Location-Aware Business Process Management for Real-time Monitoring of Patient Care Processes

Renaud Bougueng Tchemeube

Thesis submitted to the
Faculty of Graduate and Postdoctoral Studies
in partial fulfillment of the requirements for the degree of

Master of Computer Science

Under the auspices of the Ottawa-Carleton Institute for Computer Science

uOttawa

University of Ottawa
Ottawa, Ontario, Canada
May 2013

Abstract

Long wait times are a global issue in the healthcare sector, particularly in Canada. Despite numerous research findings on wait time management, the issue persists. This is partly because for a given hospital, the data required to conduct wait times analysis is currently scattered across various information systems. Moreover, such data is usually not accurate (because of possible human errors), imprecise and late. The whole situation contributes to the current state of wait times.

This thesis proposes a location-aware business process management system for real-time care process monitoring. More precisely, the system enables an improved visibility of process execution by gathering, as processes execute, accurate and granular process information including wait time measurements. The major contributions of this thesis include an architecture for the system, a prototype taking advantages of commercial real-time location system combined with a business process management system to accurately measure wait times, as well as a case study based on a real cardiology process from an Ontario hospital.

Acknowledgment

First of all, I would like to thank my dear supervisor, Professor Daniel Amyot, for allowing me to pursuit this tremendous opportunity and for guiding me throughout my research during the last two years. He has always made sure I kept learning in face of the multiple challenges of this research. He always ensured I stayed on track, while allowing me to fail in my process of learning. This whole experience has been far richer than what I had envisioned. To him, I am grateful for who I am today.

I also need to thank my team and colleagues, those that I have met in the Osler project, but also those that I have met in my academic life. They always have been supportive in my regard and have made my academic life more enjoyable and fun. We have shared unique experiences together. I particularly would like to thank Alain Mouttham, who has been, really early, a sharp mentor, teaching me about the industrial world, and supporting me in my learning. I also would like to thank Sepideh Bahrani who has broadened my knowledge about a different social world and with whom I have shared many random discussions about life, numerous burger lunches and of course, countless hours of thesis writing sessions. I also need to mention properly the other members of the team that I have been lucky to meet: Shirley Baffoe, Aladdin Baarah, Curtis Naphan, Ruogu Luo, Randy Giffen, Omar Bahy Badreddin, Saeed Behnam, Professor Liam Peyton, Amanpal Singh Bhatia and Gurpreet Dhadda.

I wish to deeply thank the University of Ottawa for enabling this research program, providing facilities, support and the critical environment to its graduate students to effectively conduct research. As a former bachelor in Software Engineering from the University, it feels really much like home.

I would like to thank the NSERC’s Business Intelligence Network and MITACS for providing full financial support for me to have peace of mind while studying in the labs. I also need to thank William Osler Health Centre and IBM Canada for their openness and interest in academic research, their support, and the opportunities they provided to us to learn about interesting technologies and gain exposure to the inner workings of complex industrial domains.

Last but not least, I wish to thank the Tchemeube family, my precious family: my brother Armel, my father Thomas, my mother Colette and my little sister Linda, for supporting me throughout this process and compromising a lot for me to focus on my studies. They were my shadows, and have certainly participated with me in this work. I dedicate this to everyone I know and for the world of tomorrow to be a better place.

Table of Contents

List of Figures

List of Tables

List of Acronyms

Acronym	Definition
3GPP	3rd Generation Partnership Project
ACS	Acute Coronary Syndrome
BE	Business Event
BI	Business Intelligence
BPM	Business Process Management
BPMN	Business Process Model & Notation
BPMS	Business Process Management System
CCL	Cardiac Catheterization Lab
CDA	Clinical Document Architecture
CEP	Complex Event Processing
CMU	Central Monitoring Unit
CT	Computed Tomography
CW	Cardiology Ward
DECT	Digital Enhanced Cordless Telecommunications
DS	Design Science
DSS	Decision Support System
ECG	Electrocardiogram
ED	Emergency Department
EDXL	Emergency Data Exchange Language
EMS	Emergency Medical System
ER	Emergency Room
ERC	Ekahau RTLS Controller
ESS	Ekahau Site Survey
ETL	Extract, Transform and Load
GPS	Global Positioning System
GRL	Goal-oriented Requirement Language
GSM	Global System for Mobile Communications
HIS	Hospital Information System
HL7	Health Level 7
IBM	International Business Machines
IR	Infrared
IS	Information System
IT	Information Technology
KPI	Key Performance Indicator
LA-BPMS	Location-Aware Business Process Management System
LAN	Local Area Network
LOS	Length of Stay
LTE	Long-Term Evolution

MB	Message Broker
MRI	Magnetic Resonance Imaging
OLAP	Online Analytical Processing
PCI	Percutaneous Coronary Intervention
PDA	Personal Digital Assistant
PFM	Patient Flow Monitoring
PIA	Physician Initial Assessment
PM	Project Management
REST	Representational State Transfer
RF	Radio Frequency
RIS	Radiology Information System
RTLS	Real-Time Location Service
SOA	Service-Oriented Architecture
UCM	Use Case Map
UML	Unified Modeling Language
VM	Virtual Machine
WBE	Websphere Business Event
WfMS	Workflow Management System
WPAN	Wireless Personal Area Network
WS	Web Service
WSDL	Web Service Description Language
YAWL	Yet Another Workflow Language

Chapter 1. Introduction

In this thesis, we support the idea of the practical use of a novel system which, by combining location awareness and business process management technologies in a unique way, enables an integrated, detailed and real-time monitoring of care processes in a healthcare environment, bringing better engineering tools to improve wait time management in hospitals. The following sections of this chapter present the motivations for our work, the problem context, as well as the research methodology that was followed. A research hypothesis is formulated, the research contributions are exposed, and a thesis outline is presented that describes the global structure and content for the remaining chapters of this document.

1.1. Motivation

This work was deeply motivated by the current state of patient wait times in Canada, and especially in Ontario hospitals where wait time reports, as of the last quarter of 2012, indicate average values as high as 14.5 hours for wait times in emergency departments (ED), 6.5 hours more than the "8 hours" threshold set by the government for acceptable wait times in Ontario (Ontario Ministry of Health's Website, 2013). In a nearby Gatineau hospital (in the province of Québec), the average wait time in the ED is over 25 hours, with 13.8% of the patients waiting more than 48 hours[1]. Such values of hospital performance are unreasonable when one is aware of the negative consequences that wait times have on various aspects of the healthcare system: they significantly limit hospitals' abilities to deliver quality care. Better understanding and control of wait times should result in higher performing hospitals, optimized operations, quicker service and better patient satisfaction.

[1] http://www.lapresse.ca/multimedias/201305/08/01-4648652-huitieme-palmares-des-urgences-tableau-complet-et-carte-interactive.php, accessed May 2013.

Another source of motivation, which represents the original incentive of this research work, was the exposure to a concrete instance of the problem of wait times through the *Osler Project*. This project is a joint academic research partnership between the University of Ottawa, William Osler Health Centre in Brampton, and IBM Canada. It provided us with tremendous insights into the problem context: a deeper visibility and understanding of wait time issues in a real hospital, hospital clinical and technological operations, patient flow management practices, hospital social environments and general management. The project also acted as an adequate case study supporting the development, implementation and validation of our research work, ensuring the latter is truly relevant to current concerns.

A third driving force for this work was the awareness gained about new and powerful technologies emerging in the healthcare IT area; particularly, the use of business process management (BPM) tools and location awareness technologies. BPM technologies have proven their usefulness globally in multiple industries for business organizations to control, shape and optimize their business processes, meeting their goals more effectively and efficiently. Whereas BPM technologies have long been adopted (e.g., in manufacturing), real-time location tracking systems are newer technologies in term of their use in the public domain. In the recent years, the possibilities to leverage these capabilities for healthcare organizations have started to be explored, demonstrating their potential applicability in various problematic areas of healthcare.

A personal source of motivation was my keen interest to study and understand to which extent new technologies applied to business process management can provide concrete benefits to organizations desiring to measure and achieve their goals.

1.2. Problem Description

The main problem on which this research focuses is wait times in hospitals, and mainly patient wait times. A *patient wait time* quantifies the occurrence, in the patient experience, of delays in care service delivery. It logically represents the interval between the time the requested care service is expected to be delivered and the time at which the delivery actually takes place.

Wait times are experienced at different levels in various care processes. Depending on the nature of the care process, wait times will exhibit different characteristics. Patient care processes are recognizable by the specific type of care they deliver such as routine care, emergency care, acute care, etc. Emergency wait times (in emergency care) are arguably the most problematic and perceptible of all. The fact that they are more likely to occur, but also more difficult to resolve is due to the high variability of two main influential factors: the pace of activities and the throughput in patient arrival. As a result, emergency wait times have a stronger impact on hospital performance. For example, it is not uncommon to observe patients on stretchers around ED corridors due to congestion caused by increasing patient wait times. The naturally higher volume of incoming patients observed in ED compared to other units needs to be supported by more efficient emergency care delivery processes (Haraden and Resar, 2004). In fact, it is shown that to avoid wait times, patients need to be processed out at least at the same pace as the one at which they arrived in ED (Hall, 2006): wait time is hence a **patient flow issue**.

That being said, in reality, hospitals are far from being able to achieve this. Patient flow management strategies are loosely implemented and sometimes ad-hoc, resulting in uncontrolled and unsatisfying wait time performances. This indicates the "unfitness" in hospital care and management processes currently in place to effectively cope with wait times and thus motivates scientific investigation. Concrete implementations of working wait time management strategies rely on quantitative approaches where sufficient data accuracy can be achieved. Those strategies should be developed in the context of process management and analytics supported by working methodologies and techniques. The analysis of the problem context has identified care process visibility to be the initial road block to wait times management and will thus be the specific focus of our work.

1.3. Research Methodology

In order to ensure this research is conducted successfully, we needed to employ a suitable research methodology. Like any other significant endeavour, high-level research, in order to be successfully conducted, needs to be adequately planned and to follow a proven strategy. Such a strategy involves methodologies, tools and techniques that allow the researcher to conduct research efficiently. Because the approach that we employed to research the problem of wait times primarily involves the development of an IT system, the Design Science (DS) research methodology was chosen and applied. This methodology exhibits a natural fit to the objectives and fundamental characteristics of our work, and has been used in the Information Science field by many research communities, as presented by Vaishnavi et al. (2011).

The Design Science research methodology is rooted in engineering and involves the "creation and evaluation of IT artifacts intended to solve identified organizational problems" (Hevner et al., 2004). By creating novel systems, it seeks to "extend the boundaries of human and organizational capabilities" (Hevner et al., 2004). The design artifacts are in the form of constructs, models, methods and instantiations built to solve a recognized problem. Figure 1 depicts the conceptual framework proposed by Hevner et al. (2004) for understanding and conducting Design Science research.

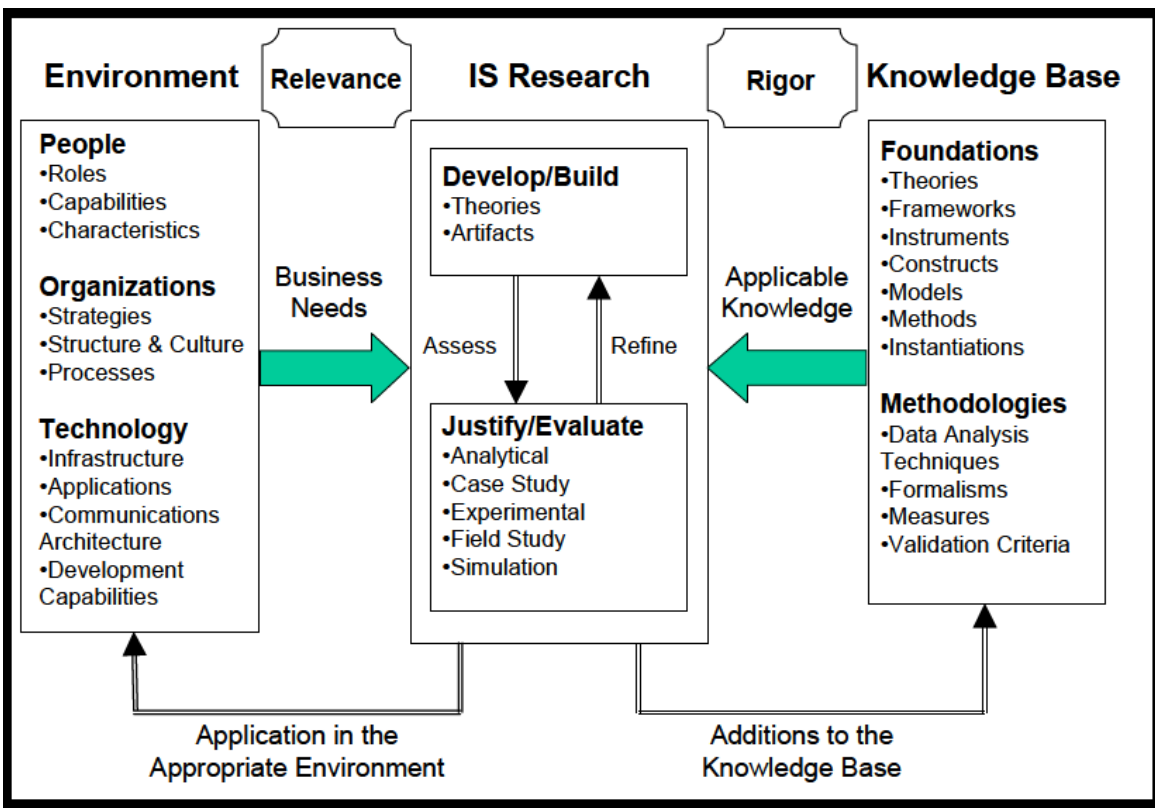


Figure 1 Design Science Research Framework (Hevner et al., 2004)

The Design Science research effort lies in the development, but also in the evaluation of a novel "IT artifact" (Hevner et al., 2004). The artifact is evaluated regarding its utility towards the targeted problem, and its adoption by domain organizations. The benefits of the evaluation of the system are that it provides insight and understanding regarding the original problem. Thus, the Design Science research process usually consists of several iterations of two major phases: the development phase and the evaluation phase. Hevner et al. (2004) define seven guidelines as requirements for effective design research (Figure 2).

Guideline	Description
Guideline 1: Design as an Artifact	Design-science research must produce a viable artifact in the form of a construct, a model, a method, or an instantiation.
Guideline 2: Problem Relevance	The objective of design-science research is to develop technology-based solutions to important and relevant business problems.
Guideline 3: Design Evaluation	The utility, quality, and efficacy of a design artifact must be rigorously demonstrated via well-executed evaluation methods.
Guideline 4: Research Contributions	Effective design-science research must provide clear and verifiable contributions in the areas of the design artifact, design foundations, and/or design methodologies.
Guideline 5: Research Rigor	Design-science research relies upon the application of rigorous methods in both the construction and evaluation of the design artifact.
Guideline 6: Design as a Search Process	The search for an effective artifact requires utilizing available means to reach desired ends while satisfying laws in the problem environment.
Guideline 7: Communication of Research	Design-science research must be presented effectively both to technology-oriented as well as management-oriented audiences.

Figure 2 Design Science Research Guidelines (Hevner et al., 2004)

On a similar approach, Peffers et al. (2006) formulate a six-step process model for doing Design Science research. The different steps can be easily understood and thus represent a useful mental model to appreciate DS research. The process model comprises the following activities: problem identification and motivation, objectives for a solution, design and development, evaluation, and communication. This is the specific process model used in this research.

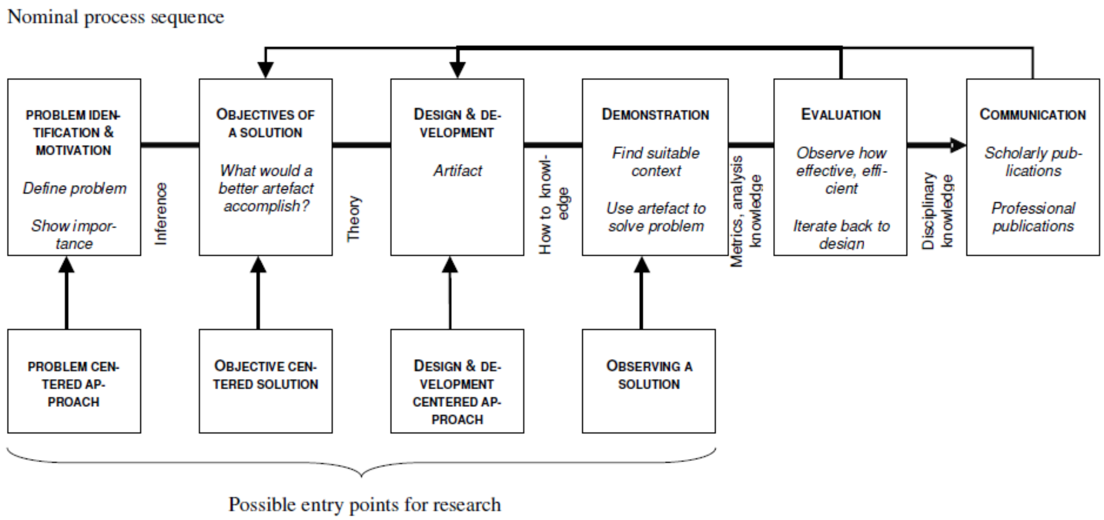


Figure 3 Research Process (Peffers et al., 2006)

We have used the DS research methodology to define and implement the following research steps (with an iterative approach, as presented in Figure 3):

- **Problem identification and motivation:** identify the problem of reducing wait times motivated by the need to improve patient care delivery.
- **Objective of the solution:** monitor patient care processes in real time.
- **Design and development:** design a location-aware business process management system (LA-BPMS) and build a sample prototype.
- **Demonstration:** demonstrate the prototype with a real clinical process from William Osler Health Centre as the problem context.
- **Evaluation:** evaluate system design and case study results.
- **Communication:** write thesis, publish papers and articles to share contributions.

1.4. Research Hypothesis

The research hypothesis is the following:

> *"The design of a software system that monitors patient care processes and measures wait times in real time is feasible in a way that provides an accurate, granular, flexible and integrated end-to-end visibility over processes in order to support and enhance control over patient flows."*

The challenge is in understanding how we can design the software system such that care process data is accurately captured as processes execute. To clarify the scope of this hypothesis, we need to explain the concepts that it references:

- **Patient care process:** refers to the series of patient-centric activities a given patient goes through as part of care delivery. This encompasses activities such as triage, consultation, X-ray procedures, etc. Care process views bring emphasis to the flow of care service delivery as experienced from the patient's point of view (patient journey).
- **Real-time:** The intention of real-time here need to be discussed. The "real-time" characteristic of our system pertains to its capacity to monitor process states as they evolve. Such a requirement can be loosely understood since the context of healthcare process management is not of a "hard" real-time nature. Even if we are talking about control flow systems, we are far from chemical, electrical or even optical applications where "real-time" concerns are on the order of microseconds or shorter. The "real-time" characteristic here refers to the delay in feedback that can be afforded on a process state monitoring system without impacting negatively effectiveness and efficiency of care process management activities in the hospital. In the context of patient flow, the affordable delay is relative to the speed of the flow of care and management processes in the hospital. Such processes typically operate at the level of minutes and sometimes hours because of the human component intrinsic to the healthcare operational domain. For example, a given nurse only moves from a unit to another every minute or so (Kamel et al., 2012). Hence, we argue that the "real-time" characteristic of the care process monitoring system can be at most, at the level of minutes. It also only needs to be relative to the speed at which the flow of activities evolves.

- **Data accuracy, granularity and flexibility:** constitute the “soft” characteristics the data provided by the system needs to exhibit. These characteristics will be critical to the ability for the system to be used and adopted in hospitals:
    - **Accuracy:** the capacity for the data to reflect valid measurement of what is being captured.
    - **Granularity:** the capacity for the data to propose different levels of details.
    - **Flexibility:** the capacity for the data to be easily accessed and manipulated.
- **Integrated visibility:** refers to the ability to understand and visualize processes in a holistic and complete way (as opposed to scattered or partial visibility). In a hospital context, this means to be able to visualize patient care activities from arrival to discharge, i.e., the entire patient journey.
- **Control over patient flow:** refers to the ability to manage patient flows and their instances as desired. This can be achieved by the development of a patient flow management strategy based on solid and in-depth analysis of accurate and timely records of hospital operations.

1.5. Thesis Contributions

The contributions of this thesis are:

- The design of a novel real-time, location-based patient flow management artifact: it is presented as a unique approach to the problem of wait times, which enables an accurate, granular, flexible and integrated visibility over process performance. A solution architecture is presented that combines location awareness and business process management to achieve that goal.
- A prototype implementation and simulation report: The prototype provides a concrete implementation of the proposed design artifact to support evaluation, but also for replicability. It informs us about the concrete benefits of our solution to the problem domain and enables other research work on this complex healthcare management domain. The prototype is demonstrated in the context of a simulation involving the

Osler hospital and other related parties. In fact, the whole development of the prototype and simulation was driven by the Osler Project. Thus, we feel that it is important to present the team as well as their original contributions in relation to the project:

- Bougueng Tchemeube R., "Location-Aware Business Process Management for Real-time Monitoring of Patient Care Processes", *Thesis: Master in Computer Science* (this thesis). My contributions include the development of a care process application system augmented with location tracking capabilities for accurate, granular, flexible and real-time visibility over care processes.
- Mouttham A., "A Framework for Real-Time Analytics and Decision Support in Patient Flow Management", *Thesis: PhD in Computer Science*. University of Ottawa, expected in December 2013. His main contributions include the design of a framework for real-time measurement, analysis and decision support in the management of patient flows inside a hospital, based on patient tracking, without burdening clinicians. He also oversaw the management of the Osler Project.
- Baarah A., "An Application Framework for Monitoring Care Processes", *Thesis: PhD in Computer Science*. University of Ottawa, expected in August 2013. The main contributions of his research include the design of a generic application meta-model, architecture and methodology (requirements, application model, implementation, and test) for engineering care process monitoring applications.
- Baffoe S., "A Generic BI Application for Real-time Monitoring of Care Processes", *Thesis: Master in Electronic Business Technology*. University of Ottawa, expected in May 2013. Her main contributions are a generic Business Intelligence (BI) portal, *Patient Tracking Dashboard*, an Online Analytical Processing (OLAP) data model and data mart, and a generic state monitoring engine based on a state-based application model for care process monitoring that can process clinical events and maintain an OLAP data model of the care process using a state handler architecture and state inference rules.

- A paper to appear in the *2013 Summer Simulation Multi-Conference*: “Real-time Simulations to Support Operational Decision Making in Healthcare” by Bahrani S., **Bougueng Tchemeube R.**, Mouttham A. and Amyot D. The paper concretely presents how real-time care process monitoring capabilities of the platform are leveraged by real-time simulation to assist short-term decisions about wait time management and resource scheduling in ED.

1.6. Thesis Outline

The subsequent parts of this thesis are organized as follows:

- Chapter 2 presents general yet relevant **background information** on the subjects discussed throughout this thesis. This is done in order for the reader to establish familiarity with the problem context and the nature of the technological fields involved.
- Chapter 3 presents a **review of** existing **related work** on the subject, i.e., papers that have addressed technical issues on similar problems, as well as the current state of research.
- Chapter 4 is the **core** of this thesis. It focuses on the **design** and the development of a location-aware business process management system (**LA-BPMS)** for healthcare monitoring. A thorough analysis of system design, challenges and trade-offs is unveiled.
- Chapter 5 showcases the **implementation** of the system in a **case study** conducted during research. We present the concrete elements implemented for the system and the process of planning, setting up and conducting the simulation. We discuss case study results, but also various challenges emerging from this application.
- Chapter 6 **evaluates** the validity of the entire body of work. It features a comparative analysis of LA-BPMS against related candidate approaches and identifies the strengths and the limitations of our work, as well as threats to validity.
- Chapter 7 **concludes** the overall thesis and highlights its main contributions in the context of the original problem domain. It also exposes **future work** that can be done to improve our work as well as several additional opportunities offered by the solution platform.

Chapter 2. Background

This chapter revisits existing concepts and related topics needed to understand and situate our work. First, we present background on wait times and the Canadian healthcare system. Next, we review the topics of business process management (BPM) principles and of location-aware technologies.

2.1. Wait Times and the Canadian Healthcare System

To better understand the particular characteristics of the wait time problem, this section describes wait times in the context of the general functioning of the current Canadian healthcare system.

As stated earlier in Section 1.1, patient wait times represent a major issue for Canadian healthcare, especially in EDs. EDs are among the most fluctuating units in terms of patient arrivals. A significant majority of patients whose condition requires immediate care usually end up in emergency departments (ED) (CIHI, 2005). So, they are also the units where the challenges related to healthcare management are the most tangible (Hall, 2006). As a result, it is not uncommon for many hospitals, in Ontario for example, to operate near congestion levels in their ED, so much that the Ontario Ministry of Health made it its top-priority health concern through the Ontario Wait Times Strategy program launched in 2008 (CIHI, 2012). The main objectives of this program are:

- To reduce wait times in Emergency Room (ER) by reporting patient's length of stay (LOS).
- To facilitate access to key surgery and procedure services by reducing wait times for services such as Magnetic Resonance Imaging (MRI) and X-ray Computed Tomography (CT scan), cardiac procedures or cancer surgery.

As a result of this program, the Ontario Ministry of Health's website publishes periodic reports on wait times in order to monitor hospital performance against established federal thresholds.

Wait times cost financially more to hospitals every year and significantly limit work efficiency of staff. They also negatively affect patient safety, health and overall quality of care (Pizer et al., 2011). A poor wait time performance represents a major roadblock to a significant improvement for care accessibility and quality.

2.1.1 Types of Healthcare Processes

So far, we have introduced the concept of wait time in its general context. However, it is important to understand that because wait times are actually a characterization of delays in processes, they can be of different types depending of the nature of the processes being observed.

Healthcare processes can be decomposed into two main general types (Mans et al., 2010):

- **Medical treatment processes:** All processes that strictly contribute to patient care.
- **Organizational processes:** The processes in place that involve not only patient care activities, but also operational activities inside and between hospital units and departments.

Figure 4 presents high-level types of healthcare processes as a simple process ontology (Mans et al., 2010):

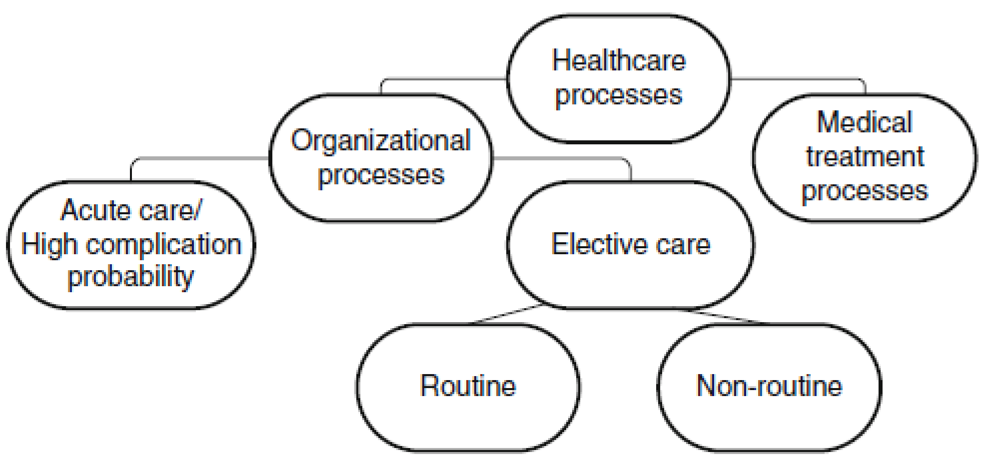


Figure 4 High-level Process Ontology (Mans et al., 2010)

Another view, provided as a hierarchical reference model (Figure 5), shows the separate clinical and operational sub-processes that can compose a patient care process.

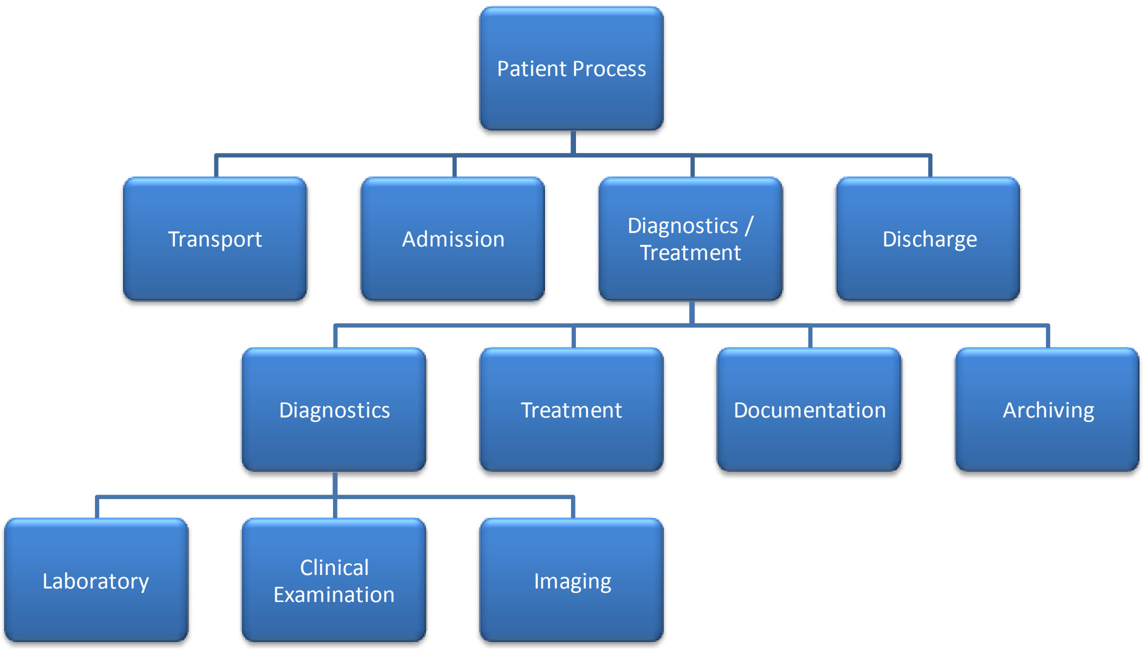


Figure 5 Hierarchical Reference Model of Patient Processes (inspired from Gattnar et al., 2011)

Those two models give an idea about the type of processes (organizational and medical) we are dealing with and the type of steps of care (admission, transport, diagnostics, treatment, discharge, etc.) involved in the problem domain. The scope of our work is limited to handling organizational patient care processes which cover, not only the steps of care previously presented, but also the tasks involved in supporting ancillary activities such as communication between providers and data management.

2.1.2 Types of Wait Times in Hospitals

The type of a wait time impacts the way it can be dealt with. Here are presented some of the existing types of wait time following the nature of the process in which they occur (CIHI 2012):

- **Wait for routine care:** This is the interval between the time the request for care is filed and the time at which the patient can actually access the care. Routine care encompasses any publicly funded services, primary care (e.g., offered by family physicians), referred specialists or requested screening.
- **Wait for emergency care:** This represents the time a patient waits before accessing emergency care services. Examples of wait times for emergency care are:
    - **Wait for ambulance to arrive**: This is the interval between the time an emergency (9-1-1) call is made and the time an ambulance actually arrives at the location of the incident.
    - **Wait for physician initial assessment (PIA):** this is the interval between the time a patient has been triaged and requested a physician, and the time the appointed physician becomes available and can initiate the activity.
    - **Wait for electrocardiogram (ECG):** The interval between the time an ECG procedure has been requested and the time it actually starts.
    - **Wait for bed:** The interval between the time no bed is said to be available and the time a newly free bed is assigned to the patient.
    - **Wait for discharge:** The interval between the moment when discharge has been requested for a given patient and the time the discharge process physically starts.
- **Wait for acute care:** This wait is the interval between the time when the patient is requested acute care delivery and the time he is assigned to receive the care. Acute care settings correspond to active treatment of severe injuries or illness for a short period of time. Elective surgery is a typical example of acute care service.
- **Wait for specialized care:** This wait corresponds to the time the patient has to wait before accessing specialized care. Specialized care corresponds to care such as mental health, home care or rehabilitation requiring particular settings for care.

2.1.3 Wait Times in a Process Context

With the size of today's society and the unpredictability of injuries and diseases, the healthcare system has evolved to become quite complex. To better understand the problem of wait times, we can embed ourselves inside the functioning of a hospital and analyse the specifics of its operations.

Multiple patients come to the hospital every day with various conditions. Healthcare professionals and administrative staff are on site to receive and treat them. Along all the patients visiting the hospital, some simply receive a diagnosis and a treatment without staying overnight; those are called *outpatients*. On the other side, others are admitted and stay overnight for days, weeks or months depending on the needs; those are referred to as *inpatients*. In both cases, patients are usually screened to detect symptoms, assess vitals and then proceed to other administrative tasks (e.g., registration). Depending on observed patient symptoms, different regulated procedures have to be taken and completed by competent healthcare professionals. Those procedures or series of steps of care constitute the « *care process* » the patient has to go through based of his condition. It is the complete end-to-end (from admission to discharge) care delivery process as experienced from the patient's point of view. Figure 6 represents a high-level general hospital process map depicting this idea.

Figure 6 General Hospital Process Map

The care process of a given patient can sometimes be complex to achieve. It is composed of tasks conducted to identify, with a certain level of confidence, the medical pathology the patient is experiencing, of tasks for curing the patient, as well as of ancillary tasks such as transportation or registration. This can represent a challenging work of orchestration between the various providers; typically nurses, physicians, transporters, cardiologists, etc. Moreover, some of those procedures required specialized equipment (MRI scanner, CT scanner, etc.), dedicated treatment rooms, as well as beds. They also rely on existing Hospital Information Systems (HIS) to access required information at critical times. *People* and *equipment* represent the resources needed by processes to be successfully completed. Process activities rely on a certain number of resources to be available at a right time (and for some activities, at the right location). The invalidation of this condition can constitute a source of wait times.

From a simple and yet general viewpoint, waiting occurs whenever at least one necessary condition for the execution of a given activity is compromised. For example, waiting can occur for an activity of conducting a consultation on a patient when the required physician is not available at the time of demand. The condition on the activity can further be related to the required state of a resource not to be busy, unavailable (temporarily), or missing. As a result, the patient care process will be unexpectedly delayed.

2.1.4 Dealing with Wait Times on a Daily Basis

To cope with patient wait times, hospitals generally have *access and flow managers*. Those are typically well-experienced nurses whose specific and daily objective is to ensure a « smooth » patient flow. They have to identify and resolve bottlenecks across their assigned unit. Bottlenecks occur as wait times get sustained and basically refer to patient flow congestion. Their occurrence is facilitated by the fact that multiple patient care processes compete for the same resource. Bottlenecks hinder the proper execution of care processes and, if not managed appropriately and in a timely manner, can cause a serious increase in wait times and compromise patient flow.

Access and flow managers work on a per patient basis (yet concurrently on many patients) identifying the particular blocking issues for one patient care process instance as well as resolution mechanisms. They then take required actions for the situation to be unblocked (e.g., by calling the transporter, advising admission to assign the next bed to the patient, etc.). It is not unusual for two access and flow managers, for example, to compete over the same bed for their waiting patients. To make managerial decisions, they rely on physical observations, communication with staff and “gut feelings” to assess the state of the care processes in their unit. This means they have to manually chase information, running around units, physically observing activities, take decisions about who to talk to and where to find them in order to access the information they need to accurately understand the care process states relating to their patients.

Those techniques and procedures are very time consuming, error-prone and limited with respect to the amount of information graspable, especially when we consider the fact that the context of hospital environments (such as EDs) is fast paced and continuously evolving. Appropriate patient flow management is a time-sensitive activity also because troubleshooting actions to resolve bottlenecks need to be implemented as soon as possible or even anticipated (whenever possible) for best efficiency. A late action can fail to effectively manage wait times and cause many negative repercussions.

2.1.5 Consequences of Wait Times

Arguably, the impacts of wait times include (Hall, 2006):

- **Difficult access to care:** wait times are directly related to the accessibility to care. They correspond to care services not being accessible at the required moment. Some patients will sometimes choose to go to one hospital or another based on which one will serve them quicker. Accessibility represents a competitive edge and an important factor in hospital performance.
- **Patient congestion:** With more and more patients waiting for care, bottlenecks can occur. Overcrowding usually results from long wait times.
- **Low patient satisfaction:** For a patient, waiting for more minutes or hours for care can really increase anxiety, stress or pain. Those conditions negatively contribute to the evolution of the patient's health and negatively impact the overall patient experience with the hospital.
- **Adverse effects on patient outcomes:** For some particular diseases, the patient's health condition can simply naturally deteriorate while waiting as a result of its medical condition worsening due to delays in the treatment of that disease. It could go as far as introducing new medical conditions to the patient, or even causing death.
- **High and Inefficient Resource Utilization:** Whenever wait times evolve to a problematic level, the intuitive reaction of hospitals is usually to add more resources to manage the backlog. This approach has been pointed out in many research efforts to be ineffective, only making the problem worse at the next point of care in the flow path (Haraden and Resar, 2004). Staff will be working at congestion level with more

resource than they would have normally needed if the problem had been addressed appropriately.

- **Higher cost:** For most of the consequences earlier presented, extra cost will occur for the hospital as a result of managing those undesirable issues.

2.2. Business Process Management (BPM)

BPM is the "management of diverse and cross-organizational processes using methods and tools to support the design, execution, management, and analysis of business processes" (Pourshahid et al., 2009). It seeks to enhance business operations and optimize business performance within a given organization. A business process can be understood as a set of value-added activities, performed by defined roles and collaborators, and organized in a way to serve an intended common business goal. BPM tools have been extensively used for more than 30 years by businesses to control and drive performance, and enhance process documentation. Some of the industries that have benefited from this technology include Manufacturing, Transport, Financial Services, Retailing, Government, Aviation, Healthcare and Telecommunications. The BPM approach is typically organized over 5 major activities:

- **Define:** Based on particular business objectives, goals and values, existing processes that require improvement, or new processes to be developed are identified. Process discovery and project scoping generally occur at this point. Typical concerns for process change are to: eliminate waste, minimize process errors or increase productivity.
- **Model:** This is the design of process models representing processes to be managed. Today, technological support allows process models to be designed and maintained using modeling software. A business process model is a flow-oriented representation of a process model (Ouyang et al., 2009). Processes are often modeled using control flow notational languages (Decker et al., 2009; Pourshahid et al., 2009), such as:
    - Business Process Model and Notation (BPMN) (OMG, 2011a)
    - Unified Modeling Language's (UML) Activity Diagrams (OMG, 2012b)
    - Event-driven Process Chain (EPC) (Keller et al., 1992)
    - Yet Another Workflow Language (YAWL) (YAWL Foundation, 2013)

    - User Requirements Notation (URN) (ITU, 2012; Amyot and Mussbacher, 2011)

- **Execute:** In this step, process models are manually implemented and executed within the organization. Today, this task can be partially or fully supported by the use of Workflow Management Systems (WfMS) or Business Process Management Systems (BPMS).
- **Monitor:** This activity consists in measuring process performance against accurately selected metrics in order to assess compliance of the implemented process models against business goals. Business Activity Monitoring (BAM) is usually looked for in this step. It is the "real-time reporting, analysis and alerting of significant business events, accomplished by gathering data, key performance indicators and business events from multiple applications" (Dresner, 2003).
- **Optimize:** This last activity consists in looking for ways to improve the managed processes based on Business Intelligence (BI) techniques such as the analysis of historical performance through performance reporting.

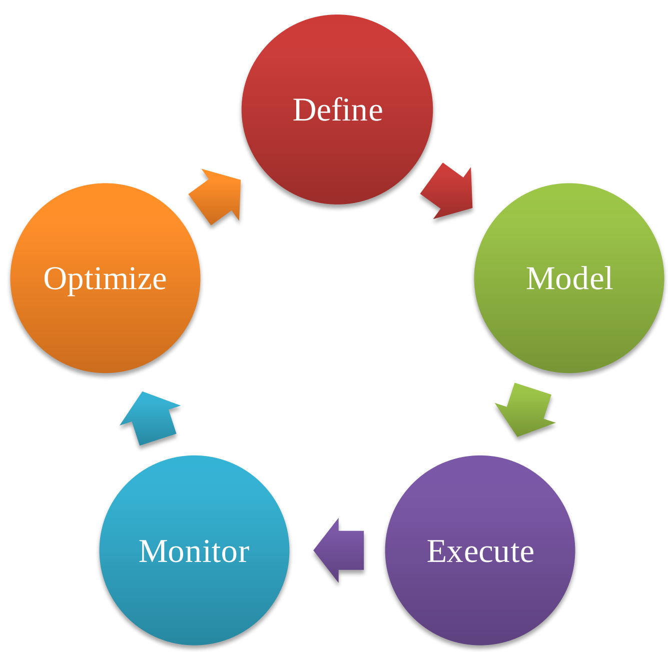


Figure 7 BPM Life Cycle

An important point to keep in mind is that those five BPM activities represent the five major steps of the BPM life cycle (Figure 7). Moreover, they are meant to be integrated as part of a continuous improvement loop; ensuring organization continuous maintenance and betterment of business operations and performance.

BPM seeks to achieve end-to-end process visibility and control to ensure a global and homogeneous impact of management activities toward business goals.

2.2.1 Process Automation: Workflows

Sometimes, process changes involve the necessity to automate a business process within an organization. Those types of activities rely heavily on the use of Workflow Management Systems (WfMS) or business process management systems (BPMS) to manage workflow development and deployment. In those cases, the process modeling activity includes additional tasks to define and implement workflow constructs such as input and output data structures, user forms, automated services, exception handling, business rules and other necessary elements of the implementation.

The term "*workflow*" refers to an artifact constructed in order to automate fully or in part, a business process. A workflow model is interpreted, instantiated and executed in a workflow management engine (or a business process management engine). Workflows have provided a natural extension to the applicability of BPM technologies in industries. Automated "lines of work activities" help streamline and augment productivity and control of organization's processes.

Executable processes built from documented process models need to be expressed using a computer language in order to be executable by a computer machine. A few popular languages have been adopted by the industries. According to Wieland et al. (2007), one of the most used and supported by BPM tools is the Business Process Execution Language (BPEL). BPEL is the short version of Web services Business Process Execution Language (WS-BPEL) (OASIS, 2007). In BEPL, business processes are defined by implementing web services-based orchestration and execution logic. In fact, WS-BPEL is an XML-based language designed to enable process activity execution and collaboration in a distributed environment by implementing process-oriented orchestrations of web services. This is a standard for process automation and builds on existing traditional web service technologies such as Representational state transfer (REST) (Pautasso, 2009) or Web Service Description Language (WSDL) (W3C, 2011).

The YAWL language, a Petri net based language, is also extremely popular in academia for designing and implementing workflows using YAWL-based WfMS.

2.2.2 Process Modeling Language Example: BPMN

BPMN stands for Business Process Model and Notation (OMG, 2011a). Bridgeland and Zahavi (2008) observed that this is one of the most popular process modeling languages across the BPM industry. BPMN is a graphical notation allowing one to specify models for business processes. One of its strengths is that it is an intuitive language, even for general business users, and it possesses a rich and extensive grammar necessary to express complex process semantics. BPMN was originally designed for representation and documentation of processes. Like most process modeling languages, it uses a notation based on flow charts. Figure 8 summarizes the main language elements of the notation.

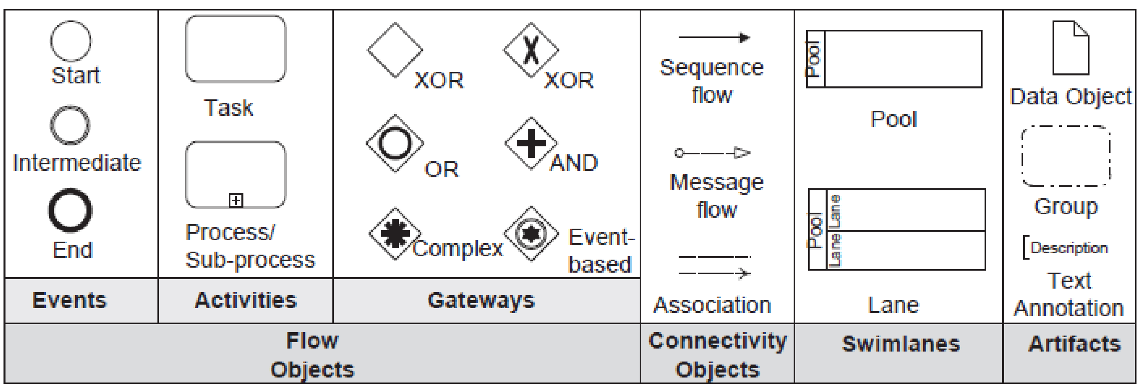


Figure 8 BPMN Language Elements (Wohed et al., 2005)

Regrouped by type, we have several language elements for BPMN (OMG 2011a):

- **Flow objects:**
    - Activities: an activity represents the unit of work to be done as part of the process it belongs to. The activity can be a simple *task* (atomic unit of work) or a *sub-process* (compound unit of work). Sub-processes are used to provide abstraction and hierarchy in process representation.
    - Events: an event indicates that something external to the process has occurred, which affects the behaviour of that process. There are three general types of events: *start events* indicate the start of a new process, *intermedi-*

ate events occur in the middle of the process flow, and *end events* indicate that the process will end upon reception of the event.

- Gateways: gateways organize control flows in the process. They indicate in which kind of flow sequence flow objects can be executed. For example, an *AND-gateway* forks an incoming flow into two or more outgoing paths indicating that outgoing flow sequences will be executed in parallel, whereas an *XOR-gateway* indicates that only one of the multiple outgoing paths can be selected. The XOR-gateway requires condition-based evaluation to determine which path should be selected. The forked flow sequences are typically (but not always) joined back using a *merging* gateway. Other more complex gateways exist as shown in Figure 8.

- **Connectivity objects:**
    - Sequence flow: such flow indicates a sequential ordering of activities over time. They basically are the arrows that connect activities together in the flow.
    - Message flow: such flow indicates flow of messages between two entities of the model. The fact that they are directed arrow allows one to identify the sender and the receiver of the messages among the two model entities.
    - Association: associations specify links between flow objects and data information or artifacts.
- **Swimlanes**:
    - Pool: a pool is a *participant* in the business process. A participant is said to be a business entity (e.g., a company).
    - Lane: a lane is a *sub-partition* of a pool and generally corresponds to business roles, actors or similar resources inside that lane.
- **Artifacts**:
    - Data object: data objects represent the data involved in the process model. More precisely, they can be used to specify the nature of data inputs and/or data outputs on a given activity in the process. They are linked to process elements using *associations*.

- Group: groups are used to categorize a subset of elements in the process model. They are useful to highlight and delimit the different “phases” occurring in a given process by regrouping the process elements belonging to each phase. Contrary to sub-process model elements, groups are not constrained to pools and lanes.
- Text annotation: an annotation allows attaching a *textual note* to an element in the process model (like a comment).

Figure 9 shows a simple “Patient Arrival in ED” BPMN process model example capturing the sequence of activities done upon arrival of a patient in the ED department of a hospital.

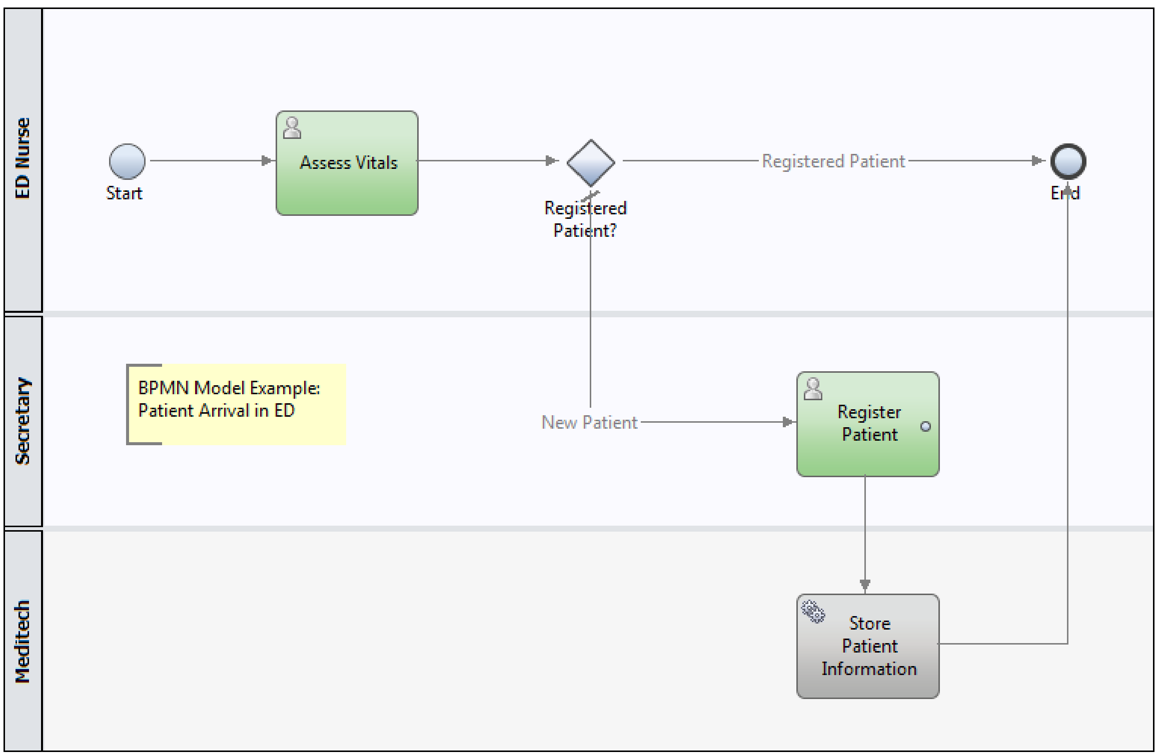


Figure 9 Example of BPMN Process Model: Patient Arrival in ED

The process starts with the ED nurse assessing vitals of the patient (task). Then an XOR-gateway evaluates whether the patient is already registered with the hospital (e.g., he/she is a returning patient). In the case the patient is already registered, the process simply terminates. Otherwise (if the patient is new), the task “Register Patient” is conducted by the secretary to have the patient registered into the system. This task consists in a clerk entering patient information on a form. Finally, the data entered is stored in the Meditech Hospital Information System (HIS) and the process ends.

2.3. Emergence of Location Awareness

Location awareness refers to the ability for a system to hold knowledge of location of some entities of interest, and to operate in a way that accounts for that information. The term applies to navigation systems, and to real-time locating and positioning support systems. The development of reliable positioning technologies such as Global Positioning System (GPS) has enabled applications in traffic, logistics, business administration and leisure applications. Location awareness is actually a specialization of the more global concept of context awareness. Context models have been proposed to support context-aware applications that use location for various goals.

2.3.1 Location Technologies

Today's location technologies are developed to track the position of an asset with high accuracy and coverage while limiting the cost of the infrastructure needed to accomplish this. There exist 3 major types of locations technologies based on their coverage capabilities (Wikipedia, 2013):

- **Global:** This represents a global coverage over a significantly wide area (e.g., around the earth). In this category, the GPS technology is the predominant option. Assets are equipped with GPS receivers, and tracked via the use of GPS satellites.
- **Regional:** This approach corresponds to a medium coverage (e.g., a town or a country). Here, mobile phone technologies, such as 3rd Generation Partnership Project (3GPP), Global System for Mobile Communications (GSM) or Long Term Evolution (LTE), can be used for tracking. Mobile devices rely on near antenna towers to identify their position.
- **Local:** A local coverage refers to a single and contained unit of space (e.g., a buildings and rooms). For this requirement, solutions such as real-time location system (RTLS) technologies and Wireless Personal Area Network (WPAN), Wireless Local

Area Network (wireless LAN) or Digital Enhanced Cordless Telecommunications (DECT) exist. The performance of local systems degrades as the distance from the target area increases.

2.3.2 Real-Time Location Systems

Real-time location systems (RTLS) are “local systems designed for the identification and tracking of location of assets and/or persons in real or near-real time” (Kamel et al., 2012) around a contained zone.

One straightforward way of implementing a RTLS involves the use of wireless tags attached to assets or humans, which send wireless signals to fixed receiver devices (location sensors). These devices (e.g., an existing wireless network infrastructure, or explicit beacons) dispatch the information received from the tags to the location software system. The location software system typically uses advanced computational detection techniques to infer tag location based on the received information which generally communicates: the tag identifier (ID), the sensor’s location and other relevant information required for location detection (e.g., signal strength). The location software system logs the information received from the tag, stores the computed information indicating the physical location of the tag, so to make it available to third-party applications interested in accessing that information.

In radio-frequency (RF) communication, RFID tags are used. They can operate in two modes: active and passive. In active mode, a tag requires power supply (such as battery) and emits a radio signal on a periodic basis (ensuring continuous tracking). In passive mode, no power is required and the tag is simply activated by a scanning signal sent by a location sensor whenever the two are in proximity. That being said, concrete implementations of RTLS sometimes use variations of the presented model of tracking and employ more diverse telecommunication technologies depending on the tracking requirements, the constraints, as well as the opportunities pertaining to the problem context.

Depending on the technology employed and the extent of configurability, the system can capture location information at various levels of location concerns (Kamel et al., 2012): presence-based, room level, sub-room level, choke points, association or precise. Some of the technologies used are: GPS, Wi-Fi, RFID, infrared (IR), ultrasound, Bluetooth, etc. Each approach provides various strengths and weaknesses in rendering location tracking. For example, GPS provides extremely high precision and reliable signal for worldwide tracking coverage with a location accuracy of 5 to 20m. However, it requires skyline of sight to function properly and as a result, does not work well inside buildings. Wi-Fi technology uses access points to offer around 50 to 200m of tracking coverage and a location accuracy of about 2 to 3m. At the other end of the spectrum, IR technology offers centimeters in location accuracy, but can hardly go through materials. Therefore, in practice, this often limits its coverage to few meters.

Sometimes, the best performance for a given application can be achieved by adequately combining the different technologies. Some of the important qualities to consider when selecting/configuring a location tracking system are: data transmission rate, signal strength, geophysical range of tracking coverage, granularity/resolution of location information (location accuracy), system power requirements, battery lifespan (when applicable), installation, configurability and scalability. It is worth mentioning that real-time location systems are rarely used in hard real-time problem (e.g., turbine control systems). The assets or individuals being tracked usually move every few hours or minutes. As such, the ability to update location of the tag every minute, for example, is a sufficient requirement. In fact, real-time location systems often only need to update location information of tracked resources whenever their location changes. That is why some solutions offer multiple update mechanisms. For example, the location could be updated only when the asset/person moves as detected by a motion sensor placed on the tag. Precision of location tracking can even be made more granular by also updating whenever the asset/person stops its motion. This can be combined with periodic location updates occurring every minute to offer flexibility in tracking capabilities.

Potential benefits of such technology include cost reduction, improved processes and communication, and enhanced mobility (Tzeng et al., 2008).

2.4. Chapter Summary

This chapter has presented the background necessary to understand and evaluate the remaining contents of this thesis. Prior to our approach to design a location aware business process management system (LA-BPMS) for monitoring care processes, we have reviewed basic notions of BPM and of location technologies, as well as the nature of wait times in the Canadian healthcare system. Our next step is to research and present existing academic work relevant to our problem domain to better understand what has been achieved so far and what the remaining challenges are.

Chapter 3. Related Work

3.1. Literature Review Methodology

To conduct this literature review on wait times, BPM, and RTLS in healthcare, multiple sources of information have been queried at different points in time. The primary literature sources used were:

- Online publication systems: SpringerLink, ACM Digital Library, IEEE Xplore, GoogleScholar, ScienceDirect, and Elsevier were the main online sources used to search for academic papers relating to the topic of wait times, BPMS and RTLS applications in healthcare. Those systems were queried with specific keywords[2]. Results were then skimmed by manually evaluating relevance based on papers' titles and abstracts. Selected papers were then read through and locally organized in folders by topic. Many other papers were found by following references and citations from original papers. Some papers were latter discarded based on low relevance to the specific topic of this thesis.
- University library: Search engines of the University of Ottawa's library were used to further the literature review. Some books were identified that fit our subject. Other books were also found online (Google Books). A similar skimming process was applied to discard less relevant ones.
- Various sources: this includes sources used following an unsupervised methodology. Papers coming from peer recommendations, as well as author and professor websites, Wikipedia and other resourceful websites (white papers, industrial case studies, etc.)

[2] Several queries were made using combinations of the following keywords: "location-aware business process", "location-aware workflows", "context-aware business process", "context-aware workflow", "process automation in healthcare", "design science research", "human intensive processes", "BPMN", "BPEL", "business process modeling", "workflow modeling", "wait times Canada", "wait times ED Canada", "hospital process map", "measuring wait times in healthcare", "measuring wait times", "wait time theory", "queuing theory", "real-time monitoring healthcare", "measuring quality of care", "hospital performance metrics", "patient flow management", "hospital information system", "complex event processing", "CEP in healthcare", "HL7", "RFID literature review", "location tracking system", "RFID Fingerprinting", and "RFID trilateration".

and information channels. The selection of documents from these sources was more selective based on their lower authenticity and validity.

A total of 53 documents were gathered for this literature review based on their titles and abstracts. All 53 had their content reviewed but only 14 were retained based on the problems they were addressing. Finally, after studying these papers, the 9 most relevant papers were selected to be presented in this thesis.

The next sections present related work on the subject separated in three parts: wait times in healthcare, development of BPM solutions in healthcare, and the use of RTLS technology in healthcare.

3.2. Wait Times Measurement Strategy and Real-Time Data Analytics

In hospitals nowadays, the way wait times are calculated is cumbersome. We need to investigate existing methods of collecting data and calculating wait times.

In most hospitals, business-value information about clinical and operational processes is captured using paper forms. This means that a human resource has to manually write data on forms. Those forms then have to be manually reported into the corresponding hospital information system (e.g., Meditech or OACIS) depending on the kind of data. Data from the HIS can be later *extracted*, *transformed* and *loaded* into a data warehouse (ETL process). Reports can then be built that extract wait times information from the data warehouse and make it accessible to analysts. This process can be loosely summarized by the diagram shown in Figure 10:

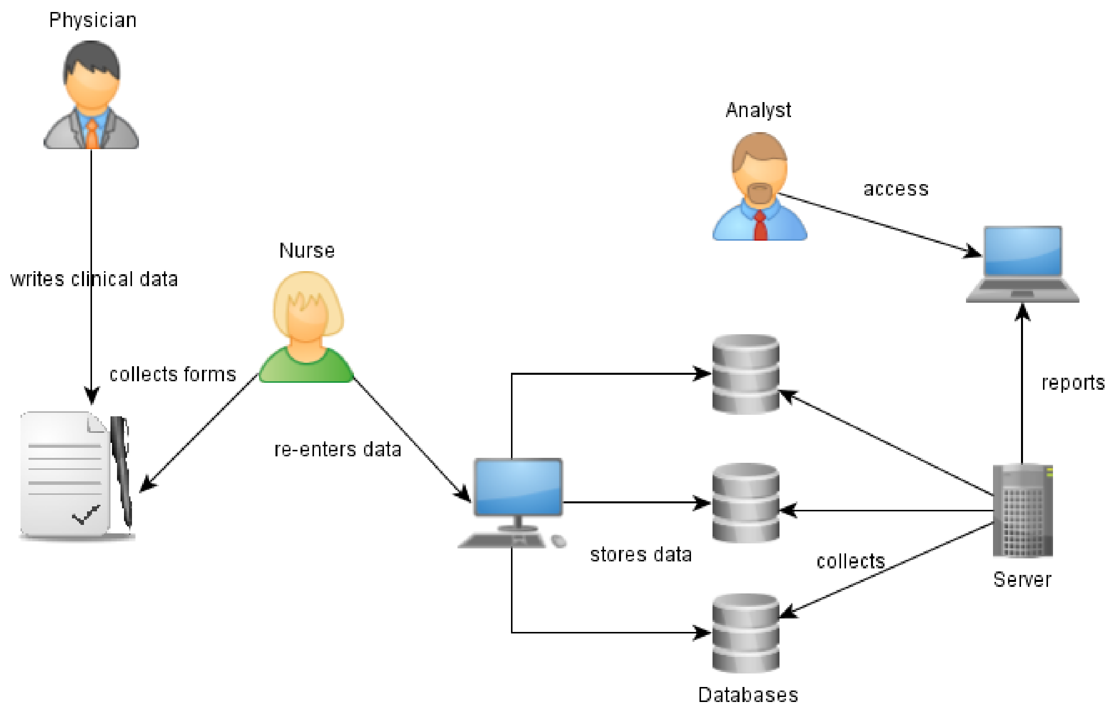


Figure 10 Traditional Manual Data Flow in Hospitals

The issues with this approach are numerous.

First, writing data on paper forms is a *manual* activity (e.g., using pen and paper) that can be time-consuming and error-prone. The fact this activity may have to be performed by a nurse or a doctor in parallel with his/her other clinical tasks can also impact efficiency of workflows. Moreover, because the data collection process intertwines with clinical activities, often only clinically important information is captured through forms.

Second, entering the data captured on forms into the HIS adds a costly *redundancy* to the data collection process since the same data is always manually entered twice. This is also a belated and time-consuming activity. It is important to remember that HIS are deployed as computer stations at specific locations across hospitals. This implies that the staff cannot always afford walking long distances to reach the HIS multiple times a day. As a result, forms are often collected and input to the corresponding HIS *later* during the day or at the end of the day. This is done by clerks, doctors, nurse or other staff depending on the particular organizational dynamics of the hospital and circumstances.

This process makes for a late availability of data and an increased probability of issues regarding data *consistency* such as data omission, data entry misspelling, etc.

If form filling is the primary way of collecting data, some data is directly input to the system. For example, an access and flow manager might verbally request a bed for a patient, or physicians might request a schedule for a procedure (e.g., hip surgery) for their patients. Those requests are directly entered in the system. Another example is the triage activity, which is commonly done by nurses with the help of their computer station to directly input triage information.

Now, for wait time measurement, the first thing to realize is that *timestamps* need to be captured for the various clinical activities. The problem is that, among the data collected, the amount of information pertaining to wait times is extremely limited. To solve this issue, hospital sometimes use surveys to collect wait times data for a selected pool of treated patients. For example, the questionnaire of the survey could ask patients to state the amount of time they *think* they have waited for a specific service. Apart from the probable lack of accuracy of the data collected about wait times (especially if the survey is conducted long after facts), this kind of surveys actually allows hospital to collect data about patient satisfaction with regard to wait times and other quality aspects of care services (relation with staff, layout of facilities, etc.).

Another issue of data forms for wait times is that they usually do not have a sufficient level of granularity and details for time tracking. It is thus impractical with the described methods to capture timestamps for all activities given the number of activities to manually track (with pen and paper), and given the limited accuracy that the collected timestamp data brings. This is why only certain wait times are accessible at the moment. Some of the available timestamps today include timestamps for when patient triage is done by the nurse, whenever a request is made (bed, room, or procedure), whenever the patient is "logically" the requested resource in the information system, or when the patient leaves the emergency room. The wait times calculated from those situations usually are too inclusive of other activities and do not offer any precise insight regarding the circumstances of the wait.

Capturing information from a HIS leads to various challenges. First, the extraction of data (e.g., patient vital information or physician diagnosis) from such systems requires clearing and processing data in such a way as to remove confidential information or other information that will not be used. Data has to be reorganized/restructured to be adequately stored in a database or data warehouse. Then, multi-dimensional data “cubes” are usually built in order to enable proper and efficient reporting of performance data. The whole process is however demanding in terms of time and technical skills.

Overall, the existing methods for collecting data and computing wait times in hospitals are neither efficient nor effective. This makes it very difficult for hospitals to understand their processes and strategically improve their wait time performance. Analysis of reports is also quite limited and any conclusive result is thus approximate. Hence, institutions who want to try any corrective measure based on such results do it at their own costs/risks. Accurate monitoring of wait times is nearly impossible. The major identified roadblocks to achieving better control and strategic performance are:

- **Quality of data:** accuracy, precision, completeness, and consistency.
- **Interference with clinical tasks/Burden on staff:** doctors/nurses have to think about the patient’s related activities and organizational tasks.
- **Lack of mobility of information:** information is conveyed by human agents. This means that the agent has to move to the location of the HIS (or call) in order to have data persisted in the system.
- **Redundancy in data collection process:** The manual activity of data entry has to be performed twice (in paper form and into the computer).
- **Decentralized IS model for patient flow data source:** process-related data is distributed across multiple HIS (organized by functions) and it is difficult to reconcile this data in order to present a care process or patient flow view.
- **Late access to wait time data:** the turnaround between the time wait times that “physically” occur and the time they are quantitatively observed and made available through reports to performance managers is too slow and thus introduces bias in hospital performance measurements.

3.3. BPM-Based Systems

In this section, we investigate and discuss applications of BPM technologies such as process automation (workflows) in healthcare.

Over the recent years, many studies around BPM-based systems have been conducted in order to solve problems in healthcare identified as process-level issues. Widjaya et al. (2011) present the use of Business Process Re-engineering to develop a supporting IS meant to improve performance in hospital processes. In the context of analyzing "ordering laboratory test" processes, this paper identifies global issues with clinical and organizational operations such as the redundancy of some ordering requests, the potential for human error (because of paper form filling) and complex multi-actor and multi-departmental characteristics of these ordering processes. The authors specify four types of "waste" in processes: waiting, redundancy, rework, and over processing. They then explain a redesign approach of actual business processes that leverages the capabilities of a Digital Order Entry system. Simulation is used to support the argument that the performance of the new process of ordering tests would be significantly better, solving the process issues presented earlier.

Another application of BPM in healthcare is to tackle the issues of medical errors and to increase the quality of care by facilitating an efficient flow of information between healthcare providers and by supplying patients with personalized medical information regarding their medical treatments. In their paper, van Hee et al. (2008) apply their adaptive workflow nets framework to bring a process orientation to medical data flows. Criticizing the limited nature of HIS as being data-centered, they advocate the necessity to "shift the focus of healthcare information systems from data to processes" as a way to provide process-relevant information to medical data, and to reduce patient safety risks deriving from the lack of communication between different healthcare providers. This paper talks about modeling patient care processes using workflow nets (inspired from Petri nets) in a way that is adaptive. That is, the ability for the created workflow to modify itself on-the-fly. Process flexibility as well as process exception are discussed to provide solid foundations developing an IS that implements the framework so as to fully support healthcare processes. A motivating example is presented that highlights the benefits of a process-oriented web-based personalized medical IS such as enhanced communication and better quality of care.

Zhang et al. (2009) present the development of a workflow-based information system in a radiology department. This is a use case to communicate a novel methodology for supporting hospital process management in improving healthcare efficiency and quality. They present the detailed steps of deploying a workflow-based system by consistently discussing development activities of workflow modeling as well as an architecture and implementation of a solution system featuring a “worklist” handler to manage process work items, an integration engine to connect to HIS, and a workflow dashboard to track process states. The result is a comprehensive workflow-based information system deployed in the radiology department of the Chinese Hospital of Medicine. The authors argue that the system’s decoupled architecture and workflow-oriented design allow for flexibility in operations, workflow-aware integration, and process monitoring and improvement.

Zhu et al. (2010) propose a workflow-based monitoring dashboard to provide an integrated overall view of the radiology process. The tool is meant to be used by radiology managers to monitor, in real time, the state of their workflow in a heterogeneous environment that originally prevented it. Again, the authors design and implement a YAWL-based workflow system, with a process model capturing an integral and comprehensive view of patient-related process information. This paper discusses integration with legacy systems and the finding of a set of indicators suitable to communicate radiology performance. The authors have identified two particular metrics to be critical for monitoring radiology processes: the examination turnaround time (time from preliminary examination to examination finalization) and report turnaround time (time from preliminary reporting to review finalization). They claim the tool helps radiology managers better anticipate process issues that may be of concern and facilitate decision making.

The work of Poulymenopoulou et al. (2012) consists in the development of a computer system for emergency medical services (EMS). The platform is an intelligent, cross-institutional emergency medical system that leverages process automation in a cloud infrastructure to support emergency care processes in an integrated way. The considered processes cover pre-hospital and in-hospital care activities (from the time request for ambulance is made to the time patient exits ED of the hospital). The authors emphasize the problem related to the distributed, yet interdependent nature of those activities and focus on improving collaboration and coordination of the various parties involved (namely the EMS agencies in charge of pre-hospital activities, and the hospital organization receiving the patient in its ED) in those procedures through information sharing. A custom EMS workflow application is proposed and built using REST services. It is composed of a context manager managing emergency tracking, a process aggregator module for process orchestration and a messaging system to communicate with process actors' devices (mobile or desktop). A proposed context ontology is used by the context manager module to help manage context information regarding assets' location and time at various points of interest (e.g., arrival of the ambulance at the place of an incident). This allows the required process steps to be triggered. More precisely, system-to-system services are implemented directly in the workflow module whereas for system-to-human services (human tasks), the messaging system is used to call the corresponding REST service (located in a cloud environment) that communicates with the provider's device for him to complete his task prior to passing security measures. The normally distributed emergency data are now brought together on a single platform and communicated in the form of XML documents following XML schema definitions supporting the appropriate medical standards such as the OASIS Emergency Data Exchange Language (EDXL) used here for encoding hospital resource information and Clinical Document Architecture (CDA) and Integrating Healthcare Enterprise (IHE) profiles used for patient information relating to emergency history. As a result, the system provides automation of EMS processes following an integrated and standardized approach and merging information from various sources for easier accessibility and efficient completion of process activities; the overall goal being to improve quality of emergency care.

3.4. RTLS-Based Systems

This section reviews location tracking applications and research work in healthcare. Numerous applications of RTLS technologies have been observed in the healthcare industry so far. Most location aware solutions have been applied to address issues related to areas such as asset management, inventory management, authenticity management, identity management or process management (Buyurgan et al., 2013).

A first example of such application is presented by Maglogiannis et al. (2007). They present the design architecture and implementation of a location-based system for tracking emergency medical incidents. The system is built for emergency situation detection on any location (on-site at the hospital as well as in uncontrolled environments such as urban areas) for chronic patients requiring continuous health monitoring. It uses GPS for outdoor tracking and the UCLA *Nibble* system (a Java-based system built by the University of California in Los Angeles for location identification) for indoor. The patient wears a personal device capable of tracking location as well as monitoring patient's vital signs such blood pressure, heart rate, etc. At the other side (and independently of his location), the doctor carries a portable device (e.g., a Personal Digital Assistant – PDA) capable of receiving and displaying patient vital signs, accessing other patient information from the patient central monitoring unit (CMU). The patient's device continuously collects vital signs and automatically compares them against set thresholds such that whenever thresholds are exceeding, the device raises an alarm to the CMU, which can then relay the information (including patient's current location) to the doctor's PDA and other appropriate bodies (e.g., nurses) for prompt actions to be taken. This is an example of location-aware medical applications for emergency management.

One example of RTLS deployment in asset management is the tracking of medical equipment across a hospital. It is well known that it is often difficult for staff to locate portable equipment (e.g., intensive care ventilators, intravenous pumps, defibrillators, wheelchairs, etc.) that they need in large hospitals. This makes the process more time-consuming than it should be. As a result, it is reported that some hospitals buy 10% to 20% more equipment than required for staff to be able to find and use it (Kamel et al., 2012). This represents a significantly higher cost when one considers the amount of portable equipment a hospital has to manage on a daily basis. The deployment of a RTLS solution to locate the portable equipment proves to reduce those costs and increase efficiency. More precisely, equipment can be found at any moment so staff members do not need to search for it. This application helps improve visibility of medical equipment across the hospital.

Another use of RTLS asset management involves bed interaction of the RTLS solution with a HIS for open bed management where the RTLS engine interacts with a bed management system to monitor bed states. The process also helps improve the synchronization of housekeepers for bed cleaning.

Tzeng et al. (2008) report on the development of a wireless PDA integrated with RFID technology to be used in emergency rooms. For this configuration, passive tags are worn by patients upon arrival in ED, and a patient's ID number is associated with the tag. Then, whenever a patient tag is scanned against a reader (location sensor) and the patient is identified (through its patient's ID), appropriate medical-related information for that patient is pushed to the PDA device of the caregiver. This project was launched in Wan Fang Hospital (Taiwan) to improve patient satisfaction and safety by limiting medical errors in emergency room. Whenever tags need to be worn by a patient, the patient's written consent is commonly requested prior to the operation.

Other works have also shown the use of RTLS to monitor patient movement between departments (e.g., transfer from ED to cardiology ward) in order to track how much time patients spent in different areas, and adjust staff or equipment allocation across those departments (Kamel et al., 2012).

One last area where RTLS technology has been investigated is for sanity compliance and more generally error prevention. Personnel can be tracked in order to detect any unauthorised access to certain areas, or for mental health patients to ensure they stay in their dedicated zone. Tracking staff in such a way can also ensure compliance to sanity regulations such as proper disinfection of equipment or hand hygiene protocols to prevent on-site infections. "Error prevention constitutes the biggest use of RFID in healthcare in terms of number of tags sold" (Buyurgan et al., 2013).

The general success of RTLS in improving healthcare operations as led to its gradual adoption in the industry worldwide. However, major roadblocks still exist for new adoptions as each hospital presents its own set of particular challenges. Technological limitations, interference concerns, prohibitive costs, lack of global standards and privacy concerns usually slow down the process. On this matter, Kamel et al. (2012) insist that choosing a proper RTLS technology that matches its intended application by making the right trade-offs in RTLS qualities to meet key requirements, deployment environment, budget and future expansion plans, is a crucial step towards achieving a successful implementation.

3.5. Uniqueness of Healthcare System

As some of the presented work might suggest, the healthcare "industry" is particularly singular by nature. The adoption of new technological advances depends on the ability of technological solutions to consistently "blend" with the challenging healthcare context of domain features whose combination has not yet been witnessed in other sectors (e.g., manufacturing, finance, education, transport, law, etc.) (Hall, 2006):

- **Human intensive processes** (Clarke et al., 2010)**:** This refers to the fact that a majority of care activities heavily involve knowledgeable human actors to be completed. This human dependency becomes an important factor that requires technological approaches to process management to be extremely flexible (e.g., account for human errors).
- **Processes dealing with maintaining human lives:** The healthcare system is about dealing with lives of citizens of a society. This makes it a naturally sensitive domain where fault-tolerance is important.
- **Dependency of clinical activities on location:** Most clinical activities are location-dependent. That is, in order to be completed, they required the involved resources (actors, equipment and patient) to be at certain physical location in the hospital. This is usually due to 1) the geophysical organization of the hospital and 2) the "fixed" characteristic of the equipment (MRI machines, etc.). More specifically, most care activities require the patient to be physically present.
- **Patient health can deteriorate while waiting:** this is presented as one of the consequences of wait times in Chapter 2.1.5 (adverse events on patient outcomes).
- **Variability of patient arrival:** The high variability of patient arrival makes scheduling and planning disruptive.
- **Though regulations:** Healthcare is one of the most regulated systems in America. The numerous regulations hospitals need to comply with on a daily basis make pro-

cesses and procedures, in some cases, too rigid. This inflexibility adds complexity to technological requirements to be considered for a given solution.

- **Fragmented environment** (Lee and Duffy, 2009): Services are physically fragmented per specialization across the hospital. Collaboration and coordination between processes, providers and departments is difficult.
- **High volume of digital transactions** (Lee and Duffy, 2009): Much data is generated per visit for each information system. There is a naturally high volume of data and transactions to support and manage.

The combination of the described characteristics makes issues pertaining to healthcare uniquely challenging. The development of adequate technological solutions, especially for process-related matters, is a huge endeavour, and those characteristics must be kept in mind.

3.6. Chapter Summary

This chapter has presented an overall view of the existing body of knowledge pertaining to our problematic of improving visibility over care process data. The actual process of performance data gathering has been presented along with the current issues associated to it. Then, we have reviewed existing work using BPM-based systems to improve process performance and quality of care. We also have explored recent applications of RTLS-based systems in various areas of healthcare and have observed how they can benefit hospitals. The unique traits that characterize the healthcare system have been presented. The combination of BPM and RTLS technology into a single system becomes an eventuality than we can now explore. The next chapter presents a unique solution platform that has been put together, which answer our need to improve data visibility and better integrate process activities.

Chapter 4. Location-Aware BPMS

This chapter defines a location-aware business process management system (LA-BPMS), which aims to provide real-time process monitoring of current care process performance to improve wait time visibility and patient flow control. This chapter provides insights into our solution approach, for understandability and replicability. We present and discuss requirements for LA-BPMS and its architecture, including an analysis of design choices and rationales regarding its components as well as their impact on system integration and quality.

4.1. Problem Specification

4.1.1 Identified Issues

The background literature presented in Chapter 2 shows that the wait time problem in healthcare is not necessarily a resource issue. Also, it is not a completely unpredictable phenomenon (Hall, 2006). One thing that is known is that it is a patient flow-related issue. A patient flow emerges from a holistic view at the concurrent patient "service lines" sharing common resources. In that context, efficient wait time management comes from a solid and educated patient flow management strategy.

It is argued that wait times cannot be and should not be completely eradicated. However, a working patient flow strategy should aim to limit long wait times and their impact on hospital business values (patient health condition, patient satisfaction, cost, resource utilization, etc.). Such a strategy should leverage known and proven methodologies. One example is to match demand to capacity, as presented by Hall (2006). In that sense, patient flow management relies on accurate, detailed enough and timely visibility over wait times and care process states (or patient states along care processes) in order to be effective. This idea is particularly where our approach focuses on. Patient flow agents rely on the information about "what is going on" in order to make proactive decisions and cope with bottlenecks within a critical amount of time. The accurate visibility over process information directly affecting wait times, together with measures of wait time performance, are a must to sketch a way towards improvement. As Hall says in his book: "what you don't measure is hard to improve" (Hall, 2006). So, existing **issues** in patient flow management procedures can be summarized as follows (Chapter 3.2 and Chapter 2.1):

- **I1:** Lack of visibility over the execution of process operations (e.g., incompleteness, and current state of resources affecting patient flow).
- **I2:** Insufficient data capture mechanisms to accurately monitor patient care process activities and patient wait state (e.g., paper forms add human redundancies and delays).
- **I3:** Lack of clear understanding of patient flow issues (in the way they apply in a given unit or hospital).
- **I4:** Lack of reactiveness caused by late accesses to data that are usually incomplete and not reliable.
- **I5:** Difficult process mining for distributed HIS and costly/inefficient data processing (data access & data processing).

4.1.2 Goals

Regarding the five issues from the previous section, the four objectives/goals for the proposed solution are defined in Table 1.

Table 1 System Goals

Goal ID	**Goal Descriptions**	**Issues Addressed**
G1	Achieve end-to-end and accurate capture of all care process executions. That is, monitoring current patient care process states (including wait states).	I1, I2
G2	Provide a more granular model of process state execution. This should give a more detailed view, whenever needed, of the exact steps in process execution that are occurring at a given time.	I1, I3
G3	Capture and provide data in real time.	I4

Goal ID	Goal Descriptions	Issues Addressed
G4	Enable easier off-line processing of process performance data.	I5

4.1.3 Functionalities and Guidelines

The LA-BPMS functionalities and guidelines chosen to realize the four objectives are as follows:

- Capture process data as processes execute.
- Track the process execution state of care for each patient of interest. Process states will be indicated by patient states aligned with clinical process execution (e.g., in procedures, in consultation, waiting for consultation, etc.).
- Track location of process-related resources of interest (e.g., patients, physicians, nurses, transporters, housekeepers) across the entire hospital.
- Detect and monitor care process wait state across the complete clinical workflow by combining location awareness of physical process resources and care process execution records for a given patient.
- Record service times across the processes.
- Store captured process information in a centralized and process-oriented fashion.
- Expose recorded and computed critical information to users or third-party systems as needed.
- Avoid negatively impacting staff productivity.

4.1.4 System Requirements

In Table 2, we transform the functionalities and guidelines into requirements for our system, with links to the goals they help satisfy.

Table 2 System Requirements List

Req ID	Requirements Descriptions	Goals
R1	The system shall start a business process model instance for each new patient visit.	G1
R2	The system shall collect end-to-end accurate process data information for each care process.	G1
R3	The system shall monitor process execution state and process wait state as processes execute.	G3
R4	The system shall track location information of process resources (e.g., physicians, patient, nurses).	G2
R5	The system shall automatically and accurately detect and monitor patient wait state for each care process instance by combining expected operational state and knowledge of contextual location of process resources.	G1, G2
R6	The system shall operate in real time, meaning there shall be a continuous feedback loop with a business relevant internal time interval. This loop should not jeopardize flow managers to react on time to wait time issues.	G3
R7	The system shall track start and end times of each care delivery activity (i.e., service times).	G1, G2
R8	The system shall persist data as it is collected. A process-centric and centralized way of storing process data records for easier off-line processing.	G4
R9	The system shall make recorded performance data to be accessible to end-users and third-parties.	G4

4.2. System Architecture

Based on the system requirements from Table 2, we present the initial high-level system architecture in Figure 11.

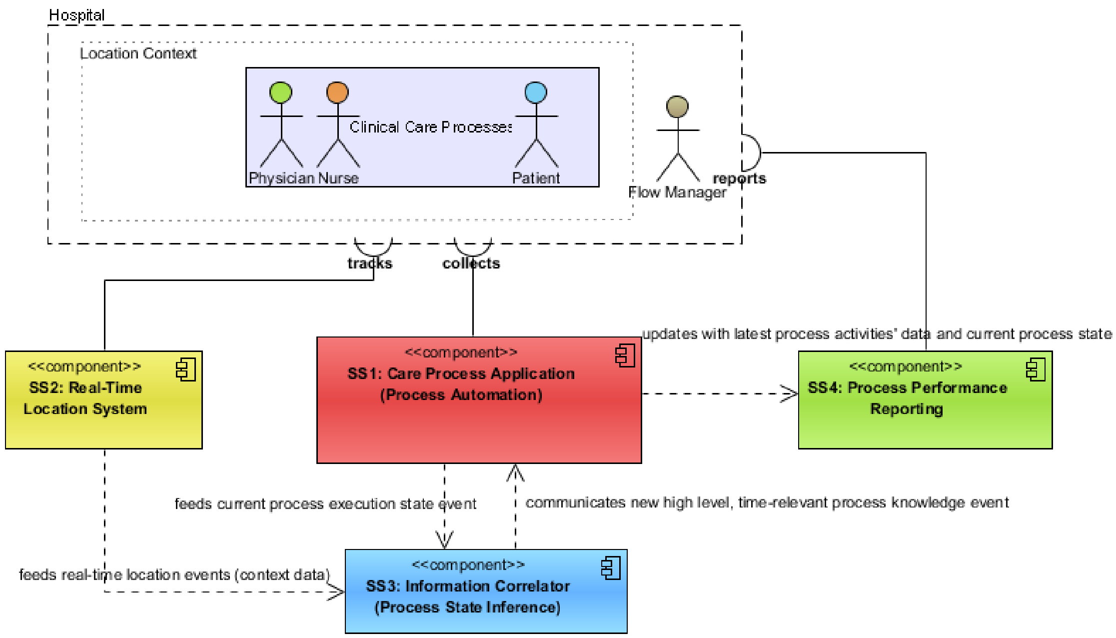


Figure 11 High-Level System Architecture

The nature and functions of the components (or subsystems) of this architecture are as follows:

SS1[3] : Care Process Application. This component is the main data collection enabler. It is at the heart of the monitoring capabilities, providing the ability to capture data in a way such that the collected information is organized in a process-oriented view. This component contains process model representations of the clinical care processes to support, and helps monitor their performance by recording updates of the process model instances. At each step of the execution, it feeds updates on execution status about its process model instances to the information correlator (SS3). It also receives inferred "knowledge event" by the information correlator so it can take further actions as designed in the process model definition. This helps the business process application becoming "context-aware". This capability is specifically used to identify wait states in a process and infer other location-dependent information to improve visibility over the process.

Process performance data will be collected in real time and internally stored in a dedicated database. It will, by the same way, act as a data provider for other system com-

[3] Labels with an SS*x* format (where *x* is an integer) are used to reference system components.

ponents or third-parties. Because of the natural evolution of operational processes in a given hospital context, the care process application should preferably provide functions for updating and maintaining process models. These functions will be used to define and maintain process models for patient care processes throughout their lifecycle.

SS2: Real-Time Location System. This component provides a solution for the location tracking of physical resources in a specified environment. Different levels of granularity can be achieved here. This component is used to track location of resources involved and influencing process performance across the hospital. Real-time location information is sent to the information correlator. For efficiency, filtering is done on the location events to be sent. We specify filters identifying specific resources, location update types and geographical zones of interest. This way, only relevant location events are collected. By tracking the position of resources across an entire hospital, we can predict patient wait states by detecting discrepancies between current and expected locations of resources of immediate need for activities that are location-dependent.

For patient care processes, the tracked resources are specifically patients and hospital staff, including physicians, nurses, transporters and housekeepers. Note that other resources could be tracked if needed. Filtering location information helps focus context awareness on the right information needed to accurately monitor processes.

SS3: Information Correlator. The information correlator has the particular functionality of inferring new high-level information by *correlating* captured process performance information, current expected process states, and context-aware information. In other words, by knowing what happened (performance), what is supposed to be happening (next business action), and what is currently going on (context), this component is able to infer divergence in the behaviour of the physical care process compared to the expected one. In other words, it can be used to detect discrepancies between the expected location of a resource and its actual one, thus recognizing the impossibility for the dependent activity to be executed. That is why this component is specifically used in this architecture to detect and monitor wait states (for wait times) occurring whenever a verifiable condition is not satisfied at a specific moment.

SS4: Process Performance Reporting. This component continuously displays process execution information in a way that is visually consumable by humans such as hospital managers or access and flow managers. This is a typical and traditional information system or a business intelligence engine.

SL: Subsystem Links. Links between subsystems in the architecture are realized using web service technologies. Web services implement communication functionalities of subsystems. More precisely, they define, for a given subsystem (e.g., process application, information correlator or location system), an interface that exposes opportunities of interacting with the subsystem. The interface is thus made of operations that can be called by another subsystem in order to achieve a certain interaction. Each subsystem thus communicates by calling web service operations of other subsystems' end-points. For example, the information correlator subscribes to real-time location events using the web service operations exposed by the RTLS. The business process application and information correlator conversely call each other's web services to exchange information. Each web service operation corresponds to a unique message definition and its parameters correspond to the data transported by the message. A WSDL file is defined for each subsystem such that each web service operation corresponds to an event that the subsystem can expect to receive. The use of web service technologies contributes to a Service-Oriented Architecture (SOA) approach and improves the interoperability between components, promoting loose coupling and reusability.

This briefly presents an overview of the canonical structure needed to adequately monitor patient care process performance. The next sections provide further detail for each component and the activities they involve.

4.3. SS1: Care Process Application

The care process application is a contained solution for prompt process data collection. This component plays a major responsibility because it is the main point for information capture in the system. Information is collected by the process application at the point of care (traditional paper filling is replaced by process-driven technology). More specifically, user interface (UI) report pages are built in accordance to the clinical processes or care pathways to be followed and are offered to caregivers at the right time to be filled. Those report pages include patient medical information needed by the caregiver to complete the task at hand. Process automation helps support this view, and process models are developed according to the expected workflow. This component interacts directly with business actors (e.g., nurses, physicians, etc.) providing support (e.g., data entry and data access) on operational processes and allowing information generated from each activity to be collected in real time. This information is then relayed to the relevant hospital information systems (HIS). In addition, for each necessary activity in the workflow model, the required medical information to be displayed is extracted from the HIS. Figure 12 shows the design of this subsystem.

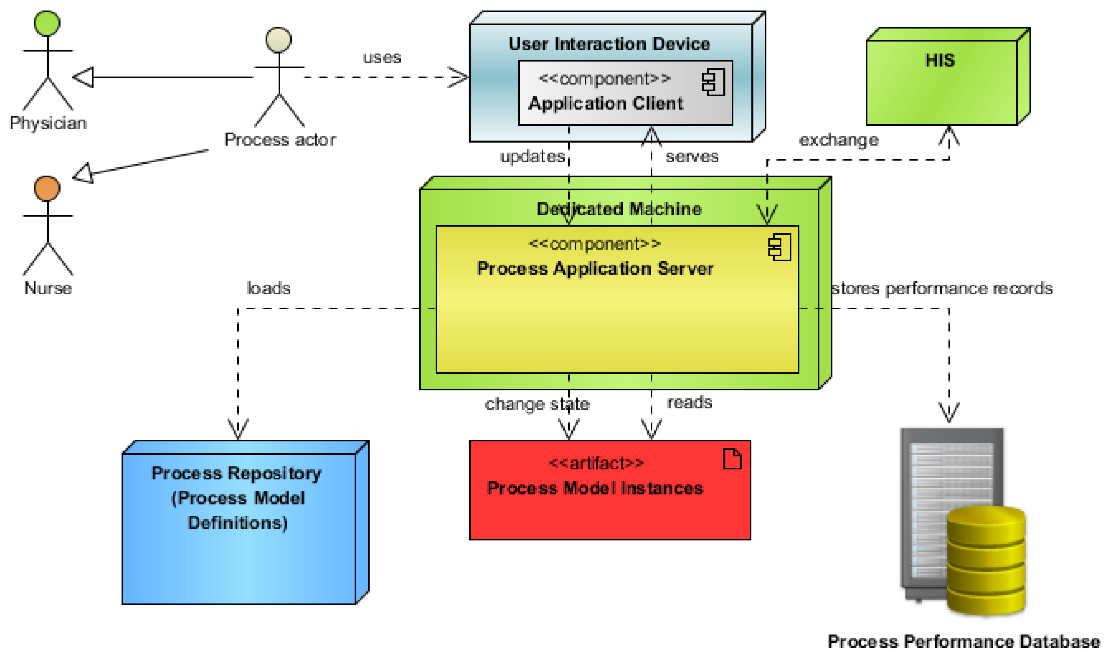


Figure 12 Care Process Application Subsystem

The architecture leverages the following elements:

- **Application Client:** this client is a software program used for user interaction. The client reads data about the process instances to display a relevant user interface for business actors to complete their tasks.
- **User Interaction Device:** the device runs the application client that allows the business actors (physicians, nurses, etc.) to complete process tasks, driven by care process

automation. This is mostly intended to be a mobile device such as an iPad, an Android tablet or similar.

- **Process Models:** These models are built using a workflow (executable business process) design environment and saved to the process repository. They encode detailed information about the clinical processes composing the care flow for given patients. They are the central piece on which process monitoring builds. Process models are application dependent; their definition is dependent on the particular hospital or department being studied as well as on the nature of the information that needs to be collected.
- **Process Model Instances:** a process model instance is a runtime instantiation of a process model definition. In the case of clinical process models, each process model instance corresponds to a patient being treated in the hospital as part of a given clinical pathway. The runtime model state for an instance is updated as its activities are completed by business actors. Collected data is stored in the process performance database.
- **Process Application Server:** manages the runtime execution of code for user interaction and system interface interaction (web service communication from other systems). This server is also a business process engine as it manages execution of process models (workflows) by handling the creation, update and destruction of process model instances. It automatically stores, as they are captured, process performance records in the performance database. It is the orchestrator of process execution in the system.
- **Process Repository:** The process database is the database containing all information related to process model definitions and process server configuration. The process server relies on this database to run and manage a given process model instance adequately.
- **Process Performance Database:** This database is used to record all process performance records. It holds all monitored information about all executed processes and is solely designed to manage process performance data. It can provide database views as interfaces for third-party tools to access those information (e.g., for business intelli-

gence purposes). Of course, because of privacy concerns, information access should be tailored and limited accordingly.

- **Dedicated Machine:** the system would ideally be running on a dedicated machine to ensure it handles the high load of computation typical of the healthcare computing domain (presented in Section 3.5).

There are three activities involved in managing the process application: process modeling, process execution and process monitoring.

4.3.1 Process Modeling

A process model is needed for two primary reasons:

1. As a descriptive artifact of its corresponding physical care process.
2. To create process model instances that will support care process execution, and from which monitoring of performance results can be enabled.

Another requirement for the process model is that it should be easy to maintain. Therefore, we need to build a model that describes the workflow of physical patient care process, and that is detailed enough to also model the structure and type of data manipulated by the different process activities.

By patient care process, we mean the complete workflow of activities required to take care of a given patient. Hospitals generally have specific procedures to be followed. This makes it easier to collect information regarding how things are supposed to be done. There typically exist separate clinical workflows depending on specific patient conditions and symptoms. Those are called *clinical pathways*. The use of clinical pathways in hospital is for providing a standardized way of conducting care processes. Pathways are claimed to improve clinical process outcomes by limiting variability in clinical operations (Tongchuan and Deyu, 2012). A clinical pathway represents the set of clinical activities to be undertaken on the patient in order to treat him. Clinical pathways are adequate inputs for process modeling since they capture the necessary clinical activities that need to be followed according to a specific patient condition and/or CTAS level (Canadian Triage and Acuity Scale).

A modeling language also needs to be chosen. For a particular given pathway, we capture the information defining the clinical workflow, activities and data associated with it. The workflow is usually specified in a textual (or graphical) and descriptive form. The modeling language also needs to be supported by the process model designer. For our work, we used the BPMN language (OMG, 2011a) because it supports all the process constructs that we need to capture clinical path flows, it is extremely intuitive to understand, standardized and possesses a wide community for resources. Note that it is important for the process model to capture the care process as it specifically occurs for the given hospital. This is part of the process discovery step presented in Section 2.2. However, clinical pathways are insufficient for process model specification. Many refinements are needed to provide additional details for process model design. Those complementary information can be captured through staff interview and plain observation of the way activities are done. The latter is usually the quickest to reach a given consistency. A combination of the two is usually best. Once the process model is built, it needs to be reviewed by business analysts, hospital managers in charge of clinical processes (e.g., access and flow managers), and business actors (e.g., nurse, physician) to ensure compliance to processes in place. Such a process also ensures a common alignment among stakeholders regarding the functioning of their processes.

The knowledge of context information (such as location) can enhance process modeling activities. One way of understanding this statement is to imagine that process models built solely on "process knowledge" will usually bear many limitations; especially when the physical process being model is of high variability in execution. For example, considering we have a process for which, depending of the location of a resource A or a resource B, a task A or B (respectively) of the system is executed. On one side, without any ability of knowing the location of the resource, the process will probably execute a generic task C which do not provide the necessary flexibility. Another option will be to have a task asking the user about the choice for task A or B. However, in most business context (particularly in healthcare), this approach will likely be inappropriate and costly. On the other side, with the knowledge of the location of the resource, the process model can be designed to support both options (A and B), thus providing the needed flexibility for the process to execute efficiently. This is a simple example showing the advantage of context-aware business processes over traditional ones. Context knowledge provides more flexibility to the process, thus allowing to optimize process flow. For example, given that a physician is requested for a patient consultation, and knowing that the location information about that physician indicates that he is not present in the patient room, the system will prevent the expected task of patient consultation from being displayed to the physician's device until he reaches the patient room. This helps streamline the care processes and prevent medical errors by exploiting process' location dependence in the implementation of process modeling adaptation techniques.

4.3.2 Process Execution

Once the process model is designed in the Care Process Application, it can then be run. In many business process management systems, the constructed model is deployed and instrumented as a process-driven web application. When a new process instance starts, the next activity to be performed according to the process model is made available to the specified user group or user. The user logs in and sees the activity as a task in his/her list. The user can then complete the task and validate the execution of the activity. The input data provided (if any) during that activity is captured in the process model instance and stored in the process performance database. Each activity's start and end time can also be captured to measure service times.

4.3.3 Process Monitoring

Process monitoring capabilities include that, at any given time, current process state (task running, waiting state or error state) is accessible, along with some history about the previous process states. This is naturally supported by the care process application since process state is automatically recorded during execution: Process state is saved in the form of process data to the process performance database as process executes. In order to provide a monitoring view to the process performance reporting subsystem (SS4), data needs to be fed by the care process application. To do so, web service integration is used to send the relevant and programmatically-defined output data. More precisely, at the process model definition level, web service integration modules are directly embedded in the process model definition and called during execution. Another alternative would have been to have an outbound messaging construct starting a new process whose sole purpose would be to pass the necessary data to the web service integration module. However, this indirection has been judged to be potentially costly, considering the overhead involved in starting a new process instance and the amount of messages that may be needed at a given time in a real scenario.

4.4. SS2: Real-Time Location Tracking

This section describes the RTLS subsystem. More precisely, it focuses on what the functionalities of this subsystem are, and on what we need to design in order to implement those functionalities.

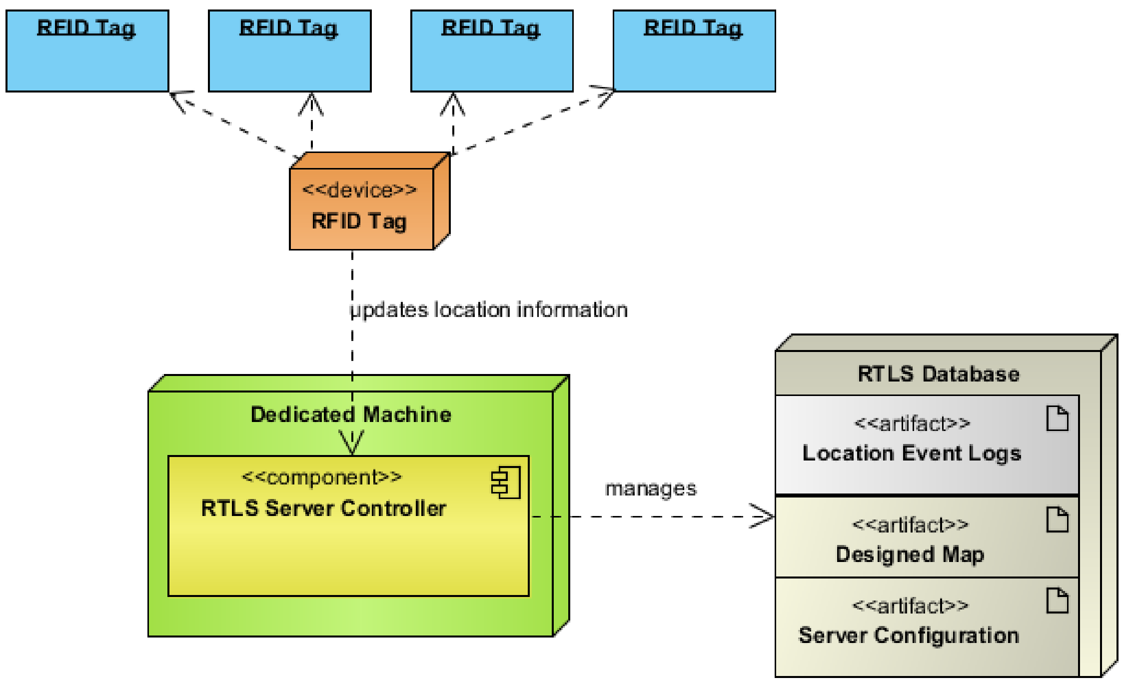


Figure 13 Real-Time Location Subsystem

As presented in Figure 13, the real-time location subsystem is composed of the following elements:

- **RFID Tags:** Tag are rechargeable RFID devices used to track physical resources. Tags are configured with operational and connectivity settings such as the type of location updates, frequency of location updates and server communication information (network access credential, network channels, server IP address, etc.).

- **RTLS Server Controller:** The server helps manage tags and other RTLS assets such as all loaded maps, the active map, user administration, etc. It also enables monitoring tag location information and accessing system event logs. The server is primarily responsible for computing received raw data information in real-time using various complex algorithms, in order to infer the actual location of the emitting tags.
- **RTLS Database:** this database is a traditional relational database that stores all received location information, logs, tag list and configurations, as well as a user list. It also contains designed map models, which are necessary to situate location information with respect to the physical world. This implies that maps need to be "faithful" representations of the real geographical spaces where the location tracking occurs.
- **Dedicated Machine:** A dedicated machine is ideally required to provide enough server resources and power to support tracking of assets and people in an entire unit or across the entire hospital.

Using the tags, the real-time monitoring system can receive information about the tracked resources. In practice, such information is usually some data (e.g., signal strength of access points or infrared information between the tags and an infrared emitter) computed to infer the position of the resource. That position is relative to a point of origin in the geographical space. This is why the designed map model needs to reflect, in adequate proportion, the dimensions and specific geophysical characteristics of the environment where the tracking occurs. The RTLS subsystem should also ideally provide an interface (in our case, a web service) for third-party applications (e.g., information correlator) to access the server functionalities including location information. This subsystem can be supported by existing location tracking solutions, which already provide most of the presented functionalities and mechanisms.

4.5. SS3: Information Correlator

This subsystem correlates location data and process state updates in order to identify patient wait states. Specifically, some rules need to be designed that define the conditions on the process state and the circumstances on resource locations under which patient wait states can be accurately inferred. The information correlator makes use of those rules to infer, in real time, new process-related events (which are more abstract) including patient wait states. To meet the “real-time” constraint, the process has to be a continuous mechanism assessing arriving streams of information (location updates from the RTLS and state updates from care process application) against the rules each time the knowledge base is updated and sending out inferred knowledge events to designated elements. Ideally, this component also needs to provide management functionalities for the inference rules (e.g., edition, deletion, structuring). This is best done by using a Complex Event Processing (CEP) engine.

4.5.1 Correlation with Complex Event Processing

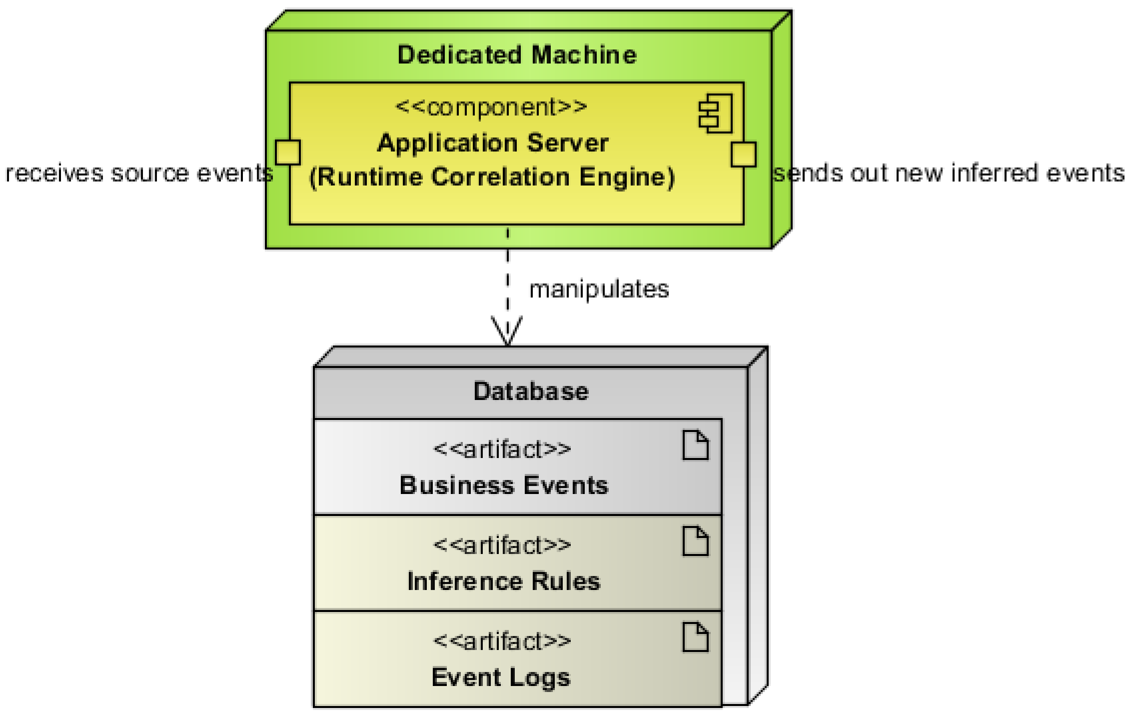


Figure 14 Information Correlator Subsystem

A CEP engine is a tool built to process streams of information representing events that occur in a given environment. It receives events from various providers and correlates them in real time to infer new intelligible events (complex events) representing *actionable* information. In our work, this capability is applied into the system by inferring meaningful facts about process state (namely patient wait state) through the processing of events related to the location of process resources and the state of care process execution. Actions are then taken by the CEP upon recognition of those new facts. Figure 14 presents the different elements involved in this subsystem:

- **Application Server:** This server hosts the runtime correlation engine that implements the real-time mechanism for evaluating incoming events against rules and generating and sending out new events to other systems. It also provides management of runtime execution through monitoring of received events (also called *source events*), output events and the inference rules that have been "fired" in the process (with timestamps).
- **Business Events:** These artifacts model the different types of events that they receive or generate. Attributes have to be defined for all event types needed by the system. For example, the attributes for an event named "PatientInED" used to inform that the patient has entered ED include: Patient_ID, Location_ID, timestamp. Those definitions are used to manage the events "understood" by CEP and are also referred in the definition of inference rules.
- **Inference Rules:** They are artifacts modeling the nature and semantics of the rules we need to have to infer new events. A rule definition primarily consists of: a ***triggering event*** and ***inference logic***. A triggering event identifies the event which, upon reception as a source, triggers the evaluation of the rule. This evaluation consists in the checking of the inference logic (usually a conditional expression). The rule is evaluated against the knowledge base of *facts* historically available (historical events). If the premises of the logic are validated, then the resulting/conclusive knowledge information can be inferred. An action is triggered that typically consists in one or more output events to be created and dispatched to appropriate destinations.
- **Event Logs:** those logs are recorded by the system as things occur. They are system events.
- **Dedicated Machine:** a dedicated machine is preferred to ensure suitable allocation of system resources (memory, processor speed and other configurations) for a given hospital and facilitate system maintenance.

Most CEP engines allow rules to be grouped by *sets* for easier management. Many rules are defined that represent the intelligence of the CEP to make valuable sense of its environment. The access by the care process application to those new pieces of knowledge allows process monitoring to be fine-tuned and process automation to be adjusted towards better productivity and efficiency.

4.5.2 Correlation Rule Definition for Patient Wait Times

For the specific objectives of identifying patient wait times, the logic of the inference rules underlining the work of detecting wait states revolves around the following principles:

1. Each care process activity required specific conditions to be satisfied in order for it to be conducted. Those conditions could be about the value and availability of certain data (e.g., HIS data), the presence of a specific human resource at a specific location and/or the occurrence of a timing event. The conditions or constraints are generally on the nature and the status of the resources influencing the execution of the activity.
2. Patient wait states in the hospital can be inferred for all location-dependent care activities. A location-dependent activity is defined as an activity which, in order to be executed, requires the location of at least one of the resources involved in that activity to match a specific geographical location in the environment context. In other terms, a wait state is observed whenever the provider is not at the expected location at the time the activity needs to be executed. This can result in delays and possibly bottlenecks if no resolution action is promptly applied.

This logic of *state detection* is thus implemented in the form of inference rules and automated using the CEP engine: the wait state is said to be inferred whenever the expected location of the resources (necessary conditions) involved in the activity to be executed is not observed (discrepancy).

4.6. SS4: Data Capture and Reporting

Figure 15 shows the reporting subsystem. This is a traditional information system for displaying process information. It receives data directly from web service integration with the care process application as opposed to reading the performance database. This is to maximize time efficiency and satisfy real-time constraints. Database's pull operations can sometimes be really slow and cumbersome for real-time applications since they correspond to file system access. Our approach is to not have the reporting subsystem initiate data requests and use it as a data consumer tool. Process information is sent to it by the care process application module as soon as data becomes available. This allows the system to operate in real time or near real time in an efficient way.

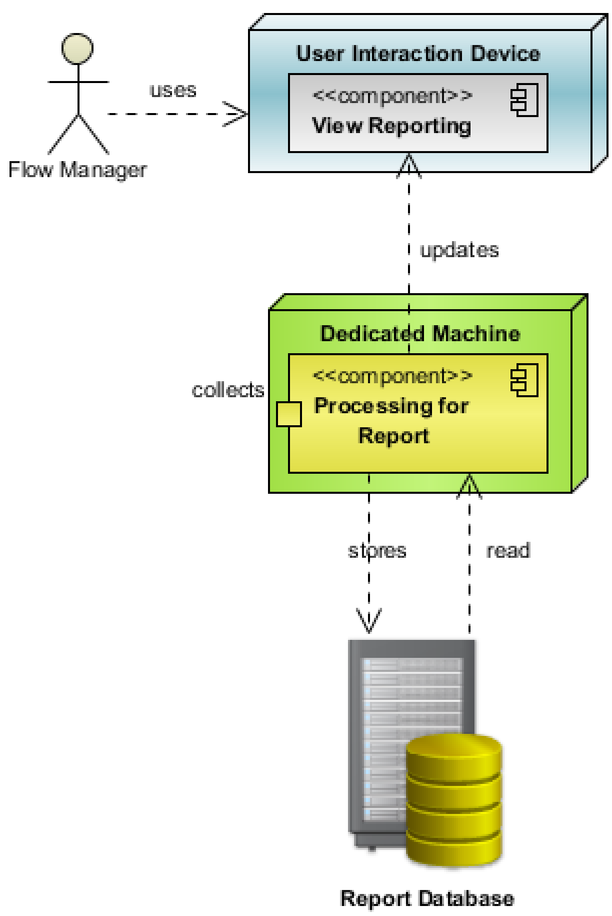


Figure 15 Reporting Subsystem

4.7. Detailed Architecture

Now that all areas of architectural design have been presented, the following detailed system architecture (Figure 16) summarizes what has been presented. It particularly highlights all the identified project-specific artifacts (in green in the figure) that would need to be developed across the LA-BPMS to enable its use; namely: process model definitions of real life processes, map designs of the real physical context environment, server configuration for RTLS equipment, business events identifying incoming and outgoing events on the correlation engine level, and inference rules representing the logic used to infer new knowledge information.

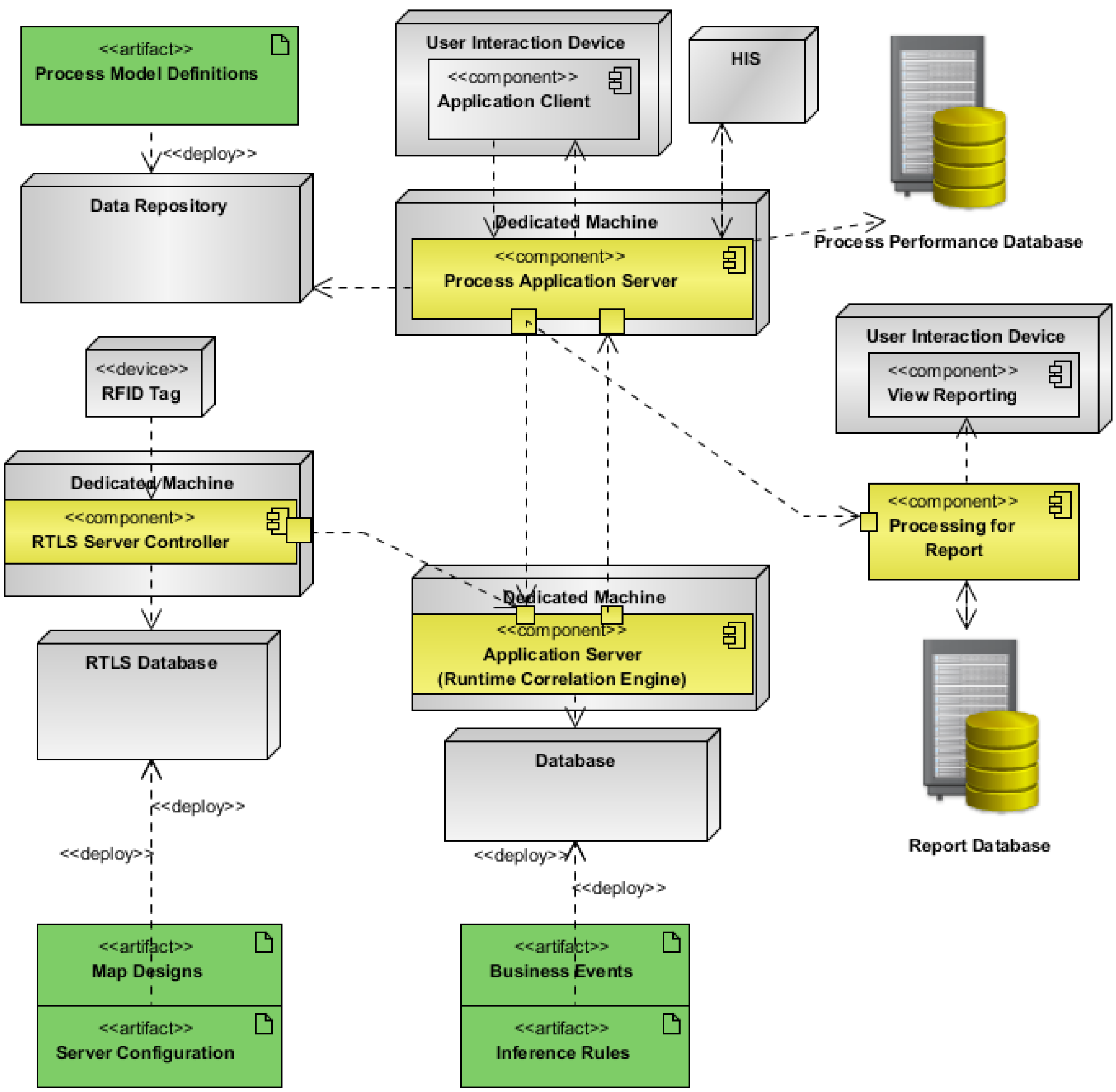


Figure 16 Detailed System Architecture Highlighting Development Artifacts

4.8. Design Discussion

The architecture presented and the technologies chosen so far allow us to understand how this solution can be used for real-time monitoring of patient care processes. However, there are various concerns related to this platform that are worth discussing. An evaluation of the quality of the design architecture is still required and is discussed in this section. The major areas identified as important discussion points are:

- **Point of Data Collection:** This refers to the trade-offs related to the choice of the point of data capture (for process monitoring) in the information flow route in a hospital. Two particular points of data capture are compared: at the actor level and at the HIS level.
- **Data Quality:** This relates to the potential quality of the data captured and output by our system. We will compare them later in the thesis to current practices.
- **Process Model Flexibility:** This recognizes the complex nature of human-driven clinical processes and discusses flexibility modeling techniques used to deal with it.
- **Event-Based System Integration:** This pertains to the system design qualities of the event-based approach to system integration.
- **HIS Integration:** The specific challenges related to the interaction of the LA-BPMS with HIS are also analysed.

4.8.1 Point of Data Collection: Actor or HIS Level

The integration of the BPM component with the hospital system environment presents different alternatives that are worth examining. The reason lies in the way our architecture captures process data. We need to identify the possible points of capture. So, we shall first understand how process information "flows" currently in hospitals.

If we recall the information in Figure 10, every new data information generated by a given care process activity (e.g., data information about the patient physical condition during a "physician consultation" activity) is typically collected at the point where the activity takes place and is relayed to other actors (e.g., nurses and clerks) in the information chain. The data is commonly manually entered and stored into dedicated HIS. Such HIS are organized by departments and business nature for efficient management, and accessed by dependent parties (e.g., Lab IS, Patient IS such as Meditech, PulseCheck, etc.).

This allows us to identify two primary possible points of source for the data in our system: at the human actor level, or at the HIS level.

- **At the HIS level:** Capturing data from the HIS can make the monitoring process "invisible" and non-invasive in the hospital environment. However, data collected at that point might arrive late, and may be inaccurate. As we know, data can be entered manually in the HIS several minutes or hours later after the fact. This will be an unacceptable mechanism for a real-time monitoring system where the system is expected to provide updates in near-real time (e.g., every minute).
- **At the actor level:** If the collection of data is done at the actor level, then we have a more direct connection at the point of capture where care activities actually take place. However, this approach is more "invasive"; the monitoring system is brought at the forefront. This also implies that using process automation, data is captured and soon as it is entered by the qualified human resource right at its point and moment of creation. With such a design, an issue might occur: The human actor will now have to manually report information to 1) the monitoring module, and 2) the HIS module he/she usually updates in the process. Thus, to avoid this redundancy of plain data entry work, the monitoring system will now have to ensure the relay of the captured data to the appropriate HIS. This new procedure will replace traditional paper-based data capture but as a result, will hold the critical responsibility of ensuring information flow between actors and the HIS at the required times.

In both cases (actor level and HIS level), an integration of our system in a healthcare environment necessarily involves interacting with existing HIS systems. Depending on the task (storing information or displaying it), the system acts as a **data provider** or as a **data fetcher** for the HIS. Particularly, ensuring that collected data is fed to the appropriate HIS is extremely important for transparency and reliability. The necessary information about "what data goes to which system" has to be captured at design time. The automation of communication with the HIS for data persistence is also an improvement compared to current practices (see Section 3.2).

4.8.2 Data Quality

We are interested in understanding and assessing the quality of the data captured through our architecture. This platform provides a high potential regarding data accuracy, completeness, accessibility, timeliness, granularity and reliability, each of which being discussed in this section.

Accuracy

Making sure that the data is accurate is a must, especially in an environment where software defects are to be strongly avoided. Medical professionals rely on data information to take decisions regarding patient health. Thus, it is clear that accurate data is a hard requirement. For data information actually generated by our system, we will need to demonstrate to which level we meet accuracy and how this impacts our objectives. The generated data includes location tracking information, location events, process state, process state events, inferred patient wait state and patient wait state events. Assessing accuracy is an engineering task of verifying system implementation and this can be handled using traditional testing methods. The system's accuracy in our case highly relies on the data accuracy provided by subsystems. For the process application and the reporting system, the accuracy of data captured and displayed depends on their ability to maintain data consistency throughout their components. For the information correlator, the accuracy of new generated output events depends on the accuracy of its inference rules. For the RTLS, the accuracy of location data depends on the location tracking technology and on the logical artifacts (e.g., map design, filtering configurations) used to infer location events. Also, it is fair to argue that for data that does not originate from the system (e.g., nurse entering the data), the LA-BPMS can only guarantee to maintain the consistency of the quality of the collected data the way it was provided.

Completeness

Compared to other technologies, the amount of unique process-related data that LA-BPMS can potentially capture arguably outperforms existing solutions. This was the primary intent of this system, to provide as much insight as possible on actual care process execution. The system provides many views of relevant information about the process, mainly the clinical state of affair and the patient wait states.

Accessibility

Data accessibility is ensured by making appropriate data accessible at relevant levels of the system. An integrated view of the collected data is accessible from the performance database. The choice of database technology has a decisive impact on flexibility and reliability of data access.

Timeliness

The real-time property of the system ensures a suitable timeliness on the availability of data. By using real-time system functionalities such as real-time location tracking, real-time process monitoring and the continuous mechanisms from the information correlator, we ensure that data is delivered as soon as it is input to the system. Compared to, for example, a paper-based approach, the data collection process is less cumbersome and naturally faster. The fact that the collection process is automated creates unique benefits that, along with data completeness, help fulfil major objectives of this system.

Granularity

The level of granularity depends on what is needed from the business users of the system. The system can get as granular as the technology allows it, as long as concrete requirements are captured. For example, in the context of a patient waiting for a physician's initial assessment (PIA) after a triage activity, a granular visibility can be achieved by having access to all the different process "micro-steps" occurring during the patient wait state. More precisely, the system can monitor and inform the manager whenever after the triage, 1) the nurse actually sends a request to the physician, 2) the physician acknowledges and accepts the request, 3) the physician actually arrives at the patient's room for the PIA Such a degree of granularity in the process information, which combines the power of process monitoring and location awareness, allows hospital managers to understand the evolution of their processes and of the patient wait states. The hospital manager can then initiates troubleshooting actions based on his granular visibility on the state of his care processes.

Comparatively, today's data collection techniques only capture a single patient wait time data measuring the time from triage to PIA. Such information does not provide any sufficient insight for action, and it is also likely to be available long after the incident, preventing its timely use. The level of granularity of our proposed system can only be achieved by tracking location of resources across the hospital and by knowing the current clinical process state for the related patient.

4.8.3 Process Model Flexibility

There are numerous existing challenges pertaining to modeling a clinical process. Here, we explore the concerns related to the design of an accurate, usable and flexible process model for process execution.

The problem of flexibility is an issue intrinsic to the activity of modeling processes to support automation. van der Aalst et al. (2009) discuss this issue as being a trade-off between process support and process flexibility. The problem comes from the fact that the process model is meant to be executed, i.e., it is instrumented to follow and provide support to a corresponding physical process. Hence, the process model state needs to reflect the real state of the physical process at any given time. This is where the challenges of modeling a process come into play: adequately abstracting the physical process behaviour using available modeling language constructs. Important questions include: What happens if the physical process slightly deviates from its intended course of execution? How should such situation be managed? How about an unexpected behaviour occurring at the physical process level?

The flexibility of a process model is its ability to support/capture the multiple possible paths of execution (*expressibility*) of a real-life process. In process automation, those situations need to be dealt with in such a way that the process model state remains synchronized with a physical process that can be quite dynamic. Also, such support needs to be handled in a manageable way (given available computing resources). To achieve this, a simple approach is to use a common process flexibility technique called *flexibility by design* (Schonenberg, 2007).

Process Flexibility by Design

Flexibility by design is a technique that consists in identifying and modeling the possible *known* execution alternatives of the physical process at design time. This is usually realized by using control flow principles such as (Schonenberg, 2007; Reijers et al., 2010): parallelism, choice, iteration, interleaving, multiples instances or cancellation. Each of those notions provides expressive power to abstract the process' variant behaviors. One example of application of flexibility by design is the execution of medical test orders for a patient in a cardiac care process. In the cardiac care process, the physician typically orders, during the initial consultation, for at least an ECG and a blood sample collection to be done on the patient. Those two tasks, which will be completed by the nurse in charge, do not need to be done in a specific sequence; this is a degree of flexibility allowed by the process. In order for our model to support this characteristic, the "parallelism" design principle will be applied to provide flexibility to the process control flow as illustrated in Figure 17.

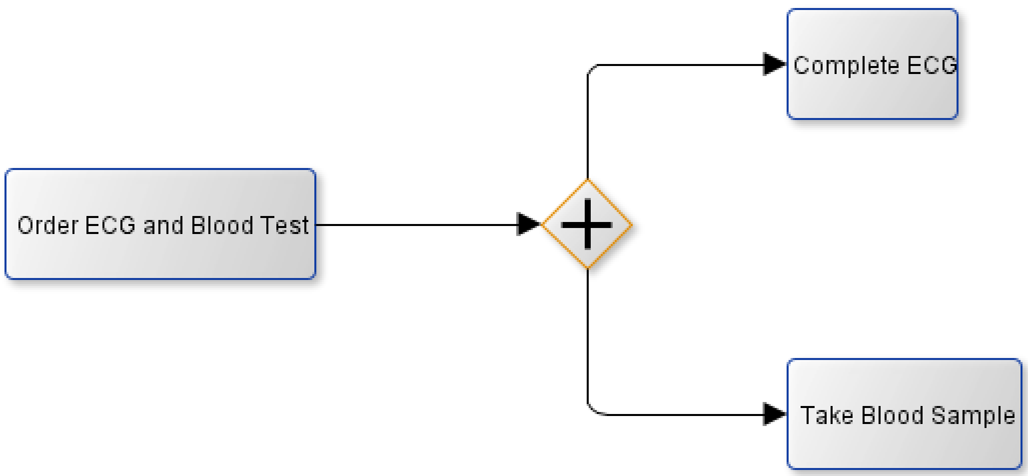


Figure 17 Example of Flexibility by Design: Parallelism

One important thing that this example highlights is that flexibility of a model can be achieved only to the extent of what flexibility principles are supported by the tools employed (e.g., modeling language, BPMS). In BPMN, flexibility by design will be limited to the use of gateway elements (e.g., AND, OR, XOR), which also support iterations indirectly.

This approach can be sufficient for healthcare processes, which are generally strongly regulated/structured, while providing flexibility only in specific areas of the process. However, clinical processes are usually driven by patient care objectives, and they can sometimes be dynamically altered (e.g., a procedure/activity needs to be escalated). The risk in using flexibility by design for such processes is to get to a model that is highly over-specified, i.e., a rigid process model that trades complexity in size for flexibility.

This is a major issue for the performance and maintainability of the process over its lifecycle and is to be generally avoided.

Process Exception Modeling

Giving flexibility to processes can easily result in other issues, namely exceptions. In process automation, like in any other computer program, exceptions need to be handled adequately to ensure system reliability. Process exception can be referred to as "deviation from the optimal (or acceptable) process execution that prevents the delivery of services with the desired (or agree) quality" (Grigori, 2011). In process modeling, this consists in identifying the areas of process behaviour where potential exceptions need to be captured. Exceptions allow the process model to handle properly possible errors in the physical process without compromising its execution. They are used for "modelling undesirable, abnormal or irregular events of such a nature that the process cannot decide itself how to continue" (van Hee et al., 2008). For example, imagine the system fails to update a HIS for unknown reasons (e.g., maybe the system connection is down), the process can log the exception information and gracefully terminate, follow other necessary tasks (trigger a recovery or alert process) or request human intervention.

Combining the Two Approaches and Beyond

The combination of exception modeling techniques and flexibility by design techniques allows us to create process models that can *consistently* capture process behaviour at the adequate level of abstraction needed to provide effective and efficient support. Characteristics of a process that are known at design time can be properly supported by combining process modeling techniques, process flexibility and exception modeling. However, behaviour of a process that is not known at design time cannot be addressed by the precedent techniques. For example, a contextual change in the real-life process can force an educated actor to process the next steps in the process in a way that could not have been planned by the process model (because unknown at design time). Those more advanced complex cases require techniques that allow the system to adapt at runtime using, for example, context awareness mechanisms like those presented by Sell C. and Springer T. (2009) in their paper. The authors present an approach that consists in automatically calculating workflow adaptations and semi-executing them on an adaptive workflow management system (WfMS). Workflow adaptations are calculated based on the changes in the context. Then, they are suggested to the user and upon user confirmation, are dynamically integrated to the designated workflow at runtime. The authors advocate the use of an adaptation layer to separate adaptation logic from workflow logic, facilitating reusability and code management.

4.8.4 Event-Based System Integration

Building a LA-BPMS to support an entire process of patient care across hospital is a major challenge, especially in terms of robustness and computing power. Because of this, the system is to follow a distributed deployment infrastructure allowing subsystems to be assigned dedicated computing machines and to exchange events over a computer network using messaging interfaces such as web services. The latter provides enough flexibility and modularity for the system to operate reliably while being easily maintainable. The dedicated computing machines also allow each subsystem to be optimally configured and to be provided dedicated computer resources. Moreover, due of the modularity of the architecture, each subsystem can be tested independently of others prior to creating contracts between them using a “communication event protocol” that defines the nature and structure of the events to be exchanged between the subsystems. Test mock-ups can then be developed to simulate other components in order to test a single subsystem independently. Robustness is a critical requirement given the high information load existing in the healthcare environment, and the critical nature of the information being exchanged.

4.8.5 HIS Integration

For an automated approach to be developed, the HIS needs to allow for other systems to access it, and an interface needs to be built and managed. Collected data will have to be stored in the corresponding HIS by respecting the format of data communication between the systems. Some existing HIS use rather old technology and can be cumbersome to interact with. However, most of them use well-standardized protocols. For example, most patient medical information system use the Health Level 7 (HL7) encoding protocol. HL7 is a standard from Health Level Seven International (2013) that defines the structure and data types used to encode messages and exchange information between medical information systems and equipment. Such protocol needs to be implemented by the real-time monitoring system in order to enable interactions with existing systems. Data to be sent or received from the HIS will have to be converted (respectively) to or from the HL7 format. Given the potential size of such components to cover the numerous existing HIS in a given hospital, separate adapter modules will have to be defined to specifically handle this task. There exist numerous HL7 APIs that could be investigated. However, this task is outside the scope of this thesis.

4.9. Chapter Summary

This chapter presented the central idea of this thesis. More precisely, an architecture for real-time monitoring of patient care processes has been unveiled. Goals and requirements for such system have been identified and traceability from system capabilities has been provided. The Goal-oriented Requirement Language (GRL) model Figure 18 is used to graphically summarize traceability relationships between goals of the system from Table 1 (yellow GRL goals:), system requirements from Table 2 (green GRL goals), and subsystem functions identified in the system architecture of Figure 11 (blue GRL tasks:). This model also describes the impact of each subsystem to the requirements (arrows with contribution values on a 0..100 scale) and in turn to the goals associated with the two main stakeholders (GRL actors:), namely BI analysts, and access and flow managers. Note that not satisfying any of the subsystem will lead to a situation where at least one stakeholder will not be entirely satisfied.

The solution architecture was presented at different levels of detail and a comprehensive design discussion has helped understand the broad challenges, rationales, strengths and limitations of the solution approach.

The next chapter provides a comprehensive and concrete implementation and evaluation of our architecture in the form of a case study involving an Ontario hospital.

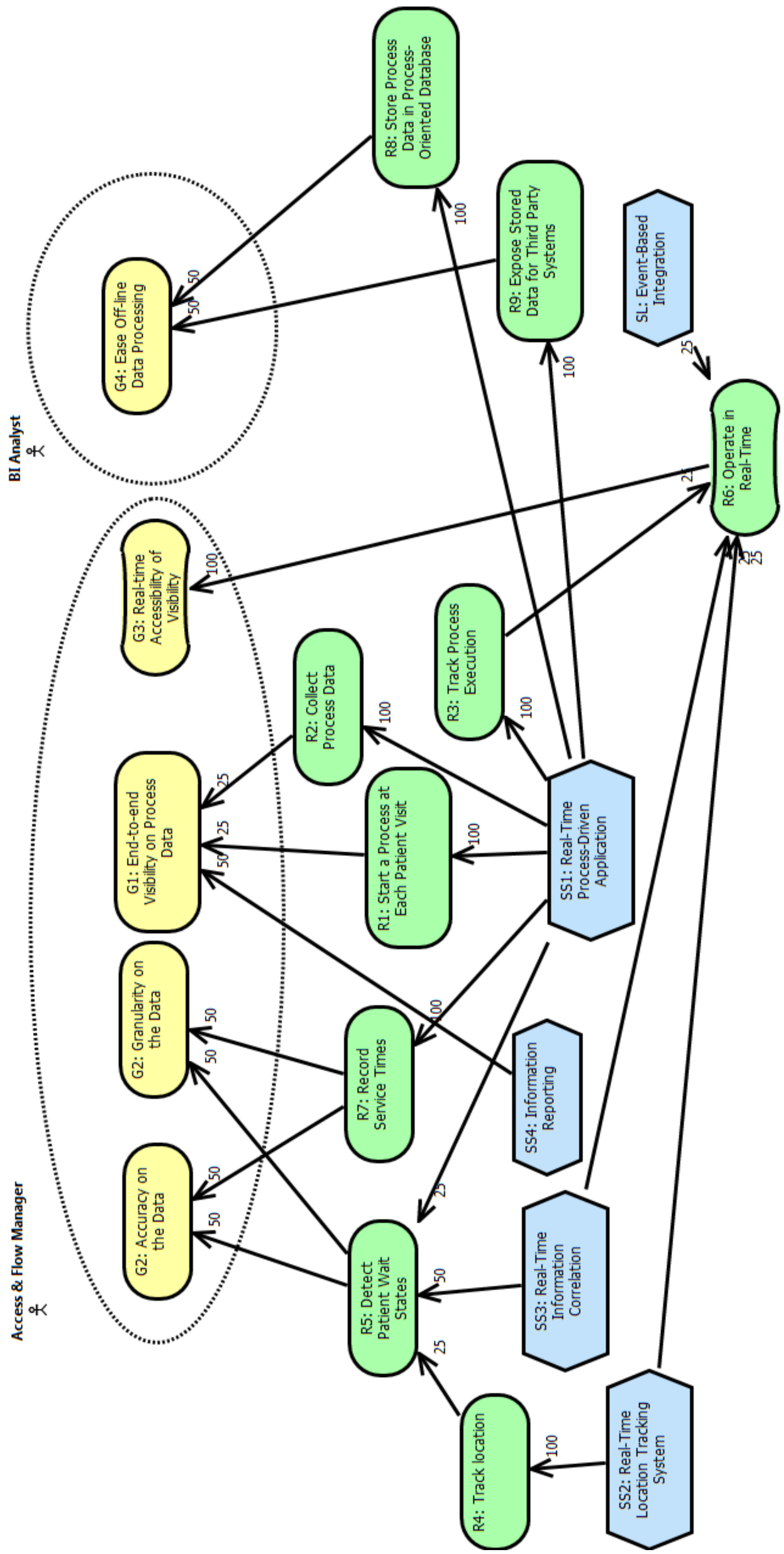


Figure 18 LA-BPMS GRL's Model

Chapter 5. Case Study: Cardiac Emergency Patient Flow

This chapter presents the case study that has been conducted as part of this thesis. This project has actually been an important resource for the research, in terms of concretely understanding the research issues, extending our domain experience, learning about specific operating environment of a real hospital (doctors, nurses, patient behaviours, etc.) and allowing our work to be evaluated against a real problem context.

5.1. Overview

The case study was done at William Osler Health Centre in Brampton, Ontario, and was driven by a University of Ottawa research team. The hospital was interested in how patient wait times and patient flow management in general can be improved.

- **Background:** William Osler Health Centre is a community hospital located in Brampton, Ontario. The hospital possesses around 600 beds.
- **Subject:** Investigate how cardiac patient flow management can be improved using a system for real-time monitoring of care processes.
- **Goal:** Deploy an implementation of the solution system in an instance of the intended environment and evaluate how it satisfies the thesis hypothesis, i.e., to provide an accurate, granular, flexible and integrated end-to-end visibility over processes in order to support and enhance control over this patient flow in the hospital.
- **Procedure:** The wait times that have the most impact on patient flow and are of the highest interest were identified. Service times for certain activities were also tracked in real time. A prototype was built off the system and a simulated environment was designed, replicating real environment settings in order to demonstrate system operations. The reason a simulated environment was chosen instead of a real environment is due to the limitations of access to real hospital resources (including HIS systems, patients, and buildings). Moreover, even if access to those resources might have been

possible, this would likely have represented an important overhead for a prototype that was the first of its kind and would have increased the time of development outside of the amount of research time allocable. A simulated environment was judged to be sufficient to observe and evaluate capabilities of the system against the hypothesis and also towards improvement of the original ideas. The simulated environment was however deployed in two different locations (University of Ottawa lab, and at the hospital itself), with two different map settings. Note that because the prototype was not to interact directly with HIS, mock-ups modules were built to simulate those interactions.

- **Evaluation:** After the simulation setup, over 20 simulation sessions were conducted and demonstrated to various stakeholders of the hospital and partners. This was done during one work-intensive month at William Osler Health Centre that included a dozen demonstrations, as well as with another nine demonstrations between July 2012 and May 2013 at the University of Ottawa. System behaviour data was automatically logged by the system and surveys were filled by participants at the end of each demonstration at the hospital, allowing us to gather important feedback about the subjective appreciation of the system.

5.2. Context

The research project was conducted by a research team from the University of Ottawa, in collaboration with William Osler Health Centre and IBM Canada, with financial support from NSERC’s Business Intelligence Network, IBM, and MITACS. The objective for the case study was to demonstrate the feasibility of building a real-time patient flow management system that provides the following functionalities:

- Tracking of patient care process in real time using process automation.
- Measurement of patient state and bed state.
- Measurement of wait times and service times.

The project was scoped to tackle the problem of wait times specifically in the context of cardiac patients (where wait times are most critical) and following Osler’s *Acute Coronary Syndrome* (ACS) clinical pathway. This *clinical pathway* documents the steps of care to be completed for a given cardiac patient throughout his/her stay, from arrival to discharge. The pathway implements the concept of continuum of care. The *continuum of care* is a conceptual framework for a fully integrated system of health services to help guide and track the patient across the entire process of care. The benefits of its application are found at all entity levels with facilitating healthcare delivery and making efficient use of resource (Evashwick, 1989). This is also a concept for cost-effective, safe, patient-centered healthcare (Lyndsay, 2006) turned into an important tool for patient flow control.

William Osler Health Centre's facilities include an emergency department (**ED**) to deal with immediate and urgent threats to health. Other units cardiac patients might go through include the Cardiology Ward (**CW**) with beds, the Cardiac Catheterization Lab (**CCL**) and the Cardiac Care Unit (**CCU**). Figure 12 presents the different units with all the possible paths the journey of a given cardiac patient might consist of.

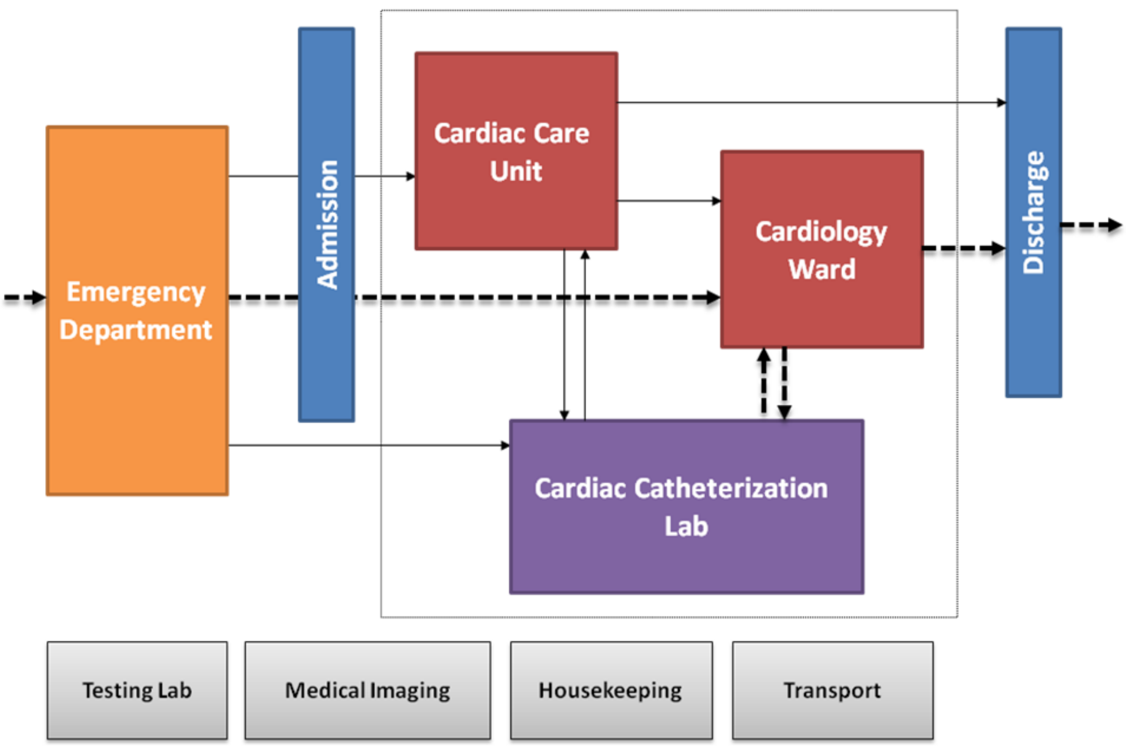


Figure 19 Cardiac Emergency Patient Flow at William Osler Health Centre[4]

[4] Note that the "Admission" and "Discharge" (in blue) are not physical units. They simply represent processes of transition in and out of the hospital (as it is the case for in-patients). Ancillary processes occurring during patient stays are represented at the bottom of the figure (in grey).

The dashed-bold arrows represent the path that has been covered and automated in the system prototype. This is essentially an end-to-end journey (from admission to discharge) of a cardiac patient across the entire hospital. This is critically important in order to get a complete view of wait times and care process behaviour for informed decision making.

5.3. Deployment Architecture

The following diagram depicts the deployment architecture chosen for this case study. It shows the technologies selected for each of the subsystems in the high-level abstract architecture of Figure 11.

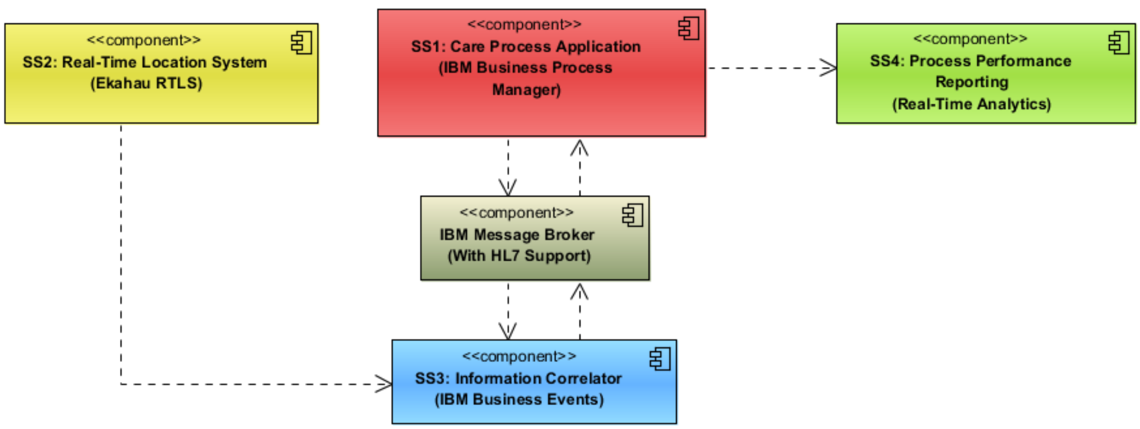


Figure 20 Overview of the System Architecture

This architecture shows the configuration that was proposed to Osler for deploying a real-time monitoring system in their particular settings, with off-the-shelf components whenever possible. In particular:

- A leading commercial business process modeling tool, namely IBM Business Process Manager[5] (version 7.5) is used as the Care Process Application (SS1). This tool supports the modeling and execution of business/care processes.
- Ekahau's RTLS[6], the leading solution in this space for healthcare, is used as the Real-Time Location System (SS2) and supports the tags, maps, events and database functionalities identified in Section 4.4.

[5] http://www.ibm.com/developerworks/downloads/bpm/
[6] http://www.ekahau.com/products/real-time-location-system/overview.html

- IBM WebSphere Business Events[7] (version 7.0) is the Complex Event Processing engine used as Information Correlator (SS3). Other more recent solutions such as IBM Operational Decision Management[8] could also have been used, but this choice was made by another student in the team prior to the availability of suitable alternatives.
- As for Process Performance Reporting, a tool developed by Baffoe (2013) and other students of the project (Baffoe et al., 2013), namely Patient Flow Monitor (PFM), was used.
- An additional component enabling easier communication between these subsystems and eventually with external Health Information Systems was also used: IBM WebSphere Message Broker (8.0), recently renamed IBM Integration Bus[9].

We were told many times by IBM collaborators that his combination of several major tools was unique in the healthcare process, and they were excited to see them integrated together in such a context. Note that interactions with Osler's HIS were mocked up, given the unavailability of these information systems for the purpose of this case study.

5.4. Hardware and Software Configuration

This section presents the hardware and software configuration that we used for this case study.

5.4.1 Hardware

The hardware equipment used for the prototype is as follows:

- Ekahau RFID tags (12): For location information capture. See the left part of Figure 21, where a tag is attached to the wrist of a patient. Tags can be washed and disinfected.
- Ekahau IR beacons (5): Standalone infrared emitters used for zone/room precision. See the right part of Figure 21, where a beacon covers exactly one bed. Beacons can be set to also cover an entire room or a very small area (e.g., one square meter).

[7] http://www-01.ibm.com/software/integration/wbe/
[8] http://www-03.ibm.com/software/products/us/en/odm/
[9] http://www-03.ibm.com/software/products/us/en/integration-bus/

Figure 21 Ekahau RTLS Tags (for PatientsStaff) and Beacons (for Rooms/Beds)

- Map visualization laptop: To display the Ekahau floor map of the simulated environment on a projector, with real-time location of people and devices.
- Two Apple iPad: Used for care process task completion and data capture.
- Patient flow monitoring (PFM) visualization laptop: For displaying the real-time dashboard for patient states, bed states and wait times.
- Visualization android tablet: To showcase the real-time dashboard on alternative devices and technologies.
- Single master server running VMWare iESX: 12GB, 6GHz, running all 3 virtual machines (VMs):
    - 1 VM running IBM Business Events and PFM.
    - 1 VM running IBM Business Process Manager (BPM), version 8.0.
    - 1 VM running the Ekahau RTLS server and IBM Message Broker (IBM MB).
- 3 dedicated access points and 1 router (Cisco): For enabling messaging exchange i) between system components and ii) between Ekahau tags and the RTLS management server for location tracking.

For this project, we had to work on developing the system at the University (Ottawa) and bring it to Hospital for simulation test (in Brampton) several times. Because of such particular settings, we chose to design a portable system. That is why a single physical server was used to host all components. Of course, this should be different in real life scenarios.

5.4.2 Software

The list of software tools used for the prototype is as follows:

- IBM Business Process Manager: used for process modeling, execution and management, fulfills the role of the Care Process Application (Figure 16).
- Ekahau RTLS: this complete RTLS solution from Ekahau is used for real-time location tracking of tags and fulfills the role of the Real-Time Location System (Figure 11). It is composed of:
    - Ekahau RTLS Controller (ERC): a software server for managing tags, and triangulating the location based on a tag's received information. This implements the RTLS Server Controller presented in Figure 16. ERC also has a Tomcat database to manage all the data it manipulates.
    - Ekahau Site Survey (ESS): a design tool for map modeling and network design.
    - Ekahau Vision: RTLS web-based user application for alarms definition, tag management and other user-level functionalities (such as visualization maps for tracking tags in real-time).
- IBM WebSphere Business Events: supports correlation rule definition, execution and management. This tool possesses a complex event processing (CEP) engine that implements functionalities of the runtime correlation engine presented in Figure 16. It also offers a designer module for creating and managing rules and a database for storing rule definitions and logging events. It plays the role of the Information Correlator subsystem (Figure 11).
- PFM: Grails application providing an end-user visualization of wait times, patient states in process of care and bed states (Baffoe, 2013). It receives information from the IBM BPM and displays it. It latter stores the data in an Apache Tomcat database. This software implements the functionalities of the Process Performance Reporting subsystem presented in Figure 11.
- IBM Message Broker: acting as a broker for events between system components.
- Standalone Java App: middleware registering to ERC's specific location events and relaying those to CEP for correlation.

- HIS Mockup: HIS were being mocked up by building internal BPM events generator with IBM BPM.

5.5. Process Modeling

In order to develop a process model capturing the journey of a cardiac patient, process model information had to be gathered from Osler clinical pathway and scenario enunciation provided by an Osler domain expert.

5.5.1 Scenario

This section presents the sample scenario used to guide process modeling and understanding:

1. *Patient with cardiac symptoms arrives in ED at 08:45;*
2. *Triage Nurse check vitals;*
3. *Patient is seen in ED Assessment Room by Physician at 08:58, who then orders ECG and Blood Test.*
4. *Physician then leaves to visit another patient.*
5. *Nurse does ECG and takes blood sample at 09:10;*
6. *Nurse communicates ECG results to Physician at 09:15;*
7. *Patient waits in ED Assessment Room.*
8. *Blood test results are communicated to Physician at 09:35.*
9. *At 10:15, Physician sees Patient again; he communicates Diagnosis of NSTEMI*
10. *Physician requests Admission to CW and writes referral for angiogram and eventual PCI, at 10:25*
11. *Response to CW bed request at 10:26: No bed available for now. (The current patient will have to wait for a patient occupying a bed in the CW unit to be discharged and for that bed to be cleaned up)*
12. *Response to referral at 10:40: Patient is scheduled for angiogram next day at 11:00*
13. *Patient is given medication by Nurse while in ED Assessment Room, per ACS Clinical Pathway - Day of Admission;*
14. *Another Patient in CW bed 106 is discharged at 16:30*
15. *Housekeeping is informed at 16:35, and comes in to clean bed at 17:00*
16. *Bed/Room is cleaned up at 17:45, and Admission is informed at 17:50*
17. *CW bed 106 becomes available and assigned to Patient at 18:15*
18. *Patient is transported to CW at 18:35*
19. *Patient arrives in CW bed 106 at 18:40*
20. *Patient keeps getting care per ACS Clinical Pathway – Day of Admission*
21. *Next morning, Patient gets care per ACS Clinical Pathway – Day 1, which is also Intervention Day*

22. *Patient is called to CCL at 10:30*
23. *Patient is transported to CPU, and is prepared for procedure*
24. *Angiogram, followed by same-sitting PCI, is started at 11:40*
25. *Patient is sent back to CPU at 12:30 for post-procedure monitoring*
26. *Procedure is deemed to be successfully completed at 16:00, and patient can leave CCL*
27. *Patient is transported back to CW at 16:20*
28. *Patient keeps getting care per ACS Clinical Pathway – Intervention Day*
29. *Next morning, during 08:30 round, Physician sees that Patient is recovering well and decides to discharge him*
30. *Discharge Nurse schedules Patient for 11:00 and goes through the checklist per ACS Clinical Pathway – Discharge Day*
31. *Patient leaves his bed in CW at 12:40, and is picked up by his family*
32. *Housekeeping is informed at 12:45, and comes in to clean bed at 1:30*
33. *Bed/Room is cleaned up at 2:00, and Admission is informed at 2:10*
34. *Bed is assigned to a new patient at 2:50"*

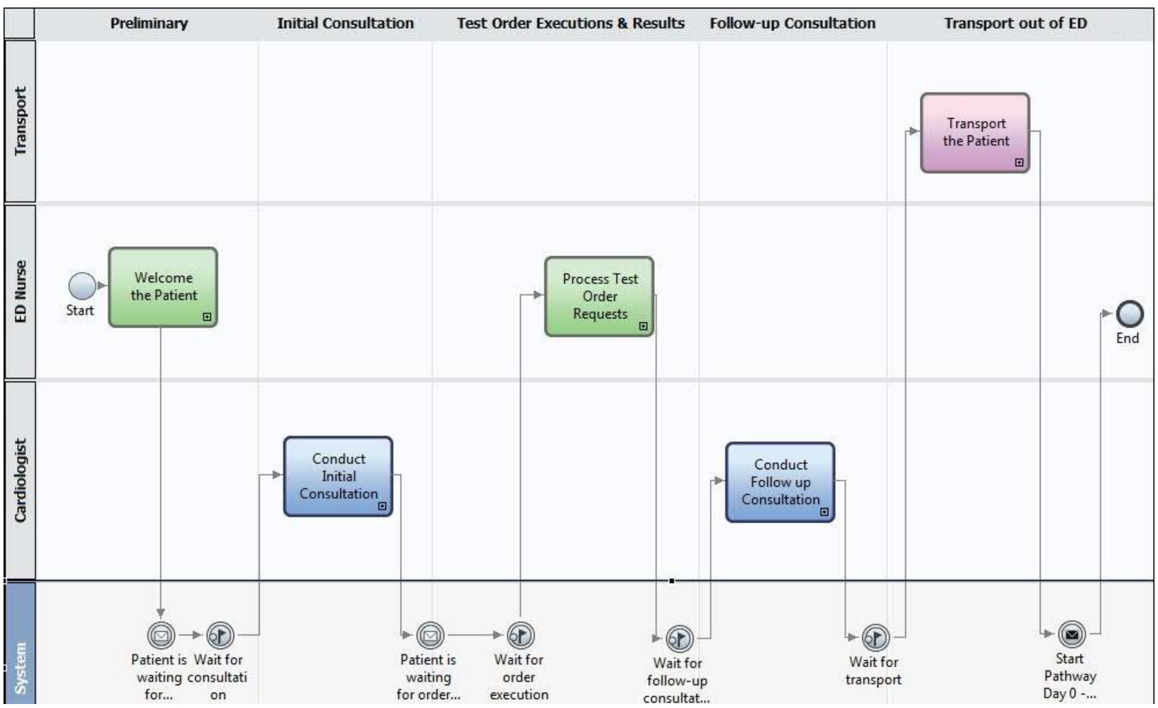


Figure 22 Overview of ED Process of Cardiac Care

Based on this information and following workflow modeling practices, process models were developed and validated with the hospital staff. The initial modeling was actually done on a simple online BPMN tool (IBM Blueworks Live[10]) and then imported in IBM BPM for completion to make it executable and for including suitable user interfaces. Fig-

[10] https://www.blueworkslive.com/

ure 22 shows an extract of the process model developed in compliance with Osler's ACS clinical pathway. In this figure, sub-processes are used to separate concerns from the global "episodes" involved in ACS care process in ED:

- Welcome: all preparation of the patient for consultation. This usually involves patient identification or registration (if this is a new patient), checking of vitals, request for a physician, etc.
- Initial consultation: encapsulates the series of activities involved in the initial consultation (physical examination, test orders, etc.)
- Test order execution and results: covers the execution of orders by the assigned nurse (e.g., ECG and blood samples) and the lab evaluation.
- Follow-up consultation: the steps where the physician comes back for diagnosis and decides whether the patient will be admitted or not for treatment[11] (if so, the request for bed will occur).
- Transport out of ED: this sub-process is composed of the steps involved in transporting the patient out of ED to either discharge or to the relevant unit of admission.

The IBM BPM environment allows for reusability of activity modules. This powerful approach gives much flexibility in designing or modifying process models.

Figure 23 shows the steps involved in the cardiac care process for a patient after he/she is admitted to CW. The patient is given an appointment for the treatment procedures required by the ED physician, namely angiogram and Percutaneous Coronary Intervention (PCI), while receiving orders as prescribed by the physician according to the ACS clinical pathway for each day. On the day of his/her procedures, the patient is transported to the CCL room for procedures and brought back to his bed when done. A post-procedure consultation is later conducted to assess the condition of the patient during recovery. A discharge is then requested for the next day if patient condition is approved by the cardiologist.

[11] Note that in the case of an inconclusive diagnosis, the physician might require additional tests to be performed on the patient. This case, although not modeled in our actual implementation, can easily be supported by adding a link from the "Conduct Follow-up Consultation" sub-process back to the "Process Test Order Requests" sub-process. This shows the power of adequate separation of concerns and reusability of activity modules.

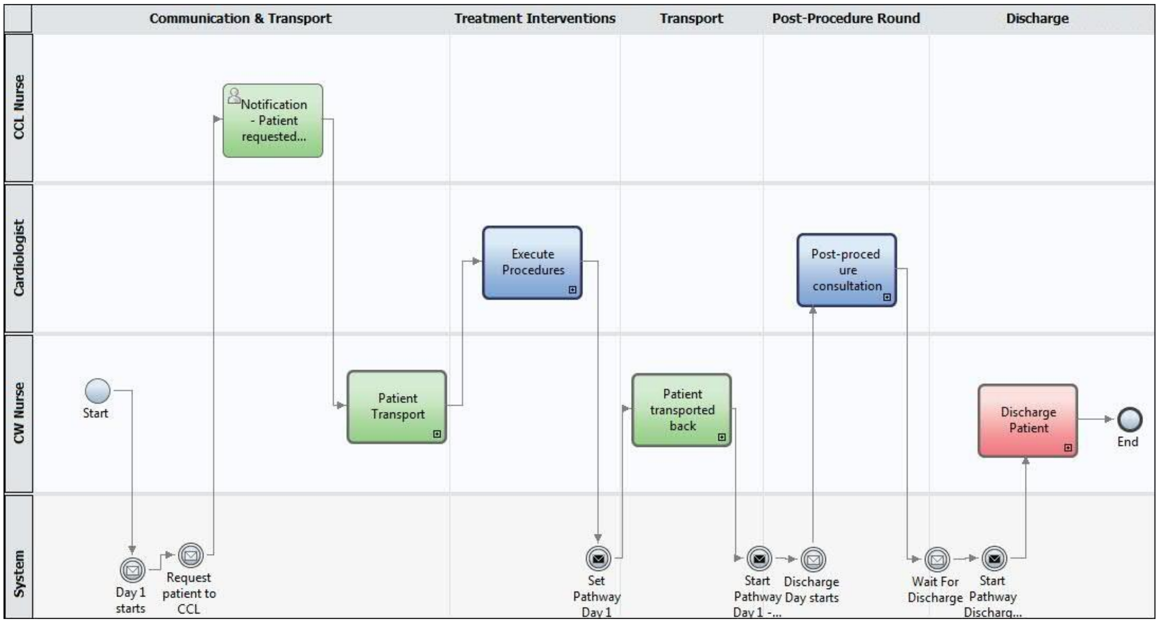


Figure 23 Overview of CW Process of Cardiac Care

At various points of the model where we needed to track time, time trackers ("flag" nodes in Figure 22) were implemented. Those elements basically generate timestamps whenever process execution reaches the specified point in the model. Timestamps are automatically stored in the process database.

5.5.2 Workflow Model Development

As mention earlier in this thesis, a complete workflow model needs to be built in order to have an executable process model. This includes implementation of process activities, definition and implementation of web services, process data modeling, development of coaches (user interfaces), interoperability, etc. The IBM BPM authoring environment (featured in Figure 24) was used to this end.

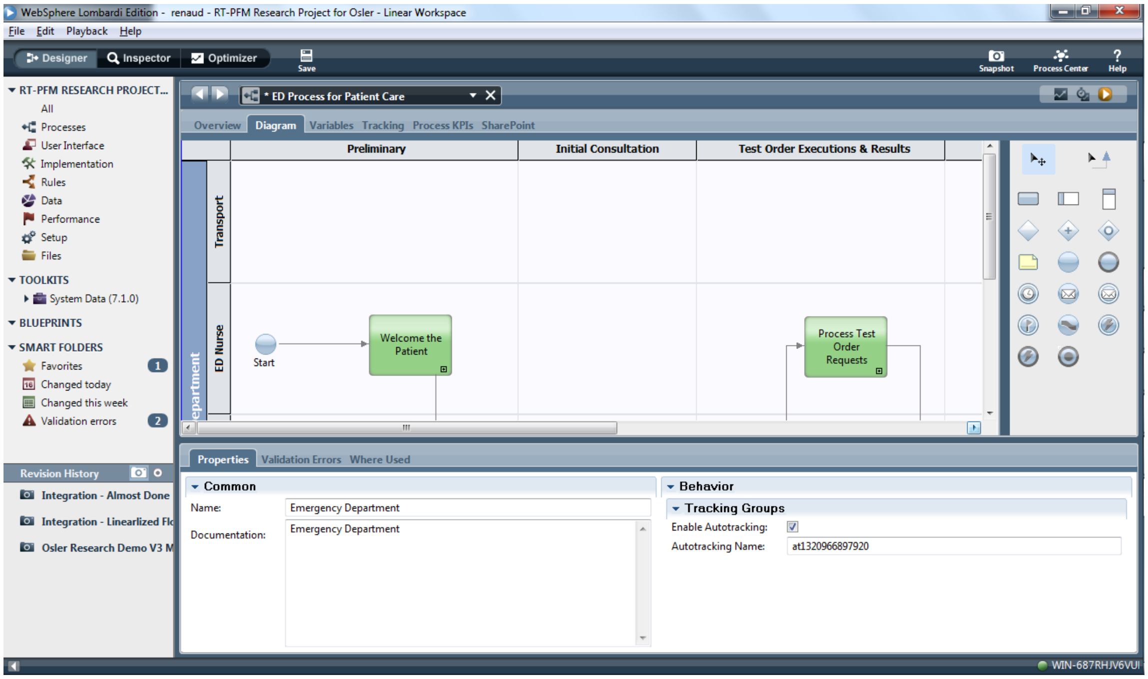


Figure 24 BPM Authoring Environment

5.5.3 Implementation: Coaches and Services

As explained before, workflow definitions require participating activities to be concretely implemented in order to be programmatically executed. For a human-to-system task, a user interface needs to be defined. For a system-to-system task, an implementation service needs to be defined. This way, the process orchestrator engine can follow the control flow definition and trigger the right task at the right time by instantiating the corresponding implementation through internal web service calls.

Figure 25 shows an example of human-to-system task's implementation for conducting "Physical Examination" on the patient (done by the physician). The user interface (called *coach* in the IBM BPM terminology) features various output fields for viewing patient demographics and triage information and an input field for entering physical examination report.

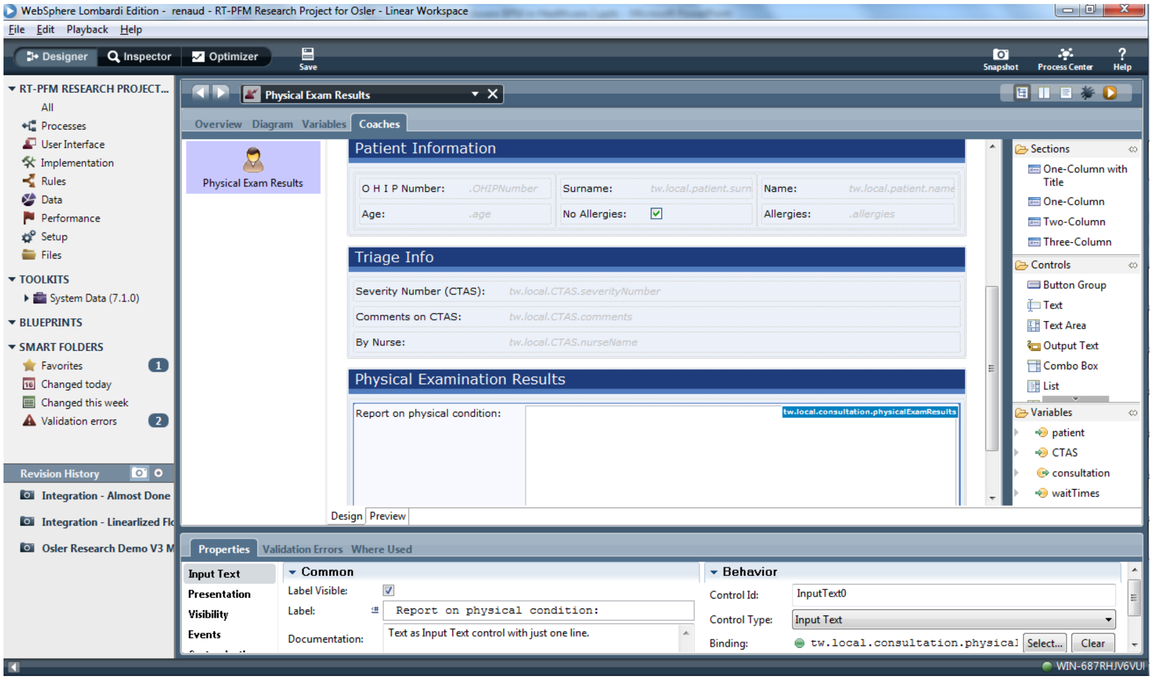


Figure 25 Coach Implementation

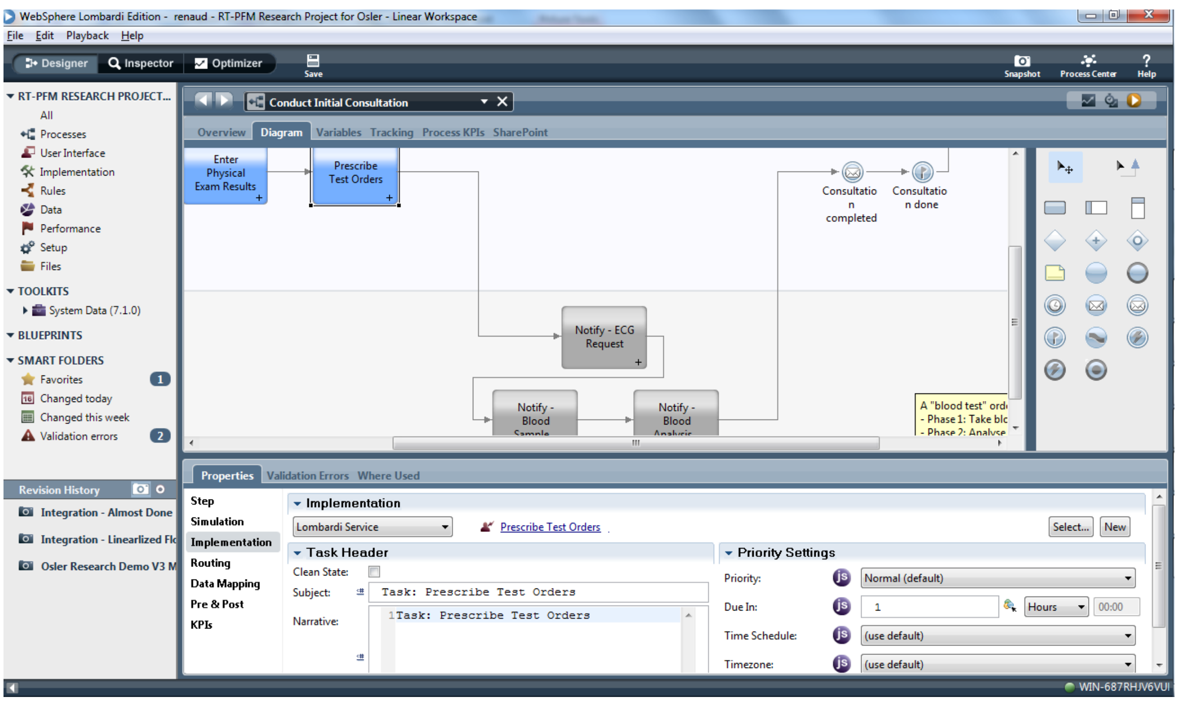


Figure 26 Service Implementation

Figure 26 shows an example of system-to-system task implementation. In IBM BPM, multiple types of system-to-system task implementations called *services* are possible such as a Java integration service (for embedding Java calls), web service integration (for web service calls) or a general system service (for writing BPM internal scripts). All those services are flow-based, meaning that they can implement an orchestration of multiple atomic operations/calls.

5.5.4 User Experience for Nurses and Physicians

This section highlights the user experience when completing tasks. The first two figures (Figure 27 and Figure 28) show snapshots of the iPad user interface for user task management. The iPad client application, included in IBM BPM, connects to the server application interface to retrieve/persist process information.

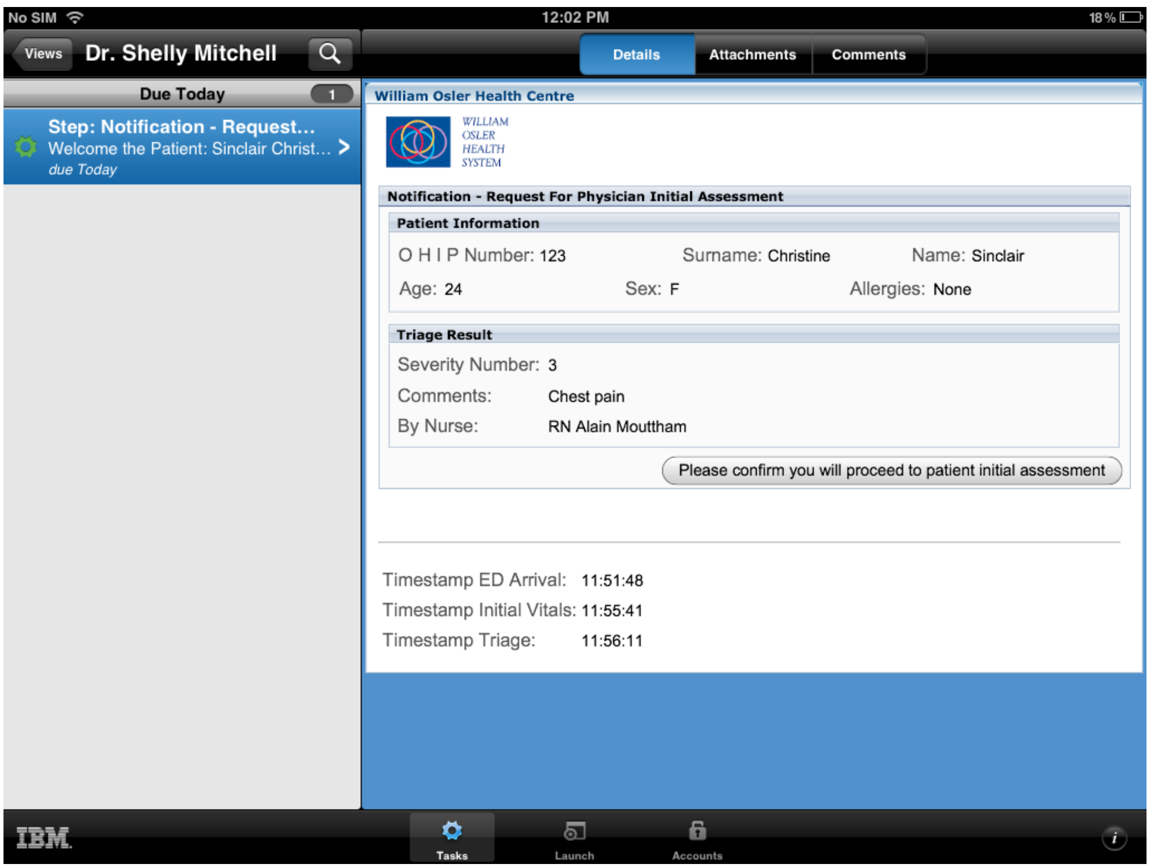


Figure 27 iPad Care Process Activity Example

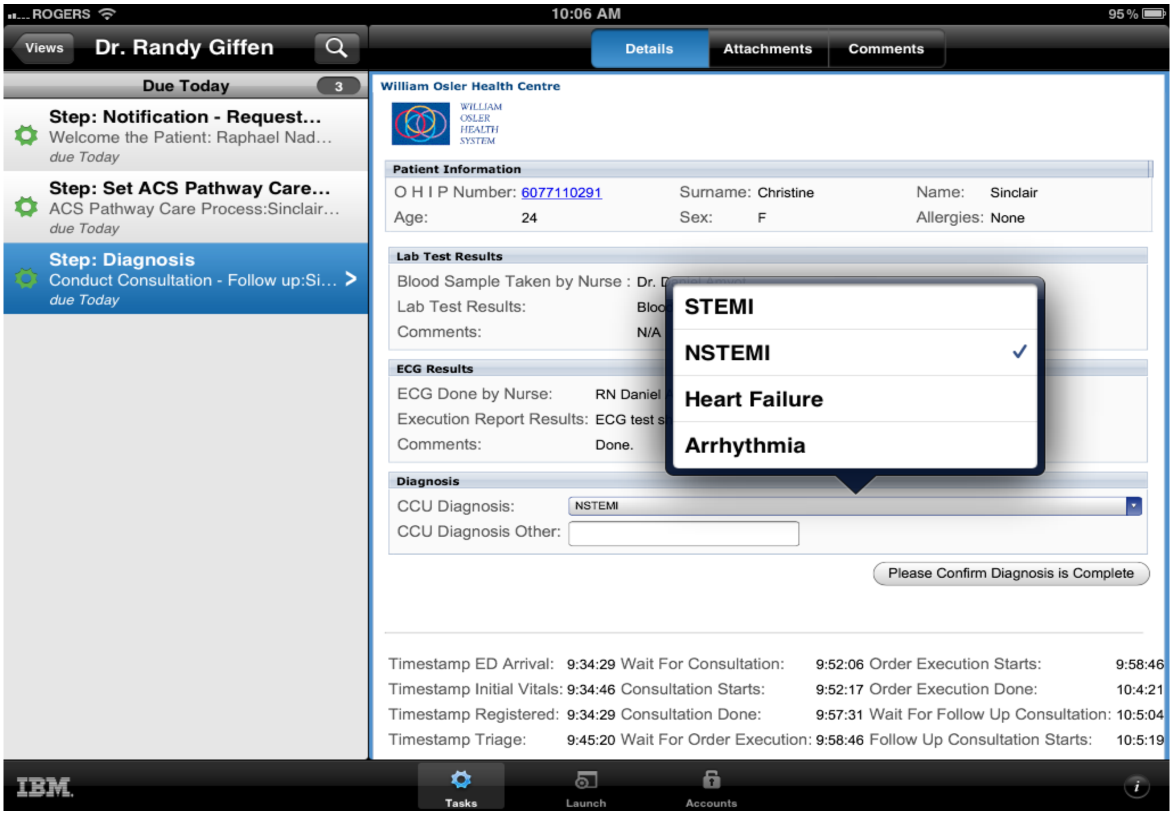


Figure 28 iPad Care Process Activity Example 2

Figure 29 shows task management functionalities on the laptop version. The user receives automatically newly assigned tasks, and can view old tasks as well as personal performance and team performance related to task completion The user can also collaborate with other users as part of the process.

One important point here is that the coaches where implemented in such as was as too:

- Only provide the information relevant to the task to be completed, in order to avoid information overflow (while giving the possibility to obtain more information if really needed)
- Minimize required interactions and intrusiveness

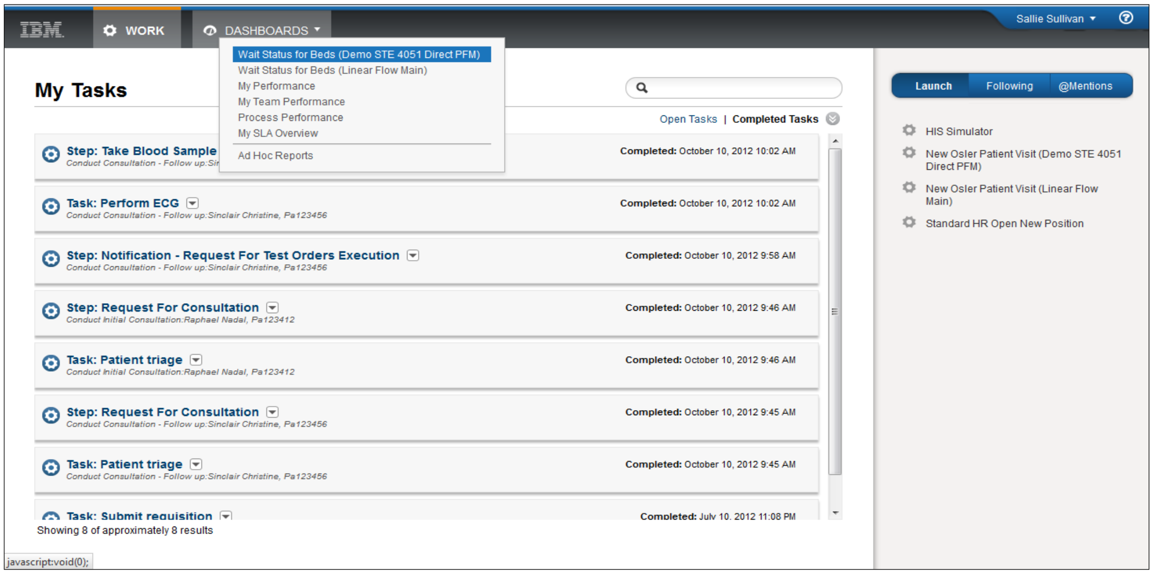


Figure 29 User Workspace showing Captured Process Task History

5.6. RTLS Model and Zone Definition

To enable location tracking based on an existing wireless network, we need to define a map reflecting the physical environment. For map design, the *Ekahau ESS* was used in this case study. The map is a simple image representation (e.g., JPEG bitmap) tailored to respect the relative dimensions/ratios of the map elements. This map is then loaded in the ESS software with an indication of the unit scale (specifying to what length on the image map a unit of 1 meter corresponds to). The scale is used to later show on the map real-time positions of the tags being tracked. Figure 30 shows the image map of the laboratory at the University of Ottawa used to conduct local development and testing. The map depicts the facilities of a real hospital simulated into a single room, and capturing the units involved in the cardiac emergency patient flow from admission in ED to discharge out of the hospital.

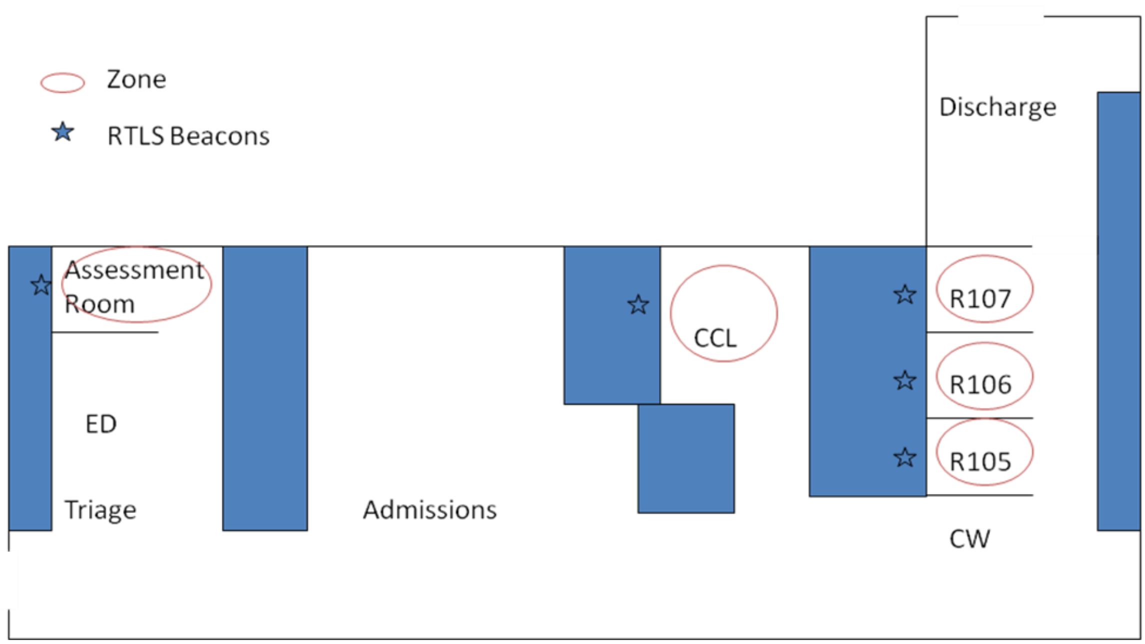


Figure 30 Floor Map with Different Units (University Lab)

In this implementation, the choke point technique (Kamel et al., 2012) was used since we were interested in tracking entrances and exits out of the following zones (circled in Figure 30) in the hospital: Assessment Room, CW_R105, CW_R106, CW_R107, CCL and Discharge. To achieve this, infra-red beacons (represented by stars on the map) were placed in each of the zones and their identifiers were reported to the RTLS controller by updating the map information. Because Ekahau's RTLS solution uses a fingerprinting indoor localization algorithm (Yang et al., 2012), a site survey needed to be conducted around the simulated "hospital" for an adequate network model to be created that can enable accurate (2m to 3m) wireless network triangulation.

With this configuration, tags were being tracked around the hospital using the wireless network, and whenever they were entering or exiting a room, they were detected by the infrared beacons. The following are two examples of XML objects sent from the RTLS. The first one indicates that patient Pa123456 has entered the emergency department (ED), while the second one indicates that the physician (Phy777777) has exited room 107 of the cardiology ward (CW):

- **<ZONEENTERED>**

```
  <ruleid>551</ruleid>
  <eventtime>1341327680953</eventtime>
  <tagid>105465783089</tagid>
```

<mac>00:18:8E:40:23:31</mac>
<name>Pa123456</name>
<type>t301w</type>
<posx>102</posx>
<posy>138</posy>
<poszoneid>3</poszoneid>
<poszonename>ED</poszonename>
<posmapid>0</posmapid>
<posmapname>Floor Map SITE4051</posmapname>
<posmodelid>18</posmodelid>
<postime>1341327680805</postime>
<postimestamp>2012-07-03 11:01:20-0400</postimestamp>
<eventquality>96</eventquality>
</ZONEENTERED>

- **<ZONEEXITED>**
<ruleid>545</ruleid>
<eventtime>1341327790518</eventtime>
<tagid>105465783085</tagid>
<mac>00:18:8E:40:23:2D</mac>
<name>Phy777777</name>
<type>t301w</type>
<posx>773</posx>
<posy>107</posy>
<posmapid>0</posmapid>
<posmapname>Floor Map SITE4051</posmapname>
<posmodelid>18</posmodelid>
<postime>1341327783800</postime>
<postimestamp>2012-07-03 11:03:03-0400</postimestamp>
<eventquality>100</eventquality>
<oldzoneid>5</oldzoneid>
<oldzonename>CW_R107</oldzonename>
<oldmapid>0</oldmapid>
<oldmapname>Floor Map SITE4051</oldmapname>
</ZONEEXITED>

These low-level events are processed and filtered by the Ekahau system into high-level events. Only application-relevant information are considered (attributes in bold). The resulting events are: *PatientInED, PatientOutED, PhysicianInED, PhysicianOutED, PatientInCW, PatientOutCW, PatientInCCL, PatientOutCCL, HouseKeepingInCW, HouseKeepingOutCW*. Each event had the following attributes: the *tagname*, the high-level location identifier (e.g., *zone name*) and a *timestamp*. The following is an example high-level event, ready to be processed by the information correlator (IBM Websphere Business Event):

```
<event name="PhysicianOutCW" sourceSuffix="RTLS">
    <Patient_ID>Phy777777</Patient_ID>
    <Location_ID>Bed107</Location_ID>
    <timestamp>2012-07-03 11:03:03</timestamp>
  </event>
```

5.7. Definition of Correlation Rules

Several inference rules were developed by Baraah (2013). Those rules define the intelligence of the system to infer wait states. The rules that were covered are for the specific following wait times:

- Patient wait time for Physician Initial Assessment (PIA).
- **Patient wait time for order tests to be done.**
- Patient wait time for lab results.
- Patient wait time for Physician Re-Assessment.
- Patient wait time for bed cleanup by housekeeping.
- **Patient wait time for bed to be assign (wait for bed).**
- Patient wait time for transport to come.
- Patient wait time for CCL procedures (angiogram, PCI).
- Patient wait time for discharge.

The following are two sample correlation rules (in bold in the above list) from Baraah's work, implemented for the IBM WebSphere Business Event environment:

- ```
  Interaction Set: Patient Wait For Order Execution Related
  by Patient.Patient_ID

  In response to: ConsultationCompleted1Syn From CEP
       Immediately
             This Event Follows ConsultationStarted1Syn
             AND
             All Occurrences of OrderRequestTypeECGBloodSyn
             Then: Immediately WaitForOrderExecution (BPM)
                 : Immediately WaitForOrderExecutionSyn (CEP)
  ```

This first inference rule example expresses how the CEP detects a wait for order execution. The reception of a "ConsultationCompleted1Syn" event (indicating the first consultation is completed) triggers the evaluation of the rule in the following way: if the start of
  ```

that consultation ("ConsultationStarted1Syn") and any occurrences of an order request for ECG or Blood test have been previously recorded, then infer that the patient will now start waiting for order execution. As a result, immediately send a "WaitForOrderExecution" to the BPM subsystem and send (to itself) a "WaitForOrderExecutionSyn" for record purpose.

- Interaction Set: **Patient Wait For Bed**

```
Interaction Set: Patient Wait For Bed
                  Related by Patient.Patient_ID

In response to: PatientAdmittedWithNoBed From BPM
     Immediately
          All Occurrences of BedRequest Is 1
          AND
          If All Occurrences of ConsultationCompleted2Syn
                                                    Is 1

          Then: Immediately WaitForBed (BPM)
              : Immediately WaitForBedSyn (CEP)
```

Similarly, the evaluation of the patient wait for bed event is triggered upon reception of the event indicating that the patient has been admitted, but no bed has been assigned yet ("PatientAdmittedWithNoBed" event). Then, if a bed request was once made (received from BPM and recorded by CEP) and one second physician consultation has been completed in the past, then a "WaitForBed" event will be generated and sent back to the BPM engine while being logged locally by the CEP (stored in its database).

Those two examples show how the powerful feedback communication loop between CEP and BPM allows intelligent tracking of patient wait states.

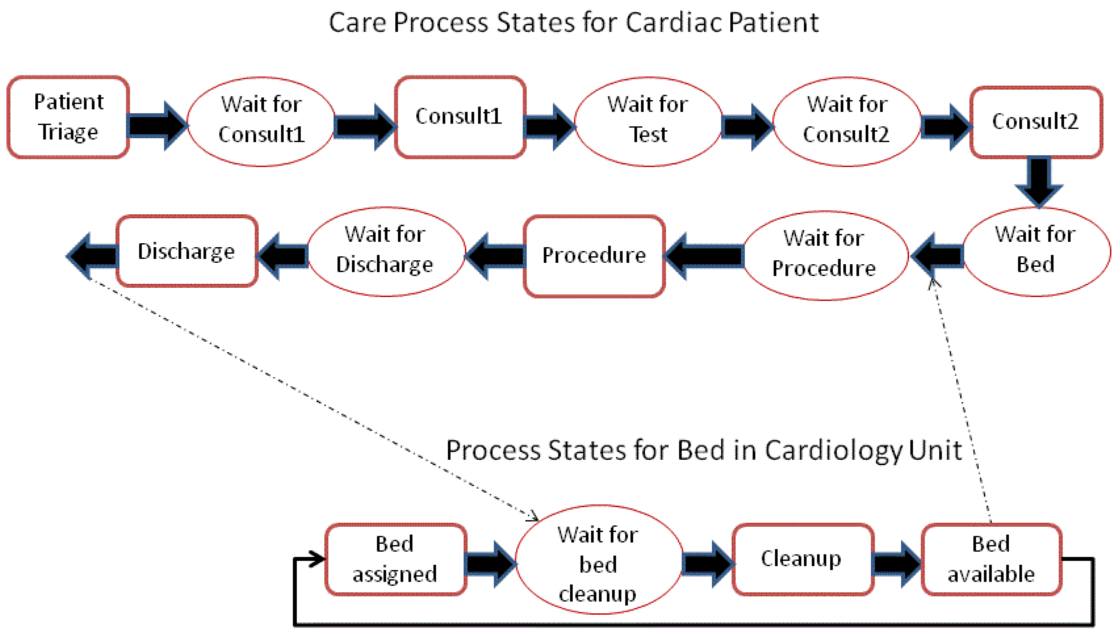
5.8. Process Performance Reporting

The work here was mainly conducted by Baffoe (2013). This consisted in the implementation of the reporting component displaying real-time metrics for process activities, patient states (including wait states) and bed states in real time. The following group of states for patient and bed were identified and implemented:

- **Patient states:** patient triaged, waiting for consultation, in consultation, waiting for test, tests completed, waiting for bed, wait for transport, in transport, waiting for procedure, procedure started, procedure executed, wait for discharge, discharge.

- **Bed states:** bed assigned, wait for bed, cleanup bed, bed available.

Figure 31 shows a diagram providing relationships between the patient states and bed



states in the context of the process model (Baffoe 2013):

Figure 31 Patient and Bed States

PFM directly receives events from BPM (time tracker and process data information) and CEP (process context events) to display wait times in real time. PFM's dashboard is accessible to access and flow managers.

A more recent version of PFM is not being prototyped with IBM Cognos BI[12] for near real-time reporting of performance by accessing BPM's performance database (Baffoe, 2013).

Wait time timestamps were also recorded in the process performance database using time trackers earlier defined in the BPM model.

[12] http://www-01.ibm.com/software/analytics/cognos/

5.9. Application Testing

5.9.1 System Testing and Pre-Runs

The system components were first separately tested by each lead developer using mock-ups modules built using the SoapUI[13] messaging tool in order to simulate events exchange with other components. Then, integration testing was done. This process took place along several phases of development in an incremental way at the University of Ottawa.

5.9.2 Environment Setup and Logistics

In addition to the lab setting at the University of Ottawa, a simulation room was put at our disposal by Osler for a few weeks. A specific RTLS configuration (map and zone definition for the room, together with tag configuration) was set up. A dedicated wireless network was also installed and used during the simulation. The room contained patient mannequins and medical equipment, making it suitable for the simulation. We were simulating the entire process of cardiac care in a small room. This provided various technical challenges and represented a serious stress test for our prototype, especially in terms of making the environment self-contained and movable.

5.10. Evaluation and Results

5.10.1 Conducting the Simulation

The simulation involved walking the audience (a selected group of people) through the journey of a cardiac patient from ED to CW, CCL and Discharge. The simulation scenario was composed of three patients, one doctor, one nurse, one housekeeping employee and one transporter. They were all wearing RFID tags, allowing their position to be tracked in real time on the Osler map accurately depicting the simulation room (shown in Figure 32). The scenario was going through the journey of a newly arrived cardiac patient (one of the three patients) exhibiting chest pain symptoms, and highlighted in Section 5.5.1.

---

[13] http://www.soapui.org/

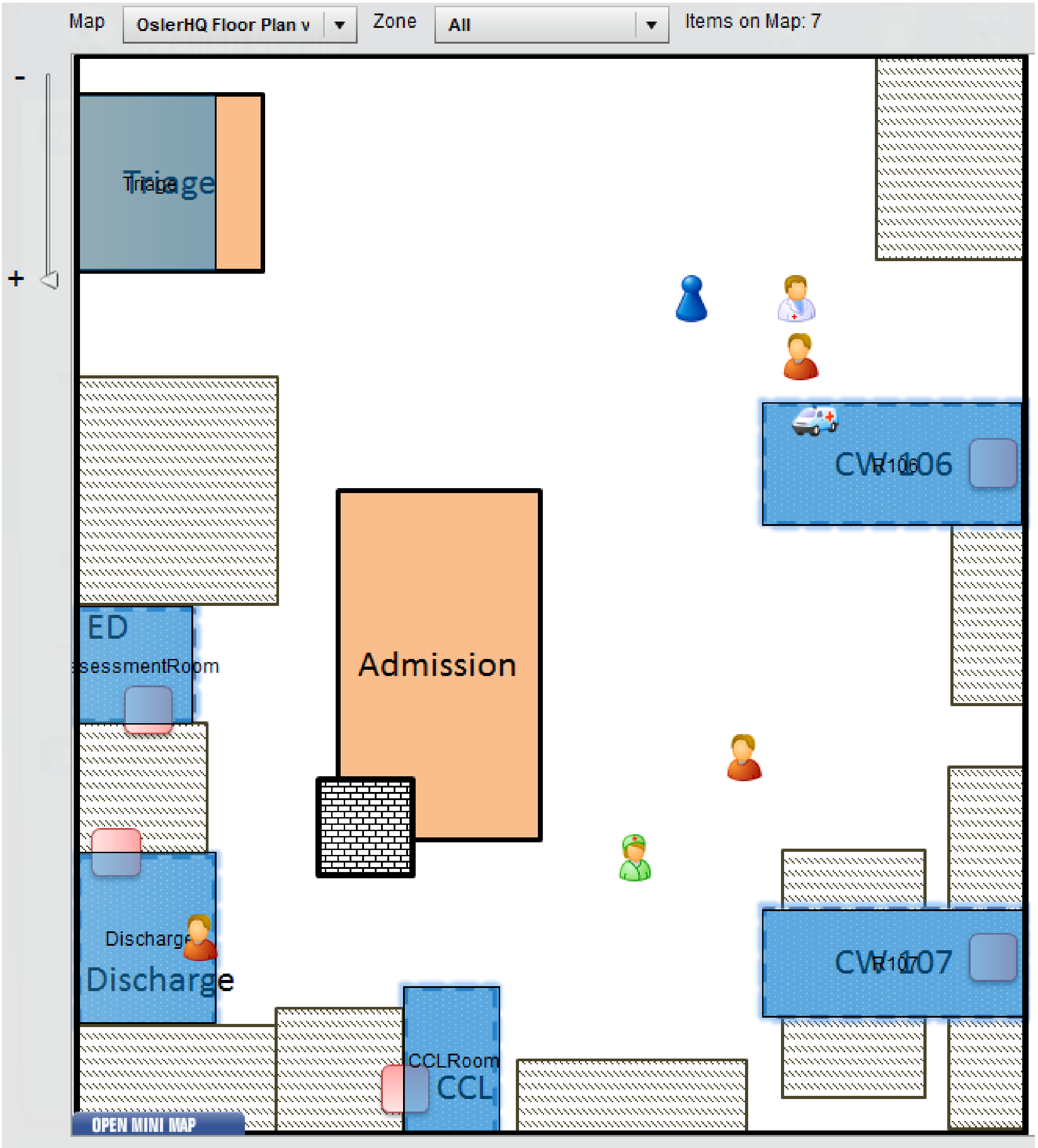
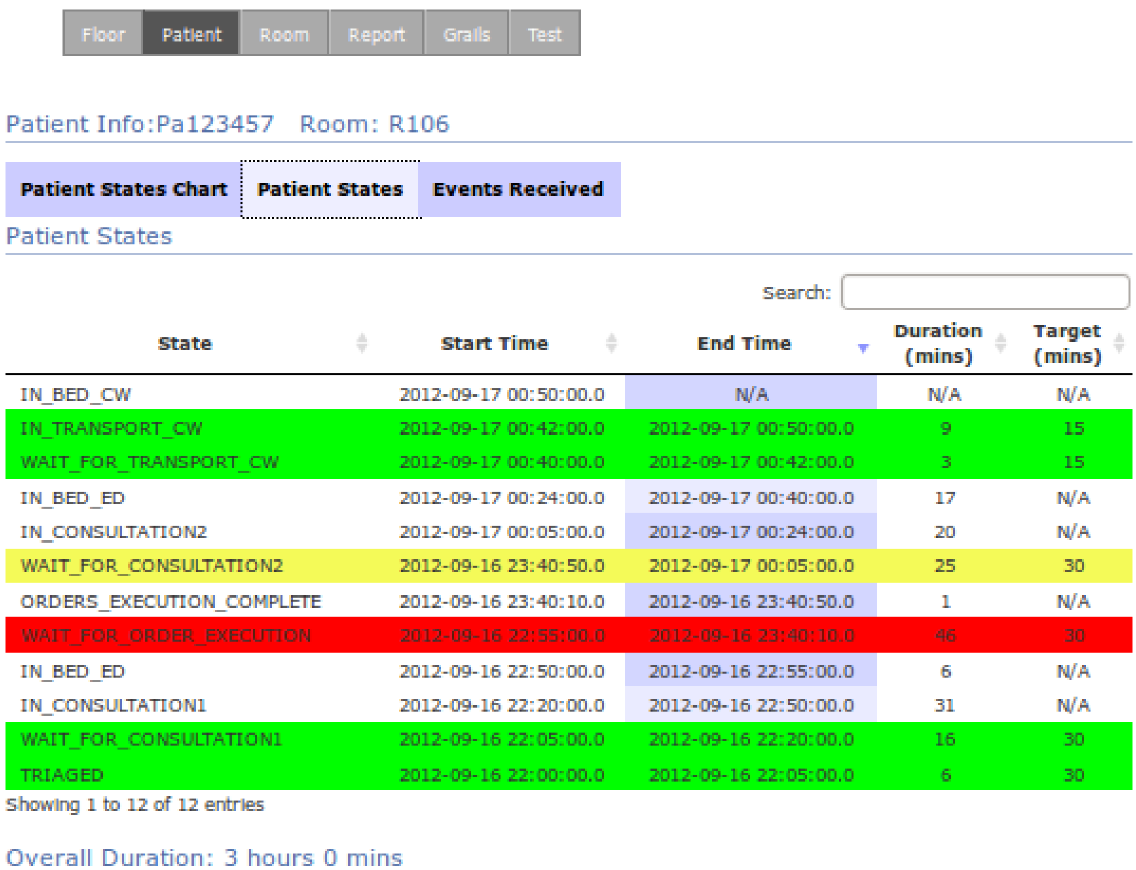


Figure 32 Ekahau Vision's Real-Time Tracking Map of Osler's Simulation Room

The capabilities of the tool were demonstrated by the abilities for the doctor and the nurse to receive notifications of new tasks and complete the next required task (using task management) according to the clinical pathway, with all task completion information being recorded and saved into the process performance database. This information was successfully communicated to the CEP by the BPM application, and the CEP, also receiving constant updates about the location of the actors, was able to adequately infer patient wait states accurately and forward them to the reporting component. The whole monitoring process only takes a few seconds, and hence represent a significant improvement over current manual practices.

During the simulation, the PFM dashboard was being updated as we were check marking the various points of interest specified in the process model. Figure 33 shows a screenshot of the dashboard monitoring the process states for the patient with identifier “Pa123457” and currently assigned to Room “R106”. For each new state in the care process, the start time and the end time are recorded and the *duration* is calculated. For example, the duration for the “IN_CONSULTATION1” process state is 31 minutes (corresponding to the service time for that activity).



State	Start Time	End Time	Duration (mins)	Target (mins)
IN_BED_CW	2012-09-17 00:50:00.0	N/A	N/A	N/A
IN_TRANSPORT_CW	2012-09-17 00:42:00.0	2012-09-17 00:50:00.0	9	15
WAIT_FOR_TRANSPORT_CW	2012-09-17 00:40:00.0	2012-09-17 00:42:00.0	3	15
IN_BED_ED	2012-09-17 00:24:00.0	2012-09-17 00:40:00.0	17	N/A
IN_CONSULTATION2	2012-09-17 00:05:00.0	2012-09-17 00:24:00.0	20	N/A
WAIT_FOR_CONSULTATION2	2012-09-16 23:40:50.0	2012-09-17 00:05:00.0	25	30
ORDERS_EXECUTION_COMPLETE	2012-09-16 23:40:10.0	2012-09-16 23:40:50.0	1	N/A
WAIT_FOR_ORDER_EXECUTION	2012-09-16 22:55:00.0	2012-09-16 23:40:10.0	46	30
IN_BED_ED	2012-09-16 22:50:00.0	2012-09-16 22:55:00.0	6	N/A
IN_CONSULTATION1	2012-09-16 22:20:00.0	2012-09-16 22:50:00.0	31	N/A
WAIT_FOR_CONSULTATION1	2012-09-16 22:05:00.0	2012-09-16 22:20:00.0	16	30
TRIAGED	2012-09-16 22:00:00.0	2012-09-16 22:05:00.0	6	30

Figure 33 PFM Real-Time Patient State Monitoring

In addition, the duration of each process state is compared against a service *target* time (e.g., service-level agreements). This allows the system to flag important information about the state of the care process in real time using color codes. More specifically, a state is flagged in:

- Green, if the duration is well below the target time (informing that the activity was performed on time).
- Yellow, whenever the duration is close to the target time (informing that the activity might need to be watched closely).
- Red, whenever the duration is equal or exceeds the target time (indicating that immediate action should be taken). Notifications or alerts can also be fired when a duration exceeds its target.

Note that no color code is used for process states that do not have any set target time.

With all available historical data, it is possible to build valuable data reports (as is traditionally done) and leverage visualization features to better grasp the behaviour of the processes being observed. Figure 34 shows a state chart report example displaying the average time spent in each of the recorded steps of the cardiac care process.

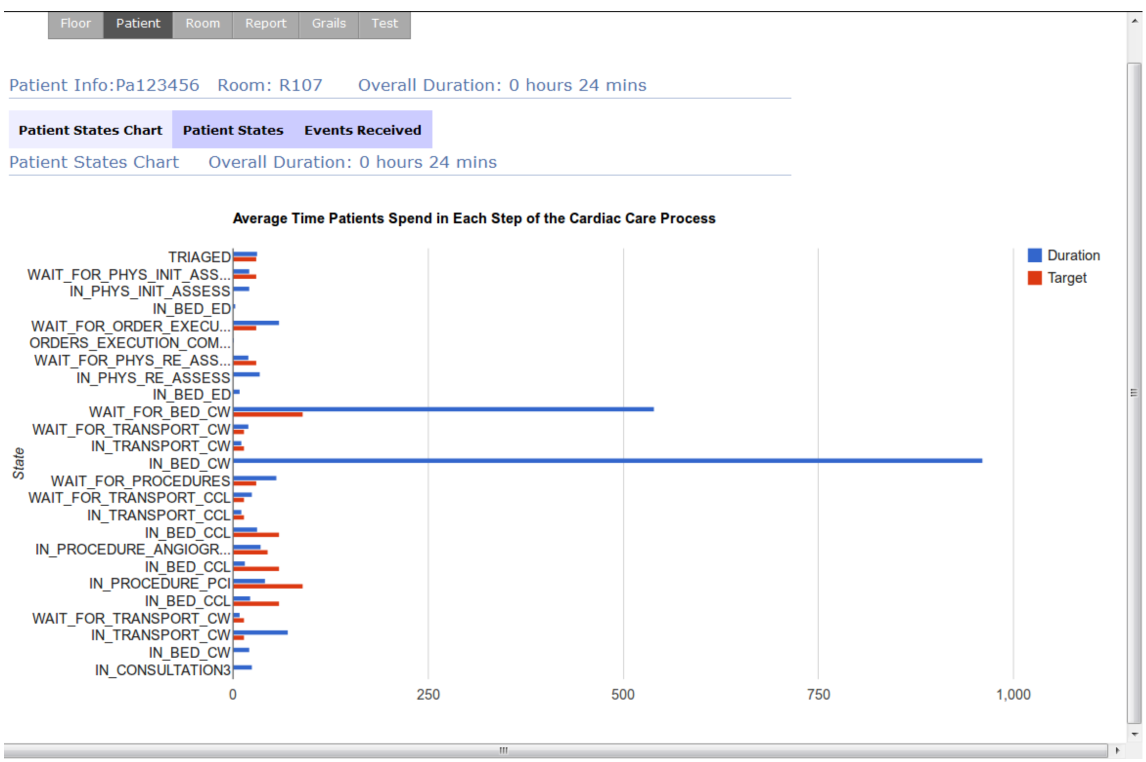


Figure 34 PFM State Chart

5.10.2 Simulation Results

Twelve simulations were conducted over six days at William Osler Health Centre. In total, they involved 95 participants from different business groups: the senior leadership team, the access and flow managers team, the IT team, the nurse team, the BI team, the project management team, a group of transporters, an IBM group and the general hospital public.

Participants were able to understand the problem statement and follow the patients through the care steps. Events generated during the journey of the cardiac patient going through the process (following the Osler ACS clinical pathway) were visible in real time on the PFM dashboard. Participants were also able to make sense of the data captured and exposed on the dashboard, understanding the state of the patient in the cardiac care process, but also the contextual situation around the process when it is in a wait state (e.g., the physician is in another department, treating another patient).

Nine other simulations were also conducted at the University of Ottawa[14] throughout the course of the last year (following the gradual evolution of the prototype) and involving various groups of participants:

- 2 sessions with University of Ottawa professors and researchers,
- 2 sessions with Queensway-Carleton Hospital representatives,
- 2 sessions with representatives from The Ottawa Hospital,
- 1 session with a small group from William Osler Health Center,
- 1 session with IBM representatives and doctors/researchers from Thailand, and
- 1 session with University of Ottawa alumni.

The feedback received allowed us to better understand the general opinion, the technical challenges and the overall potential of the platform. In a survey led by Mouttham on behalf of the research team, 80 participants from Osler provided answers that indicated two interesting results: 82% considered this technology would improve patient flows and would be better than what exists today, and more than 50% believed that the technology

[14] Figure 30 shows the RTLS map of the university laboratory used for the simulation.

should ideally be deployed within the next 2 years. A multi-month pilot study is now being organized at Osler.

Overall, the simulations validated the feasibility of a system capable of 1) process automation of real clinical flows, 2) location tracking of patients and staff, 3) detection and tracking of patient wait times, and 4) reporting patient and bed states to access and flow managers.

5.11. Chapter Summary

This chapter has presented a concrete implementation prototype of our solution architecture. The prototype has been designed, implemented and evaluated both in a laboratory at the University of Ottawa and on-site at William Osler Health Center. Technological infrastructure as well as implementation details for the logical artifacts of the system have been specified and presented. The specifics behind conducting the simulation where presented along with general simulation results. This evaluation took place in the form of multiple demonstrations with positive feedback gathered from tangible and qualified stakeholders.

The next chapter discusses the extent to which the presented work is valid, along with the challenges we have faced during this research.

Chapter 6. Discussion

This chapter discusses several challenges faced during this project, together with a comparison with closely-related work, and threats to the validity of this work.

6.1. Challenges

Multiple challenges have been faced while developing this system. The major ones are presented in this section.

As we have witnessed, the problem of wait time measurement is not trivial. While this is a concept relatively natural to understand, identifying the underlying factors that influence wait times and defining how to apply methodologies to control them is not obvious. Moreover, because these factors are usually of different natures, the main difficulty is to specify, for a given hospital in a given environment, where visibility is scattered and what the specific factors are. This has constituted a serious challenge throughout this work. This is why an approach that first promotes visibility and real-time reactive control over process execution has been proposed.

The challenges involved in developing a working solution for our research hypothesis (specified in Section 1.4) are:

- **Design of clinical workflow models:** The design of clinical workflow models was probably one of the biggest challenges. Understanding how to model the complexity inherent to clinical workflows was not obvious. Luckily, this work benefited from the help of Osler nurses and of modeling experts from IBM.
- **Testing of workflow models:** Workflow testing was done manually as we struggled to find automated workflow testing tools. Testing is time-consuming and the manual approach begs more rigour. Papers like the one from Robić (2010) helped us substantially. Robić presents an approach to automated workflow testing featuring his own testing tool built to test workflows running on the IBM FileNet Process Engine. Still,

we were able to automate the testing of messaging interfaces using SoapUI testing modules.

- **Context-sensitivity of healthcare:** The context-sensitive nature of the healthcare environment made access to information very delicate. Also, in term of access to resources, it made it difficult to access a real environment for conducting formal experiments. We benefited from the availability of documentation of a real healthcare (cardiology) process, with access to a hospital room for demonstrations.
- **Location Technologies:** The use of location tracking devices was also an important challenge. We found that different RTLS solutions have particular behaviours and strengths depending on their vendors. It is important to understand the specific characteristics of a given RTLS solution in order to find out the best configuration that will maximize location tracking performance in a given. Comparisons are however difficult on paper because each vendor will usually have their own proprietary algorithm for location detection.

6.2. Comparison

In this section, we compare the work presented in the thesis against existing related approaches (overviewed in Chapter 3). The approaches used for comparison were chosen based on their close similarity with respect to system objectives, the application domain and technologies.

The first approach that deserves our attention is the one from van Hee et al. (2008). The authors present a framework that uses adaptive workflows as a replacement for existing data-centered HIS approach. Its objectives are to improve the information flow between healthcare providers (for better visibility and communication between actors in care processes) and to allow patients to have visibility over the evolution of their care process through information updates. Three central *concepts* are introduced in the framework to achieve an adaptive workflow design: adaptivity, adaptability and separation of concerns. *Adaptivity* refers to the ability of the workflow to modify itself (adapt) “on-the-fly” whereas *adaptability* refers to the ability of the workflow to request a provider to make certain manual decision about the course of the treatment process during process execution. *Separation of concerns* corresponds to the vision that a process actor should be the only one responsible of his own process activities. To achieve this, the paper suggests the development of separate process models for each actor and the use of communication mechanisms for those processes to work together towards patient care delivery. The authors compare the application of this process-oriented approach to care delivery against existing data-centered HIS and demonstrate a significant improvement in information flow. If a good work is done on facilitating information flow and visibility, for example, by offering tracking and labeling (using explanative notes) of patient information changes performed by providers, we argue that the implementation proposed for the framework itself can be improved. The author uses the YAWL language to model care processes, which we believe does not render well the concepts. A more "business-oriented" language such as BPMN would have been more adequate to implement the concepts of adaptability and separation of concerns (using lanes and pools). Also, adaptivity is argued in the paper to be about "on-the-fly" adaptation of the workflow. However, the proposed example of a "special" process being started in parallel as a result of the patient Rh(D) blood test being negative can sufficiently be modeled using the technique of flexibility by design presented in Section 4.8.3, e.g., through the use of a BPMN decision gateway for which the negative evaluation of the condition triggers a *start message* to start the special process. Moreover, no aspect of performance is studied and no concern about scalability of their framework and integration of workflows is addressed.

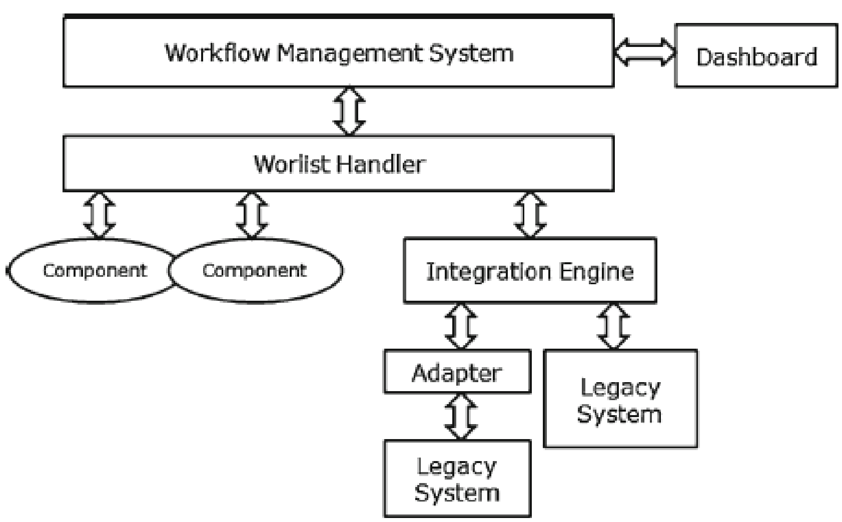


Figure 35 System Architecture Workflow-based RIS (Zhang et al., 2009)

The second related approach is the one from Zhang et al. (2009). Zhang et al. present a workflow-based Radiology Information System (RIS) for monitoring radiology processes in real time. Figure 35 presents their solution architecture.

This architecture revolves around the use of a YAWL-based workflow management system enacting a workflow model of a radiology process. The major strength of this work compared to ours is that integration with the existing heterogeneous HIS ("Legacy System" on the Figure 35) is implemented in a prototype that was deployed in a Chinese Hospital. The experiment reports the ability to measure wait times and service times of activities and presents them in an integrated view. Classic workflow systems are able to measure wait times by simply measuring the time between each activity occurring in the flow. This type of approaches might be fine for radiology processes but it is not sufficient for processes where actions need to be taken promptly in real-time settings for early mitigation of occurring wait times (e.g., ED processes). This is because no insight is provided about the measured wait times as waiting actually occurs. No information about what is happening between activities A and B can possibly be automatically obtained. That is one important advantage of our work, where granularity is offered on the information captured by bringing context information such as location to a process management engine through inference of the "hidden" process wait states. The direct benefit is an increase of visibility over the specific steps of wait times and the care processes in general. As a result, early patient flow control operations can take place as soon as issues are detected and armed with better knowledge about the process situation (or state).

The third related approach is the one from Poulymenopoulou et al. (2011). The idea they present is to build a system that integrates the disparate activities (both administrative and clinical) involved in an emergency care process, in an automated way. The process supports both pre-hospital and in-hospital activities, meaning that the complete flow (from the point of emergency to the point at which the patient is being discharged) is covered. Figure 36, which depicts the proposed architecture for their system, uses typical components such as a workflow module for workflow automation and a software module for implementing system and user activities.

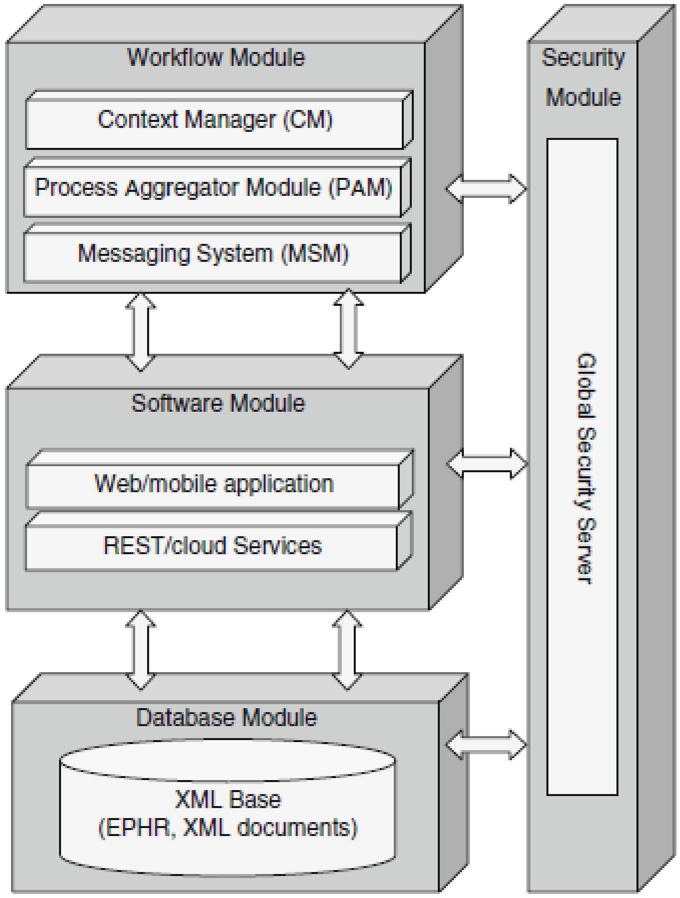


Figure 36 Cloud-based Workflow System for EMS (Poulymenopoulou et al., 2011)

An implementation of the system featuring Oracle BPM Studio is presented. If monitoring does not seem to be directly addressed, their platform allows for an improved visibility over process data with data captured during process execution and data accessibility from a centralized interoperable cloud data infrastructure. However, although the work focuses on an overall improvement of emergency care, no concern regarding the support for wait time management processes is addressed. Nevertheless, a focus is put on supporting medical information systems' interoperability and data exchange using a data layer implementing mainly Integrating Healthcare Enterprise (IHE) profiles and the OASIS EDXL specification. Security concerns are also addressed using popular authentication and authorization methods, such as open authentication (oAuth), to secure REST service execution and traditional data encryption and exchange, such as Advanced Encryption Standard (AES) and Secure Socket Layer (SSL) respectively. They aim to bridge the information gap between related emergency activities by proposing a cloud infrastructure to host patient information accessible to both ED personnel and EMS agency staff. An emergency care ontology for the patient is used.

As we can see, little existing literature tackles the problem of providing integrated data to hospital managers, especially with a process management focus. In addition, existing work that does leverage BPM technologies does not come close to accurately measuring wait times.

Table 3 summarizes how each approach performs against our requirements of interest (defined in Section 4.1.4); Y means that the requirement is satisfied, N that it is not satisfied, and P that it is partially satisfied.

Table 3 Comparison Between LA-BPMS and Related Work

Event Name	**R1**	**R2**	**R3**	**R4**	**R5**	**R6**	**R7**	**R8**	**R9**
LA-BPMS (Bougueng, 2013)	**Y**	**Y**	**Y**	**Y**	**Y**	**Y**	**Y**	**Y**	**Y**
Cloud-based EMS (Poulymenopoulou et al., 2011)	Y	Y	P	N	N	Y	N	Y	Y
Workflow-based RIS (Zhang et al., 2009)	Y	Y	P	N	N	Y	Y	Y	Y
Adaptative WfMS (van Hee et al., 2008)	Y	Y	P	N	N	Y	N	Y	N

It is also very important to note that some types of wait times can be accurately measurable with simple process automation techniques while *others require more sophisticated approaches*. Table 4 depicts, in the context of the case study presented in Chapter 5, what pieces of technology are sufficient to measure wait times for each of the types of wait times considered in the ACS clinical pathway supported. Many types of wait times actually *require* a BPM combined to a RTLS to be computed (i.e., one or the other on its own is insufficient), which is an interesting result of this case study.

Table 4 Technology Score Card for LA-BPMS Events

Event Name	RTLS	BPM	BPM + RTLS
TriageScore	–	X	–
PatientRegistered	–	X	–
PatientInED	X	–	–
WaitForConsultation	–	–	**X**
PhysicianInED	X	–	–
ConsultationStarted	–	–	**X**
PhysicianOutED	X	–	–
ConsultationCompleted	–	–	**X**
OrderRequest	–	X	–
WaitForOrderExecution	–	–	**X**
OrderExecutionCompleted	–	X	–
RequestForAdmission	–	X	–
WaitForBed	–	–	**X**
BedAssigned	–	X	–
PatientTransportRequest	–	X	–
WaitForTransport	–	–	**X**
PatientOutED	X	–	–
PatientTransportStarted	–	–	**X**
PatientTransportCompleted	–	–	**X**
PatientInCW	X	–	–
PatientOutCW	X	–	–
ProcedureStarted	–	X	–
ProcedureCompleted	–	X	–
PatientInCCL	X	–	–
PatientOutCCL	X	–	–
RequestForDischarge	–	X	–
PatientDischarged	–	–	**X**
HouseKeepingInCW	X	–	–
HouseKeepingOutCW	X	–	–

If the idea of context-aware workflows has existed for a while now, but its application to healthcare management is still in its infancy.

6.3. Threats to Validity

This section presents threats to validity identified in this thesis and discusses mitigation approaches taken along the way. Three types of threat to validity are addressed (Perry et al., 2000):

- Construct validity: refers to which extent the case study actually answers research our hypothesis.
- Internal validity: examines any bias and other confounding factors.
- External validity: verifies the extent to which our results can be generalized.

6.3.1 Construct Validity

One clear threat to construct validity for our study is its limited results. The results of our case study are arguably not sufficiently strong for us to drive conclusions with certainty. This is primarily due to the absence of quantifiable results from the simulation. However, our approach was strongly focused on the *feasibility* of the presented architecture to achieve real-time monitoring of patient care processes. So, data measurement has been traded for human validation. We have provided strong evidence with human recognition from professors, IT partners and healthcare professionals of the conviction of the realization of our goals (see Section 4.1.2) by a complete system reflecting on the prototype. This validation by domain experts is also significantly valuable and adopted in the context of Design Science research. The philosophy is that the input received by stakeholders will help the artifact (the prototype) evolve, be refined, and re-evaluated as part of the incremental process of Design Science research.

Another threat to validity is the fact that some design concerns were not specifically addressed, e.g., HIS integration, and regulatory and security compliance. Whereas regulatory and security compliance of business processes should be discussed in our platform, we felt that this is an entirely other domain by itself, and hence was left out of scope. However, we understand than extensive work will have to be conducted in the future in those areas to ensure modeled processes can be implemented in hospitals, e.g., based on the healthcare compliance work of Ghanavati (Ghanavati et al., 2007, Ghanavati 2013). The technical integration of real HIS (e.g., Osler's Meditech) with the solution system was not done, but we have acknowledged this limitation and have addressed its feasibility in Section 4.8.5.

6.3.2 Internal Validity

One internal threat to validity is related to the content of the cardiac process model developed for our evaluation, which could have been biased to "work well" with RTLS technologies. However, this model actually originates *from other people*, i.e., from Osler nurses who used IBM Blueworks Live to create its first version. Blueworks allows one to create non-executable process models mainly for process documentation. Moreover, the BPMN workflow model developed from that process model was initially started by an IBM BPM specialist who previously worked on it. We augmented this model to augment it with the concepts, logic and GUIs required to compute wait times and enable interactions with physicians/nurses and related systems (e.g., the RTLS via the complex event processing engine).

Another threat to internal validity could be about the nature of the validation survey conducted by a collaborator (A. Mouttham) during the simulations at the hospital. However, that survey was approved by the Research Ethics Board of William Osler Health Centre.

6.3.3 External Validity

Given the singular character of wait times and processes, and the general approach of our solution architecture to tackle related issues, we have shown how the system is capable of being applied to a representative cardiac process in a given hospital, empowering process execution, tracking process state and capturing wait times and service times. However, it is arguable to which extent this can be generalizable to other types of healthcare processes. Moreover, we have never demonstrated the applicability of our solution to the general problem domain since we have not evaluated it in different hospitals or with different pieces of technology. A mitigation point is that we have tried the system at two different locations (University of Ottawa and William Osler Health Centre), with different lab settings and different demo participants, and obtained similar positive results. Nevertheless, more case studies will have to be conducted with other processes, with other hospitals, with alternative technologies and in different settings.

Another important limitation to recognize in our work is that the use of location awareness to compute wait times is only relevant to the context where the state of an activity being delayed or not is at least partially dependent on the location of the involved assets or actors. This is true as those are the only types of information inferable by combining together location and process knowledge. However, because wait time occurs on an activity as a condition of the state of its environment (or context), we argue that the idea of context awareness in general can embody the strategy to be leveraged, along with real-time process monitoring, to get visibility over the occurrences of wait times on activities.

6.4. Chapter Summary

This chapter summarized the challenges we have faced when dealing with the problem domain that we have tackled and the technologies involved in the proposed solution. A comparison with closely-related approaches has been presented to better situate and evaluate our contribution to the problem resolution. Various threats to validity have also been identified, with a discussion of some mitigation strategies.

In the next chapter, we will conclude the entire body of our work and identify future work items.

Chapter 7. Conclusions

This chapter concludes this thesis by summarizing its contributions and by presenting future work opportunities.

7.1. Contributions

This thesis was mainly motivated by the necessity to reduce wait times in hospitals. Throughout its development, we have presented our work in understanding the problem domain and expressing our main objective to enable real-time care process visibility and improve patient care process state monitoring across a given hospital. We have presented a research solution leveraging existing technologies capable of providing the necessary functionality to achieve our objectives; notably BPM technologies and location tracking systems. Then, following the Design Science research methodology, we have presented the conceptual ideas for the development of a LA-BPMS to demonstrate our research hypothesis. We have also conducted different evaluations of our proposed solution. First, the presentation of a discussion assessing the quality of our design architecture against expected design characteristics, identifying strengths and weaknesses. Second, the development, construction and evaluation of a system prototype driven by the Osler hospital's specific problem context. Third, a comparative analysis of our proposed solution against closely related work, which presented the advantages of our solution over others towards the system objectives, but also highlighted the areas where work remains. From all this work, the contributions of our thesis can be expressed as follows:

- A proposition for a novel location-aware business process management system (LA-BPMS) for real-time monitoring of care processes. This approach has been put to the forefront with the purpose of improving real-time visibility over process execution, patient states and, consequently, patient wait times. This was done with an interest to bring in a managed view, accurate and granular process data in real time. To achieve this, we have presented the automation of a hospital care pathway using BPM techniques and a BPMS.

- The integration of location awareness capabilities to the BPMS using a RTLS for tracking patient and staff location, coupled with an information correlator to detect wait states in the process and hence accurately measure wait times in real time.
- The development of a system prototype instantiating the conceptual design with a specific process (cardiac emergency patient flow) and specific technologies.
- A demonstration that some wait times actually require a combination of BPMS and RTLS to be computed.
- The conduct of over twenty simulation sessions using the system prototype at Osler hospital and at the University of Ottawa, to validate research.
- A paper to be published at the *2013 Summer Simulation Multi-Conference* (SummerSim'13) showing concretely how our solution platform can be leveraged by running real-time simulations to aid short-term operational decision making regarding patient flow management and, precisely, reduction of wait times.

7.2. Future work

The future work items for this thesis are numerous in terms of the improvements to be brought to the current platform and the new work to be done for exploiting the platform to benefit intelligent tools for wait times reduction.

7.2.1 Integration with Health Information Systems to Facilitate Data Access and Communication

As explained at the end of Section 4.8.5, the development of HIS connectors supporting HL7 interfaces needs to be done in order to bring a certain level of completeness to the existing body of work. This will provide the ability to test the system prototype in a real context, integrating with real HIS. This will also provide a better evaluation of our solution idea.

7.2.2 Validation and Experiments

Stronger validation is required by conducting formal experiments whenever possible. Those experiments should aim at validating the hypothesis using a real context and involving a pool of healthcare professionals *in action*. This can also be used to evaluate the *usability* of the system.

7.2.3 Clinical Pathways

The existing Osler cardiac process developed for the prototype needs to be extended. Other *alternatives* in the flow should be implemented to really testify achievable flexibility and support. Also, given the high variability of clinical processes depending on the type of diseases we are dealing with, more clinical pathways should be modeled and tested in order to assess extensibility of the solution. For example, the complex case where a given patient is *simultaneously* engaged in multiple care pathways needs to be managed appropriately at the process model level.

7.2.4 Process Exception Design

A more complete exception model needs to be designed and integrated to the cardiac process model in order to improve its robustness and support. This will involve a prior exception design activity to identify known exceptions and resolution mechanisms to be implemented for each of them.

7.2.5 Security and Privacy

Security and privacy measures need to be investigated and implemented into the system to comply with the security and Privacy regulations of hospital environment. Work such as the one presented by Kim (2013) also needs to be conducted for our platform. Kim presents in his paper an analysis of security measures prior to the introduction of his RFID-enabled mobile solution in a ubiquitous healthcare network environment. He proposes a security protocol and authentication mechanism for RFID-enabled mobile device to securely exchange information with hospital systems on the network.

7.2.6 Extensions

The future work items listed in this section represents extension opportunities that should be conducted, leveraging the existing system to lower wait times and improve control over patient flow management:

- The platform can be extended to support additional real-time reporting capabilities on business metrics such as cost or resource utilization for high-level managers.
- Decision support systems can be built that leverage collected high-quality historical data to support patient flow management strategy. One example of this has been, as shown in the paper from Bahrani et al. (2013), the development of a real-time simulation system to predict process state in near future (e.g., 4 to 8 hours ahead) based on knowledge of initial current state provided by the real-time monitoring engine.
- Process mining techniques can be used to mine data and extract process behavior patterns based on accurate historical process information. Examples of such work can be found in papers like the one from Rebuge and Ferreira (2012), and the one from Mans et al. (2009). This can help hospitals better understand the particularities of their operations for better control and efficiency.
- Extension of the capabilities of the system by combining process knowledge with other context information other than location (and in addition to location) can also be achieved to provide a better intelligence to the system.

Much work remains to be done before location aware process management technologies become widely adopted in hospitals. However, it is the role of research to build artifacts that allow the science community to concretely investigate the problem and hopefully come up with working and *actionable* solutions that benefit both business organizations in their operations, and academic institutions in their quest for knowledge.

References

Amyot D. and Mussbacher G.: User Requirements Notation: The First Ten Years, The Next Ten Years. *Journal of Software (JSW)*, Vol. 6, No. 5, Academy Publisher, pp 747-768, May 2011.

Baarah A.: *An Application Framework for Monitoring Care Processes*. PhD thesis in Computer Science, University of Ottawa, expected in August 2013.

Baffoe S.: A Generic BI Application for Real-time Monitoring of Care Processes. Master's thesis in Electronic Business Technology, University of Ottawa, expected in May 2013.

Baffoe S., Baarah A., and Peyton L.: Inferring State for Real-Time Monitoring of Care Processes. *5th Int. Workshop on Software Engineering in Health Care (SEHC 2013)*, San Francisco, USA, IEEE CS, May 2013

Bahrani S., Bougueng R. T., Mouttham A. and Amyot D.: Real-time Simulations to Support Operational Decision Making in Healthcare. To be published in *Summer Simulation Multi-Conference,* SummerSim'13, July 2013.

Boulos M. N. K. and Berry G.: Real-time locating systems (RTLS) in healthcare: a condensed primer. *International Journal of Health Geographics*, Vol. 11, No. 1, June 2012, pp 25.

Bridgeland D. M. and Zahavi R.: Business Modeling: A Practical Guide to Realizing Business Value. *Elsevier Science, The MK/OMG Press*, Dec. 2008.

Buyurgan N., Landry S. and Philippe R.: RFID Adoption in Healthcare and ROI Analysis. In: *The Value of RFID, Springer London*, 2013, pp 81-96.

Canadian Institute for Health Information (CIHI).: Health Care in Canada, 2012: A Focus on Wait Times. *Government of Canada Publication* CN# H118-82/2012E, 2012, pp 31-37.

Canadian Institute for Health Information (CIHI).: Who Is Using Emergency Departments and How Long are They Waiting?. *CIHI Website*, ISBN: 1-55392-676-5, 2005.

Cheng C. and Chai J.: Deployment of RFID in Healthcare Facilities — Experimental Design in MRI Department. *Journal of Medical Systems*, Vol. 36, No. 6, Dec. 2012, pp 3423-3433.

Clarke L. A., Osterweil L. J. and Avrunin G. S.: Supporting human-intensive systems. *FoSER '10 Proceedings of the FSE/SDP workshop on Future of software engineering research*, ACM, 2010, pp 87-92.

Decker G., Dijkman R., Dumas M. and Garcia-Banuelos L.: The Business Process Modeling Notation. *Modern Business Process Automation*, 2010, pp 347-368.

Dresner H.: Business activity monitoring. *Gartner symposium*, Nov. 2003.

Evashwick C.: Creating the continuum of care. *Health Matrix*, Vol. 7, No. 1, 1989, pp 30.

Gattnar E., Ekinci O. and Detschew V.: Clinical Process Modeling and Performance Measurement in Hospitals. *Enterprise Distributed Object Computing Conference Workshops (EDOCW), 2011 15th IEEE International*, Sept. 2011, pp 132-140.

Ghanavati S.: Legal-URN Framework for Legal Compliance of Business Processes. *Ph.D. Thesis in Computer Science*, Ottawa-Carleton Institute for Computer Science, University of Ottawa, 2013.

Ghanavati S., Amyot D., and Peyton L.: Towards a Framework for Tracking Legal Compliance in Healthcare. *19th Int. Conf. on Advanced Information Systems Engineering (CAiSE'07)*, Trondheim, Norway, June 2007. LNCS 4495, Springer, pp 218-232.

Hall R. W.: Patient Flow: Reducing Delay in Healthcare Delivery. *Springer Science+ Business Media*, Vol. 91, 2006.

Haraden C. and Resar R.: Patient flow in hospitals: understanding and controlling it better. *Frontiers of Health Services Management*, Vol. 20, No. 4, 2004, pp 3-15.

Hevner A. R., March S. T., Park J. and Ram S.: Design Science in Information Systems Research. *MIS Quarterly*, Vol. 28, No. 1, March 2004, pp 75-105.

ITU-T: *Recommendation Z.151 (10/12) User Requirements Notation – Language definition*, Geneva, Switzerland, October 2012.

Keller G., Nüttgens M., and Scheer A.-W.: *Semantische Prozessmodellierung auf der Grundlage Ereignisgesteuerter Prozessketten (EPK)*. Veröffentlichungen des Instituts für Wirtschaftsinformatik (IWi), Heft 89, Universität des Saarlandes, Germany, January 1992.

Kim J. T.: Analyses of Security Issues with RFID Application Under U-Healthcare System. *Proceedings of the 2012 International Conference on Information Technology and Software Engineering, Lecture Notes in Electrical Engineering*, Vol. 210, 2013, pp 331-342.

Lee B. C. and Duffy V. G.: Impact of Healthcare Information Technology Systems on Patient Safety. *Human Interface and the Management of Information, Information and Interaction Lecture Notes in Computer Science*, Vol. 5618, 2009, pp 559-565.

Maglogiannis I. and Hadjiefthymiadesb S.: EmerLoc: Location-based services for emergency medical incidents. *International Journal of Medical Informatics*, Vol. 76, No. 10, 2007, pp 747-759.

Malik A.: RTLS for Dummies. *For Dummies, Wiley Publishing*, March 2009.

Mans R., van der Aalst W., Russell N., Moleman A., Bakker P. and Jaspers M.: YAWL4Healthcare. *Modern Business Process Automation*, Springer Berlin Heidelberg, 2010, pp 543-565.

Mans R. S., Schonenberg M. H., Song M., van der Aalst W. M. P. and Bakker P. J. M.: Application of Process Mining in Healthcare – A Case Study in a Dutch Hospital. *Biomedical Engineering Systems and Technologies*, Springer Berlin Heidelberg, 2009, pp 425-438.

OASIS: *Web Services Business Process Execution Language v2.0*, April 2007. http://docs.oasis-open.org/wsbpel/2.0/OS/wsbpel-v2.0-OS.html

OMG: *Business Process Model and Notation (BPMN)*, version 2.0, January 2011 (a). http://www.omg.org/spec/BPMN/2.0/

OMG: *Unified Modeling Language (UML)*, version 2.4.1, August 2011 (b), http://www.omg.org/spec/UML/

Ontario Ministry of Health and Long-Term Care.: Time spent in ER (Hours) by CTAS and by Patient Type. *Ontario Ministry of Health and Long-Term Care Website*, 2013 http://edrs.waittimes.net/En/CTAS.aspx?LHIN=0&hosptID=0&period=19

Ouyang C., Dumas M., van der Aalst W. M. P. and ter Hofstede A. H. T.: From Business Process Models to Process-oriented Software Systems. *ACM Transactions on Software Engineering and Methodology (TOSEM)*, Vol.19, No. 1, Aug. 2009, pp 2.

Pizer S. D. and Prentice J. C.: What Are the Consequences of Waiting for Health Care in the Veteran Population?. *Journal of General Internal Medicine*, Vol. 26, No. 2, Nov. 2011, pp 676-682.

Pautasso C.: RESTful Web service composition with BPEL for REST. *Data & Knowledge Engineering*, Vol. 68, 2009, pp 851-866.

Peffers K., Tuunanen T., Gengler C. E., Rossi M., Hui W., Virtanen V. and Bragge J.: The Design Science Research Process: A Model For Producing And Presenting Information Systems Research. *Proceedings of the first international conference on design science research in information systems and technology (DESRIST 2006)*, 2006, pp 83-106.

Perry D. E., Porter A. A., and Votta L. G.: Empirical studies of software engineering: a roadmap. *Conf. on The Future of Software Engineering*, ICSE’2000, ACM, 2000, pp 345-355.

Poulymenopoulou M., Malamateniou F. and Vassilacopoulos G.: Emergency Healthcare Process Automation Using Mobile Computing and Cloud Services. *Journal of Medical Systems*, Vol. 36, No. 5, Oct. 2012, pp 3233-3241.

Pourshahid A., Amyot D., Peyton L., Ghanavati S., Chen P., Weiss M. and Forster A.J.: Business process management with the user requirements notation. *Electronic Commerce Research*, Vol. 9, No. 4, Dec. 2009, pp 269-316.

Rebuge A. and Ferreira D. R.: Business process analysis in healthcare environments: A methodology based on process mining. *Journal Information Systems*, Vol. 37, No. 2, April 2012, pp 99-116.

Reijers H. A., Russell N., van der Geer S. and Krekels G. A. M.: Workflow for Healthcare: A Methodology for Realizing Flexible Medical Treatment Processes. *Business Process Management Workshops*, *Lecture Notes in Business Information Processing*, Springer Berlin Heidelberg, Vol. 43, 2010, pp 593-604.

Robić D.: Workflow testing. *MIPRO, 2010 Proceedings of the 33rd International Convention*, *IEEE*, May 2010, pp 320-323.

Schonenberg M. H., Mans R. S., Russell N. C., Mulyar N. A. and van der Aalst W. M. P.: Towards a Taxonomy of Process Flexibility (Extended Version). *BPM Center Report BPM-07-11*, 2007

Sell C. and Springer T.: Context-sensitive adaptation of workflows. *Proceeding ESEC/FSE Doctoral Symposium'09,* ACM, Aug. 2009, pp 1-4.

Tongchuan L. and Deyu Q.: Dynamic Workflow of Clinical Pathway System. *Proceedings of the 2012 International Conference on Communication, Electronics and Automation Engineering, Advances in Intelligent Systems and Computing*, Springer Berlin Heidelberg, Vol. 181, Jan. 2013, pp 109-113.

Tzeng S. F., Chen W. H. and Pai F. Y.: Evaluating the business value of RFID: Evidence from five case studies. *International Journal of Production Economics*, Vol. 112, No. 2, 2008, pp 601-613.

van der Aalst W. M. P., Pesic M. and Schonenberg H.: Declarative Workflows: Balancing between flexibility and support. *Computer Science - Research and Development*, May 2009, Vol. 23, No. 2, pp 99-113.

van Hee K., Schonenberg H., Serebrenik A., Sidorova N. and van der Werf J. M.: Adaptive Workflows for Healthcare Information Systems. *Business Process Management Workshops, Lecture Notes in Computer Science*, Springer Berlin Heidelberg, Vol. 4928, 2008, pp 359-370.

Vaishnavi V. and Kuechler B.: Design Science Research in Information Systems. *Association for Information System*, http://desrist.org/design-research-in-information-systems, 2011.

W3C: *Web Service Description Language (WSDL)*, version 1.1, March 2001. http://www.w3.org/TR/wsdl

Widjaya R., van der Horst F. and Seck M.: Performance Improvement in Health-care Processes. *Enterprise and Organizational Modeling and Simulation, Lecture Notes in Business Information Processing*, Springer Berlin Heidelberg, Vol. 88, 2011, pp 116-129.

Wieland M., Kopp O., Nicklas D. and Leymann F.: Towards Context-aware Workflows. *CAISE´07 Proceedings of the Workshops and Doctoral Consortium*, 2007, pp 11-15.

Wikipedia: *Location awareness*, 2013 (last accessed May 2013). http://en.wikipedia.org/wiki/Location_awareness

Wohed P., van der Aalst W. M. P., Dumas M., ter Hofstede A. H. M. and Russell N.: Pattern-based Analysis of BPMN - an extensive evaluation of the Control-flow, the Data and the Resource Perspectives (revised version). *BPM Center Report BPM-06-17*, 2005, BPMcenter.org

Yang Z., Wu C., Liu Y.: Locating in fingerprint space: wireless indoor localization with little human intervention. *Proceedings of the 18th annual international conference on mobile computing and networking*, ACM, 2012, pp 269-280.

YAWL Foundation: *YAWL: Yet Another Workflow Language*, 2013. http://www.yawlfoundation.org/

Zhang J., Lu X., Nie H., Huang Z. and van der Aalst W. M. P.: Radiology information system: a workflow-based approach. *International Journal of Computer Assisted Radiology and Surgery*, Vol. 4, No. 5, Sept. 2009, pp 509-516.

Zhu Q., Nie H., Lu X. and Duan H.: Radiology Workflow-Based Monitoring Dashboard in a Heterogeneous Environment. *2010 3rd International Conference on Biomedical Engineering and Informatics (BMEI 2010)*, IEEE, Vol. 6, 2010, pp 2494-2498.