ARC 101 - Architecture Overview

Author: Khalid Subratee
Owner: Stewart McIntyre

Document History

Document Location

This is a snapshot of an on-line document. Paper copies are valid only on the day they are printed. Refer to the author if you are in any doubt about the currency of this document.

Revision History

Date of this revision: 08-31-2010	Date of next revision *(date)*

Revision Number	Revision Date	Summary of Changes	Changes marked
1.0	08-31-2010	Initial - incomplete	N

Contents

1. Introduction

This chapter identifies the document and the (sub)system to which it relates, describes the contents of the document, and states its purpose.

1.1 Identification

This document describes the architecture overview for Science Studio and how it relates to the changes resulting from integration.

1.2 Description

The architecture overview represents the governing ideas and candidate building blocks of an IT system and enterprise architecture. It provides an overview of the main conceptual elements and relationships in an architecture, including candidate subsystems, components, nodes, connections, data stores, users and external systems.

This document will provide a high level understanding of the integration architecture on how the many Science Studio installs (or satellites installs) will be integrated with the core Science Studio at UWO.

1.3 Purpose

The main purpose of the architecture overview is to communicate a simple, brief, clear and understandable overview of the target IT systems and how they generally interact.

This architecture overview will:

- Communicate a conceptual understanding of the target IT system
- Provide a high-level shared vision of the architecture and scope of the proposed IT system
- Enable early recognition and validation of the implications of the architectural approach
- Facilitate effective communication between different communities of stakeholders and developers
- Facilitate orientation for new people who join the project

2. Architecture Overview

This chapter illustrates the architecture of the system and describes the key concepts of the solution.

2.1 Enterprise Level

This section describes the architecture at the enterprise level.

Science Studio (SS) has and will continue to evolve where there are expected to be many SS satellite installs at various facilities. University of Western Ontario (UWO) will be the core install and host the development, test and production environment. The initial satellite install will be located at Canadian Light Source (CLS), however this architecture provides the basis on how the integration will occur and is a model for additional facilities to integrate with SS Core, should they wish to host their own SS install. Alternatively, the additional facilities may simply utilize SS Core to link and control the devices at their facility.

The major component that provides the glue to integrate applications, services or data with SS core at UWO is a JMS server. The JMS server utilizes a Message Broker and message technology as the backbone for SS in its current state as well as for future growth potential of SS.

2.1.1 Integration Approaches

There are 2 main integration approached that have emerged from SS as it continues to expand its install base facilities. The first will see some facilities use SS as its base and extend it for the facility's specific requirements. This would allow complete flexibility for the facility while still maintaining integration with the SS core at UWO, albeit such a facility would be required to maintain hosting environments and development work products. The second approach allows for a facility to fully utilize SS at UWO as its main gate for remote access to its devices via web services, thereby relieving such a facility from hosting and supporting a SS satellite install.

Advanced data computing services will continue to reside at UWO. Having a simplified approach to permit facilities that would only like to send data for processing and receive its results may be desirable and allows SS / ANISE to offer computing services to a larger population. Utilizing JMS rather than developing custom transportation code for each facility is attractive and provides a simplified, consistent transportation method.

2.1.1.1 Hub and Spoke Integration

For the satellite SS installs, the main aspect of this integration utilizes a hub and spoke model, where UWO will be the hub and any satellite SS installs will be connected to the hub via Integration spokes. Figure 1 illustrates this model.

This model allows for each satellite to extend the base SS core to their specific needs, yet still able to communicate with SS Core or other SS satellite installs via the hub. Integration back with the hub (SS core) would be user driven, not a always on synchronization between hub and spoke (satellite). At its highest level in SS, the Project may utilize more that one facility. The owner of the Project would identify it for integration back into the SS core, where this specific Project can utilize the service offered by the SS core. The Project owner determines when to perform the integration back to its local SS satellite. Facilities are not restricted from extending a Project object, however SS core will have defined the canonical model of a Project and any extension of it would not be available in the SS core during the integration processes.

This type of integration relieves the various SS installs from point to point integration, where it will quickly become unmanageable.

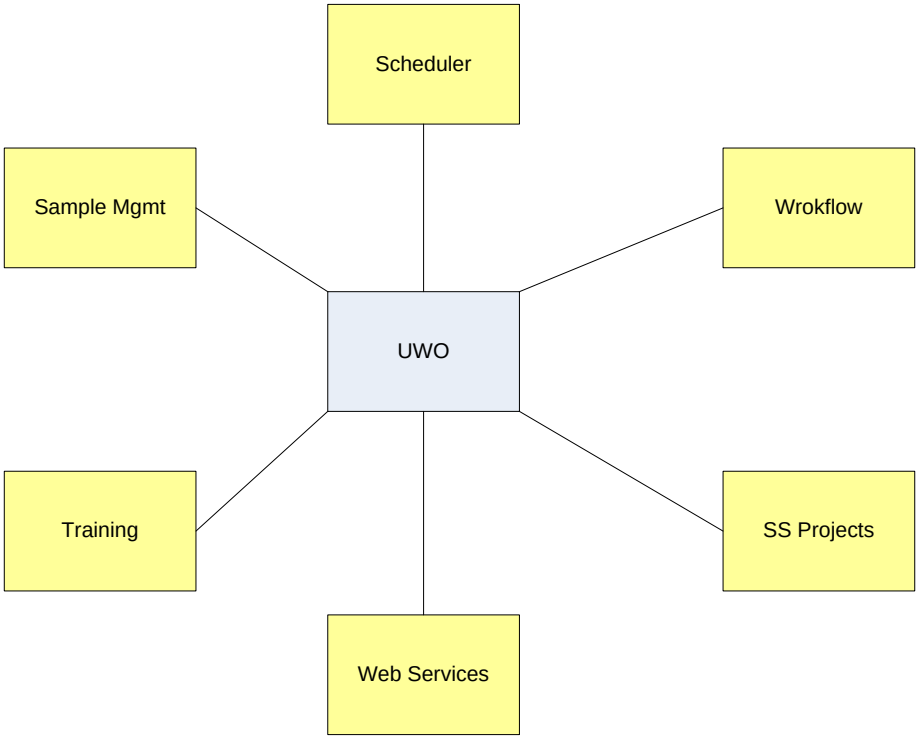

Figure 1 – Hub and Spoke Model

2.1.1.1.1 Key Concepts

- Sites host their own SS
- Integration to and from UWO SS is user driven
- Project level object integration. Projects are versioned and timestamped
- Integration performed via XML and JMS, conforming to a canonical model at UWO

2.1.1.2 Web Services

The web services integration approach permits operation of remote devices with SS, omitting the requirement for a facility to host and support a web application. The facility is required to produce a set of APIs that control the devices / instruments, however the interaction of these devices are from within SS core. These Web Services are used to communicate and control the remote devices. Figure 2 shows the relationship of Web Services.

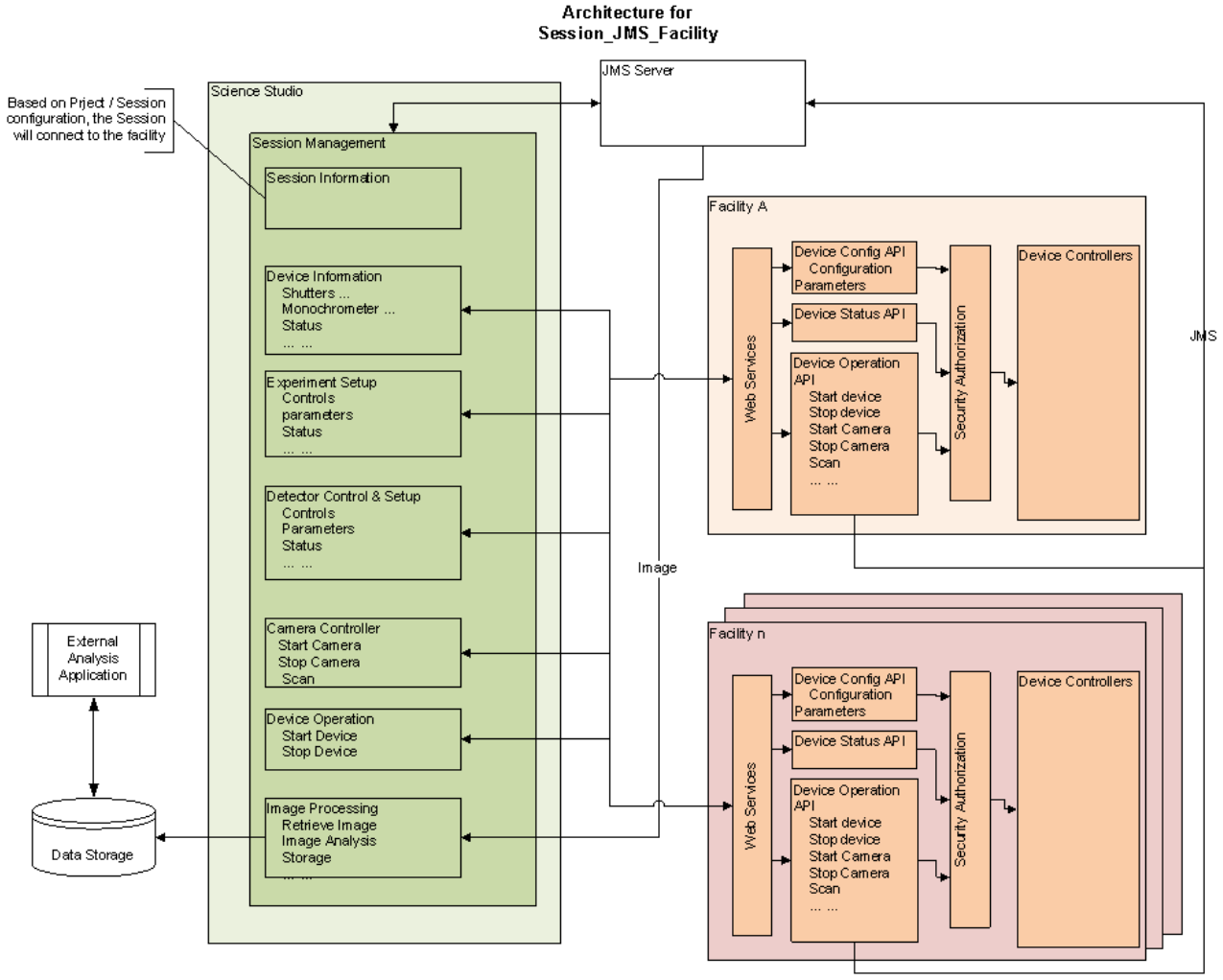

Figure 2 – Web Services Integration

2.1.1.2.1 Key Concepts

- The JMS Server is again central for web services (used for hub and spoke integration and data movement for advanced computing services)
- External facility does not need to host and support a web application
- Web Services are reusable

2.1.1.3 Integration for Advanced Computing Services at UWO

Established facilities that do not require the usage of SS for its remote operation may be attracted to the advanced computing services that SS / ANISE offers. Such facilities are mainly interested in sending raw data for processing and receiving the results. This service can be established independently of a remote facility desire to use SS core.

Utilizing a JMS facility, such as MQ Series, permits a simple yet assured method for data transportation to and from UWO. This approach removes the complexity for a facility to develop custom transportation code to efficiently move vast quantities of data for processing. MQ series will take care of the underlying

transportation of the data, allowing the facilities to concentrate on acquiring the data (scans) and the presentation of its results.

Streams are utilized in the advanced computing service at UWO. MQ adapters for the Streams are in the IBM labs and will require work to see them through to being generally available for this project, however using this type of an adapter further extends the Streams offering and becomes immediately utilized for SS / ANISE. Figure 3 illustrates this concept.

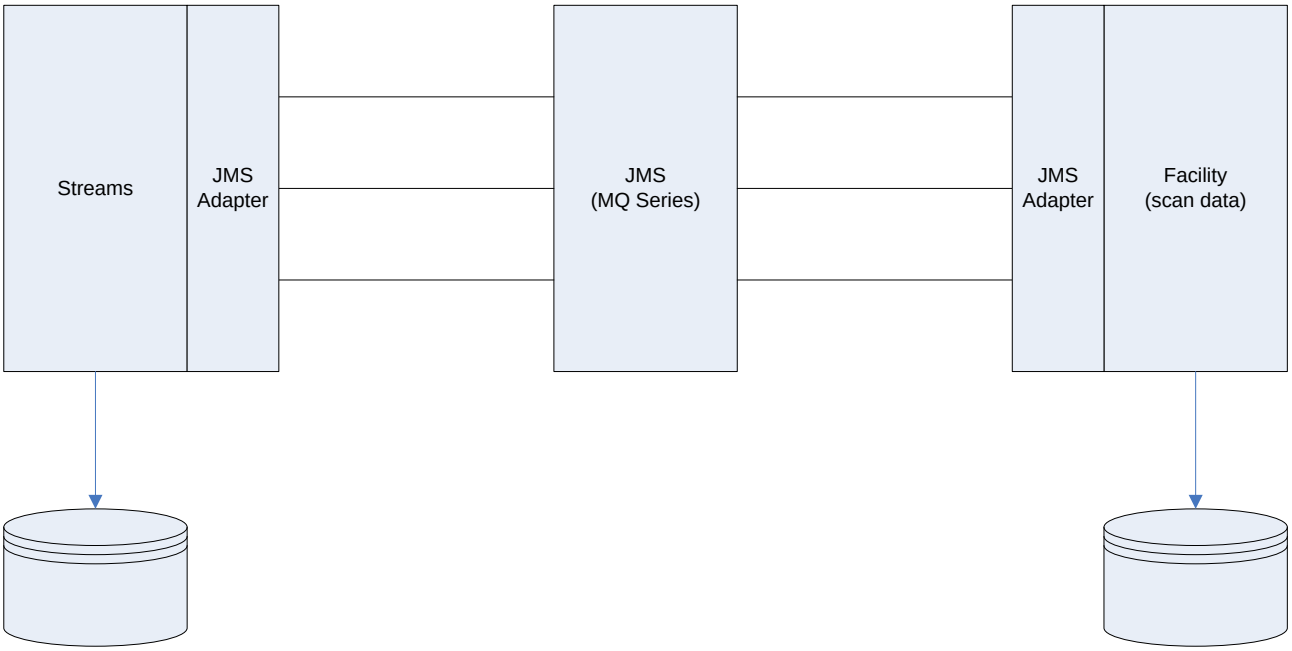

Figure 3 – Data Processing with Streams via JMS

A facility may have multiple types of data being sent for computing at UWO and needs to have the ability to keep this aligned and identified. The Streams computing will not care about the order in which that data is processed, but the order or location of the scan of each piece of data is important to the presentation layer (weather at the source site or within SS). Messaging maintains order of each scan as well as isolating differing data from the same facility. Figure 4 illustrates this relationship with Streams.

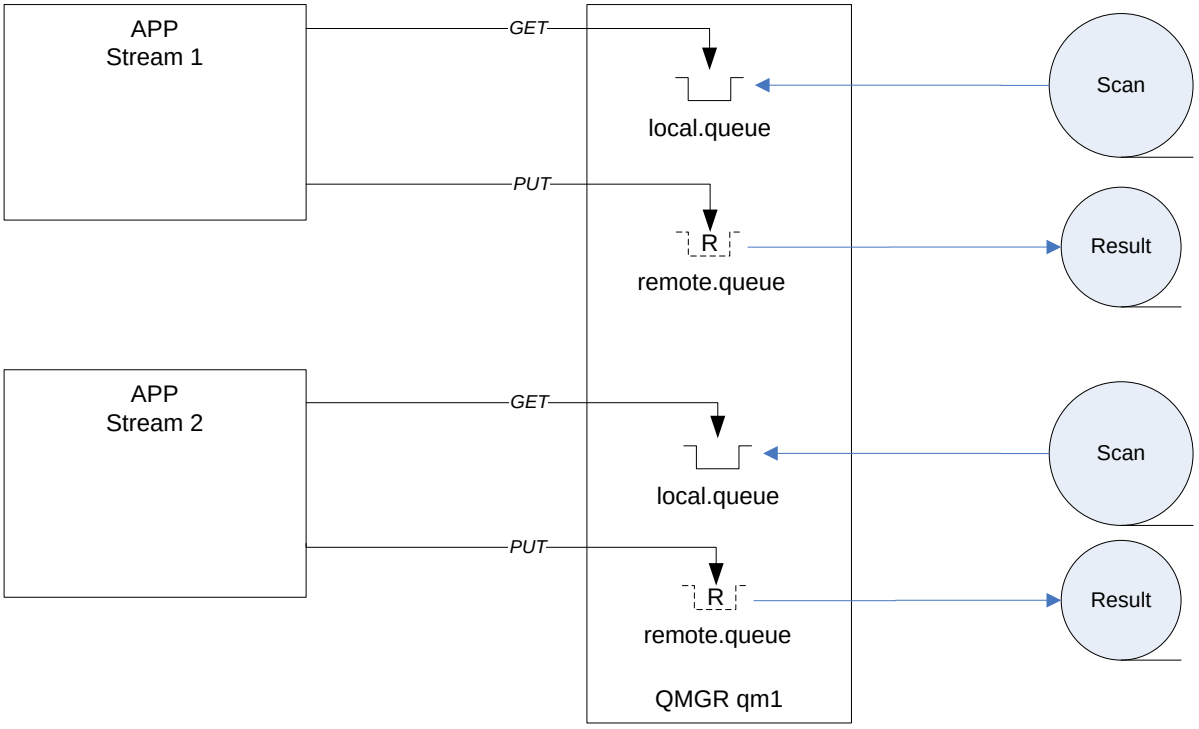

Figure 4 – Independent Message Processing

2.1.1.3.1 Key Concepts

- Isolated data / results set
- Multiple data inputs from a facility
- No custom transportation code
- Product configuration on JMS provider, easily enables new or additional facilities
- Assured delivery of data

2.2 IT System Level

This paragraph describes the architecture at the IT system level.

The foundational framework to be applied uses the Message Broker as the core integration hub. Interaction with external applications or services would be controlled and routed by the Message Broker. As services expand and SS matures, the Message Broker becomes the fundamental backbone for SS integration. Figure 5 illustrates the conceptual view of the Message Broker

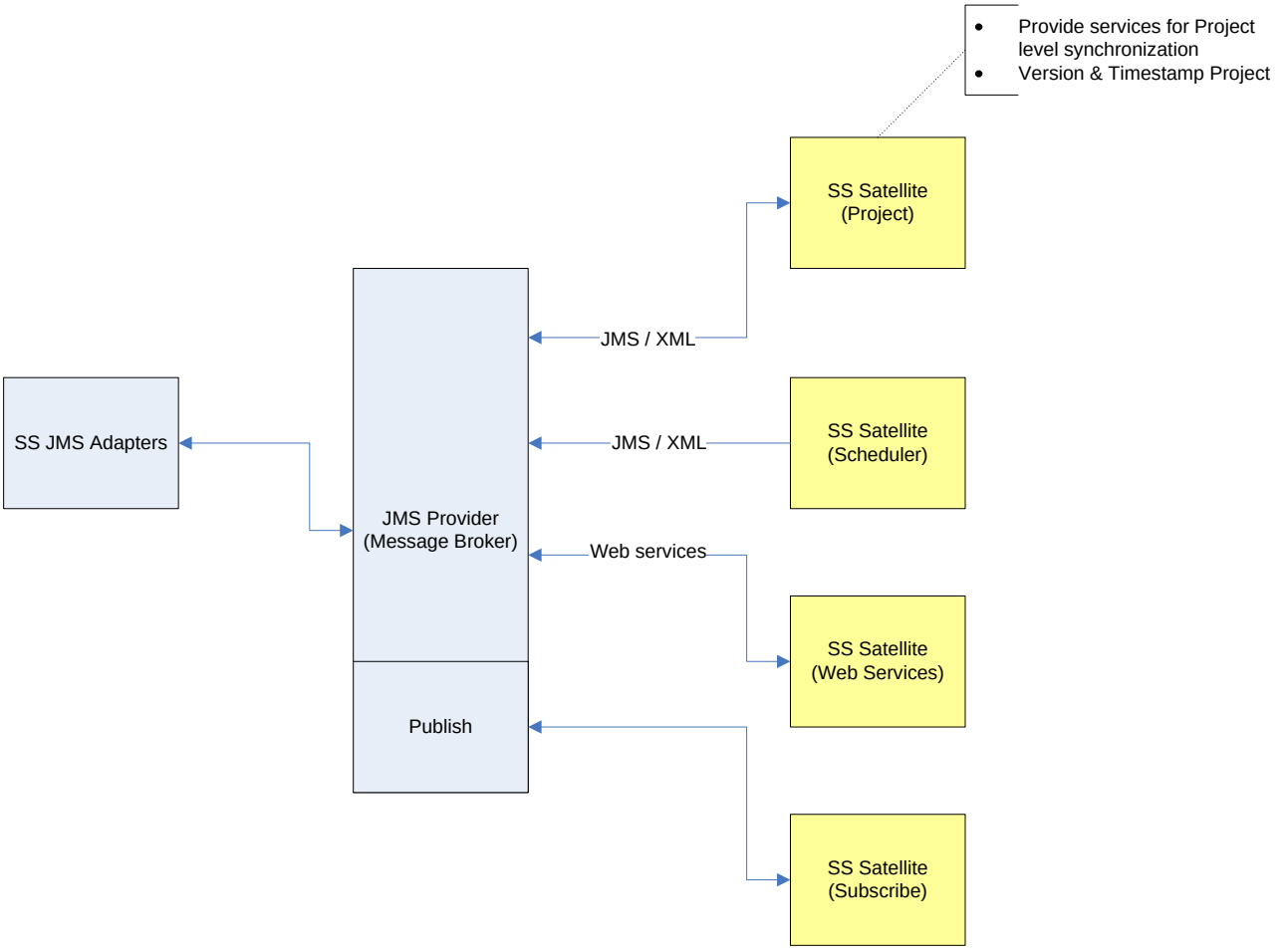

Figure 5 – Message Broker

The Message Broker will utilize a JMS messaging services (i.e. MQ Series). This messaging service is also utilized for data transportation for advanced computing services at UWO. Message Broker provides a loosely coupled application integration model, permitting efficient enablement to bring on new SS satellites or integrate new services with SS core.