Maritime Freight Container Management System
Using RFID

S. J. Barro-Torres, T. M. Fernandez-Carames, M. Gonzalez-Lopez, C. J. Escudero-Cascon
Dep. of Electronics and Systems
University of A Coruña, A Coruña, Spain 15071
{sbarro, tmfernandez, mgonzalezlopez, escudero}@udc.es

Abstract – Efficiency in cargo exchange in a maritime container terminal is very important due to its associated cost. Among the different available technologies for optimising the terminal operation, RFID identification is one of the most promising ones. In this article, a freight container management system is presented, composed of a software platform supporting the typical use cases, as well as a wireless RFID network deployed throughout the terminal. This system allows for container traceability, efficient reservation and vehicle management, automatic loading plan calculation, and fault tolerance. Analysis, design and implementation stages are discussed, including the selection of the most suitable hardware and an execution example of one of the use cases.

I. INTRODUCTION

Intermodal transportation involves the use of freight containers using multiple modes of transport (railway, freighter and lorry) without any handling of the freight itself when changing modes. The freight containers are exchanged in intermodal stations or terminals. The optimisation in the management of such terminals is very important because the load times have a great impact in the economic profit. Thus, the efficiency of identifying and locating freight containers inside the terminal is critical.

Among the different possible intermodal transportation scenarios, we have centered our study in the optimisation of the management of a maritime freight container terminal, from the time they arrive in lorry until they depart in the container ship. Some terminal operations are susceptible of being optimised by using RFID (Radio Frequency Identification) technology. In this article a freight container management system developed by the authors is shown, describing all the steps followed to obtain such system: analysis, design and implementation, including the selection of the most suitable RFID reader for this application.

The article is structured as follows. Section II introduces the state of the art, describing the identification methods currently used in freight container terminals. Section III describes the system description itself. Finally, Section IV is devoted to the conclusions.

II. STATE OF THE ART

The major objectives of container ID tracking consist in performing quickly and accurately the following tasks: container identification, seal check and damage inspection. Container identification regards the correct reading of the id mark, which is usually compliant with a standard, such as ISO 6346:1995 [1]. Nowadays, this operation is performed mainly by human operators (shippers, forwarders, and consignees), OCR-based systems or RFID-based systems. OCR (Optical Character Recognition) is an error-prone process: there might exist damaged characters, different container surfaces, manufacture-dependent font colours and shapes, need for LOS (Line-Of-Sight), etc. For example, the project MOCONT (Monitoring the yard in CONtainer Terminals) [2] includes a specific visualization module in its architecture, using a camera to capture the container images. More details about the character recognition process used in MOCONT-II, an improvement of the former, can be found in [3]. With respect to character extraction algorithms, there are different approaches depending on the application context. For instance, [4] deals specifically with feature extraction in gray-scale images, while the article referenced in [5] describes a more general algorithm. Despite the mentioned problems, it is easy to find commercial products using OCR technology, since it does not need very sophisticated hardware [6-9].

On the other hand, RFID is a suitable technology for terminal operation, as several recent applications have shown [10]: Port of Houston is using RFID to ensure the security of cargo containers, Port of Virginia has improved the security and efficiency of the processes surrounding its cargo container shipments, Port of Savannah has enabled shipments tracking, Port of Busan (South Korea’s largest port) has secured and speeded up its business processes, and Port Louis Harbor has adopted a tracking system [11]. It is worth mentioning an initiative to create a worldwide RFID network for tracking and securing cargo containers using ISO 18000-7 standard and active tags made by Savi Technology [12].

D. Mullen [13] highlights the benefits of using RFID in a port/terminal application: automation of data acquisition, accuracy and completeness of the collected data as there is no human intervention, and better utilization of employees’ time. This article also identifies the five major areas of potential RFID applications: access control, container security, identification and localization, activity tracking and regulatory compliance. But RFID has also several disadvantages: until recently there was not standardization criteria to identify containers, tag cost, reader and tag collisions, and electromagnetic propagation issues.

III. MARITIME FREIGHT CONTAINER MANAGEMENT SYSTEM

3.1 Analysis
This section is divided into two main parts: the software analysis, where requisites of the container management software in terms of use cases and requirements are explained, and hardware analysis, where the most appropriated RFID reader to this application is selected based on a set of candidates.

### 3.1.1 Software Analysis

A container terminal is a place devoted to loading and unloading containers by means of lorries and freighters. Figure 1 shows a typical layout of a terminal, displaying the most representative elements. Generally speaking, a freight container terminal is structured into three different parts. A Stack Area is where containers are stored waiting for their destination. This area consists of a series of stack cells, located in different roads, with a gap between them to facilitate transport operations inside the terminal. The Loading Bay is a temporary storage used for loading and unloading ships, also consisting of a set of cells. And finally, the Access Gates, where External Lorries can get in and out of the terminal.

Several actors, described below, interact inside the terminal. On the one hand, there are External Lorries, in charge of road transport; meanwhile, Freighters carry containers across the sea. Freighters have cells to transport containers, and their layout depends on the kind of ship. On the other hand, there are several cranes, with different mechanical arms to attach containers and move them within the terminal: the Reach-Stack Cranes put the containers on the Stack Area, having complete freedom of movement; the Quay Cranes, fixed to their corresponding Dock, load and unload freighters; and the Gantry Cranes, used to move containers within the road. Finally, the Security Control Area is used for watching over the access gates.

The terminal operation is described below in terms of use cases. Managing the terminal involves using information about all the containers inside the terminal, including source, destination and a reservation system. Despite their origin, containers must always be registered before entering the terminal, indicating arrival and departure dates. External Lorries may bring containers or take them away. In both situations, a reach-stacker crane is assigned to move the container from/to the cargo area in the lorry. However, when a container departs it is necessary to know beforehand the destination vehicle, to prevent the container from being taken by an unauthorized lorry. Freighters, on the contrary, must always be registered, indicating the container list to load/unload by sea. When the freighter is docked, the loading plan is calculated, consisting of a sequence of operations that must be followed scrupulously. For this goal, it is necessary to know the ship structure. All freighter loading operations are performed using the loading dock as temporary storage. Then, taking into consideration the plan previously calculated, a container list is associated to the quay crane to proceed with the unloading operation. Once the containers are on the loading dock, they are moved towards the stack cell which has been reserved for them. Then, they are assigned a reach-stacker crane. To load the freighter we proceed in the opposite way, assigning a reach-stacker crane to each container included in the loading plan, which are responsible for moving the containers from the stack area to the loading dock. Finally, the quay cranes are assigned to perform the loading of the containers into the freighter.

In addition to supporting the use cases explained above, the system must ideally meet the following requirements:

- Traceability. It must be able to provide location information about a container at any time.
- Reservation operations. Apart from managing the containers currently placed in the terminal, it must be possible to include a reservation subsystem, improving the terminal resource management.
- Efficient crane management. An efficient assignment of cranes is fundamental to reduce the amount of movements of vehicles inside the terminal and, thus, reduce their fuel consumption.
- Loading plan. From the ship layout information (containers to load and unloading) the system should be able to calculate the most appropriate loading plan automatically.
- Fault tolerance. The system must recover its previous state and continue working normally when an error occurs.

### 3.1.2 Hardware Analysis

There are four important parameters when selecting an RFID reader: operation frequency, communications range, memory size and security features. On this application, we will focus our attention mainly in the operation frequency and communications range.

Briefly, the two types of RFID tags that are of primary interest are passive and active. Active RFID tags can have a range up to 100 m, and can contain sensors, global positioning devices (GPS) or any other enhancements. Active RFID systems are not as widely deployed as passive systems, are usually custom designed for specific applications and have not standardized until recently. EPC (Electronic Product Code) Class 1 Generation 2 standard, also known as ISO 18000-6C, is one of the most common passive UHF RFID systems.

UHF (Ultra High Frequency) readers are the most appropriate and popular considering the particular needs for this project, because of their communication range and signal propagation properties, although other alternatives exist in the microwave frequency band of...
different systems and configurations, adapting it to the specific needs of the container yard. TrolleyScan’s solution is available as a complete solution, including the TrolleyScan Reader and the TrolleyScan Agent. The TrolleyScan Reader is a robust, high-performance reader that can operate in harsh environments. It is equipped with a powerful processor and a large memory, allowing it to handle complex tasks and processes. The TrolleyScan Agent is a software component that communicates with the TrolleyScan Reader and provides the necessary interfaces for integration with other systems. The solution is scalable, allowing it to be deployed in container yards of different sizes. Figure 2 shows a diagram of the TrolleyScan solution, with the TrolleyScan Reader and the TrolleyScan Agent highlighted.

The implementation was divided into three main parts, as shown in Figure 3: a software platform supporting the different use cases and classes described in the section above, an UDP/IP network allowing remote communications between the different actors and a test GUI client. The software platform has been implemented using .NET technology, mainly because of the ease of integration with the database and the RFID reader. Several design patterns were used: Facade, Abstract Factory, DAO (Data Access Object) and VO (Value Object) patterns [22]. As it can be seen in Figure 3, the platform is formed by three different levels. The first level is composed of the Façades, a group of classes implementing the use cases described in section 3.2. To perform these use cases, the Façades make use of DAO classes, which perform basic database operations at element level (select, insert, find, update and remove), establishing the second level. The third level is based on a database server, Microsoft® SQL Server, which stores the state of the terminal. All database operations are transactional, so data integrity is assured in case of failure. Concerning the UDP/IP network, a remote command protocol was implemented, enabling the invocation of the different use cases by any of the different actors of the terminal, no matter where they are located. Finally, a client was developed to test the mentioned protocol easily from a GUI environment, as well as to demonstrate the integration of the RFID reader with the rest of the system. Adaptations of the code of this client can be used to run the protocol on any hardware.

**3.3 Implementation**

The implementation was divided into three main parts, as shown in Figure 3: a software platform supporting the different use cases and classes described in the section above, an UDP/IP network allowing remote communications between the different actors and a test GUI client. The software platform has been implemented using .NET technology, mainly because of the ease of integration with the database and the RFID reader. Several design patterns were used: Façade, Abstract Factory, DAO (Data Access Object) and VO (Value Object) patterns [22]. As it can be seen in Figure 3, the platform is formed by three different levels. The first level is composed of the Façades, a group of classes implementing the use cases described in section 3.2. To perform these use cases, the Façades make use of DAO classes, which perform basic database operations at element level (select, insert, find, update and remove), establishing the second level. The third level is based on a database server, Microsoft® SQL Server, which stores the state of the terminal. All database operations are transactional, so data integrity is assured in case of failure. Concerning the UDP/IP network, a remote command protocol was implemented, enabling the invocation of the different use cases by any of the different actors of the terminal, no matter where they are located. Finally, a client was developed to test the mentioned protocol easily from a GUI environment, as well as to demonstrate the integration of the RFID reader with the rest of the system. Adaptations of the code of this client can be used to run the protocol on any hardware.
3.4 Execution Example

Figure 4 shows a sequence diagram exemplifying one operation from the use cases described. More specifically, it is about the validation performed when a reach-stacker crane hooks a container with its mechanical arm.

The crane sends the remote command “RFID reading” (step 1), indicating its plate number along with the RFID ID which has just been read. Then, the UDP server reacts executing the “RFID reading” method found in Terminal Vehicle Façade (step 2). This Façade make use of Terminal Vehicle (steps 3-5) and Container (steps 6-8) DAOs in order to obtain the object representation of both reach-stacker and container elements, respectively. Step 9 performs the validation by checking if the current crane is the same as the one indicated by “next transport” container attribute. Then, steps 10 and 11 set the value of “next transport” and “located in” to reflect the new state of the container. Step 12 save the mentioned changes in the database and finally, steps 13 and 14 sends the remote response back to the reach-stacker crane.

III. CONCLUSIONS

This article has presented a freight container management system based on RFID identification. This system meets the requirements of freight container traceability, efficient reservation and vehicle management, and fault tolerance. Its design, analysis and implementation have been discussed with diagrams, including the selection of the most appropriate hardware to this application.

ACKNOWLEDGEMENTS

This work has been possible thanks to the support of the following projects: TIMI (2007/CENTIT/6731), LOCUS (07TIC019105PR) and PIRAmIDE (TSI-020301-2008-2).