# WIRELESS PLATFORM FOR ZIGBEE AND RFID
# [RFID PLATFORM FOR MONITORING TRAVELS IN THE LISBON METRO]

*Teresa Vazão, Rui Rocha, Luís Pedro*

*Instituto Superior Técnico – Taguspark*
*Av. Prof. Dr. Aníbal Cavaco Silva, 2744-016 Porto Salvo, Portugal*
*tvazao@tagus.inesc-id.pt, rui.rocha@lx.it.pt, luismdpedro@gmail.com*

## ABSTRACT

**RFID (Radio Frequency Identifier) is a key enabler of new business models that rely heavily on automation. Transportation is one of the areas that benefits from this automation, since RFID allows for automated tracking of in motion vehicles. This work presents the conception, development and implementation of an RFID platform that has the objective of track trains in the Lisbon Metro. The solution presented here uses active RFID technology with readers installed in key locations and tags mounted in trains. It is also presented a software architecture for a data collecting system and a client that interacts with it. The presented solution was implemented in a laboratorial environment that resulted in a functional prototype that uses scale model trains to simulate the Metro network and passive RFID to track the scale trains, and uses Biztalk RFID to implement the collecting data system.**

***Index Terms—RFID (Radio Frequency Identifier), tags, readers, trains, travels, data collecting, Phidgets, Biztalk RFID***

## 1. INTRODUCTION

New wireless technologies are enabling companies to support new business models that rely heavily on automation to improve operational efficiencies. One key enabler is RFID (Radio Frequency Identifier), which changed the way information is gathered and allows companies to have access to real time information regarding the localization of their goods and equipments.

Aware of these opportunities, the Lisbon Metro company felt the need to automate the monitoring of travels in their network, through the use of RFID technology. This work is the result of collaboration between Instituto Superior Técnico and Tecmic to conceive, develop and test an RFID platform to track trains in the Lisbon Metro and store the travels performed by each train.

Nowadays there are several RFID platforms used to track in motion vehicles. The main applications of this systems are tolling systems, vehicle fleet management, access control, accurate positioning, arrival and departure control and in board information updates. These RFID platforms can be found in London, Bilbao, Hamburg and several other locations.

This work proposes a system architecture to solve the posed problem. Through the use of active RFID technology, with readers installed in key locations and tags mounted on each train, is possible to track all the trains. The information gathered by the readers is then sent to a centralized data collecting system, whose main objectives are to integrate and manage the RFID equipment, and to store and analyze the received information. It is also described a client to interact remotely with the centralized system.

This system architecture was implemented in a laboratorial environment, and the result was a functional prototype that uses scale train models and passive RFID to simulate the Metro network. Biztalk RFID is explored to implement the business logic of the platform.

The functionality of the solution is proved by several tests executed in the functional prototype.

## 2. STATE OF ART

Radio Frequency Identifier is a technology that allows for entities identification by the means of radio frequencies. The most common way to do this identification is by the installation of a tag in the entity to be identified that when enters the reading zone of a reader will be detected, and the information regarding this detection will then be sent to a data collecting system. The next figure contains the elements of an RFID system [1] [2] [3] [4].

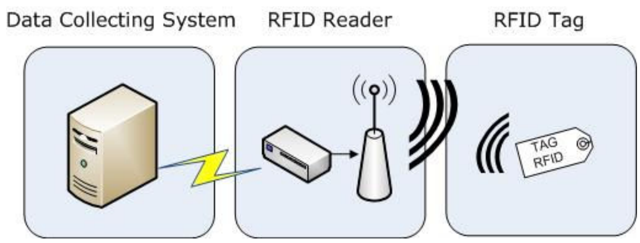


**Figure 1 - RFID System components**

Tags can either be passive, active or semi-passive. The main difference between them is that active tags make use of power sources to enhance their capacities while passive tags don't. The use of power sources enables active tags to have longer communications range and allows for faster data transmission than passive tags. This capabilities make the active technology the one to be used to track in motion vehicles. [5] [6]

Readers are connected to an antenna that creates a magnetic field, which corresponds to the reading range of a reader, in order to detect a tag. Readers can operate in various frequencies, but the ones commonly used to track in motion vehicles are the higher ones like 2.4 GHz, since they allow for faster data transmission between the reader and the tags. Readers have built-in communication interfaces in order to communicate with data collecting systems. The most common communication interfaces are serial port, TCP/IP and USB.

The advantages of using RFID in transportation are [7] [8]:

- Improvement of the fleet management
- Operation costs reduction
- Easy access to vehicles location
- Elimination of accidents caused by human errors
- Higher security, effectiveness and reliability

And the main applications of RFID technology in transportation are [9] [10]:

- Tolling systems [11] [12] [13]
- Accurate vehicle positioning information [14] [15]
- Vehicles tracking
- Access control
- Identification of vehicles [16]
- Speed monitoring [9]
- Arrival and departure information updates
- On board vehicle information updates

There are two major ways to configure the equipment for trains tracking [16]. One uses the reader installed on the train and tags mounted on the tracks, as figured in figure 2 and the other uses readers installed on the tracks and tags mounted on the trains as shown in figure 3 [14].

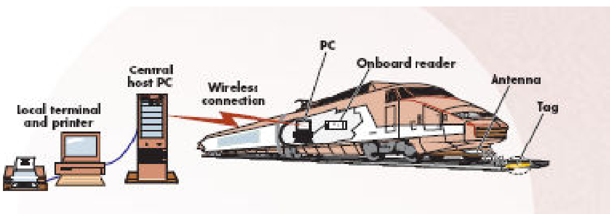


**Figure 2 - Configuration with readers installed on the train**

A configuration with a reader installed on board is used when a train needs to have access to information regarding is localization. It is possible to communicate that information to an exterior data collecting system through the means of a wireless connection [17] [18]. The applications where this configurations may be applied are in board updates, speed monitoring and colisions avoidance.

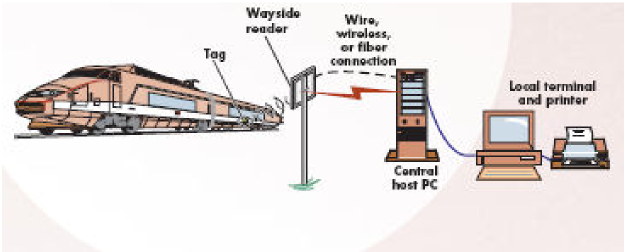


**Figure 3 - Configuration with reader installed on the tracks [14]**

The configuration with readers installed on the tracks is more appropriate for applications where the train doesn´t need to have information regarding his location. Examples of these applications are fleet management, vehicles tracking and arrival and departure information updates.

There are several RFID platforms installed nowadays to track trains [19]. In the London Metro [20] this kind of platform is used to change the TETRA conversation groups inside a train. The chosen configuration was the one with the track installed on the train and tags on the tracks.

In the Hamburg Metro [21] and Bilbao local trains RFID is used to update the departure and arrival information given to the passengers. The configuration used is the one with readers installed on the tracks and tags on the trains.

In the Netherlands [22] railways an RFID platform is used to complement a train axle's load measurement system. There are several sensors installed along the rails and there is a need to identify the trains that pass in those sensors. The used configuration is the one with readers installed on tracks and tags on the trains.

## 3. SYSTEM ARCHITECTURE

The system architecture to be described here, has to accomplish the following requirements, that where identified during the specification of the problem.

Hardware Requirements:

- Usage of RFID technology

- Equipment has to track readings at 70km/h (maximum speed of a Metro train)
- Tracking has to be made in both traffic directions
- Tracking has to be made in four locations, rail, park, garage and terminal
- The equipment to be used must have TCP/IP communication interface and allow remote configuration

Software Requirements:

- Independent to the RFID equipment used
- Integration and monitoring of the RFID equipment
- Storage of the received data
- Tolerant to RFID equipment faults
- Data replication
- Calculation of travels by train
- Remote access to information and configuration of the platform

### 3.1. System Overview

The design of this platform follows the same of a regular RFID system. It is composed by a hardware block composed by tags and readers, connected to a data collecting system and a third component, the remote access terminals. The following figure represents the system overview.

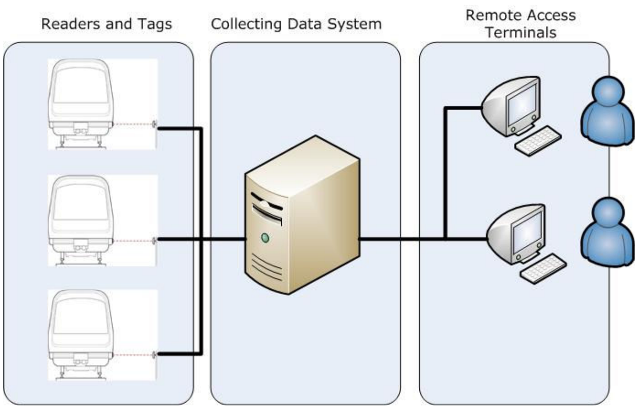


**Figure 4 - System Architecture**

The connection between the readers and the data collecting system will be made through TCP/IP interfaces, since there is an installed Ethernet network over the Metro installations. The communication between the remote access terminals and the data collecting system will be made through the Internet.

The option to use a centralized data collecting system has to do with the maximum information input that the system has to support. As it will be presented in the hardware architecture the number of readers expected to be installed is equal to 28, so the maximum number of tags read per second will never be superior to that number. This option may have some availability issues, so there is a need to use some redundancy mechanism. However this work does not explore those.

### 3.2. Hardware architecture

The hardware architecture includes the installation of the RFID readers and tags.

In this architecture the readers are installed in the rails and the tags mounted in the trains. This configuration is the most appropriate for this problem since the train doesn´t need to have access to the information regarding his position. Informations regarding this project assure that the number of sites where a detection needs to be done is equal to 14, while the number of trains is equal to 330, so the total cost of installing readers in the trains would be much superior to the chosen configuration.

The places chosen to install the readers are those that represent a border between two of the four possible places to make an identification, rail, park, garage and terminal.

There is a need to install a reader for each existing track in the border. The most common case is to install two readers by border, since most borders only have two tracks, one for each traffic way. The next figure represents this example.

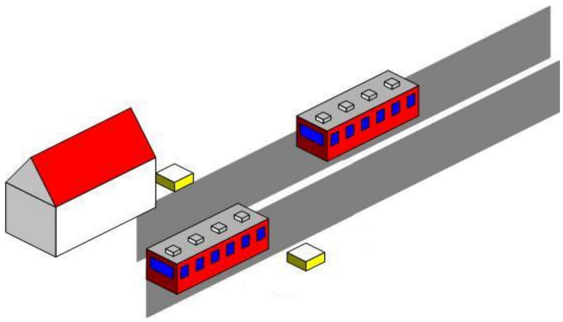

**Figure 5 - Readers installation**

The tags will be mounted in both laterals of a train, in order to track readings in both directions.

Each reader will be installed near a network connection, and this installation must obey a pre specified height, so that all the tags are mounted in the surface of the trains in that specified height.

In order to fulfill most of the requirements the equipment to be installed has to use the active RFID technology. This is the only technology that assures a reading at such speeds, since the passive RFID technology does not support them.

The chosen equipment to be used is the following:

**LR3-HD Tagmaster Reader** [23]

- Operating frequency : 2,4 GHz
- Reading range: up to 3 meters
- TCP/IP communication
- Track's readings at 400 km/h
- Heavy Duty
- Linux based operating system

**MarkTag S1456 HDS Tagmaster tags** [24]

- Operating frequency: 2,4 GHz
- Reading range: up to 3 meters with the LR3-HD reader
- Battery lifetime: 10 years
- High passage speed, up to 400 km/h
- Heavy Duty design for tough environments
- Read Only with an unique identifier

### 3.3. Software architecture

The software architecture includes the specification of the data collecting system and a client to interact with it. The next figure shows an overview of this architecture.

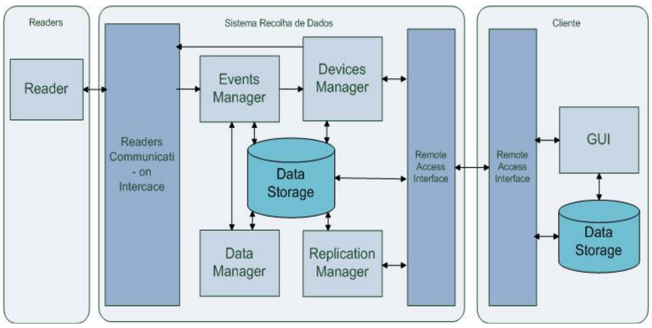


**Figure 6 - Software Architecture**

This architecture is based in an event model, where the RFID readers are the main source of events that must be subscribed by the data collecting system.

The most common events to be generated by readers are the following:

- Tag Gain Event – tag enters the reading zone of a reader
- Tag Lost Event – tag exits the reading zone of a reader
- Reader Attached – reader connects to the system
- Reader Dettached – reader disconnects from the system

The collecting data system is a modular system where each component fulfills one requirement. Next will be described each component and their functionality.

**Readers Communication Interface**

The main functionality of this interface is to integrate the readers with the data collecting system, creating an abstraction layer between the hardware and the software. In order to subscribe the events generated by the readers there is a need to implement an adapter for each type of reader to be used. This adapter implements the communication protocol of the devices, and is responsible to manage the connection with the reader, the reader operations and subscribing of events. The received events will then be transmitted to the Event Manager.

**Event Manager**

As the name suggests this component is responsible for the management of events. To direct an event to the right component there is a need to use handlers. The components that are going to subscribe events are the Devices Manager, Data Manager and the Data Storage. When an event arrives to the Event Manager it will apply all the handlers to it, in order to fulfill the application logic and store the information contained in the events.

**Data Manager**

This module is responsible for the execution of the logic of this platform. The solving of RFID device errors, manipulation of data and travels calculation are this component responsibility. Some of the problems that the Data Manager has to solve are:

- Duplicated events
- Out of order events
- Reading faults

The resolution of problems is made through the analysis of the data stored in the Data Storage and the creation of a virtual mapping of the RFID readers network. The calculation of travels is also made based on this mapping. The notion of travel for this platform is each group of two readings tracked between two consecutive readers. In case of detected errors the Data Manager should infer the readings that should have been made by the readers.

**Devices Manager**

The goal of this module is to manage the RFID devices. All the configuration options of the equipment should be available here. This component is also responsible for the manual and automatic addition of new readers. It is also important to provide an interface to the user in order to make it possible to access the configurations.

**Data Storage**

This is the component where all the gathered information will be stored. Metadata of the trains, readers and connections between readers should be stored here along with the tracked readings and calculated travels.

**Data Replication Manager**

The Data Replication Manager is responsible for the replication of the data to the final clients of this application.

**Remote Access Interface**

This Interface is used to allow the remote access to the system. It should implement methods that let the client manipulate the information stored in the Data Storage and interact with the Devices Manager. It is also through this interface that the Data Replication Manager communicates with the clients.

**Client**

The client is composed by a Data Storage and a Remote Access Interface that have the same functionalities as the ones in the data collecting system. The client also has a graphic user interface that allows the user to have a visual support for the data stored and ways to invoke the methods to communicate with the data collecting system.

## 4. IMPLEMENTATION

A functional prototype was implemented in laboratorial environment, to make a proof of the concepts presented in the last section.

This prototype uses scale train models to simulate the Metro network. Due to the small speed of the scale trains, and the much smaller price, the technology used to make the tracking of the trains was passive RFID. The used equipment was the following:

**Phidget RFID Reader** [25] [26]

- Operating frequency: 125 KHz
- Reading range: up to 6 cm
- USB (Universal Serial Bus) communication
- LED (Light Emitting Diode) who flashes on readings
- Uses the EM Marrin (EM4102) protocol [27]

**Phidget Tags**

- 40 bits unique identifier
- Uses the EM Marrin (EM4102) protocol
- Read-Only

Since the models used only have one rail and one traffic direction, the readers where installed in a different way than specified. The next figure represents it.

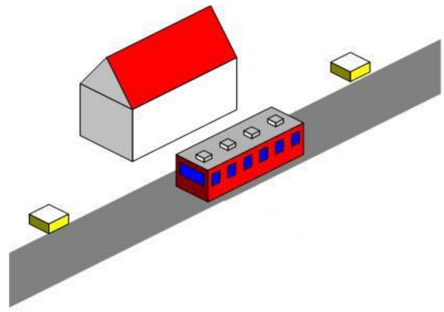

**Figure 7 - Prototype readers installation**

To emulate the existance of the four areas where a detection needs to be done were installed readers right before, and after those areas. This way the readers are still installed in borders of rail's, park, garage and terminal. A tag is mounted in the exterior lateral of the scale train.

The connection to the data collecting system is made through USB, which makes impossible to test the impact of a communication network between the readers and the data collecting system [25] [28] [29].

### 4.1. Collecting Data System

In order to implement the data collecting system, Microsoft Biztalk RFID [30] [31] was used. Biztalk RFID provides the mechanisms to implement the Readers Communication Interface, Event Manager and Devices Manager.

The Device Service Provider Interface (DSPI) [32] from Biztalk RFID offers the same capabilities as the ones specified in the system architecture. This DSPI allows the integration of the readers by the means of a provider, which may be developed by the user or supplied by hardware vendors. In the case of the Phidgets readers, the provider was developed by the Biztalk RFID creators [33] as a case study. So in order to integrate the Phidget reader with

Biztalk RFID, there was only the need to publish the provider in Biztalk.

The detections made by the readers will originate Tag Gain events that will be translated into Tag Read Events inside the provider. These events contain the 40 bit unique identifier of the tag, the id of the reader who made the detection and a timestamp with of the detection. The Tag Read Events will then be redirected to the Events Manager.

Biztalk RFID provides an Event Manager that allows the user to create processes which can be bound to a data source, in this case the RFID readers. The user can program handlers and associate them with these processes to achieve the application logic. The implemented handlers will be explained in the 4.3 section.

Biztalk RFID has a built-in Devices Manager. This device manager allows the user to control the configuration of the reader and have access to statistics regarding the number of tags read by the reader. The configurations are made through a graphical interface that includes a list of readers connected to Biztalk RFID and the state of the connection. When a reader is connected to the system a unique identifier is automatically generated by the Device Manager to identify the reader in the system.

### 4.1.1. Data Model

The Data Storage was implemented with the use of Microsoft SQL Server 2005 [34]. The data to be stored in the data storage is the following:

- **Trains:** Trains are the entity that this platform intends to identify. There is a need to make an association between a tag id and a physical train. In this table is stored the name of the train, the tag id, the current location and a timestamp of the last detection.
- **Readers:** As it happens with the trains there is a need to associate an id (in this case generated by Biztalk RFID) with a physical device. In this table is stored the metadata associated to a reader, like the location, the border and the side of the rails where it is installed and a timestamp of the last reading.
- **Reader Links:** For the system to be able to make a virtual mapping of the readers network, there is a need to store the connections between consecutive readers.
- **Detections:** Each detection is stored in this table. A detection is composed by the id of the train tracked, the id of the reader that made the detection and the timestamp of the detection.
- **Travels:** In this table is stored the travels made by each train. A travel is a group of two consecutive detections of a train. The information stored are the source and destiny readers, the id of the train, the timestamps of the departure and arrival and if a travel was inferred by the system due to errors.

### 4.1.2. Data Manager

The results of the implementation of the Data Manager are three handlers that are used in the Events Manager of Biztalk RFID. The handlers were implemented in C# and made use of the Biztalk RFID API (Application Programming Interface).

The first handler applied to the events verifies if an event is duplicated. In case of duplication the event is discarded and the execution pipeline is stopped.

The next handler verifies the existence of a tag id associated to a train in the data storage. If it doesn´t exist the pipeline is also stopped.

The last handler to be executed is the most important and where all the logic of the application is executed. This handler as the following objectives:

- Update the location information of the train
- Update the timestamp of the detections in the train and reader
- Store the detection
- Calculate and store the travels

The location update of the train is made based on the reader id contained in the Tag Read Event. It is then possible to lookup in the database for the respective reader and with the reader metadata it is possible to know the location of the train.

In order to store the detection and the calculation of the travels the manager should create a virtual mapping of the readers network. This handler creates a graph [35], based on the information contained in the Reader Links and Readers tables. In this graph each reader is a node and each reader link in the table is a link between the nodes. The next figure as an example of a graph generated that contains four readers connected between them in a circular configuration.

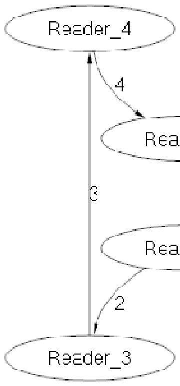


**Figure 8 - Graph example**

Before the insertion of a detection this handler verifies if it is a valid detection. A detection is only considered valid if the last detection made to a train was made in a reader that contains a direct link to the reader responsible for the actual detection.

If a detection is not valid, there is a need to infer the detections that should have been made between the last and the current detection. To infer the detections the Dijkstra [36] algorithm is used. This way the platform will always have all the detections that should have been made even in case of hardware error. However this algorithm only returns the shortest path between two nodes, so if there are more than two paths between two nodes only the detections in the shortest path are inferred, which may not correspond to the real path traveled by a train.

The travels are calculated based on the detections stored in the database. For each new detection this handler will store travels that include the past detections for the same train and the actual detection. If a travel is calculated with inferred detections, the result will be an inferred travel.

#### 4.1.3. Remote Communication Interface and Replication Manager

Web Services [37] where the technology used to implement both Remote Communication Interfaces. The methods implemented are:

Collecting Data System:

- Insert Reader
- Modify Reader
- Remove Reader
- Insert Train
- Modify Train
- Remove Train
- Insert Reader Link
- Remove Reader Link

Client:

- Insert Travel

The Replication Manager was implemented in C#. This manager is a process that runs along with Biztalk RFID. The replication mechanism developed only allows one client to simplify the prototype. The replication manager periodically checks the database for new travels, and if new travels are detected they are replicated to the client database. This replication is executed through the Insert Travel method implemented in the Remote Communication Interface of the client.

### 4.2. Client

The data model of the client is equal to the one in the data collecting system, since the information stored in the client is only a replica from the information in the data collecting system.

The graphic user interface was implemented with C#. This interface displays the information stored in the client database and allows the user to invoke the methods implemented in the data collecting system Web Service. This GUI is composed by four views, travels, trains, readers and reader links that can be navigated through a toolbar with shortcuts for them. Each view contains forms and visual information regarding the type of object they deal with. The next figure contains the travels view.

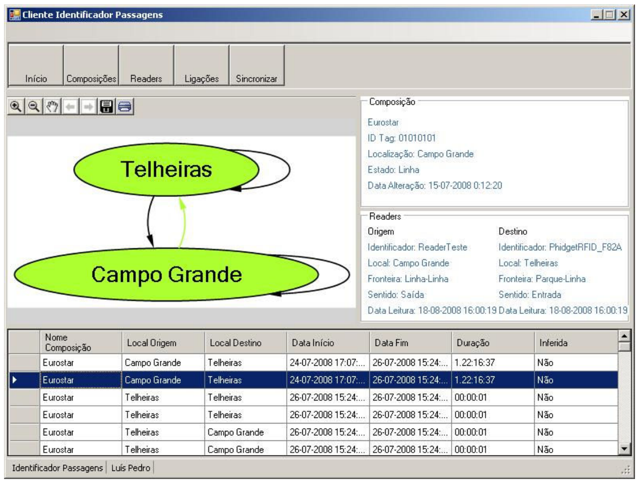


**Figure 9 - Graphic user interface**

All the views contain a graph [38] that represents the places where an entity is located, or in the case of travels to identify the departure and arrival location.

## 5. TESTS

The tests run were mainly functional due to the nature of the functional prototype. The tests scenario is the same as the described in the implementation section.

The integration of the readers with the data collecting system was tested by connecting the readers to the USB ports of the computer running the data collecting system. The result was the expected, and Biztalk generated the unique identifiers for each reader. The client was tested through the insertion of several data about trains, readers and reader links to configure the system. All the data was inserted correctly, which proves that the implemented Web Services also have the desired functionality. To test the tracking of the trains the scale model trains where used. With the readers installed in key locations, it was possible to detect the train movements as expected.

The storage of the detections and travels calculations worked as they should. In order to test the travels calculation algorithm, built in mechanisms from Biztalk RFID to simulate events were used. Travels where calculated correctly with the injection of events that simulated out of order, and in order train detections.

In order to verify if the data collecting system supports the data input that will be delivered in the real system, a capacity test was executed. In this test the throughput of Biztalk RFID [39] was measured along with the latency to execute the event processing pipeline. The procedure of this test consisted in the injection of several tags per second in the system and the measurement of the desired variables. The test was repeated several times for different tags per second rates. The next table shows the results of this test.

| Tags per Second | *Throughput* | Latency (in ms) |
|---|---|---|
| 10 | 10 | 14 |
| 20 | 20 | 24 |
| 30 | 30 | 32 |
| 40 | 40 | 58 |
| 50 | 50 | 79 |
| 60 | 60 | 123 |
| 70 | 70 | 242 |
| 80 | 80 | 323 |
| 90 | 90 | 400 |
| 100 | 100 | 550 |

**Table 1 - Capacity test results**

The results shown that for this range of tags per second the throughput is always the maximum. It is also noticeable that with the growth of tags per second the latency of processing each tag also grows.

## 6. CONCLUSIONS

This works presents the conception, development and testing of an RFID platform that makes the tracking of trains in the Lisbon Metro.

A system architecture that solves the posed problem is presented. The use of active RFID is the most adequate to make the tracking, and a configuration with readers installed in tracks and tags mounted in both laterals is the most suitable for travels monitoring. This work also presents a data collecting system that meets the requirements and is responsible for the RFID equipment monitoring and configuration, and data storage and data analysis. It is also presented a client specification that can interact with the collecting data system.

The presented system architecture was implemented in a laboratorial environment, and the result was a functional prototype. This prototype makes use of scale model trains to simulate the Metro network and passive RFID to track the scale trains. In order to implement the data collecting system Biztalk RFID was used, which proved to be adequate for the posed problem.

The several functional tests executed to the implemented prototype demonstrate the proof of the presented concepts. The platform has shown the right behavior and the integration of the used technologies is possible. It was also executed a capacity test to the platform, and the results shown that it supports the delivered input rates.

### 6.1. Future Work

The use of a centralized data collecting system may pose service availability issues, so there is a need to study redundancy mechanisms for this platform.

In order to understand the impact of a communications network between the readers and the data collecting system there is a need to make a study of the Metro communications network.

There is a need to buy the equipment presented in the system architecture and to implement the mechanisms to allow the communication between those readers and the data collecting system.

In the future this platform will provide information to a drivers schedule planning tool, so there is a need to specify the information structure and develop the integration module.

## 7. REFERENCES


[1]. **John Curtin, Robert J. Kauffman, Frederick J. Riggins.** Making the 'MOST' out of RFID technology: a research agenda for the study of the adoption, usage and impact of RFID. *Springer Science Business Media.* 21 de Abril 2007.

[2]. **Harrison, Steve Hodges & Mark.** *Demystifying RFID: Principles & Practicalities.* 1 de Outubro de 2003.

[3]. **Haver, Torstein.** Security and Privacy in RFID Applications. *Master of Science in Communication Technology.* Junho 2006.

[4]. **Microsoft.** *The Bussiness Value of RFID.* Janeiro 2006.

[5]. **Piçarra, Francisco Miguel de Sousa Gomes Teixeira e Sérgio Manuel Maçoas.** Estado da Arte em RFID. *Comunidade RFID Portugal.* [Online] Outubro 12, 2007. http://www.portalrfid.net/.

[6]. **Mark L. McKelvin, Jr., Mitchel L. Williams, Mitchel L. Williams.** *Integrated Radio Frequency Identification and Wireless Sensor Network Architecture for Automated Inventory Management and Tracking Applications.* Outubro 2005.

[7]. **Group, Metro.** *RFID - a key technology with great potential for innovation - experiences in trade and retail.* 2005.

[8]. **Guitton, Antoine.** The Value of RFID in Transportation. *Massachussets Institute of Technology* . Junho 2004.

[9]. **Transcore.** Rail Mass Transit. *Transcore.* [Online] [Cited: 2 11, 2008.] http://www.transcore.com/.

[10]. **Tagmaster.** Tagmaster. *Tagmaster.* [Online] [Cited: 2 11, 2008.] http://tagmaster.com/.

[11]. **Q-Free.** *Multilane Free Flow System.* s.l. : Q-Free ASA, 2005.

[12]. —. *Stockholm Congestion Tax Charging System.* s.l. : Q-Free ASA, 2006.

[13]. —. *Sydney Cross City Tunnel (SCCT) Project.* s.l. : Q-Free ASA, 2006.

[14]. **Transcore.** WORLD OF EXPERIENCE. *Transcore.* [Online] [Cited: 22 11, 2008.] http://www.transcore.com/.

[15]. —. RFID Products and Vehicle Tracking Systems. *Transcore.* [Online] [Cited: 2 11, 2008.] http://www.transcore.com/.

[16]. **Tagmaster.** High Performance RFID for Railway Solutions. *Tagmaster.* [Online] [Cited: 2 11, 2008.] www.tagmaster.com.

[17]. **Pahlavan, Kaveh.** *Principles of Wireless Networks: A Unified Approach.* s.l. : Prentice Hall, 2002.

[18]. **Schiller, Jochen.** *Mobile Communications.* s.l. : Addison-Wesley, 2003.

[19]. **Railway, Technology.** *Railway Technology.* [Online] [Cited: 2 11, 2008.] http://www.railway-technology.com/.

[20]. **Tagmaster.** TagMaster RFID system for automatic change of talkgroups in the London Underground. [Online] [Cited: Janeiro 15, 2008.] http://www.tagmaster.com.

[21]. —. TagMaster RFID for Automatic Vehicle Identification in the Hamburg Metro. [Online] [Cited: Janeiro 15, 2008.] http://www.tagmaster.com.

[22]. —. A joint solution for wheel flat detection and axle load measurement. [Online] Janeiro 15, 2008. www.tagmaster.com.

[23]. —. *LR-3HD reader Product Sheet.* 2006.

[24]. —. *S1456 MarkTag HDS brochure.* s.l. : Tagmaster, 2006.

[25]. **Phidgets.** *1023 - PhidgetRFID Product Manual.* s.l. : Phidgets, 2008.

[26]. —. Phidgets - Products for USB sensing and control. [Online] [Cited: Maio 15, 2008.] www.phidgets.com.

[27]. **Marin, EM MICROELECTRONIC.** *EM4102 Read Only Contactless Identification Device.* s.l. : Marin SA.

[28]. **Phidgets.** Phidgets Programming Manual. *Phidgets.* [Online] Maio 15, 2008. [Cited: Junho 1, 2008.] http://www.phidgets.com/.

[29]. —. A Single RFID Reader in Visual C#. *Phidgets.* [Online] Junho 1, 2008. [Cited: Julho 2, 2008.] http://www.phidgets.com/.

[30]. **Microsoft.** Biztalk RFID Architecture. [Online] [Cited: Março 4, 2008.] http://msdn.microsoft.com/en-us/library/bb749746.aspx.

[31]. —. Microsoft Biztalk Server RFID. [Online] [Cited: Março 4, 2008.] http://www.microsoft.com/biztalk/en/us/rfid.aspx.

[32]. —. Creating an RFID Device Provider. [Online] Maio 10, 2008. http://www.microsoft.com/downloads/.

[33]. **Cruz, Irving De La.** Biztalk RFID - Device Deployment Kit for Phidgets Devices. [Online] Maio 12, 2007. [Cited: Junho 2, 2008.] http://blogs.msdn.com/irvingd/.

[34]. **Microsoft.** Sql Server 2005. [Online] [Cited: Setembro 20, 2008.] http://www.microsoft.com/sql/default2.mspx.

[35]. **Codeplex.** Quickgraph. *Quickgraph.* [Online] Code Plex. [Cited: Setembro 1, 2008.] http://www.codeplex.com/quickgraph/.

[36]. **Keshav, Srinivasan.** *An Engineering Approach to Computer Networking.* s.l. : Addison Wesley, 2003.

[37]. **Gustavo Alonso, Fabio Casati, Harumi Kuno, Vijay Machiraju.** *Web Services: Concepts, Architecture and Applications.* s.l. : Springer, 2004.

[38]. **Microsoft.** GLEE. [Online] [Cited: Setembro 20, 2008.] http://research.microsoft.com/research/downloads/Details/c927728f-8872-4826-80ee-ecb842d10371/Details.aspx.

[39]. —. BizTalk RFID: Capacity Planning and Performance Tuning . [Online] [Cited: Setembro 21, 2008.] http://msdn.microsoft.com/en-us/library/bb870305.aspx.