output toward the application is in the form of references to the implementation that the application must use. It sends the handler to the application after it creates and starts the Executable Implementation.

- Output is also toward the S&D Query, which the Manager uses to find appropriate classes, patterns and implementations.
- Output towards the console is the direct link of the S&D Manager to the end user, presenting information about the system’s status and requesting input.
- Output towards the Monitor Service is in the form of monitoring rules that should be monitored when a new pattern is applied, or monitoring rules that need to be removed when a pattern is deactivated.
- Finally, output towards the Context Manager is in the form of updates to the context.

## 1.2. S&D Manager within SRF Architecture

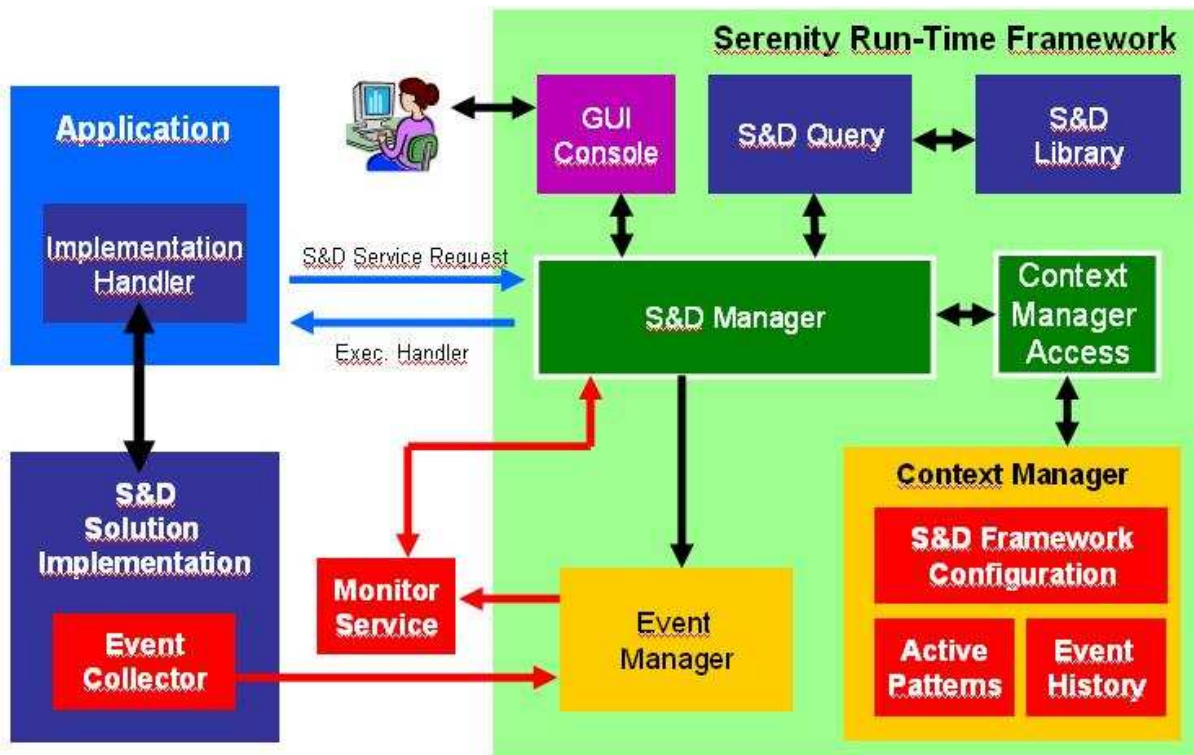
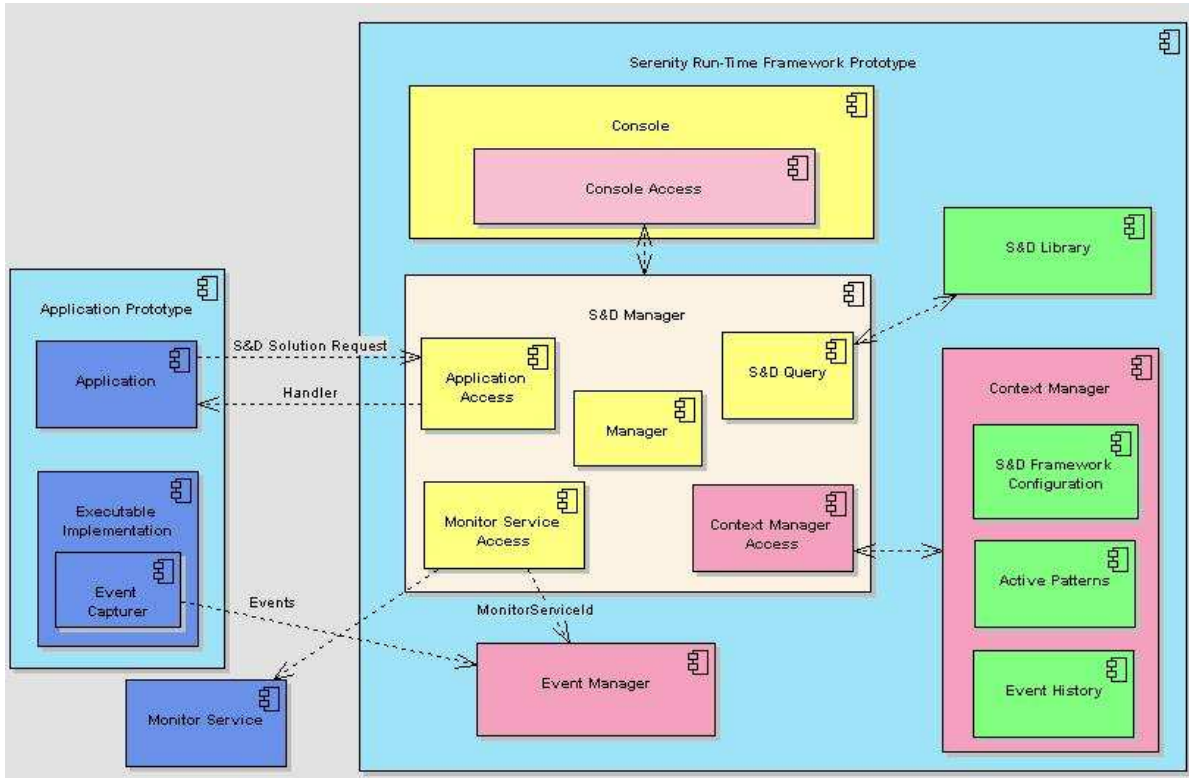


Figure 1 – General SRF Architecture

Figure 1 shows the general Serenity Runtime Framework architecture. S&D Manager is the heart of the SRF Architecture, it receives the most of requests from external elements to the SRF and manage the answer. So, it has several communications with internal and external elements (Console, S&D Query, Context Manager Access, Event Manager, Application, Monitor Service, etc.). In the next section, several parts of S&D Manager will be seen in detail.

## 2. S&D Manager components

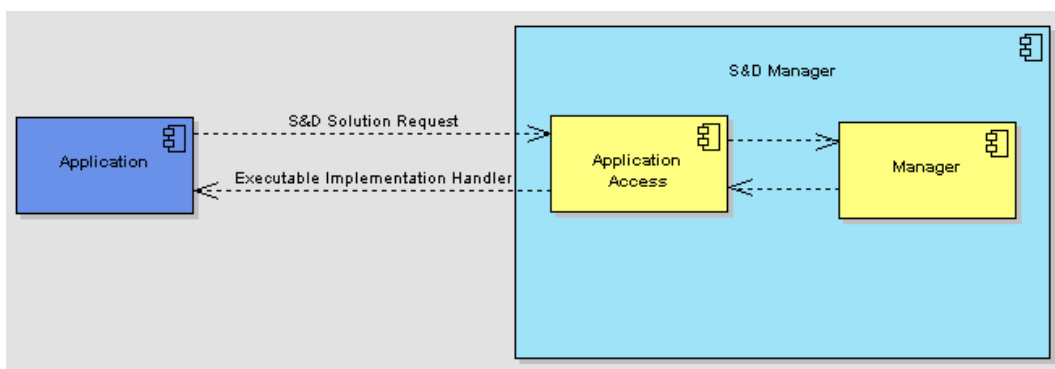
### 2.1. Introduction



**Figure 2 – Serenity Run-Time Framework Prototype Architecture**

Figure 2 shows the general SRF Prototype architecture. S&D Manager is the heart of the architecture with connections with several components (Console, S&D Library, Context Manager, Event Manager, Application and Monitor Service).

### 2.2. Application Access



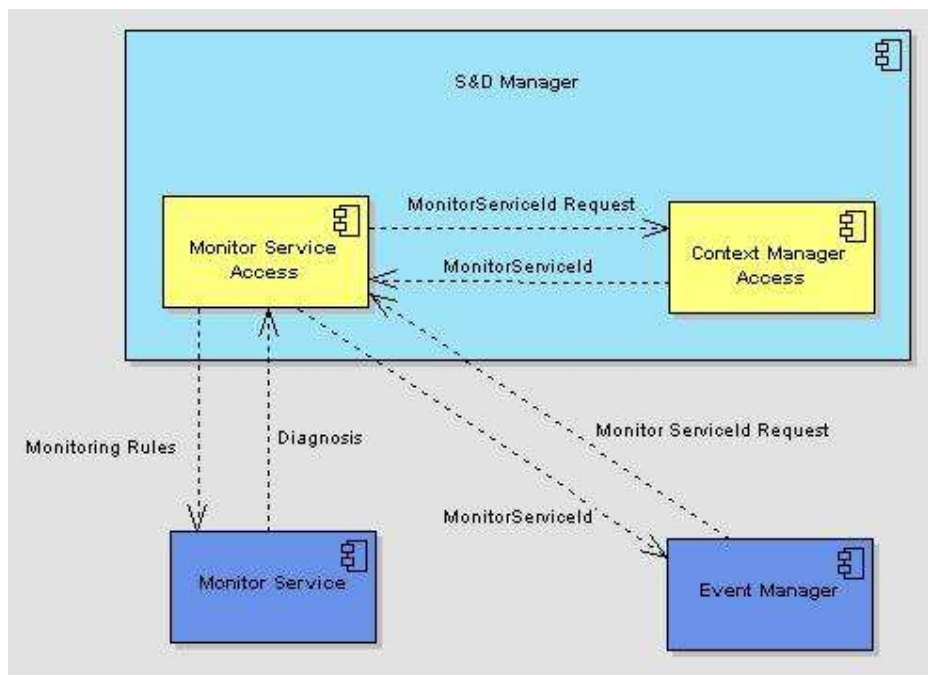
**Figure 3 – Application Access**

Figure 3 shows the communications of *Application Access Module*. This module acts as an interface between *Application* and *S&D Manager*.

The main actions of *Application Access Module* are:

- It receives a S&D Solution Request. S&D Solution Request include the following information:
  - S&D Class – specify the class that *Application* needs
  - S&D Properties – specify some properties that required pattern should have
  - S&D Preconditions – specify some preconditions to fulfil by the required pattern.
- It sends the S&D Solution Request to the *Manager*
- It receives the answer of the *Manager*
- It sends the Executable Implementation Handler if succeeded. Sends an error message in other case.

### 2.3. Monitor Service Access



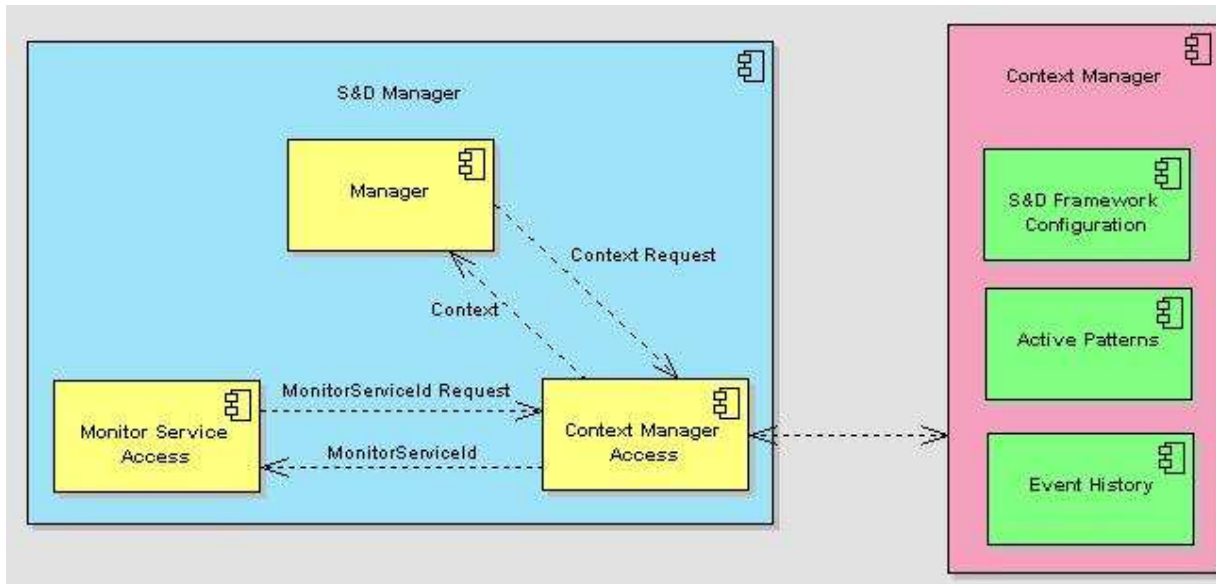
**Figure 4 – Monitor Service Access**

Figure 4 shows *Monitor Service Access Module* and some elements related. *Monitor Service Access* is communicated with *Monitor Service*, *Event Manager* and *Context Manager Access*. The workflow and main actions of this module are:

- *Monitor Service Access* sends the Monitoring Rules associated to a pattern to the *Monitor Service* when this pattern will be selected and activated.

- It receives diagnosis information from *Monitor Service* when some events have been received and some rules have been violated.
- *Event Manager* need to know the information of the *Monitor Service* that have to send the events that it is receiving. *Monitor Service Access* is in charge of provide this information through *Context Manager Access*

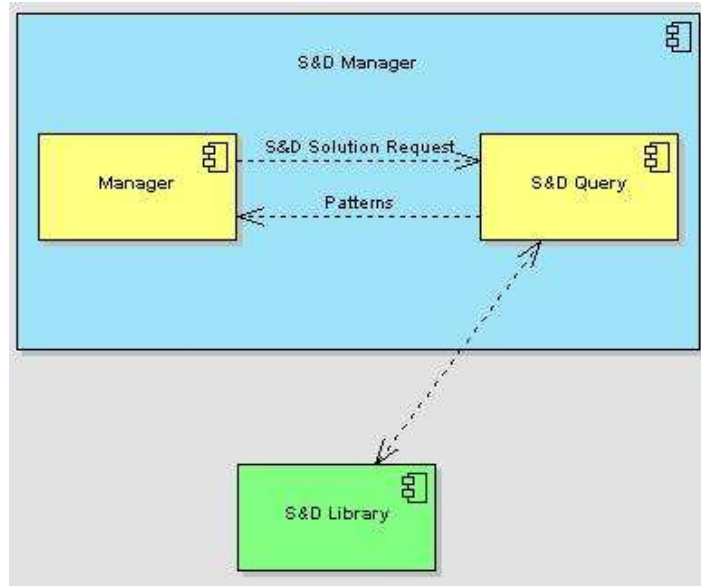
## 2.4. Context Manager Access



**Figure 5 – Context Manager Access**

Figure 5 shows an architecture of Context Manager Access. Context Manager Access is the module in charge of access to the Context Manager. All modules that need some information about the context should request this information to the Context Manager Access. So, this module receives request from Monitor Service Access (see section 2.4 in this deliverable) and from Manager.

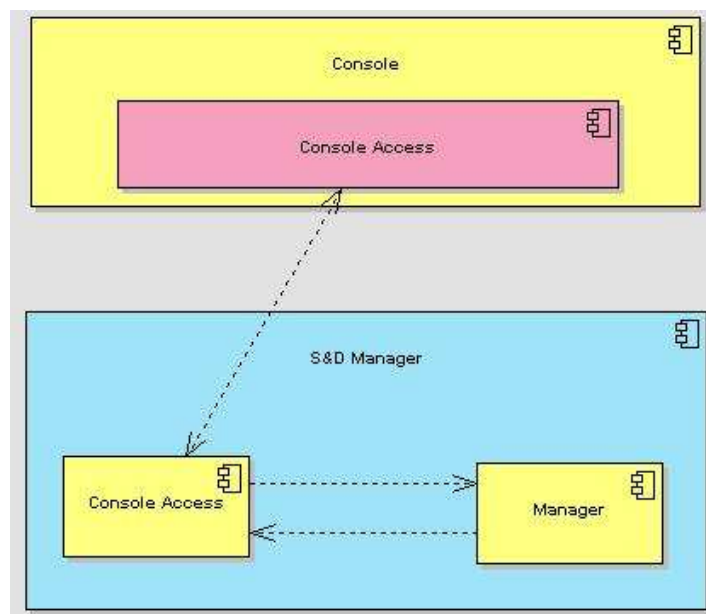
## 2.5. S&D Query



**Figure 6 – S&D Query**

The S&D Query is responsible for contacting the S&D Library and retrieving either classes, patterns or implementations and forwarding them to the Manager component. Figure 6 shows the architecture of the S&D Query and the communications with Manager and S&D Library. There is more information about S&D Query in section 3 of the deliverable ‘A6.D4.3 Security Library’

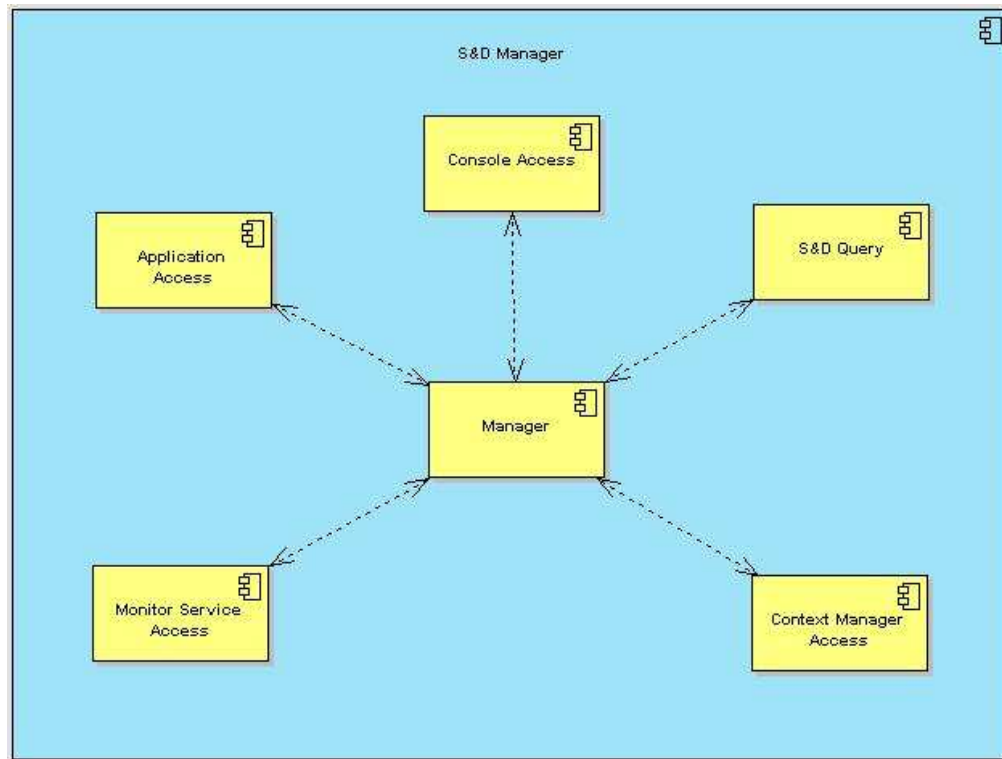
## 2.6. Console Access



**Figure 7 – Console Access**

There is a module with SRF Prototype but independent of S&D Manager, Console. Console is the module in charge of provide a graphical user interface to the S&D Authority. Console Access is the module in charge of provide the interface between the Manager and Console. Figure 7 shows the communications of Console Access.

## 2.7. Manager



**Figure 8 – Manager**

Manager is the heart of S&D Manager. It receives inputs of all the modules of S&D Manager and provide to them the information that they need. Figure 8 shows that Manager has connectivity with the other modules within S&D Manager.

- Console Access will send information to the Manager about which S&D Solutions (S&D Classes, S&D Patterns and/or S&D Implementations) want to show in the GUI. Manager will request this S&D Solutions to the S&D Library (through S&D Query)
- Console Access will also send information to the Manager about some changes in the S&D Configuration that the S&D Authority has introduced in the GUI (Console). Manager receives this information and is in charge of update the Context Manager (through Context Manager Access)
- Application Access will send a S&D Solution Request (see section 2.2 in this deliverable). Manager will send this information to the S&D Query. S&D Query will search in the S&D Library the patterns that fulfil the S&D Solution Request. If S&D Query returns more than



one pattern, Manager must select the most suitable pattern using the information stored in the Context Manager (using Context Manager Access to get this information)

## References

- [1] A6.D3.1 – Specification of serenity architecture, A6 SERENITY Support Environment
- [2] A6.D4.2 – Security Library, A6 SERENITY Support Environment