Rule-based Analytic Asset Management for Space Exploration Systems (RAMSES)

Phase II STTR Final Report

Prepared for
National Aeronautics and Space Administration
Stennis Space Center (NASA/SSC)

Contract Number: NNS07AB25C

Aurora Report Number: AR09-107

May 21, 2009

Principal Investigator	Program Manager	Corporate Official
Prof. Olivier de Weck	James Francis	Dianna Eichfield
MIT	Aurora Flight Sciences	Aurora Flight Sciences
33-410, 77 Mass Ave	One Broadway, 12th Floor	9950 Wakeman Drive
Cambridge, MA 02139	Cambridge, MA 02141	Manassas, VA 20110
Phone: (617) 715-5195	Phone: (617) 500-0276	Phone: (703) 396-6329
	Fax: (617) 550-4810	Fax: (703) 369-3633
deweck@mit.edu	jfrancis@aurora.aero	deichfeld@aurora.aero

Prepared by
James Francis
Aurora Flight Sciences Corporation
9950 Wakeman Drive
Manassas, Virginia 20110

www.aurora.aero

Project Summary

Payload Systems, a wholly owned subsidiary of Aurora Flight Sciences, and the Massachusetts Institute of Technology (MIT) has developed a Rule-based Analytic Asset Management for Space Exploration System (RAMSES), a system that utilizes a modular layered architecture to enable automated multi-level asset tracking and management for both space and ground applications based on state-of-the-art RFID technology. The main advantages of this system over current bar-code based asset tracking are: (i) significant time savings through automation, (ii) real-time remote status monitoring through the internet, and (iii) rule-based analytics for proactive asset management.

During Phase 1 we demonstrated the feasibility of the RAMSES architecture and designed, implemented, and tested the two core pieces of the architecture, namely the *Smart Container* and the *RDF-based Asset Information and Location Software (RAILS)*. Our formal tests have shown that a read accuracy of >95% is achievable for the smart container while reducing inventory time by 50% or more. We also developed an end-to-end Earth-localized demonstration of the system with browser web access, inventory querying, item geo-location via a Google-maps based Geographic Information System (GIS) as well rule-based analytics with automated remote user notification.

In Phase 2 we brought the RAMSES system to a level of maturity suitable for commercialization. First, we redesigned the *Smart Container* to take advantage of Generation 2 RFID technology, lightweight RFID readers, and batteries, as well as integrating container technologies in a more tight and robust manner. We built a total of three (3) smart containers and demonstrated the scalability of the system and interaction among containers. The RAILS software was upgraded to provide enterprise-level capabilities such as user authentication, inventory and item time history analysis, and an expanded analytic rule set. We performed testing both in the laboratory and in the field and demonstrated seamless three-level tracking (container level, room level, outdoors). Finally, we performed a detailed RAMSES cost/benefit analysis and market survey in preparation for Phase 3.

Table of Contents

1 Identification and Significance of the Innovation

1.1 Summary

Payload Systems, a wholly owned subsidiary of Aurora Flight Sciences, and the Massachusetts Institute of Technology (MIT) has developed a Rule-based Analytic Asset Management for Space Exploration System (RAMSES), a system that utilizes a modular layered architecture to enable automated multi-level asset tracking and management for both space and ground applications based on state-of-the-art RFID technology. The main advantages of this system over current bar-code based asset tracking are: (i) significant time savings through automation, (ii) real-time remote status monitoring through the internet, and (iii) rule-based analytics for proactive asset management.

During Phase 1 we demonstrated the feasibility of the RAMSES architecture and designed, implemented, and tested the two core pieces of the architecture, namely the *Smart Container* and the *RDF-based Asset Information and Location Software (RAILS)*. Our formal tests have shown that a read accuracy of >95% is achievable for the smart container while reducing inventory time by 50% or more. We also developed an end-to-end Earth-localized demonstration of the system with browser web access, inventory querying, item geo-location via a Google-maps based Geographic Information System (GIS) as well rule-based analytics with automated remote user notification.

In Phase 2 we brought the RAMSES system to a level of maturity suitable for commercialization. First, we redesigned the *Smart Container* to take advantage of Generation 2 RFID technology, lightweight RFID readers, and batteries, as well as integrating container technologies in a more tight and robust manner. We built a total of three (3) smart containers and demonstrated the scalability of the system and interaction among containers. The RAILS software was upgraded to provide enterprise-level capabilities such as user authentication, inventory and item time history analysis, and an expanded analytic rule set. We performed testing both in the laboratory and in the field and demonstrated seamless three-level tracking (container level, room level, outdoors). Finally, we performed a detailed RAMSES cost/benefit analysis and market survey in preparation for Phase 3.

1.2 Overview of RAMSES

RAMSES is a system that applies state-of-the-art commercial RFID technology, rule-based inventory management and networked sensor technologies to tracking NASA's materials and supplies from ground infrastructure, to their use on-orbit, and possible return to the ground.

1.2.1 Existing ISS Inventory Management Approach

The current asset management approach for the International Space Station (ISS) program is a barcode-based system, coupled with a relational database. This system is referred to as the Inventory Management System (IMS) and is operated jointly by NASA and the Russian Space Agency (RSA). The system is relatively accurate (only about 3% of the 20,000 items in IMS are tagged as “lost” = 97% “read rate”), but is it very labor intensive and the quality of the data depends entirely on the diligence of the astronauts and inventory stowage officers (ISO) on the ground. The system is shown in Figure 1.

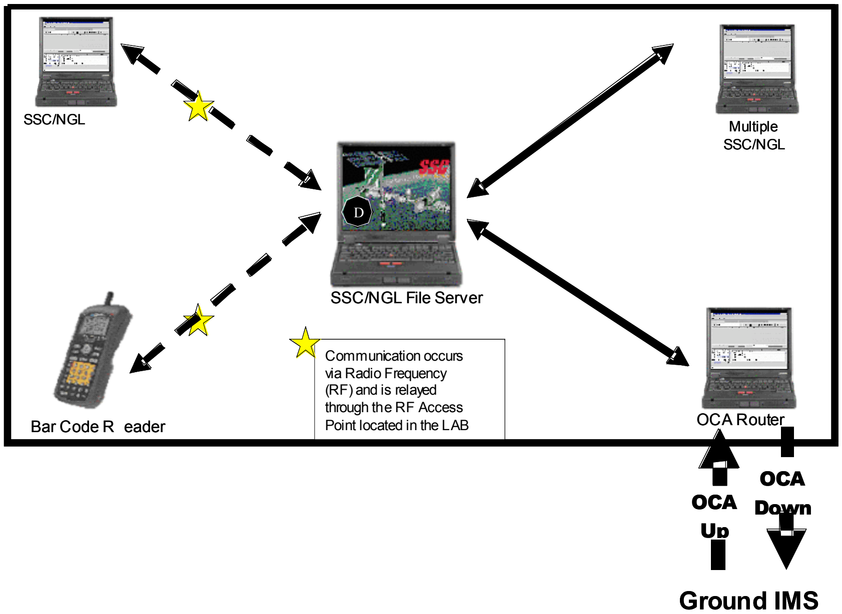


Figure 1: Asset Tracking System on ISS

Individual items on orbit are tagged with a unique barcode identifier. Stowage locations such as lockers and racks are also tagged with a unique barcode. Every time an item is removed and placed in a new location the transaction has to be manually logged (Figure 2).

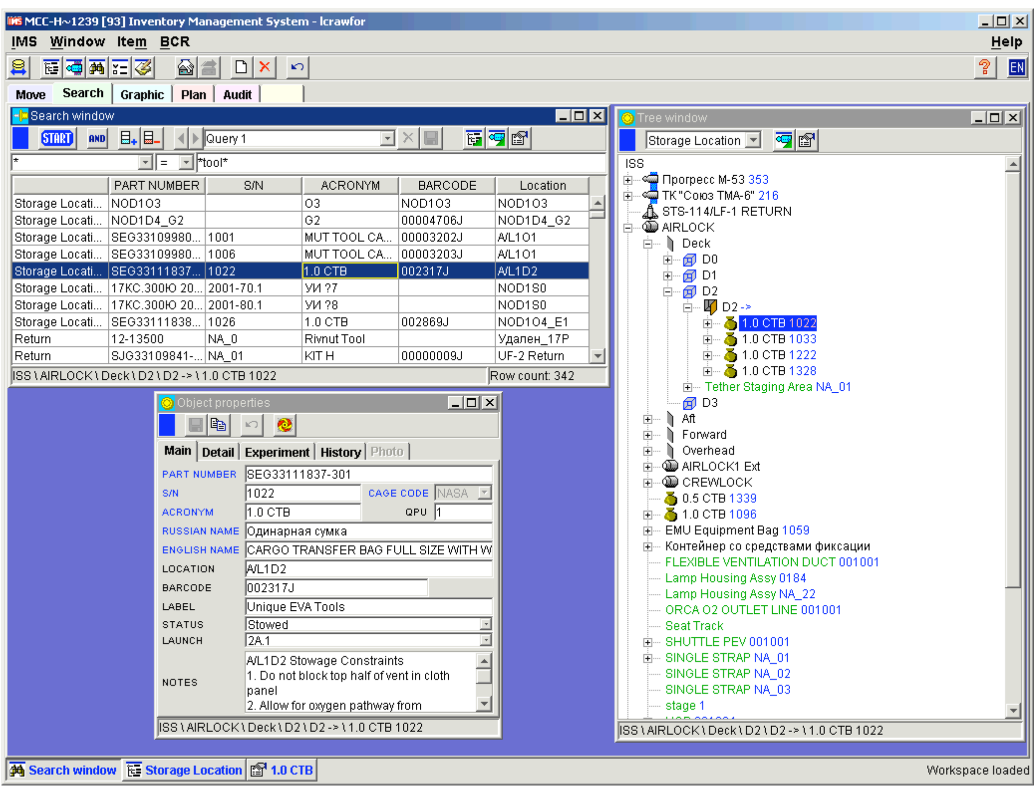


Figure 2: Joint US-Russian Inventory Management System (IMS)

The screenshot shown in Figure 2 shows the tabular, form-based user interface of the Inventory Management System (IMS). While usable and updated daily, this system requires a large group of personnel to ensure manual entries, daily batch updates and coordination between Houston, Moscow and ISS.

Although astronauts are allocated 20 minutes per day for IMS updates and asset management, the actual time spent on asset management and tracking is often much larger. If an item (e.g. a specialized tool for a certain maintenance procedure or a scientific experiment) cannot be found after several hours of searching (a single astronaut hour in space can be valued at about $180,000) flight managers will have to decide whether to search longer or resupply a new item on the next flight, potentially displacing other items.

1.2.2 RAMSES Approach

The RAMSES system consists of the following elements:

- Smart Containers
- Network-accessible database
- Rule-based Conditioning and Processing
- User Interface via Web-application

We envision a wide variety of mission-level NASA ground and flight system applications for the RAMSES system, as shown in Figure 3. The idea is that of hierarchical decomposition of the system with tracking being done at the lowest level (e.g. inside the nodes of ISS, in the CEV, in a ground test firing cell) and the results being sent to and aggregated at higher levels in a real-time data capture platform.

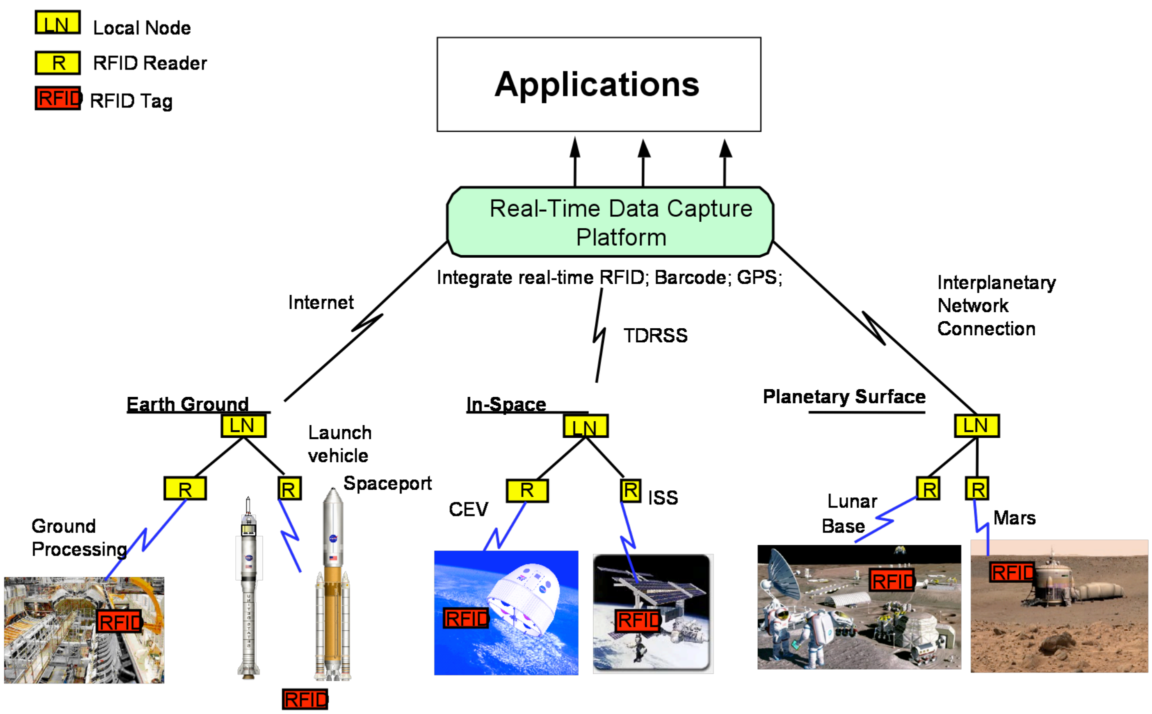


Figure 3: NASA ground/flight potential system applications of RAMSES

In order to bound the RAMSES effort, we have defined a set of specific asset tracking applications that can be implemented at a detailed level, and we use those applications to drive the overall system definition. We have defined requirements in five topical areas:

1. Over-arching/Architectural
2. Smart Container
3. Mobile Asset Tracking

4. Information Kiosk
5. External Tracking Application

Figure 4 shows the relationship between the physical and information infrastructures of RAMSES. We used the physical infrastructure (applications) to drive the major requirements, which were then implemented in the preliminary RAMSES information infrastructure. The physical infrastructure of RAMSES is decomposed into three levels:

1. Smart Container Level
2. Indoor Room Tracking Level
3. Outdoor and Facility Level Tracking

(Note that this architecture is defined in the context of a terrestrial application with the notion of rooms, indoors, and outdoors—but the extension to space exploration is obvious, with modules, vehicle/base interior, and vehicle/base exterior locations applied analogously. Furthermore, Google Maps are shown as the mapping information database, but other GIS systems may also be implemented to provide uniquely available data, e.g., Lunar/Mars surface maps.)

One of the aspects that makes RAMSES unique is that it has the ability to seamlessly track assets across multiple levels of this hierarchy and at all aspects of the ground/flight life cycle of any asset of interest—from procurement through integration and launch processing into flight implementation. This is achieved by a state-of-the art information architecture and software implementation (Figure 4, right side).

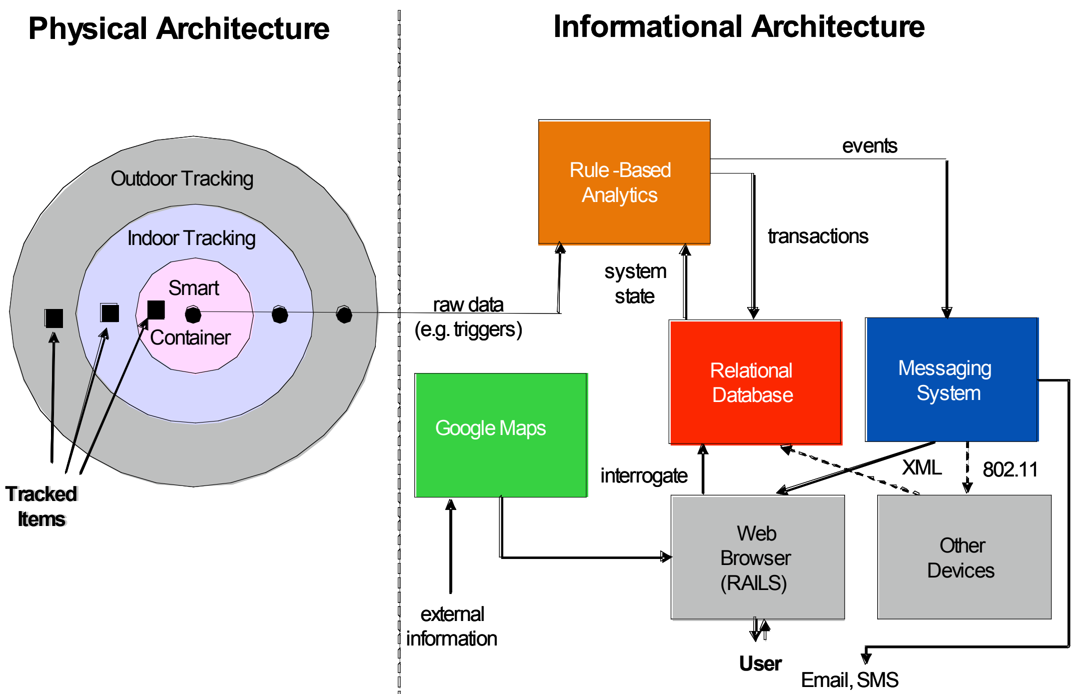


Figure 4: RAMSES Information and Physical Architecture (Terrestrial)

1.3 Significance of RAMSES

RAMSES addresses NASA's needs for reliable and low-cost asset management for Earth-based activities, robotic and human lunar exploration, and planning for expeditions to Mars and beyond. Specific advantages include the following:

- **Safety**: By tracking critical supplies and saving monitoring time, RAMSES will ensure that exploration and work locations are adequately supplied and provide a backup to astronauts and ground support crew monitoring.

- **Cost Savings**: When exploring, time is literally money. By reducing the time to track, locate, monitor, and check-out exploration assets, RAMSES will greatly increase astronaut and ground crew productivity, and this will increase the return on investment for a space mission.
- **Simplicity**: RAMSES will give NASA the ability to track any asset through its entire life cycle using only one system. From procurement through disposal an items status, location, history, etc. will be known. (Note: It is likely that this will only be implemented for a subset of assets where automated tracking is likely to yield a substantial payoff.) RAMSES can be accessed with a simple web browser and does not require installation of a local program. It is essentially a web service.
- **Unification**: The RAMSES asset management system will be able to interface with existing NASA systems to allow users to run extensive queries.
- **Autonomy**: Using RFID, assets will be tracked with minimal human interaction. This will decrease the crew/ground-time involved with inventory management and also decrease the "human error" associate with bar-code based tracking.

2 Technical Objectives

In Phase 2, we redesigned the *Smart Container* to take advantage of Generation 2 RFID technology, lightweight RFID readers and batteries, as well as integrate container technologies in a more tight and robust manner. We planned to build a total of three (3) smart containers to demonstrate scalability of the system and interaction among containers. The RAILS software was to be upgraded to provide enterprise-level capabilities such as user authentication, inventory and item time history analysis, and an expanded analytic rule set. Testing was to be conducted both in the laboratory and the field to demonstrate seamless three level tracking (container level, room level, outdoors). Finally, we were to perform a detailed cost/benefit and market analysis in preparation for Phase 3.

The four technical objectives of Phase 2 were:

1. **Second generation smart container design and manufacturing.** In order to meet our Phase 1 schedule and budget constraints, we focused our efforts on the smart container for which the most equipment was already on hand at PSI and MIT-SSL. Specifically we used a commercially available cooler and equipped it with pre-existing Alien RFID readers and antennas (details in Appendix B of the Phase 1 final report). This configuration was workable but far from optimal. For Phase 2, we developed a second-generation smart container through the integration of four new major hardware/software items - SkyeTek RFID reader boards, lightweight/low-power Gumstix Verdex/Bluetooth host electronics, WiFi communications, and lightweight, rechargeable lithium batteries.
2. **RAILS software implementation**. Upgrade the RAILS software to provide a friendlier GUI design, enterprise-level capabilities such as user authentication, inventory and item time history analysis, and an expanded analytic rule set. Additionally implement an external interface layer. This will was to allow RAILS to interact with a large variety of third party equipment and software.
3. **RAMSES testing**. Once the initial improvements in the smart container and RAILS had been accomplished, we began testing of hardware, software and usability. In general Phase 2 placed a heavier emphasis on system-level testing than did Phase 1. The testing strategy was to proceed "inside out", starting with rigorous testing of the smart container(s) first, followed by indoor testing, outdoor testing and multi-level testing.
4. **Cost/benefit and market Analysis**. In Phase 2 we were to conduct a detailed cost-benefit analysis of the RAMSES system. This is critical to understand where and when such a system would be a worthwhile investment. Additionally we were to perform a competitive market analysis and Phase 3 planning.

3 Work Conducted and Results Obtained

The following describes each of the key proposed Phase II effort tasks, quoting the original Phase II descriptions of those tasks, describes the work the team performed for each task, and describes the results or outputs from each activity. Additional documents provided with the software delivery package provide additional information and some of the available work products from those tasks.

3.1 Task 1: Requirements Definition

3.1.1 Work Performed

The purpose of the requirements definition task was to define the key top-level requirements for the RAMSES Smart Container, RAILS System Requirements and RAMSES Testing Requirements. During the early part of the Phase II effort, the team elaborated those top-level requirements, which then guided the remainder of the development effort.

The Smart Container requirements effort focused on the following description:

> **1.1 Smart Container**. The smart container Phase 1 prototype has demonstrated the basic functionality of automated item identification and inventory broadcast to an 802.11 network, but it has significant limitations in terms of interior volume intrusions, external local data display, and battery life. The Phase 2 requirements definition effort will address these issues and any others needed to advance the smart container to detailed design status.

During this phase of Phase II, the team elaborated the Smart Container requirements for two types of containers: a rigid body Smart Container, similar to that developed during Phase I; and a soft-shelled container, similar to the Cargo Transfer Bags (CTBs) used by NASA to deliver and store inventory items aboard the ISS. This analysis focused on the operations concepts for each type of container and the systems specifications for those containers. Additionally the team performed a subsystem requirements analysis, outlining the high-level functional operations sequences to be used by operators of the containers. The next section (Section 3.1.2) summarizes the resulting set of high-level requirements produced by this effort.

The RDF-based Asset Information and Location Software (RAILS) System requirements definition effort during Phase II was guided by the initial implementation from Phase I, as described below:

> **1.2 RAILS System**. Similarly to the smart container status described above, the RAILS system as of Phase 1 conclusion is adequate to demonstrate the feasibility of the concept, but is focused on smart container room-level tracking and does not fully implement the hierarchical architecture of RAMSES. The Phase 2 requirements definition effort focus on bringing the RAILS capability to full detailed design and coding for implementation on the complete RAMSES system.

The Phase II RAILS requirements work focused mobile asset flow monitoring and the notion of a RAMSES Information Kiosk. Both the mobile asset flow monitoring and information kiosk requirements work elaborated the operations concept and system specifications, but left a number of items open. Later development efforts effectively closed most of those open items.

Finally, the RAMSES Testing requirements definition effort focused on the following:

> **1.3 RAMSES Testing**. Thus far, the RAMSES system has operated within the confines of the PSI and MIT laboratory facilities. We are conscious that RAMSES is targeted to challenging applications in harsh environments, and we intend to mature the technology and system capabilities to accommodate these requirements. The RAMSES testing requirements development will focus on defining realistic testing environments and test objectives to demonstrate the flexibility and robustness of the system.

During Phase II, the team elaborated a set of high level testing requirements for RAMSES. These requirements elaborated the high-level goals for usability testing, indoor and outdoor tracking tests, testing the tracking of items as they transition from one container to another, and testing the ability of the system to maintain a history of an item's location. The work provided descriptions of each of these key goals and rationale behind each. The next section provides the results of this effort.

3.1.2 Results

The following enumerates and describes the resulting requirements and goals from the Phase II requirements development effort:

3.1.2.1 Smart Container/Smart Bag Requirements Analysis

This section lists the requirements for the smart container and smart-bag with the aim of identifying potentially common sub-components.

I. Smart Container Requirements

A. Operations Concepts

The smart container is a hard compartment used to track and monitor items, primarily in terrestrial environments. Metrics for the success of a smart container include, but are not limited to:

1. Cost
2. Reliability of reading mechanisms
3. Reliability of container electronics
4. Weight

B. System Specifications

- The container shall use RFID to monitor its own contents.
- The container shall be RF opaque so that items not in the smart container are not sensed.
- Initial prototypes shall be designed using an off the shelf cooler.
- The smart container shall be able to easily interface with a hand-held unit to display the contents of the container. Possibilities include a Tablet PC or PDA-type device.
- The smart container shall be able to “transmit” content information to a centralized database for inventory tracking.
- It is highly desired that the smart cabinet shall record when and by whom an item has been taken/inserted. The ability to record “by whom” requires associated RFID badges for the crew.
- The smart container should have the ability to produce a visible or audible alert when the RFID tag on an item has been read as the item is entering and leaving the container. If the alert is audible, this option must be able to be turned off.
- The smart container shall exhibit a 95% or better system accuracy.
- The smart container shall be designed to utilize a minimum amount of crew time.

II. Smart Cargo Bag Requirements

A. Operations Concepts

The smart cargo bag is envisioned as a flexible, modified NASA CTB that can be deployed in the Space Station or Lunar bases within years. Important metrics for the success of a Smart NASA Cargo Bag are similar to those of a smart container, however they differ in their *rank order*. In particular, Weight is probably the most important consideration, with cost perhaps the least important. Further, compatibility with existing cargo bags is added. Metrics for evaluation include:

1. Weight
2. Robustness of electronics
3. Reliability of reading mechanism
4. Level of compatibility with existing cargo bags and logistics technologies
5. Cost

B. System Specifications

- The container shall use RFID to monitor its own contents.
- The container shall be RF opaque so that items not in the smart container are not sensed.
- Initial prototypes shall be designed using an off the shelf cooler.
- The smart container shall be able to easily interface with a hand-held unit to display the contents of the container. Possibilities include a Tablet PC or PDA-type device.
- The smart container shall be able to "transmit" content information to a centralized database for inventory tracking.
- It is highly desired that the smart cabinet shall record when and by whom an item has been taken/inserted. The ability to record "by whom" requires associated RFID badges for the crew.
- The smart container should have the ability to produce a visible or audible alert when the RFID tag on an item has been read as the item is entering and leaving the container. If the alert is audible, this option must be able to be turned off.
- The smart container shall exhibit a 95% or better system accuracy.
- The smart container shall be designed to utilize a minimum amount of crew time.

- The smart container-CTB shall be sized comparable to an existing single, double or triple cargo transfer bag (CTB). Single CTB dimensions are 25 cm x 50 cm x 43 cm with an internal volume of 0.05 m3.
- The smart container-CTB shall be made of a flexible, lightweight material.

We find that, to first order, the system specifications for the smart bag are identical to those for the smart container, with the addition of compatibility requirements and the need for flexible elements.

III. Sub-System Analysis

Current Operations for the Smart Container Prototype are as Follows:

1. Upon opening of the lid, power up the electronics
2. When powered, determine the location area of the container by polling power levels of registered and locally scanned WiFi access points
3. Upon closure of the lid, take an inventory of all items within the container
4. Upon connection to the network, transmit the closest access point id and container inventory to the RAILS server
5. Upon connection from a local bluetooth connected client computer, display of that inventory through the client's web browser
6. If configured by the power switch, turn off the electronics about 5 minutes after the lid is closed

These basic operations should be compared against similar operations performed by a smart cargo bag. The following table is an initial sketch of this comparison:

Table 1: Functional Comparison of Smart Container and Smart Cargo Bag.

FUNCTIONS	Container	Cargo Bag
Power On	Open Lid	Open Bag Zipper
Self Location	Poll power-levels of registered and locally scanned WiFi access points	Does the station have wifi? Yes.
Self Inventory	Internal tag, reader, antenna	similar tag, light-weight reader, flexible antenna
Transmit location and inventory		

Local display after bluetooth connect	bluetooth	bluetooth
Power Off	self-timer	self-timer

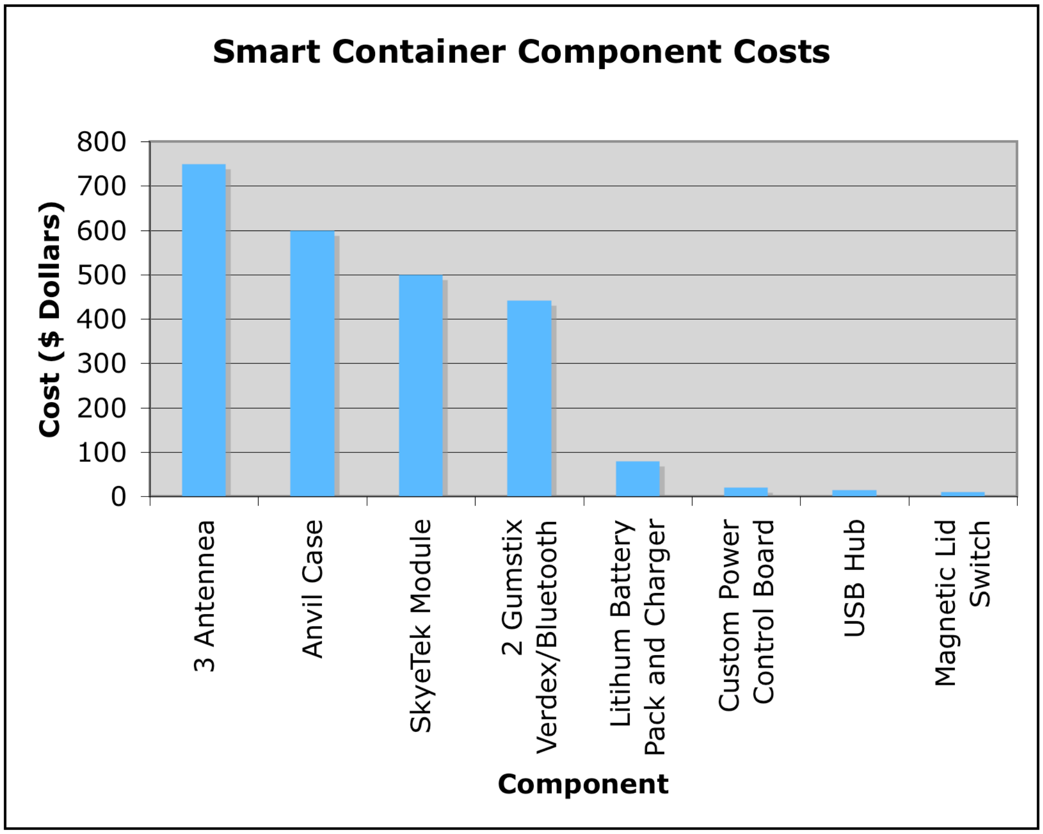


Figure 5: Estimated Smart Container Component Costs. Many of these costs are an artifact of previous decisions (ie Alien Antennae) or work-arounds (ie Skytek Module) and will drop with further design.

IV. Conclusions

- The rank order of metrics for the Cargo Bag and the Smart Container or roughly reversed. The importance of read-reliability is roughly compatible, however, cost reduction and weight reduction are valued differently for each.
- Software (which is a cost driver) can be made nearly identical for both units
- Hardware: A general task is to examine smart container components to determine how to minimize weight and size and to determine *specifically* how this increases cost and affects reliability.
- Good news is that our current highest cost elements are precisely those which would need to be altered for a smart bag—the casing and the antennae. An analysis of alternative technology for these components could prove very valuable for both development efforts.
- The other components will demand a trade between size/weight minimization and cost. Break points for this trade will likely differ for each unit, and result in different sub-components for some of the electronics. That is, unless we find a way to integrate most of the functions into one chips, which is possible.
- Specific Hardware-related questions to be answered:

    1. What is the cost differential on a Flexible Antennae?

2. Look at existing bags to determine compatibility, especially wrt open/close activation
3. Determine if Station has WiFi and if not, how beacons can be used for locating
4. Identify RF opaque material for smart container

3.1.2.2 RAILS System Requirements Analysis

I. RAMSES Mobile Asset Flow Monitoring

The purpose of these requirements are to describe the monitoring of the flow of assets of a given size or greater around the interior and exterior of an exploration site or vehicle; integrating this information with complementary data-sources using rule-based analytics; transmitting reports, alerts, and other value-added information.

A. Operations Concept

1. Complete visibility of the location of critical assets and smart containers across the site at the *room level.*
2. Integration with higher fidelity tracking capabilities interior to a site such as smart containers, smart bags, and exterior to a site such as GPS or other methods of triangulation.
3. Integration with any number of additional data-sources, such as temperature, radiation, diagnostics, and other sensors to create value-added capabilities using rule-based analytics.

B. System Specifications

1. The system shall track the movement, history and location of smart containers and selected items of equal or larger size than smart containers both inside and outside a site.
2. The system shall identify the location of *safety critical* items with accuracy to be determined.
3. The system shall identify the location of *non-safety critical* items/smart containers at the *room-level*.
4. The system shall be integrated with additional data sources and tracking devices, and easily scalable.
5. The system shall relay processed data, using rule-based analytics, to an information kiosk or other presentation means.

C. Open Issues

1. Basic Design Goals: Simplicity of use, Autonomy, Reliability, low weight
2. RF allowed interior to a space vehicle? If not, then the wider tracking system may be designed only for a planetary site or remote earth environment.
3. Some technical issues: Tag type; Transceiver Design; Transceiver Power Level; “tunability”
4. Levels of redundancy
5. How big should a smart bag be?
6. Should bag be actively tracked or just queried manually with a read/hinge?
7. Kind of information that could be integrated with location data to create value

II. Requirements for the RAMSES Information Kiosk

These requirements establish the operational concept and requirements for the rule based analytic asset management for space exploration systems (RAMSES) information kiosk.

A. Operations Concept

The RAMSES system seeks to provide integrated asset tracking and management using advanced technologies such as Radio Frequency Identification (RFID) and rule-based analysis. Data associated with the assets (i.e., location, operator, temperature history) will be stored in an integrated database. User access to the RAMSES database is provided by several tools, including handheld devices (e.g., PDA), computer displays, and a stand-alone ***information kiosk***.

The information kiosk is intended to permit a user to access any type of available RAMSES data without the need for a keyboard or other separate input device. It is also intended to provide the information service function in a harsh environment, such as a remote field site (e.g., Haughton Mars Project) with eventual extension to a lunar or Mars base, so environmental considerations must also be addressed.

B. System Specifications

1. The information kiosk shall provide end user access to RAMSES data in accordance with established user access privileges.
2. The information kiosk shall provide users with information in a format applicable to user type (e.g., field geologist, logistics officer).
3. The information kiosk shall enable user access under the environmental conditions associated with field operations (e.g., temperature, dust).
4. The information kiosk shall support user input via a single interface (e.g., touchscreen) and not require the user to provide a separate input device.
5. The information kiosk shall support the delivery of information to the user via screen and/or printed output.
6. The information kiosk shall support the upload of remotely collected user data via external storage device (e.g., USB keychain drive, PDA).
7. The information kiosk shall support the download of information to an external storage device (eg. USB keychain drive, PDA).

C. Open Issues

1. Input device (touchscreen?)
2. Supported input and output devices/interfaces (USB, Bluetooth, etc.)

3.1.2.3 RAMSES Testing Requirements Analysis

Figure 4 in Section 1.2.2 illustrates the overall RAMSES Information and Physical Architecture. The Physical Architecture delineates the nesting of item tracking intended by the RAMSES architecture, from the contents in a Smart Container, to items within an indoor environment, to full outdoor tracking of assets.

During Phase 1, the team performed limited evaluation of Smart Container item tracking performance in a laboratory environment and informal testing of zone tracking in an office environment based on WiFi locations. For Phase 2, the test environments will be expanded to cover indoor and outdoor tracking of assets. The following summarizes the main test environments for Phase 2:

1. **Usability Testing.** The RAMSES testing activities shall assess the ability to locate individual assets during regular use using the Smart Container in a research or exploration analogous environment.

During Phase 1, the students from MIT performed accuracy and performance tests through controlled scenarios and experiments. For Phase 2, the goal is to determine how well the Smart Container and RAILS software perform under regular operating conditions.

2. **Indoor Tracking.** The RAMSES testing activities shall assess the ability to locate assets contained within a Smart Container as the container moves within a facility or campus.

   During Phase 1, the team performed informal proof-of-concept tests and demonstrations of tracking the Smart Container as it moved through the facilities at Payload Systems. The goal is to measure the effectiveness of this tracking technique and expand the test environment to additional facilities.

3. **Outdoor Tracking.** The RAMSES testing activities shall assess the ability to locate assets within a Smart Container as the Smart Container moves between facilities.

   The goal is to determine the effectiveness and accuracy of tracking items beyond the confines of a controlled facility and environment. During Phase 1, the technologies were limited to local RFID and WiFi asset management. During Phase 2, we will expand this to integrate technologies such as GPS.

4. **Item Transitions.** The RAMSES testing activities shall assess the ability to locate of assets as they move from one Smart Container to another within a facility or campus.

   During Phase 1, the team was limited to a single RFID system. This allowed us to test items appearing and disappearing from the same container, but did not allow testing the transfer of assets from one container to another. The goal of these tests is to determine the effectiveness of tracking such transitions.

5. **Location History.** The RAMSES testing activities shall assess the ability to determine the location history of an item as it transitions between locations and/or Smart Containers

   During conversations with JSC for the market analysis portion of this project, interviewees indicated the importance of knowing the environmental conditions of particular types of assets, particularly during transit to international facilities. By tracking the location history and timestamp of an item, one may infer the environmental exposure history of a particular item. The goal of these tests is to determine the effectiveness of RAMSES in determining this location history.

3.2 Task 2: Smart Container Design and Manufacturing

3.2.1 Work Performed

The purpose of the Phase II Smart Container Design and Manufacturing task was to advance the rapid demonstration Smart Container prototype developed during Phase I to a more robust, demonstrable system for Phase II. As proposed, this task was broken into a series of key subtasks, as described in the original proposal:

2.1 **Integration of RFID readers and antennas** into the rigid side walls of the smart container to ensure that no RFID equipment protrudes, as well as weight minimization by embedding only RFID PCB assemblies and connectors, while omitting OEM housings. All RFID readers and antennas will be flush with container walls. RF analysis will be conducted to optimize antenna location and orientation such that the highest antenna gain and read accuracy can be achieved inside the smart container, while avoiding dead corners.

2.2 **Optimization of rechargeable power system** to allow for lightweight and rechargeable batteries (e.g. lithium ion), DC power circuitry and external recharging adapters. The Phase I prototype smart container power circuitry and logic will be refined into production level PCBs with optimized standby and wake-up modes and adjustable timing.

2.3 **Lid-integrated tablet computer**. A small lightweight PC-based computer will be integrated into the lid of the computer in order to act as the “brains” of the smart container. This unit will both control the power circuitry and the RFID reader operations. In addition it will act as a node in the wireless 802.11 network in order to provide continuous smart container room-level tracking in all environments that are 802.11 equipped. Additionally, the tablet PC will act as a local information kiosk for each smart container. Through its touch-screen (integrated in the lid) the user will be able to directly query the contents of the smart container without having to open the lid directly. It can also act as an interface to the rest of the RAMSES system. Finally, the second generation smart container will be equipped with a set of user friendly colored LEDs on the lid (or side) to directly indicate the current status of the smart container (on/off, RFID reader on standby/active, recharging).

2.4 **Robust RF shielding and RFID compatibility**. The smart container will be coated or painted with an RF opaque material both on the inside and outside to avoid interference issues with the container. This will ensure both that external readers don’t read the inside contents of the smart container and that the RFID reader doesn’t pickup tags on the outside of the container. Additionally careful attention will be paid to issues such as multi-path within the smart container as well as accidental short-circuiting or grounding of tags within the container.

2.5 **Smart Container Manufacturing**. Our plan is to again use commercially available large coolers to create three (3) instances of smart containers in Phase 2. The difference, however is that the RFID, power and other equipment will be optimized and tightly integrated.

The Smart Container design and manufacturing work performed during Phase II took on several phases. The first phase was driven by an opportunity to demonstrate the RAMSES technology as part of Wired Magazine's NextFest in September of 2007 and resulted in the design and manufacture of two rigid Smart Containers.[1] Following this initial effort, the team supported the construction of a third Smart Container by students at MIT (the structure was paid for and owned by MIT) for a field test at Mars Desert Research Station (MDRS) in the early spring of 2008. The team loaned the students electronics and antennae from one the rigid Smart Containers and provided access to the RAILS software for their research effort. As a result, these two design and manufacturing efforts supported field test of the Smart Container and two different RAILS software architectures (more on this later). The final phases of the Smart Container development adapted the design to a soft-shelled Cargo Transfer Bag (CTB), on loan to MIT from Kennedy Space Center (KSC). MIT students performed the initial design and test, borrowing the electronics and antennae from one of the rigid Smart Containers. Realizing that the CTB shells would eventually be returned to KSC, the team developed the shell and electronics for a third container utilizing a soft-shelled luggage bag as a proxy for a CTB, known as the CTB-proxy. During this effort, MIT's cost-benefit analysis indicated that the ability to retrofit a CTB already on orbit would provide significant benefits. As a result, the team prototyped a retrofit kit for a CTB and built and integrated the final set of electronics to integrate with either the CTB-proxy or the CTB retrofit kit. This final version, an instrumented CTB built up from the retrofit kit, was eventually used by MIT for the final usability testing in the winter of 2008/2009.

During this effort, the team undertook a number of quick studies to attempt to address some of the issues of the Smart Container designs. Some of these resulted in design changes during the effort, particularly with respect to the CTB and CTB-proxy implementations. These included:

[1] The original plan was to have one Calzone/Anvil case for Wired NextFest, but due to uncertainty in the delivery schedule of the Anvil case, the team hand-built the 2nd case to ensure at least one would be ready for the event. As it happened, both containers were available and provided the opportunity to test tracking inventory items from one container to another (see Item 4 in Section 3.1.2.3).

- Identification and selection of a lightweight, customizable RFID reader. This resulted in the selection of the SkyeTek RFID reader
- Identification and selection of small, lightweight host electronics for the RFID reader. This resulted in the selection of the Gumstix Verdex processor. In addition to this selection, the team expected the eventual availability of Gumstix Goliath electronics, a GPRS/Edge board (cell-phone communications) with Ground Position Sensor (GPS). Unfortunately, Gumstix abandoned this effort and the team relied instead on off-the-shelf GPS and cell-phone network USB devices for outdoor tracking and communications.
- Examination and selection of thin RFID antennae. By the end of Phase I, the team had been using Alien Technology's RFID circularly polarized antennae. While adequate for the rigid Smart Containers, these antennae proved too large and heavy for the CTB and CTB-proxy containers. As a result, the team researched alternative antennae and, after an initial performance evaluation, settled on Mobile Mark BP6-915LCP for the CTB and CTB-proxy containers.
- Examination of potential water and metal tolerant RFID tags. During Phase I, the team relied on Gen-1 UHF RFID tags. These tags had significant difficulty when placed too close to water and metal. The original expectation was that Gen-2 tag technology would mitigate these issues. Unfortunately, the Alien Gen-2 tags exhibited similar issues to the Gen-1 tags. Late in the program, however, developments in metal and water tolerant tags advanced, and the team purchased a small quantity of these for testing, which resulted in significant performance improvement to the read rates next to metal and water. These tags, however, were comparatively large and rigid compared to the small Gen-2 sticker tags we'd been using. Since that time, however, the team discovered a promising review of the Omni-Id tags but, at this stage of the program, do not have the resources to acquire and test these tags and left this task for a potential Phase III effort.
- Examination of alternative shielding approaches for the Smart Containers to optimize interior space. In order to prevent reads of RFID tags outside the bounds of a given container, each container utilizes RF shielding. By the end of Phase I, the team settled on an initial approach utilizing aluminum foil attached to the inner surface but recommended examination of RF-absorbing material rather than reflective material for shielding. During Phase II, the team obtained some small samples of broadband and narrow band RF-absorbing material and tested both the shielding ability and compatibility of the material with the Gen-2 tags the team had on-hand. The team determined that these materials were not sufficiently effective. Finally, during the prototyping of the soft-shelled containers, the MIT team settled on copper mesh rather than aluminum foil. This made the material attachable to the spacer foam inserts for the soft-shelled container and more robust to tears and general wear.
- Optimization of separation foam thickness. Due to a cost-benefit analysis performed by MIT, the team determined that the high usable volume efficiency of the container is an important factor in the overall value of the Smart Container approach for ISS. Enabled by the thinner and smaller RFID antennae selection, the team worked to reduce the spacing material thickness within the CTB and CTB-proxy containers.

In addition to the design and construction of the rigid and soft-shelled Smart Containers, the team needed to perform refurbishment of one of the Smart Containers due to shipping damage coming back from Wired NextFest and demonstrations at Disney.

3.2.2 Results

3.2.2.1 Rigid Containers

During Phase II, the team created two prototype rigid RAMSES containers. The first container was developed at Payload Systems Inc. The box was built with ¼" plywood with aluminum sheets attached to the exterior of the container.

Figure 6: RAMSES Container #1 exterior

Figure 7: RAMSES Container #1 interior.

The outer dimensions of the PSI RAMSES container is 24" x 30" x 24". The inner usable dimension is 18" x 24" x 18'. Two trap doors were cut into the container. One trap door is located in the lid.

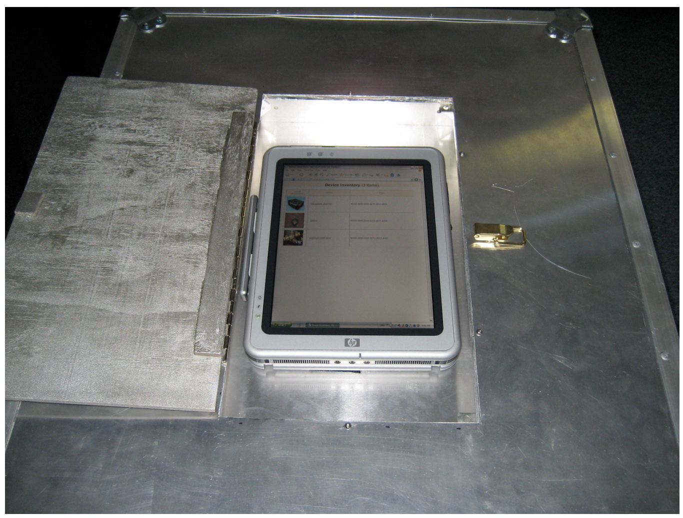


Figure 8: RAMSES Container #1 with tablet computer.

This trap door is used to hold a Tablet PC. Attached under the trap door is an aluminum container used to attach the Tablet PC. The other trap door is located on the back panel. It is used to allow access to the internal electronics.

Internally, the box was lined with aluminum shielding in order to allow the electronics to only see the RFID tags placed into the RAMSES container. On top of the aluminum is a 3" wide foam insert to shield the RFID tagged items from the aluminum shielding. If the RFID tag were to come in contact with the aluminum shielding the tag may short and not function properly.

Figure 9: RAMSES Container #1 with RFID-tagged items.

The electronics included in the RAMSES Container are listed below:

- 3 Alien Technology UHF antennas attached to the bottom wall, front wall and right side wall of the RAMSES container.
- 1 Skytek Inc. SkyePlus M9 UHF reader.
- 1 Skytek Inc. SkyePlus UHF 4-port Multiplexer for M9. It is used to allow the three Alien antennas to be read by the SkyePlus M9.
- 1 Belkin WiFi Dongle used to allow WiFi communications.
- 2 Gumstix Verdex XM4-bt processors each with a Console-vx module. The Gumstix processors communicate via Bluetooth. One of the Gumstix is used to interface with the SkyePlus M9 module. The other Gumstix module is used to perform WiFi communications with a host computer.

  Note: The use of two Gumstix was the result of early Phase II tests with the SkyeTek M9. These tests determined that if another USB slave device was attached to the same bus as the M9, the M9 would fail to communicate properly with the Host computer (regardless of if the SkyeTek was powered through the USB bus or was powered independently). Due to scheduling constraints and the lack of an immediate WiFi solution from Gumstix for the Verdex (this has since changed), the team opted for a 2nd Gumstix, networked through Bluetooth, to handle the WiFi communications task. Once the team developed the electronics for the CTB system, a WiFi solution for the Verdex became available, and the design for that electronics set consists of a single Gumstix with a WiFi attachment on-board.

- 1 Magnetic lid switch. When the lid is opened and then closed, this will cause the system to do an inventory scan for all RFID tagged items.
- 1 Custom-made lid switch/power timer card. This card is used to allow a user to control the power for the RAMSES container electronics. This circuit will allow the user to set the amount of time the power will stay on once the lid switch is closed. The settings for the power circuit are as follows: Off, Always on, and Timer Mode (pre-set time).
- 1 5V Lithium Battery Pack and Wall Charger.

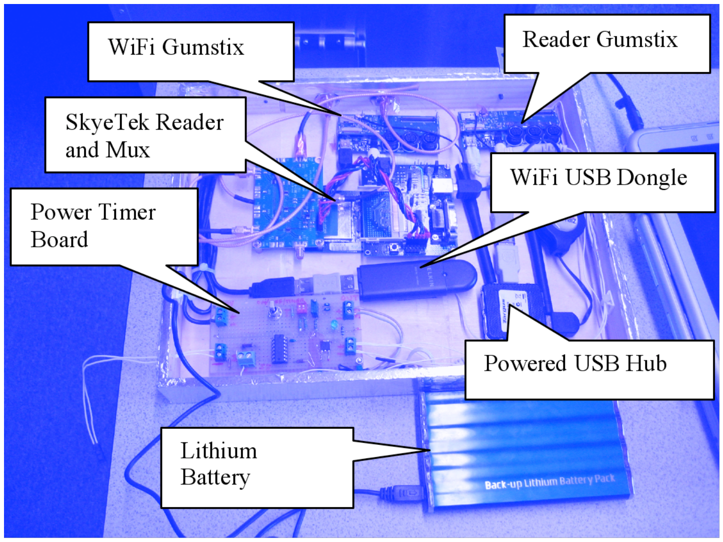


Figure 10: RAMSES Container #1 electronics and power supply.

The electronics for the RAMSES Container is located in a box that is attached to the container on the back panel behind the trap door.

The Tablet PC communicates with the embedded Gumstix electronics through Bluetooth and views the current container inventory provided by a web-server running on the Gumstix. The 2nd Gumstix and its attached WiFi USB dongle sends the inventory information back to the RAILS software when connected to the internet through a WiFi access point. Alternatively, administrators can configure the system to route the network information from the Bluetooth network, through the Tablet PC (or any other Bluetooth PC) to a wired or wireless network attached to the Tablet. This was the configuration the team used to demonstrate communications through a cell-phone data service.

The second RAMSES container is a custom built case from Calzone-Anvil Cases and contained identical electronics to the first RAMSES container.

Figure 11: RAMSES Container #2.

The second container is similar to the first container except for the following:

- The outer dimensions are 24” x 30” x27” and the inner dimensions are 18” x 24” x21”.
- The container is made out of ½” plywood and the outer shell is grey laminate.
- Larger rear trap door.
- Recessed handles and latches
- 4 Wheels for more mobility.

Figure 12: RAMSES Container #2 with RFID-tagged items and tablet PC.

3.2.2.2 Soft-shelled Containers

The RAMSES team adapted the Smart Container design to soft-shelled container in order to support the integration of RFID technology into NASA Cargo Transfer Bags. Figure 13 illustrates a module filled with such containers. There are essentially two sizes of containers the team focused on: single-height CTBs (shown on the top right corner) and double height CTBs (shown on the top left corner).

Figure 13. Cargo Transfer Bag storage area

MIT students performed the initial work on adapting the Smart Container approach to some CTBs on loan to MIT from Johnson Space Center (see . The students started by shielding a CTB, with spacer inserts and attaching an off-the-shelf High Frequency (HF) RFID System. The students discovered that HF RFID systems have a range of only a few centimeters, although had better tolerance for liquids and metals than

Ultra High Frequency (UHF) systems. For a single-height CTB, this seemed sufficent for the tests, but inadequate for a double-height CTB. Later, the students did an initial adapatation to a double-height CTB using borrowed UHF electronics.

Figure 14. Single-height CTB with HF RFID Reader

Based on the student's work, the team realized that the CTBs would need to be returned and began work on a proxy for the CTB, utilizing a soft-shelled luggage bag. Using the foam spacers and copper mesh shielding approach used by the students, and using the smaller and lighter Mobile Mark UHF antennae discovered during the development effort, the RAMSES team built a CTB-proxy shown in Figure 15 and Figure 16. The electronics were simplified down to one Verdex board, due to the availability of a WiFi attachment for the Verdex. Additionally, the electronics were housed in a separate electronics box that fit within the pocket of the soft bag. The remainder of the electronics system is identical to those use for the rigid containers.

Figure 15. Antenna within CTB-proxy shielded insert

Figure 16. CTB-proxy with lid open and insert lid removed.

During these efforts, MIT performed a cost-benefit analysis study of different approaches to the deployment of a Smart Container to the ISS. One result of this study was the recommendation that, rather than ship an entire instrumented Smart Container to the ISS, one might instead send up a retrofit kit, enabling the crew to instrument a container already on-board the ISS. As a result of this recommendation, the team developed a prototype retrofit kit and assembly instructions for a Smart CTB and tested the assembly and installation of the kit into the double height CTB on loan from KSC and the electronics and antennae from the CTB-proxy (see Figure 17, Figure 18 and Figure 19).

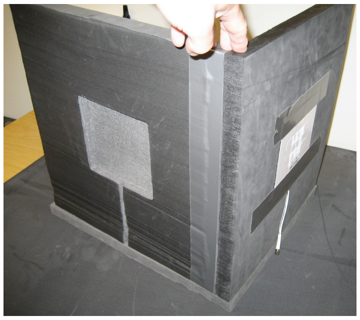


Figure 17. Partial assembly of retrofit kit

Figure 18. Installed retrofit kit (lid open)

Figure 19. Completed retrofitted CTB (lid closed)

3.3 Task 3: RAILS Implementation

3.3.1 Work Performed

The purpose of the Phase II RDF-based Asset Information and Location Software (RAILS) Implementation task was to advance the rapid demonstration service and client software prototype developed during Phase I to a more robust, demonstrable system for Phase II. As proposed, this task was broken into a series of key subtasks, as described in the original proposal:

3.1 **GUI Design**. The layout of the RAMSES GUI will be refined. The guiding principles are usability and simplicity. The GUI will be structured in a layered fashion, ensuring that those functions that are used most often (e.g. locating an item) are most easily accessible. This task also includes the improvement of graphical capabilities, more complete exploitation of the Google Maps API and incorporation of advanced GIS functionality. The GUI will be subjected to usability testing using the IEEE 829 software testing standard. The RAMSES GUI will be compatible with Internet Explorer and Firefox.

3.2 **Enterprise-Level Functions.** RAILS will be upgraded to include user authentication (user names, passwords) and enterprise level functions. Examples are the concept of item ownership as well as various levels of privileges for locating, viewing and editing asset information. The issue of permissions and security are paramount to making RAILS into a commercially viable tool, both inside and outside of NASA.

3.3 **Inventory and Item History Analysis.** RAILS will include new functionality to facilitate complete inventory updating and analysis of a certain smart container, a certain geographical area or only a subset of items. The class of supply (COS) system developed for SpaceNet 1.3 will be implemented. E.g. RAMSES will be able to report the aggregate mass and geographical distribution of all items of COS 2 (crew consumables). It will be able to store and show graphically the history of a tagged item over time.

3.4 **Expanded Analytic Rule Set**. The rule set will be upgraded to include more useful rules and demonstrate the versatility of the system. The exact rule set is still to be defined, but a candidate list is given below:

- Enhanced incompatibility rule (two or more items cannot be located within a smart container or less than X distance [m] from each other

- Critical supply level rule: A “low inventory” rule generates warnings when certain classes of supply reach pre-defined safety stock levels, prompting messages and potentially automatic reordering
- Expiration data: RAMSES RAILS (v2) will allow handling a range of shelf life and expiration rules. The simplest rule is based on a fixed expiration calendar day. More advanced rules will include conditional expiration (e.g. based on location, temperature history …)
- Temperature history profile: this capability will involve data fusion. We will allow RAILS to read temperature sensor data and automatically associate this temperature with any item that is located with a distance X [m] from this sensor. As items move, different temperature sensors are associated with the asset, and the correct temperature history is stored with the item

3.5 **External Interface Layer**: Future work includes improving the portability of the underlying server software to non-GNU/Linux operating systems, and improvements in scalability and usability of the system. The center of Figure 20 illustrates a notional scheme to extract and integrate information from third party sources, such as a shipping supplier or external database. The current version of the prototype does not yet implement this third-party component, but this will be tackled in Phase 2.

During the initial stage of Phase II, the team enhanced and modified the original Phase I prototype server software. This software was based on a set of Python scripts in conjunction with hand-written Javascript and HTML. These scripts utilized an open-source Resource Definition Framework (RDF) and Semantic Rule Engine library by Tim Berners Lee known as the Closed World Machine (CWM). This software ran under an Apache web server on a GNU/Linux operating system and accepted inventory lists and case locations through URL posts to the server from the Smart Containers. The software also utilized the Google Maps Application Program Interface (API) to render mapping and floor plans of geo-located inventory items and RAMSES facilities. This was the architecture the team used for the demonstrations at Wired NextFest in 2007 - Smart Containers reporting their contents to the RAILS server back in Cambridge, which then presented the inventory and mapping information back to the demonstration in Los Angeles via local web browsers.

This demonstration uncovered two key issues with the system:

- Reliance soley on WiFi for external communications is unreliable in unfamiliar or mis-configured environments
- Although extremely flexible, tools supporting RDF/Semantic Web technologies were not yet mature enough to scale with the number of items in an inventory

To address the first issue, the team examined the use of off-the-shelf cell-phone data networking, such as a GPRS/Edge card or cell-phone networking card. Around 2007/2008, Gumstix was advertising a GPRS/Edge card that would integrate with the Verdex host computer. After several months, it became clear that Gumstix would not be capable of delivering these cards and eventually dropped the product. At this point, the team moved to a less integrated solution based on cell-provider solutions. The first used a Verizon cell-phone USB network dongle attached to the Tablet PC. The Tablet PC was then configured to route the network traffic from the Bluetooth connection through to the card. This enabled the Smart Container to communicate with the RAILS server from anywhere were there was cell coverage. The second test utilized one of the team member's personal Bluetooth cell phones network to route the traffic. This also proved to be successful.

To address the second issue, Fabrice Granzoto, a visiting student to MIT, performed a study of available rule engines and approaches. Based on his study, the team transitioned from the Apache/RDF/CWM architecture to one based on RedHat's JBoss and JBoss Rules. During the winter of 2007/2008, Fabrice wrote a relatively complete RAILS server application and provided this to the RAMSES development team. Fabrice returned to his home university in early 2008 and the RAMSES development team continued completing the remaining features and discovering and fixing defects with the new server. The software delivery contains Fabrice's thesis report on the server development and provides the User's Guide to the server.

At this stage the RAILS server is feature complete, with some known issues and pending enhancements.

3.3.2 Results

At this state, the RAILS server is feature complete, with some know issues and pending enhancements. The following lists the key proposed enhancements and provides screen shots and brief descriptions of how the current version of the RAILS server software addresses each. A detailed description of the RAILS software server is available in the "RAMSES User's Guide", and in Fabrice's thesis report.

3.3.2.1 GUI Design

The final version of the RAILS Version 2 (RailsV2) software GUI uses RichFaces, an open-source component library for JavaServer Faces, for much of the web application rendering and user interaction. This library provides active widgets and user-interactive elements, such as scrollable or paged lists and visual elements, without the need to hand-code Javascript.

Figure 20 provides an illustrative example of one of the RAILS screens. On the top left of the screen is the owner's logo, in this case a NASA logo. On the top right is the currently connected user and a command button to disconnect (i.e. logout) from the system. The center menu list provides selection items to view the **Home**, **Asset**, **Facility**, **Map**, **Rules** and **Analysis** pages; this example shows the **Facility** view. Below the main menu list is the page-specific sub menus; in this case the **Facility Search** and **Browse** submenus. Below the menu bar, the left side contains a search input and results section. User's enter all or a portion of a facility or asset name, depending on the main page they're on, and all matches appear as command links below the search. The user can then select one of the results and the details of that item will appear on the right hand of the screen, in this example, the details of the "Ramses Case 1." Below the facility information summary, the screen displays a series of tabs. Each tab provides detailed information and/or lists about the selected item.

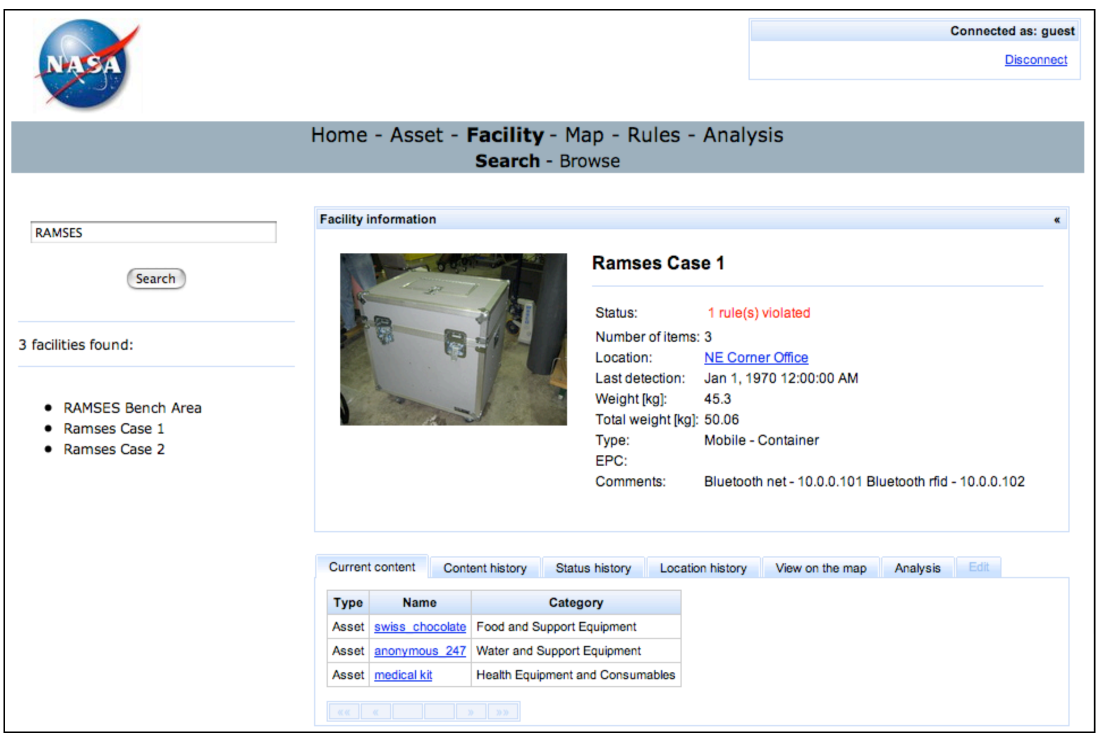


Figure 20. Example Facility Search Result

One of the asset and facility detail tabs show a Google Maps view of the location (shown in Figure 21) of the selected asset or facility. This display will outline the bounding box of the facility or region, or facility or region containing the asset. Alternatively, the main **Map** menu will show the map display with its own search parameters (details of the selected item will not be shown in that view).

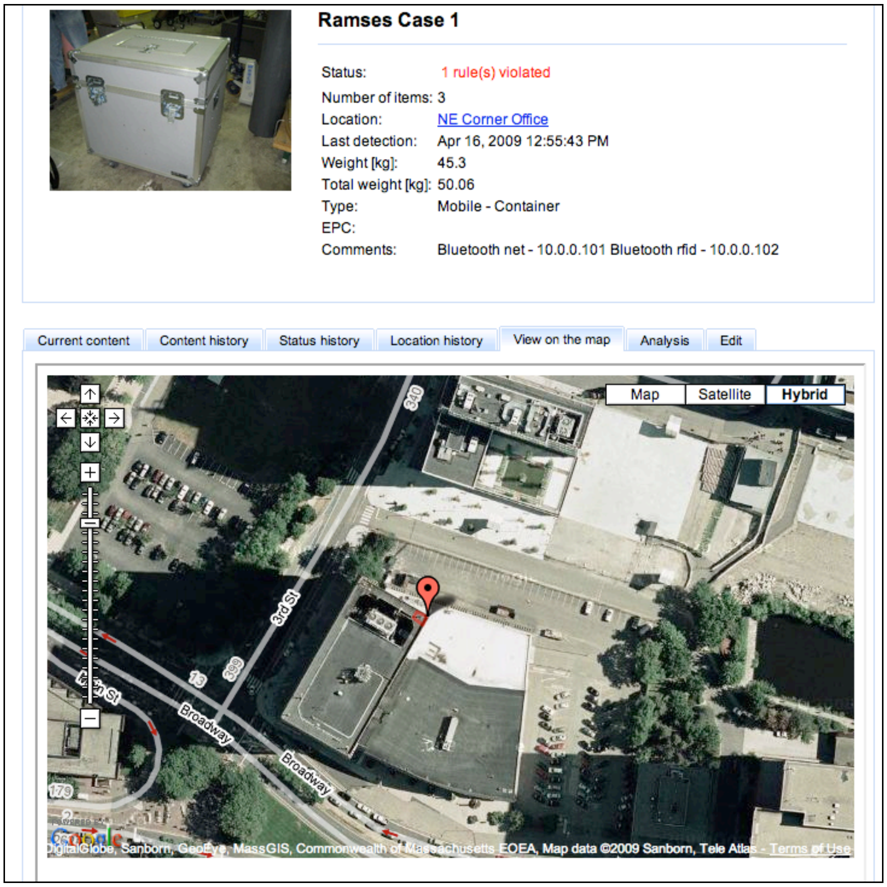


Figure 21. Google Maps Display

3.3.2.2 Enterprise-Level Functions

Using the JBoss authentication services, the RailsV2 server provides user authentication and access control to inventory assets. RailsV2 is currently organized around users and user roles. The software currently provides four fixed roles: Administrator, Inventory Manager, Inventory User and Guest. It relies on the JBoss authentication configuration files to manage users and user passwords and map those users to one or more of those four roles. Subsequent versions of the software may expand the role set, or make the role sets configurable. The RailsV2 GUI configures its displays based on the permissions granted to the current user's set of roles. Figure 22 illustrates the main login screen and Figure 23 illustrates the default **Home** screen for an Administrator or Inventory Manager role.

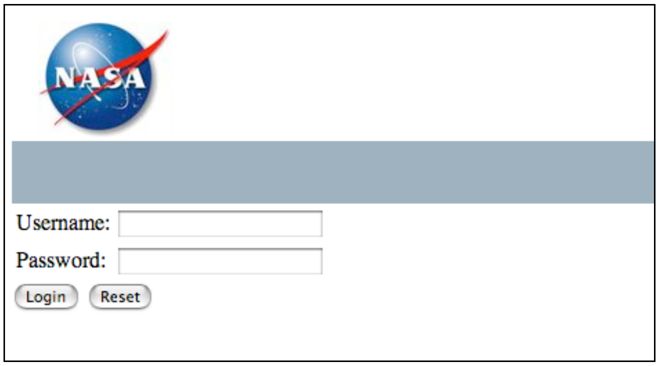


Figure 22. RailsV2 User Login

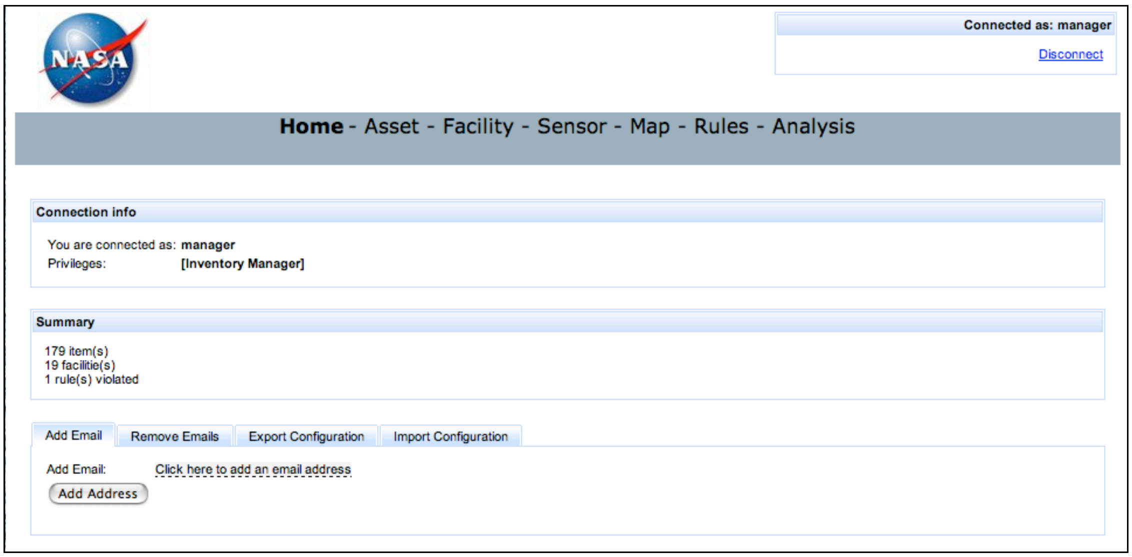


Figure 23. Home Screen after Login

The roles provided by RailsV2 govern what assets are viewable by the current user and whether or not the current user can edit a given asset's properties. These permissions are configurable on an asset-by-asset basis, as shown in Figure 24.

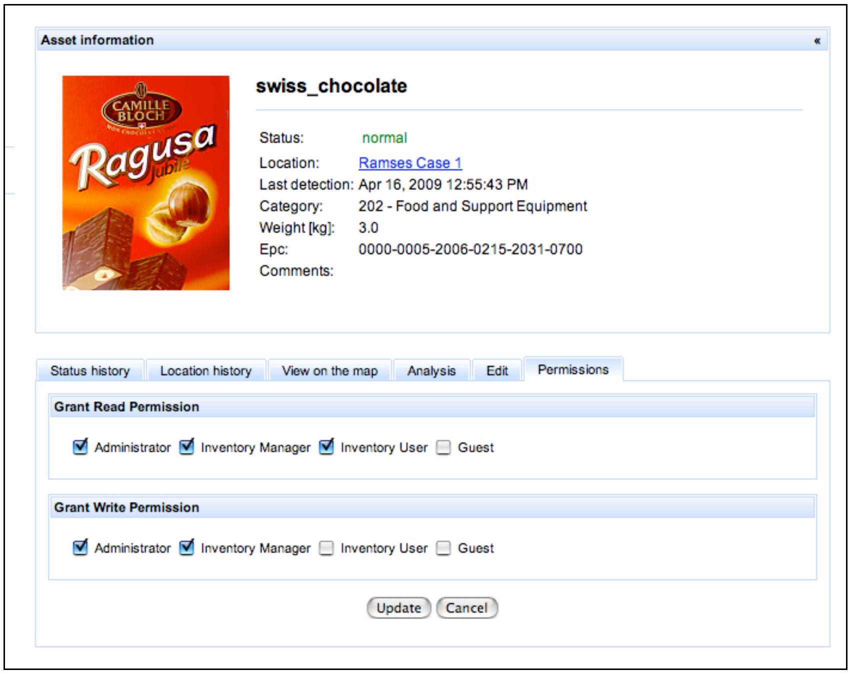


Figure 24. Asset Permissions

3.3.2.3 Inventory and Item History Analysis

This version of the RAILS software provide the ability to assign and extract various properties from a given asset or collection of assets within a container. One key change from the Phase I prototype was the transition from free-text keyword based asset categories to a more structured form based on MIT's SpaceNet Classes of Supply (SpaceNet 1.3). Figure 25 illustrates a subset of these based on the edit menu pick for a particular asset. These codes can be then used in the rule sets to group items according to their class or invoke actions based on the numbers or mix of assets within a given geographic area. One thing to note is the current implementation's use of a numeric structure of these supply codes. In the current version, these codes are simple numeric values with implied structure. This causes an ambiguity with some of the rule matching software and later versions of the system will utilize a more formal notation, possibly using delimiters such as "." to denote the structure of the supply codes (such as 2.4 for Hygiene items rather than 204).

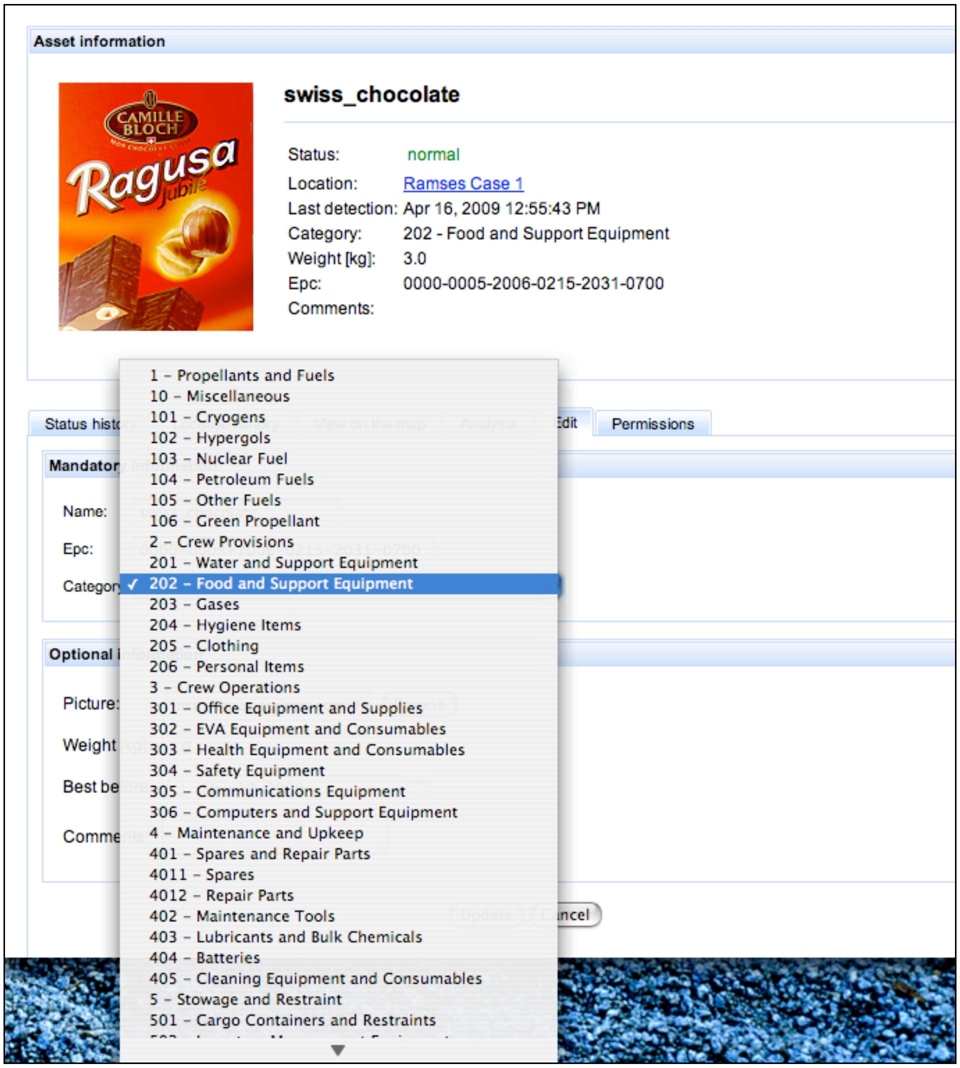


Figure 25. Classes of Supply

In addition to asset classes of supply, users's can assign mass values to a given asset. The system can then use these to compute the total mass of a container and its contents (see Figure 26). Both these individual and aggregate mass values are then accessible to the rule engine, allowing the development of triggering rules based on mass properties in combination with other properties, such as container location, classes of supply, or other properties.

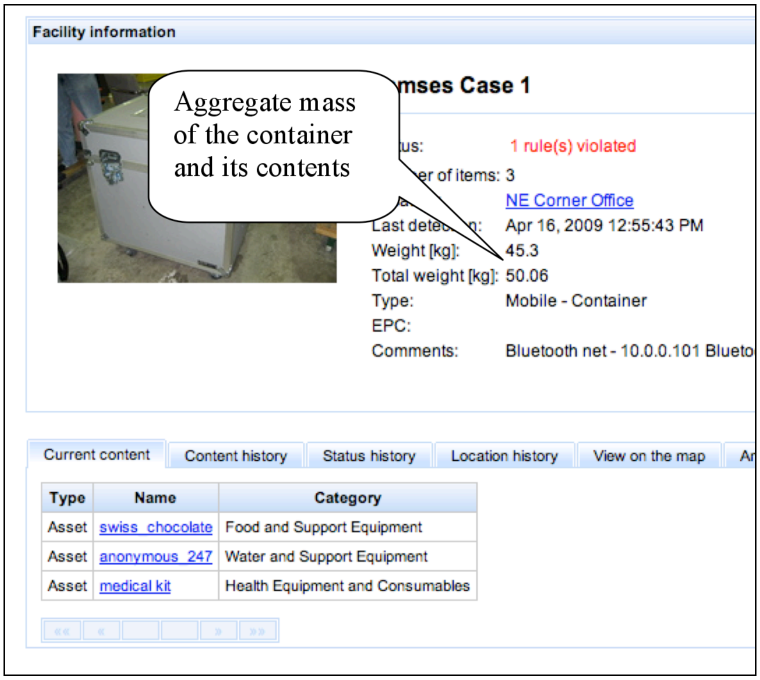


Figure 26. Aggregate Container Mass

Utilizing the asset classes of supply (categories) and known locations of assets and the containers holding those assets, the Analysis menu provides a bar graph indicating the distribution of various categories across all facilities within the system. These bar graphs are non-exclusive, in that they show all facilities, including those nested within one another.

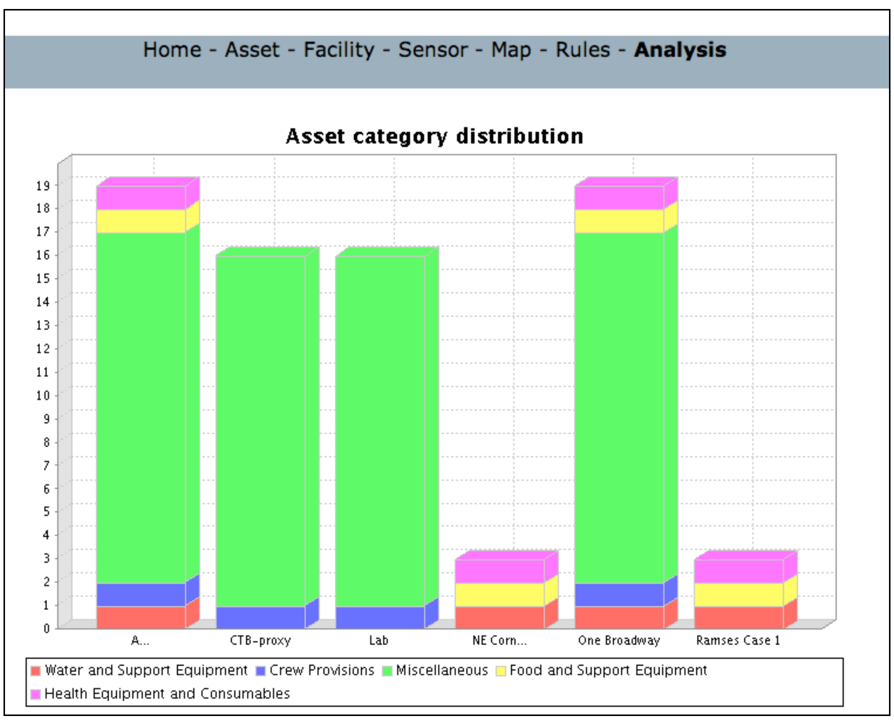
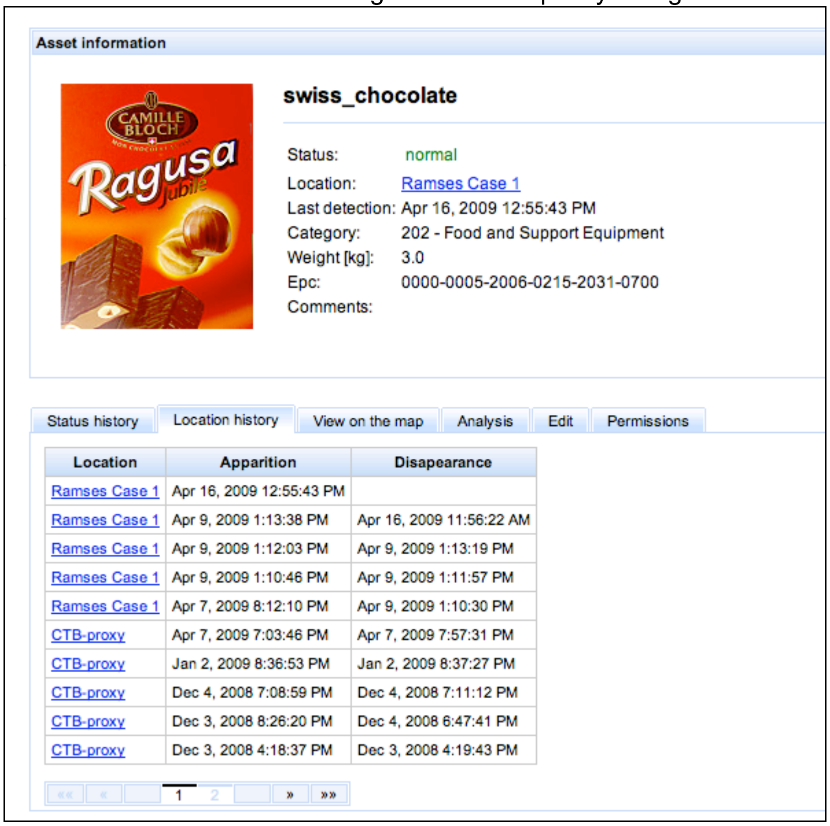


Figure 27. Asset classes of supply (category) distribution by location

Knowledge of when a given asset was at a particular time (i.e. construction of the location history of an item) can be an important piece of information when considering the environmental impacts on the quality and/or lifetime of a given asset. For example, if perishable food spent a significant amount of time in an container lacking temperature controls on a hot tarmac or warehouse, the lifetime of that food may have to be shortened accordingly. For this reason, RailsV2 maintains a location history for each asset in its inventory, as shown in Figure 28. This history can be then used by the rule sets to derive properties or trigger alerts if a series of conditions arise that might affect the quality of a given asset.



Location	Apparition	Disapearance
Ramses Case 1	Apr 16, 2009 12:55:43 PM	
Ramses Case 1	Apr 9, 2009 1:13:38 PM	Apr 16, 2009 11:56:22 AM
Ramses Case 1	Apr 9, 2009 1:12:03 PM	Apr 9, 2009 1:13:19 PM
Ramses Case 1	Apr 9, 2009 1:10:46 PM	Apr 9, 2009 1:11:57 PM
Ramses Case 1	Apr 7, 2009 8:12:10 PM	Apr 9, 2009 1:10:30 PM
CTB-proxy	Apr 7, 2009 7:03:46 PM	Apr 7, 2009 7:57:31 PM
CTB-proxy	Jan 2, 2009 8:36:53 PM	Jan 2, 2009 8:37:27 PM
CTB-proxy	Dec 4, 2008 7:08:59 PM	Dec 4, 2008 7:11:12 PM
CTB-proxy	Dec 3, 2008 8:26:20 PM	Dec 4, 2008 6:47:41 PM
CTB-proxy	Dec 3, 2008 4:18:37 PM	Dec 3, 2008 4:19:43 PM

Figure 28. Asset Location History

3.3.2.4 Expanded Analytic Rule Set

During Phase I, the team developed a set of simple rules using RDF and CWM. During the first part of Phase II, the team continued development using this approach. After the switch to JBoss Rules, known as Drools, the team re-implemented the existing rules sets using the Drools core language and began implementation of the remaining rules targeted to the Phase II effort. Phase I, the initial plan was to continue that development, however the switch to JBoss Rules Incompatibility Rule. The following illustrate the rule sets outlined in the Phase II proposal.

Incompatibility Range Rule

Figure 29 illustrates the implementation of the "Incompatibility Range Rule" implemented using Drools and a domain language the team developed for the RAILS application (from within Drools). This rule an extension of the Phase I "Incompatibility Rule" that triggers when two incompatible items are within the same container. In this case, it triggers when two incompatible items are within a given range of one another, based on the location of their respective containers. Figure 30 illustrates the effect of the trigger on the Status History of a given inventory item.

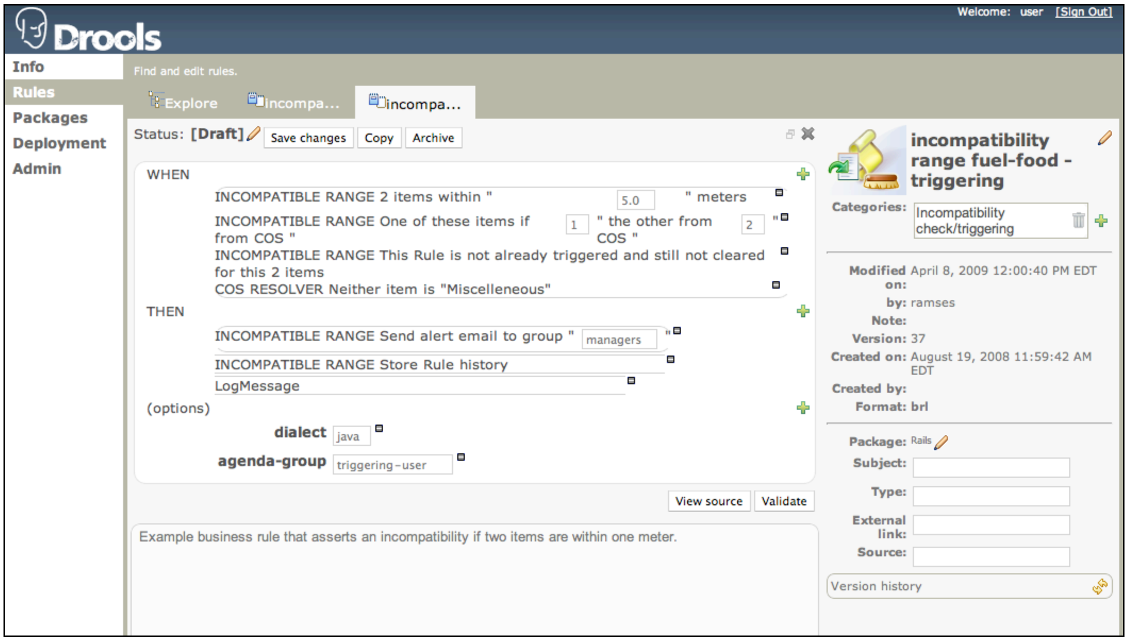


Figure 29. Incompatible Range Triggering Rule

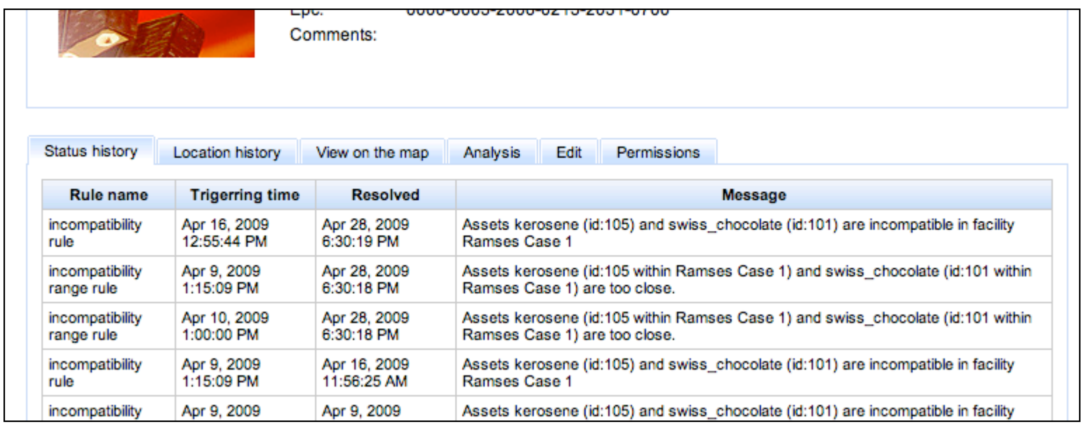


Figure 30. Incompatible Range within the Status History

Critical Supply Level

During Phase I, the team had implemented a missing item rule, that triggered if a critical item wasn't seen by one of the Smart Containers in the system. During Phase II, the team extended this type of rule to trigger when a certain quantity of a particular class of supply went below a given threshold. The effects of this rule trigger are shown in Figure 31.

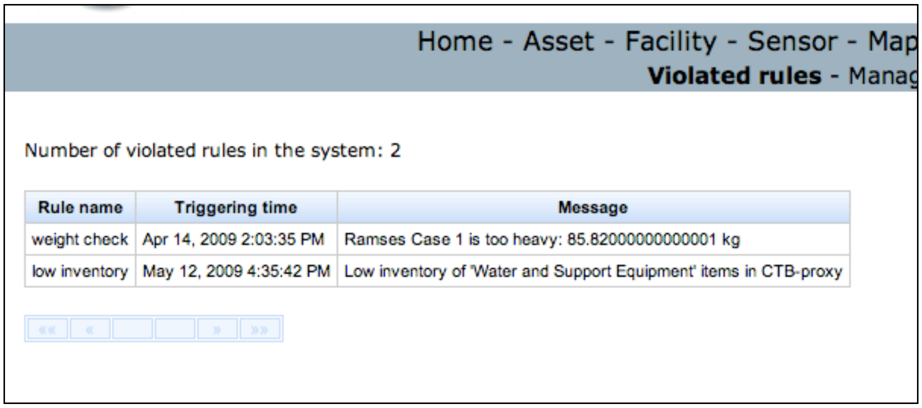


Figure 31. Critical Supply Level (i.e. Low Inventory) Status History

Expiration Date

For Phase II, the team implemented an Expiration Date trigger rule. Before implementing this rule, the team augmented the information associated within assets to include a "Best Before" date, as shown in Figure 32. When an item appears in a tracked container or area, the rule checks the current date against this "Best Before" date and triggers a rule if the asset is too old, as shown in Figure 33. The rule is cleared once the item disappears, but will reassert if the asset ever reappears in the system.

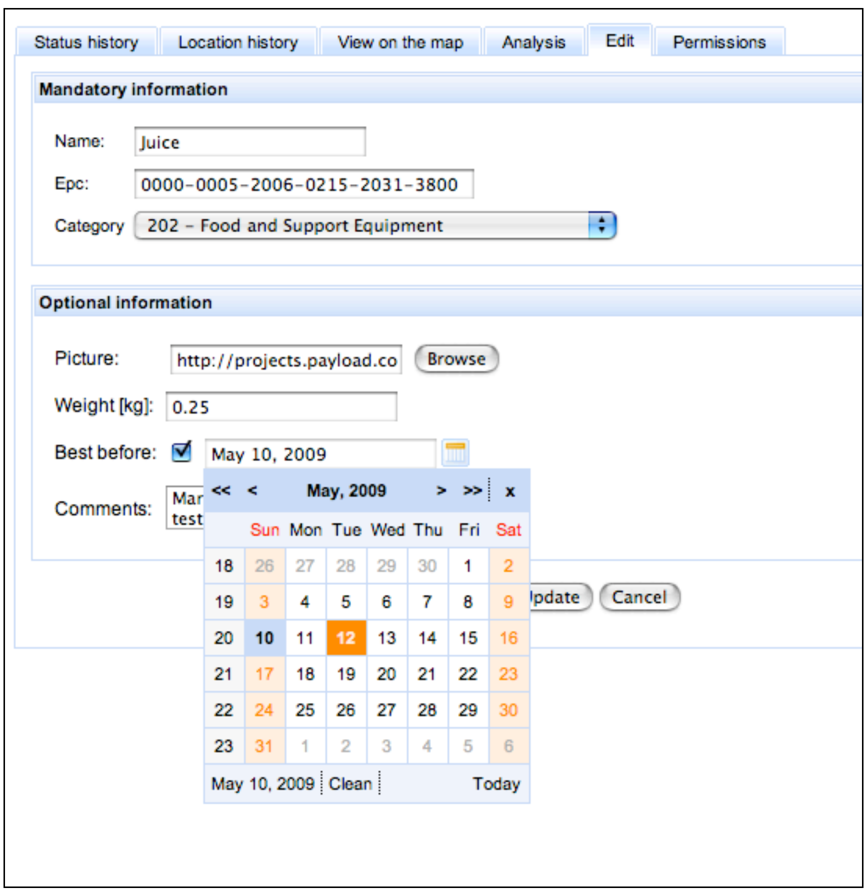


Figure 32. Configuring an asset's expiration date

Figure 33. Item Expiration Status History

Temperature

One of the objectives for Phase II was to integrate sensor measurements with the inventory management system. To accomplish this, the team created a proxy temperature sensor. The temperature sensor posts its readings through a URL to the device server running within RailsV2 (similar to the inventory, GPS and WiFi-location reports for the Smart Containers. When a temperature reading arrives or an asset becomes listed in a container, the system invokes the temperature log entry rule, shown in Figure 34. This rule creates a status history event for the temperature reading, shown in Figure 35.

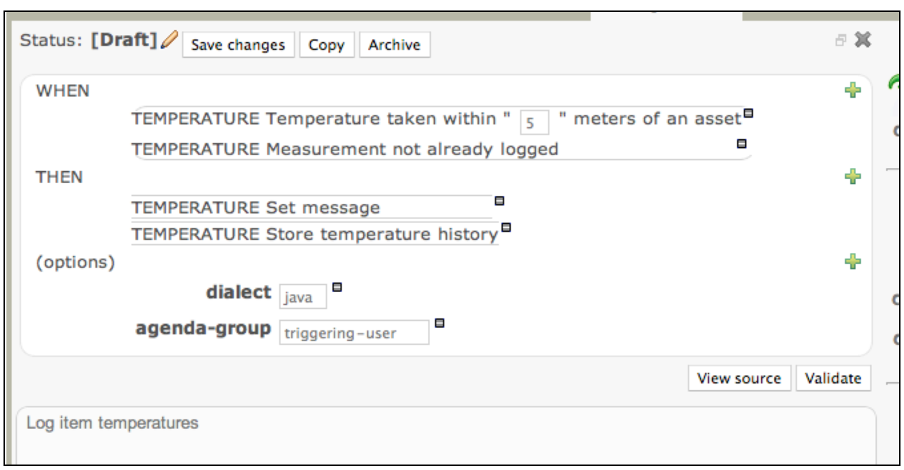


Figure 34. Temperature Logging Rule

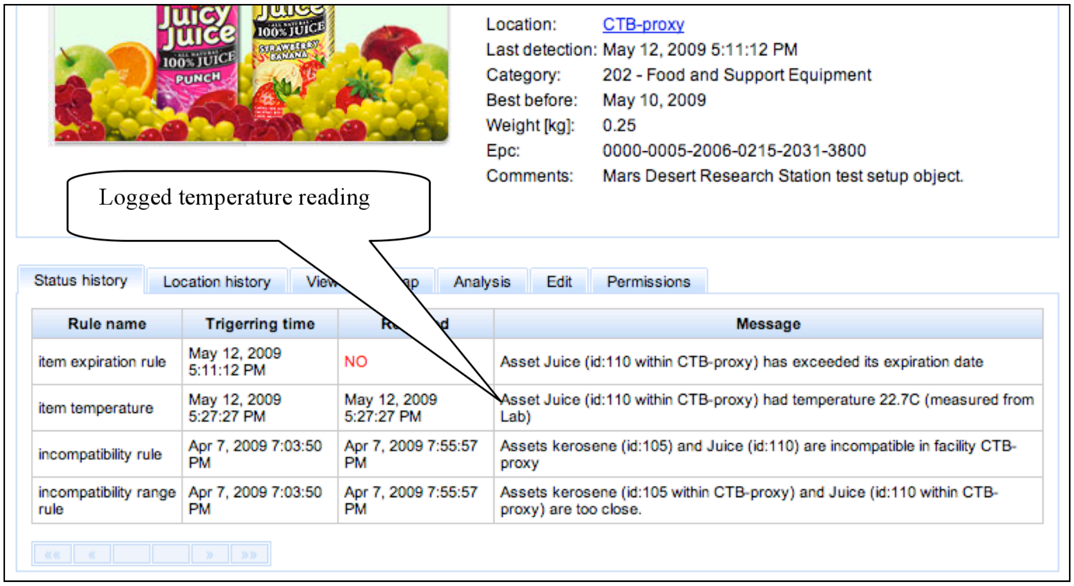


Figure 35. Temperature event

3.3.2.5 External Interface Layer

The Phase I RAILS prototype had limited ability to import and export information. During Phase II, the team implemented an XML-based data export and import capability to help support external, 3rd party inventory management systems. The XML information provides the framework to write translators, using translation template languages such as XSLT, and bridges between the RAILS information representation and other information systems.

Figure 36 illustrates the main menu for information import and export. Access to these menus is based on the current user role; the "Inventory User", "Inventory Manager" and "Administrator" roles are all able to export inventory information, however, only "Administrators" and "Inventory Managers" are allowed to import information.

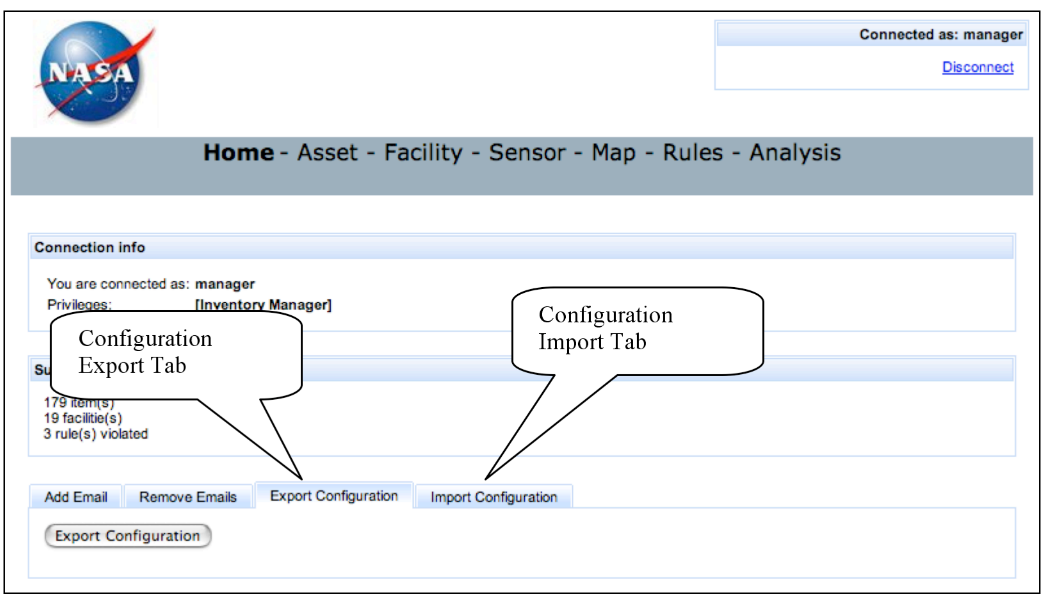


Figure 36. Configuration Export and Import Tabs

Figure 37 illustrates a portion of the XML file generated by the "Export Configuration" command. Using the exported XML file, integrators can write translators that convert the tagged elements into other forms, such as a simple comma separated table, illustrated in Figure 38.


```xml
<edu.mit.ramses.external.AssetLocationExport>
  <asset>
    <id>110</id>
    <objectId>110</objectId>
    <name>Juice</name>
    <picture>http://projects.payload.com/ramsesproto/0.1/images/JuiceBox.jpg</picture>
    <weight>0.25</weight>
    <assetCategory>
      <name>Food and Support Equipment</name>
      <cosCode>202</cosCode>
      <id>61</id>
    </assetCategory>
    <comments>Mars Desert Research Station test setup object.</comments>
    <bestBefore>1241928000000</bestBefore>
  </asset>
  <epc>
    <epc>0000-0005-2006-0215-2031-3800</epc>
  </epc>
  <location>
    <point>false</point>
    <noGpsCoordinates>false</noGpsCoordinates>
    <surfaceNorthWestGpsLatitude>42.36305082311405</surfaceNorthWestGpsLatitude>
    <surfaceNorthWestGpsLongitude>-71.08369141817093</surfaceNorthWestGpsLongitude>
    <surfaceNorthEastGpsLatitude>42.36298145806284</surfaceNorthEastGpsLatitude>
    <surfaceNorthEastGpsLongitude>-71.08352780342102</surfaceNorthEastGpsLongitude>
    <surfaceSouthWestGpsLatitude>42.362995331079226</surfaceSouthWestGpsLatitude>
    <surfaceSouthWestGpsLongitude>-71.08373433351517</surfaceSouthWestGpsLongitude>
    <surfaceSouthEastGpsLatitude>42.36292398410529</surfaceSouthEastGpsLatitude>
    <surfaceSouthEastGpsLongitude>-71.08357608318329</surfaceSouthEastGpsLongitude>
    <pointGpsLontitude>0.0</pointGpsLontitude>
    <pointGpsLatitude>0.0</pointGpsLatitude>
    <locationName>CTB-proxy</locationName>
    <locationId>299</locationId>
  </location>
</edu.mit.ramses.external.AssetLocationExport>
```


Figure 37. Sample exported asset information

101	swiss_chocolate	0000-0005-2006-0215-2031-0700	Ramses Case 1	42.36296560318327 -71.0834527015686
102	space ice cream	0000-0005-2006-0215-2031-9001		0.0 0.0
103	bananas	0000-0005-2006-0215-2031-9002		0.0 0.0
104	water	0000-0005-2006-0215-2031-0400	Ramses Case 1	42.36296560318327 -71.0834527015686
105	kerosene	0000-0005-2006-0215-2031-1000	Ramses Case 1	42.36296560318327 -71.0834527015686
106	space suite	0000-0005-2006-0215-2031-1006		0.0 0.0
107	medical kit	0000-0005-2006-0215-2031-0600	Ramses Case 1	42.36296560318327 -71.0834527015686
108	laptop	0000-0005-2006-0215-2031-1008		0.0 0.0
109	Tissue	0000-0005-2006-0215-2031-1500		0.0 0.0
110	Juice	0000-0005-2006-0215-2031-3800	CTB-proxy	42.36298145806284 -71.08352780342102
111	Hershey's Milk Chocolate Bar	0000-0005-2006-0215-2031-3700		0.0 0.0
112	4 pack of AA Batteries	0000-0005-2006-0215-2031-2500		0.0 0.0
113	2 pack of AA batteries	0000-0005-2006-0215-2031-2200		0.0 0.0
114	Garbage Bag	0000-0005-2006-0215-2031-2700		0.0 0.0
115	Toilet Paper Roll	0000-0005-2006-0215-2031-3000		0.0 0.0

Figure 38. Example table converted from inventory export

3.4 Task 4: RAMSES Testing

3.4.1 Work Performed

The purpose of the Phase II RAMSES Testing was to establish a degree of systems level testing to the RAMSES architecture, particularly outside the confines of the RAMSES lab. This task was broken into a series of key subtasks, as described in the original proposal:

4.1 **Smart Container Testing**. Each of the three (3) Phase 2 smart containers will be subjected to individual quality control and usability testing. The tests will verify and validate the requirements developed in phase (1). In addition we will perform multi-container testing to ensure that multiple containers in close vicinity do not interfere with each other and that room-level wireless (802.11) bandwidth is sufficient to handle multiple containers. We will repeat similar experiments as those performed by MIT in Phase 1, but potentially with fewer users. We will try to find the boundaries of system performance, especially in terms of the number of items handled by a single smart container.

4.2 **Room-Level and Facility-Level Testing**. We will expand the room-level 802.11 wireless access point tracking demonstrated in Phase 1. As an ideal test site we propose to use the MIT campus, which is completely wireless and has one of the highest densities of 802.11 access points of any facility in the world. We will track the movement of supply items and containers across a multi-day campaign on campus and electronically document the results.

4.3 **Field Testing**. Once the RAMSES system has been matured, the three 2nd generation smart containers exist and the RAILS system has been upgraded and is stable we propose to conduct a test campaign out in the field. How exactly and where exactly this will be done is still open and will depend on technical objectives, budget, system availability and logistics. The exact format of the field test campaign will be decided together with the COTR. Some examples of potentially meaningful field test campaigns are:

- RAMSES testing at a NASA Center, potentially at Stennis for a period of up to 4 weeks.
- Testing in the local area, but beyond the MIT campus. this would be an expansion of the proposed MIT campus test campaign, and demonstrate some of the outdoor tracking technologies discussed earlier in this report
- Testing at a planetary analog site on Earth. We would propose to select a site with easy access and travel logistics, such as D-RATS. D-RATS is a yearly field test campaign conducted by NASA in the desert Southwest. It is designed to test new exploration equipment in a Moon- or Mars-analogue environment. In any case any field test campaign will be restricted to CONUS.

During Phase II, the RAMSES Testing activity proceeded in a slightly different order than originally intended. Originally, the test effort was to start with single container testing and then branch out to higher level tests, such as inter-container testing and container tracking. Due to the Wired NextFest opportunity in the fall of 2007, however, the team had the opportunity to perform some fielding tests with the cases while at the event. After the NextFest event, in the winter/spring of 2008, students from MIT participated in the Mars Desert Research Station (MDRS) in the Utah desert, providing an additional opportunity to test the RAMSES system by users in a Mars-exploration analog environment, with limit communications and support resources. Following MRDS, a separate group of MIT students examined integration and test of RFID systems into a Cargo Transfer Bag (CTB), using off-the-shelf components for the High-Frequency (HF) RFID tests, and the RAMSES electronics for the Ultra-High Frequency (UHF) tests. The final set of testing took place with a retrofitted double-height CTB, measuring the usability of the RAILS software and accuracy of the CTB inventory reads. The following describes the activities and results from these various tests.

<u>Field Testing: Wired NextFest</u>

Prior to the NextFest trip, the team recognized the uncertainty of utilizing WiFi-based devices at the event, primarily through lack of communication from the event's organizers about the WiFi environment. As a result, we ensured that the containers could communicate through WiFi, but also could present information to a local computer through its Bluetooth connection. We also tested routing network traffic from the Bluetooth through the local computer's Ethernet connection back to Payload Systems. Once we arrived at the event, we also determined that demonstrating container locations purely by which access point had the strongest signal wouldn't work due to the proximity of the zone's we'd be moving the container through.

The tracking via access-points problem led us to make a field-modification to the WiFi tracking script. Rather than locating which access point had the strongest signal, we modified the script to use reference zones. Each reference zone would consist of a list of visible access points and the power measured from each within that zone. The algorithm determined which zone the container was in was based on a closest match of access point power measurements against a given zone's reference list. Within the confines of a hotel room, the algorithm managed to discriminate between once side of the room and another, albeit with some errors due to access point signal dropouts.

Once the team arrived at the event, however, the WiFi environment (and network environment in general) was less reliable that anticipated. We encountered significant difficulty with the WiFi connections, both due to miss-configuration of access points by other vendors at the event and due to channel collisions between large numbers of vendors at the event. Additionally, the WiFi environment had a significant impact on the local performance of the readers, causing significant delays in reading tags from the containers. Ultimately, we determined the WiFi tracking demo was impractical for the event and we removed the WiFi sticks from the containers for the remainder of the event.

In addition to the WiFi issues, the team encountered significant performance delays (5 - 10 seconds for a read/display cycle and sometimes up to 30 seconds to display updates from the Payload Systems server). Some of these delays are due to the SkyeTek reader, which imposes a 5 second timeout when it fails to see tags from a particular antenna. Other delays were due to the nature of the RAILS server and its rule-engine. The RAILS system uses Resource Description Framework (RDF) and Tim Berner Lee's Closed World Machine (CWM) rule engine to display information from the server. The RAILS architecture as it stands, although extremely flexible, does not scale well to large numbers of items nor to near real-time responsiveness.

Due to the WiFi problems encountered at NextFest, the team added a Cellular Network card to a support laptop for the Disney trip. If needed, the laptop could be configured to route traffic from its Bluetooth connection to the containers through the card and back to Payload Systems. Despite this precaution, an issue (which was later identified as a broken lid switch wire) arose at Disney where the container failed to update its contents, even. In the absence of a working local container, the team was able to demonstrate the system remotely using the 2nd container, shipped back from NextFest to Payload Systems.

As a result of both trips, the team examined the main encountered issues; connectivity and performance. On the connectivity front, the team began to identify Cellular Network devices that can be integrated with the Gumstix electronics. One option was a USB-based device, while another was the Goliath-vx GRPS/EDGE + GPS card being developed by Gumstix (note: GUmstix never released a final version of this card and abandoned the project). On the performance front, the team identified the bottlenecks in the system and evaluated alternative, higher performance architectures to support near-real time responsiveness in the presence of large numbers of inventory items. The bottleneck analysis examined the relative times between each stage of the system and possible optimization of the amount of data transmitted between devices and the server, and between the web servers and client browsers. The rule-engine and server performance analysis examined the scalability of the server architecture (i.e. how will the performance degrade as the number of inventory and other items grow) and rule-engine performance.

Field Testing: Mars Desert Research Station (MDRS)

In addition to the field work at Wired NextFest, MIT students participating in the Mars exploration analog site, the Mars Desert Research Station (MDRS), undertook their own construction of a RAMSES compatible Smart Container, show at MDRS in the center of Figure 39. The students designed and constructed their own rigid container, designed to fit the standard envelope for airline baggage, and had intended to purchase and build their own electronics for the container, based on the RAMSES design. Unfortunately, due to time constraints and electronics delivery schedules, the students were not able to complete this second step, and instead, borrowed one of the electronics from one of the RAMSES Smart Containers. With their own container and RAMSES electronics, the students undertook two mission phases at MDRS. The following is a field report from their testing effort:

Figure 39. RAMSES Smart Hexagonal Container #3 at MRDS

Phillip Cunio and the rest of the Expedition Epsilon crew were stationed at MDRS for the two weeks immediately following the successful completion of Expedition Delta. Phillip's research objectives included performing quantitative testing on the various factors related to the read rate and accuracy of the electronics system in the SSLC as well as testing mobility issues involved in using the SSLC on EVA.

Both Phillip's becoming ill and technical issues with the test set-up initially hampered the testing of the RFID system. While we are pleased to report that Phillip was able to regain his health during the mission, the same cannot be said about the resolution of the technical issues. A Bluetooth network could be created without too much difficulty, allowing the computer to communicate with the Gumstix in the container, but unfortunately an automated, continuous bridge between the container and the Payload server could never be established. This was compounded by technical issues at the Payload server site caused by storms in the Cambridge area, but we are very grateful to Jim Francis and Joe Zapetis for fixing the server in amazingly short order. As a work-around to the automated update process, we were able to take the URLs from the sendtaglist.log file off of the Gumstix and input these into Firefox to create a manual update method.

Despite the network issues, Phillip collected data on site using the Gumstix log files on various factors related to the accuracy of the RFID system. During initial discussion and planning, it was decided that several key variables would be tested. These variables are listed in Table 2. The data from these tests was collected in Excel files and sent back to MIT for analysis.

Table 2. Variable tested during Expedition Epsilon

Variables Tested	Values Tested
Type of RFID Tags	Gen-2, Omni-directional
Number of Antennas	1, 2, 3
Number of Items in SSLC	5, 10, 20

The other focus of the research completed by Phillip was on the mobility issues raised with taking a SSLC out on EVA. These efforts involved brief walkarounds of EVA crew while carrying the SSLC. For the first test, three crew cycled through the airlock with the unloaded SSLC and then two of these crew carried the SSLC by hand a distance of approximately 100 meters, and then carried the SSLC back into the airlock. With proof of ability thus obtained, the SSLC was loaded with about 14 person-days of freeze-dried food and the test was repeated, although this time two separate 100-meter routes were traversed a total of three full times. Notably, the container and personnel bore up well, with no major damage or injuries.

The primary complaint from crew members involved the difficulty of grasping the small handles and holding them while wearing heavy EVA gloves. Initial conclusions drawn from this research effort are that the handles should be made bigger (or that allowances should be made to carry the SSLC in a way which will reduce hand fatigue due to the need to grip the handles), but that the basic design is amenable to crewed transport on the surface of planetary bodies over short distances.

A later analysis of the bandwidth challenges encountered at MDRS indicated that the issue wasn't the transmission of inventory information to the RailsV2 server, but instead the frequent polling of the web browsers to the server for updates to images and inventory tables generated by the server. This realization leads to potential solutions for bandwidth-limited environments.

Initial CTB Testing

MIT's Experimental Projects Laboratory Course 16.62 project did the initial integration of the RAMSES RFID concept with some Cargo Transfer Bags (CTBs) loaned to MIT by Kennedy Space Center (KSC). Jonathan Banahan and Paul Estrada, working with readers borrowed from MIT's Auto-Id lab, investiaged the use of High-Frequency (HF) RFID tags with a single-height Cargo Transfer Bag and with the RAMSES electronics for an investigation of the Ultra-High Frequency (UHF) RFID tags in a double-height Cargo Transfer Bag. Due to time constraints, the students were able to complete their HF study, but marginally completed the UHF study. Their main contribution, however, to the UHF system was their design of the shielded foam inserts for the CTB, which the team later adapted to the CTB-proxy design and CTB retrofit kit prototype. The following is a report from the students on the HF testing:

For this setup, the team of Banahan and Estrada instrumented a standard-height cargo transfer bag with a single, High-Frequency (HF) RFID antenna and shielded the inside of the bag with fine copper mesh. The CTB is on loan from KSC and the HF electronics and software were loaned from MIT Auto Id laboratory.

For the evaluation, the team used three material types: non-metal, metal and liquids; and three orientation and position types: best case, worst case and random. The three material types reflect different sensitivities with RFID equipment; metal and liquids tend to shield or interfere with RFID systems. The three orientation types were chosen to determine the performance limits based on how items are packed into the CTB. The best and worst positions and orientations were determined experimentally by varying the orientation of each type of item with respect to the single HF antenna and measuring the number of successful reads of item's tag. Generally, the worst position and orientation was furthest from the antenna (near the top of the bag) with the tag perpendicular to the antenna.

Figure 40 illustrates the experiment test matrix. The rows indicate the number of items packed into the CTB for each test. The columns indicate the item material type selection and position/orientation class. In order to mitigate systematic errors in the system, the team used multiple runs and averaged the results from those runs.

Number of Items	Liquids			Nonmetals			Metals			Mixed		
	BC	WC	RD	BC	WC	RD	BC	WC	RD	BC	WC	RD
6	6	0	4.8	6	0	6	6	0	2.8	6	0	4.4
12	12	2.8	9.5	12	0	11.3	8.8	0	3.8	11.6	0	10
18	17.4	2.6	14.5	18	0	16.8	7.2	0	3.9	16.4	0	12.1
24	17.6	3.2	18.1	14.2	0	20.2	7.2	0	3.5	17.6	0.4	12.9
30	22.6	7.4	19.7	26.4	0	24.7	5.6	0	3.4	19.6	0	15.4

BC-Best Case
WC-Worst Case
RD-Random Case

Figure 40. CTB HF Test Matrix

Figure 41 and Figure 42 illustrate the results from these tests. For the worst-case position/orientation, only non-metallic materials were readable and significant interference from the metal and liquid items interfered with non-metal item reads in the mixed case.. With the best orientation and position, liquids and non-metallic items could be read over 90% of the time up to 18 items, rolling off after 18 items. Metal items dropped significantly between 6 and 12 items. With a random position and orientation, non-metallic items had consistently good read rates (above 90%) up to 18 items, with liquids and metals with rates below 80%.

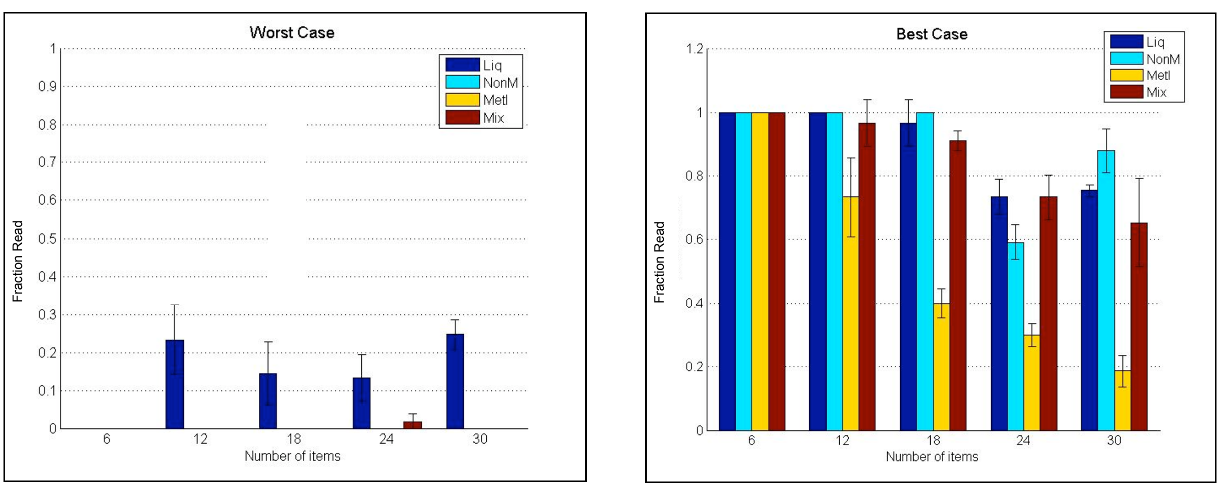


Figure 41. Results of Worst and Best Case Orientations

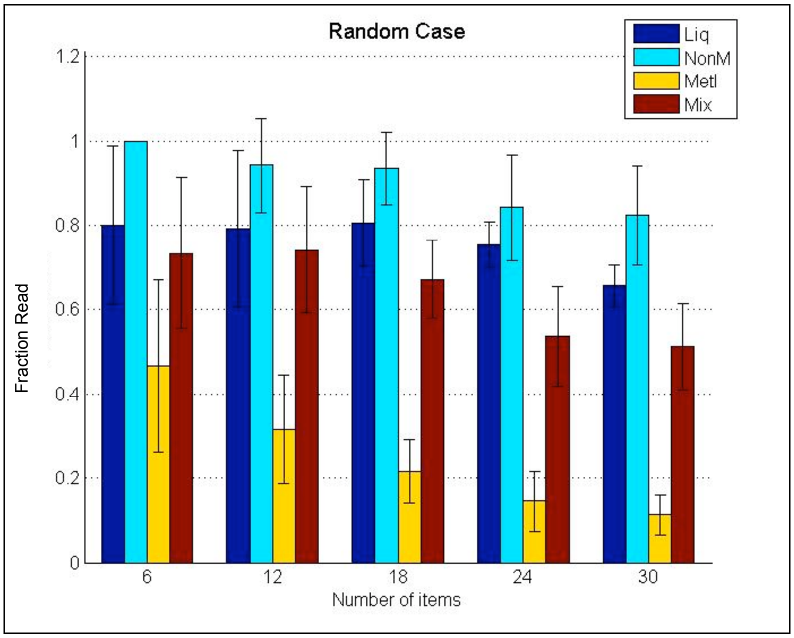


Figure 42. Results of Random Orientations

Earlier tests during the Phase I effort, and informal experience with the first and second smart containers, indicate significantly better performance than these results suggest. We believe there are at least two main differences in the setups that account for the divergent results and experience:

1. **Number and orientation of antennas**. Prior to this setup, all the smart containers utilized three antennas, each aligned along a given axis within the box. This minimized situations where RFID tags were perpendicular to all three antennas.
2. **Different RFID technology.** This experiment utilized HF RFID readers whereas the earlier smart containers utilized UHF RFID Readers. HF readers are designed for shorter range than UHF, typically centimeters versus about a meter. The worst case position for the HF readers were near the top of the CTB - near the far end of the reader range. The dimensions of the smart containers with the UHF readers, particularly with the three antennas, were well within the designed range limits of the installed UHF systems.

One Broadway Testing

After the acquisition of Payload Systems Inc. by Aurora Flight Sciences, the team moved its offices and laboratories from the old Payload Systems offices at 247 Third St in Cambridge down the street to offices within the Cambridge Innovation Center's One Broadway office tower. After the move, the team conducted three types of tests in and around the One Broadway facility. These included adaptation of the WiFi-based location system using the new wireless environment, GPS testing inside and within the area around the building, and cell-phone routed data transfers from the Smart Container to the RailsV2 servers.

During both Phase I and during the early portion of Phase II, the team performed qualitative testing of room-level tracking of the Smart Container locations through a simple selection algorithm based on which WiFi access point had the strongest signal and selecting the current location zone accordingly. While effective within the limited confines of the 247 Third St. facility, this approach didn't work when the access point signals saturated the power measurement provided by the WiFi device, such as those we encountered at the Wired NextFest. As reported earlier, we modified the location system to use the aggregate power of a set of reference access points to select the current location. After the move to One Broadway, the team calibrated and tested the ability to locate the Smart Container within the One Broadway facility. The results were mixed. Generally, the approach worked, however the team discovered that the power readings the WiFi device reported for the access points would fluctuate based on the beacon timing of the given access point. This led to spurious jumps between locations, and occasionally persistent misidentification of a location if an access point was moved or material within the building moved to change the power-level calibration of the location reference points.

After the move to One Broadway, the team performed two qualitative tests of cell-phone based networking with the Smart Containers. In the first of these tests, the team attached a Verizon network adapter to the Tablet PC, installed the network provider's software, and configured the Tablet PC to route its Bluetooth network traffic from the Smart Container through to the Verizon network. This approach successfully allowed the Smart Container to send updates to its inventory to the main RailsV2 server while out of range of one of its configured WiFi access points. The 2nd test repeated this exercise using a personal Bluetooth-enabled cell phone. Here, one of the team members signed up for data access through their personal cell phones, installed software on the laptop and routed Smart Container inventory updates from the container, through the laptop, to the phone and out to the RailsV2 server through the provider's network.

In order to track Smart Containers beyond the confines of instrumented rooms and calibrated WiFi regions, the RAMSES team modified the RailsV2 software to accept specific Ground Position System coordinates. Using this software change, the team integrated a GPS USB dongle onto the Tablet PC and wrote some software that captured GPS measurements and routed the reported positions back to the RailsV2 server. The team tested this approach with the Smart Container within the One Broadway building and in the areas around Kendall Square in Cambridge, MA. The building at One Broadway was originally designed as secure building and has some RF shielding within the building structure. As a result, the team failed to obtain a GPS signal anywhere within the tested areas of the building (including adjacent to large windows within the building). The team then transitioned to testing outside. The area within Kendall square has a mixture of building heights, however, most are above a few stories tall. As a result, it was difficult for the GPS receiver to get enough satellite signals while close to some of the building areas. Eventually, the team moved to more open areas within Kendal square and was able to get a good GPS location and track the container location within those open areas (to within tens of meters).

Retrofit Kit Assembly Tests

Late in the Phase II effort, MIT's cost-benefit analysis demonstrated the need for a CTB retrofit kit supported the instrumentation of CTBs already on orbit. As a result of this analysis, the team developed a set of retrofit components and procedure to enable the construction of such an instrumented CTB. MIT then took the components and procedure and tested the construction and installation of those components within the double-height CTB on loan from KSC. This test was successful, but did result in some change recommendations for the retrofit components and procedure. These change recommendations can then be implemented as part of a Phase III effort.

Performance Testing

During the winter of 2008/2009, MIT conducted performance tests of the RAMSES system using the CTB instrumented using the CTB retrofit kit. The test campaign for the RAMSES prototype included tests for interference caused by the material of the cargo items (metal, liquid, and paper/plastic), the orientation of the passive Gen-II RFID tags affixed to the items, the mechanism by which the tags were attached to the cargo items, and the number of tags packed in the prototype at one time. Figure 43 provides a sample of the results obtained, showing the mean percentage of RFID tags successfully detected by the RAMSES system for various test scenarios[2]. Several consequential trends were observed:

- System performance degraded as the number of tags in the CTB increased.

[2] Each test scenario included 5 independent tests for a given combination of cargo material, number of tags, orientation of tags, and tag mounting mechanism.

- Cargo items made of paper/plastic were read more reliably than those including metals and/or liquids.
- Metal and liquid objects proved very hard to read unless the tags were offset from the item by some mechanism (foam spacers worked best, loose plastic bags also showed some impact).
- Scenarios in which items (and thus tags) randomly oriented with respect to the RFID antennas mounted in the walls of the CTB, generally yielded the highest read accuracy for a given set of variables.

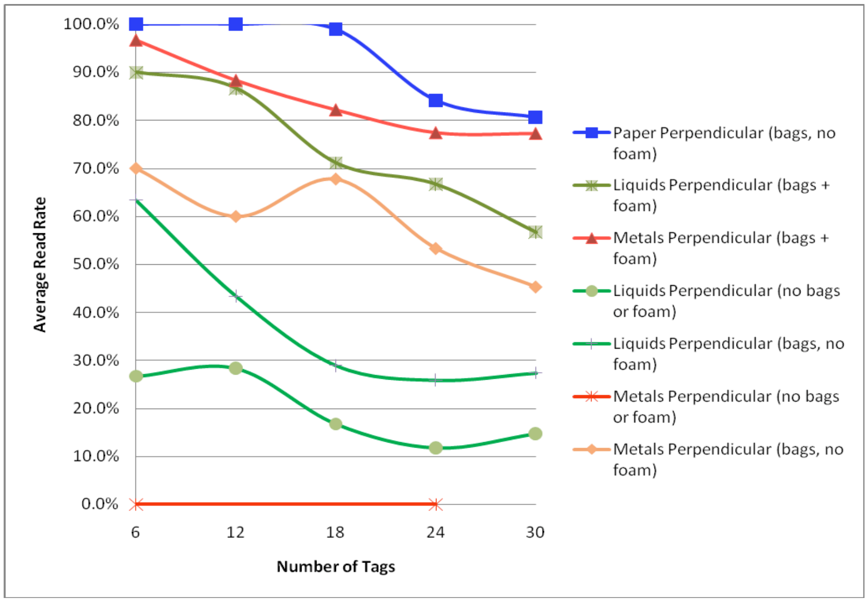


Figure 43. Mean % of RFID tags read in a particular test scenario.

3.4.2 Results

The following summarizes the key results from the testing effort:

- General building area tracking utilizing WiFi access point visibility is effective, although does not provide the granularity to perform room level tracking
- Room level tracking based on WiFi access point signal strength, although effective under controlled conditions, can be intermittent under real-world conditions, with 3rd party access points and moving materials and personnel affecting signal strength. Alternative solutions include room instrumentation utilizing localized beacons or RFID readers and active or passive tags, time-delay based triangulation utilizing specialized access points, and low cost acoustic beacon solutions
- GPS tracking is effective but requires effective line-of-sight, unshielded reception from GPS satellites
- The Gen-2 tags used during most of Phase II continue to have significant problems with water and metals. Qualitative tests with metal/water tolerant tags, however, provide optimism that the technology is advancing to address these issues
- The use of RF-absorbing shielding was ineffective as an alternative to the use of conductive shielding with spacing material to prevent missed reads due to Gen-2 tags being placed too close the metal shielding

- For normal use, the RailsV2 web presentation is effective. However, for bandwidth limited clients, such as those found at the Mars Desert Research Station, continued display updates saturated the available bandwidth. For these situations and for mobile web browsers, such as cell-phone browsers, RailsV2 should provide optimized views that limit the amount of information to be sent from the server tot he browser

During the testing effort, the team maintained a software defect/enhancement request system. During Phase II, a total of 37 software defects were reported and 15 enhancements were requested. During the course of Phase II, the team fixed 25 of those defects (see Figure 44 and Figure 45) and implemented 4 of the requested enhancements (see Figure 46 and Figure 47). At the conclusion of the Phase II effort, 12 software defects remain, although work-around exist for all, and 11 enhancement requires remain unfulfilled. The team will incorporate these reports and recommendations into a Phase III effort.

Status: closed (25 matches)

Ticket	Summary
~~#9~~	Removal of items didn't clear triggers
~~#10~~	Drools fails to remove a snapshot
~~#17~~	Critical item alert not clearing
~~#20~~	Asset coordinates don't track container
~~#32~~	Remove email list view is sometimes stale
~~#33~~	Asset map has garbage marker
~~#36~~	Map view of buildings broken
~~#38~~	Rfid runs multiple times on CTB-proxy electronics
~~#39~~	Multiple copies of scanwifi run on CTB-proxy electronics
~~#45~~	scanwifi_aggregate_railsv2 botches different iwlist output formats
~~#46~~	Wifi aggregate location algorithm fails on obvious cases
~~#48~~	Wifi server fails to find "Lab" despite being in the database
~~#52~~	Facility and Asset Analysis tabs broken
~~#56~~	Import into empty database errors
~~#57~~	CTB-proxy Trigger Switch not read on Gumstix
~~#58~~	Map view reverts to default zoom on update
~~#59~~	live rule engine assertFact throws null pointer exception
~~#60~~	EPC tags with letters don't work
~~#67~~	Map menu is missing updateLocation method
~~#69~~	Ramses Case 2 fails to trigger reads
~~#71~~	Incompatible range rule seems to clear prematurely
~~#72~~	RAMSES Case 1 switch sense not responding to environment variable
~~#74~~	Setting any asset write permission prevents all edits
~~#75~~	Facility edit tab available to all roles
~~#16~~	Initial page after login sometimes blank (Safari)

Figure 44. Closed software defects

Status: assigned (12 matches)

Ticket	Summary
#21	Facility/Assets don't display on Firefox/Ubuntu
#22	Low inventory trigger stores multiple categories
#35	Very slow display of items with long histories/status
#37	Multiple AP WiFi tracking favors areas with few visible APs
#54	Jboss runs out of heap
#55	Location history scroll arrow's intermittent
#62	Inventory name change not reflected in facility display
#64	System seems to be repeating scans
#73	Read permission does not affect browse tree view and rules
#76	Tabs jump back to previous tab
#77	Weight rule does not clear
#47	rfid fails to reconnect (or exit?) if the SkyeTek is reset

Figure 45. Open software defects

Status: closed (4 matches)

Ticket	Summary
~~#18~~	Automatic authentication on Rules
~~#34~~	Display location/alert histories newest to oldest
~~#61~~	Limit display precision of weight
~~#70~~	Hitting "Enter" should initiate search

Figure 46. Closed software enhancement requests

Status: assigned (11 matches)

Ticket	Summary
#8	Rules and database changes don't affect existing triggers
#11	Ambiguous Class of Supply code partial matches
#44	Implement delete feature
#49	Redesign Database Access for History Items
#50	Cleanup Logo and Login/Login Error Screens
#53	Wifi limited to either a known AP list or any open network, but not both
#63	Sort facility assets by type, name or category
#65	User selection of number of assets to view on Facility List
#66	Add email feature should show the current list
#51	Add help popups on all buttons and fields
#68	Partial EPC asset search doesn't work

Figure 47. Open software enhancement requests

3.5 Task 5: Cost/Benefit and Market Analysis

3.5.1 Work Performed

The purpose of the Phase II RAMSES Cost/Benefit and Market Analysis was to establish a quantitative assessment of the benefits versus costs of the RAMSES system, particularly as it applies to use for Space Exploration Systems such as the international Space Station, and to identify markets and potential markets for the technology, both leading into an approach for further Phase III work. The following list the original set of subtasks described in the proposal:

5.1 **Cost/Benefit Analysis**. This will consist of a detailed bottom-up cost model for the smart container. The Phase 1 smart container, including all RFID equipment and electronics costs more than $5,000. We believe that such a high price would not allow for wide dissemination of such a system. Our cost target for a smart container in Phase 2 is $1,000 (excluding the embedded tablet-PC), with a target of below $500 for Phase 3. Additionally, this task will also include RAMSES maintenance and upkeep costs as well as monetization of the benefits of using the system such as personnel time saved, fewer lost items, less time lost looking for item and so forth. We will do this cost/benefit analysis for three different settings:

- ISS ongoing operations post-2010
- Constellation program (lunar outpost logistics)
- Ground operations on Earth (e.g. at NASA Stennis)

5.2 **Market Analysis**. The applications of RFID in terrestrial and military supply chains are beginning to proliferate at a fast rate. We believe that RAMSES does not yet, but will within the next 1-3 years, have competition in the area of smart containers and multi-level asset tracking. We will conduct a more detailed search and analysis for both competing and complementary technologies. This task also involves searching for specific applications beyond NASA. We believe that the uniqueness of RAMSES lies primarily in:

- The smart container concept, allowing multi-level tracking
- RAILS web-enabled middleware
- Flexibility to tailor the system to many applications and configurations, also by incorporation of third party RFID equipment and software components

5.3 **Phase 3 Planning**. Phase 3 planning will entail the specific identification of lead users, the refinement of key technical and business features of RAMSES and plans for initial deployment at NASA. We will develop a business plan, focusing both on the smart container, RAILS and the overall RAMSES architecture. While we have decided not to pursue the soft-bag concept (see Phase 1 final report, Section 4.6.7) in Phase 1 due to time and resource constraints, we believe that this concept has enormous long-term potential. We will develop a more detailed conceptual design for smart-bags as part of Phase 3 planning. One advantage is that at that time (2 years from the beginning of Phase 2), miniaturization of RFID readers and tags will have likely progressed to a point where a smart bag implementation becomes realistic.

Throughout the Phase III effort, MIT applied cost/benefit analysis techniques to the RAMSES technology, particularly as it applied to use on the International Space Station. This work included research of published materials and interviews with NASA personnel to determine the overall operational and time costs on station, particularly the costs spent performing inventory management. The MIT team quantified and integrated this information with the RAMSES Smart Container equipment costs, plus additional costs such as the mass and volume overhead introduced by the Smart Container equipment and spacing materials. Additionally, the MIT team considered the frequency and distribution of CTB use, for example, if all CTBs were accessed evenly, versus if the crew accesses only a small number of CTBs most of the time.

The results of this analysis then affected the on-going work of the team, particularly the design of the soft-shelled version of the Smart Container. For example, the MIT analysis determined that launch volume has having a significantly higher opportunity cost (i.e. volume taken by the Smart Container overhead limiting the opportunity to ship additional material to the ISS) than the overall mass of the container materials. This result led the team to attempt to optimize and reduce the volume overhead introduced by the Smart Container, particularly the antenna thickness and spacing material thickness used to keep the UHF tags from being too close to the container's shielding. Additionally, this result, combined with a result that showed that the crew tends to spend most of the inventory management time on a subset of the overall CTBs on-station. This led to the conclusion that NASA may obtain a higher value by having the option to retrofit CTBs already on-station rather than occur the volume overhead of an instrumented every CTB sent to the ISS.

Early in the Phase II effort, MIT performed some market analysis work, based on the initial work during Phase I. This work examine potential market areas, including Humanitarian/Disaster Relief, opportunities within the US DoD and with NATO, hospital applications, applications in aviation maintenance and repair, on-going work within NASA, and general consumer or commercial product opportunities, such as "Smart Cabinets" and "Smart Bags/Fabrics". This market analysis work identified key market opportunity areas and identified areas where others are finding applications and market traction for enhanced RFID technologies, such as instrumented containers and rule-based applications that utilize geo-location information to identify conditions and derive system properties.

During Phase II, the team undertook a number of initiatives to plan for possible follow-on work and a Phase III effort. Near the end of Phase II, an opportunity arose to perform additional Smart Container tests as part of NASA's Facilitated Access to the Space Environment for Technology Development and Training (FAST) program. MIT submitted and won a proposal to perform zero-G testing with the Smart Container to determine the effects of objects moving with six degrees of freedom within the container on the RFID read accuracy. Additionally, the team has been in on-going contact with Johnson Space Center for potential on-going work moving the Smart Container technology from the laboratory environment to eventual flight test and deployment. In addition to these efforts, through out the program, from the Wired NextFest event to meetings with companies such as Disney, the team has explored commercial opportunities and interest in the Smart Container concept coupled with the Rule-based Analytic approach available in the RAILS software.

3.5.2 Results

Cost/Benefit Analysis

The following summarizes the cost/benefit conclusions from a results paper provided by MIT and JSC[3]:

> "Based on the best information available and the study's assumptions, the cost/benefit analysis study suggests that the RAMSES system is likely to yield positive net present value if implemented in International Space Station CTBs via modification kits installed by the crew, provided that inventory transactions are at least somewhat concentrated within the subset of CTBs in which the hardware is installed. As one would expect, the more the inventory transactions are concentrated into a smaller percentage of CTBs, the greater the value delivered by the system (assuming that installation is limited to those bags).
>
> This result is intuitive, given that the largest costs associated with the system are directly tied to the quantity of CTBs that are equipped with RFID. The three largest costs are the opportunity costs for the launch volume (~42% of the total cost for the discrete analysis of the Modification Kits Implementation Scenario), launch mass (~33%), and crew time to install the modification kits (~21%).

[3] Abraham Grindle/MIT, Olivier de Weck/MIT, Sarah Shull/JSC, "An Autonomous, Real-Time Asset Management System for the Internalational Space Station: Net Present Value Analysis", 2008

On the benefits side of the equation, the single term that dominates is the value of the crew time saved that would otherwise be spent on routine updates of the Inventory Management System. The value of this saved time accounts for roughly 94% of the total benefits accrued by the system in the discrete analysis of the Modification Kits Implementation Scenario.

Also intuitive is the fact that the Modification Kits Implementation Scenario performs better than the Phase-In Scenario. This is a result of the fact that the modification kits are modeled as being launched and installed in FY 2009, and thus from FY 2010 – FY 2016 the full, recurring benefits of the system are able to add up and overcome the initial one-time costs associated with launch and installation. If the lifetime of the Station were to be extended, the recurring benefits would further accumulate; the net present value of all scenarios would increase, and it would become much more probable that the Phase-In Scenario would result in positive net present value.

It should be understood that at least one potential cost and one potential benefit were not included in this study, as they could not be quantified rationally. The cost of integrating the RAMSES system into the International Space Station's existing Inventory Management System is extremely uncertain. The system currently has the capability to receive and process input from external devices, such as barcode readers, but the difficulty involved in scaling this capability, or supplementing the current system with a separate RAMSES software module that simply feeds updates to the current, is quite uncertain. There is also a political challenge associated with such modifications, namely to obtain approval from all International Partners.

The unquantifiable benefits of this work include the enhancement of both crew safety and mission assurance. The ability to locate any item (or at least many items and/or any critical items, depending on the extent of implementation) at a moment's notice can potentially make the difference between life and death in an emergency situation. Furthermore, real-time knowledge of stock levels of supplies such as food, medicine, and spare parts can prevent unexpected stock-outs that could jeopardize crew health and/or Station reliability.

Clearly, NASA decision-makers and their international counterparts must carefully weigh all of these factors in their evaluation of this technology. There are substantial benefits associated with this proposed application of RAMSES, but also significant costs, challenges, and uncertainties. Further hardware and software development is necessary before any large-scale deployment can begin, particularly related to reducing hardware volume and mass, attaining sufficient battery life, and exploring compatibility issues with the current ISS Inventory Management System. Alternative hardware architectures - such as tagging all items but installing a fixed network of RFID readers and antennas rather than wiring and insulating all CTBs – could provide even greater value by requiring less mass and volume, if challenges such as location triangulation and EMF interference could be overcome. At the very least, this work demonstrates that an RFID inventory management system merits serious consideration for future human spaceflight applications. "

<u>Market Analysis</u>

During the early portion of Phase II, MIT performed a market analysis of potential areas of interest for RFID technologies in general, and the RAMSES approach in particular. This analysis covered a number of areas, some summarized below:

- Humanitarian/Disaster Relief
  - International
    Erica Gralla, from MIT, performed some a brief study of the needs of international humanitarian relief issues. She reported two key findings from this study that point to where RFID and Smart Container technology might prove the most useful:
    - Organizations are most interested in "last-mile" tracking of materials
      This is where most items are lost due either mishandling or outright corruption
    - It's easier to get medicine to field hospitals than to get food
  - Domestic
    A paper (from Hoguin-Veras, et. al.), identified relief supply issues pertaining to Katrina

        - Three unknowns to relief supplies: What? Where? When?
        - Use of GPS on trucks was insufficient. From the paper "The information available shoild not be limited to the location of vehicles. The use of technologies such as radio frequency identification devices would provide field commanders and state officials about ht specific contents of vehicles and pallets."
        - How does one quickly add RFID Readers to trucks? "Backpack"?
- US Department of Defense and North Atlantic Treaty Organization
    - DoD and NATO are already in the RFID business
    - 1994: DoD signed a contract with Savi Technologies for an RFID asset tracking system
        - The system is now operational in 42 countries at 2000 locations
        - The system tracks at the pallet level, not at the individual item level
    - 2004: NATO began a pilot to phase in a similar and internationally compatible system
- Hospital Applications
    - Interested in tracking high-value, portable equipment,; patients; and staff
    - Very strong return-on-investment in pilot studies; a very active market
    - A variety of solutions in pilots and in -place:
        - Passive tags - Hartford Hospital Pilot, 2006
        - Active (Wi-Fi) - Beth-Israel DMC pilot, 2004
        - Ultrasound - Brigham and Women's, 2006
        - Infrared systems
        - 802.154 systems
- Aerospace
    - Active areas of interest, particularly in maintenance, repair and overhaul
    - Challenge is that one part can ground an aircraft
        - Short haul domestic cancellation costs approximately $60K
        - International cancellation costs approximately $600K
    - Possible RFID applications to permanently tagged parts
        - Support/Accelerate spare parts handling
            - Each shipping container is opened 5 to 10 times to determine part identity
        - Aircraft Configuration Management
            - Know exactly which parts/models are in the aircraft in real-time
        - Predictive/Proactive Maintenance
            - for example, store engine sensor data on the engine RFID tag
        - Track and Trace engine shipments with GPS/GMS
        - Note: Boeing already uses roughly 2000 RFID tags in the new 787s

<u>Phase III</u>

Late during the Phase II effort, MIT proposed and won an opportunity to test the RAMSES Smart Container concept on the FAST program, parabolic flight providing a simulation of zero-Gs, giving the team an opportunity to bring the technology the technology from a Technology Readiness Level (TRL) 4 (laboratory) to TRL 6 (relevant environment). Additionally, the team has been in contact with inventory management research personnel at JSC to try to establish a technology demonstration in the summer of 2009. Throughout the Phase II effort, members of the team have provided technology demonstrations to potential commercial partners and customers and continue to work these leads for support on a Phase III effort.

3.6 Task 6: Critical Design Review

3.6.1 Work Performed

The goal of the Critical Design Review, as proposed in the original proposal, was to provide the COTR an opportunity to review the key system and component design and recommend changes. The originally proposed subtasks for this effort consisted of:

6.1 **Electronic Documents Preparation**. The CDR will involve a review of project requirements, mechanical, electronics, and software designs, and testing plans. The associated documents (requirements, specifications, drawings, circuit diagrams, software flow diagrams, parts lists, testing plans) will be updated and correlated for distribution to the review personnel.

6.2 **Critical Design Review**. PSI and MIT will host the CDR at PSI facilities in Cambridge, MA. This will permit the review personnel to inspect the RAMSES Phase 1 and Phase 2 prototypes and to interact directly with the RAMSES project staff. The timing of the CDR will be coordinated with the NASA COTR.

Unfortunately, due to several factors, the team was unable to present a Critical Design Review to the program's COTR. Despite this, however, the team conducted informal and on-going internal reviews and meetings to adapt the design. These reviews and meetings took the to-date design and adapted the design based on the early field tests and based on the results of MIT's cost/benefit analysis. The built hardware and software elements were the result of an evolutionary and iterative approach and benefited from the test and analysis experience developed over the course of the program.

3.6.2 Results

The results of the on-going development and review the system were reflected in the rigid and soft-shelled Smart Container designs, in the migration from the original Rails software architecture to the Java-based architecture. Additional design and implementation comments and items were recorded in the issue tracking system, shown in Section 3.4.2.

3.7 Task 7: Reporting

3.7.1 Work Performed

The originally proposed reporting plans for Phase II were as follows:

7.1 **Phase II STTR Reporting**. Throughout the Phase 2 activity, this task will produce the quarterly and final reports. In addition to required contractual reporting, periodic contact with the Phase 2 technical monitor will be maintained in order to involve the monitor on a more interactive level than is provided by reports alone. (This approach was extremely beneficial to this project during Phase 1.) This will also ensure that the final hardware capabilities coincide with the needs of the programs that will benefit from the research.

7.2 **NASA RFID and Wireless Working Group**. Throughout the Phase II effort, we will continue to interact with the NASA RFID and Wireless Working Group. This group, chaired by NASA JSC, is focusing its efforts on technology enhancements to logistics management on ISS with the intention of influencing further exploration logistics management technologies. We have already begun coordinating our efforts with this group, and plan to participate in a workshop on space applications of RFID to be held in Houston in spring 2007.

Despite proposing quarterly reports, the contact stipulated monthly progress reports. Initially, the team failed to submit the initial reports on a monthly basis and aggregated the first series of reports into two quarterly reports. The program's contract manager noticed and informed the team of the discrepancy, and subsequently the team began the required series of monthly progress reports for the remainder of the program.

3.7.2 Results

With the submission of this final report, the team has submitted all required status reports, aggregating the first 6 months into one report, the following three months into the 2nd report, and with all remaining reports provided on a monthly basis.

3.8 Recommendations for Future Development

During Phase II, the team identified some areas for future development and deployment of the technology, some of which are already in progress. Some of these recommendations include

- Advance the technology demonstration from TRL 4 (demonstration in a laboratory environment) to TRL 6 (demonstration in a relevant environment) through NASA's FAST program
- Improve the Smart Container power management and measure the charge shelf life-time of the Smart Container's battery Smart Container
- Reduce the overall electronics cost by removing the intermediate host electronics board between the RFID reader and the control/communications electronics
- Perform additional testing with water and metal tolerant UHF tags
- Integrate GPRS/GPS capability into the core electronics
- Improve the room-level tracking technology - possibly integrating 3rd party beacon-based systems
- Simply and streamline the RailsV2 server software
- Resolve performance issues with the RailsV2 software
- Provide support for user-defined screens and displays
- Perform customer surveys and tests and integrate those surveys into the overall product development plan
- Integrate 3rd party interoperability support into the RailsV2 server software
- Resolve all outstanding issues and enhancements within the RailsV2 software
- Provide bandwidth-optimized views from the RailsV2 server software
- Simplify and extend the RailsV2 rule-language to improve 3rd party and customer development

4 Potential Applications

4.1 Potential NASA Applications

During Phase II the team focused on inventory management for space missions, particularly the use of instrumented Cargo Transfer Bags, providing automated inventory tracking of stored and stowed items within the ISS and future vehicles and systems.

Applications of the RAMSES system for NASA include:

- ISS Inventory Management
- ISS Trash and Discard Monitoring
- Inventory Management within the "Orion" Crew Exploration Vehicle
- Lunar Habitat
- Ground tracking of high-value assets

4.2 Potential Non-NASA Commercial Applications

Part of the market research, described in 3.5, was to identify non-NASA areas of applicability for the RAMSES and related technology. The team identified key areas of opportunity and applications such as:

- Tracking of humanitarian and disaster relief supplies, particularly in the "last mile" of delivery of those supplies
- Tracking of individual items within DoD and NATO container systems
- Parts tracking in aerospace maintenance operations
- Tools and equipment tracking in aerospace manufacturing environments
- Equipment, patient and staff tracking within hospital environments
- Control of medicines, equipment and chemicals in hospital and research environments

5 Contacts

5.1 Key Contractor Participants

Prof. Olivier de Weck, Principal Investigator, MIT, 617-715-5195

Mr. Abraham Grindle, Researcher, MIT, 617-650-3792

Mr. Joseph Parish, Program Manager/VP Payload Systems Div, Aurora Flight Sciences, 617-500-0248

Mr. James Francis - Program Manager/Software Engineer, Aurora Flight Sciences, 617-500-0276

5.2 Key NASA Participants

Ray Bryant, COTR, NASA/SSC, 228-688-3964