Ambient RFID Services Infrastructure & RFID Deployment in Wood industry

Eddy Bajic, André Thomas

Research Center for Automatic Control, CRAN-CNRS 7039
Nancy University,
Faculté des Sciences, BP 239
54506 Vandoeuvre les Nancy Cedex, France
Tél : +33 3 83 68 44 42
{Eddy.Bajic | Andre.Thomas}@cran.uhp-nancy.fr

Abstract: In this paper, we present academic research activities and industrial technology transfer conducted at CRAN-CNRS UMR 7039 Laboratory concerning on one hand high level services environment definition based on low cost RFID Tagged product for smart product concept achievement in an ambient service architecture based on standard IP technologies, and on a second hand RFID deployment in wood industry for logistic and product driven factory management.

1. INTRODUCTION

The presented works rely on research activities carried out at the Research Centre for Automatic Control (CRAN - CNRS 7039), within the Ambient Production System group SYMPA and the TRACILOG[1] team, which aim to develop product-centric approaches in the fields of industrial production and management activities where on one hand intelligence is given to products by means of RFID technologies helped by ambient services systems architecture and on a second hand control and decision making model-based methods are conceived and evaluated.

Industrial Application cases of RFID are conducted for implementing traceability and improving supply chain management in wood and fiber industries. One of the objectives of the CRAN is to provide a benchmarking tool enabling a comparative study of different control systems, like Holonic Manufacturing Systems (HMS) or Product-Driven Manufacturing systems. Several investigations in the fiber industry area, and especially in furniture industry, led to highlight the following context characteristics.

First, production systems, including physical workshop and the planning system, has been developed to answer to a mass production context, that is to say to produce large quantities per lot for few part numbers. Secondly, clients ask for more and more customized products, at lowest cost and in shortening delays. To face these growing requirements, companies deploy several new tools, like autonomous teams management or simulation tools, in order to improve flexibility and reactivity of the production system.

Today, new paradigms concerning intelligent products or ambient and communicating products appear. In a mass production context, this kind of concept must be evaluated according to global performance indicator such as throughput, for example. Consequently, in a HMS context the product has become a kind of gain inductor.

In this paper, we will present first the communicating object concept applied to low cost RFID tagged product and then the enhancement of product/ process/operator relationship from simple data exchange to high level service providing and invocation at the product level. An Ambient services architecture is defined allowing intelligent product services capabilities supported by UPnP (Universal Plug and Play) technology to develop product services modeling and components design to provide intelligent product design.

In a third part we present a RFID deployment application case in wood industry. The studied system is a panel board, ready-to-assemble furniture production system. In this kind of company with high production volume is to maintain as low as possible the WIP level, as high as possible the load of the bottleneck and the service level. To achieve to these objectives a Product information based control project was released. In the part 4 we present the modeling methodology and the architecture implemented. Then we describe the simulation tool developed for the system control.

Finally, we conclude and invoke open issues and new researches at CRAN lab.

[1] Traceability and management of the supply chain in fibers industries (wood, paper, textile)

2. RFID AND SERVICES FOR AMBIENT INTELLIGENCE SYSTEM

Increasing requirements of reactivity in the supply chain topic are observed among product, processes and clients across the product lifecycle. In this context, the interactions between processes, operators and product, beginning with manufacturing level until its use, require more information and automated intelligent exchanges between partners, in a sure and relatively quick way. Specifically, the requirements that emerge in the supply chain are associated to product customization, traceability and information management along its lifecycle, and, in general, all kind of services related to the product lifecycle using Internet technologies. In order to respond to these new requirements we propose an approach by considering a physical object as an active actor managing its evolution in the phases of its lifecycle, cooperating with multiple actors in the supply chain (supplier, producer, distributor, and consumer) to implement an intelligent service connectivity inside/beside the product figuring a trend toward an intelligent product behavior.

Intelligent Product and Smart Object :

Conceptually, the terms of intelligent product and smart product [1] are synonyms indicating the capability of an object to communicate with its active environment, and interact with its users or the other objects. A smart object is able to acquire, to receive and to distribute information in a near or distant environment, and is able to carry out actions on its own initiative. According to [2], a smart object is aware of its environment and is able to perceive its surroundings through sensors, work with peers using short-range wireless communication technologies, and provide context-aware services to users in smart environments. The new paradigm of smart object provides an ability to embed new capabilities into object allowing extended access information up to complex services invocation, and interactions from virtually anywhere at any time, potentially transforming the way we live and work in a society of objects.
The automatic identification technology plays an important role in the identification methods. Radio frequency identification (RFID) is a support tool that allows to automate object identification process and to improve operations managements [3]. A RFID system contains electronic labels or tags, tag readers that can read and write data, and a controller that controls the system. By attaching an electronic tag to a physical product, it can be automatically identified and located into the system vicinity. RFID represents the union between the physical product and the virtual product maximizing the use of the information and knowledge along product lifecycle [3].

Ambient services architecture :

In order to develop the interactions between intelligent products and other actors of the production system or the supply chain, the ambient services concept is analyzed and characterized therafter. In ambient services context, we consider the product as a **service provider** or a **service requester**, demanding transparent and associated control mechanisms of all the offered services in its environment [9]. **Ambient service** is an abstract view of a system that provides information management capabilities, processing capabilities and event messages in an ambient network, representing a working domain. A product inserted in a network can appear or disappear in a domain of work [10]. Ambient architecture based on Internet technologies have been developed integrating concepts such as ad-hoc network, mobility management, and service discoverproposed by y. With respect to the ubiquitous concept proposed earlier by [10], a generic Ambient Services Architecture needs the following standard services:

- **Identification:** To know the product unique identity in the ambient network.
- **Localization:** To know where the product is situated in the ambient network.
- **Information brokering:** To identify the information sources allowing the execution of a given service
- **Service Discovery and Registry:** To automatically discover and advertise for services in the ambient network. Each new service provider is broadcasted at the time of its connection to the network
- **Service Invocation:** Process by which a service execution is requested by a client
- **Event notification:** To notify about change in the services variables. Users must first subscribe to event notifications mechanism.

Currently most known services architectures are UPnP, Jini, OSGi, CORBA. UPnP generic architecture rely on Remote Procedure Call model instead of code mobility as do Jini and OSGi which are Java-dependant and thus sensitive to code vulnerability problems. UPnP model is supported by API for Windows and Linux, and makes large use of XML data exchange over IP networks [5].
UPnP is a distributed, open networking architecture [6] that leverages Internet and Web technologies, such as Hypertext Transport Protocol (HTTP), Simple Object Access Protocol (SOAP), Generic Event Notification Architecture (GENA), Simple Service Discovery Protocol (SSDP) and eXtended Mark-up Language (XML). The generic UPnP architecture includes the two following entities: Devices and Control Points. The term **Device**, noted **UPnP(dv),** is used to define a logical container of others devices and services. The **Services** are logical entities providing a specific service to UPnP device network. A service in an UPnP device consists of a state table, a control server and an event server. A state table models the state of the service through state variables at run time and updates them when the state changes. A control server receives actions request, executes them, updates the state table and returns

responses. An event server publishes events to interested subscribers anytime the state of the service changes. Services are controlled by Control Points. A **Control Point**, noted **UPnP(cp)**, is a logical entity that controls Device Points. After devices discovery, a control point retrieves the device description and get a list of associated services; retrieves service descriptions for interesting services; invokes actions to control the service; subscribes to the service's event source. Any time the state of the service changes, the event server will send an event to the control point. It is expected that devices incorporate control point functionality, noted UPnP(dv/cp), to enable true peer-to-peer networking interaction.

Principle of a Service-based relationship :

Once the object or product has entered the ambient architecture, the object is detected by an RFID ambient interface, and then the object is mounted in the ambient services architecture as a service provider entity ubiquitously accessible by any control point system existing. The Physical Object is so transformed in a Device Object providing and performing transparent services. Fig. 1 depicts the basic primitives needed to achieve a Service based relationship between product and actors within a supply chain (Fig.1).

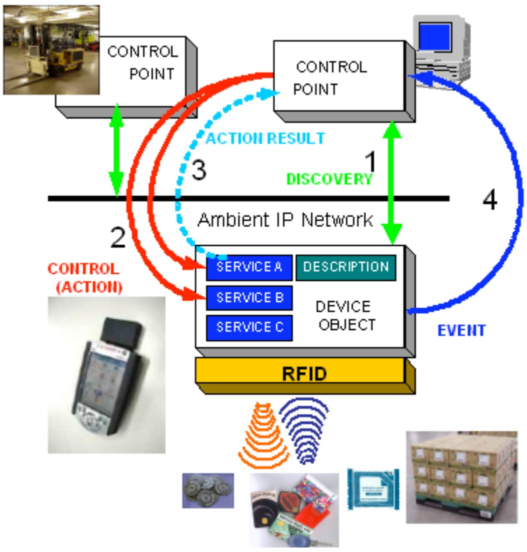

Fig. 1: Principle of a Service based relationship between product and actors in a supply chain

Point 1. The device object announces the IP network participants of its presence. Control point entities can also discover automatically new mounted device points. Device description files and service description files are transmitted by the device point to inform the ambient environment of its capabilities.

Point 2. Any control point in the ambient network can invoke the device services.

Point 3. When the service is requested, the device-point execute the corresponding actions and returns acknowledgement or actions results to the initiator.

Point 4. A device object figuring a Smart Object behavior can performs automatic eventing to the ambient partners so to performs reactivity from the object to the system.

Interaction Model :

The general mechanism used by a Smart Object acting as a control point in order to choose one resource between several, is represented in Fig. 2 Inspired from the *Contract Net* interaction protocol, standardized by FIPA, this interaction begins with the service request (the use of a resource, transport, etc.) sent by the service requester (the active Smart Object) to all available service providers (M) in the ambient UPnP IP network. The service providers respond by asking for the desired characteristics of the requested service (storage conditions, transport conditions, etc.). The providers, which offer the desired service and can answer to the request (P) will inform the requester of their availability, and propose the execution of the service (meeting, execution time, etc…). By this way, the service requester can choose amongst the providers the most adequate one for its needs. After the service execution is done, the service provider will send a response message with the execution status.

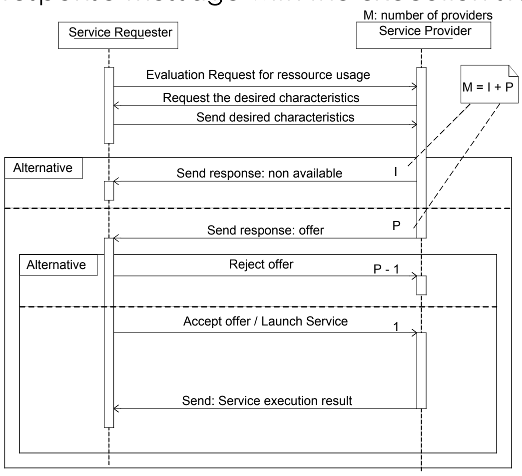

Fig. 2: UML Sequence diagram describes the action invocation between an active communication object and another communicating object

Methodological proposition :

A methodological offer is proposed so as to create an effective integration between a physical product and the interactive world of the information technologies within industrial processes and logistics [11]. To present the conceptual solution three generic cases are considered:

- **Case One - UPnP assisted Passive Object:** Case one (Fig. 3) represents an **object** carrying an electronic tag, which is managed by an UPnP(dv). Process begins with the automatic identification of the tagged object by an UPnP(dv). Then services associated to the identified product type are mounted in the device memory by means of XML files uploaded from local memory or remote database. A XML file summarizes the information about services, including actions and state variables. The services now available in the UPnP(dv) represent a virtual image of the product parameterized by the information stored in the tag or in a remote information system [6]. At that time, all the services associated to the product are known and can be remotely called by all UPnP(pc) in the ambient environment [5]. Conceptually, the merger between the **tagged object** and an **UPnP(dv)** plugged in an UPnP network forms an entity called **UPnP assisted passive object**, representing a low-cost solution with a versatile technological integration.

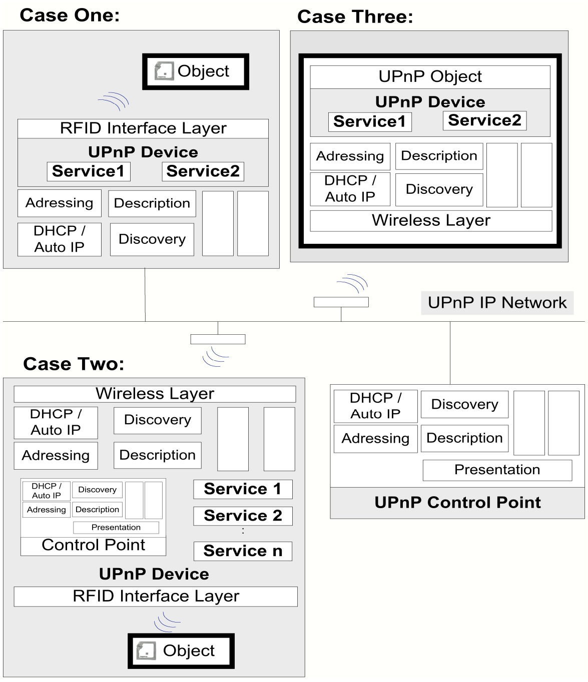

Fig. 3: Ambient service based UPnP architecture integrating RFID smart object.

Case Two - UPnP assisted Active Object: the UPnP(dv) is enriched with Control Point capabilities, UPnP(dv/cp). The **tagged object** and an **UPnP(dv/cp)** plugged in an UPnP network thus form a new entity called **UPnP assisted active object**, or for short "**service requester/provider"**. This active object entity offers services and has the ability to request services to other services provider and thus makes its own decision in the ambient architecture. A software layer in the control point is parameterized by identified information in the tagged product to manage decision-making at product level and corresponding services calls in the UPnP ambient architecture.

- **Case Three - UPnP integrated object:** UPnP device is embedded into the physical product, with no more need of RFID communication. Nowadays, such industrial products are almost non-existent due to its high cost of manufacturing and complexity. Nevertheless, a Mote device (Wireless Sensor network) or a PDA with WiFi and RFID communication capabilities can act as though. Interactions between the actors across the invocation or execution of services represent the key element in ambient services modeling process. Therefore, the characterization of innovative services contributes a significant added value for the supply chain. UML language is used to model services and interactions between products and processes. Class Diagram in Fig. 4, summarizes the relationship between device, services, state variables and actions.

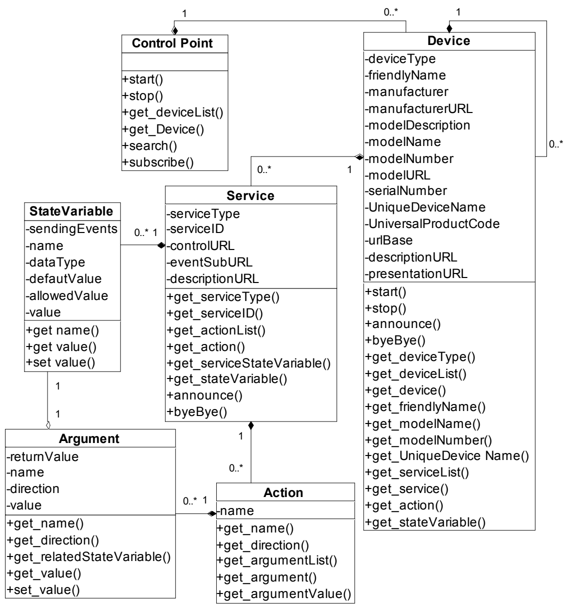

Fig. 4: Class Diagram: Device, Service, Actions and State Variable in UPnP**.**

Static view in the service modeling considers that each UPnP service has a service type Uniform Resource Identifier (URI) that uniquely identifies the service. Every service also has a serviceID URI that uniquely identifies the service among all device's services. Additionally, every service maintains three URLs that provide the information necessary for control points to communicate with services : the ControlURL is where control points post request to control this service; the EventSubURL is where control points post request to subscribe to events; the DescriptionURL tells controls point the location from which they can retrieve the service description document. Each service has zero or more state variable, having a name, a type, and a current value. A state variable also has a set of allowed values. Any of the state variables can trigger events on state changes as determined by the implementer of the service. Every input argument to an action is associated with one of the service's state variable.

Device and Service Description Documents :

Fig. 5 shows a device description document in XML that reflects the logical structure of a device, containing static information about the device. Device description contains a list of all of the services that the device provide. The service list provides the service's type, identifier, URL for retrieving the service description, and a URL for eventing services.

```
<device> <serviceList> <service>
<serviceType>serviceType:v</serviceType>
<serviceId>serviceID</serviceId>
<SCPDURL>URL to service description</SCPDURL>
<controlURL>URL for control</controlURL>
<eventSubURL>URL for eventing</eventSubURL>
</service> declarations for other services
here
</serviceList> </device>
```

Fig. 5 : Device Description Template.

XML service description file document is summarized in Fig. 6, containing detailed information about a service, including the action list supported by the service.

```
<actionList>
<action>  <name>actionName</name>
<argumentList> list arguments </argumentList>
</action> </actionList>
<serviceStateTable>
<stateVariable sendEvents="yes">
<name>variableName</name>
<dataType>variable data type</dataType>
<defaultValue> default value</defaultValue>
</stateVariable> </serviceStatetable>
```

Fig. 6 : Service Description Document.

Service actions typically (Fig.6) have the following forms: query specific state variable, set a device state variable, or tell to the device to perform some kind of task. Service state table is a list of the services state variables. State variables are used to model the state of the service at run time.

A UPnP(cp) starts the subscription process by first retrieving the device description document form the device. The device description document contains an event subscription URL (the <eventSubURL>) and a service identifier (the <serviceId>) for each service provide by the device. Then, to subscribe to events from a particular service, the UPnP(cp) sends a subscription request to the service's event subscription URL. If the service accepts the subscription, the service responds with a unique identifier and a duration for the subscription. UML Sequence Diagram are used to represent how a UPnP(cp) subscribes and receives notifications of states variable changes in a service.

Modeling of interaction between intelligent product and supply chain processes is thus supported by definition of a set of service classes associated to different stages of the product life cycle according to the capabilities of the tagged or smart object. In the following section, meta-services based on standards UPnP services are defined to manage an intelligent interaction in warehousing activity.

3. RFID DEPLOYMENT APPLICATION CASE IN WOOD INDUSTRY

The studied system is a panel board, ready-to-assemble furniture production system. Furniture's manufacturing rely on a composition of three main functions: first, large panels are cut into pieces; secondly, pieces receive some transformations, like drilling or grooving, and then finished pieces are assembled into a package, which will be brought to the final customer through the distribution system.

The cutting operations are particularly awkward to manage because of the important cadence variability. This fact could be explained by the following reasons. The panel price being one of the most significant costs constituting the unit cost, a special process optimizes the raw material use. So as to reduce cutting scraps, the produced quantity varies for each batch:

with the aim of optimizing material use, the production of 400 desks could lead to produce 401 desktop and 404 left bases. The next session for this reference will take the remainder into account, which changes quantities for each component, so cutting plans will be really different. Moreover, the Cut-to-Size productivity curve is non-linear: the operating time to produce 800 pieces is not equal to the double of the time to produce 400 pieces. So the Cut-to-Size tool has variable operating times, which depend on many parameters. Indeed, the cutting plan influences the operating time due to the machine architecture, when the theoretic time, used for scheduling and for raw material ordering, is given by a formula based on constant throughput. The machine is seen as a "black box", but the resulting lead time of this operation varies a lot. Consequently robust global lead time is a problem.

After this cutting process, pieces are manufactured to become finished pieces. The workshop is constituted of several machines, each one able to perform some operations. The variety of different pieces, the significant volume produced and the important number of lots being processed simultaneously generate complexity of physical flow, which makes their control difficult. Consequently robust product mix schedule is also a problem.

In that context, the production management follows traditional MRP2 rules [11]: a Sales and Operation Planning is generated according to commercial forecasts by family of products, and then is disaggregated into Master Production Schedules. The next step consists in computing the Material Requirements Plan and in balancing workloads on work centers by establishing a predictive finite capacity planning for each of them. This step enables to establish raw materials orders and finished furniture's delivery dates, which appears as constraints to be respected for the lower level.

The recently acquired scheduling tool aims at optimizing resources usage rate. But its implementation reveals some imperfections:

- To be pertinent, a scheduling must be based on accurate technical data,
- This scheduling tool does not reflect variability and dynamics of the actual process,
- This scheduling tool does not take into account WIP level and its impact on productivity.

What particularly interests us is the launching on the following drilling and grooving machines. Each has particular features, and is able to accomplish several operations. Consequently, there are many possible operation sequences, and the pieces diversity makes physical flows and their control complex. Moreover, average lot size is about 400 units, which implies several important batches at the same time in the workshop and some quality scraps could be discovered in the decoupling point (St1 in Fig.7) between these two first steps. Consequently the quantity insurance could be still a problem.

The end of the process is the packaging line, where pieces have to be grouped by furniture reference. Of course, all the pieces of the bill of material of furniture have to be completed before packaging can begin.

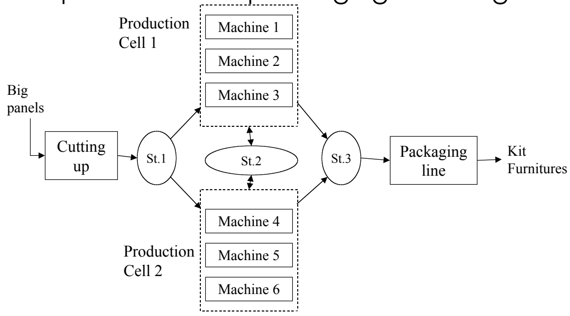

Fig. 7 : Synthetic view of the workshop

Three intermediate stocks enable to buffer physical flow between machines. Those stocks are "rolling stocks", equivalent to drawer stocks. The existence of these stocks leads the management more complex. Indeed, there are several FIFO stacks, and it is only possible to catch the first lot of each drawer.

Consequently, we decided to focus on WIP and lead time control. Studying these two major objectives implies to take the system dynamic into account. In that sense, Product information based control appeared like a really adapted tool to improve physical flow management [8].

4. MODELLING METHODOLOGY

Assessing the impact of synchronizing physical and informational flows needs to model them as distinct flows [12]. In that way, we represent them in two distinct models, that works simultaneously and have to be synchronized. Moreover, the interface standardization enables to exchange different control models with the same physical emulation model [13]. To represent the implementation of RFID technologies, with fixed readers, we will use the notion of synchronization points, which are points where the physical system emits events to update the information system. This event update could launch a decision process that will react by acting on the emulation model after impact measure on performance indicators.

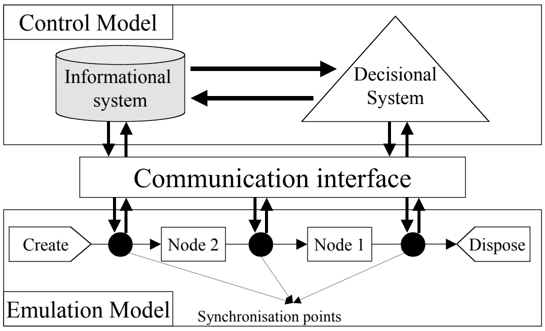

Fig. 8 : Simulation model structure

Emulation model :

The emulation model does not integrate any decision rule, that is to say each point where decision have to be done is converted to a synchronization point: a message, called synchronization event, is sent to the control model, and the execution is paused until a resume order is received. On the synchronization event reception, the control model could compute and send some information or parameters to the emulation model, in order to realize the control of it. For example, in front of a machine, the control model send back the machining program, or at the exit point of a stock, the next step where the pieces batch have to been sent.
According to the system theory, we distinguish three types of transformation: time, space and form, each corresponding to one kind of elements of the system : inventories, transport's and machines. So every type of element is represented by a basic element, that could be tuned function of the physical system it represents (batch size, transfer time…). A basic element is constituted of five components (Fig. 79):

- an entrance label,
- a first synchronization point, called pre-synchronization, in charge of correctly setting up the corresponding resource,
- a macro block representing the process performed by the resource, like in standard simulation model,
- a second synchronization point, called post-synchronization, managing the piece way and launching the next work on the machine,
- An exit point, where the entity is routed to the next step of the process.

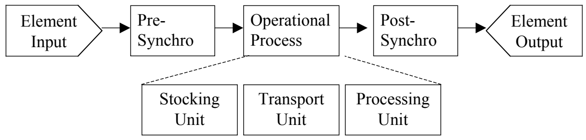

Fig. 9 : Emulation model basic element

The synchronization procedure is quite simple: when a moving entity comes in a synchronization block, a message holding entity and localization identifier is sent to the control system, then the entity is held until an order is sent back by the control system. This procedure is shown on figure 10. The number of event type is finite : we could define one for the pre-synchronization and one for the post-synchronization for each type of resource.
Opposite to usual simulation models, moving entities only hold an unique identifier, which refers to corresponding data in the control model. As a consequence, the emulation model couldn't run without a control model, because entity related data are not stored like attributes of an entity but in a database integrated to the control model.

Control model :

Control model is a discrete event system, which reacts to external events coming from the emulation module, from the user or from any other source (representing lead time, quality, geo-localization lot information). To enable control decisions making, the control model needs a physical world representation and a decision making process. These components could be or not separated.

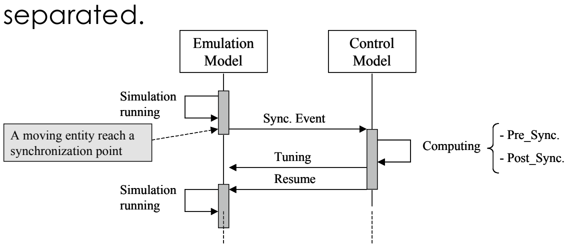

Fig. 7 : The synchronization procedure

To ensure the ability to exchange control model, we defined a synchronization procedure (Fig.10) and a communication interface between emulation and control model: that is to say to standardize messages between emulation and control models. Such a message contains following information's:

- A Synchronization point identifier, identify the resource, type, and synchronization type (pre or post).
- An Entity identifier, pointer on the entity description in the database.

This information enables the control model to update the information system, and then to compute the attempted decision. This decision would lead to a parameterization of the emulation model (route or process tuning), and then a "resume signal" would be send to the emulation model.

Simulation results:

The simulation model has been validated following a two steps process: first, functional properties have been validated, and in a second time, stochastic parameters have been identified using a statistic study. Functional properties integrate:

- Any lot follows the way corresponding to the real machining sequence (routings)
- When more than one lot is available in front of a machine, the schedule is respected
- Machine calendars are observed.

Moreover, some alerts have been developed. These alerts generate a log file each time an error appears, which enable us to track the error.
Machining process being more standard than the cutting one, the characterization of their stochastic parameters has been done analyzing previously recorded times for each machine.
According to a 24 to 48 hours horizon, we were able to show the reliability of our system. Effectively, if the schedule is not or few changed, the forecast is relevant, but if the schedule is completely changed, due to commercial constraints for example, the forecast error grows up quickly (See comparative between real and simulated inventory on Fig. 8).

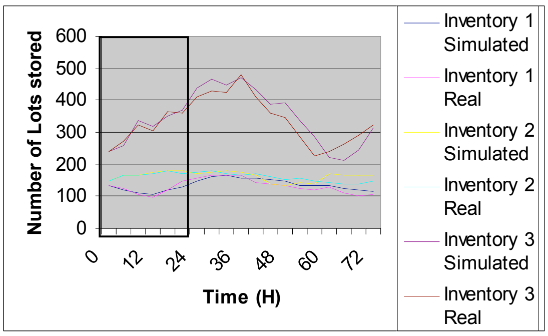

Fig. 8: Real and simulated inventory evolution

Opposite to the Cut-to-size simulation tool, the reverse loop from the simulator to the informational system is manual: that is to say the simulator generates a report, which is analyzed and used by the scheduling operator. This way of establishing a schedule enables it to react in order to maintain the inventory level between efficiency boundaries. The daily use allowed improving the scheduling rules. The results are generated with a period of four hours: the global evolution is presented with curves, like on 5. This global evolution enables us to establish what are the interesting areas and then to obtain the detailed content of WIP to engage some corrective actions.
On the other hand, the workshop simulator has been used like a training tool, what enables to considerably cut back the training time for a new scheduling operator, and to reduce problems while releasing new orders.
The daily use of the workshop simulator let foresee an enlarged use of it, by validating some modifications of control rules before to implement them in the workshop if their impact are relevant. One of the concrete uses of the simulation was to decide how much operators have to be allocated to transport operations, or to choose the best location to install some new stocking structures. Moreover, this initially unforeseen use needs, ,the intervention of a specialist, when quickly educated users could do the daily use.
The impact of major management rules could be studied too. For example, currently, the typical lot size is about four hundred pieces, but the growing need for reactivity incites to reduce this lot size.
But the ability to test some new management rules drove to new needs, more and more complex and fastidious to apply for the model designer. This observation led us to develop a new modeling architecture, separating, on the one hand, the physical controlling system and, on the other hand, the control system, interfaced by an informational system according to the system theory.

5. CONCLUSION AND OPEN ISSUES

This paper has proposed a methodology offering a high level of integration of a physical product into an IP ambient network for developing wireless communication and intelligent interactions with the supply chain processes during product lifecycle. RFID technology permits to identify and seamlessly link the physical product to its extended services for maximizing the use of the information and knowledge along product lifecycle. Ambient services architecture provides high-level standards functionalities for discovering and learning about product's services in a domain of work. As a UPnP entity, the physical product assumes intelligent interactions in an IP ambient network. This RFID/UPnP based object represents an intelligent product that advertises and informs about its capabilities to the actors of the supply chain as also request services from others. The Smart Objects community proposed with the association of RFID and UPnP allows to create reactive and context sensitive environments, in which products are empowered through a digital environment based on services instead of elementary data. As though, the product's capacities are met in a given situation to form the Ambient Intelligence concept. Concerning the introduction of such concepts in industries, a case was presented. This industrial application highlights the interest to develop emulation and control simulation systems to achieve the SC global indicators.

6. REFERENCES

1. Kintzig G., Poulain G., Privat G. and Favennec P (2002) Objets Communicants. Hermès Ed
2. Duda A. (2003). Ambient Networking. Smart Objects Conference 2003, Grenoble, France.
3. McFarlane D., Sarma S., Chirn Jin Lung, Wong C.Y., Ashton K. (2002). The intelligent product in manufacturing control and management. 15th Triennial World Congress, Barcelona, Spain. 2002
4. Loke S.W., Krishnaswamy S., Naing T.T. (2005). Service Domains for Ambient Services: Concept and Experimentation. Mobile Networks and Applications 10, 395-404, 2005.
5 Jeronimo M. and Weast J. (2003). UPnP Design by Example. Intel Press. USA.
6. UPnP Forum (2004), http://www.upnp.org
7. Bajic, E. (2005). Ambient Services Modeling Framework for Intelligent Products. UbiComp 2005. Tokyo.
8. Karkkainen M., Holmstrom J., Framling K. and Artto K. (2003). Intelligent products – a step towards a more effective project delivery chain. Computer in industry 50, pp 141-151.
9. Cea A. and Bajic E. (2004) Ambient Services for Smarts Objects in the Supply Chain Based on RFID and UPnP Technology. IFAC MCPL 2004. Chile.
10. Weiser M. (1991). The computer for the 21st century. Scientific American 265. N°3, 1991. 94-104.
11. Vollman, T., Berry, W., Whybark, D. (1988). Manufacturing Planning and Control Systems, Irwin.
12. Pannequin R., Thomas A. 2004. Centralized Versus Distributed Decisions, an Industrial Case. 11th IFAC Symposium on Information Control Problems in Manufacturing (INCOM'04), Salvador-Bahia, Brésil.
13. EL HAOUZI H., THOMAS A., PETIN JF. – "Contribution to reusability and modularity of Manufacturing Systems Simulation Models: application to distributed control simulation within DFT context3 - special issue of the International Journal of Production Economics – 2007