

Pipe-DEPCAS: A Middleware Solution for EPC-RFID Data Acquisition Systems

Carlos Cerrada, Ismael Abad, José Antonio Cerrada and Vicente Dies

Departamento de Ingeniería de Software y Sistemas Informáticos
Escuela Técnica Superior de Ingeniería Informática, UNED
C/ Juan del Rosal 16, 28040 Madrid, Spain

Abstract. The huge growth of the RFID devices market, such as EPC (Electronic Product Code) elements in the most varied areas of the industry and the services, joined to the new advantages that every day are discovered on their intensive use, have provoked the immediate attention of multitude of HW, SW and custom solutions providers. The heterogeneous applications in which these devices can be used makes that the methods and develops for their management and control differ from the classical ones known till now. In this type of applications it is necessary to include some mechanism of acquisition and management of the radio frequency information. In this article a possible solution for this question is presented. It represents a middleware solution based on a *pipeline and filter* architecture that brings together the suitable proportion of flexibility, power of calculation, connectivity and simplicity of use to achieve suitable solutions for automation, management and control of goods and services. It has been developed in a more generic framework, that we call DEPCAS (Data EPC Acquisition System), and it is denoted as Pipe-DEPCAS. This specific solution contributes in this field with the best practices of design at the time that it assures an effective control of cost on having used standards of opened code and very known, simple and refined architectures. The paper describes DEPCAS and Pipe-DEPCAS new concepts and developments.

1 Introduction

The extended use of information contained in the RFID devices [5] allows to design and define platforms and systems of information management for the most diverse applications. In particular, the information inserted in low cost RFID devices that can be generated from the standard EPC (Electronic Product Code) allows having systems that use the auto identification in varied environments and with very diverse applications.

Existing software approaches that handle this kind of information can be grouped in two types [2]. In one side, there are software products supplied by the manufacturers of the reading devices. In the other, there are software solutions developed by the big software companies to fulfill the standard definitions of the architecture EPC.

In the first group, every supplier of tag reader devices has developed, achieving more or less functionality, the software that allows the management of their devices.

The following companies, among others, are involved in this approach and have their own software:

- **Symbol**. Company dedicated to everything related with tags, readers, printers, etc.. They have centered their efforts on the area of the EPC, after the absorption of the company Matrics.
- **Samsys**. They only have EPC-type devices, readers and antennas, and have own specific software (RAPID).
- **Alien Technologies**. They work with three EPC elements: antenna, readers and tags. They have their own software.
- **Checkpoint**: Company dedicated to complete systems of distribution, labeling, etc. With respect to EPC, they do not have individual products, they only customize complete engineering and they have laboratories of certification.

On the other hand, there exist other companies that provide from particular parts to custom solutions. Let's do a brief revision:

- **Sun Microsystems**. They have their own EPC platform named *Sun Java System RFID*, which follows the basic architecture of the EPCGlobal Inc., with an orientation of being B2B software. The architecture [6] includes support to acquire information at least of the following EPC readers: ThinMagic Mercury, Sensor-Matic, FEIG, Zebra.
- **SamSys, Alien, AWID, Printronix**. They have a kernel for services support based on the J2EE, and the repository based on ORACLE.
- **Microsoft**. They have tried from the beginning to get involved in everything related to the development of software for RFID, the EPC proposals, etc. The first prototype has been developed in Denmark for a local SCM (Supply Chain Management) called KIM. They have, in the Redmond's head office, a group of hundred persons being working at the topic, and they name it *RFID middleware*. It integrates SQLServer, BizTalk Server, and everything under the .NET environment. They have foreseen to include it in the ERP (Axapta 4.0), and in the MBS (Navision 5.0 and Great Plains 9.0). The software is not available.
- **IBM**. They have developed a prototype for METRO Group in Germany, based on WebSphere (therefore with Java virtual machine, DB2 database, Workplace Client Technology Micro Edition, etc.) and with compatibility with the Alien readers.
- **UCLA**. This American university has produced, with .NET technology, the product WinRFID [8]. This product is commercialized under the formula of consultancy and as key in hand solution. They also commercialize wireless solutions under the WinMEC mark.
- **TAGWARE**. This company has produced, with Java technology, the product "Tagware" [7]. This product is commercialized under the formula of consultancy and as key in hand solution.

Another classification can be done for the existing RFID software when it is not linked to the EPC standard. In this case three groups can be distinguished:

- **GROUP 1**: Companies purely dedicated to RFID. In this group are companies like ConnecTerra(now in BEA Systems), GlobeRanger and OATSystem.
- **GROUP 2**: Distribution of applications companies, as Provia Software, Handcuff Associates, RedPrairie or SAP. UCLA, TAGWARE.

- **GROUP 3:** The giant companies, as Sun, Microsoft, IBM or ORACLE.

Each of these groups would dedicate to a functional part of the RFID middleware. The Group 1 would devote to the management and physical reading of the devices. The Group 2 would centre on integrating and selling closed packages with RFID information. The Group 3 would devote to the management, integration and development of the RFID information in the whole process. It might appear another group exclusively dedicated to the data, where one might place to ORACLE, Sybase, etc., that have their generic middleware and that, in this case, would be applied for RFID information.

Our approach will be global and it would be more close to the Group 2 philosophy, i.e., to develop the software and then to present it as a custom solution. This approach seems to be the most ambitious from the research point of view since the implementation of a middleware solution, though relatively simple, it is not exempt from multiple details and factors to bearing in mind (communications, software, networks connectivity, radio electric interferences, connection to databases, etc.).

2 DEPCAS Objectives and Functional Specifications

As it has been stated, we are concerned with defining and developing a system as middleware solution for acquisition and management of the information RFID-EPC. In forward, we will refer to this system as DEPCAS (Data EPC Acquisition System). This section is dedicated to describe the objectives of such system and to present its functional specifications.

By definition, any software that can be developed to build DEPCAS must observe the following generic objectives:

- It must fit with a level of middleware. We do not try to construct generic software for acquisition of EPC information for all the possible readers with all the particular characteristics of each one of them. Neither is it a question of solving software for EPCglobal Network Architecture information nor of EPC-associated business information (General Identifier: General Manager Number, Object Class and Serial Number). It is a question of having the generic aptitude to transmit the EPC information to the applications of business.
- It must be configurable with respect to the function of the business in which the EPC information is applied.
- It must be configurable with respect to the physical EPC architecture chosen, independent from the make of readers and antennas used.
- It must be as opened as possible concerning to platforms or software components required for its functioning.

Any DEPCAS development must include, at least, the following set of functionalities:

- Communication with EPC physical devices at a very elemental level of protocol.
- Basic processing of the information acquired from them.
- Storage capabilities for configuration-level information concerning the DEPCAS model, both of static and dynamic nature.

- Basic tools allowing the analysis and showing of some EPC information of interest that can be generated in real time.
- Send/receive configuration information about the DEPCAS model.
- Send/receive business sensitive information generated/required by DEPCAS to/from other ERP systems.

Fig. 1 shows a first approach of the functional architecture of DEPCAS. Notice that the above mentioned functionalities are schematically represented.

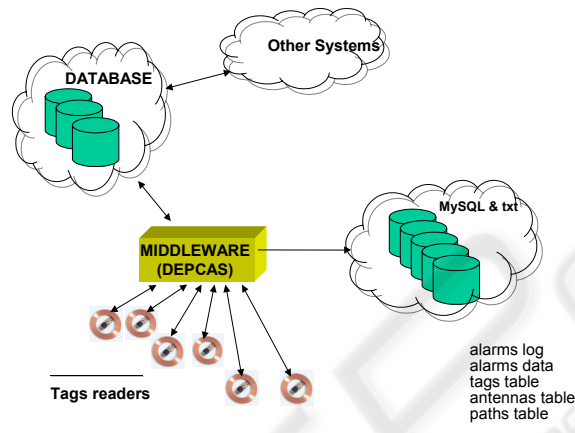


Fig. 1. Functional architecture of DEPCAS.

3 Proposed Solution: Pipe-DEPCAS

We have developed Pipe-DEPCAS as a specific software solution for above defined DEPCAS proposal. Pipe-DEPCAS is middleware solution based on the *pipeline and filter* architecture [4]. It has been designed to simplify the management of the information contained inside the RFID devices used by any company in its business processes.

The approach of this solution is simple and coherent with DEPCAS objectives mentioned in the previous section: to provide an autonomous platform for interfacing with the RFID readers, containing processes for showing the read information, and communicating to the corporate information system.

Pipe-DEPCAS supports very complex configurations of systems, which can need multiple interfaces for its management, complex flows of logic of control, and can be distributed along very diverse platforms. These configurations can be constructed without using traditional methods of programming, just by using (*chaining*) predefined modules (*filters*) on a *pipeline* process.

Pipe-DEPCAS can be extended by the users with their own source code, in order to add new functionalities. They also can be included in the application as new objects in the Java virtual machine.

Fig. 2 shows a more detailed scheme of the functional architecture of Pipe-DEPCAS. Some of its features are detailed in the following paragraphs.

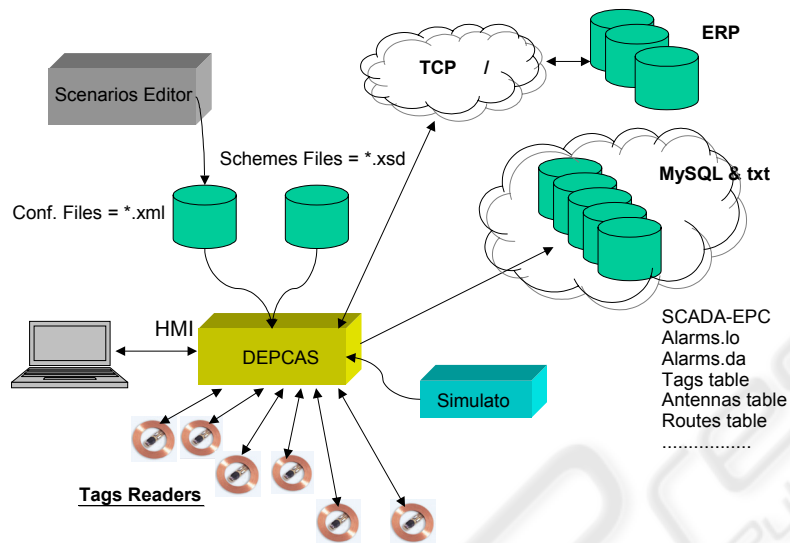


Fig. 2. Detailed scheme of the Pipe-DEPCAS architecture design.

3.1 Principles of the Used Architecture

The basic principle of the pipeline & filter architecture is to use a single data flow [3], in a relatively simple format, that crosses several processes. On it, every process transforms the content of the data somehow, data is continuously inputting to the pipeline (entrance to the pipe), processes (chain of links) are executed concurrently. The architecture is very flexible because, in the most of the cases, software components can be replaced or modified very quickly and because it is possible to insert new components in an easy mode.

3.2 Pipe-DEPCAS Kernel Description

The kernel of Pipe-DEPCAS is a database in the memory organized in tables to store the following type of information:

- Configuration of the system (static)
- Specific functional modeling for what the system is being used (static)
- Data acquired by the EPC readers (dynamic)
- Dynamic information of the stored components

3.3 Topological Workflow in Pipeline & Filter Architecture

The information in the pipeline & filter architecture is processed in a unidirectional way, as can be seen in the example of Fig. 3. This figure shows how is processed a specific RFID event, but it works the same for any of the general purpose developed events. Notice that in this architecture each vertical filter adds incremental processing to the circulating information. Some of the terms used in such figure mean:

- Scene-type: physical environment studied
- Chain: set of processes running in a given pipeline
- Link: each of the generic function, control or sub process used in a chain
- Filter: application of a link with specific parameters

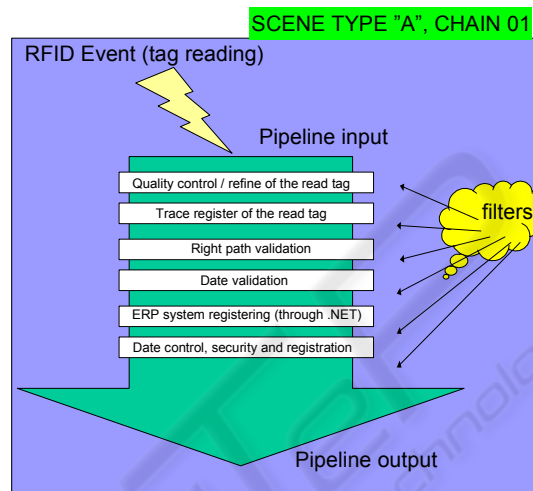


Fig. 3. Data workflow in a chain of Pipe-DEPCAS.

3.4 Some System Model Details

The pipeline & filter architecture gives a support structure to systems that process data flows. This is the case of the Pipe-DEPCAS system, which flow of information between the RFID tag readers and the specific purpose processing systems appears in Fig. 4. Every process' step is encapsulated in a so called filter component. The data is transferred across pipes between adjacent filters. The recombination of filters allows constructing families of related filters.

In this structure, the context consists on programs that must process data flows of small size, in a certain sequence.

Come to this point, and summarizing, we can say that Pipe-DEPCAS is a multi-platform system (UNIX, MS-WINDOWS) that executes programs of configuration (files .XML) that have been validated (compiled) syntactically and formally against files of schemes (*.XSD). The programs contain software modules ("chains") that will be executed as process threads that, in turn, are codified by means of parameterized sentences or commands (links).

The data model used for the links is based on JAVA classes, which are obtained by means of a JAVA Code Generator (as JAXB, BEAVER or XMPSpy 2006) from the file scheme *.XSD. This way of working, and the mentioned file types, has been also represented in Fig. 2.

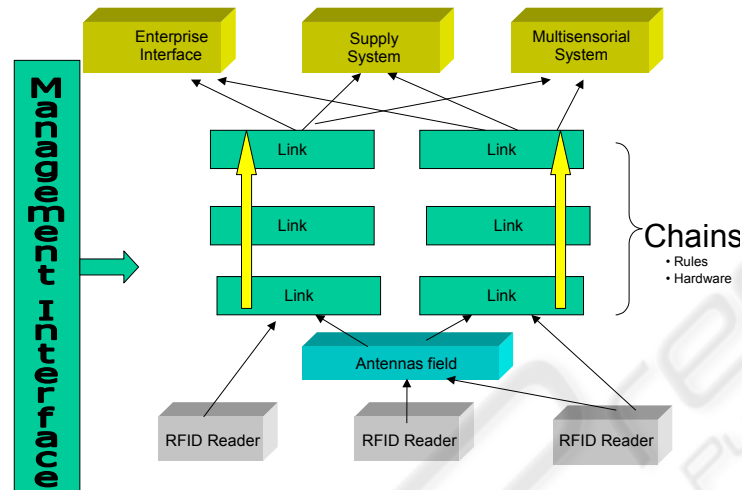


Fig. 4: Data workflow among RFID tag-readers and dedicated systems in Pipe-DEPCAS.

3.5 Characteristics of the Developed System

Finally, it must be said that capabilities of the middleware product developed, known as Pipe-DEPCAS, can be grouped in different documental sections. This is made so in order to improve the documentation organization, as well as to facilitate the localization of a given special feature. These groups correspond to the following items:

- Fundamentals of the implemented architecture
- Operating system
- Configuring the overall system
- Configuration of the Timers
- Configuration of the Readers
- Connecting to external interfaces
- User interface
- READER Simulator capabilities
- Management of the control interface
- Logging configuration
- Application of Pipe-DEPCAS to particular Scene-types
- Extending Pipe-DEPCAS inside the ERP systems

For more detailed information about each of these topics consult [1], where an extensive description of Pipe-DEPCAS features is included.

4 Conclusions

This work is concerned with the definition and development of a system as middleware solution for acquisition and management of the information RFID-EPC. The specification for this generic system, whose acronym is DEPCAS, has been stated. And specific solution based on the pipeline and filter software architecture has been developed, and the paper is dedicated to present its mains characteristics of this product.

The specific solution presented, denoted as Pipe-DEPCAS, contributes to the RFID middleware software development with the best practices of design at the time that it assures an effective control of cost on having used standards of opened code and very known, simple and refined architectures.

At present time we are improving our work trying to define and implement SCADA based software architecture, instead of the pipeline and filter solution shown here. As it is well known, SCADA software architecture is a conventional one for analogical and digital data acquisition systems in industrial environments. Our purpose is to extend that concept to the data EPC information systems.

Acknowledgements

This research has been carried out under contract with the Spanish CICYT through the DPI2005-03769 project.

References

1. V. Dies: "Base de datos de apoyo a un sistema multisensorial para el reconocimiento y localización de objetos empleando tag's RFID" Internal Report, November 2006.
2. "Draft paper on the Characteristics of RFID Systems", www.aimglobal.org
3. Mariño P., C. Sigüenza, J. Nogueira, F. Poza y M. Domínguez, An event driven software architecture for enterprise-wide data source integration. In Proc. Conf. on Information Technology: Coding and Computing, Information Technology, pp. 140–145, Las Vegas, Nevada, USA (2000).
4. Mary Shaw and David Garlan. Software Architecture: Perspectives on an Emerging Discipline, Prentice-Hall, 1996.
5. R. Das "RFID Explained", IDtechnex White Paper, www.idtechex.com, 2002
6. Sun MircoSystems JAVA suite www.sunmicrosystems.com
7. TagsWare Agile RFID Solutions www.tagsware.com
8. UCLA <http://wireless.ucla.edu/rfid/winrfid>
9. <http://www.epcglobalinc.org/>