

Mercury2 Revised Design

Document 0.3.0

Authors: Jimmy Blanchard
Date: December 11, 2012



Rev	Date	Notes	Authors
0.3.0	December 2, 2012	Mercury2 design revisions.	jimblanc

Contents

1	Introduction	3
1.1	Terms and Definitions	3
1.2	Application Requirements	4
1.2.1	User Interface Requirements	4
1.2.2	Hardware Manager Requirements	4
2	Architecture Details	6
2.1	User Interface	6
2.1.1	User Manager	7
2.1.2	Reservation System	7
2.1.3	Satellite Tracking	7
2.1.4	Ground Station Manager	7
2.1.5	Administration Panel	7
2.2	Hardware Manager	7
2.2.1	Network Manager	7
2.2.2	Schedule Manager	7
2.2.3	Hardware Drivers	7
2.2.4	Telemetry Buffer	7
2.3	Data Flow	7
2.3.1	Security	7

1 Introduction

Mercury2 is the next evolution of the Mercury Ground Station System. It will allow satellite operators to reserve, configure, and use ground station hardware while still giving ground station operators complete control over their hardware. Mercury2 will include a feature rich web interface to enable ground station control over the network by satellite and ground station operators alike.

1.1 Terms and Definitions

This section contains a list terms and acronyms commonly used in this document.

Mercury2	Refers to the ground station management system as a whole, including all sub-components (e.g. the hardware manager and user interface).
Hardware Manager	An application that runs on a computer that is physically attached to the ground station hardware. It is responsible for processing ground station commands (from the user interface) and facilitating data transfer.
User Interface	This application runs on a web server and provides the interfaces to allow ground station and satellite operators to interact with and control the ground station. It relays commands from the user to the hardware manager.
Hardware Pipeline	A collection of related hardware used to either transmit or receive information to and from the radio (or both, if the hardware supports it).
Ground Station	Refers to the complete Mercury2 system (i.e. the user interface and any hardware managers associated with it). Generally, this means several computers and pieces of hardware on the same local area network.
Satellite	A device in orbit that Mercury2 configured ground stations can connect to.
Pass	A transit of a satellite over a ground station. Passes can be scheduled, which reserves a specific hardware pipeline for the duration of the pass. Scheduled passes are identified by the satellite name, orbit number, and ground station.
Timestamp	The duration of scheduled command sessions for Mercury2 will be defined by a start and end timestamp. These timestamps will be simple UNIX timestamps indicating the start and end of the reservation.
Satellite Operator	An entity that remotely reserves ground station use (via the user interface) and uses it to send and receive data from a satellite.
Ground Station Operator	A user that is associated with the ground station and has some privileges over it (e.g. the ability to approve/reject reservation requests or the ability to configure hardware pipelines).
Ground Station Administrator	The user with complete administrative control over the ground station.
MVC Framework	Model-View-Controller framework. Refers to a common web application software design pattern.
YAML	An easy-to-use configuration format. Will be used to configure the hardware manager.
Asynchronous	A type of program design that allows tasks to be executed in an undefined order. This will be used in the hardware manager to allow it to respond to radio events.

Table 1: General Definitions

1.2 Application Requirements

This section outlines the various requirements for the Mercury2 system. These are the requirements for the initial release of the application.

1.2.1 User Interface Requirements

The user interface component of Mercury2 will run on a net-accessible web server and will allow satellite and ground station operators to interact with the various configured hardware managers. Its primary features consist of the following items.

- User authentication, authorization, and management
 - Registration and account management
 - API access key management
 - User permissions
- Ground station administration
 - Hardware pipeline configuration
 - Enable or disable ground station access
 - View and modify pending ground station schedules
 - Approve or deny reservation requests (if approval required)
 - Manual override
- Ground station reservation
 - Reservation utility to allow satellite operators to reserve ground station pipeline use
 - Current reservation schedule viewer
 - Upcoming passes over the ground station
 - Automatic TLE updates
- Satellite tracking during pass
 - retroTrack-esque tracker
 - Various data streams from the hardware manager (connection permitting) such as a waterfall plot or web cam feed
 - Telemetry stream connection settings (i.e. IP address, port, etc.)
- SSL encryption and protection from various exploits
- Complete access and error logs

1.2.2 Hardware Manager Requirements

The hardware manager component of Mercury2 will run on a computer physically connected to the ground station hardware. It is responsible for parsing ground station commands from the user interface as well as providing the sockets that satellite operators will use to transmit and receive data to and from their satellite.

- Run schedules and commands received from the user interface

- Asynchronously manage hardware and data streams
- Connect to hardware via drivers
- Buffer and record telemetry data
- Periodically sync schedules from the user interface
- Provide sockets (defined by the schedule) to allow satellite operators to transmit commands to and receive telemetry from the ground station
- Key encrypted security for all data streams (using keys from the user interface)

2 Architecture Details

This section will detail each component of the Mercury2 system displayed in figure 1 (below).

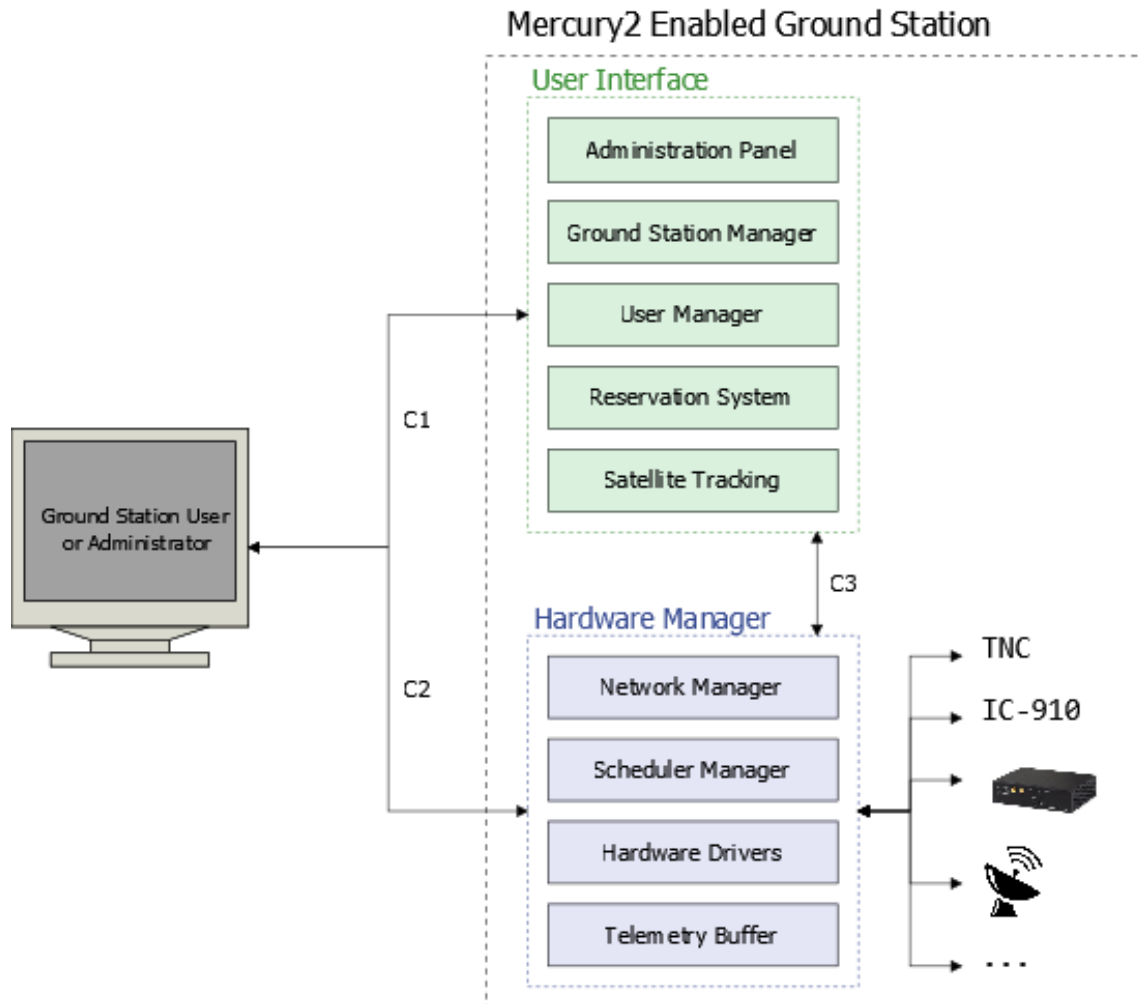


Figure 1: Mercury2 Architecture Overview

2.1 User Interface

The user interface for Mercury2 will consist of an application running on a web server that will allow satellite and ground station operators to interact with the ground station. This web application will be responsible for managing user permissions, maintaining the ground station reservation schedule, providing ground station feedback to satellite operators during reservations, and sending commands to the hardware manager. Each major component of the user interface (illustrated in figure 1) will be described in further detail in the following sections.

Platform Details The user interface application will be developed on top of a popular Python MVC web framework, known as *Django*, running on an Apache web server. Django comes with many useful features

right out of the box such as user management, user input sterilization, and a well developed templating system which will greatly reduce development time. The application will make use of a MySQL database to store user information, schedule details, and user activity logs, among other things.

Security Because the user interface has access to sensitive user information and direct control over the ground station, security and access control will be very important. Fortunately, Django comes with many useful security features by default such as user input sterilization, protection against cross-site scripting attacks, password hashing, and user permission management. User permissions will be configured to give users various levels of control over the ground station depending on their role (satellite operator, ground station operator, administrator, etc.). This mechanism will be detailed in the User Manager section (2.1.1). In addition, the security protocols used to protect the hardware manager data and command streams will be detailed in section 2.3.1.

2.1.1 User Manager

2.1.2 Reservation System

2.1.3 Satellite Tracking

2.1.4 Ground Station Manager

2.1.5 Administration Panel

2.2 Hardware Manager

2.2.1 Network Manager

2.2.2 Schedule Manager

2.2.3 Hardware Drivers

2.2.4 Telemetry Buffer

2.3 Data Flow

2.3.1 Security