RedUNCI
RED DE UNIVERSIDADES CON CARRERAS EN INFORMÁTICA

Computer Science & Technology Series

XIX Argentine Congress of Computer Science Selected Papers

Jorge Raúl Finocchietto | Patricia Mabel Pesado
(Eds.)

edulp
Editorial de la Universidad de La Plata

Computer Science & Technology Series

XIX Argentine Congress of Computer Science
Selected Papers

Computer Science & Technology Series

XIX Argentine Congress of Computer Science
Selected Papers

Jorge Raúl Finochietto / Patricia Mabel Pesado
(Eds.)

Edulp
Editorial
de la Universidad
de La Plata

Finochietto, Jorge Raúl
Computer Science & Technology Series: XIX Argentine Congress of Computer Science Selected Papers / Jorge Raúl Finochietto y Patricia Mabel Pesado; edición literaria a cargo de Jorge Raúl Finochietto y Patricia Mabel Pesado. - 1a ed. - La Plata: EDULP, 2014.
304 p.; 24x16 cm.

ISBN 978-987-1985-49-4

1. Informática. I. Pesado, Patricia Mabel II. Finochietto, Jorge Raúl, ed. lit. III. Pesado, Patricia Mabel, ed. lit. IV. Título
CDD 005.3

Computer Science & Technology Series
XIX Argentine Congress of Computer Science
Selected Papers

Diagramación: Andrea López Osornio

Editorial de la Universidad de La Plata (Edulp)
47 N.° 380 / La Plata B1900AJP / Buenos Aires, Argentina
+54 221 427 3992 / 427 4898
editorial@editorial.unlp.edu.ar
www.editorial.unlp.edu.ar

Edulp integra la Red de Editoriales Universitarias Nacionales (REUN)

Primera edición, 2014
ISBN 978-987-1985-49-4
Queda hecho el depósito que marca la Ley 11.723

Impreso en Argentina

Topics

XIV Intelligent Agents and Systems Workshop
Chairs Edgardo Ferretti (UNSL) Ricardo Rodríguez (UBA) Sergio A. Gómez (UNS)

XIII Distributed and Parallel Processing Workshop
Chairs Armando De Giusti (UNLP) Fabiana Picoli (UNSL) Javier Balladini

XI Information Technology Applied to Education Workshop
Chairs Cristina Madoz (UNLP) Sonia Rueda (UNS) Alejandra Malberti (UNSJ) Gladys Dapozo (UNNE)

XI Graphic Computation, Images and Visualization Workshop
Chairs Silvia Castro (UNS) Roberto Guerrero (UNSL) María José Abásolo (CIC – UNLP)

X Software Engineering Workshop
Chairs Patricia Pesado (UNLP) Elsa Estévez (UNS-UNU) Alejandra Cechich, Horacio Kuna (UNM)

X Database and Data Mining Workshop
Chairs Hugo Alfonso (UNLPam) Laura Lanzarini (UNLP) Nora Reyes (UNSL) Claudia Deco (UNR)

VIII Architecture, Nets and Operating Systems Workshop
Chairs Jorge Ardenghi (UNS) Nelson Acosta (UNCPBA) Hugo Padovani (UMorón)

V Innovation in Software Systems Workshop
Chairs Marcelo Estayno (UNLZ) Osvaldo Spositto (UNLaM) Pablo Fillottrani (UNS) Rodolfo Bertone (UNLP)

IV Signal Processing and Real-Time Systems Workshop
Chairs Oscar Bría (INVAP) Horacio Villagarcia Wanza (UNLP) Hugo Ramón (UNNOBA)

II Computer Security Workshop
Chairs Javier Diaz (UNLP) Antonio Castro Lechtaller (IESE) Javier Echaiz (UNS)

II Innovation in Computer Science Education Workshop
Chairs Cecilia Sanz (UNLP) Beatriz Depetris (UNTDF) Marcelo De Vincenzi (UAI) Andres Bursztyn (UTN)

III ETHICOMP LatinAmerica
Chairs Simon Rogerson (Monfort University - Reino Unido) William Fleischman (Villanova University - EE. UU.) Anne Gerdes (University of Southern Denmark - Dinamarca) Guillermo Feierherd (Universidad Nacional de Tierra del Fuego - Argentina) Mario Arias Oliva (Universitat Rovira i Virgili - España)

Scientific Committee

Coordination: Armando De Giusti – Guillermo Simari (Argentina)

Abásolo, María José (Argentina)
Acosta, Nelson (Argentina)
Aguirre, Jorge Ramió (España)
Alfonso, Hugo (Argentina)
Ardenghi, Jorge (Argentina)
Baldasarri, Sandra (España)
Balladini, Javier (Argentina)
Bertone, Rodolfo (Argentina)
Bría, Oscar (Argentina)
Brisaboa, Nieves (España)
Bursztyn, Andrés (Argentina)
Cañas, Alberto (EE.UU)
Casali, Ana (Argentina)
Castro Lechtaller, Antonio (Argentina)
Castro, Silvia (Argentina)
Cechich, Alejandra (Argentina)
Coello Coello, Carlos (México)
Constantini, Roberto (Argentina)
Dapozo, Gladys (Argentina)
De Vicenzi, Marcelo (Argentina)
Deco, Claudia (Argentina)
Depetris, Beatriz (Argentina)
Diaz, Javier (Argentina)
Dix, Juerguen (Alemania)
Doallo, Ramón (España)
Docampo, Domingo
Echaiz, Javier (Argentina)
Esquivel, Susana (Argentina)
Estayno, Marcelo (Argentina)
Estevez, Elsa (Naciones Unidas)
Falappa, Marcelo (Argentina)
Feierherd, Guillermo (Argentina)
Ferreti, Edgardo (Argentina)
Fillottrani, Pablo (Argentina)
Fleischman, William (EEUU)
GarcíaGarino, Carlos (Argentina)
GarcíaVillalba, Javier (España)
Género, Marcela (España)
Giacomantone, Javier (Argentina)
Gómez, Sergio (Argentina)
Guerrero, Roberto (Argentina)
Henning, Gabriela (Argentina)
Janowski, Tomasz (Naciones Unidas)
Kantor, Raul (Argentina)
Kuna, Horacio (Argentina)
Lanzarini, Laura (Argentina)
Leguizamón, Guillermo (Argentina)
Loui, Ronald Prescott (EEUU)
Luque, Emilio (España)
Madoz, Cristina (Argentina)
Malbrán, Maria (Argentina)
Malverti, Alejandra (Argentina)
Manresa-Yee, Cristina (España)
Marín, Mauricio (Chile)
Motz, Regina (Uruguay)
Naiouf, Marcelo (Argentina)
Navarro Martín, Antonio (España)
Olivas Varela, José Ángel (España)
Orozco Javier (Argentina)
Padovani, Hugo (Argentina)
Pardo, Álvaro (Uruguay)
Pesado, Patricia (Argentina)
Piattini, Mario (España)
Piccoli, María Fabiana (Argentina)
Printista, Marcela (Argentina)
Ramón, Hugo (Argentina)
Reyes, Nora (Argentina)
Riesco, Daniel (Argentina)
Rodríguez, Ricardo (Argentina)
Roig Vila,Rosabel (España)
Rossi, Gustavo (Argentina)
Rosso, Paolo (España)
Rueda, Sonia (Argentina)
Sanz, Cecilia (Argentina)
Spositto, Osvaldo (Argentina)
Steinmetz, Ralf (Alemania)
Suppi, Remo (España)
Tarouco, Liane (Brasil)
Tirado, Francisco (España)
Vendrell, Eduardo (España)
Vénere, Marcelo (Argentina)
VillagarciaWanza, Horacio (Argentina)

Organizing Committee

Universidad CAECE
Mar del Plata - Argentina

President: Finochietto, Jorge

Members:
Bacigalupo, Gustavo
Malbernat, Lucía
Varela, Analía
Scolari, Florencia
Fullana, Mayra
Pellerini, Cecilia
Vives, Juan Pablo
Wehrli, Luciano
Meijome, María Isabel

Preface

CACIC Congress

CACIC is an annual Congress dedicated to the promotion and advancement of all aspects of Computer Science. The major topics can be divided into the broad categories included as Workshops (Intelligent Agents and Systems, Distributed and Parallel Processing, Software Engineering, Architecture, Nets and Operating Systems, Graphic Computation, Visualization and Image Processing, Information Technology applied to Education, Databases and Data Mining, Innovation in Software Systems, Security, Innovation in Computer Education, Computer Science Theory, Signal Processing, Real time Systems and Ethics in Computer Science).

The objective of CACIC is to provide a forum within which to promote the development of Computer Science as an academic discipline with industrial applications, trying to extend the frontier of both the state of the art and the state of the practice.

The main audience for, and participants in, CACIC are seen as researchers in academic departments, laboratories and industrial software organizations.

CACIC started in 1995 as a Congress organized by the Network of National Universities with courses of study in Computer Science (RedUNCI), and each year it is hosted by one of these Universities. RedUNCI has a permanent Web site where its history and organization are described: http://redunci.info.unlp.edu.ar.

CACIC 2013 in Mar del Plata

CACIC'13 was the nineteenth Congress in the CACIC series. It was organized by the Department of Computer Systems at the CAECE University (http://www.ucaecemdp.edu.ar/) in Mar del Plata.

The Congress included 13 Workshops with 165 accepted papers, 5 Conferences, 3 invited tutorials, different meetings related with Computer

Science Education (Professors, PhD students, Curricula) and an International School with 5 courses. (http://cacic2013.ucaecemdp.edu.ar/escuela/cursos.php/).

CACIC 2013 was organized following the traditional Congress format, with 13 Workshops covering a diversity of dimensions of Computer Science Research. Each topic was supervised by a committee of 3-5 chairs of different Universities.

The call for papers attracted a total of 247 submissions. An average of 2.5 review reports were collected for each paper, for a grand total of 676 review reports that involved about 210 different reviewers.

A total of 165 full papers, involving 489 authors and 80 Universities, were accepted and 25 of them were selected for this book.

Acknowledgments

CACIC 2013 was made possible due to the support of many individuals and organizations. The Department of Computer Systems at CAECE University in Mar del Plata, RedUNCI, the Secretary of University Policies, the National ministry of Science and Technology, CIC and CONICET were the main institutional sponsors.

This book is a very careful selection of best qualified papers. Special thanks are due to the authors, the members of the workshop committees, and all reviewers, for their contributions to the success of this book.

Ing. Armando De Giusti
RedUNCI

Table of Contents

XIV

Intelligent Agents and Systems Workshop

G-JASON: An Extension of JASON to Engineer Agents Capable to Reason under Uncertainty

ADRIÁN BIGA[1] AND ANA CASALI[1,2]

[1] Facultad de Cs. Exactas, Ingeniería y Agrimensura
Universidad Nacional de Rosario (UNR)
Av. Pellegrini 250-S2000BTP, Rosario, Argentina

[2] Centro Internacional Franco Argentino de
Ciencias de la Información y de Sistemas (CIFASIS)
aebiga@gmail.com, acasali@fceia.unr.edu.ar

Abstract. *The Procedural Reasoning Systems (PRS) are one of the best implementations of the family of BDI agents (B: Belief, D: Desire, I: Intention). In this paper we present an extension of PRS to allow to create more flexible agents capable to represent the uncertainty of the environment and different relevance degrees in the agent's plans. The proposed extension was implemented in the JASON platform that permits to develop PRS agents in JAVA, giving them high portability.*

Keywords: *Procedural Reasoning Agents, JASON, BDI model, uncertainty.*

1. Introduction

In recent years there has been growing interest in modeling complex systems as multi-agent systems (MAS) [10]. The BDI architecture (B: Belief, D: Desire, I: Intention) ([7], [9]) is one of the most notorious architectures to model the agents that composed these systems. This architecture has been widely studied and used in relevant real applications [4]. The Procedural Reasoning Systems (PRS) [6] are the best known implementation of agents based on BDI paradigm. Since the first implementation of PRS system [5], several versions have been developed, regarding Java implementations we highlight two platforms currently used JACK[1] and JASON[2]. Actually, both the BDI architecture proposed by Rao & Georgeff [9] and the PRS systems [5] do not account for uncertain or gradual information. These models are based on bi-valued logics to represent the agent mental states (i.e. B, D and I) and this information support the agent decisions.
We consider that making the BDI architecture more flexible, will allow us to design and develop agents potentially capable to have a better performance in

1 http://aosgrp.com/products/jack/
2 jason.sourceforge.net/

uncertain and dynamic environments. Along this research line Casali et al. [2, 3] have developed a general model for Graded BDI Agents (g-BDI), specifying an architecture able to deal with the environment uncertainty and with graded mental attitudes. In this agent model, belief degrees represent to what extent the agent believes a formula is true. Degrees of positive or negative desires allow the agent to set different levels of preference or rejection respectively. Intention degrees give also a preference measure but, in this case, modeling the cost/benefit trade off of reaching an agent's goal. Then, agents having different kinds of behavior can be modeled on the basis of the representation and interaction of these three attitudes. The graded BDI model is based on the notion of multi-context system. This framework allows the definition of different formal components and their interrelation. In this agent model separate contexts are used to represent the agent's mental states (i.e. B, D and I) and each context is formalized with the most appropriate logic apparatus base on many-valued logics. The interactions between the components are specified by using inter-unit rules, called bridge rules. A detailed specification of g-BDI agents and their components can be seen in [3].

It has been observed that the g-BDI model provides a formal framework to develop more flexible agents that can properly represent the uncertainty of the environment and the agent's preferences [4]. However, a general platform that allows to implement particular g-BDI agents has not been developed yet. Inspired in the graded BDI agent model, in this work we present an extension of PRS systems to represent graded data structures for the agent's beliefs and plans.

2. Agent Architectures and PRS Systems

There exist different proposals for the classification of agent architectures. Following Wooldridge [10], we consider four classes for intelligent agents: logic-based, reactive, hybrid and practical reasoning architectures. The BDI agent model can be placed in this last category.
In the BDI architecture the agents have an explicit representation of their mental states:

- Beliefs (B): represent the agent's knowledge of the environment (including other interacting agents) and about himself.
- Desires (D): represent the agent's desires that he wants to achieve.
- Intentions (I): is a subset of consistent desires that the agent decides to achieve. Intentions derive in the actions the agent will execute at each time.

The PRS architecture developed by Georgeff and Lansky [6] was possibly the first architecture based on the BDI paradigm and has been used in diverse applications ([7], [8]). PRS agents seek to achieve their goals based on their beliefs about the world (environment) and they can also react in the presence of the occurrence of a new event. In this way PRS provides a framework were the goal-driven and event-driven behaviors can be integrated.
The PRS systems are composed by a set of tools and methods for the plan representation and execution. These plans or procedures are conditional sequences of actions which can be executed to achieve certain objectives, or react in particular situations.

3. JASON Architecture

The language interpreted by JASON is an extension of AgentSpeak (L) [1], an abstract agent language. AgentSpeak has a clear notation and is an extension of logic programming to design BDI agents. JASON compared to other BDI agent systems has the advantage of being multi-platform since it is developed in JAVA language.
An AgentSpeak (L) agent is created by the specification of a set of beliefs and a set of plans. Other important elements are the goals of the agent and triggers events that are useful to represent the reactive part of an agent. Then, the principal components in this agent model are the following:

- **Beliefs:** *represent the agent's beliefs about its environment.*

- **Goals:** represent the agent's goals; AgentSpeak (L) distinguishes two types of goals: achievement goals and test goals. Achievement goals state that the agent wants to achieve a state of the world where the associated predicate is true. A test goal returns unification for the associated predicate with one of the agent's beliefs, otherwise it fails.

- **Trigger Event:** defines which event may initiate the execution of a plan. An event can be internal, when a subgoal needs to be achieved, or external, when is generated from belief updates as a result of perceiving the environment.

- **Plans:** *invo*lve of the basic actions that an agent is able to perform on its environment. A plan is formed by a triggering event (denoting he purpose for that plan), followed by a conjunction of belief literals that represent the context. The context must be a logical consequence of the agent's current beliefs; in this case the plan is applicable. The remainder of the plan is a sequence of basic actions or (sub)goals that the agent has to achieve (or test) when an applicable plan is chosen for execution.

AgentSpeak (L) syntax is defined by the grammar shown in Figure 1.

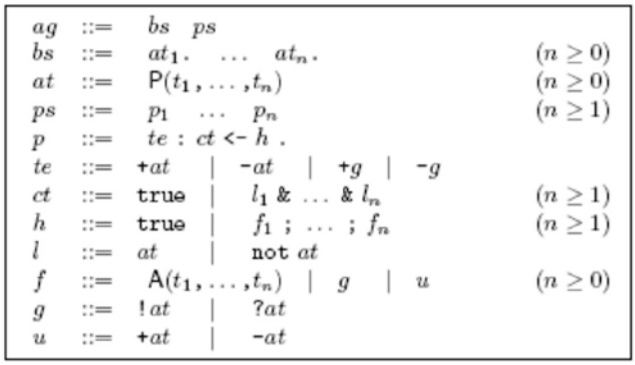


Fig. 1. *AgentSpeak syntax (Source: [1])*

An agent *ag* is basically specified by a set *bs* of beliefs (the agent's initial belief base) and a set *ps* of plans (the agent's plan library). A plan in AgentSpeak(L) is given by $p•:•:••=•te•:•ct•<•-h$, where *te* is the *triggering event*, *ct* is the plan's context, and *h* is a sequence of actions, goals, or belief updates; *te : ct* is referred as the *head* of the plan, and *h* is its *body.* Figure 2 describes how an interpreter of AgentSpeak (L) works. At every interpretation cycle of the agent program, the following procedures occur. AgentSpeak (L) updates a list of events, which may be generated from the perceptions of the environment (external events), or from the execution of intentions (actions), which produces an internal event (see (1) in Figure 2). After the event selection function (SE) has selected an event (2), AgentSpeak (L) has to unify that event with triggering events in the heads of plans (3). This generates a set of all *relevant plans*. By checking whether the context part of the plans in that set follow from the agent's beliefs, AgentSpeak(L) determines a set of *applicable plans* (4).This set of plans are called *options*. These options are generated from an external or internal event. Then, the option selection function (SO) chooses a single applicable plan from that set (5), which becomes the *intended means* for handling that event, and either pushes that plan on the top of an existing intention (if the event was an internal one), or creates a new intention in the set of intentions (if the event was external, i.e., generated from perception of the environment). All that remains to be done at this stage is to select a single *intention* (6) to be executed in that cycle (7). The intention selection function (SI) selects one of the agent's intentions.

Jason Grammar presents improvements over that of AgentSpeak (L) and can see its complete definition in [1]. The state of an agent, along its life cycle is represented by a set of beliefs and a set of plans and their behavior will be reflected in the behavior of the selection functions. One of the most important

differences that distinguish JASON syntax of AgentSpeak (L) (see Figure 1) is the inclusion of annotations in both beliefs and plans. These annotations in the beliefs and plans can be generated by *atomic_formula* using the following rules BNF:

beliefs -> (literal ``.")*; (3.1)

literal -> [~] atomic_formula (3.2)

plan -> [@ atomic_formula] triggering_event [: context] [``<-" body] ``." (3 . 3)

atomic_formula-> (<ATOM> | <VAR>) [``(" list_of_terms ``)"] [``[" list_of_terms ``]"] (3.4)

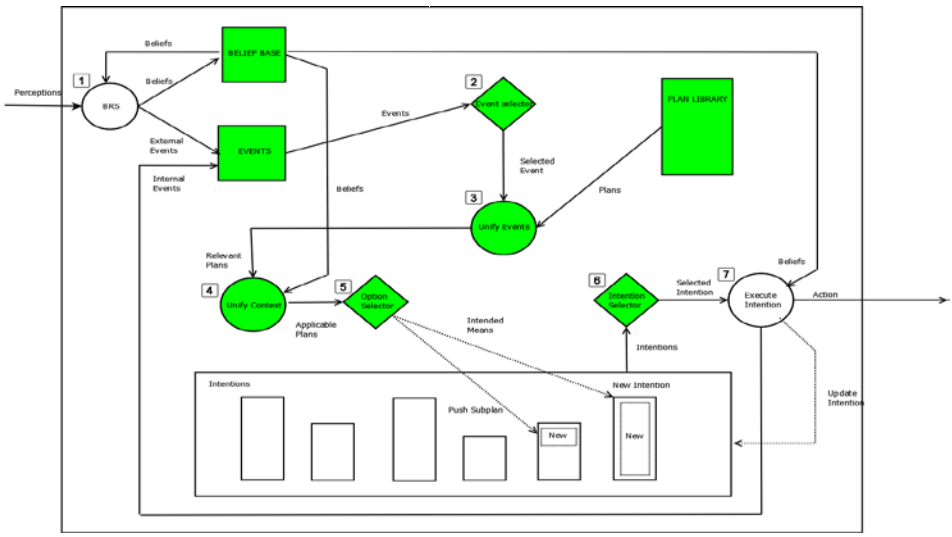


Fig. 2. *Diagram of an AgentSpeak (L) agent interpreter (Source: [1])*

The atomic formulas may contain annotations, this is a list of terms enclosed in brackets immediately following a formula (see rule 3.4). Plans are represented by labels like [@atomic_formula] (see rule 3.3).

4. G-JASON Extension

With the inspiration of the g-BDI model, we propose a syntactic and semantic extension that allows JASON to represent belief degrees and grades in plans, as these are the key elements in its architecture:

1. *Beliefs Degrees:* add numeric values in the range [0, 1] to represent the degree of certainty that a fact is true for this, that annotations are used. Then, a belief is defined by:

X[*degOfCert*($value_x$)] where X represents an atomic formula (representing a fact) and their degree of certainty $value_x$.
For example, the belief that there is a slight wind with a certainty of 0.7 is represented as: slight_wind [degOfCert (0.7)].
This degree will impact the selection order of events because the function selector SE will take first the belief with the highest degree.

2. *Plan Degrees:* the plan is formalized by:

@label[planRelevance(LabelDegree)] te [degOfCert(TEdegree)]:

ct [degOfCert(CTdegree)] <-body;

We propose to add each plan three values: the degree to label annotation (LabelDegree), the trigger event (TEdegree) and context (CTdegree):

- *LabelDegree* represents the measure of success of the plan (planRelevance), this value influences the selection of intentions and the selection of options. So that if two plans have passed the filters due to unifications, will run the one with the highest LabelDegree.

- *TEdegree* is applied to the trigger event and influences the selection and the unification of events.

- *CTdegree* is a degree in the context of the plan and will influence the unifying context.

Both degrees: TEdegree applied to the trigger event and CTdegree applied to the context, represent additional preconditions. They establish a kind of threshold, over these degrees these beliefs must be true in the current agent base to consider a plan as applicable. We have included in G-JASON the aforementioned degrees syntactically. Then, to have the desired semantics for them, we had to modify the selecting functions described and the processes of unification for the events and contexts of the plans. Next, we present how we have extended the syntax of JASON to get the new features.

4.1 G-JASON grammar

The words "planRelevance" (relevance of a plan) and "degOfCert" (degree of certainty) are reserved words and the degrees will be represented by floating numbers in the interval [0, 1]. JASON original syntax for beliefs, plans, trigger events (te) and contexts (ct) is modified:

1. Beliefs: are extended to allow them to have a degree of certainty:

beliefs-> (**literalC** ``.")*

literalC->[“~”] atomic_formula **anotC**

anotC->[degOfCert(``<valor>")]"

valor − •>•*dig*1,•*dig*2

*dig*1 − •>•0|1

*dig*2 − •>•0|1|...|9

2. Trigger Event (TE) of plan: is allowed to be graduated for this, the annotation is specialized to represent the degree of certainty:

triggering_event->(``+" | ``-") [``!" | ``?"] **literalC**

3. Context: the change takes place more directly reusing *anotC* (defined in item 1)

context-> log_expr **anotC** | true

4. Plans: the plan degree (planRelevance) is added into the plan

plan->``@" atomic_formula **anotG** triggering_event [: context]

[``<-" body]``."

anotG->”[planRelevance(``<valor>")]"

4.2 G-JASON: Implementing Extensions

Modifications were made to the JASON code, both in the agent's beliefs as in his plans. Also, the proposed changes have been implemented in the selector functions and processes of unification. G-JASON code is available in https://github.com/secharte/ab. We describe the proposed modifications by the cycle interpretation order of an agent program, following the flow of execution in its state machine (Figure 2). In this Figure we have colored all the functions and processes we have modified in G-JASON.
Selection of events: during the execution of the initialization process of the state machine of the event set and the set of all plans are loaded. The set of events has been generated from the beliefs defined for the agent and the plan library generated with all plans containing the agent.

1. *Option using belief degrees:* the agent current beliefs generate events that will unify with the triggering events of the plans. It was decided that the "event selector" select the event with a greater or

equal degree of certainty (degOfCert), modeling that the agent takes into account the most reliable facts first. The method receives as a parameter an event queue but returns the event that has the highest degree of certainty (degOfCert).

If events E1 and E2 are generated, where:
*E*1•=•[+ *sun*[*degOfCert*(0.8)]] y *E*2•=•[+ *wind*[*degOfCert*(0.7)]]
The selector returns the event E1 because it is the event with the highest degree of certainty.

2. *Enabling priorities:* to handle agent reactivity more directly, we equip the agents with the ability to activate a file "Priorities" where events are ordered by their priority. Then, when the priority is higher, the events will be more reactive since the event selector will consider these events first. For the treatment of this file, we have modified the event selection process again and these properties determine the order in which events are fired and in this case, the selection of events observed previously has no effect. For example, in a situation where you need to establish the order of importance of three factors: gas, light, vibration, where the importance of the treatment of these variables follows that order then, is appropriate to set the order of precedence in the file set “Priorities”.

Unification of Events: proceeds to unify the event selected by the selector against triggers events contained in the plans of the library plans. For example, if the agent has a Plan P:
@PlanP + heat_sensor(TEMP)[degOfCert(0.7)] : (TEMP < 300) <- init_alarm;call_supervisor
and an event : *E*•=•[+ heat_*sensor*(100)[*degOfCert*(0.8)]]:
Event E and plan P according to the original JASON unification process: heat_sensor (100) event unifies with heat_sensor(TEMP) of the plan and TEMP = 100.
This unification was extended and plans are selected if besides the original unification, the degree of the unified event is greater than the degree of the trigger event in the considered plan. In the example, the degree of the event E is 0.8 and the degree of trigger event of PlanP is 0.7. Then, this plan will be considered relevant.
Unification of context: the process of unification of events result in a set of relevant plans which are evaluated in the process of unifying context. Originally in JASON, the agent evaluates if the context is true.
For example, if we have *PlanP* (defined above) is relevant and if the agent has a belief in his base: *B*•=•[+ heat_*sensor* (100)[*degOfCert*(0.8)]]. As we mentioned in the previous unification, TEMP takes the value 100 and then the context (TEMP < 300) is true. The unification process was extended and the relevant plan will be selected if the belief degree of a fact in the base is greater than the corresponding degree of the context in the plan. In the

example, the degree of belief B is 0.8 and the degree of the context of the relevant PlanP is 0.6 therefore, PlanP is considered applicable.
Selection of options: Unification of contexts yield a set of applicable plans, the selector of options choose one of these applicable plans and then links this plan with an existing intention (if the event was internal, generated from a belief) or create a new intention that contains this option with its own plan associated . The options are represented by the applicable associated plans. In G-JASON the method "selectOption" was modified to select the most relevant plan (i.e. with higher planRelevance). Then, if these options (plans) are presented for the agent:
P1=@Plan1[planRelevance(0.85)] + heat_sensor(TEMP)[degOfCert(0.7)]: (TEMP < 300) [degOfCert(0.6)]<- init_alarm;call_supervisor
P2=@Plan2[planRelevance(0.65)] + heat_sensor(TEMP)[degOfCert(0.5)]: (TEMP < 100) [degOfCert(0.3)]<- init_heater;init_turbine
The selector returns the option P1 because it is the plan with highest value of planRelevance.
Selection of intentions: In the selection of options the set of intentions that the agent possesses are created or updated. The intention set is the input of the selection method of intentions, where an intention to run at the end of the cycle of interpretation of the agent will be chosen. Each intention is represented by a plan which body contains a list of actions for later execution. The order of execution of actions corresponding to an intention remains as the original version of JASON. As in the selection of options, the selector of intentions will consider the value of the plan (planRelevance) and the intention with greater relevance will be selected. After selecting the intention, the state machine executes one of the actions contained in the body of this plan.

5. Case study

We present a case study that shows the potential of the above-described platform. Assume that we want to model a supervisor agent of a rotary furnace used for melting metals. Three key variables are constantly analyzed in the operation: temperature, pressure and vibration. For this task the system has three sensors strategically placed. In this scenario the sensors provide information to the agent about the readings and the associated degrees of certainty that will represent the accuracy of the instruments used. The agent also has recommended actions for the different sensor readings, which will be modeled in the plans. For the case of the variable pressure for example, if the reading exceeds 35 bars the supervisor must order to close an injection valve oven (P3 plan) and if the pressure exceeds 70 bars should activate a general alarm, because exists the danger of explosion in the oven and they should take urgent actions (P4 plan) .
Furthermore, it is known that the accuracy of the different readings, for instance, the precision of the temperature sensors are 70% at 700 degrees so the agent represent this information by the degree of certainty of the

temperature reading (B1) setting it as 0.7. The plans degrees (planRelevance) were added to give relevance to the urgent need to perform certain actions by the supervisor agent. Given the importance of the pressure for furnaces, the agent represent with the highest degree of relevance the plans related to pressure, P3 (0.9) and P4 (0.85) respectively.
Then, the supervisor agent is modeled by the following belief set and plans:

B1= *heat_sensor*(700)[*degOfCert*(0.7)]

B2= *pressure_sensor*(80)[*degOfCert*(0.9]

B3= *vibration_sensor*(8)[*degOfCert*(0.6)]

P1=@temperature_alert[planRelevance(0.7)]: + *heat_sensor* (*TEMP*)[*degOfCert*(0.5)]•:•• *TEMP*•>•300[*degOfCert*(0.6)] •<• − *init_fan*

P2=@temperature_urgency[planRelevance(0.8)]: + *heat_sensor* (*TEMP*)[*degOfCert*(0.5)]•:• *TEMP*•>•600[*degOfCert*(0.6)] •<• − s***top_oven***

P3=@pressure_alert[planRelevance(0.85)]:+*pressure_sensor* (PRES)[degOfCert(0.7)]:*PRES*•>•35[*degOfCert*(0.8)] •<• − *close_valve*

P4=@pressure_urgency[planRelevance(0.9)]:+*pressure_sensor* (PRES)[degOfCert(0.7)]:*PRES*•>•70[*degOfCert*(0.8)] •<• − ***init_alarm***

P5=@vibration_management[planRelevance(0.6)]: +sensor_vibracion(NVL)[degOfCert(0.4)]:*NVL*•>•5[*degOfCert*(0.5)] •<• − *stop_motor*

Without using a "priorities file", the resulting order of actions executed by the agent is: (1) init_alarm, (2) stop_oven and (3) stop_motor.
It is observed that the first action is init_alarm, this action is related to the management of pressure modeled by the belief B2 (certainty 0.9) which is the most reliable. In addition, between the applicable plans (P3 and P4) is first executed plan P4 (Relevance 0.9), because is the most relevant plan. Executing this example in JASON original version, there is no possibility of representing uncertainty of beliefs or priority of plans and has executed the first action related to the temperature, given by the first input of sensor readings. The supervisor agent implemented in G-JASON improves its performance treating first the most accurate readings (or with higher priority activating the priorities file). In this example, they were the readings related to pressure sensor and execute actions based on this event, in the most relevant order.

6. Conclusion

In this paper we have presented a syntactic and semantic extension of JASON. This extension, called G-JASON, allow to engineer agents that can represent beliefs degrees and grades in the different components of plans. Also, the agent can manage a priority file to order the events depending on how reactive needs to be each one. To obtain the desired functionalities from this graded representation, the selection functions and unification processes for events and contexts were modified. G-JASON was implemented and is available as free software. Notice, that the syntactic and semantic extension proposed can be applied to any other PRS systems and their different implementations. Besides, we have presented a case study of a supervisor agent to show how the proposed extension is more expressive than the original version of JASON and can be suitable use to represent the uncertainty of the environment and relevance of plans. This model of agent using G-JASON allows as to obtain better results than the ones with the original version. Regarding to the graded BDI agent model that has inspired our work, it is pending to model degrees in desires (goals in JASON) since in the structure of PRS systems, the agent desires are not considered as a basic components and do not have an adequate representation. As future work we plan to move in this direction and also, we want to evaluate how these graded desires can impact the agent intention selection.

References

1. Bordini, R., Hübner, J. (2007). BDI Agent Programming in AgentSpeak Using JASON. John Wiley and Sons.
2. Casali, A., Godo, Ll., Sierra, C. (2011). A graded BDI agent model to represent and reason about preferences: Artificial Intelligence, Special Issue on Preferences Artificial Intelligence. vol. 175, pp. 1468–1478.
3. Casali, A., Godo, Ll., Sierra, C. (2005). Lecture Notes in Artificial Intelligence: Graded BDI Models For Agent Architectures. Leite, Joao and Torroni, Paolo, Springer-Verlag. 126–143. Berling Heidelberg.
4. Casali, A., Godo, Ll., Sierra, C. (2008). A Tourism Recommender Agent: From theory to practice. In: Revista Iberoamericana de Inteligencia Articial, vol. 12:40, pp. 23–38.
5. Georgeff, M., Pell, B., Pollack, M., Tambe, M., Wooldridge, M. (1987). The Georgeff, M. P., Lansky, A. L.: Reactive reasoning and planning. AAAI-87, 677–682, Seattle.
6. Belief-Desire-Intention Model of Agency. (1999). Intelligent Agents. Muller, J. P. and Singh, M. and Rao, A. S. Springer-Verlag, vol. 1365, Berling Heidelberg.
7. D'Inverno, M., Kinny, D., Luck, M., Wooldridge, M. (1998). A formal specification of dMARS. Intelligent Agents IV: Proc. Fourth International

Workshop on Agent Theories, Architectures and Languages. Singh, M.P. and Rao, A.S. and Wooldridge, M. Springer-Verlag, 155–176, Montreal.
8. Krapf, A., Casali, A. (2007). Desarrollo de Sistemas Inteligentes aplicados a redes eléctricas industriales. In: WASI-CACIC, Corrientes, Argentina.
9. Rao, A., Georgeff, M. (1995). BDI Agents from Theory to Practice. In: AAII.
10. Wooldridge, M. (2009). Introduction to Multiagent Systems, 2° Ed., John Wiley and Sons.

Spiking Neural Network with Hebbian Learning for Sparse Pattern Classification

IVÁN PERALTA[1], JOSÉ T. MOLAS[1], CÉSAR E. MARTÍNEZ[1,2], AND HUGO L. RUFINER[1,2,3]

[1] Laboratorio de Cibernética, Facultad de Ingeniería,
Universidad Nacional de Entre Ríos, CC 47 Suc. 3, E3100,
Ruta 11, km. 10, Oro Verde, Entre Ríos, Argentina.

[2] Centro de I+D en Señales, Sistemas e Inteligencia Computacional (SINC(i)),
Facultad de Ingeniería y Cs. Hídricas, Universidad Nacional del Litoral.

[3] CONICET, Argentina.

Abstract. *In past years it were different attempts to develop more realistic artificial neural networks that mimic the characteristics of their biological counterparts with more precision. They led to the proposal of* spiking neural networks*, which were mainly devoted to pattern classification. However, its applicability in the real world has been limited due to the lack of efficient training methods. In this paper, a new model of spiking network thought to classify sparse patterns is presented. This network can be trained using simple learning rules underlying hebbian principles. The architecture, operation and proposed training algorithm is described, along with test results on artificially generated patterns. Finally, the feasibility of its implementation in a programmable logic device FPGA is discussed.*

Keywords: spiking neural network*, hebbian learning, sparse patterns.*

1. Introduction

Artificial networks, inspired by biological neurons, are composed of basic units called neurons. These are interconnected by different weights that determine the intensity with which these neurons interact. The search for an analogy closer to biological reality has resulted in the last two decades to the appearance of the so-called Spiking Neural Networks (SNN). The use of SNN is growing due to their ability to cope with different problems in several areas such as pattern classification, mechanical monitoring, image processing, among others. These networks faithfully reproduce the neural biological systems with two effects. On the one hand they try to imitate the transfer of information between neurons through pulses, as performed in the biological synapses with Action Potentials. On the other hand, the networks make a dynamic processing of the signals within neurons.

Innumerable works have been developed that use SNN in different applications. In [2] is described biomedical engineering application where three SNN training algorithms are analyzed for detection of epilepsy and convulsions through classifications of EEG patterns. MuSpiNN is presented, another SNN model, to address the above problem. They have been also used in control systems, in [12] one SNN is used to control the movements of a robot to avoid obstacles using ultrasonic signals and in [4] a system for position control of laser is shown. More recently in [1] one network capable of memorizing sequences of events is described and in [6] one dynamic SNN is analyzed for spatio-and spectrum-temporal pattern recognition. In turn, it has been shown that this type of networks can be mapped more easily than traditional networks within programmable logic device such as FPGA [9, 10].
This work was motivated by the need to develop an efficient classifier for a special kind of patterns, originating from *sparse representations* of signals of interest. Such representations arise when analyzing signals by discrete dictionaries that use a lot of atoms, but where each one could described in terms of a small fraction of these elements [8]. Thus, the encoding achieved has most of its coefficients equal to (or near) zero [5]. This work focuses on training a pulsating network for recognition of sparse patterns, regardless of the method of obtaining such a representation. Therefore, we initially evaluate it on artificial sparse binary patterns randomly generated for each class.
The rest of the paper is organized as follow. Section 2 introduces the SNN structure, the neural models and its internal operation. Section 3 describes the sparse codification and training rules of the SNN. Section 4 describes the method, the experimental conditions, the results and a discussion of the feasibility to replicate this model of SNN in programmable logical device such as FPGA. Finally, Section 5 summarizes the conclusions and possible lines of future work.

2. Structure of the SNN

2.1 Connections and neurons types

SNN structure consists of two layers of neurons: an input detector layer and an output integrator layer. The first layer is connected on one side to the input pattern, which is a vector in $\mathbf{B}^{\mathbf{N}}$, where **B** is the set {0, 1}, and on the other side to the integrating layer. The amount of integrative neurons **I** is equal to the number of detectors **D** and pattern classes **C** that are desired classify, that is, **I** = **D** = **C**. In Figure 1, a network scheme is presented.
Between input vector coefficient i and each neuron d of the detector layer there is an interconnection weight $\mathbf{W}_{nd}$ that determine the importance of "1" (one) on classification of this pattern by the SNN. If the coefficient has value "0" (zero) does not stimulate the neurons in the classification of that particular pattern. Similarly, there exist weights between the sensing layer and the integrative layer, but in this case, there are two types of connections: excitatory –blue lines- and inhibitory –red lines-. Between input vector

coefficient ⅛ and each neuron ***d*** of the detector layer there is an interconnection weight $\mathbf{W}_{nd}$ that determine the importance of "1" (one) on classification of this pattern by the SNN. If the coefficient has value "0" (zero) does not stimulate the neurons in the classification of that particular pattern. Similarly, there exist weights between the sensing layer and the integrative layer, but in this case, there are two types of connections: excitatory –blue lines- and inhibitory –red lines-.

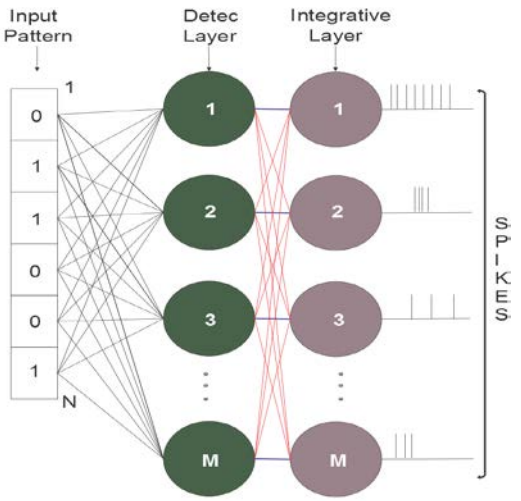


Fig. 1. *Structural diagram of the SNN, with* ***N*** *coefficients for input patterns vectors and* ***M*** *pairs of integrating-detector neurons.*

Each detector neuron excites its corresponding integrator neuron through positive weight ($\mathbf{W}_{ii}$ for all i) and inhibits the other by a connection with negative weights ($\mathbf{W}_{di}$ with i ` d). The communication between layers is produced by spikes. Each detector neuron receives N connections; one per each coefficient of the in put vector ½ A neuron (d) of this layer has N registers corresponding to coefficients ⅛ which are called $\mathbf{R}_{nd}$, also it has a $\mathbf{S}_d$ register that stores the sum of all $\mathbf{R}_{nd}$. Initially all registers are set to zero. The operation of unit is in discrete-time k. If in k_o it is introduced a pattern ½$^{(0)}$ to a detector neuron, each register $\mathbf{R}_{nd}$ associated to coefficient ⅛=1 increases in time with slope equal to the interconnection weight between register and coefficient. This increase will continue a preset number of iterations k_f, then each register is returned to zero in next iteration, waiting for the next input pattern ½$^{(1)}$. Therefore, a pattern is presented every k_f+2 iterations and each neuron processes the input to emit spikes if the register $\mathbf{S}_d$ overcomes the threshold. Once the neuron outputs a spike, there is a refractory period **T** in which no pulses are emitted although $\mathbf{S}_d$ is above the threshold. The structure and operation of the detector neuron is summarized in Figure 2.

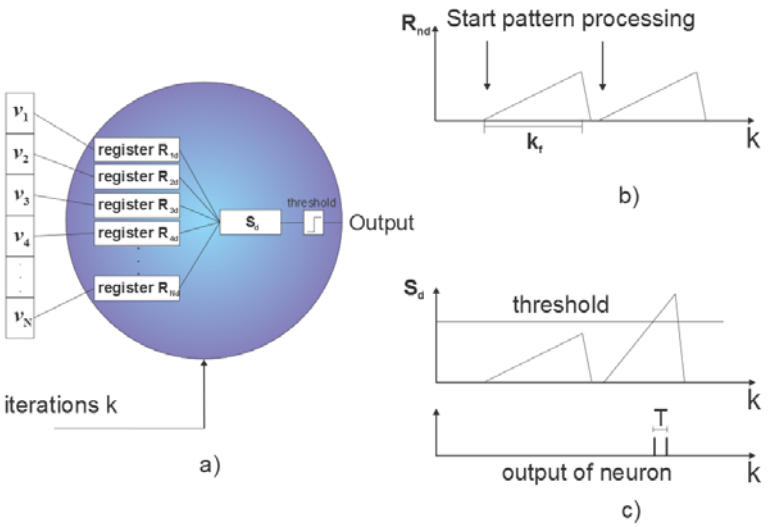


Fig. 2. *Structural and functional diagram of a detector neuron. a) Internal structure. b) Evolution of register after that a pattern is presented with ½=1. c) Internal registers, spikes output and refractory periods.*

2.1 Integrative Neurons

The integrating neuron model proposed in this work has a very similar structure to the detector neuron, but its internal operation is different. In Figure 3 the behavior for the activation of several of its inputs is described. When a spike of any input arrives, its corresponding register $\mathbf{R}_{di}$ will increase in time k with a slope equal to the weight $\mathbf{W}_{di}$ of interconnection between these neurons. If $i = d$ the weight is positive (excitatory connection) and if i ` d the weight is negative (inhibitory connection). Unlike the operation of a detector neuron, this increase is only for one iteration, then, each register will approach to zero with a negative slope $\mathbf{W'}_{di}$ that is less than $\mathbf{W}_{di}$. If the neuron receives another pulse, the register increases again. The register $\mathbf{S}_i$ store the sum of all register $\mathbf{R}_{di}$ of that neuron and if such registration exceeds a certain threshold, a spike is produced at its output. As in the detector neuron, after that a spike is emitted, there exist a refractory period **T** in which no spikes are emitted though $\mathbf{S}_i$ is above of threshold. The value of the refractory period is equal for all integrative neurons but is different at detector neurons.

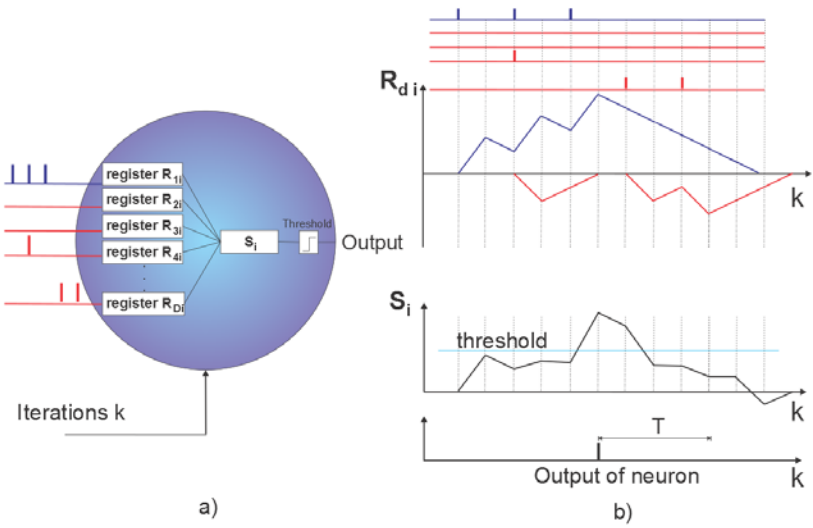


Fig. 3. *Structural and functional diagram of an integrative neuron. a) Representation of the internal structure. b) Integration of spikes.*

3. Learning algorithm

The learning method of this work is based on the idea of Hebb [4]. The training is carried out on the connections between detector layer and input patterns, that is, the weights $\mathbf{W}_{nd}$ described in Section 2.1 are determined. There are CE-vectors of each class to build the training patterns. Let $\mathbf{M}_d$ be a binary matrix with CE-columns and N-rows formed by training patterns of class d (recall that **D** = **C**, one detector for class). Here, each pattern forms a column vector of the matrix. Let $\mathbf{M}_d(n, k)$ be the coefficient of n-th row and k-th column of matrix. Then, search of weights can be written as:

$$W_{nd} = \begin{cases} \sum_{k=1}^{CE} M_d(n,k) & \text{si } \sum_{k=1}^{CE} M_d(n,k) > 0 \\ -\alpha \max_n \left\{ \sum_{k=1}^{CE} M_d(n,k) \right\} & \text{si } \sum_{k=1}^{CE} M_d(n,k) = 0 \end{cases} \quad \textbf{(1)}$$

where ± is a positive constant integer.

The interpretation of this rule is simple: the coefficients that are more active in one class determine big weights in its corresponding connection. If a coefficient is never active during training, is assigned a weight equal to the

negative of the maximum weight calculated for that class, which is scaled in ±-value. During evaluation, when a pattern is presented with coefficients "1" matching the most frequent activations occurred during training for any of the classes, the sum of the neuron registration for that class exceeds the threshold and the pattern is detected.

3.1. Function of integrative neurons

A detector neuron will emit more spikes if a pattern corresponds with the class of that neuron, since it will reach its threshold faster than otherwise. The function of integrative neuron is to integrate the spikes that arrive from its corresponding detector neuron, this way it will maximize the $\mathbf{S}_i$ register if the pattern belongs to the same class.
Often it is necessary to consider the history of patterns that were presented to the network through time, for example when it is required to detect patterns that belong to class of phonemes which can have several patterns [11]. The integrative neuron can remember the class of patterns of previous analysis. When the slope ($\mathbf{W'}_{di}$) is small, the present pattern has more influence in the next detection, but if it is excessively limited then it can take several iterations for $\mathbf{R}_i$ to return to zero, thus producing spikes that could affect the detection of patterns of other classes. In Figure 3, first an excitatory spikes arrives and then an inhibitory one, so it is presumed that first was presented a pattern belonging to the same class and then a pattern of another class. The inhibitory weights ($\mathbf{W}_{di}$ negative) increase detection specificity by sending inhibitory spikes towards neighboring integrative neurons.

3.2. Threshold determination

An important issue is the threshold determination, because the values can not be very high due to bad sensitivity on pattern detection, and additionally they can not have very low values due to bad specificity. Each detector-neuron has a threshold proportional to the maximum achieved by registers $\mathbf{R}_{nd}$ along all training as

$$threshold_d = \beta k_f \max_n \{W_{nd}\} \quad \textbf{(2)}$$

where ² < 1 is the proportionality constant.

A higher value of the iterations k_f of the detector neuron will rise the excitatory pulses at its corresponding integrative neuron and also its internal registers. Therefore, the threshold for these neurons was set as a ratio of k_f and is the same for all neurons:

$$threshold_i = \gamma k_f \tag{3}$$

where $^3 > 1$ is the proportionality constant.

4. Experiments and results

As mentioned above, this work does not address the problem of obtaining the sparse patterns starting from temporal signals. These patterns are randomly generated and used to train and evaluate the SNN. For this purpouse, two sets of patterns are needed: the training and the test ones, both with *C* classes.
The following algorithm is used to artificially generate the patterns.

1. Initially a set of binary vectors B^N is generated where B is the set {0, 1} and *N* an integer, with a priori higher probability of occurrence of 0's (sparse pattern).
2. All-zero vectors are removed.
3. The *K-Means* algorithm is applied to group the vectors into *C* different classes.
4. If the number of patterns in the class with fewer elements is less than the amount of the training patterns required, go back to step 1 with more initial vectors.
5. *CE* vectors of each class are taken to build the training set and so *CT* vectors to build the test set.

The same experiment was repeated 30 times to obtain an average result in the recognition rate. Figure 4 shows a set of patterns of 60 coefficients used in a experiment.
In order to evaluate the performance of the integrative layer, different number of patterns of the same class are introduced consecutively, from 1 to 5. In the first case, the patterns are placed alternately, in this case the effect of integrating frames will not be realized. In the last case, the introduction of the patterns is carried out with less mixing of the classes, so a maximum integration is expected.
The tuning of the parameters of the SNN was made in preliminary experiments. Table 1 list the parameters used.
Figure 5 show the recognition rates for each run and the average recognition rate for the complete experiment, depending on the number of same-class consecutive patterns (modes). It can be observed a recognition rate higher than 65% and near 83% for mode 1; while the average rate of global recognition of all realizations is close to 75%. While increasing the mode, these values increase due to the operation of the integrative layer.

Table 1. Parameters used with SNN architecture proposed in this work

Symbol	Value	Description
C	5	Number of classes
D	5	Number of neurons of the detecting layer
I	5	Number of neurons of the integrating layer
k_f	8	Rise iterations of a detector neuron
T_detector	1	Refractory period of the detector neuron
T_integrative	4	Refractory period of the integrating neuron
W_{di}	16 ($i = d$)	Weight interconnection between detector and integrating layer
	-13 ($i \grave{} d$)	
W'_{di}	5	Descent gradient registers of the integrative neuron
CE	200	Number of training patterns
CT	50	Number of test patterns
‘	4	Weights in detecting neuron
ß	0,33	Threshold in detecting neuron
“	1,5	Threshold in integrative neuron

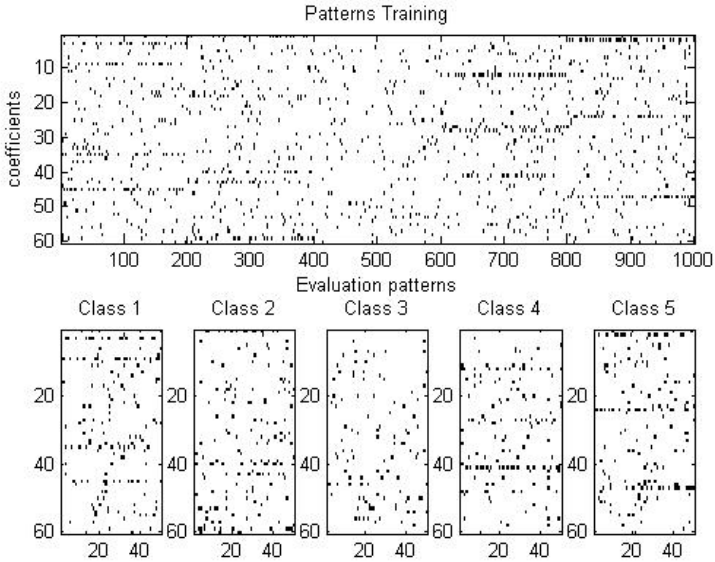


Fig. 4. *Patterns used in one realization of the experiments. Top: 200 patterns of training for each class. Bottom: 50 patterns of evaluation for each class.*

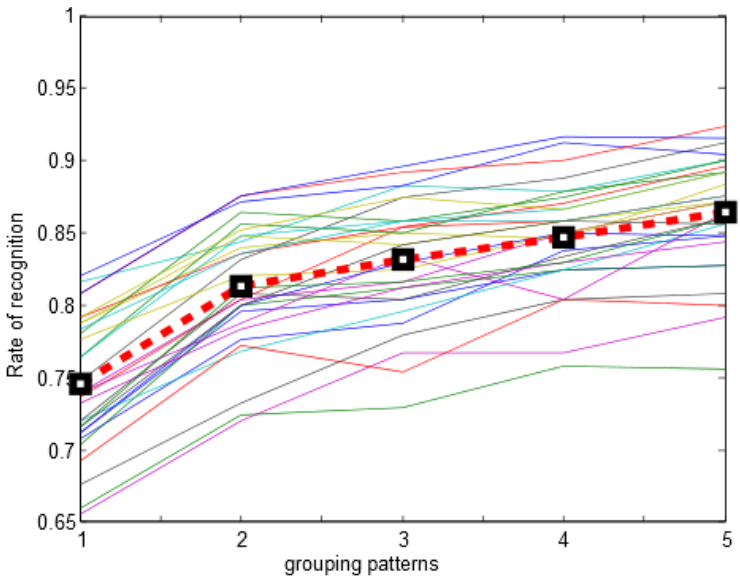


Fig. 5. *Recognition rates according to the 5 modes. In thin solid line, overall rate of pattern recognition for each realization. In dashed line, average recognition rate.*

The use of integer interconnection weights between layers, the structure of the SNN based on register and almost all integer parameters, becomes this architecture a very feasible design to be mapped on programmable logic devices such as FPGA with minimal changes. In a previous work [10], we presented the implementation of a trained SNN in an FPGA.

5. Conclusions

In this work we presented the design and implementation of a Spiking Neural Network, together with a training algorithm that allow the recognition of sparse input patterns.
The performance obtained is satisfactory, especially when the integration capabilities of these networks are exploited by presenting patterns consecutively for each class. The results are quite encouraging for several applications, specially ones involving sparse patterns, which were the motivation of this paper.
As a future work, we can mention that, with minimal changes, we could achieve a successful implementation of the SNN in a programmable logic device, e.g. a FPGA.

References

1. Borisyuk, R., Chik, D., Kazanovich, Y., Gomes, J.D.S. (2013). Spiking neural network model for memorizing sequences with forward and backward recall. Biosystems.
2. Ghosh-Dastidar, S., Adeli, H. (2009). A new supervised learning algorithm for multiple spiking neural networks with application in epilepsy and seizure detection. Neural Networks 22(10), 1419–1431.
3. Hebb, D.O. (1949). The organization of behavior: A neuropsychological approach. John Wiley & Sons.
4. Hulea, M. (2012). Using spiking neural networks for light spot tracking. In: Signal Processing Conference (EUSIPCO), 2012 Proceedings of the 20th European. pp. 1708–1712.
5. Hyvärinen, A. (1998). Sparse code shrinkage: Denoising of nongaussian data by maximumlikelihood estimation. Tech. rep., Helsinki University of Technology.
6. Kasabov, N., Dhoble, K., Nuntalid, N., Indiveri, G. (2012). Dynamic evolving spiking neural networks for on-line spatio-and spectro-temporal pattern recognition. Neural Networks.
7. MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations. In: Proceedings of the fifth Berkeley symposium on mathematical statistics and probability. vol. 1, p. 14.

8. Olshausen, B., Field, D. (1996). Emergence of simple cell receptive field properties by learning a sparse code for natural images. Nature 381, 607–609.
9. Pearson, M.J., Melhuish, C., Pipe, A.G., Nibouche, M., Gilhesphy, L., Gurney, K., Mitchinson, B. (2005). Design and FPGA implementation of an embedded real-time biologically plausible spiking neural network processor. In: Field Programmable Logic and Applications, 2005. International Conference on. pp. 582–585.
10. Peralta, I., Molas, J.T., Martínez, C.E., Rufiner, H.L. (2011). Implementación de una red neuronal pulsante parametrizable en FPGA. Anales de la XIV Reunión de Procesamiento de la Información y Control.
11. Rufiner, H.L. (2009). Análisis y modelado digital de la voz: técnicas recientes y aplicaciones. Ediciones UNL, Colección Ciencia y Técnica 284.
12. Wang, X., Hou, Z.G., Zou, A., Tan, M., Cheng, L. (2008). A behavior controller based on spiking neural networks for mobile robots. Neurocomputing 71(4), 655–666.

XIII

Distributed and Parallel Processing Workshop

N-Body Simulation Using GP-GPU: Evaluating Host/Device Memory Transference Overhead

SERGIO M. MARTIN[1], FERNANDO G. TINETTI[2,3], NICANOR B. CASAS[1], GRACIELA E. DE LUCA[1], DANIEL A. GIULIANELLI[1]

[1] Universidad Nacional de La Matanza
Florencio Varela 1903 - San Justo, Argentina

[2] III-LIDI, Facultad de Informática, UNLP
Calle 50 y 120, 1900, La Plata, Argentina

[3] Comisión de Inv. Científicas de la Prov. de Bs. As.
fernando@info.unlp.edu.ar, {smartin, ncasas, gdeluca, dgiulian}@ing.unlam.edu.ar

Abstract. *N-Body simulation algorithms are amongst the most commonly used within the field of scientific computing. Especially in computational astrophysics, they are used to simulate gravitational scenarios for solar systems or galactic collisions. Parallel versions of such N-Body algorithms have been extensively designed and optimized for multicore and distributed computing schemes. However, N-Body algorithms are still a novelty in the field of GP-GPU computing. Although several N-body algorithms have been proved to harness the potential of a modern GPU processor, there are additional complexities that this architecture presents that could be analyzed for possible optimizations. In this article, we introduce the problem of host to device (GPU) –and vice versa– data transferring overhead and analyze a way to estimate its impact in the performance of simulations.*

Keywords: *N-Body Simulation, GPU Optimization, Data Transference Overhead.*

1. Introduction

The N-body problem is largely known within the physics, engineering, and mathematical research faculty. It is commonly used to calculate – as precisely as possible – the future position, velocity, momentum, charge, potential, or any other aspect of a massive/charged body in regard to other bodies that interact with it within a time interval. Although some efforts have been made [1], many theorists have unsuccessfully tried for centuries to find a purely mathematical solution that could resolve any application of this problem in a series of steps linearly related to the amount (n) of bodies. Therefore, currently, the only way to approximate to a real solution is to use a differential method with tiny time slices (differentials) using the power of modern computers. However, this approach presents some downsides.

First, the usage of finite (as opposed to infinitesimal) time differentials is detrimental to the precision of the result. All positions and momentums are taken from the starting moment of the differential and are kept as constants during the calculation. Since the simulated forces remain constant during such differential, the results obtained suffer from a subtle degradation after each iteration. In consequence, the larger time differential is used, the more error is produced [2].
On the other hand, if we use smaller time differentials for the simulation, more iterations will have to be calculated until the end time is reached. As a result, simulations will require more computing time.
It is therefore important to keep in mind that the length of the selected time differential ultimately defines the precision admissible for the result expected, and the time that a computer will take to complete the simulation. Using high-precision libraries to augment the precision will also redound in increased computing time [3].
A way to calculate the amount iterations (*i)* to be simulated is to evaluate the inverse relation between the entire simulation time interval (Δ_t), and the time differential (∂t) as shown in Eq. (1).

$$i = \frac{\Delta_t}{\partial t} \qquad (1)$$

Yet another reason why time differential is an important factor to be taken into account is that it defines the amount of data being transferred between the host memory – traditionally known as RAM – and the device – graphics processing unit – through the PCI-Express bus. The time taken for the simulation will increase if more resources/time should be spent on unnecessary data transmission rather than just processing [4].
In traditional CPU-based schemes, this kind of data transference overhead is negligible since all data is present and up-to-date within the host memory after each iteration is calculated. In those cases, it is possible to use/access to all the positions of all *n* bodies and use them in real-time – for instance, for saving them into an output file, or rendering them into the screen. However, when GPU devices are used for these algorithms it is required to define explicit data transferences from the results obtained within the device memory back to the host in order to enable them for any use. Such overhead is detrimental to the overall performance and any efforts made to reduce it can yield significant optimizations [5]. Estimating such overhead is the object of our analysis in this article.

2. CUDA Implementation of N-BODY

The CUDA programming model as an extension of the C language provides an excellent scheme to parallelize scalable N-Body algorithms for GP-GPU architectures [6]. In this model, in opposition to the conventional multicore CPU model, the programmer is encouraged to create as many threads as needed depending on the amount of data elements in the problem. By doing this, it is possible, in the generality of cases, to yield the maximum performance from the many simple yet extremely parallelizable GPU cores. Of course, existing algorithms similar to the one used in this article are not exceptions [7] [8].

In the particular implementation of our N-body algorithm for CUDA, we created one thread per body in the simulation that will be in charge to execute the same function – called *CUDA kernel* – within the GPU processor. This kernel is programmed to execute the following steps:

- Load a single body's initial values from the device global memory. Each thread will load a different body based on its thread ID.
- For each other body in the simulation:
    - Load the body's values from the device global or shared memory.
    - Calculate the force that all other bodies impose to the loaded body.
- Save the new values for acceleration in the body data back into the device global memory.

Although threads perform better when no synchronization or communication functions are executed within the kernel, the CUDA architecture allows the programmer to specify *blocks* where a certain number of threads – depending on the infrastructure capability – can work coordinately. Based on this possibility, several memory access optimizations can be done in order to reduce the memory latency overhead.

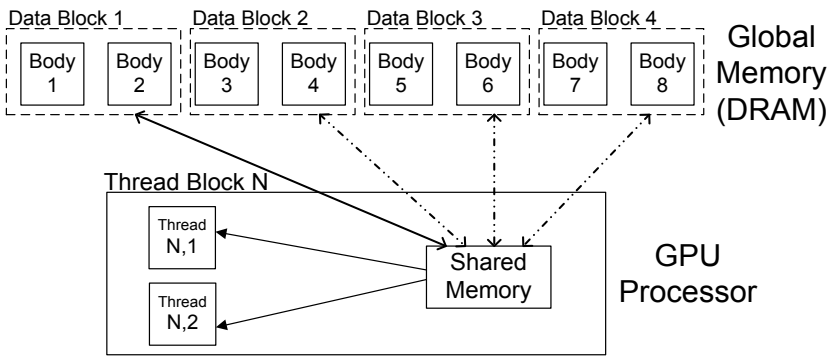


***Fig. 1**: Shared Memory utilization for the N-Body CUDA kernel*

The most successful optimization that we implemented was the usage of intra-block shared memory. Since constantly accessing global memory (low latency) forces executing threads to stall until data is effectively loaded, the overall performance is greatly reduced. For that reason, this architecture provides the programmer with intermediate memory banks – such as shared and register memory – which reside within the processor and could be used to reduce the amount of accesses to global memory.
Fig. 1 shows an example of such optimization where threads are grouped into one-dimensional blocks of size two[1]. In the same fashion, bodies' data present in global memory were divided in data blocks that are loaded, one by one, into the block shared memory. By doing this, all threads read data from the global memory only once and access it several times within the shared memory, thus reducing the total memory latency overhead.

The pseudo-code shown next represents the N-body kernel to be executed by every thread using the CUDA terminology:

```
void nbodyKernel(vec)
{
  thread_body = vec[TID + BID * BSIZE]

  For each i in  BCOUNT Do
     Shared_data[TID] = vec[TID + BSIZE * i]
     For each j in BSIZE Do
       Compute(thread_body, i*BSIZE + j)
       Update acceleration of thread_body
     End for
  End For

  vec[TID + BID*BSIZE] = thread_body
 }
```

Where:

- **thread_body** is the private memory for the body data pertaining to each thread.
- **vec** is the collection of bodies' data stored in the global memory of the device.
- **Shared_data** is a vector of size **BSIZE** where a complete block of data is stored and used as shared memory by a particular block of threads.
- **TID** is the thread identifier within the block.
- **BID** is the block identifier.
- **BSIZE** is the size of each block.
- **BCOUNT** is the amount of blocks created.

[1] This size was arbitrarily defined for simplicity reasons in this article while, in fact, blocks of 512 threads were actually used in our experiments.

A variety of other optimizations has been applied to the algorithm used in our experiments. Some of them have been already described in our previous work regarding the usage of multi-core clusters described in [9] and [10], and were used as the base for the CUDA version of our algorithm. However, more GPU-specific optimizations such as memory coalescing, shared memory usage, loop unrolling, interleaved thread-body processing were applied. Most of these optimizations are defined as good practices for any CUDA algorithm [11] [12]. Consequently, we assume for our experiments that the algorithm cannot be optimized any further.

3. Memory Transference Overhead

There are many types of research that requires scientists to run N-Body simulations in physics or engineering topics. In some, only the final result – for example, final position of the bodies involved – is needed; in others, it is more important to know the path that those bodies took during the simulation. Depending on each case – or a combination thereof –, scientists could choose to have the intermediate results stored in a device, transmitted through a network or displayed on a screen. In other cases, they would discard part or the entire journey in order to reduce memory transference overhead.

As mentioned for CPU based algorithms, all information is present in the host memory to be used at all times. Even if it is not used, stored, or sent through a network during the simulation, no extra time is required for memory transmission. However, in the case of GP-GPU algorithms, copying the data back to the host is necessary if some action is to be performed with them.

It is important to mention that, even if no intermediate data is needed for the simulation purposes, it is still necessary to guarantee results with acceptable precision by calculating the necessary amount of iterations of rather small time differentials until the total simulation time is reached. This forces every simulation to be performed with a certain number of iterations, even if only the final result is needed.

In this research, we sought to measure the impact of data transmission on the overall performance of the algorithms, letting aside other possible overheads introduced by its usage. By measuring this, we were be able to determine how much performance can be gained by only obtaining the final results of a N-Body simulation, in comparison with transmitting the intermediate results at each iteration. This allowed us to define the minimum and maximum performance gain possible regarding data transmission between the host (CPU) and device (GPU), having all other possible combinations (for instance, transmitting one result every two iterations) in between those two results.

We have verified through experimentation that these relations do not vary when the iteration count[2] is changed. Using a rather high amount of iterations, deviation becomes insignificant. For iterations counts close to 1,

[2]We used 65536 simulation iterations in all our experiments.

however, execution interference from the operating system introduces a more noticeable deviation.
In order to measure how much overhead is introduced by transmitting data at each iteration in relation to doing so only at the beginning and the end of the simulation, we ran the same set of tests to compare two algorithms. Algorithm *Nbody1* transmits – yet it does not use – intermediate results after each iteration, and *Nbody2* calculates all iterations without interruptions. The architecture used for our tests is shown in Table 1, and the results obtained are shown in Table 2.

Table 1: GPU Architecture used.

GPU Device	GeForce GTX 550Ti
CUDA Cores	192
Capability	CUDA 2.1
DRAM	1 GB GDDR5

Table 2: GFlop/s obtained for both versions

n	*Nbody1*	*Nbody2*
4096	271	307
8192	315	333
16384	343	353
32768	357	362

The first detail to notice from the results is that the difference between the GFlop/s obtained from both versions – the amount of overhead introduced by data transmission – reduces as *n* (amount of bodies being simulated) increases. This can be explained by the fact that the algorithm complexity is quadratic – becomes 4 times bigger, when we double the data – while data transmission increases only linearly regarding the problem size – transmission time will only double. In other words, as the threads take more time to execute the kernel, the overhead of data transmission becomes less significant. This relationship can be seen in the results presented in Fig. 2.

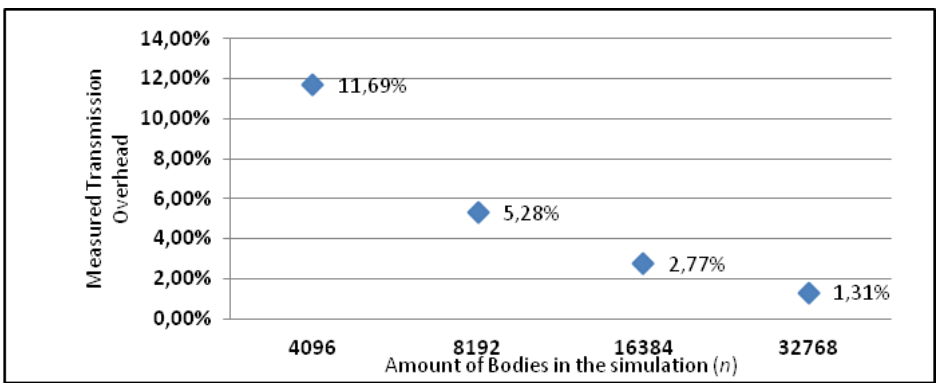


***Fig.2**: Measured transmission overhead ratio.*

4. Transmission/execution ratio evaluation

Since we have empirically obtained values of ratio between the transmission overhead size (expressed in Flop) for several cases of n, we deemed necessary to look for a relationship that could allow us to evaluate this ratio for any given n. Moreover, expressing this relationship in terms of bytes and Flops could allow calculating an estimate of transmission overhead for other types of algorithms, and not only for N-body problems. The first step in order to obtain such relationship is to find how data transmission requirement increases given a discrete increase in one body. We used profiling tools [13] and techniques [14] that obtain precise information about memory usage directly from the hardware counters. Fig. 3 shows the host/device transference volume for a single iteration with n=4096 elements and Fig. 4 shows the host/device transference volume for the same N count.

	Source	Destination	Size (bytes)
1	Host Unpinned	Device	917504
2	Device	Host Unpinned	917504

***Fig.3**: Transmitted data for one simulation iteration.*

	Function Name	Achieved FLOPS [1]: Single FLOP Count
1	nbodyKernel	20,400,472,064.00

***Fig.4**: Single-Precision Flop Count for the N-Body kernel per iteration.*

We need now a way to tell how much data transmission increases when adding another body to the simulation. This can be obtained by multiplying the calculated data transmission per iteration by count of the iterations being simulated and dividing it by the n used. The resulting expression will determine $T(n)$ – total transmission requirements – as a function of n. Eq. (2) shows n for the performed test:

$$T(n) = \frac{917504\ bytes}{4096}\ n = 224n\ bytes \tag{2}$$

The second step is to determine the size of the problem – expressed in MFlop – increases, given a similar increase in one element. To obtain this value, the same profiling tool allowed us to know how many floating operations were performed during the execution of the N-body kernel. As a result we can consider $F(n)$ – total amount of Flop – as a function of n, using the obtained single-precision Flop count per body/body compute as in Eq. (3):

$$F(n) = 1216\ n(n-1)\ FLOP \tag{3}$$

Having $T(n)$ and $F(n)$ as functions of n, it is possible to establish the relationship between the bytes of data being transmitted and the amount of Flops for each additional element of an algorithm with quadratic complexity. As a result, we can obtain a data overhead ratio (*dor*) as in Eq. (4):

$$dor(n) = \frac{N(n)}{F(n)} = \frac{224\ n\ bytes}{1216\ n(n-1)\ FLOP} \cong \frac{0{,}185}{(n-1)} \left[\frac{byte}{FLOP}\right] \quad (4)$$

The data overhead ratio (*dor*) obtained indicates, for this algorithm, how many bytes will be transmitted per floating point operation to be executed, given *n* elements. The *dor* value for every integer between 4096 and 32768 resemble the same inverse relation that our experimental measures shown in Fig. 2.

What is most important about this relation is that it is architecture-independent. This means that, no matter which GPU device model we use, the execution of this kernel will have the same ratio between data transmission and Flop processing. Thus, we only have to link it with the actual cost of transmission of this specific architecture to get its fraction of the performance overhead.

This proportion can be easily calculated since we know that the optimal performance of the GPU device does not vary, and it is only being reduced by the data transmission overhead. Thus, we can assume that the increase in the problem size – measured in GFlop – is the *r* relation for the performance drop observed in Table 2. For N = 4096, Eq. (5) reflects this increase:

$$r(4096) = \frac{307 - 271}{307} = 0.117\ \frac{Bytes\ (transfered)}{FLOP\ (processed)} \quad (5)$$

Therefore, if this relation is observed for $n = 4096$, there has to be a constant k that allows to represent perfectly the percentage of performance drop due to data transmission as seen from our measurements for this specific architecture. Calculating it from the $r(4096)$ ratio value, we obtained the result shown in Eq. (6):

$$r(n) = k * dor(n) = \frac{480}{(n-1)} \quad (6)$$

Having the relation *r* as a function of *n* will allow us to obtain the data overhead ratio for any *n* positive integer without having to perform any additional tests. As can be seen in Fig. 5, this inference matches perfectly with those measured in experiments and shown in Fig. 2.

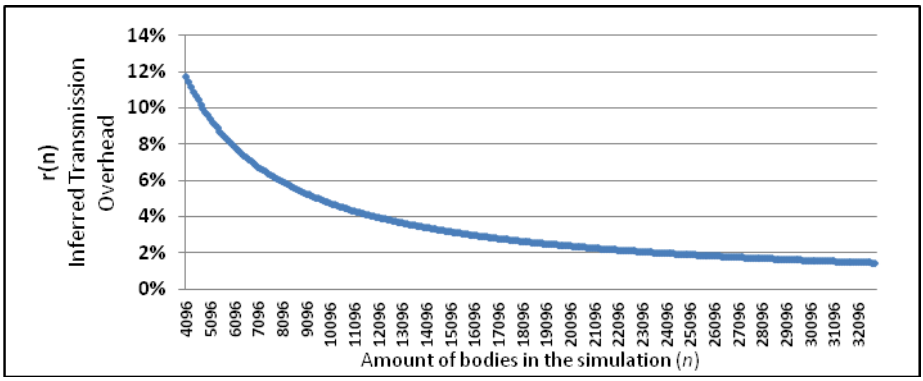


***Fig.5**: Inferred transmission overhead ratio.*

It is important to note that the proportion *k* obtained is the particular value of the GPU device – and underlying architecture – we used. Therefore, for each other architecture used to execute our kernel, a new value for *k* should be provided that reflects the estimated performance drop.

On the other hand, since the *dor(n)* ratio will not vary between different architectures, it should be calculated only once per algorithm. Then, just combining it with the appropriate *k* proportion to obtain the *r(n)* of that specific algorithm-architecture scenario.

The most valuable aspect of having such pre-calculated proportions is that a table containing different *k* values for the available architectures, and *dor(n)* values for the available algorithms, we could predict the performance drop for data transmission for combinations of algorithms and architectures that were not tested in actual experiments.

4.1 Interleaved transference per iteration ratios

We have surmised through our tests that the performance overhead of transmitting the simulation's intermediate results at each iteration for different values of *n* can be estimated. However, it could also be helpful to calculate the overhead if just a certain portion of intermediate results should be gathered. In such case, we would have data transmissions every *m* number of iterations.

As we could appreciate in the previous section, the *dor(n)* ratio for this algorithm was calculated for a 1/1 proportion of transmissions per iteration. However, if we wanted to change that proportion to 1/2, (which means: transmitting every two iterations) its value would proportionately drop to a half. Thus, we can extend our definition of *r* to take into account the amount of iterations per transmission as in Eq. (7):

$$r(n,m) = \frac{k * dor(n)}{m} = \frac{480}{(n-1)m} \tag{7}$$

In order to test the accuracy of the estimations made with the *r(n,m)* equation, we verified its estimations with a series of tests using variations for the values of *n* and *m*. We confirmed that every result approached the estimations with negligible deviations. Therefore, such equation could effectively determine the impact of data transmission in a wide variety of cases. In Fig. 6, we show different curves as functions of *n*, using different values for *m*.

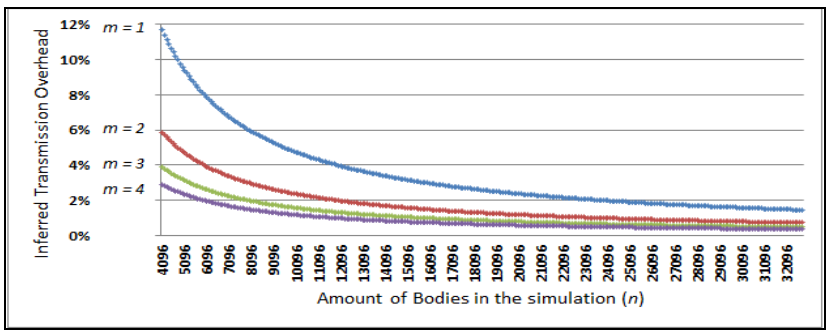


***Fig.6**: Inferred transmission overhead ratio for different values of m.*

The extent at which scientists will be willing to sacrifice intermediate data to be discarded by this approach should be considered for each case. However, having estimations for all combinations of *n* and *m* we can provide valuable clues for establishing the best option in each case.

5. Conclusions and Further Work

Balancing and fine-tuning the two factors that define the numerical precision of a simulation (total time interval and differential) can be a very complicated task. Since they define the amount of iterations being calculated, they will also define how much real time will be spent on the actual calculations. Certainly, for scientists only interested in a final result, estimating the negligible data transference overhead is of a little interest. However, for simulations that need to store intermediate data, time spent on device/host transference would become an important issue.

Providing scientists with a way to estimate how much processing time will be added in data overhead – given the amount of iterations and the interleave transfers – could allow them to estimate the best option for their time/architecture availability without having to try all the possible combinations, which could demand more effort than the performing the simulation itself.

In that sense, we have defined and tested a method to estimate the impact of data transmission vs. processing time in GPU-based simulations and N-Body algorithms. It could be evaluated for other types of GP-GPU algorithms since

we were able to narrow it down to a bytes/Flop relationship. We estimate that it would only require to calculate a data overhead relation – a constant for the algorithm –, and a data transmission cost – a constant for the device, as a metric for size in Flops. However, more testing on a diversity of algorithms and architectures should be performed in order to verify whether this relationship could be extrapolated.

The next step on this research will be focused in evaluating how other device performance counters could best allow us to estimate the costs of transmitting data, and how it could be optimized. Additionally, it will be necessary to determine how to estimate the transference overhead N-body algorithms ran in multiple device architectures or GPU clusters. Those cases hold much larger penalties for data transferences, and thus offer more challenges for data overhead estimation.

References

1. Diacu, F. (1996). "The solution of the n-body problem", *The Mathematical Intelligencer*, 18(3), 66-70.
2. Zadunaisky, P. E. (1964). "A method for the estimation of errors propagated in the numerical solution of a system of ordinary differential equations" *Proceedings from Symposium on The Theory of Orbits in the Solar System and in Stellar Systems*. August. Thessaloniki, Greece.
3. Nakayama, T., Takahashi, D. (2011). "Implementation of Multiple-Precision Floating-Point Arithmetic Library for GPU Computing". *Parallel and Distributed Computing and Systems*. Dallas, United States.
4. Gregg, C., Hazelwood, K. (2011). "Where is the data? Why you cannot debate CPU vs. GPU performance without the answer".*IEEE International Symposium on Performance Analysis of Systems and Software.* Texas, United States.
5. Mudigere, D. (2009). "Data access optimized applications on the GPU using NVIDIA CUDA". Master's thesis, *Technische Universität München*. Munich, Germany.
6. Nickolls, J., Buck, I., Garland, M., Skadron, K. (2008). "Scalable parallel programming with CUDA". *Queue - GPU Computing*, 6(2), 40-53.
7. Siegel, J., Ributzka, J., Xiaoming, L. (2009). "CUDA memory optimizations for large data-structures in the Gravit simulator". *International Conference on Parallel Processing Workshops*. Vienna, Austria.
8. Belleman, R. G., Bedorf, J., Zwart, S. P. (2008). "High Performance Direct Gravitational N-body Simulations on Graphics Processing Units". *New Astronomy*, 13(2), 103-112.
9. Tinetti, F. G., Martin, S. M. (2012). "Sequential optimization and shared and distributed memory parallelization in clusters: N-Body/Particle Simulation." *Proceedings of Parallel and Distributed Computing and Systems*. Las Vegas, United States.

10. Tinetti, F. G., Martin, S. M., Frati, F. E., Méndez, M. (2013). "Optimization and parallelization experiences using hardware performance counters". *International Supercomputing Conference Mexico*. Colima, Mexico.
11. NVIDIA CUDA™ Programming Guide Version 5.0, NVIDIA Corporation (2012).
12. NVIDIA CUDA™ Best Practices Guide Version 5.0, NVIDIA Corporation. (2012).
13. NVIDIA Nsight™ Visual Studio Edition 3.0 User Guide. NVIDIA Corporation. (2013).
14. Teodoro, G.L.M., Oliveira, R.S., Neto, D.O.G., Ferreira, R.A.C. (2009). "Profiling General Purpose GPU Applications". *21st International Symposium on Computer Architecture and High Performance Computing*. Sao Paulo, Brazil.

Managing Receiver-Based Message Logging Overheads in Parallel Applications

HUGO MEYER[1,1], DOLORES REXACHS[1] AND EMILIO LUQUE[1]

[1] Computer Architecture and Operating Systems Department,
Universitat Autònoma de Barcelona, Barcelona, Spain
{hugo.meyer}@caos.uab.es,
{dolores.rexachs, emilio.luque}@uab.es,
http://uab.es

Abstract. *Using rollback-recovery based fault tolerance (FT) techniques in applications executed on Multicore Clusters is still a challenge, because the overheads added depend on the applications' behavior and resource utilization. Many FT mechanisms have been developed in recent years, but analysis is lacking concerning how parallel applications are affected when applying such mechanisms. In this work we address the combination of process mapping and FT tasks mapping on multicore environments. Our main goal is to determine the configuration of a pessimistic receiver-based message logging approach which generates the least disturbance to the parallel application. We propose to characterize the parallel application in combination with the message logging approach in order to determine the most significant aspects of the application such as computation-communication ratio and then, according to the values obtained, we suggest a configuration that can minimize the added overhead for each specific scenario. In this work we show that in some situations is better to save some resources for the FT tasks in order to lower the disturbance in parallel executions and also to save memory for these FT tasks. Initial results have demonstrated that when saving resources for the FT tasks we can achieve 25% overhead reduction when using a pessimistic message logging approach as FT support.*

Keywords: *Fault Tolerance, Mapping, Message Logging, Multicore, Overheads.*

1. Introduction

Current High Performance Computing (HPC) systems are composed of nodes containing many processing units in order to execute more work in a short amount of time [1]. In order to take full advantage of the parallel environment, a good process mapping is essential. It is also important to consider that when executing parallel applications the fundamental objectives are: speedup as close as possible to the ideal (scalability) and efficient resource utilization.

Considering that applications are mapped into parallel environments in order to fulfill the above mentioned objectives, any disturbance may render all the mapping work useless. Currently, it is increasingly relevant to consider node failure probability since the mean time between failure in computer clusters has become lower [2] and this may cause loss of significant computation time in long-running applications. Indeed, successful completion of executions should be added to the list of fundamental objectives. In this vein FT techniques are gaining importance when running parallel applications. Nevertheless, FT mechanisms introduce disturbance to parallel applications in the form of overheads, which if not managed can result in large performance degradations, thus FT mechanisms that do not endanger scalability (uncoordinated approaches) are preferred.

Many recent works focus on finding the best checkpoint interval, or determining the best checkpoint or message logging approach for parallel applications [3][4] but few works address assigning resources for fault tolerance tasks considering applications' behavior [5].

When single-core clusters were the only option to execute parallel applications, there were not too many choices when talking about sharing resources. As there was only one computational core available, parallel applications share this resource (as well as the memory and cache levels) with the FT tasks if there was not dedicated resources. Considering that current clusters have more cores and memory levels, there is a need to develop mapping policies that allow parallel applications to coexist with the FT tasks in order to reduce the disturbance caused by these tasks. There is also important to consider that the number of cores has been multiplied by 8, 16, 32, 64 and usually the networks used in these clusters have not increase their speed to the same extent.

The main objective of this work is to determine the configuration of parallel applications in combination with a pessimistic receiver-based logging approach that minimizes the added overhead. We analyze parallel applications and obtain information that allows us to configure properly the FT tasks, specifically we determine if the best option is to share (compete for) resources with application processes or save resources for the FT tasks in order to reduce the introduced disturbance. In order to provide the configurations we consider the balance between computation and communication, message sizes and per-process memory consumption among other values.

The rest of the paper is organized as follows: Section 2 describes related work. Section 3 presents an analysis of the possible scenarios when executing a parallel application. Section 4 describes how to analyze a parallel application in order to find the most suitable message logging configuration. Section 5 shows the experimental validation and finally section 6 draws the main conclusions and mentions future works.

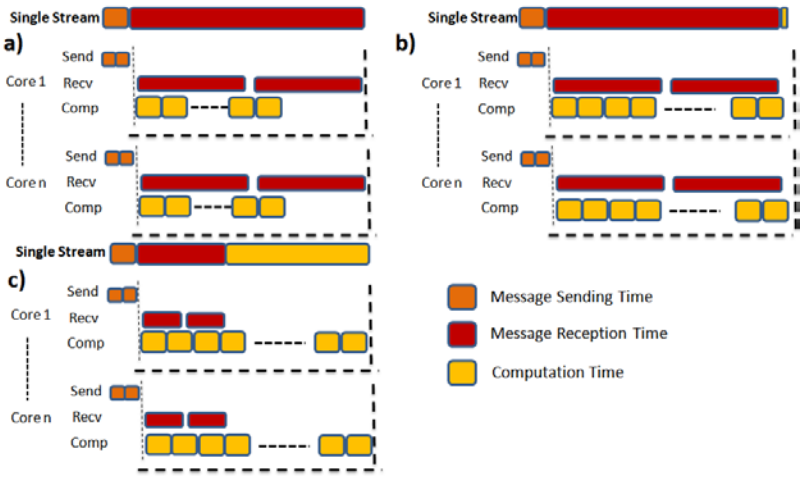


Fig. 1. *Parallel Executions Scenarios in a SPMD App. a) Communication Bound. b) Computation and Communication overlapped. c) Computation Bound.*

2. Related Work

In order to provide fault tolerance to parallel applications many strategies have been designed using message logging approaches [2][4][3]. Message logging approaches can sustain a much more adverse failure pattern, mainly due to a faster failure recovery. The main disadvantage of message log schemes is that they suffer from high overhead during failure-free executions [6] but they are a scalable solution since only failed processes must rollback, unless the domino effect is not addressed. Usually message logging techniques are used in combination with uncoordinated checkpoint approaches. Uncoordinated approaches are a good solution because there is no need for coordination between processes and there is no dependency on global components that could cause bottlenecks and compromise applications' scalability.

Following these lines, to develop this work we have used the RADIC (Redundant Array of Distributed and Independent Controllers) architecture [7] which uses a pessimistic receiver-based message logging technique in combination with an uncoordinated checkpoint approach in order to give application-transparent and scalable FT support for message passing applications.

In [3] a comparison between pessimistic and optimistic sender-based logging approaches is presented where both seem to have a comparable performance. Nevertheless, when using sender-based approaches we should consider that in the presence of failures, processes that were not involved in the failure may need to re-send messages to restarted processes, and also garbage collection is complex. The pessimistic receiver based message logging

approach of RADIC may be more costly than a sender based approach, but it guarantees that only failed processes will rollback to a previous state, without needing the intervention of other processes during re-execution.
In [8] was proposed a mechanism to reduce the overhead added using the pessimistic receiver-based message logging of RADIC. The technique consists in dividing messages into smaller pieces, so receptors can overlap receiving pieces with the message logging mechanism. This technique and all the RADIC Architecture has been introduced into Open MPI in order to support message passing applications.
In [9] was presented an algorithm for distributing processes of parallel applications across processing resources paying attention to on-node hardware topologies and memory locales. When it comes to combine the mapping of FT tasks, specifically message logging tasks, with application process mapping, to date, no works have been published to the best of our knowledge.

3. Analyzing Parallel Applications Behavior

Current HPC parallel applications are executed on multicore systems, and the executions usually aim for almost lineal speedup and efficient resource utilization. In Figure 1 we present the three main scenarios possible when mapping applications in multicore systems. It is important to highlight that in this figure we are considering one iteration of a SPMD application. In Figure 1 we decompose the Single Stream in communications and computations operations. The main scenarios are:

1. **Communication Bound:** Applications in which the processes are waiting because the communications take more time than the computations belong to this scenario. In Figure 1a we show how a communication bound application behaves (we are using as an example a SPMD application, where all processes do the same thing and each message goes from one process to another in a different core). In this figure we focus on showing how reception times (non-blocking send operations do not delay considerably the execution) can influence highly the execution time of a parallel application.
2. **Balanced Application:** This scenario is the best regarding efficient resource utilization, because the computational elements are working while the communication takes place. However, this behavior is very difficult to obtain because a previous study of the application is needed in order to completely overlap computations and communications (Figure 1b).
3. **Computation Bound:** When operators try to make a good use of the parallel environment they try to maintain the CPU efficiency high. Then in order to avoid the communication bound scenario it is recommended to give more workload per process which usually leads to a computation bound scenario. Figure 1c illustrates this scenario.

When characterizing a parallel application, it is also important to consider the number of processes that will be used, the number of nodes and the memory consumption of each process. This analysis should be done in combination with the analysis of the parallel environment in order to determine resource utilization. In this paper, we have characterized the parallel environment using application kernels and we consider the application phases (repetitive pieces of the parallel execution) that have the biggest weights during application execution.

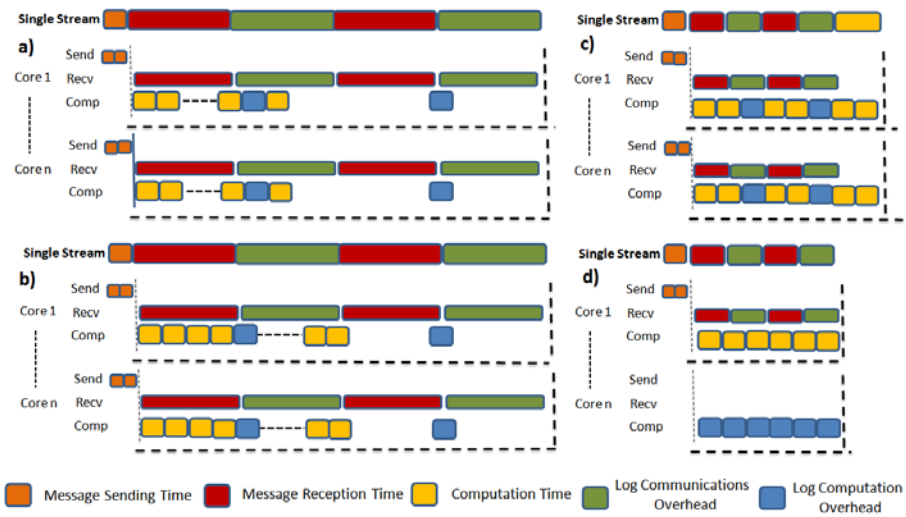


Fig. 2. *Parallel Executions Scenarios with Message Logging. a) A communication bound application. b) A balanced application becomes comm. bound. c) A computation bound application stays as it was. d) A computation bound application becomes balanced.*

In order to find the most appropriate configuration of the message logging approach, we should analyze how the parallel application and the logging approach coexist in the parallel machine. There will be two parallel applications that will compete for resources, thus it is critical to analyze the influence of FT in application behavior.

4. Analyzing Message Logging Process Mapping

Most of the impact of a pessimistic receiver-based message logging protocol concentrates on communications and storage (memory or hard disks), but there is also a small impact on computations because FT tasks also need some CPU cycles in order to carry on their work.

For the analysis in this paper we have considered the pessimistic receiver-based message logging protocol used in RADIC. RADIC main components are shown on Figure 3, Protectors' main functions during the protection stage are: establish a heartbeat/watchdog mechanism between nodes (low CPU consumption operation, do not depend on application behavior) and to receive and manage message logs (CPU consumption depends on application) and checkpoints from observers (infrequent operation). All communications between processes go through Observers and each received message is sent to a corresponding logger thread (usually the protector of RADIC is drawn as an equilateral triangle, but in this case we have split it two right triangles to distinguish the main operations).
In order to reduce the impact of the pessimistic receiver-based logging protocol of RADIC we propose to save computational cores for the logger threads (Namely, threads that are in charge of receiving and logging messages of other processes), thus avoiding the competition for CPU between application processes and logger threads. According to this protocol every message should be logged in a different computing node (there are no dedicated nodes, but the usage of Spare Nodes is considered in [7]), then there is a considerable increase in the transmission time of each message when RADIC protection is activated. Thus, when executing a parallel application with RADIC the previous scenarios change (Figure 1), because the processes will be waiting a longer time for each message and the computation will be affected by the logger threads.

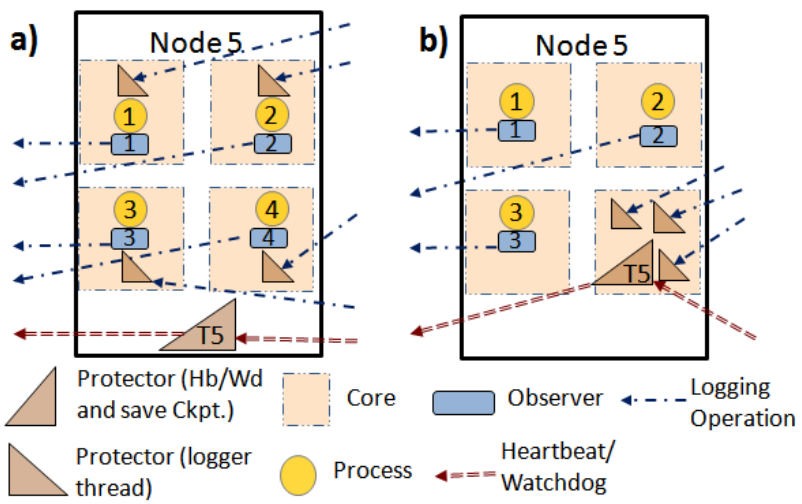


Fig. 3. *RADIC Processes' Mapping. a) Logging threads equitably distributed among cores. b) Protector with own resources.*

In order to reduce the overheads of the message logging approach, we analyzed how the application will be affected when introducing this logging technique. In Figure 2a we can observe how in a communication bound application, the difference between communications and computations becomes higher, and the overhead added in computations does not affect the iteration time. A balanced application without message logging will become communication bound when message logging is used (Figure 2b).
In these two scenarios the message logging overheads cannot be hidden, but when it comes to computation bound applications we can manage the mapping of the logger threads so as to distribute the overheads equally among the processes (Figure 2c). Alternatively, we can choose to save some computational cores in each node in order to avoid context switch between application processes and logger threads (Figure 2d).
Considering that many parallel applications are executed with balanced per-core workload, our default proposal is to distribute the overhead in computation produced by the logger threads among application processes residing in a node (Figure 3a). Moreover, we characterize the parallel application in order to find the computation-communication ratio, and if the application is computation bound, we analyze the overheads produced in computations. If these overheads make the application behave as in Figure 2c, we propose saving cores in each node in order to assign them to the logger threads, obtaining the behavior showed in Figure 2d. Figure 3b shows how we assign the logger threads and other protectors' functionalities when using own resources for them.
As saving cores may make the initial mapping change, we also analyze if the new mapping does not negatively affect the execution, resulting in a worse performance than the default option.
Another important aspect that we analyze is the per-process memory consumption. This is significant because we have the option of storing the message log in memory instead of hard disks as this allows us to avoid bigger delays when storing messages. When we put less processes per node, we can save more memory for the message log, thus there is more time to overlap the flush-to-disk operation with receptions of new messages to log. Also, we can use longer checkpoint intervals if we consider an event-triggered checkpoint mechanism where a full message log buffer triggers a checkpoint.

5. Experimental Validation

The main approach presented in this paper focus on resource assignation to decrease logging overheads and save memory for FT tasks. In this section we present experimental evaluation that has been carried out in order to probe our hypothesis.
The experiments and characterizations have been made using a Dell PowerEdge M600 with 8 nodes, each node with 2 quad-core Intel® Xeon®

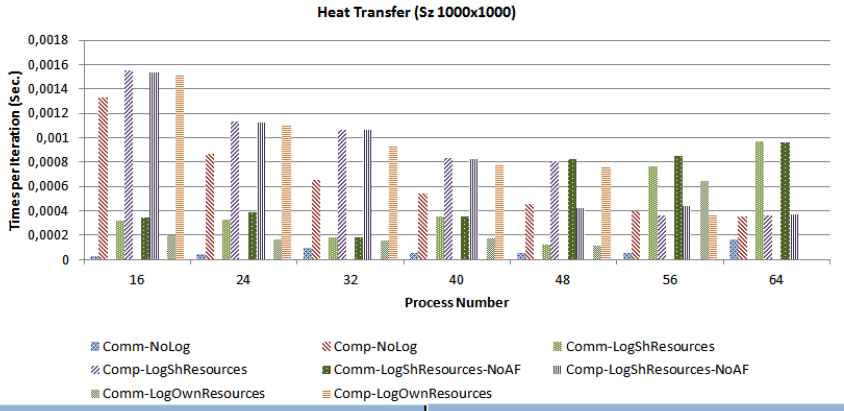
E5430 running at 2.66 GHz. Each node has 16 GB of main memory and a dual embedded Broadcom® NetXtreme IITM 5708 Gigabit Ethernet. RADIC features have been integrated into Open MPI 1.7.

Most of the overhead added by a logging protocol affects communications. In order to lower the impact of a message logging technique we can assign more work per process which allows us to hide the overheads in communications (Figure 2c). However, if there are no available computational resources for the fault tolerance tasks, the overheads in computations could become relevant. Moreover, if we are executing a parallel application where memory consumption per process is high, there will be no room for the FT mechanisms.

When executing a parallel application with FT support is desirable to store checkpoints and message logs in main memory avoiding the file system, thus allowing FT mechanisms to execute faster. Also, if we consider an event triggered checkpoint mechanism where checkpoints take place when a message-log-buffer in memory is full and we save memory by executing less application processes per node we can use a bigger message-log-buffer, thus the checkpoint interval could be bigger.

Our testbed here is composed by two SPMD applications: a Heat Transfer application and a Laplace Solver. Both applications allow overlapping between the computation steps and the communication steps as was shown in Figure 2 and are written using non-blocking communications. The computation and communication times per iteration showed in bars in Figure 4 and Figure 5 are obtained by executing a few iterations of the parallel applications observing all processes and then selecting the slowest one for each number of processes. The execution times have been obtained using 10000 iterations.

In these experiments we have only considered the overlapped times (communication and computations) because they represent the higher portion of both applications. We have discarded the delays caused by desynchronization and the computation time spent in computing the edges of each sub-matrix before sending it to the neighbors.



Overheads With Barriers				Overheads Without Barriers			
# Process	Ovh-ShResources-NoAF	Ovh-ShResources	Ovh-OwnResources	# Process	Ovh-ShResources-NoAF	Ovh-ShResources	Ovh-OwnResources
16	55,25	51,75	46,55	16	35,85	35,00	36,57
24	69,42	61,96	56,94	24	56,07	48,34	45,47
32	44,06	41,24	28,94	32	69,48	59,40	53,98
40	116,03	86,14	71,52	40	85,41	78,13	66,55
48	123,38	96,99	75,58	48	93,09	78,62	65,14
56	129,86	99,10	75,40	56	103,10	91,09	65,97
64	54,58	43,28		64	180,31	145,58	

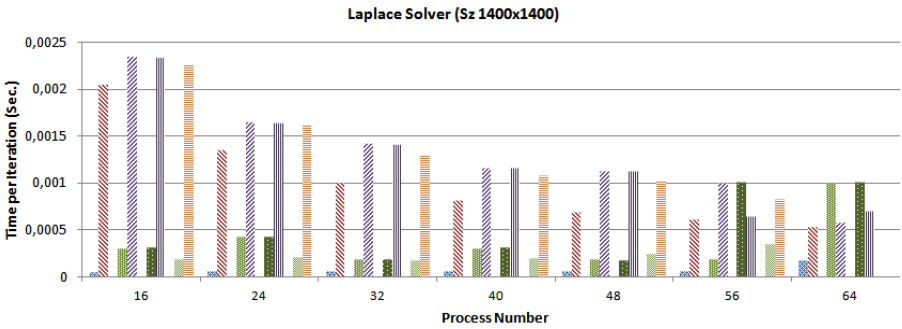
Fig. 4. *Characterization results and Overhead Analysis of the Heat Transfer Application.*

For both applications we have measure communication and computation times with the following options:

1. Without using message logging (Comm-NoLog and Comp-NoLog).
2. With message logging using all available cores in each node and giving affinity to each logger thread in order to ensure an equally distributed overhead in computation among all application processes (Comm-LogShResources and Comp-LogShResources).
3. With message logging using all available cores in each node without giving affinity to each logger thread (Comm-LogShResources-NoAF and Comp-LogShResources-NoAF).
4. With message logging saving one core per node and assigning all logger threads to the core not used by application processes (Comm-LogOwnResources and Comp-LogOwnResources).

With the purpose of measuring the communication and computation times of each application, we have inserted a barrier (MPI_Barrier) that allows us to properly measure them. The tables of Figure 4 and Figure 5 show the overhead in percentage introduced by each message logging approach with the barriers and also without them. The executions without barriers are faster than the execution with barriers and we present both overheads in order to

prove that the measures taken are consistent when removing them and executing the original versions.



Overheads With Barriers			
# Process	Ovh-ShResources-NoAF	Ovh-ShResources	Ovh-OwnResources
16	39,91	38,79	38,25
24	55,79	50,99	50,02
32	56,33	37,68	25,74
40	75,11	65,36	51,97
48	89,70	79,68	65,60
56	101,65	82,75	61,83
64	87,01	74,10	

Overheads Without Barriers			
# Process	Ovh-ShResources-NoAF	Ovh-ShResources	Ovh-OwnResources
16	24,24	24,41	26,25
24	40,31	39,60	35,52
32	50,28	47,66	42,94
40	56,07	57,63	44,82
48	80,68	69,55	58,41
56	83,45	73,34	52,41
64	179,03	138,30	

Fig. 5. *Characterization results and Overhead Analysis of the Laplace Solver.*

In Figure 4 we can observe how the computation times when using the version with own resources are lower. Even when the application becomes communication bound (56 processes) the logging version with own resources behaves better than the other versions. We do not show results of the version with own resources with 64 processes because our test environment has 64 cores, and we have to save 8 cores (1 per node) for the logger threads.
The tables of Figure 4 reflect what we have observed when characterizing the application, using message logging with own resources for the logger threads introduces less overhead in almost all cases (except with 16 cores without barriers). At best, we have reduced 25% overhead when comparing the own resources version with the version with shared resources and affinity. We can also observe that when increasing the number of processes without increasing the problem size, the overhead added becomes bigger.
Figure 5 shows the execution of the Laplace Solver. As was in the previous experiment, here we can observe how the computation times are lower when using the version with own resources.
The tables of Figure 5 reflect again what we have observed when characterizing the application, using message logging with own resources for

the logger threads introduces less overhead in almost all cases (except with 16 cores without barriers).At best, we have reduced 20% overhead when comparing the own resources version with the version with shared resources and affinity.
As we have observed, in both applications the computation time of the versions with FT with own resources is lower than the versions with shared resources, but is not equal to the version without message logging. This is because when logging is activated and a process call MPI_Irecv, this process should save the request, re-send the message to its logger thread and free the request when the message was totally received, thus there is a slight increase in computation.

5. Conclusions

The main contribution of this paper consists on analyzing possible configurations of the pessimistic receiver-based logging approach in order to find the most suitable according to application behavior. This is done by characterizing the parallel application (or a small kernel of it) obtaining the computation and communication times and the disturbance caused by the logging approaches. Our initial results have demonstrated that saving resources for the FT tasks reduces overheads and also allows us to save memory for a message log buffer. In our experimental validation we have obtained 25% overhead reduction at best.
Future work will extend the analysis made in this paper to a bigger set of applications. We will focus on obtaining traces of parallel applications and use them to find the FT configuration that will be more suitable to them. We will also analyze the relationship between message sizes and logging overheads, in order to determine the number of resources that should be save for FT tasks, because with bigger message sizes delays could increase.

References

1. Nielsen, I., Janssen, C.L. (2008). Multicore challenges and benefits for High Performance Scientific Computing. Sci. Program. 277-285.
2. Bouteiller, A., Herault, T., Bosilca, G., Dongarra, J.J. (2011). Correlated Set Coordination in Fault Tolerant Message Logging Protocols. 51-64.
3. Bouteiller, A., Ropars, T., Bosilca, G., Morin, C., Dongarra, J. (2009). Reasons for a Pessimistic or Optimistic Message Logging Protocol in MPI Uncoordinated Failure Recovery. Cluster Computing and Workshops. 229-236.
4. Bouteiller, A., Bosilca, G., Dongarra, J. (2010). Redesigning the Message Logging Model for High Performance. Concurr. Comput.: Pract. Exper. 2196-2211

5. Fialho, L., Rexachs, D., Luque, E. (2012). What is missing in current checkpoint interval models? IEEE 32nd International Conference on Distributed Computing Systems. 322-332.
6. Lemarinier, P., Bouteiller, A., Herault, T., Krawezik, G., Cappello, F. (2012). Improved Message Logging versus Improved Coordinated Checkpointing for Fault Tolerant MPI. IEEE International Conference on Cluster Computing. 115-124.
7. Meyer, H., Rexachs, D., Luque, E. (2012). RADIC: A Fault Tolerant Middleware with Automatic Management of Spare Nodes. International Conference on Parallel and Distributed Processing Techniques and Applications. 17-23.
8. Santos, G., Fialho, L., Rexachs, D., Luque, E. (2009). Increasing the Availability provided by RADIC with low overhead. IEEE International Conference on Cluster Computing and Workshops. 1-8.
9. Hursey, J., Squyres, J., Dontje, T. (2011). Locality-aware Parallel Process Mapping for Multi-core HPC Systems. IEEE International Conference on Cluster Computing. 527-531.

Scalability and Energy Consumption Analysis in Parallel Solutions on Multicore Clusters and GPU for a High Computational Demand Problem

ERICA MONTES DE OCA[1], LAURA DE GIUSTI[1], ARMANDO DE GIUSTI[1,2], MARCELO NAIOUF[1].

[1] Instituto de Investigación en Informática LIDI (III-LIDI)
Facultad de Informática – Universidad Nacional de La Plata

[2] CONICET – Consejo Nacional de
Investigaciones Científicas y Técnicas
{emontesdeoca,ldgiusti,degiusti,mnaiouf}@lidi.info.unlp.edu.ar

Abstract. *This paper describes a study carried out on scalability and energy consumption when using a multicore cluster and a GPU board with 384 cores for solving the N-body problem. A parallel solution on shared memory was implemented for CPU using Pthreads, a shared memory solution was developed for GPU using CUDA, and a distributed memory solution was implemented on a CPU using MPI. The results obtained are presented and discussed, showing that, for this problem, the use of the GPU not only speeds up computation, but also reduces energy consumption.*

Key words: *multicore, multicore cluster, GPU, N-body, scalability, green computing, energy consumption.*

1. Introduction

Technology has allowed improving the quality of life all over the globe. Recent processor advances have been used to speed up the solution process for many real-life problems. The arrival of multicores not only has allowed reducing computation times for various applications [1], but it has also decreased processor energy consumption because, despite the fact that multicores are formed by several processors, these are much more simple than before.

Multicores can be grouped into clusters combining the shared memory resources contributed by each of the cores with the distributed memory available for intra-multicore communication [2], thus resulting in a hybrid scheme. On the other hand, a new specific-purpose platform has gained ground in recent decades as an alternative for High Performance Computing – the GPU [3] [4] [5] [6].

However, this computation speedup involves a significant increase in energy consumption, which has become an important aspect both for hardware manufacture as well as software implementations.
From today's perspective, it is our responsibility to provide not only a technological breakthrough but a responsible use of the same [7]. Be it through hardware or software, unnecessary expenses in energy consumption should be reduced to a minimum. In this sense, we present a comparison of possible solutions to the problem of gravitational attraction between celestial bodies using a multicore cluster and GPU.
This paper is organized as follows: Section 2 introduces the concept of Green Computing; Section 3 briefly discusses the N-body problem; Section 4 presents the experimental results obtained; and Section 5 presents the conclusions and future work.

2. Green Computing

Green Computing includes the efficient use of resources. Its goals are reducing the use of hazardous materials, maximizing energy efficiency during the production life cycle, and promoting recycling and biodegradability of products and factory waste [8].
Oftentimes, consumers do not take into account the ecological impact when buying their computers, they only look at processing speed and price. However, greater processing speeds require more energy power, which brings along the problem of heat dissipation; and this in turn results in additional power being consumed to keep the processor at a normal operation temperature. Hardware designers have already planned several strategies to help reduce energy consumption that span the entire cycle, from manufacture to recycling [9].
The move from monoprocessors to multicore processors represents a significant progress, since the latter are usually simpler and, therefore, make a more efficient use of energy power. Large companies (Intel and AMD) are aware of this need for an efficient use of resources during production [10] [11].
In recent years, power efficiency has become one of the most relevant factors in application development. There are current research projects underway in the field of High Performance Computing (HPC) aimed not only at reducing the amount of energy consumed, but also at producing an energy management system that, given an HPC application and platform, offers execution alternatives based on: energy consumption, maximum power (capacity of the electrical infrastructure and the cooling system), and performance [12]. Energy efficiency is an application development-limiting factor in HPC, since the energy needed to process large volumes of data has become an issue that requires more attention as technology advances.
In addition to energy efficiency, system scalability and power costs should also be taken into account. If a system consumes large amounts of energy, it is less scalable and this in turn makes it less useful in the long term. Therefore, scalability has to be considered not only in relation to the problem and the architecture at hand, but also from the perspective of the total

consumption from all applications, so that, on the one hand, it is not greater than the available power supplied and, on the other, it is aligned with available economic resources.

3. Study Case: N-body Problem

N-body is a classical scientific computation problem, and it has been widely studied due to its adaptability to different real-world applications [13]. This paper focuses on the application of the gravitational attraction force, based on Newton's Law of Attraction, which states: "The gravitational force between two bodies is proportional to their masses and inversely proportional to the square of their distances" [14].
The following information is required for modeling the problem: mass, speed, position, and initial attraction force for each body. Equation (0) is the main calculation for all processing, and is based on the magnitude force of each body [15].

$$F = (G \times m_i \times m_j) / r^2 \quad \textbf{(0)}$$

where r = distance, G = gravitational force (6.67×10^{11})

The sequential algorithm in which are based the parallel solutions is called *all pair* or *direct simulation.* All of the bodies in the problem space calculate their gravitational attraction force with the rest. Therefore, problem complexity is $O(N^2)$ [16]. To optimize access to cache memory and reduce run time, data are processed in blocks, using the optimal block size for the architecture being tested. The parallel algorithms that are implemented for shared memory (in Pthreads) and distributed memory (in MPI) also use block processing to reduce run times by decreasing cache failures.
In the case of the shared memory algorithm, there is a main thread responsible for initializing the data and then distributing cache-optimal-sized data blocks to the T-1 threads that are part of the simulation. To solve the problem, T-1 user-specified threads are created, since the main thread also has a workload. Once all threads have their data to process, they start calculating the gravitational attraction force for each body for their respective data sets. These calculations will be repeated for each step of the simulation, and each thread will have to wait at a synchronization barrier before starting a new step so that all other threads have the updated data from the previous step.
For the solution developed with MPI, the implementation of the algorithm is similar to the previous one, with the difference that the working units are processes rather than threads, and that, once a process finishes its calculations for a given step, it must communicate the results obtained to the other processes before starting a new simulation step.
In the case of the algorithm developed for GPU in CUDA, data are initialized at the CPU and then communicated through PCI-E to the GPU. Once the

GPU has the data copied to its global memory, each participating thread copies the corresponding data to its shared memory to calculate the gravitational attraction force; thus, access to memory is optimized. Since shared memory is reduced in size [17] [18] [4] [19], the data from the global memory are copied in blocks whose size is the same as that of the thread block. This data transfer between memories is carried out as many times as necessary until all required data have been copied to perform the calculations. After processing ends, the GPU sends the results to the CPU through PCI-E.
The remaining equations used for the solutions are described in greater detail in [20], accompanied by a more thorough discussion of the parallel solutions that were implemented.

4. Experimental Results

The test environment is formed by the following architectures:

- A multicore cluster with Quad Core Intel i5-2300 2.8-GHz processors, 6MB cache, and 64-bit Debian operating system.
- GPU Geforce TX 560TI with 384 processors and a maximum number of 768 threads and 1 GB of RAM.

The number of bodies used for every simulation varies between 65535, 128000 and 256000 bodies for two simulation steps. The data collected correspond to the average of several runs.
For the distributed memory solution, runs were carried out using one machine per process (and therefore, inter-process communication was carried out through a network). Also, a test was made with all processes running in a single machine.
Table 1 shows execution times in seconds for two simulation steps of the problem for the CPU solutions (both shared and distributed memory) using 2 and 4 processors. Table 2 shows the execution times (in seconds) obtained for the GPU solution, using CUDA, with thread block sizes of 256, which is the optimal size for the architecture used, and two simulation steps.

Table 1. Execution times in seconds for MPI algorithms (both execution modes) and Pthreads on CPU. P = number of processes, T = number of threads.

No. of bodies in simulation	Pthread (T = 2)	MPI (P = 2)	MPI one machine (P = 2)	Pthread (T = 4)	MPI (P = 4)	MPI one machine (P = 4)
65535	101.68	96.63	92.11	53.87	50.56	53.91
128000	397.39	352.39	352.93	213.94	175.66	182.08
256000	1572.81	1417.61	1427.42	810.17	717.80	728.45

Table 2. Execution times in seconds for the CUDA algorithm on GPU. T = number of threads per block.

No. of bodies in simulation	CUDA (T = 256)
65535	1.04
128000	3.97
256000	15.75

To measure energy consumption, an 8-bit resolution digital oscilloscope was used. The oscilloscope had two inputs, one to capture voltage information and the other one for electric current. The latter is provided through a transducer clamp that can be adjusted to the following sensitivity values: 1A/100mV, 1A/10mV, and 1A/1mV.

Electrical voltage was measured directly from the power line to which the multicore cluster is connected. The information collected by the digital oscilloscope is sent to other machine to be analyzed. The information on electrical current is obtained from the input wire to the energy sources of the architecture used.

The digital oscilloscope uses buffers that store 10,240 samples of calculated data (10 KB). Each buffer represents approximately 40 milliseconds, which is equivalent to a sampling interval of 40 ms/10 KB = 3.9 μs.

Table 3 shows the results measured in total Joules consumed by the application with the different configurations and solutions with the CPU architecture, for two simulation steps, for problems with 65535, 128000 and 256000 bodies.

Table 3. Total Joules consumed by MPI and Pthread algorithms on CPU. P = number of processes, T = number of threads.

No. of bodies in simulation	Pthread (T = 2)	MPI (P = 2)	MPI one machine (P = 2)	Pthread (T = 4)	MPI (P = 4)	MPI one machine (P = 4)
65535	740.74	660.05	699.72	497.14	970.46	428.23
128000	2858.60	2415.74	2840.93	1953.98	3237.15	1798.35
256000	11290.59	9235.81	10040.86	7971.86	13123.58	6394.27

Table 4 presents the total consumption in Joules for the GPU algorithm with 65535, 128000 and 256000 bodies in two simulation steps. Power consumption measurements for the GPU architecture were carried out following the same procedure as for the CPU architecture. Consumption information was obtained by measuring power on the input wire to the source of the machine containing the GPU board.

Table 4. Total Joules consumed by the CUDA algorithm on GPU. T = number of threads per block.

No. of bodies in simulation	CUDA (T = 256)
65535	13.67
128000	60.94
256000	239.12

Figures 1 and 2 show the total consumption measured for the algorithms on CPU using shared and distributed memory, for the different problem sizes (65535, 128000 and 256000) and two simulation steps, with a configuration of 2 and 4 processors, respectively. Figure 3 presents the total consumption for running the application on GPU using 256-thread blocks for 65535, 128000 and 256000 bodies and two simulation steps.

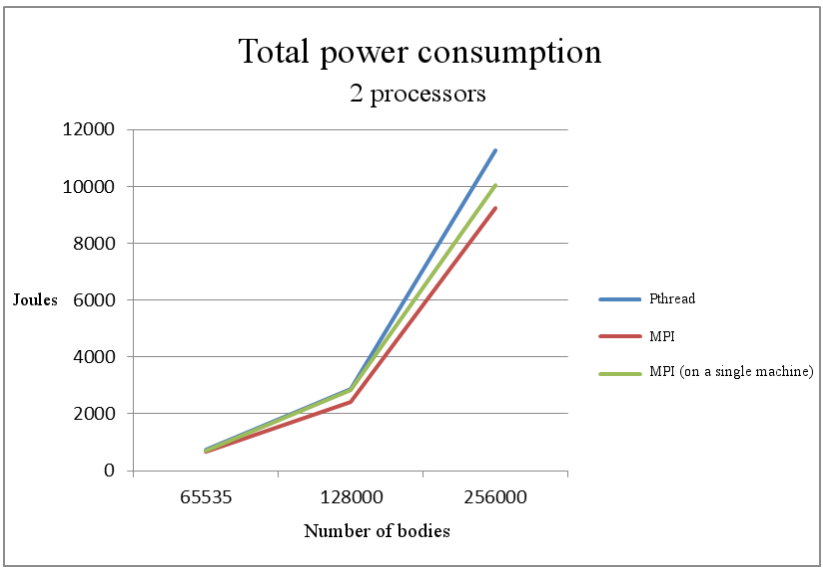


Fig. 1. *Total consumption in Joules for the MPI and Pthreads algorithms with two processors for 65535, 128000 and 256000 bodies and two simulation steps.*

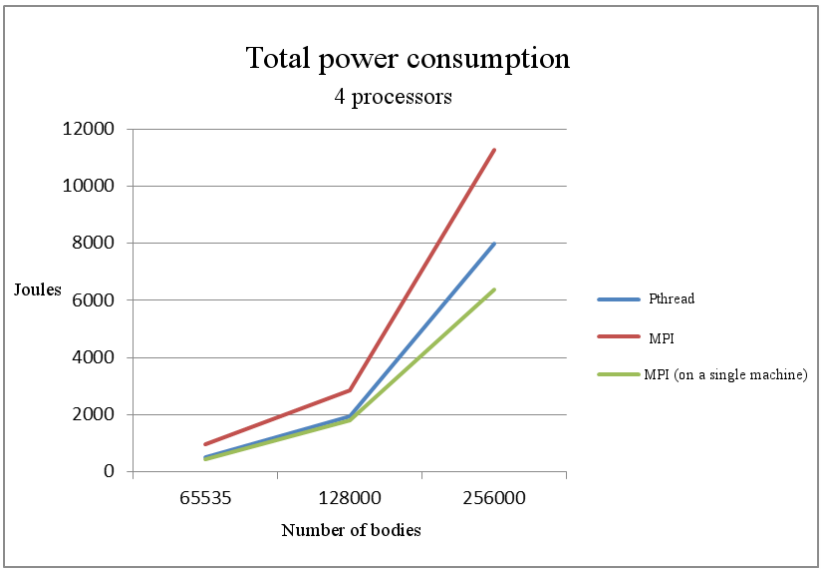


Fig. 2. *Total consumption in Joules for the MPI and Pthreads algorithms with four processors for 65535, 128000 and 256000 bodies and two simulation steps.*

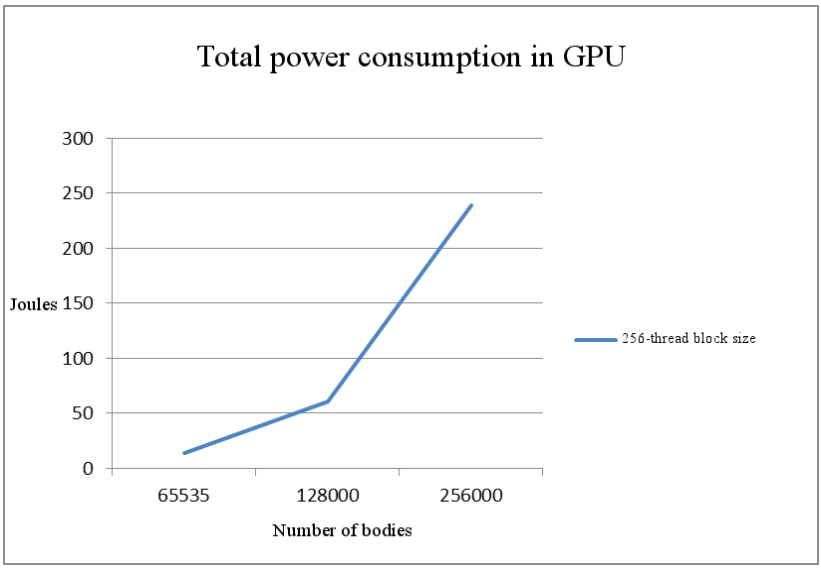


Fig. 3. *Total consumption in Joules for the CUDA algorithm on GPU with 256-thread block size, for 65535, 128000 and 256000 bodies and two simulation steps.*

The results described above were obtained using the test environment presented. As it can be seen, the difference in execution time for the parallel solutions running on CPU is not significant. In the case of the algorithm developed on MPI, execution times are very similar if all processes are run on the same machine or if each process is run on its own machine. As regards GPU, it can be seen that execution time is significantly lower than that of the parallel solutions developed for the CPU architecture for all problem sizes.
From the point of view of total energy consumed, it can be seen that the parallel solution on GPU is the one with the lowest consumption, which is very remarkable in light of the great computation speedup obtained. While the highest consumption of the GPU solution corresponds to the largest problem for the tests carried out, the total consumption in the case of the parallel versions on Pthreads and MPI on a single machine is double that consumption for the smallest of the problems tested.
In the case of the parallel solutions on CPU, better results were achieved in the distributed memory version in wich execution is performed on a single machine compared with the solution for shared memory. However, the reduction of total energy consumption is not as significant between the two solutions compared with GPU solution. If every process is run on its own machine, consumption increases as the number of processors increases, since total power consumption is given by the sum of the consumption of each individual machine.
The scalability of a parallel system reflects its capability to effectively add new processing resources. As it can be seen on the execution times obtained, an increase in the number of processors reduces application processing time for the different problem sizes. If we also consider scalability from the perspective of total energy consumption, in the cases of Pthreads and MPI on a single machine, consumption is reduced as the architecture grows because execution time is reduced. In the case of GPU, even though more total power is consumed to solve larger problems, consumption is still well below that required by the parallel solutions on CPU.

5. Conclusions and Future Work

The purpose of parallelism is to speed up the data calculations carried out by the applications, since sometimes the execution times obtained with sequential algorithms are non-acceptable for the response times required. In particular, in this paper we present the N-body high computational demand problem as study case. If this problem is solved sequentially, processing times are very high [20]; therefore, parallel solutions have been implemented using two different architectures – CPU and GPU. It has been successfully proven that a significant speedup can be achieved using the GPU architecture for the study case.
On the other hand, in recent years energy consumption and the ability to minimize it has become a limiting factor in the use of technological equipment for solving real-world problems. Be it with a view to future

system scalability or in order to reduce expenses arising from energy consumption, the significance of the efficient use of energy by both hardware and software is rapidly growing.

The results presented in this article show a decrease in energy consumption when using GPU due to the high computational speedup it offers versus parallel versions running on CPUs. In the case of parallel solutions implemented for CPUs, the best results were obtained with the distributed memory version running on a single machine, versus the shared memory solution. However, the reduction in total energy consumed by the application is not as significant for these two solutions when compared to the GPU solution.

Our future work will be focused on analyzing scalability and consumption when using a GPU cluster with the MPI-CUDA hybrid model. Also, other types of applications will be studied.

References

1. Silva J. M. N., Drummond L., y Boeres C. (2010). "On Modelling Multicore Clusters", 22nd International Symposium on Computer Architecture and High Performance Computing Workshops.
2. Tinetti F. G., Wolfmann G. (2009). "Parallelization Analysis on Clusters of Multicore Nodes Using Shared and Distributed Memory Parallel Computing Models", World Congress on Computer Science and Information Engineering.
3. Kirk D. B., Hwu Wen-mei W. (2010). "Programming Massively Parallel Processors: A Hands-on Approach", Morgan Kaufmann.
4. Piccoli M. F. (2011). "Computación de Alto Desempeño en GPU", XV Escuela Internacional de Informática del XVII Congreso Argentino de Ciencia de la Computación, Editorial de la Universidad Nacional de La Plata.
5. Nvidia Corporation (2003). "GPU gems", Pearson Education.
6. Song J. P. (2010). "An Analysis of GPU Parallel Computing", 2009 DoD Hogh Performance Computing Modernization Program Users Group Conference, publicado en IEEE.
7. Francis K., Richardson P. (2008). Green Maturity Model for Virtualization, The Architecture Journal, pp. 9-15.
8. Schneider Electric (2008). "Go Green, Save Green. The Benefits of Eco-Friendly Computing".
9. Verma, S.S. (2007). "GREEN COMPUTING". Departamento de Física, SLIET.
10. Nicolaisen N. (2010). "Green Computing with Intel® AtomTM Processor-Based Devices", http://software.intel.com/en-us/articles/green-computing-with-intel-tomprocessor-based-devices/.
11. amd.com/public (2011). "Meeting the challenges of the future with innovative solutions for public sector IT needs".

12. Balladini J., Uribe F., Suppi, R., Rexachs, D., Luque, E. (2011). “Factores influyentes en el consumo energético de los Sistemas de Cómputo de Altas Prestaciones basado en CPUs y GPUs”. Facultad de Informática, Universidad Nacional del Comahue, Argentina, y Departamento de Arquitectura de Computadores y Sistemas Operativos, Universidad Autónoma de Barcelona, España. XVII Congreso Argentino de Ciencias de la Computación.
13. Chinchilla, F., Gamblin, T., Sommervoll, M., Prins, J. F. (2004). “Parallel N-Body Simulation using GPUs”, Department of Computer Science, University of North Carolina at Chapel Hill, http://gamma.cs.unc.edu/GPGP, Technical Report TR04-032.
14. Bruzzone S. (2011). “LFN10, LFN10-OMP y el Método de Leapfrog en el Problema de N Cuerpos”, Instituto de Física, Departamento de Astronomía, Universidad de la República y Observatorio Astronómico los Molinos, Uruguay.
15. Gregory R., A. (2000).
16. Nvidia (2012). “CUDA C BEST PRACTICES GUIDE”.
17. Perez C., Piccoli M. F. (2010). “Estimación de los parámetros de rendimiento de una GPU”, Mecánica Computacional Vol. XXIX, pp. 3155-3167.
18. Montes de Oca, E., De Giusti, L., De Giusti, A., Naiouf, M. (2012). “Comparación del uso de GPU y Cluster de multicore en problemas de alta demanda computacional”, XVIII Congreso Argentino de Ciencias de la Computación, CACIC 2012, pp. 267-275.

Parallel implementation of a Cellular Automata in a hybrid CPU/GPU environment

EMMANUEL N. MILLÁN [1,2,3], PAULA CECILIA MARTINEZ[2], VERÓNICA GIL COSTA[4], MARIA FABIANA PICCOLI[4], MARCELA PRINTISTA[4], CARLOS BEDERIAN[5], CARLOS GARCIA GARINO[2], EDUARDO M. BRINGA[1,3]

[1] CONICET, Mendoza

[2] ITIC, Universidad Nacional de Cuyo

[3] Instituto de Ciencias Básicas, Universidad Nacional de Cuyo, Mendoza

[4] Universidad Nacional San Luis, San Luis

[4] Instituto de Física Enrique Gaviola, CONICET

emmanueln@gmail.com, ebringa@yahoo.com

http://sites.google.com/site/simafweb/

Abstract. *Cellular Automata (CA) simulations can be used to model multiple systems, in fields like biology, physics and mathematics. In this work, a possible framework to execute a popular CA in hybrid CPU and GPUs (Graphics Processing Units) environments is presented. The inherently parallel nature of CA and the parallelism offered by GPUs makes their combination attractive. Benchmarks are conducted in several hardware scenarios. The use of MPI /OMP is explored for CPUs, together with the use of MPI in GPU clusters. Speed-ups up to 20 x are found when comparing GPU implementations to the serial CPU version of the code.*

Keywords: *General purpose GPU, Cellular Automata, multi-GPU*

1. Introduction

Multicore CPUs have become widely available in recent years. As an alternative, Graphics Processing Units (GPUs) also support many cores which run in parallel, and their peak performance outperforms CPUs in the same range. Additionally, the computational power of these technologies can be increased by combining them into an inter-connected cluster of processors, making it possible to apply parallelism using multi-cores on different levels. The use of GPUs in scientific research has grown considerably since NVIDIA released CUDA [1]. Currently, 43 supercomputers from the Top 500 List (June 2013, www.top500.org) use GPUs from NVIDIA or AMD. Currently the most commonly used technologies are CUDA [1] and OpenCL [2]. Recently Intel entered the accelerator's market, with the Xeon Phi [3]

x86 accelerator, which is present in 12 supercomputers from the Top 500 List (June 2013, www.top500.org).
This work evaluates the trade-off in the collaboration between CPUs and GPUs, for cellular automata simulations of the Game of Life [4]. A cellular automaton (CA) [5] is a simple model represented in a grid, where the communication between the grid points or cells is limited to a pre-determined local neighbourhood. Each cell can have a number of finite states which change over time, depending on the state of its neighbours and its state at a given time [6]. Even though such a model is simple, it can be used to generate complex behaviour. CA have been used to implement diverse systems in different fields of science: biological systems [7], kinetics of molecular systems [8], and clustering of galaxies [9]. CA have also been implemented in GPU architectures in biology to simulate a simple heart model [10], infectious disease propagation [11], and simulations of laser dynamics [12].
The Game of Life has already been extensively studied [13][14][15]. In this paper, the model is used as a starting point to developed future CA simulations in hybrid (CPU+GPU) environments, as it has already been achieved for Lattice Boltzmann Gas simulations [16], the Cahn-Hilliard equation [17][18] and reaction-diffusion systems [19][20]. A relatively small code was developed to run in multiple parallel environments, and the source code is available in the website of the authors (https://sites.google.com/site/simafweb/proyectos/ca_gpu). Five implementations of the Game of Life code with C were developed: one serial implementation which executes in a single CPU core, and four parallel implementations, including: shared memory with OpenMP, MPI for a CPU cluster, single GPU, and Multi-GPU plus MPI for a CPU-GPU cluster. The paper is organized as follows. Section 2 describes the Game of Life in general; Section 3 contains details of the code implementation; Section 4 includes code performance; and conclusions are presented in Section 5, including possible future code improvements.

2. Description of the Problem

The Game of Life, through a set of simple rules, can simulate complex behaviour. To perform the simulations of this work, a 2 dimensional (2D) grid with periodic boundaries is used. The grid cells can be “alive” or “dead”, and a Moore neighbourhood (8 neighbours) [6] is considered for each cell. According to the cell state at time t and the state of its 8 neighbours, the new state for the cell at time $t+1$ is computed. The update is done simultaneously for all cells , which means that their change will not affect the state of neighbour cells at $t+1$ time. Time $t+1$ is often referred to as the “next generation”. The following rules define the state of a cell in the Game of Life [4]:

– Any living cell, with less than two living neighbours will die in $t+1$.
– Any living cell, with two or three living neighbours will live in $t+1$.

– Any living cell, with more than three living neighbours will die in *t+1*.
– Any dead cell, with exactly three living neighbours will be alive in *t+1*.

3. Implementation

3.1 Serial Code Implementation

The serial implementation which executes the entire simulation in one CPU processor was used to verify the correct functioning of other implementations. The serial code is quite simple. It begins initializing the main 2D grid, size *NxN*, with zeros and ones randomly placed. *N* has to be an even integer. A secondary *NxN* 2D grid is used to store the states of each cell for the next iteration. Any random number generator can be easily used, but the standard *rand(seed)* function was used in all codes. For testing purposes, the seed used to generate the random numbers is always the same. The code runs up to a maximum number of iterations defined by the user. Within each iteration, every cell and their neighbourhoods are monitored, the four rules for the CA are applied, and the next set of states are stored into the secondary array. Once all cells are analyzed, the secondary grid is copied into the main 2D grid, concluding a given iteration. Every *m* iterations, with *m* a pre-set integer number, the state of the main 2D grid is written to an output file. Since this is a simulation with periodic boundaries, care must be taken when the values of the neighbours of a cell have to be monitored. When the simulation ends, the program prints on the screen the amount of RAM memory used and the total execution time, separating by: filling time of the 2D grid, evolution time, and write-up time of output files.

3.2 OpenMP Implementation

The parallel shared memory implementation of the code was developed with OpenMP [21]. It uses two compiler directives (*pragma*) to execute in parallel the for loops in the serial implementation of the code. These two loops are the loop that iterates over the 2D grid applying the game's rules, and the loop which copies the secondary grid to the main grid. Each *pragma* was configured to use dynamic or static scheduling, with the work assigned to each thread defined by the type of scheduling method. When using static scheduling, each thread is assigned a number of for loop iterations before the execution of the loop begins. When using dynamic scheduling, each thread is assigned some portion of the total (chunk) of iterations, and when a thread finishes its allotted assignment returns to the scheduler for more iterations to process.

3.3 MPI Implementation

Using a cluster of computers to run the case of message passing with MPI significantly increases the complexity of the code, especially when dealing with periodic boundary conditions. To handle part of the communications, the strategy by Newman [22] was used. The initial grid is loaded as in the serial case, then this grid is divided into equal blocks, with block size given by the full grid size and the number of processors in a square topology, e.g. running in 4 processors will give a 2x2 topology. Then, each block is sent to a different processor.
When an iteration begins, each process exchanges its grid boundaries with its neighbouring processes. From the code by Newman [22], the functions *exchange()*, *pack()* and *unpack()* were used as basis for sending the boundaries and corners of each block from each process to its neighbours. After performing the boundary exchange, the number of living neighbours for each cell in each process is calculated, the block corresponding to that process is updated, and a new iteration begins, sending updated boundaries between neighbouring processes. When it is necessary to copy the state of the grid to a file, the block from each process is brought to the master process to be part of a single grid, which is then written. Because of the simple chosen communication scheme, the current code cannot deal with odd number of processes.

3.4 GPU Implementation

The implementation of the code for a single GPU uses NVIDIA CUDA and takes the CPU serial code as basis. Two kernels (C code functions that run on the GPU) were developed: one kernel controls the state of the neighbours and modifies the secondary grid for the next iteration; the other kernel is responsible for updating the main grid with the data obtained by the first kernel (it copies the secondary grid into the main grid). There has to be a blocking step to ensure that all the threads have finished executing their instructions within the same kernel for a given iteration, making two separate kernels necessary. In order to ensure that instructions were properly executed, it was necessary to place a barrier outside the first kernel, in the CPU host code. Is worth mentioning that there is extra communication time: to copy the input grid generated in the CPU to the GPU, and to copy the system grid to the main memory each time the grid is written to a file, because it is necessary to copy from the GPU global memory to the CPU main memory.

3.5 Multi-GPU Implementation

The parallel code with MPI and CUDA is similar to the parallel MPI implementation explained in section 3.3 and the GPU implementation from section 3.4. MPI sends a block of the principal grid to each node for processing, then each of these nodes run the simulation on a GPU. Two kernels were added: the first kernel prepares data to be sent from a GPU to

neighbouring CPU nodes, and the second kernel is responsible for loading data received from neighbouring processors into the GPU. These kernels replace the functions *pack()* and *unpack()* of the CPU MPI code. This multi-GPU code also outputs the time to copy data between the CPU and GPU at each time step, and it was called "Halo time" as in [23]. This only takes into account data transfer between CPU RAM memory and GPU global memory, regardless of MPI communication time between nodes.

4. Benchmarks

4.1 Infrastructure

Simulations were executed in three different environments.

- Workstation Phenom: 2.8 GHz AMD Phenom II 1055t 6 cores with 12 GB DDR3 of RAM memory. NVIDIA Tesla c2050 GPU, with 448 CUDA cores working at 1.15 GHz, and 3 GB memory. Slackware Linux 13.37 64 bit operating system with kernel 2.6.38.7, OpenMPI 1.4.2, Cuda 5.0 and gcc 4.5.3.
- Workstation FX-4100: 3.4 GHz AMD FX-4100 x4 with 8 GB of DDR3 RAM memory. NVIDIA GeForce 630 with 48 CUDA cores working at 810 MHz, and 1 GB of memory. Slackware Linux 14 64 bit operating system with kernel 3.2.29, OpenMPI 1.7 beta, Cuda 4.2 and gcc 4.5.2.
- Cluster Mendieta at the Universidad Nacional de Cordoba: 8 nodes with two 2.7 GHz Intel Xeon E5-2680 CPU with 32 GB of memory, 12 NVIDIA Tesla M2090 GPUs with 6 GB GDDR5 (177 GBps) of memory and 512 1.3 GHz CUDA cores. The connection between nodes is at 20 Gbps InfiniBand DDR, with the switch using star topology. With Linux CentOS 6.4, MPICH 3.0.4 and Cuda 5.0.

Since the Phenom and FX-4100 workstations have only a single GPU, the code developed with MPI + GPU executes various MPI processes in the same workstation and executes the same number of independent processes in a single GPU. Because of this, the communication time between MPI nodes through the network is not evaluated for these two environments.

4.2 Simulation Results: Analysis and Discussion

Simulations were performed for different grid sizes, for the same number of iterations (1000) in all cases. The output of the last iteration performed for all parallel codes was checked against the serial version to verify the correct operation of the parallel codes. All codes were compiled with optimization *-O3*. Grids were *NxN*, with N = 500, 1000, 2000, 4000, and 6000. Four processors were used for the parallel implementations using OpenMP, MPI, and Multi-GPUs. Simulation times for different runs with the same

configuration vary only by few percent. In figure 1 it can be seen the evolution of a small region of a particular simulation. In these frames it can be seen a complex pattern drawn from the simple set of rules that are part of the Game of Life.

The performance tests performed on the Phenom workstation can be seen in tables 1 and 2. For the smallest dimension (N=500), the code in OpenMP, MPI and GPU is always faster than the serial version. For the Multi-GPU code the cost of copying the "halo" between different GPU processes is high and performance degrades. For all other simulations, parallel codes are always faster than the serial code. The OpenMP code was executed in four independent threads and, therefore, it was expected that the simulation time would decrease approximately four times. This was not the case, because the parallelization with OpenMP was done by grid rows, and the use of the cache is far from optimal. The average speedups vs. the serial version, for all sizes, was 2.7x. The implementation using MPI in 4 processors is approximately six times faster than the serial implementation, for all dimensions. The MPI code makes a division into blocks of the grid, and it is executed for the cases shown in tables 1 and 2, on a single workstation. Implementing OpenMP with blocks would likely increase performance. Speedups are given in table 5. When a single process is run on the GPU, the average speedup obtained for all grid sizes is 13 x. In the case of parallel Multi-GPU, the average speedup is 9.1 x. The copy time between GPU processes considering the smaller cases of the grid (500, 1000 and 2000) is greater than the computational time in each process. Therefore, running on a single GPU turns to be faster than using Multi-GPU for the grid sizes considered in this work. On the Phenom workstation (which has only one GPU), the Multi-GPU code executes multiple independent processes in the same GPU, at the same time. The largest case (N=6000) executes faster in multiple processes on the same GPU (Multi-GPU) than in a single process in the GPU, giving a 17 x speedup over the serial version. This improvement is due to the way in which the GPU scheduler administers the execution of the threads [24]. In addition, domain decomposition provides data locality, improving memory latencies [24] [25].

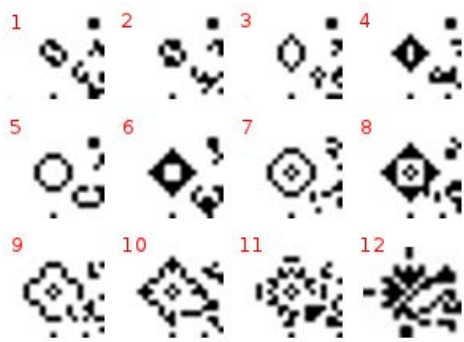


Fig. 1. *Game of life simulation, for a grid with N=500, showing the evolution of a small region of the grid, for 12 different frames.*

Tests performed on the cluster "Mendieta" were the same tests performed in the Phenom workstation. In the case of the cluster "Mendieta", two nodes with two GPUs in each node were used. All tests were performed using one GPU per node, except for the largest case with N = 6000, which was also executed for two GPUs per node.
"Mendieta" was being shared with others users at the time of testing. This might be the reason why some execution times were greater than when using the Phenom workstation, which was used exclusively for these tests. Tables 3 and 4 contain the results. There are large differences in communication time between the multi-GPU case in Mendieta and the Phenom workstation for the transfer of information between GPUs ("halo" table column). When the GPU is used in the Phenom workstation, four parallel processes are executed in the same GPU, which causes an increase in the communication time in the PCI-Express bus. Using four GPUs for the larger case shows an improvement in the simulation time, decreasing it from 26.4 to 19.3 seconds. Speedups can be seen in table 6.
Tests were also performed on the FX-4100 workstation, for a single simulation, with N=2000 and 1000 steps. Results are: Serial = 44.42 s, OpenMP= 18.63 s, MPI = 9.82 s, GPU = 17.5 s, Multi-GPU = 16.94 s. The video card installed in this machine is a low-range card and, as a result, its performance compare to CPUs is far from what could be reached with a high-range card.

Table 1. Simulation time in seconds for CPU serial, OpenMP and MPI codes, executing in the Phenom Workstation.

N	CPU Serial				OpenMP				MPI			
	fill	**evolve**	**output**	**total**	**fill**	**evolve**	**output**	**total**	**fill**	**evolve**	**output**	**total**
500	0.003	2.25	0.05	2.3	0.003	0.93	0.05	0.99	0.004	0.22	0.05	0.28
1000	0.01	10.03	0.17	10.21	0.01	3.44	0.19	3.64	0.02	1.15	0.21	1.37
2000	0.04	39.85	0.66	40.55	0.04	14.55	0.74	15.33	0.07	7.84	0.88	8.79
4000	0.16	167.15	2.64	169.95	0.16	56.75	2.95	59.86	0.25	30.23	3.47	33.95
6000	0.37	387.07	7.08	394.52	0.35	128.96	7.72	137.03	0.58	69.07	8.18	77.83

Table 2. Simulation time in seconds for GPU and Multi-GPU codes, executing in the Phenom Workstation.

N	GPU				Multi-GPU				
	fill	evolve	output	total	fill	evolve	output	halo	total
500	0.07	0.16	0.05	0.29	0.25	0.44	0.07	1.64	2.38
1000	0.09	0.56	0.2	0.86	0.26	0.77	0.24	1.65	2.93
2000	0.12	1.82	0.78	2.72	0.32	1.53	0.97	1.72	4.54
4000	0.26	7.35	3.21	10.81	0.49	4	4.06	2.99	11.53
6000	0.48	18.55	8.11	27.15	0.74	9.9	9.3	2.65	22.6

Table 3. Simulation time in seconds for CPU serial, OpenMP and MPI codes, executing in the Mendieta Cluster.

N	CPU Serial				OpenMP				MPI			
	fill	evolve	output	total	fill	evolve	output	total	fill	evolve	output	total
500	0.03	2.21	0.04	2.25	0.005	1.30	0.06	1.37	0.006	0.41	0.1	0.52
1000	0.01	8.94	0.16	9.11	0.01	5.25	1.07	6.33	0.03	1.22	0.41	1.66
2000	0.04	40.27	0.62	40.93	0.04	25.33	1.05	26.41	0.14	4.35	1.54	6.03
4000	0.16	188.49	3.78	192.42	0.27	101.05	4.23	105.54	0.59	26.66	7.01	34.26
6000	0.34	376.71	5.58	382.64	0.35	225.56	9.66	235.57	1.32	59.59	13.86	74.76

Table 4. Simulation time in seconds for GPU and Multi-GPU codes, executing in the Mendieta Cluster.

N	GPU				Multi-GPU				
	fill	evolve	output	total	fill	evolve	output	halo	total
500	3.78	0.13	1.15	5.06	3.99	0.51	0.19	0.43	5.11
1000	3.81	0.43	0.31	4.55	4.08	0.53	0.42	0.55	5.59
2000	3.9	1.64	0.8	6.34	4.06	0.75	1.73	0.8	7.34
4000	4.09	5.66	2.98	12.73	4.3	1.32	6.77	1.28	13.67
6000	4.21	14.36	13.35	31.92	4.7	3.43	15.05	3.21	26.39
6000	running in 2 GPU per node				1.48	3.29	14.31	0.18	19.27

Table 5. Speedups for all parallel codes vs the serial code in the Phenom workstation

dim	Speedup vs Serial Code			
	MPI	OMP	GPU	Multi-GPU
500	8.32	2.33	8	0.97
1000	7.44	2.8	11.92	3.49
2000	4.62	2.64	14.91	8.94
4000	5.01	2.84	15.72	14.73
6000	5.07	2.88	14.53	17.46
AVG	6.09	2.70	13.02	9.12

Table 6. Speedups for all parallel codes vs the serial code in the Mendieta cluster

dim	Speedup vs Serial Code			
	MPI	OMP	GPU	Multi-GPU
500	4.33	1.64	0.45	0.44
1000	5.50	1.44	2.00	1.63
2000	6.79	1.55	6.46	5.58
4000	5.62	1.82	15.11	14.08
6000	5.12	1.62	11.99	14.50
6000 in 2 GPUs per node				19.86
AVG	5.47	1.62	7.20	9.35

5. Conclusions and future works

The implementation of the popular Game of Life [4] was used in this paper as a possible introduction to the problem of parallel processing of a CA on CPU+GPU hybrid environments. Improvements in the execution times in the pure GPU implementation and the Multi-GPU implementation have been achieved, with speed-ups approaching 20x, when compared to a serial CPU implementation. The MPI implementation of the code gave better timing than the implementation using OpenMP, but the development time for the MPI code was higher. The OpenMP implementation did not perform as expected, and it would be necessary to carry out a detail code analysis with a tool like perf [26] to improve it. Using the Multi-GPU code provides significant speed-ups, but only for large grids (above a few million grid points).

There are several possible improvements for the codes presented here. For instance, the codes could be adapted to support square grids with N being an odd number, non-square grids, and odd number of processes. In addition, MPI and Multi-GPU codes deal with neighbours in a simple way, but MPI provides functions which could be used to perform these tasks more efficiently. There was no optimization of the codes, beyond the standard compiler optimization. There are several ways to optimize and improve performance in GPU codes: management of shared memory, monitoring and reducing the number of registers used by each kernel, etc. Starting with CUDA 4.0, GPUDirect (http://goo.gl/g7D1p) technology is available and supported by MVAPICH [27], and data can be directly transferred between GPUs without passing through the main memory system.

Once an efficient CA is implemented in multiple GPUs, one could move related problems, such as the Reaction-Diffusion Equations [19][20], the Cahn-Hilliard equation [17][18], or a CA representing some general problem of interest [28].

6. Acknowledgements

E. Millán acknowledges support from a CONICET doctoral scholarship. E.M. Bringa thanks support from a SeCTyP2011-2013 project and PICT2009-0092.

References

1. CUDA from NVIDIA: http://www.nvidia.com/cuda
2. OpenCL, The open standard for parallel programming of heterogeneous systems: http://www.khronos.org/opencl/
3. Dokulil, J., Bajrovic, E., Benkner, S., Pllana, S., Sandrieser, M. and Bachmayer, B. (2012). Efficient Hybrid Execution of C++ Applications using Intel (R) Xeon Phi (TM) Coprocessor. arXiv preprint arXiv:1211.5530.
4. Gardner, M. (1970). The fantastic combinations of John Conway's new solitaire game "life". Scientific American 223. pp 120-123.
5. Wolfram, S. (1984). Cellular Automata as models of complexity. Nature Vol. 311. pp 419- 424.
6. Ganguly, N., Sikdar, B.K., Deutsch, A., Canright, G. and Chaudhuri, P.P. (2003). A survey on Cellular Automata. Centre for High Performance Computing, Dresden University of Technology.
7. Alt, W., Deutsch, A., and Dunn, G., editors (2003). Dynamics of Cell and Tissue Motion. Birkhuser, Basel.
8. Packard, N.H. Lattice Models for Solidication and Aggregation. In First International.
9. Schonsch, B. (2002). Propagation of Fronts in Cellular Automata. Physica D, 80:433450.
10. Caux, J., Siregar, P., Hill, D. (2011). Accelerating 3D Cellular Automata Computation with GP-GPU in the Context of Integrative Biology. In: Salcido, A. (ed.) Cellular Automata - Innovative Modelling for Science and Engineering. InTech. ISBN: 978-953-307-172-5.
11. Arvizu, C., Hector M., Trueba-Espinosa, A., and Ruiz-Castilla, J. (2012). Autmata Celular Estocstico paralelizado por GPU aplicado a la simulacin de enfermedades infecciosas en grandes poblaciones. Acta Universitaria 22.6: pp. 16-22.
12. Lopez-Torres, M. R., Guisado, J. L., Jimnez-Morales, F., & Diaz-del-Rio, F. (2012). GPU-based cellular automata simulations of laser dynamics. Jornadas Sarteco.

13. Adamatzky, A., ed. (2010). Game of life cellular automata. Springer.
14. Alaniz, M., Bustos, F., Gil-Costa, V., Printista, M. (2011). Motor de simulación para modelos de Automata Celular. 2nd International Symposium on Innovation and Technology ISIT 2011. 28-30 Noviembre, Lima Peru. Noviembre 2011. ISBN: 978-612-45917-2-3. pp. 66-71.
15. Rumpf, T. (2010). Conways Game of Life accelerated with OpenCL. In Proceedings of the Eleventh International Conference on Membrane Computing (CMC11), p. 459. book-on-demand.
16. Tlke, J. (2010). Implementation of a Lattice Boltzmann kernel using the Compute Unified Device Architecture developed by nVIDIA. Computing and Visualization in Science 13, no. 1: 29-39.
17. Cahn J.W. and Hilliard J.E. (1959). Free energy of a non-uniform system III. Nucleation in a two point compressible uid, J.Chem.Phys., vol. 31, pp. 688-699.
18. Hawick, K. and Playne, D. (2008). Modelling and visualising the cahn-hilliard-cook equation. Tech. rep., Computer Science, Massey University. CSTN-049.
19. Smoller, J. (1983). Shock waves and reaction-diffusion equations. In Research supported by the US Air Force and National Science Foundation. New York and Heidelberg, Springer-Verlag (Grundlehren der Mathematischen Wissenschaften. Volume 258), 600 p. (Vol. 258).
20. Ready, A cross-platform implementation of various reaction-diffusion systems. https://code.google.com/p/reaction-diffusion/
21. The OpenMP API specification for parallel programming: http://openmp.org/
22. Newman, R.; MPI C version by Xianneng Shen. LAPLACE MPI Laplace's Equation in a Rectangle, Solved With MPI. http://people.sc.fsu.edu/ ~jburkardt/c_src/laplace mpi/laplace mpi.html
23. Lorna, S. and Bull, M. (2001). Development of mixed mode MPI/OpenMP applications. Scientific Programming 9, no. 2. pp 83-98.
24. Brown, W.M., Wang, P., Plimpton, S.J. and Tharrington, A.N. (2011). Implementing molecular dynamics on hybrid high performance computers short range forces. Computer Physics Communications 182. pp 898-911.
25. Millán, E.N., Garcıa Garino, C. and Bringa, E. (2012). Parallel execution of a parameter sweep for molecular dynamics simulations in a hybrid GPU/CPU environment. XVIII Congreso Argentino de Ciencias de la Computación 2012 (CACIC), Bahia Blanca, Buenos Aires. ISBN-978-987-1648-34-4.
26. perf: Linux profiling with performance counters. https://perf.wiki.kernel.org/index.php/Main Page
27. MVAPICH: MPI over InfiniBand, 10GigE/iWARP and RoCE. http://mvapich.cse.ohio-state.edu/
28. Tissera, P., Printista, A. M., and Errecalde, M. L. (2007). Evacuation simulations using cellular automata. Journal of Computer Science & Technology, 7.

XI

Information Technology Applied to Education Workshop

Metaplan Technique Virtualization for the Moderation of Collaborative Sessions

ALEJANDRO HÉCTOR GONZALEZ [1], CRISTINA MADOZ [1], FLORENCIA SAADI [2], DAN HUGHES [2]

Calle 50 y 120 -III-LIDI- Instituto de Investigación en Informática
La Plata Bs. As. Argentina
[1] {agonzalez, cmadoz}@lidi.info.unlp.edu.ar,
[2] {florsaadi, danlaplata}@gmail.com

Abstract. *In this paper we present the virtualization of different stages of a group moderation technique called "Metaplan". The original format of the technique is developed on-site, and the problem solving strategy by means of visualization techniques and questions is tackled. Each of the stages was analyzed and reviewed in order to propose the virtualization of the process. All work was done in the context of a graduate dissertation at the School of Computer Science of the UNLP. A prototype that allows virtualizing Metaplan to broaden the scope of the technique and facilitate team collaborative work is presented. Aspects such as time, space, style and student pace are analyzed for the virtual stages, looking for development autonomy when solving cases/problems. The results obtained are analyzed, and the division into stages for both on-site and virtual work is described. Finally, a methodology proposal for using Metaplan in virtual format is developed.*

Key words: *Metaplan, collaborative work, e-learning, groupware.*

1. Introduction

Throughout the world, and particularly in Argentina, technology changes and access to ICT (Information and Communication Technologies) are in constant development and are part of our daily activities. The use and adoption of digital technologies generates new social builds in relation to how technology is perceived and understood. Education is not exempt from this process, and students and educators find themselves in a dynamic and changing context that requires them to resort to media adoption strategies.
Education, and in particular university education, constantly produces different proposals to review classroom practices. Both at a national and international level, innovative proposals are created to incorporate different uses of the digital technology to the educational environment, which results in the creation of new teaching practices and methods and the re-signification of old ones [4][8].

Critical reviews suggest the need to reflect on: what is taught, how it is taught, and how digital technology-mediated learning is assessed [12].
There is a need for a review of teaching strategies through a process of reflection on educational practices. This process should be carried out taking into account the context in which the educational process takes place [8].
Cabero states that we are changing from a memory society to a knowledge society, where memory intelligence is replaced by a distributed intelligence that rests on the various technological tools available [4]. As a consequence, a new type of intelligence appears, the so-called environmental intelligence, which comes into being through interaction with the various ICT.
Among the strategies to incorporate ICT to the educational process, online group work can be mentioned, which is a type of work that allows several individuals to collaborate in order to generate a greater diversity of concepts and criteria through the use of digital tools that foster interaction [15].
There are different learning techniques that promote distribution and communication among the participants in a group. The Metaplan technique can be considered as a group moderation technology that helps obtaining results through visualization and questions. It can be implemented in various fields of action, such as planning, problem solving, participative decision making, requirement diagnosis, group assessments and feedback, teaching and learning processes, debates and workshops, and so forth [5][6] [11].
In its original format, this technique is used in an entirely on-site fashion. The incorporation of digital technology to this technique could be helpful to broaden its scope and incorporate more participants from various spaces and at different times. The virtualization of certain stages is proposed, fostering group interaction while generating ideas and knowledge [14].

2. Theoretical Framework

The concept of on-site and virtual collaborative work is reviewed from the standpoint of Distributed Cognition, which focuses on understanding the organization of the cognitive systems used by individuals in their environments. The boundaries of cognition can go beyond the individual to encompass phenomena that typically appear during interactions between individuals and the environments in which they move [1] [2] [3].
Distributed cognition is a branch of cognitive science that proposes that human knowledge and cognition are not limited to individuals exclusively. This perspective allows analyzing the communication between devices and their context [18].
Cognitive processes can be distributed among the members of a social group or community. They are distributed in the sense that the operation of the cognitive system includes the coordination between the internal and external components (both material and environmental) in its structure.
Processes can also be distributed through a certain period of time, so that all previous events can transform the nature of related events in a cognitive ecosystem.

From an information technology standpoint, the term "groupware" is defined by integrating the software and the human components. Groupware takes into consideration the technical problems and the organizational and social implications of introducing ICT. Group work through digital media should be developed by adaptation to support the goals of the group and in accordance to the process used. The evolution of the human and technological system should be balanced so that social implications are not disregarded and new organizational structures and roles can be created [14] [16].

The context fostered by ICT offers educators a range of tools, and they must be prepared to use them correctly. There is a term associated to the relation educators-ICT – "TLK" (Technologies for Learning and Knowledge) [17]. Juana Sancho reflects on the use of ICT and proposes a pun in Spanish – TIC/TAC, based on the Spanish versions of ICT and TLK.

TLK help direct ICT towards more educational uses, both for students and educators, to help students learn more and better. They are specially aimed at methodologies and technology uses, and not merely ensuring the mastering of a series of information tools. Ultimately, their goal is to know and explore the possible didactic uses of the ICT for learning and teaching. TLK go beyond simply learning how to use ICT – they focus on exploring these technological tools put to the service of learning and knowledge acquisition [13].

This change from ICT to TLK goes hand in hand with processes for training educators and the organization of the educational system in general, as well as applying this in the classroom.

When study materials are produced considering the communicational possibilities of the language used to convey the message and the medium through which the message will be conveyed, better learning can be achieved [11]. In this regard, it should be noted that ICT can enable massive distributed cognition processes that would be hard to organize analogically.

In relation to the group work process, this can be reinterpreted from the concept of workshop space of the on-site classroom. In this context, interacting individuals usually are exposed to new, enriching opinions and conclusions, which allows them to broaden their knowledge on the subject being discussed.

The Metaplan technique presents a possible way to achieve the intrinsic motivation to learn and ensure the use of learning processes that favor the interaction student – study concepts/materials – educator – other students, for the achievement of a certain learning objective [6].

The Metaplan is divided into stages. Each stage was analyzed to determine if its virtual implementation was possible. Thus, Metaplan could be used as a technique that uses ICT to facilitate a change of role from educator to moderator and learning facilitator, but still around the concept of teaching. This proposal could be useful for the development of teaching practices of the ICT/TLK type.

2.1 Metaplan Technique

The Metaplan technique is one of the possible techniques for fostering collaborative work, and it can be seen as a set of "communication tools" to be used by groups that are looking for ideas, solutions to their problems, opinion development, agreements, objective formulation, recommendations and plans for action.

Cisnado Torres indicates that the Metaplan technique was conceived by Eberhand Schelle in Germany, and that its fundamental pedagogical tool is an "interactional situation" that, from a question or a thesis presented by the educator, elicits simultaneous and visible responses from all participants. Attention and tension can be maintained throughout the process thanks to the interest in corroborating if other responses confirm or oppose one's idea, or if they complement our knowledge on the topic being discussed [5][6].

The Metaplan is seen as a work tool applicable to training sessions, workshops and meetings. This methodology adds orientation in the form of a "moderator". The goal of any good moderator is to keep alive the interest of students for learning, encouraging them to be active participants by asking questions and raising doubts. Interaction occurs as the foundation for the learning process [10].

From a psychological standpoint, interactional learning leverages and exploits the intrinsic motivation of the educational process itself to favor the targeted learning.

Psychologists differentiate between two types of motivation – extrinsic and intrinsic [1][9]. The former comes from causes that are external to the topic or course in which the participant is involved, and is generated by an internal desire to avoid something negative or achieve an improvement in some aspect of life. The latter appears when the individual carries out an activity simply because they enjoy doing so, with no obvious external incentive. A hobby is a typical example of intrinsic motivation, as well as a sense of pleasure, overcoming, or success.

When a benefit is expected from taking a certain course, for instance a promotion at work or better pay, individuals are extrinsically motivated to participate. In work contexts, extrinsic motivation elements are not always possible for each of the educational proposals available. However, if the same desire to participate is achieved through learning development, when individuals enjoy learning just for the pleasure of learning, without considering external benefits, motivation is intrinsic.

For an educator, it is essential that students have a desire to learn, since this will affect both the trainer and the designer of the training proposal. Thus, participants are encouraged to:

- Carry out their own activity.
- Actively affect the development of activities.
- Carry out activities in collaboration with others.
- Master a new problem.
- Experience success.

- After the training is finished, have a sense of satisfaction in relation to the little effort required.
- Commit to something.

Interactional learning eliminates pedagogical supervision as operating method, thus forcing participants to develop their intrinsic motivation [7].
Pedagogical supervision generates a subconscious and unavoidable feeling on the individual that they are forced to learn by someone who has more knowledge on the subject being learnt. The "supervisor" can be simply a book, a person or a virtual tutor. People have this mechanism rooted into their memories from their childhood, and every time we are in a position that we need to learn something, this need for guidance comes once again to the surface. Learning is usually associated with effort, usefulness in the medium or long term, difficulty, competition, situations with predominance of the inferiority and dominance dialectics.
Interactional learning tries to get participants to develop the following ideas:

- It is interesting for me or by itself.
- I can learn and I care about learning.
- It is useful, it will work.
- I can do it together with other people who share reality with me.

How does interactional learning take place with Metaplan? There is a moderator whose main role is to help improve mutual understanding. His/her purpose is that of offering the group the necessary communication techniques at the right time for participants to be able to effectively find solutions. He/she starts by asking questions to the participants [6].
Once the moderator collects the opinions of the participants, he/she groups them by similarity. For each new idea that has no similarity with the ones that have already been presented, a new group is created, called "cloud of ideas"; otherwise, it is grouped with the similar idea. The moderator puts together the clouds of ideas, generating a new sub-topic for each cloud that the moderator distributes among the participants of the so-called Metaplan "session."
The moderator is responsible for distributing the subgroups and sub-topics [6]. Then, for each group, a "recommendations list" is prepared, which ends up being a plan of actions that have to be approved and makes reference to the topics, wishes and actions proposed by the groups. These elements are added to the list and sorted by significance, so that the issues that require action are recorded.
Finally, the entire group participates in the debate and a "list of actions" is generated in relation to the activities that can be carried out. Each action to be performed is assigned an owner and a group of people in charge of carrying it out.

The entire process of the Metaplan would be as follows:

- Presentation of the central topic
- Anonymous contribution of each participant on the central topic
- Division of opinions into sub-topics -> Cloud of Ideas
- Division of participants into subgroups and assignment of sub-topics to them

- Discussion of the sub-topics assigned within each subgroup: presenting opinions and/or ranking existing answers
- Design of the recommendations list of each subgroup
- Each subgroup presents their list of recommendations to the general group
- Debate as single group in relation to the ideas presented by the subgroups
- Conclusion and summary. List of actions.

3. Prototype Development

After analyzing the Metaplan, it was observed that the virtualization of certain stages of the technique may:

- Expand the scope of the training on the technique.
- Favor the teaching and learning process of the methodology, considering the aspects of learning time, space, style, and pace of each student, promoting their autonomy in the process.
- Take advantage of the ICT for carrying out the workshops.

A prototype was developed for the administration and development of workspaces through the Web using Metaplan's group technique with the added component of virtualization of the necessary stages to be able to perform each task.

A software development model called "evolutionary prototyping" was used to test and modify development phases.

The prototype obtained allows virtualizing the stages of anonymous contribution by the participants on the central topic. The moderator can divide the opinions into sub-topics and generate the "clouds of ideas". Spaces can be created for the debate and subsequent creation of the recommendations list by the subgroups in such a way that those individuals who cannot be present in all Metaplan sessions can be involved in a workshop that uses this teaching methodology.

The prototype has an interface with three templates, one for each user profile: Administrator, Moderator and Participant.

The *Administrator* template allows having control over system data, including user management and access to the different sections of the site, course management and enrollments.

The second template is the interface used by *Moderators,* which allows managing virtual workspaces and tracking participant interaction both when building the cloud of ideas and during the discussion stage. It has different sections for managing groups of participants, the topics into which the course is divided, and the discussion forums that will be used by the groups.

The third template is the interface used by *Participants,* which allows students to enroll, express their opinion on the central topic proposed for the course, interact with the members of their group through a chat utility

(synchronous communication tool) and/or using the discussion forums (asynchronous) created for each topic assigned to the group.

3.1 Technical Requirements for the Prototype

The application was designed to work in a free software environment. It requires that certain dependencies are installed before executing it. These are:

- MySQL 5 or higher
- PHP 5.2.4 or higher
- Symfony 1.4
- Symfony plug-ins to install: sfPropelPlugin, sfJQueryUIPlugin, sfProtoculousPlugin, sfJqueryReloadedPlugin, sfFormExtraPlugin, sfTCPDFPlugin, Mail Server
- Eclipse is used as IDE (Integrated Development Environment), programming in PHP5 with a Symfony template.

The initial data model for the application prototype has the following elements:

- Session: to model Metaplan courses.
- Discussion: opinions, comments and clouds of ideas to group opinions.
- Topics: concepts around which participants will interact and discuss.
- Interaction: chat rooms and forums for the virtual discussion among participants.

3.2 Operation of the Prototype

The Moderator is responsible for distributing session information and managing the groups of participants and session sub-topics. Moderators are also responsible for organizing the topics discussed in the courses, mediating during discussions, assigning users to groups and assigning the topics to the groups. The prototype provides discussion forums and the moderators are in charge of their administration and moderation.
The *participants* can manage the information pertaining to the sub-topics assigned to their group by the moderator. They enroll in courses, express their opinions, interact with the other members of their group and propose possible solutions to the topic presented in the list of recommendations.
Site administrators are in charge of managing user-related information, menus and topics pertaining to the configuration of the website. As shown in Figure 1, Administrators manage the types of user profiles, the menus and their access.

Fig. 1. *Session management in Metaplan*

3.3 Steps to Follow in the Virtual Metaplan

Each virtual workspace has a central topic around which discussions and opinions will revolve, a range of dates that represent the validity period for the course, and a person that will act as mediator between topics and groups. During the initial session, the "central topic" is presented on which the participants will later on express their anonymous opinions and discuss in general to build the "cloud of ideas," which acts as a map that will collect the main opinions that will then be transformed into the sub-topics that will be discussed in the subsequent stages. The central topic is presented at the first on-site meeting of the Metaplan.

Once all participants present their opinions, organized in a cloud of ideas, they will publish them for the moderator to view them in the cloud map. At this stage, course participants and the moderator will have another on-site meeting and put together the final cloud map for the central topic.

The moderator can carry out several actions, but none of them without the agreement of course participants, because the Moderator is simply accompanying them in the process. For this task, the Moderator can edit a specific cloud (changing the title of the cloud or removing an opinion from it), swap opinions between clouds, and/or remove clouds.

When publishing the cloud map, the moderator will produce the end result for this stage, which will be the clouds of ideas that are selected; their titles will be the sub-topics that the groups of participants will discuss at a later stage. In this process of topic publication, the status of the Metaplan session will go from "initialized" to "topics_published," and all that remains to do is distributing the sub-topics among the participants for discussion.

The final decision making and agreement process is done on site.

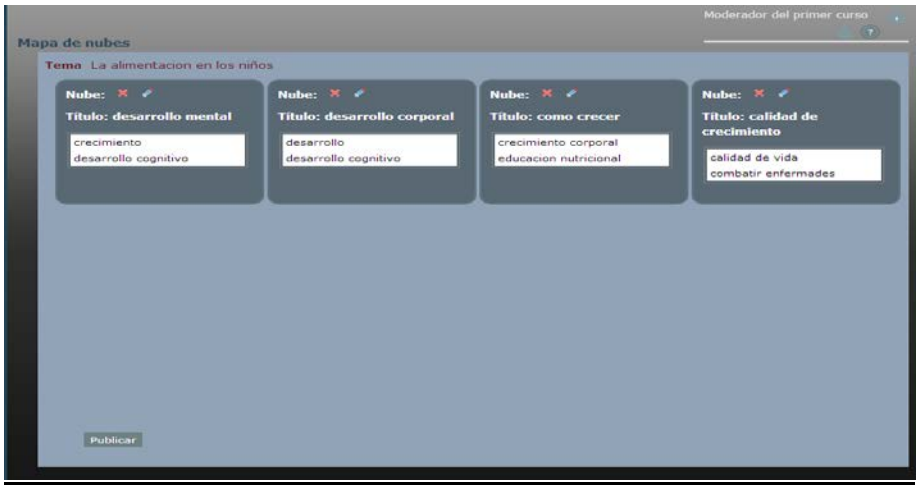


Fig 2. *Clouds of ideas*

4. Conclusions and Future Work

The Metaplan technique was analyzed and described. Stages were identified to determine which of them could be done in a virtual fashion.

An evolutionary prototype was built to present the three essential interfaces: Administrator, Moderator and Participant.

The initial idea was developed to be able to share and view the clouds of ideas.

The review of the Metaplan technique allowed analyzing each stage and proposing virtual stages. The generation of clouds of ideas is a stage that allows working asynchronously and is aimed at improving times between Metaplan sessions. The processes in each stage were related by applying the distributed cognition principle.

The concept of ICT transformed into TLK was discussed, where educators can use the tool for workshop activities and focus on ICT as media for learning and knowledge. The relation between educator-concepts-students is strengthened through the integrated work of all participants, both during on-site and virtual sessions. The change in the role of the educator is essential for the proper application of the technique.

As possible extensions to this work, a synchronous tool could be integrated to improve the visual presentation of the clouds of ideas. Additionally, there should be an option for tracking participant actions in order to have statistics that help visualize collaboration and the use of the application.

In the future, the implementation will be improved based on the feedback collected from a pilot experience, and access to the tool will be offered from a website.

Bibliography

1. Ainscow, M. & West, M. (2006). *Improving urban schools. Leadership and collaboration.* Maidenhead: Open University Press.
2. Avila Patrica M., Bosco Martha D. (2001). "Virtual environment for learning a new experience. Abstract ID: 1510. Trabajo presentado en el "20th. International Council for Open and Distance Education". Düsseldorf, Germany.
3. Barberá E. (2008). "Calidad de la enseñanza 2.0. Educational quality 2.0". RED: Revista de Educación a Distancia, ISSN 1578-7680, Nº. Extra 7. Spain.
4. Cabero Almenara J., Llorente Cejudo, M. C. (2007). "Propuestas de colaboración en educación a distancia y tecnologías para el aprendizaje". Edutec. Revista Electrónica de Tecnología Educativa. Núm. 23.
5. Cisnado Torres, X. (2008). "Metaplan, una metodología de diagnóstico y moderación grupal". Centro de capacitación. Contraloría general de la república. Costa Rica http://jaguar.cgr.go.cr/content/dav/jaguar/documentos/capacitacion/web_centro /Metaplan/Metaplan.htm
6. Cisnado Torres, X. (2007). Virtualización de la Enseñanza-Aprendizaje de METAPLAN, www.infodesarrollo.ec/component/docman /doc_ download/ 132-virtualizacion-de-la- ensenanza-de-aprendizaje-de-Metaplan.html
7. Delgado Fernández M., Solano González A. (2009). "Estrategias didácticas creativas en entornos virtuales para el aprendizaje". Revista: Actualidades Investigativas en Educación. Volumen 9, Número 2 pp. 1-21.
8. Diaz Barriga, F. (2011). "La innovación en la enseñanza soportada en TIC. Una mirada al futuro desde las condiciones actuales". VII Foro Latinoamericano de Educación / Experiencias y aplicaciones en el aula. Aprender y enseñar con nuevas tecnologías
9. Dror, I. E. & Harnad, S. (2008). Offloading Cognition onto Cognitive Technology. Cognition Distributed: How Cognitive Technology Extends Our Minds (pp 1–23). Amsterdam: John Benjamins Publishing.
10. EPISE, Metaplan (2005). Sesiones formativas y reuniones de trabajo más efectivas, http://www.epise.com/episecms/galeria/documentos/Metaplan_21_ene_08.pdf
11. Hanusyk, K. (2010). "Introducción al Método de Moderación". Available at http://www.klaushanusyk.com/, last access on March 2011.
12. Litwin E., Maggio M, Lipsman M. (2004)."Tecnologías en las aulas. Las nuevas tecnologías en las prácticas de enseñanza. Casos para el análisis". Amarrortu editores. Buenos Aires-Madrid.
13. Lozano, R. (2011). "Las 'TIC/TAC': de las tecnologías de la información y comunicación a las tecnologías del aprendizaje y del conocimiento". Anuario ThinkEPI, v. 5.
14. Madoz C., Gonzalez A., Saadi M., Hughes D. (2010). "Virtualización sobre un entorno de Enseñanza y Aprendizaje de métodos de trabajo colaborativo". Presentado en el TEyET, Calafate. Santa Cruz. Argentina.

15. Prendes Espinosa, M., Martínez, P. (2006). “Actividades individuales versus actividades colaborativas”, en E-actividades: un referente básico para la formación en Internet, ISBN 84-665-4768-1, pp. 183-202.
16. Rama, J., & Bishop, J. (2006). “A survey and comparison of cscw groupware applications”. Annual research conference of the South African. Somerset West, South Africa.
17. Sancho Gil, Juana M. (2008). “De TIC a TAC, el difícil cambio de una vocal. Revista Investigación en la escuela”. Núm: 64, Pág.: 19-30. Biblioteca: DIE. Universidad de Barcelona. Last access in April 2012 from http://www.ub.edu/esbrina/docs/proj-tic/tic_a_tac.pdf
18. Solomon, G. (2005) “Distributed Cognitions. Psychological and educational considerations”. Cambridge University Press.

First steps in developing immersive virtual learning environments by using open-source software

LILIANA LIPERA[1], IRIS SATTOLO[1], GUILLERMO SUTZ[1], HERNÁN MONTI[1], JOSÉ MANUEL GARCIA[1]

[1] Facultad de Informática Ciencias de la Comunicación y Técnicas Especiales (*Faculty of Information Technology, Communication Sciences and Special Techniques*)
Universidad de Morón (*University of Morón*), 134 Cabildo St., (B1708JPD) Morón, Buenos Aires, Argentina
54 11 5627 2000 extension 189
llipera@unimoron.edu.ar, iris.sattolo@gmail.com, gsutz@cnia.inta.gov.ar, manuel@latinled.com.ar, hernanmonti@gmail.com

Abstract: *New technologies based on multimedia and Internet offer innovative ways of learning and teaching. One of these ways, which has not existed until recently, is interaction through the senses of vision, hearing and touch with learning objects and situations as well as through the process itself of creating these objects. In the last few years, there exists a tendency in universities worldwide towards building three-dimentional virtual environments in said institutions. Therefore, this paper presents the work carried out so far in the University of Morón, as process in the construction of a metaverse allowing new learning strategies to be designed. It was developed with OpenSim as free, open-source environment allowing the exploration of this tool for developing settings that enable their construction by applying this technology to education in the future.*

Keywords*: Virtual Reality, immersive environments, E-learning, Metaverse, Open-source.*

1. Introduction

This article describes the work currently being developed within the area of artificial intelligence applied to the development of virtual learning environments and belongs to a research project to be homologated by the Secretariat of Science and Technology of the University of Morón.

1.1 Virtual environments and education

One of the challenges every teacher faces is to make the class an interesting meeting place for students, where motivation plays a decisive role in the learning process, which leads to find appropriate pedagogical strategies.

One of the strategies recently being used is the recreation of virtual environments, where the creation of own contents and multi-user interaction are allowed through text, audio and video.

Throughout history, humankind has experienced countless changes; since the last decades of the past century, we have been living in a period of communication and information technology breakthroughs. Information society, as this era has been named, has accomplished great transformations and brought benefits in all processes, administrative structures and jobs of people and institutions involved. It is then reasonable to expect that new alternatives in the education environment apply these breakthroughs. The advent of new technologies is accompanied by the knowledge generated from new information, but wider and more complex phenomena must be taken into account in a society of knowledge. [1]

"The university and teachers in particular must contribute with an innovative educational practice to accompany the transition from an information society to a knowledge society. They must acquire certain skills and attitudes allowing them to employ innovative strategies and alternative models by means of Information and Communication Technologies (ICT), where the student has an active role and more responsibility in the formation process." (Mariño 2008). [2]

Within these new ICTs, immersive environments are found. UNESCO (1998) [4] informs, in its world report on education, that immersive learning settings are a brand new form of education technology by offering a series of opportunities and tasks to teaching institutions worldwide. These Immersive Virtual Learning Environments (IVLE) are neither automatic nor generated only as a result from new technologies; the pedagogical design is decisive for virtual communities to emerge. To design these learning environments, modifying traditional attitudes, ideas and mechanisms between teachers and students must be taken into account.

Immersive environments can be defined as settings which allow the recreation of real or imaginary tridimensional contexts generated by computers which the user can interact with and feel like being part of. [6] These contexts which have been widely used in entertainment, films and video games are being used in education in the last few years. The higher education environment has taken the initiative.

A definition for Immersive Virtual Learning Environments or IVLE is "3D technological platforms for supporting the process of virtual and non-virtual (traditional) education to which you can access through Internet or a local network allowing students and tutors to connect in order to be represented by a 3D virtual character."

In this environment, students can move freely through learning contexts and communicate in real time by using voice and text systems to perform collaborative formation activities enabling a very high level of interaction with the learning objects from the environment.

Virtual worlds or metaverses, such as Second Life, Kaneva, There, Moove, Cybertown and Active Worlds, have been used since 2001 and are now becoming stronger in universities in different places (North America, Europe and Asia). Currently, the most known simulation environment probably is

Second Life, which gathers the largest number of learning centers and universities-over a hundred and forty centers since May 2008; the majority of pioneer universities are found among them. (Silva, 2009, pp. 20-21) [3]

In 2007, project OpenSim is born with the aim to create a 3D application server by analyzing the structure of the SecondLife customer (inverse engineering). Being a Free Software (BDS License), having a Modular Structure, holding multiple viewers or customers and being written in C# are the characteristics which make it attractive to use. This enabled universities to build their own spaces (islands) without having to pay for land or textures and objects offered by Second Life. [5]
A question is posed in this context: may this technological tool creating new contexts of teaching-learning bring the student-teacher-knowledge closer in a playful, novel and successful way?
The recreation of an immersive environment in the University of Morón by applying Free Software opens the doors to new proposals within education, for which new teaching strategies must be developed according to the didactics thought.

2. Development

The research team is currently made up by engineers and information system graduates (licentiate degree) working as teachers and by thesis students from the Graduate Degree in Information Systems (licentiate degree), all of them aiming at making an interdisciplinary team to contribute to the didactic and specific requirements of the subjects to be proposed to begin with field tests.
The first stage was acknowledging the existing technology by evaluating its advantages and disadvantages. It was decided to adopt the OpenSource technology, which exempts us from possible policy changes in paid contexts. Among the alternatives found, the OpenSimulator server was chosen.

2.1 Open Sim

OpenSim is a 3D server with open code which allows the creation of virtual environments which can be accessed to through a great variety of viewers (customers) or protocols (software and web). *OpenSim* is a framework easily configured for every need, and can be extended with modules. The *OpenSim* license is BSD, allowing it to be of free code and simultaneously used in commercial projects. To date (July 1st, 2013), the version 0.7.5 is available. So far, there exist two ways of configuring the server –independently (standalone mode) or using a network. Independently, the SQLite (light database not applying persistence) database is used by default, but it enables configurations with MySql and MSSQL databases. [5]
The configuration of OpenSimulator consists of data regions and services. Independently, data regions and services execute in one process. In a

network, data services are not part of the region server process. Instead, a service called Robust.exe is executed. This allows several OpenSim to be executed in different physical spaces.
The execution in a network needs a better compression of positions (x,y of regions), passwords, parcels, real estate owners; thus, it was decided to work in Standalone mode in this first stage of the research.
From the user's standpoint, there are multiple viewers (Phoenix, Imprudence, Hippo Viewer and Firestorm among others); these are customer applications installed in their computers which become the means enabling them to enter and enjoy these metaverses.

3. Proposed Solution

In order to start our metaverse, a Pentium IV HT with 2Gb RAM and Windows XP SP2 was chosen. OpenSim works with all versions of Windows XP or up. It is also possible to execute it in Linux OS.
In a first trial, the Standalone version with default database SQLite connection was used, but when trying the MySql connection, it was proven that the Simulator worked better.
The Internet connection where the server is based is a home one with 3 Mbps downstream speed and 512 Kbps. OpenSim Diva Standalone distribution was chosen for installing the metaverse. Its advantage is a Web interface allowing the creation of users (avatars) in an autonomous way, their management, region management and getting inventory information. The installed version is Diva-r22458. [7] [8]
Figure 1 presents the component diagram explored so far. From the customer's standing point, the web page interacting with the simulator and allowing user creation is accessed to. 3D viewers allow access to regions offered in the created virtual world.
The Diva version is already configured with four regions, which display an OAR file offering a starting region where the avatar can choose its appearance. This region enables the construction of objects by the different users. There exists an entry room with several classrooms. In order to build within the building, permits must be obtained; this task is performed by the administrator from the console.
The incorporation of both OAR and IAR files is done from the administration console. Another interesting console task is observing all events and mistakes that might be happening in the 3D server. The Server configurations are performed from INI files making up *OpenSim*.
Through the console, and apart from creating users, it is possible to modify the land, send messages to all users and establish security and all matters concerned with the simulator administration.

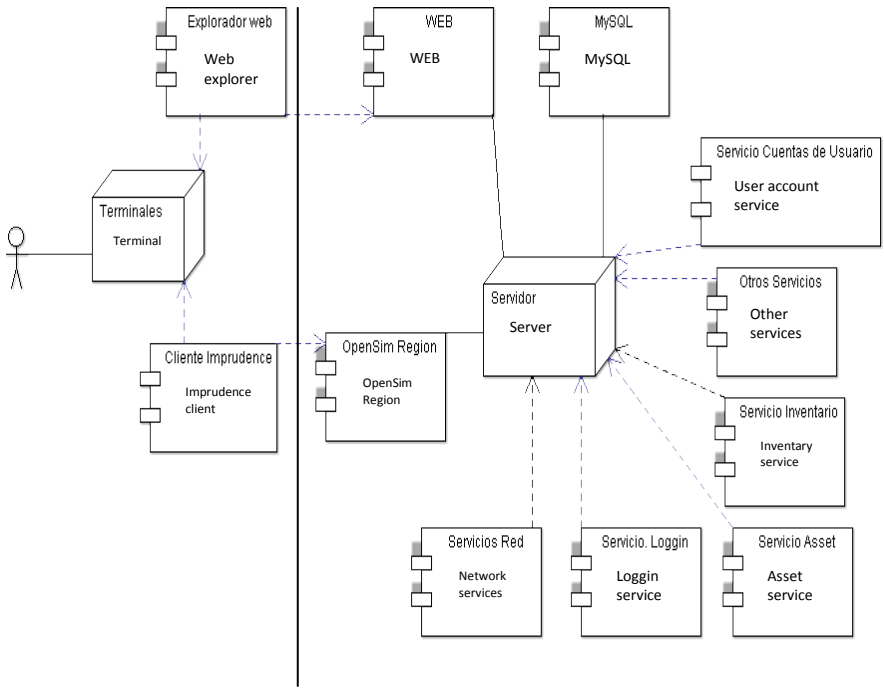


Fig.1 *Diagram of DIVA distribution components in Standalone mode.*

The DIVA distribution [7] has chat services but without voice. Research showed that the company providing voice services for Second Life is Vivox. This company offers communication services such as voice chats, instant messaging and is present in games, online, in virtual worlds and other online communities. The company provides a free version for OpenSim platforms and it is available for individuals, charity organizations, educators and small social networks. This service was requested through the web page offered by the company, and once the project approval was accepted, Vivox Free Virtual World Voice Service was connected to OpenSim server.

The first immersion trials were conducted with avatars, which were uploaded with textures from viewers without any inconvenience.

The second trial phase was conducted with avatars configured from the simulator-provided regions. As it was mentioned before, the starting region provides pre-configured avatars with textures given by the region. This trial complicated the uploading of some avatars, which were displayed diffusely (cloud-like). *Figure 2* shows a screen capture of this trial.

***Fig 2**. Screen showing the difficulty in avatar uploading with textures given by the region.*

It was decided that the textures uploaded for avatars were not going to be those given in the regions, but the clothing having the functionality stated by the viewer. Therefore, there was no inconvenience in visualizing different avatars. Trials were performed with six avatars online, which were able to use the voice service with prior panel configuration in the viewer. *Figure 3*

***Fig 3** Screen showing the interaction of six avatars with viewer-provided textures.*

The functioning of bots –autonomous program in a network, especially Internet, which can interact with computer systems or users– is being investigated. In order to run them, there are two possibilities, which basically differ in that one is programmed and executed in the client and the other is in the server. Both solutions offer advantages and disadvantages when

comparing them to one another, which are being assessed to define which one will be implemented.

4. Results

During this first stage, an OpenSimulator server was configured. It was accessed to and interacted with by the research team verifying the following items:
Freedom of access and avatar movement representing users.
These can move through scenes walking, circling objects, flying and tele-transporting from one region to another.
The possibility of managing collisions was detected, since the avatar cannot go through walls or objects showing a physical structure.
Objects can be animated from the application by means of a script or from outside.
Prospects of including AI (artificial intelligence) algorythms in behaviour simulation.
Spacial sound. The simulator enables to have own sound for different environments, proving that the voice is weaker until disappearing when the avatar walks away.
User interaction from remote places and in real time, thus allowing the construction of group knowledge on any topic arising at the time. This interaction implies having control over the created system.
There existed a first approach to lsl language used for developing script for objects with which avatar-object and object-object interactions can be performed from any client and using any compatible viewer. This language is based on events, states and functions with a similar syntax to C, enabling its quick familiarity and comprehension.

5. Conclusions and prospective research lines

The use of virtual worlds, especially in education, is being accepted worldwide as a tool, bringing knowledge in a different way to young people and adolescents and so, enabling own experiences with objects and a world, which is not indifferent to users since these worlds are used in mini-games.
The prospect of having a metaverse in the University of Morón to develop contents for several areas will allow access to students to different lectures benefiting from online and non-virtual (traditional) education. Attending a virtual lecture without having to physically attend the university class will be an influential achievement on students. Building different spaces representing a reality difficult to access or impossible to accomplish in a conventional education environment is a set goal that has an impact on the positioning of our University in the current world.

As first step in research line, it is proposed to create a metaverse in grid mode installed in the laboratory of the University of Morón. The second step deals with the creation of new tools to enhance cohesion between the virtual and real world and be used in online education platforms. This link will not only improve the sense of immersibility by fading the frontiers between both worlds but will also attract users to adopting this technology. Among the tools being considered and developed, the creation of remote desktops allowing simultaneous lectures in a virtual and real classroom and text messaging from the metaverse to real-world mobile phones are found.

References

1. Nuevos contenidos en educación a partir del EEES by Ma Teresa Piñeiro Otero (2011).
2. Mariño, Julio Cesar Gonzalez.TIC y la transformación de la práctica educativa en el contexto de las sociedades del conocimiento rusc vol. 5 n.º 2 (2008) | issn 1698-580x.
3. Silva, M. (2009). La universidad en los mundos virtuales. Educación y Mundos Virtuales Edición 24, pp. 20-21.
4. http://www.slideshare.net/RamnMartnez1/declaracin-unesco-1998
5. http://opensimulator.org/wiki/Wifi
6. http://www.revista.unam.mx/vol.8/num6/art47/jun_art47.pdf
7. http://www.marlonj.com/blog/2012/04/instalando-diva-distro-opensim-0-7-3-en-ubuntu-server-11-10/#sthash.r79wI8CD
8. http://metaverseink.com/Downloads.html consulted on June 5th, 2013

Design of a Learning Mathematics Application based in Android Technology

RUBEN CACERES[1], ROY GENOFF[1], LEANDRO AYALA[1], PATRICIA ZACHMAN[1]

[1] Basic and Applicated Science Department
Universidad Nacional del Chaco Austral.- Argentina
eu_rubens87@hotmail.com, roy1885@hotmail.com, leansvaker@gmail.com, ppz@uncaus.edu.ar

Summary: *In the last years, it has been observing an exponential rise in the use of the so-called smartphones, as in the consumers' habits.*
It's a fact that since the potential connection of data showed up; practically all tasks that need a PC's use before, now they can be carried out in the smartphones. Actually, there are four operative systems in which the development of the main mobiles applications is based on, which are: Android from Google, iOS from Apple, Windows phone from Microsoft and BlackberryOS from RIM.
Despite of having fewer applications, Google can boast about being the most widely used operative system in mobiles devices.
This project focuses on the development of a native applications pack for mobiles devices with Android operative system as a teaching resource – basic mathematical learning in Superior Education. The goal of this application is focus on allowing the integration of the manual analytic resolution, of different mathematical problems, with the technology m-learning, from formation processes, self-control and informal assessment in the context of university income.

Key Words: *Apps, Mathematics, Teaching-Learning, m-Learning, Android.*

1. Introduction

The domain of mobiles technologies by the new student's generation, has allowed identifying didactic paradigms based on the context of ubiquity [1]. It is necessary, therefore, to recognize the changes that affect these current teaching to facilitate the enabling role of teachers in a technological environment, and move toward a contemporary educational concept computer-mediated communication.

Similarly, mathematics is presented as one of the essential knowledge in societies with advanced technology development and yet, the reality shows that it is one of the areas with the greatest difficulties of performance for a great deal of the university students observed as a cause of repeated failures and dropouts during the educational income[2]. The use of cognitive strategies and cognitive goals math seems to be inconsistent with the

heuristics used to analyze or solve disputes, an intuitive inductive reasoning, and hypothesis testing.
In this regard, it is considered that the mathematical content should be strengthened to initial level, not in the context of axiomatic mathematics, but in its intuitive essence yet formal, so as to enable the new students experiment in a pleasant and creative way "learn - do math".
Applications for mobile devices or Apps are designed and created to provide a multitude of services to mobile users. The most popular applications are oriented to social networks (Facebook, Twitter,...) or messaging services (WhatsApp) but also include online banking applications, tracking applications and applications aimed at business use, among others.
This project had made emphasis in educative applications, concluding in the development of a Mobile Apps Pack, as an informal instructional support tool to students- teachers, in the self-management and self-assessment of mathematical problems solutions using Android, a Google platform, for mobile systems.

2. University Context

The National University Austral of Chaco (UNCAus) has in its academic offer 14 undergraduate programs, for which it is necessary to complete a required but not elimination Entrance Course in Mathematics. In 2012 the UNCAus received nearly 1000 new students (as thrown data by the SIU UNCAus 2012). The initial cultural and educational situation of the students shows a considerable heterogeneity. Consequently it is necessary to promote a common starting base that ensures students equal opportunities, give the diversity of preparation which the students graduate from the Medium Level.
In the context of UNCAus, one of the initiatives is the Joint Plan between the Medium level – Polymodal and Superior, through face to face and virtual courses of leveling.
Through various inclusive strategies it has been recognized the communication paradigms changings that fall on the didactic measured by the technologies to transform the educative effort, focused on the text reproduction, the discovery and exploration of the contents for the knowledge self-built and self-regulation.

3. The agile trilogy: Apps – Android – Mathematics

The mobiles apps are programs developed in order to run in mobiles devices and address a specific task [3]. A mobile mathematical informatics application is an educative program destined to solve one or various specifics problematic situations of the mathematical environment, using as a based platform, cell technology.

The mobile learning (m-learning) is the acquisition of knowledge through any mobile computing technology [4]. For mobile computer it understands as smartphones, personal digital assistants (PDAs), netbook, tablet PCs and perhaps, depending on size, laptops.

As the mobile application business is expanding and becoming profitable, we have to investigate the optimal software development methodologies for such application and environments that takes these developments to success in an attractive and efficient way. The mobile application developer faces, also, with a highly fragmented scenario, comprised of many incompatibles platforms, as Symbian, Windows Mobile Brew, iPhone SDK, Android, Linux or Java. All this makes the process of development for mobile platforms more complex.

The idea of an agile methodology has two clear motivations: a large number of projects that are delayed of failed; and the low quality of the software being developed. The search for the solution involves a number of factors: most of the effort is a creative process and requires talented people, these processes are difficultly schedulable, modify software is cheap, testing and code review are the best way to get quality and communication failures are the main source of failure.

As noted in [5], there are five main factors that affect the agility of a software development process: culture operation (operating culture, behavioral norms and expectations that govern the behavior of people, both their work and in interactions with others), size of the development team, criticality of the software (both development time and specific features the software must fulfill or imposed by the elements where it will be run), technical competence of the developers and, last but not least, stability requirements.

Also argue that a software development method works best when applied to situations with very specific characteristics, this kind of situations are called "*home ground*" of the method of software development. Table I shows the comparison between the bases of agile methods and processes of development by plans or "planned" (plan-driven).

Area	**Agile Methodic**	**Classic Methods**
Developers.	Collaborative, states, agile and understood.	Oriented to a plan with a mix of skills.
Students - Teachers (Customers)	Are representative and they were given power.	Mix of aptitude levels.
Trust.	Interpersonal tacit knowledge.	Documented explicit knowledge.
Requirements	In large part emerging and rapidly changing.	Knowable early and fairly stable.
Architecture	Designed for current requirements.	Designed for current and near-future requirements.
Refactoring	Economic.	Expensive.
Size.	Products and small devices.	Products and bigger devices.
Premium Value.	Quick Value.	High Security.

Table 1. Basis for agile and planned methods (taken from [5])

Definitely, the agile software development tries to avoid the tortuous and bureaucratic ways of traditional methodologies, focusing on people and their results. Promote iterations in the development throughout all the project life cycle. Developing software in short periods of time minimizing risks, each of these units of time is called iteration, which should last between one and four weeks. The iteration of the life cycle include: planning, requirements analysis, design, coding, review and documentation. The iteration should not add too much functionality to justify the launch of the product to the market, but the goal should be to get a working version without errors. At the end of the iteration, the team will evaluate again the project priorities.

4. Agile Methodic for the Development of Mobile Apps

Agile methodologies have certain properties that make them fully applicable to the domain of mobile software. The suitability of the agile methods as a potential solution to the choice of a development methodology is summarized in Table 2.

Agile Characteristics	Development for Mobile Platforms
High environment volatility.	High uncertainty, dynamic environments.
Small development devices.	Carried out by microenterprises (SMEs).
Identifiable Costumer.	Potentially, there are an unlimited number of end users, but customers are easy to determine.
Development environments guided to objects.	Java and C++.
Software to an application level.	While mobile systems are complex and highly dependent applications are very autonomous.
Short development cycles.	Developments periods of 1 to 6 weeks.
Small systems	The applications, although variable in size, usually don't overcome more than 10.000 code lines.

Table 2. Agile characteristics and the features observed in the development of mobile software.l

Android is a complete software solution for open source phones and mobile devices. It is a package that encompasses an operative system, an execution "runtime" based on Java, a set of low libraries and a medium level and an initial application set destined for the end user (all of them developed in Java). Android is distributed under a permissive free license (Apache) that allows integration with proprietary code solutions.
Android applications are written in Java, but not running on Java ME, but on Dalvik, a Java virtual machine developed by Google and optimized for embedded devices. The creation of an own VM is a strategic move that allows Google to avoid conflicts with Sun for the virtual machine license and make sure the power to innovate and modify it without battling within the

JCP. Every Android application runs its own process, with its own instance of the Dalvik virtual machine. Dalkiv has been written so that a device can run multiple virtual machines efficiently.

It is this context that gave the motivation for this project:

- Develop software applications (Apps) for mobile devices that allow interacting with the various concepts in the discipline field of university mathematics, using agile development.
- Establish a team of enterprising developers who can quickly create Apps for the various needs of the institution.
- Browse Apps creation in various development environments, particularly in the Android operating system.

4.1 Mobile-D, an ideal approach to the Agile Development of Apps

Mobile-D is a Finnish project created in 2005. It's a mix of agile techniques and mainly aims to get very fast development cycles at very small teams. It consists of different phases:
- Exploration: planning, defining the scope and the project functionality.
- Initialization: identification and preparation of all necessary resources.
- Productization: In this phase, the programming is repeated iteratively up implement all functionality.
- Stabilization: the latest integration actions are made to ensure that the project work properly.
- Test and Repair: testing phase, until reaching a stable version of the project, as set out in the early stages by the customer. Errors are repaired if needed, but nothing new is created.

4.2 The Development Environment

Android provides a plugin for Eclipse that extends the functionality of it and facilitates the development of applications for Android. It also provides the tools that this plugin use as ant scripts in order they can also be used from other environments such as Netbeans or Intellij IDEA15. Among the features of this plugin there are:

- Android Emulator. Allows you to choose between different mobile terminals and operating system version.
- Access to Android development tools like taking screen shots, port forwarding, the possibility to debug with breakpoints or view the status of the threads and processes running on the system.
- Assistant for the fast creation of Android applications.
- Code editors for different configuration files (XML) that facilitate their understanding and development.
- Graphical interfaces that enable the development of components visually.

5. Mo-Math

Mo-Math (Mobiles Mathematics) is a pilot Project to help the teaching-learning process, in the mathematical area, using mobile technology.
For the realization of the project were carried out the stages of Mobile-D, in the context of native applications.

5.1 Initiation

The influence of mobile devices and applications in mathematics teaching-learning process was analyzed, as a new teaching paradigm, through a series of executable applications developed with the Agile Development Methodology in Visual C # 2008 platform. These Apps were designed to run on Linux operating systems, as the starting point of the analysis. It took into account that in the context of the National University Austral of Chaco, there is a large number of teachers and students who have personal computers and netbooks operating system of this type. The applications developed are simple to use and understand, for both teachers and students. This first step allowed planning the modeling of a system, even more agile and practical, to be implemented in cellular technologies.
In this first stage, the scope and functionalities to improve of the placed applications in test mode were defined.
Another aspect to consider was established by the platform on which the project will be developed. Finally, considering a statistical analysis of operating systems installed on cellphones UNCAus students, we chose Android.

5.2 Productization

The second step was established by the realization of the pilot project on mobile technology.
There are many and various programming languages that allow us to realize the transfer of Mo-Math to mobile devices. One goal of the project was to develop the Apps as free software. Java turns to be the appropriate programming language to implement this transfer process, as this language let us develop free software.

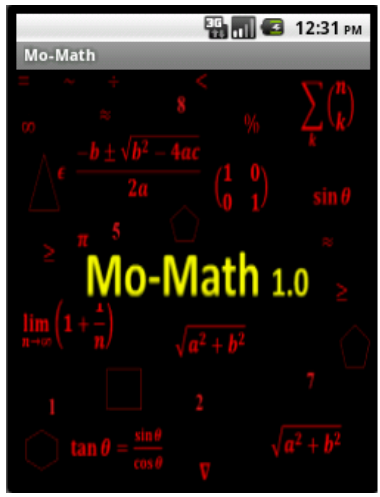


Fig. 1: *Main Screen of Mo-Math.*

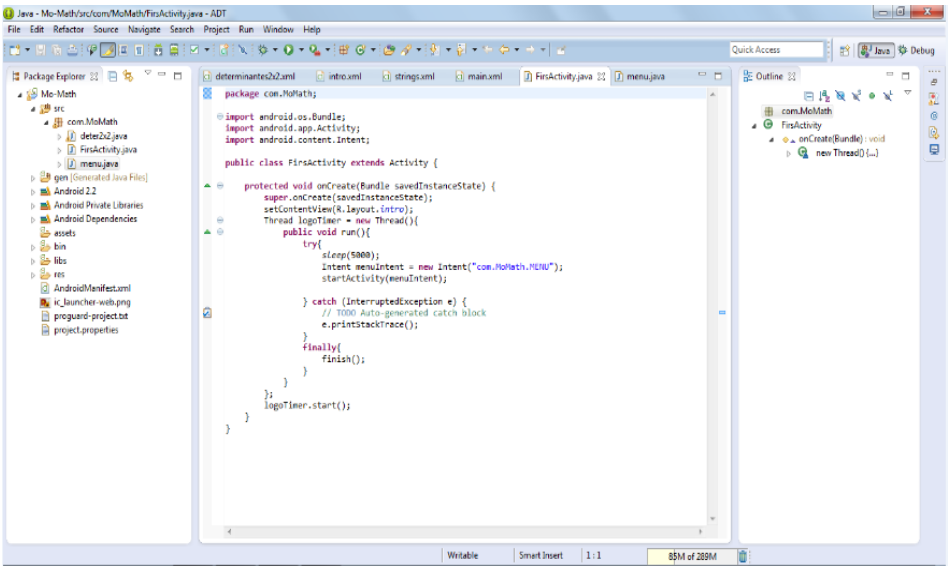


Fig. 2: *Java Job Environment used for Mo-Math*

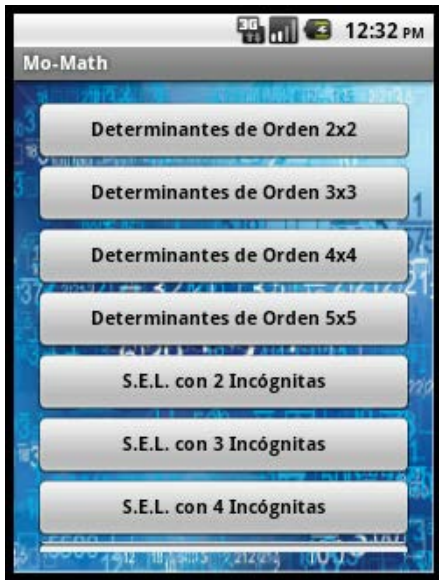


Fig. 3: *Menu of Mo-Math with the different Developed Apps.*

In terms of design, it must have analyzed whether the code fits perfectly or require changes.

5.3 Stabilization

Once developed programs that are part of this project, we took a control sample of 15 students from different races that belong to UNCAus.
The process of teaching and learning content consists of different parts. Firstly, after the corresponding explanations and demonstrations, students solved analytically by their own different mathematical exercises, and then analyzed and verified their results with mobile Apps. When the results obtained by the students did not agree with those obtained with the programs, an analysis of the errors arose, until they could reach the correct result. We concluded, in a first experience, a positive situation: a student motivation, promoting a constructive learning and self-monitoring results.

5.4 Test and Reparation

The results obtained in the test phase require review some aspects such as the interfaces and elements with respect to the addition of functionalities.
At this stage, the team is working in order to refine the product and deploy the application in other areas of UNCAus.

6. Mo-Math Features

There are much mathematical software, some of which are complex in its interface, in its way of presenting data, leading to confusion due to the large amount of present data in the complex displays, inputs and outputs. Mo-Math presents a number of advantages such as:

1. Facility with the user interaction: the interaction of Mo-Math with the users is fluid, no command use knowledge is required, besides explaining its use one is simple enough to understand its operation, only entering data and just one click the results are obtained.
2. Using simple screens. This is one of the most important advantages for the control students in the system test phase. Mo-Math for allowing the students an easy interpretation of data and results and also a simple and rapid introduction of values that do not require user manual or specific commands.
3. Easy installation of the software. The Mo-Math installation is extremely intuitive, is performed by an executable file and takes little time.

7. Conclusion and Research Areas

The software presented is intended to support the teacher and optimize the teaching-learning process. It is emphasized that these applications are aimed to final year students of secondary school and university entrants who must take the Mathematics workshop leveling area.

At each stage of the development of Mo-Math, the target audience was taken into account. The simplicity of the screens allowed fulfilling the aim of providing an easy and rapid data input and interpretation of results. The programs that include Mo-Math allow solving exercises and basic math problems of middle level and cover the fundamental mathematical modules area: Numerical Sets, Plana Trigonometry, Algebraic Expressions, Relations and Functions.

This Mo-Math software offers a new perspective on how to teach mathematics related to the use of ICTs, so that, this pack program should be taken into account and add it to the educational environment of the joint of the middle college level to exploit their potential to the fullest.

Considering a working environment of high volatility and dynamism, we success to establish as a key development element, the talent and organization of small development teams.

From the practical point of view and beyond that obtained as the results of its implementation on students and teachers, it is necessary to realize an analysis about the quick development for mobile systems that help to improve the agile cycle stages.

The results obtained in this study were positive in terms of their objectives, however, challenges to deal with connectivity issues can be larger and intends to work some areas as creating an application library evaluating current and upcoming technologies or extensions to use, generate the support to others operative systems like iOS (laptops, tables, smartphones and PDA iPod touch) and undoubtedly generate more and better dissemination of this technology.

Referencias

1. Trujillo A., Jaramillo, C. (2006). Estrategias didácticas en educación superior con mediación de la computación móvil, Revista Educación y pedagogía, Enseñanza de las Ciencias y de las Matemáticas, Medllin Universidad de Antioquia, Facultad de Educación, Vol. XVIII, Num 45, pp. 93-107.
2. Ramallo, M. (2012). Panorama Actual sobre el Acceso Universitario. Revista Academica Electrònica Sementral, Vol 1 Num 1, ISSN 2314-1530
3. Traxler, J. (2007). *Defining, Discussing, and Evaluating Mobile Learning: The moving finger writes and having writ*…International Review of Research in Open and Distance Learning, 8 (2), 1-12.
4. Traxler, J. (2009). *Learning in a Mobile Age. International Journal of Mobile and Blended Learning,* 1 (1), 1-12.
5. Boehm, B., Turner, R. (2003). Balancing agility and discipline: A guide for the perpexled, Addison-Wesley.

Art and ICT: Initial experiences with software tools in the training of Bachelor´s Degree in Combined Arts

FERNÁNDEZ M.[1,2], BARRIOS W.[2] GODOY M.V.[2] AND GENDIN G.[1]

[1] School of Arts, Design and Social Sciences

[2] Department of IT, School of Exact and Natural Sciences and Land Surveying

National University of the Northeast, Corrientes, Argentina

{mirtagf@hotmail.com, mvgg2001@yahoo.com, ggendin@yahoo.com, waltergbarrios@yahoo.com.ar}

Abstract: *An educational experience is presented in a group of university students who took part in a pedagogic innovative offer, where the principal goal was to create a sphere of reflection, experimentation and integration and establish links between science, art and technology. This educational experience consisted of three thematic axes: in the first one, the historical evolution and the relationship between the three disciplines were approached; in the second one, students worked on the creative process with the construction of an artistic device by using analogical means; in the third one, concepts about digital media were presented. Accompanied by the offered bibliography, the activities were based on the simulation in order to understand the physical phenomena and laws which science presents and their implication in art and technology. Upon completion of this project, conclusions were extracted on the subject.*

Key words: *Technology of the Information and Communication, Education on Arts, Multidiscipline Configuration, Image-Sound-Time-Space.*

1. Introduction

The Combined Arts emerged as a result of the expansion of the fields and frontiers of the artistic expressions throughout the 20th century [1]. In these processes, dialogues were established between different traditional disciplines such as literature, painting, drawing, sculpture, music, theatre, dance, photography and films. From this intermingling of specific languages, crossed curricular categories and practices have arisen, establishing multiple methods for exchange among them which articulate a strong theoretical basis and intensive practice.

Due to this fact, and as a response to the unexplored areas in the training and professional practice in the region, which started in 2012 with the new Bachelor´s Degree in Combined Arts from FADYCC[1], which belongs to the

National University of the Northeast[2] (UNNE) (Chaco), this project will make reference to the pedagogic offer of the subject "Introduction to the Technology applied to Art".

1.1 ICT and "New Media"

As a result of the investigation and the evolution of the Information and Communication Technology (ICT), today it is possible to observe the necessary bonds it has with specific disciplines as well as different expressions of art [2]. From this fact, several approaches and postures arise [3], [4], [5], [6], and [7].
These manifestations have their origin as art production through a computer and an online connection. This kind of experiences has been referred to as "new media", a term questioned for, because of the broad and indefinite meaning of the word "new".
According to Manovich [5], the popular understanding of new media identifies it with the use of the computer for the distribution and exhibition of contents, rather than for their production. The websites and e-books, which are considered new media, are an example of this, whereas ordinary books are not [7].
From this perspective, the author invites us to reflect on the following: Must this definition be accepted?

---

[1] http://www.artes.unne.edu.ar/Artes-Combinadas.html
[2] http://www.unne.edu.ar/

In relation to this, he says that "there is no reason to privilege the computer as a device for exhibition and distribution over its use as a production tool or as a device for storage", and he justifies this with two historically separated fields, such as the information and media technologies.

1.2 Divergence of Media

In 1839, Louis Daguerre presented the formal description of a new process of reproduction, called daguerreotype [8], through which it is possible to obtain a positive image from a plaque made of silver iodide.
In 1833, Charles Babbage built a device called the analytic machine, which contained the principal characteristics of the modern digital computer [9] and used punched cards to function. This machine did not prosper and as a result there exist different speculations [10] y [11].
In 1800, coincidentally, Babbage took the idea from a programmed machine. It was a loom, created by J.M. Jacquard. About that, Ada Augusta, who was the first programmer, denoted: "The analytic machine knitted algebraic patterns like the Jacquard´s loom which knitted flowers and petals" [12].
However, whereas the investigation of the daguerreotype impacted on the society immediately, the impact of the computer had not arrived yet.

Both trajectories ran parallel. During the 19th and in the beginning of the 20th century, lots of tabulators and mechanic and electric calculators were developed.
On the other hand, the success of the modern media allowed people to store images, image's sequences, sounds and texts by means of different materials such as: photographic plates, films, CDs, etcetera.

During the 1890s, the modern media put the photography into circulation. The first Edison´s cinematographic studio started to make short films of 30 seconds. Later, the Lumiére brothers showed their new hybrid camera and cinematographic projector. The same decade was crucial for informatics. Herman Hollerith consolidated the ideas from Jacquard and the analytic machine from Babbage, registering an electro mechanic information machine which used punched cards for the computer processing in the census of the Unites States.

1.3 Convergence of Medias

Different authors agree that the key decade in the media and informatics' history was the 1930s [3] and [5]; when a modern digital computer was developed. The English mathematician Alan Turing in his article about "computable numbers" provided a theoretical description of the Universal Turing Machine. It was based on reading and writing numbers on an endless tape which advanced recovering the next order, reading the information or writing the result and keeping similarity with a film projector.
A new coincidence appears: cinematograph means "written movement", that is to say, the essence of the film is to register and keep visible information in a material way. A film camera registers some information and the projector reads each piece. This device looks like the computer in an essential aspect: the program and the information from the computer also have to be kept on some storage medium.
The development of a suitable storage medium and a method to codify the information represent important parts from the prehistory of both the film and the computer. Films used discreet images, which were registered in celluloid strips; the computer adopted an electronic storage medium with a binary code.

1.4 The definitive meeting of the media

The history of the informatics and the media intertwine even more when the German engineer Konrad Zuse built the first digital computer by using punched tape to control the program. In reality, the tape consisted of useless pieces of a cinematographic film [12].
In this way, the sense and the emotion that those cinematographic sequences used to have had been cancelled by their new function as a storage device (raw material, supplies, etcetera). The iconic film code was eliminated by the

binary, which was more efficient. Therefore, the films become dependent on the computer [5].
This meeting of the media changes the identity of both the media and the computer, which stops being just a calculator, a control mechanism for a device of communication to become a media processor. All of this has an inherent explanation in the physical phenomena and laws which science presents.

1.5 Characterization of curricular area

The training of the Bachelor´s Degree in Combined Arts provides an experimentation area for the production and investigation, being the ICT a component that crosses different expressions and artistic perspectives. The career began in 2001 with an important number of students and in 2012 220 students registered.
The pedagogic challenge represented by the first year subject "Introduction to the Technology applied to Art" is to create a sphere of reflection, experimentation and integration; where it can be possible to investigate the close and problematic relations between science and arts through the use of new technologies.
To understand these emergent teaching-learning processes new multidiscipline configurations are required [13]. Therefore, one of the adopted strategies was the integration of professors with these specializations: a Bachelor´s degree in Music specialized in Multimedia Art, a Civil Engineer, an Architect and a Bachelor´s degree in Information System, who added their contributions and visions from their particular disciplines to achieve the main goal.
The students must attend classes in person and comply with 96 credit hours. There is a computer lab with a capacity of 40 students and therefore they are divided in different commissions to develop computer classes.
The specific goals of the subject are detailed below:

- Create a teaching-learning integration process in relation to the other levels or subjects from the curriculum.
- Generate inquisitiveness in the students for the development of investigations.
- Promote the training of professionals who will be able to adapt to the constant changes of the artistic labour market.

2. Methodological framework

It is based on the idea that technology is not art itself, but a tool which changed and accompanied multiple artistic expressions along the years and different periods of time, introducing theoretical and functional elements of sciences such as the quantum physics, genetics and ICT as multimedia artists, like Biopus Group, present it. [14].

For the development of the topics proposed:

- Three main axes that support the theoretical contents of the analytic programme have been delimited, denominated Thematic Axis 1 (TA1), Thematic Axis 2 (TA2) and Thematic Axis 3 (TA3).
- In and out of classroom activities for the axes have been developed.
- Evaluation criteria of the compositions made by the students in said areas are described and some works are exhibited as an example.

3. Results

The axes are hereby presented and Thematic Axis 2 (TA2) and Thematic Axis 3 (TA3) are described in more detail, with the purpose of showing the use of materials or tools (analogical o digital) and the importance of the art expression over them.

3.1 Thematic Axis 1

TA1 focus on the reflection about the new technological media, establishing the relationship between art, science and technology, by following a chronology which extends from primitive artistic expressions to contemporary, its individual development and historic convergence. Additionally, we are invited to reconsider the relations or interrelations between art, audio-visual media, digital media and ICT, promoting the reading of authors who deal with different perspectives [5].
Based on the approach or theoretical framework of the area, concepts about changing from analogical to digital are introduced, that is to say, from continuous to discreet, which are important elements to understand the digital paradigm. Moreover, optical and mechanic notions are introduced as well.

3.2 Thematic Axis 2

In **TA2** the creative process was approached, linking it to the concept of Image-Sound--Time-Space.
The link between them was shown in a practical work with its corresponding stages of production with the construction of a device (object or machine) made with analogical technology and ordinary material by starting with a discussion-triggering idea and the understanding of a selected text, a play called “Oedipus Complex”.
In relation to specific concepts, the following topics were discussed:

- Basic principles about space-time dimension
- Optical basic principles
- Resonant basic principles

3.2.1 Practical Activity TA2

For the materialization and development of the unit, we proposed the construction of a device which preceded the cinema and produces image and sound (e.g.: magical torch, zoetrope, praxinoscope, kinescope, kinetic art and resonant devices from the Italian Futurist Genre).
This experience suggests the simulation of a creative process, to which the student confronts when making an interdisciplinary work of art.
It is about a machine from the pre-cinema which was extracted from **TA1**, a Zoetrope, which produces the illusion of motion, resulting from optical and mechanic concepts. A Zoetrope, shown in **Fig.1**, consists of a cylinder with slits through which images from a set of sequenced pictures printed on the paper inside the cylinder can be seen.

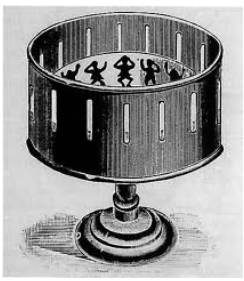


Fig.1: *A Zoetrope is considered a device or toy from the pre-cinema.*

As an example, a set of sequenced pictures (photograms) was made in the classroom (in groups). Appealing to the spontaneous creativity of the group and as extra-curricular activity, we proposed the use of the above-mentioned device and the layout of 5 sets of strips with 10 photograms, from the selection of 5 abstract concepts present in the chosen text.

3.2.2 TA2 Test

We tried to understand the creative process as an open system with different stages which accepts retro alimentation and monitoring from teachers (feedback) for the resolution of a "problem". The interdisciplinary spirit in the group and the expression were given more importance as compared to the material and its functioning. The activity was performed by starting with the discussion-triggering idea and finishing with its execution.

3.3 Thematic Axis 3

In TA3 **the incidence of the computer and software in arts** was approached by means of theoretical and practical classes [7]. From a digital point of view, the function of informatics tools, the binary code, the

codification algorithms and different types of applications were developed. In the informatics platform, and continuing with the narrative of the zoetrope, illusions of motion were presented as the origin of the cinema.
In relation to specific concepts, in this thematic axis the following topics were developed:

- Notions of digital image and sound.
- Algorithm, its characteristics and functions.
- The interaction and software tools
- Timeline, photograms, symbols, layers and actions.

3.3.1 Practical Activity TA3

Fig.2 *First images from the pre-cinema*

The text we worked with in TA2 was resumed and photograms from the zoetrope were digitalized in the platform.

3.3.2 TA3 Test

In the last practical work, the digitalized images were incorporated in an interactive collage. With this, the concept of interactivity was reinforced on a concrete informatics platform. As a result of the **TA3**, some works were obtained, such as the ones which are shown in **Fig.3, Fig.4** and **Fig.5**.

Fig. 3. *a concept represented in this collage is called "The pursuit of truth"*

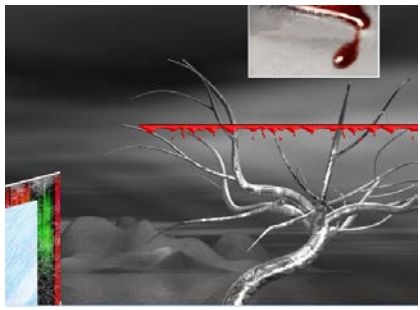


Fig.4. *A selected concept for this work is called "Life"*

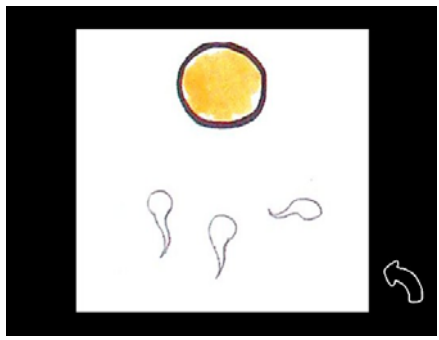


Fig.5. *The selected concept for the composition of this collage is "Destiny"*

Additionally, the linearity and non-linearity were considered in the narrative aspect, such as it is presented in [15], giving preference again to the expression of the story above the tools.

4. Conclusions and future works

The flow of the network has stimulated the dawning of new practices of communication with different grades of interactivity and possibilities of creating global and local communities. Moreover, artists have incorporated those practices to their proposals. Therefore, this experience proposes a different configuration in the use of the ICT as an unavoidable tool in a particular discipline, such as the Combined Arts, trying to stimulate practical works without "influencing" the creativity of the students.
In this way, the creativity will be explored and an integration of introductory contents will be achieved at the same time to give continuity to the rest of the curricular levels.

For this exposed case of study, the achieved results in 2013 showed the understanding of the concepts and the comprehension of goals presented by the designed activities and strategies proposed.

TA1 requires a comprehensive reading by students. This activity is considered essential for the consolidation and analysis of the relationship between science, arts and technologies.

Without any doubt whatsoever, the "invasion" of the ICT favored their easy use. Thus, the investigation of concepts related to the abstraction of ideas and problems was the difficulty appearing with more frequency in the groups. Therefore, a constant stimulation of the student's creativity and the strengthening of the retro alimentation in the creative process stages in relation to **TA2** are considered essential; which in turn will produce a better production in said axis as well as in **TA3**.

5. References

1. Resumen Descriptivo Licenciatura En Artes Combinadas. Universidad Nacional del Nordeste, http://www.artes.unne.edu.ar/documentos/Artes-Combinadas.pdf
2. Schiavo E. (2007). Investigación científica y tecnológica en el campo de las TIC: ¿conocimientos técnicos, contextuales o transversales? Rev. Iberoam. Cienc. Tecnol. Soc. v.3 n.9 Ciudad Autónoma de Buenos Aires.
3. Blanco, J., García, P. y Cherini, R. (2012). Convergencias y divergencias en la noción de computación. Rev. Iberoam. Cienc. Tecnol. Soc. [online], vol.7, n.19, pp. 111-121. ISSN 1850-0013.
4. Martínez, A. García-Beltrán, Breve historia de la informática. División de Informática Industrial ETSI Industriales – Universidad Politécnica de Madrid C/ José Gutiérrez Abascal, 2. 28006 – Madrid (España)
5 Manovich, L. (2001). The Language of New Media. http://www. manovich.net /LNM/Manovich.pdf.
6. Kurzweil R. The Age of Intelligent Machines "Chronology" http://www.calculemus.org/lect/si/dlalomzy/mchron.htm
7. Manovich L. (1999). La vanguardia como software. Departamento de Artes Visuales (Universidad de California).
8. Warner Marien M. (2010). Photography: A Cultural History. ISBN-10: 1856696669, ISBN- 13: 978-1856696661, Edition: 2.
9. Computer History Museum, http://www.computerhistory.org/babbage/
10 Babbage, H. (2010). Babbage's Calculating Engines. (New York). Edición impresa Cambridge University Press. ISBN: 1108000967, 9781108000963.
11. Giudice, J. (2010). Complejidad y dimensiones en los estudios sobre Babbage: la máquina analítica. Un análisis del fracaso cultural del primer proyecto de calculadora digital programable secuencialmente. Rev. Española de ciencia, tecnología y sociedad, y filosofía de la tecnología, ISSN 1139-3327, Nº 4, http://institucional.us.es/revistas/argumentos/4/art_1.pdf.

12. Eames, C. (1990). A Computer Perspective: Background to the Computer Age, Cambridge (Massachusetts), Harvard University Press.
13. García, R. (2006). Sistemas complejos. Conceptos, método y fundamentación epistemológica de la investigación interdisciplinaria, Barcelona, Gedisa, 200 pp. ISBN 94-9784-164-6.
14. Causa E., Romero Costas M., Rivero E., Bedoian D. Proyecto Biopus, http://www.biopus.com.ar/
15. Alé, G; Sosa, F; Verrier, F. (2004). La ruptura de la linealidad en el relato Vanguardias, Videoarte, Net Art. Facultad de Diseño y Comunicación. Universidad de Palermo. Buenos Aires, Argentina. ISSN 1668-5229.

Thanks to:
General Secretary of Science and Technology of National University of the Northeast
Faculty of Exact Sciences of National University of the Northeast
Faculty of Art, Design and Science of Culture of National University of the Northeast.

Translated by: English<> Spanish Certified Translator ALMIRON, Bárbara Antonieta
Corrected by: English<> Spanish Certified Translator MARIL, Karina

XI

Graphic Computation, Images and Visualization Workshop

Augmented Reality in Mobile Devices Applied to Public Transportation

MANUEL F. SOTO[1], MARTÍN L. LARREA[2], AND SILVIA M. CASTRO[2]

[1] Instituto de Investigaciones en Ingeniería Eléctrica (IIIE) "Alfredo Desages"
Univesidad Nacional del Sur,
Consejo Nacional de Investigaciones Científicas y Técnicas.

[2] Laboratorio de Investigación y Desarrollo en Visualización y Computación Gráfica (VyGLab),
Departamento de Ciencias e Ingeniería de la Computación,
Universidad Nacional del Sur.
{mfs,mll,smc}@cs.uns.edu.ar

Abstract. *Augmented Reality (AR) is one of the most revolutionary technologies at these times. It improves the real world by additional computer generated information. The AR paradigm opens new ways for development and innovation of different applications, where the user perceives both, virtual and real objects at the same time. With the rise of SmartPhones and the development of its characteristics, the AR on mobile devices emerging as an attractive option in this context. In this paper we present the design, implementation and testing of an application of AR on Android platform for mobile devices. This allows a person traveling through the city gets information of routes, timeouts, etc; about a particular bus line. All this information is provided on the mobile device and associated to the real world, facilitating their interpretation.*

Keywords: *Augmented Reality, OpenGL ES, Android, Public Transport, OpenStreetMap*

1. Introduction

Nowadays, technological advances in the area of mobile devices are constant. The increase in processing power, the storage, quality of cameras and screens have given rise to the development of applications of Augmented Reality (AR) on these devices. This situation is further benefit by the low cost of mobile devices, and easy Internet access from them. In this context, we designed and developed an application to assist the user that is in a certain place in the city and want to take a bus, through its mobile device the user will know if the bus route is close to its location.

In this paper we present an application of AR based in Android oriented to SmartPhones that allow the user to enrich the physical information of the environment with virtual information such as routes of bus, lines that pass within 300 meters of the place in which the user is located, the arrival times, etc.

More specifically, the system can display the bus routes over the street that the user has in his front superimposed on the video stream on his mobile device. The user can also get information about a queried bus line, an estimate of arrival times for the next bus and a view of selected bus routes. The user's position is obtained from the GPS[3], orientation from the accelerometers and gyroscopes and georeferences data are downloaded from OpenStreetMap (OSM) servers. The structure of this paper is as follows: The following section will provide an overview of the history of AR, AR on mobile devices and existing information systems relative to public transport. The third section will present the developed system architecture. Details of implementation will be provided on the fourth section. The fifth section will show the case study and finally outline the conclusions and future work are presented.

2. Background

We will introduce basic concept in references to AR, AR on mobile device and systems for visualization of maps and routes over them.

2.1 Augmented Reality and Mobile Devices

The term Augmented Reality (AR) is used to define a direct or indirect view of a real physical environment which elements are merged with virtual elements to create a real-time mixed reality. Guided by Figure 1 we can see where is placed the AR within the world of mixed reality [9].
In 1997 Ronald Azuma presented the first study of AR [4]. This publication established the physical characteristics of the AR, ergo the combination of real world and the virtual, real-time interactions and sensing in 3D.

Fig. 1. *Graphic illustration of the concept of Mixed Reality.*

[3] Global Positioning System

AR applications can be classified into two types: indoor and outdoor. While the former are used in closed environments and their goal is to work without user restrictions ([3],[6]), the latter are applications that have no environment restrictions.
Outdoor applications are based on two types of technologies: portable and immersive ([3],[6]). The first type consists in making a computer graphics overlapping on camera view of the portable device. In the second type must have generally, a Head-Mounted Display HMD that allows overlaying the images directly into the user's view, thereby achieving high levels of immersion.
In 1968 Ivan Sutherland created the first mobile AR system [14], which consisted of two trackers for correct positioning of images, each one with 6 degrees of freedom, one was ultrasonic while the other was a mechanic.
Later, in 1992, Tom Caudell and David Mizell first used the term AR [5] to refer to the computer image overlay on reality. At that time, the HMD, was the only means envisaged for mobile AR applications.
In later years there were two important developments: these were the Tobias H¨ollerer ([7],[2]) and Mathias M¨ohring ([10]). The first allowed to the user explore the story of a tourist spot through mobile device pointing it to different parts of the same spot, while the second developed a 3D tracking system for mobile devices and the screen displays information associated with AR mode.
Recently, research in this area (AR) has focused on mobile devices. In early 2000, developed projects such as Bat-Portal [11], it was based on Personal Digital Assistant (PDA) and technology Wireless. The PDA was used as a client to capture and transmit video to a dedicated server which performed the image processing and proceeded to render and compose 3D objects. While initially the prototypes were based on a distributed strategy to delegate the graphics processing, the fast advancement in mobile phones allowed the development of applications that recognize markers in the environment. Subsequently, with the integration of new sensors on devices and growth in computing processing power, the field of AR applications for mobile devices grew exponentially [10] [12] [15].
In 2007, Klein and Murray [8] presented a robust system capable of tracking in real time, using a monocular camera in a small environment. In 2008 Wikitude AR browser [1] was launched, it combines GPS and compass data entries in the Wikipedia. Finally, in 2009 White introduced SiteLens [16], an system and set of techniques for supporting site visits by visualizing relevant virtual data directly in the context of the physical site.

2.2 Maps Visualization and Routes

There are several alternatives when it comes to display maps on mobile devices. One of them is the version of GoogleMaps oriented phones, the software creates the same experience for the user as a query from GoogleMaps web page. Another alternative is Mobile Gmaps (MGMaps), it

is developed with technology J2ME, for maps obtaining the system consults sources such as Yahoo! Maps, Windows Live Local [4], Ask.com and Open Street Map (OSM), the features are similar to those of GoogleMaps.
There are also applications that use the voice as an alternative to navigation; an application of this style is Aura Navigation, which allows to the user to view route maps and move respects these using the user's voice as a guide.
Finally, some applications that use AR for display are Wikitude Drive and GPS Cyber Navi. They show a route previously configured by the user, allowing it to reach its destination in a different way.
From the literature it can be seen that there is no previous works using AR oriented to bus routes displaying on mobile devices.

3. BusWay-AR

When users are in a particular bus stop, usually they can access to the bus line that arrive to the stop; generally, there is no information about arrival times, bus routes, their location, etc. Additionally, it is useful to know what are the buslines that circulate in a certain radius close to them and where they circulate.
This motivated us to develop an application that would provide information associated with buses lines that are in a user environment.
The application developed through AR interface provides the user who is located in a certain geographical position information about: accessed bus line, the route of this and arrival times. The system can track the user and display a 2D augmentation of bus routes showing the path, if the path is round trip or return route, in addition to other information. All of this information is added to the field of view on the mobile device video stream. The information of the bus lines is obtained from the OSM servers. The application was developed on a SmartPhone equipped with camera, 3G connectivity, Wirelees, GPS, accelerometers and gyroscopes.
The system can determine the position and orientation of the user, to obtain information concerning to the bus line, routes, arrival and departure, calculate the estimated arrival times to the user's position and finally, display all this information in a graphical interface, which overlaps layers of information to the video stream of the mobile device.

3.1 System Architecture

The proposed system consists of five sub-systems: the processing of the route, the position for obtaining and calculating arrival times, the rendering, the user interface and the AR. (Figure 2)

4 MSN Virtual Earth

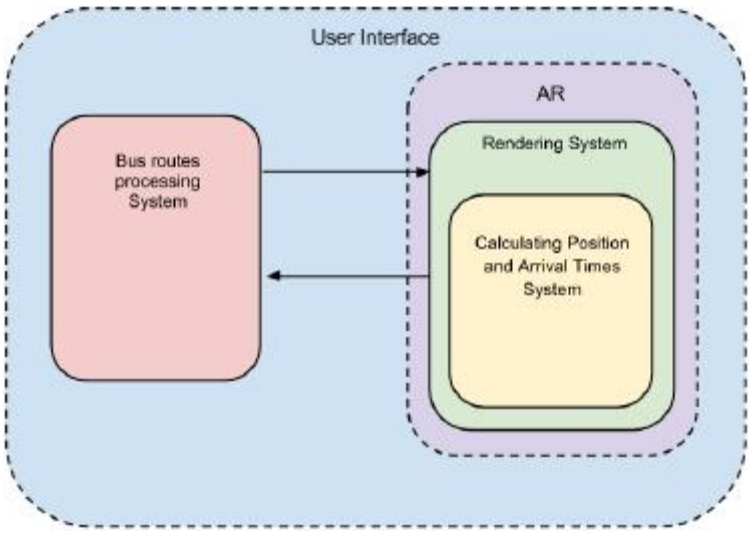


Fig. 2. *System Architecture.*

The Route Processing Subsystem is on charge of processing the path of the line selected by the user, which is communicated to the system via data provided by the graphical interface. Once the line has been obtained on request, is queried to OSM servers, the obtained information is stored in a suitable structure and is communicated to the Rendering Subsystem.

Obtaining Subsystem Calculating Position and Arrival Times is responsible for obtaining the user geographical position and the start time of the bus line consulted. From both, the subsystem proceed to calculate the arrival time and retrospective communicates to the Rendering Subsystem.

The Rendering Subsystem, which is responsible for generating the image displayed to the user, receives as input the path of the selected bus line and the user geographical position; it performs calculations for correctly positioning points of the route on the screen and also the bus position, it will be drawn only if the bus is within the range of view of the user. It is also responsible for providing information to the user, about arrival times, user geographical position, GPS status, etcetera.

The AR Subsystem is on charge of linking information from the Rendering Subsystem and the Obtaining Subsystem Calculating Position and Arrival Times, to be used for the system.

Finally the User Interface is the way by which the system communicates with the user, the latter being able to denote their needs and see the answers.

4. System Implementation

The system implementation was developed on Android and is intended to operate with bus routes of the city of Bah´ıa Blanca.
The Processing Subsystem starts to work after the user selecting a bus line, the bus line is communicated to the subsystem that is responsible for making a query to the OSM page[5] where the bus line route is stored. The results are stored in an XML file.
For the development of Positioning Subsystem, we proceeded to obtain the user geographical position. Android provides several options, one of them is to use the built-in GPS sensor on the mobile device, to use it Android provides the LocationProviders data type, which can give us the position in two different ways:
GPS-Provider and Network-Provider. We opted for the use of GPS-Provider as its accuracy was better.
Finally, the Subsystem of Calculating Arrival Times is responsible for telling the user how long it would take the next bus to arrive to the bus stop or where the bus is located, that was done by algorithms that estimate arrival times, based on data provided by the municipality of Bahia Blanca [6].
The Rendering Subsystem is responsible for drawing the scene viewed by the user. To this propose it should be taken into account the device orientation and the user interaction with the information displayed on the screen. Due to the complexity involved in the tasks outlined in the preceding paragraphs, we will see how their implementation were carried out. For obtaining the image from the camera, Android provides access to the camera frames by modifying the main configuration file. Once we get the camera preview, we had to get the device orientation. Android provides us with a set of sensors, in particular, the sensor TYPE ROTATION VECTOR gives information about accelerometer and magnetic field. In this way we obtain the orientation of the device relative to the axis of the earth (aligned to the north), which is essential given that our system will use real positions (latitudes and longitudes).
To perform rendering of objects in the AR system, we used OpenGL, this gives us an API[7] with primitive graphics for drawing simple shapes. In our case, Android provides a special version of OpenGL for mobile devices, OpenGL ES. The version used was 1.0. Since we wanted to draw on the camera frames, we had to create a scene in our OpenGL space. The scene consists in objects, routes, which are in the OpenGL world space and an associated camera which will be rotated and moved in order to observe different objects from different points of view.
Since our main goal is to draw the routes of the bus line on the camera frame and the route consist on a set of latitudes and longitudes, we must convert those latitudes and longitudes from the world coordinate system to OpenGL

5 http://wiki.openstreetmap.org/wiki/Bah´ıa Blanca/transporte publico
6 http://www.bahiablanca.gov.ar/conduce/transporte1.php
7 Application Programming Interface

coordinate system. In order to perform the conversion, we had to keep in mind that we are referring to a geodetic coordinate system (latitudes and longitudes) and a geocentric coordinate system (OpenGL), in this way we have to made the respective transformations based on data provided by the next equations:

$$1/f = \text{flattenig factor} \quad (1)$$

$$e2 = (a2 - b2)/a2 = 2f - f\,2 \quad (2)$$

$$\upsilon = a/(1\text{-}e2\text{-}\sin 2\phi)1/2 \quad (3)$$

$$X = (\upsilon + h)\cos\phi\cos\lambda \quad (4)$$

$$Y = \upsilon + h\cos\phi\sin\lambda \quad (5)$$

$$Z = [(1 - e2)\upsilon + h]\sin\phi \quad (6)$$

where h is the GPS height and the variables a and b are the length of semi-major axis and the semi-minor axis of Earth respectively. λ is the longitude and ϕ is the latitude.
From above equations (based in [13] and [17]) were transformed latitude and longitude to X, Y and Z coordinates to draw the scene in OpenGL.
Needless to say, we used a float value in the internal representation for position, latitudes and longitudes are expressed in double; the systems perform the transformation with a lost of accuracy, therefore it is possible that in some cases there is shifting between the actual data and those generated by OpenGL.

5. System Testing

For testing we proceeded to the selection of the bus line 503 of the city Bahía Blanca. Since we want to analyse the system response under different circumstances, a prototype interface was developed, the interface can select the busline manually, also the user must specify if he/she is in a default position or if the position can be get by the GPS (Figure 3).
Once the user have selected the option to show bus route, the route was displayed, both round trip (green) and return route (red), the GPS status (ON/OFF), latitude and longitude of the user geographical position, address (street/number), the arrival times from the round trip bus and the return route bus, this can be seen in Figure 4.
Data were obtained and the system was tested with both the GPS turned on and off, this can be seen in Figure 4.
It can be seen on the left of Figure 4 a map with way points recorded within a certain radius, you see green dots (round-trip), red dots (return journey) and a yellow dot(user's position). The right side of Figure 4 shows the view that the user has on the mobile device, the shifting between the bus route

and the street is due to the GPS error and the rounding data error (move from double to float).

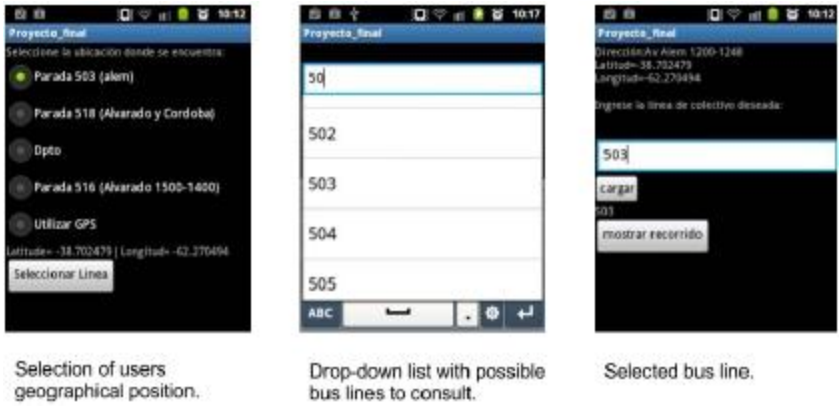


Fig. 3. *Prototype Interface.*

Fig. 4. *Caso de test del recorrido l´ınea 503.*

6. Conclusions and Future Work

Despite the technological advances in recent years, there are still difficulties, for example, in obtaining the position and orientation in large areas, the size of displays and graphics processing capabilities. However, the AR is a useful and versatile alternative to organize and contextualize the information. We have presented the design and implementation of a 2D mobile AR application

where the visualization of the route of a particular bus line is superimposed on the mobile device video stream with addition of estimated arrival times, get the user's position and display all this information contextualized in the user interface.
In the testing performed we highlight the problems generated by both GPS accuracy and the loss of precision due to transform latitude and longitude coordinates to OpenGL coordinates. These problems lead to a shift in the routes visualization. Even though the objective of obtaining different kind of information about a particular bus line, these problems must be solved yet. About the positioning, it should work better with a higher precision GPS or with DGPS8.
With respect to coordinate transformation, we should find an alternative representation in fixed point for latitude and longitude and with a defined range, perform a more accurate conversion into the OpenGL coordinate system.
In addition to seeking to solve the above problems, the future work is to be conducted online recognition of OCR, use version 2.0 of OpenGL ES and make the system has a 100% coverage information about bus lines. This paper is a starting point for the development of outdoor applications as we believe that this is a field of application in which mobile devices can be a very versatile alternative they record graphics on outdoor environments freely.

7. Acknowledgment

This work was partially funded by the project 24/N028 of Secretar´ıa General de Ciencia y Tecnolog´ıa, Universidad Nacional del Sur, PICT 2010 2657, FSTICS 001 "TEAC" and PAE 37079.

References

1. http://www.wikitude.com
2. Third International Symposium on Wearable Computers (ISWC 1999), San Francisco, California, USA, 18-19 October 1999, Proceedings. IEEE Computer Society.
3. Avery, B., Smith, R.T., Piekarski, W., Thomas, B.H. (2010). Designing outdoor mixed reality hardware systems. In: The Engineering of Mixed Reality Systems, pp. 211–231.
4. Azuma, R.T. (1997). A survey of augmented reality. Presence: Teleoperators and Virtual Environments 6(4), 355–385.
5. Caudell, T.P., Mizell, D.W. (1992). Augmented reality: an application of heads-up display technology to manual manufacturing processes. Proceedings of the TwentyFifth Hawaii International Conference on System Sciences 2, 659–669,

http:// ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=183317
6. Gotow, J.B., Zienkiewicz, K., White, J., Schmidt, D.C. (2010). Addressing challenges with augmented reality applications on smartphones. In: MOBILWARE. pp. 129–143.
7. Höllerer, T., Pavlik, J.V., Feiner, S.: Situated documentaries: Embedding multimedia presentations in the real world. In: ISWC [2], pp. 79–86
8. Klein, G., Murray, D. (2007). Parallel tracking and mapping for small ar workspaces.
   In: Proceedings of the 2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality. pp. 1–10. ISMAR '07, IEEE Computer Society, Washington, DC, USA,
   http://dx.doi.org/10.1109/ISMAR.2007.4538852
9. Milgram, P., Takemura, H., Utsumi, A., Kishino, F. (1994). Augmented reality: A class of displays on the reality-virtuality continuum. pp. 282–292.
10. M¨ohring, M., Lessig, C., Bimber, O. (2004). Video see-through ar on consumer cell-phones.
    In: Proceedings of the 3rd IEEE/ACM International Symposium on Mixed and Augmented Reality. pp. 252–253. ISMAR '04, IEEE Computer Society, Washington, DC, USA.
    http://dx.doi.org/10.1109/ISMAR.2004.63
11. Newman, J., Ingram, D., Hopper, A. (2001). Augmented reality in a wide area sentientenvironment. In: Augmented Reality, 2001. Proceedings. IEEE and ACM International Symposium on. pp. 77–86.
12. Newman, J., Schall, G., Barakonyi, I., Sch¨urzinger, A., Schmalstieg, D. (2006). Wide area tracking tools for augmented reality. In: In Advances in Pervasive Computing 2006, Vol. 207, Austrian Computer Society.
13. OGP: Coordinate conversions and transformations including formulas. OGP Publication 373-7-2 – Geomatics Guidance Note number 7 2, 131 (2012).
14. Sutherland, I.E. (1968). A head-mounted three dimensional display. In: Proceedings of the December 9-11, fall joint computer conference, part I. pp. 757–764. AFIPS'68 (Fall, part I), ACM, New York, NY, USA (1968), http://doi.acm.org/10. 1145/1476589.1476686
15. Wagner, D., Schmalstieg, D. (2003). First steps towards handheld augmented reality. pp. 127–135.
16. White, S., Feiner, S. (2009). Sitelens: situated visualization techniques for urban site visits. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. pp. 1117–1120. CHI '09, ACM, New York, NY, USA, http://doi.acm.org/10.1145/1518701.1518871
17. Wikipedia: Geodetic system-wikipedia, the free encyclopedia.
    http://en.wikipedia.org/wiki/Geodetic system (Agosto 2012)

Vertex Discard Occlusion Culling

LEANDRO R. BARBAGALLO, MATÍAS N. LEONE, RODRIGO N. GARCÍA

Proyecto de Investigación "Explotación de GPUs y Gráficos Por Computadora",
GIGC – Grupo de Investigación de Gráficos por Computadora, Departamento de Ingeniería en Sistemas de Información, UTN-FRBA, Argentina
{lbarbagallo, mleone, rgarcia}@frba.utn.edu.ar

Abstract. *Performing visibility determination in densely occluded environments is essential to avoid rendering unnecessary objects and achieve high frame rates. In this work we present an implementation of the image space Occlusion Culling algorithm done completely in GPU, avoiding the latency introduced by returning the visibility results to the CPU. Our algorithm utilizes the GPU rendering power to construct the Occlusion Map and then performs the image space visibility test by splitting the region of the screen space occludees into parallelizable blocks. Our implementation is especially applicable for low-end graphics hardware and the visibility results are accessible by GPU shaders. It can be applied with excellent results in scenes where pixel shaders alter the depth values of the pixels, without interfering with hardware Early-Z culling methods. We demonstrate the benefits and show the results of this method in real-time densely occluded scenes.*

Keywords: *Occlusion Culling, Visibility Determination, GPU, Shaders*

1. Introduction

Complex scenes with thousands of meshes and expensive shading computations are increasingly frequent in current real-time graphics applications. Although commodity hardware continues to increase its computational power every day, scenes like these cannot be directly supported at real-time frame rates. Optimization techniques are crucial in order to manage that kind of graphics complexity.

Frustum culling is a commonly used technique to avoid rendering meshes that are outside the viewing volume. These invisible models can be discarded at an early stage in the pipeline obviating expensive computation that will not contribute to the final image. Unfortunately it does not consider objects (occludees) that do not contribute to the final image because they are being blocked by others in front of them (occluders).

Because of this, several Occlusion Culling techniques were developed to overcome this limitation. Applications with expensive pixel shaders may greatly improve their performance by reducing fragments overdraw.

The Z-PrePass [1] technique avoids computing unnecessary pixel shaders following a two step procedure. First it draws the entire scene in order to store in the Z-Buffer all the depth values of the scene visible points. Second the scene is drawn once more, but this time the GPU can early reject the occluded fragments based on already present depth values in the ZBuffer. This way non visible pixel shaders are not executed.

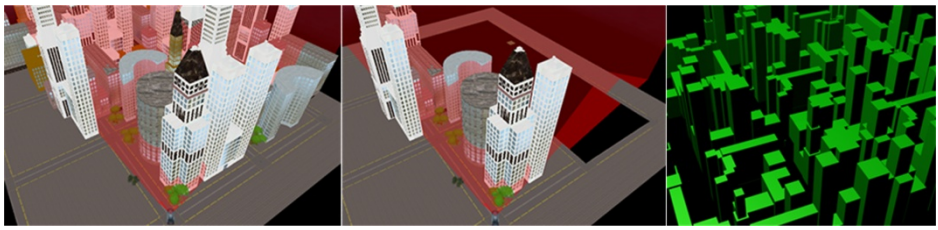


Fig. 1: Left: The densely occluded scene as viewed from the camera. Middle: The Occlusion Culling algorithm avoids rendering completely occluded objects. Right: The simplified occluder set used for occlusion.

This technique is used by many applications to reduce its pixel overdraw but its main limitation is that GPU cannot take advantage of the Early-Z [2] or [3] optimization when the pixel shader uses a depth writing operation [4], [5]. Since our method discards occluded objects before they get rasterized, no restrictions related to depth writing are imposed to pixel shaders.

Contributions: In this work we present a technique for solving Occlusion Culling in GPU, without the need for special hardware extensions or CPU read back. It includes a visibility test in the vertex shader of the application in order to discard those vertices that belong to occluded meshes. If the mesh is occluded then all its vertices can be discarded in the vertex shader, avoiding the rasterization step and the pixel computations. A previous step computes the visibility state of each mesh in the GPU and stores its result in an output texture called *Occlusion Map*. This result is acquired after performing a highly parallelized overlap and depth test comparison.

2. Related work

There is a great amount of research conducted on Occlusion Culling. A classification and overview of all these methods is presented by Cohen-Or et al. [6]. Among those techniques the ones that work in point-space are Hierarchical Z-Buffer (HZB) [7] and Hierarchical Occlusion Culling (HOM) [8].
On modern GPUs hardware occlusion queries [9] provide a built-in way to determine if a draw call contributes to the current frame, but suffer from latency and stalling effects due to the CPU read back. To address this issue temporal coherence techniques are applied [10], [11], but they require spatial hierarchies of objects to limit the number of issued queries.

Some newer hardware capabilities allow conditional rendering without CPU intervention like OpenGL conditional rendering which is implemented as GL NV conditional render [12] extension and DirectX 11 predicated rendering implemented as the ID3D11Predicate interface [13]. These methods determine whether geometry should be processed or culled depending on the results of a previous draw call. Current hardware conditional rendering does not allow the GPU shaders to access the occlusion results, but Engelhard et al. [14] implement a method that allows this. Other authors [15], [16] also implement HZB on GPU using compute shaders.
More recently Nießner [17] proposes a patch primitive based approach to perform occlusion culling applying HZB and temporal coherence. In recent years, since CPUs increased the number of cores and the set of SIMD instructions were extended, some approaches perform point based Occlusion Culling such as HOM using highly optimized software rasterizers [18], [19], [20], [21].

3. Vertex Discard Occlusion Culling

3.1 Algorithm Overview

In our proposed method we perform a *from-point*, *image-precision* [6] occlusion culling process completely in GPU without the need for the CPU to read back the results. The method consists of a series of steps that must be followed by each frame to generate the *Occlusion Map*, perform the Visibility Test and obtain the Potentially Visible Set. Finally the method uses those results, already present in GPU, to discard all the vertices of the occluded objects before they reach further stages of the pipeline. The steps are:

1. *Occludee Generation*: Select occluders and generate simplified volumes.
2. *Occlusion Map Generation*: Render occluder simplified volumes into the Occlusion Map Texture.
3. *Visibility Testing*: Determine which occludees are occluded and stored them in the Visibility Map.
4. *Vertex Discard*: Cull all the vertices that belong to invisible occludees.

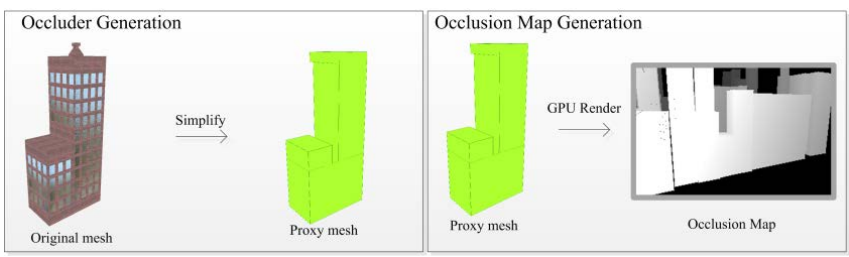


Fig. 2. First step is to obtain the simplified occluders as proxy meshes. Second step is to render all proxy meshes to the Occlusion Map texture.

3.2 Occlusion Map Generation

The method begins Offline by creating a database of selected occluders that meet a predefined criteria [22], and storing the proxy meshes which are simplified, low-poly and conservative versions of the original occluders. These simplified occluders will be rendered faster than the original meshes, even if it results more conservative. See Fig. 2.
In each frame, object-precision culling techniques such as Frustum Culling, PVS and Portal Culling [6] are applied to discard as much occluders as possible. With this obtained reduced subset of occluders we perform the first step of the method which is to render the proxy meshes into the *Occlusion Map*. This buffer stores the closest to camera depth values of every rasterized occluder and is implemented as a 32-bit floating point render target texture which is preferably a one fourth downscaled version of the screen framebuffer.
Unlike the HOM's Occlusion Map [8], our map does not contain opacity information, therefore the buffer is more similar to the HZB [7] which only stores the depth values of the occluders in each point, leaving the highest depth value to indicate no occluder presence.
The generation of the *Occlusion Map* is relatively inexpensive as the GPU massively parallel power is utilized to render the low-poly convex volumes of the proxy meshes and also because the pixel shader applied is extremely straightforward because it only outputs the depth value of each point.

3.3 Visibility Test

The core of this image based Occlusion Culling algorithm is to perform the Visibility Test for each selected occludee against the fusion of all the occluders represented by the *Occlusion Map*. Then it is used to determine whether the occludee geometry will continue along the pipeline or if it will be culled immediately. Visibility testing is performed by contrasting the points inside the occludee screen space bounding rectangle against the *Occlusion Map* depth values that contain the aggregated information of the occluders. In each frame, for every occludee in the viewing frustum, the algorithm performs a screen space projection of the occludee bounding box vertices. With those eight screen projected points, it determines the clipped 2D screen space bounding rectangle and finds the nearest from camera depth value of those extreme points. The resulting occludee bounding rectangle becomes a conservative superset of the actual pixels covered by the occludee (see Fig. 3). Afterwards, the visibility test determines if the occludee would actually contribute to the final image and starts by comparing all the depth values inside the occludee bounding rectangle against the ones in the *Occlusion Map*; when at least one point of the occludee is closer to the camera than the one stored in the same position in the *Occlusion Map*, the algorithm can now assume that such point is visible and therefore the whole occludee is considered potentially visible.
On the other hand, to determine that an occludee is completely culled, all the pixels must be examined exhaustively and proved to be farther than the values stored in the *Occlusion Map*.

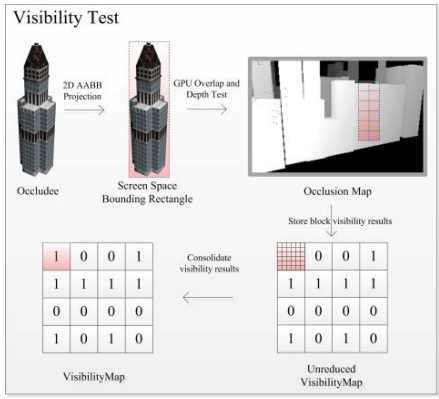


Fig. 3.*The occludees in the scene are projected in 2D and the Bounding Rectangle is calculated. For each rectangle the algorithm performs the visibility test in GPU accessing the Occlusion Map, storing the visibility result in the Visibility Map.*

Some methods implement this overlap and depth test in CPU [19], [20], [23], [21], and others use special GPU hardware capabilities such as hardware occlusion queries [9] or the more modern predicate/conditional rendering [13], [12]. Our method manually computes the visibility result pixel by pixel utilizing GPU pixel shaders.

However as explained before, to actually conclude that a occludee is culled, we have to exhaustively test all the pixels inside the occludee bounding rectangle, resulting in $N \times M$ texture fetches to the *Occlusion Map*. As the screen space regions covered by the occludees get larger, the number of texels to fetch and test can reach very large numbers.

To accelerate this, some methods build a pyramid of downsized versions of the *Occlusion Map* where each increasing level is half the size of the previous one. There are two approaches to utilize the pyramid, one is like the method used in HOM [8] and HZB [7] which they begin at some level of the pyramid depending on the occludee bounding rectangle size and have to go to the finest level to assure that the occludee is completely culled by the occluders.

The other approach [15], [16] only sticks to a selected level of the pyramid, limiting the possible number of texture fetches to a given constant to avoid the worst case scenario where they have to move to levels with greater detail. After implementing this last variation we found that the level of conservativeness was higher than expected for medium to large screen space occludees.

In this work we found that using a single level *Occlusion Map* of a fourth of the original screen buffer was a good tradeoff between number of texture fetches and level of conservativeness. In the next section we discuss the methods used to leverage the GPU hardware to perform this visibility test.

3.4 Block Subdivision

Despite having a downsized version of the *Occlusion Map*, performing all the $N \times M$ texture fetches in a single pixel shader execution does not perform as expected, because of the serial nature of the algorithm. In the best cases this inner loop could take only a few cycles whereas in other cases the same execution could take hundreds of thousands of cycles before it is finished.

For this reason, in our method the visibility test is parallelized taking advantage of the parallel execution nature of the pixel shaders, splitting the total region covered by each occludee into a series of fixed size blocks where each one only performs a maximum of 8×8 texture lookups to the *Occlusion Map* (see Fig. 4). This way each occludee bounding rectangle is split up in blocks that concurrently perform the visibility test by executing pixel shaders that return only two possible output colors: 0 meaning the block itself is completely occluded or 1 if the block is potentially visible.

The output of each pixel shader goes to a rendering target texture called *Unreduced Visibility Map* (UVM) that holds the block visibility results one next to the other as seen in Fig. 5.

In order to simplify the way each region is assigned, every occludee is assumed to have a fixed number of blocks, no matter its screen space size. In our study we determined that every occludee would have a preset number of 32×32 blocks assigned, resulting in a total of 1024 blocks. This gives us a maximum occludee screen size of 256×256 pixels and if the dimensions are larger than those, the occludee is simply considered potentially visible. To implement this algorithm using shader model 3 (without compute shaders), we carefully position a 32×32 pixel quad (GPGPU quad) and render it using a pixel shader that executes the visibility test code. Each pixel of this quad represents a block visibility test of the occluder. The shader gets the occludee bounding rectangle coordinates, depth value and the block number as parameters, and then executes the 8×8 pixels overlap and depth test.

```
Require: occludeeSize
Require: occludeePos {occludee AABB position}
Require: occludeeDepth
Require: occlusionMap
Require: pos {quad texture coordinates}
Require: quadSize
1:  base • occludeePos × pos + quadSize × 8
2:  result • 0 {not visible}
3:  for i = 0 to 8 do
4:      for j = 0 to 8 do
5:      p • base + (i, j)
6:      depth • read p from occlusionMap
7:          if occludeeDepth e depth then
8:              result • 1 {visible}
9:              break
```

```
10:                 end if
11:             end for
12: end for
```

Fig. 4. *Visibility test algorithm performed in a pixel shader*

Using this block subdivision strategy, the visibility test is split into smaller task units and performed in parallel making use of the available GPU shader execution cores. If all the blocks comprising the occludee rectangle output 0 values, then the whole occludee is considered culled, conversely when at least one of the blocks results visible the whole occludee is considered potentially visible.

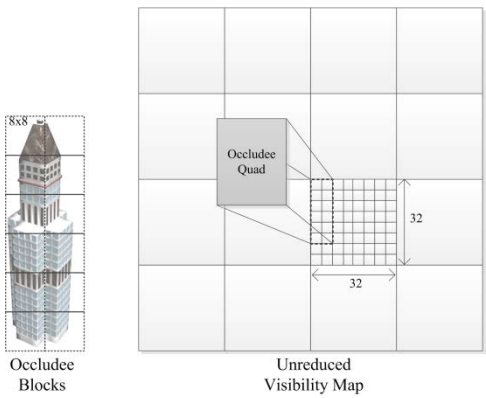


Fig. 5. *The occludee is split into 8x8 blocks, then each block performs the visibility test and stores the result into the Unreduced Visibility Map. Each occludee has an pre-assigned region of 32x32 blocks inside this Occlusion Map texture.*

Nevertheless the visibility result of each occludee is not consolidated into a single value, but spread into a series of 32×32 matrices inside some region of the UVM. The next step of our method reduces each 32×32 occludee visibility result matrix into a consolidated *Visibility Map* that will hold the results of each visibility test one next to the other.

3.5 Visibility Map Reduction

In order to reduce the UVM and consolidate each 32×32 region into a single value, we need to determine if there is at least a non-zero value inside that matrix. To achieve this, we search for the maximum value of the matrix

to see if there is any value other than zero. The search is done utilizing a parallel reduction approach with two rendering passes to limit the total number of operations. In the first pass we search the maximum value in each matrix column of 32 pixels and store it in an intermediate texture. In the second pass, we obtain the final *Visibility Map* looking for the maximum value in each row. Finally we end up with the *Visibility Map* containing the results of the occlusion culling process for each occludee tested in the current frame, which will be heavily accessed in the next step of our method.

3.6 Vertex Discard

This *Visibility Map* texture could be sent back to the CPU and processed there to avoid having to execute the draw calls to occluded objects; however this would produce a stalling effect on the GPU while sending the results back. To address this issue, we propose an asynchronous mechanism where the CPU does not need the results of the visibility test.
In our method the CPU always performs the draw calls for all the geometry that is potentially visible (the subset that passes frustum culling, portal culling, PVS, etc.), and the GPU is responsible for discarding the occluded geometry based on the *Visibility Map* content.
In our implementation we slightly modify the vertex shader that performs the *World-View-Projection* transformation as seen in Fig. 6. Before drawing an occludee, we send a parameter to the pixel shader indicating the *ID* of occludee that is about to be rendered. Based on that value, the vertex shader will perform a texture lookup in the *Visibility Map* to find the occlusion status for that particular occludee. If it is potentially visible, then the vertex shader does its usual computation letting the vertex continue throughout the pipeline. On the other hand, if the occludee is invisible we assign a negative z value to the output vertex so it can be culled by the GPU. This process is performed for every vertex that constitutes the occludee geometry.

Require: *vp* {Vertex 3D Position}
Require: *vMap* {Visibility Map}
Require: *i* {Occludee index}
1: *vis* • read visibility info from *vMap* using *i*
2: **if** *vis* = 0 **then**
3: {Continue with normal vertex shader calculations}
4: **else**
5: *vp.z* = −1 {Discard vertex}
6: **end if**

Fig. 6. *Vertex cull algorithm performed in a Vertex Shader.*

4. Implementation and Results

Our method was implemented using C# 4.0 with DirectX 9 and Shader Model 3. We decided not to use newer shader models (with Compute Shader capabilities) so we could test in the current low-end commodity hardware. The implementation of our occlusion culling module was designed in a way that can be easily adapted to other graphics frameworks, where only certain parts have to be added or modified.

We tested our method in a densely occluded 3D city scene Fig. 7, composed of 210 meshes, adding up a total of 379.664 triangles. For this scene 258 occluder proxies were generated in Offline time based on the ideas presented by [22]. In order to analyze the algorithm performance, 15 representative scene View Points were taken, where in each position we compute the following occlusion metric:

$$value = \left(\frac{t - v}{t} \right) \times 100 \tag{1}$$

Where t is the total scene meshes and v is the total visible meshes. With this metric we can determine the percentage of meshes that were discarded by the GPU in each frame due to occlusion culling (see results in Fig. 8). These values are computed with Occlusion Culling deactivated and then with it activated. We also include the frames per second that resulted from rendering the scene using a pixel shader that alters the z value to produce a displacement mapping effect with Z-PrePass and with our Occlusion Culling method. On average our method increases the FPS around 20% compared to the Z-PrePass technique (see results in Fig. 8). The values were obtained using a PC with Intel Core i3 2.40GHz processor with 2GB RAM and Intel HD Graphics 3000 GPU.

5. Conclusions and Future Work

We have implemented a method that performs image space occlusion culling completely in GPU, taking advantage of its rendering power to build the *Occlusion Map* and leveraging its parallel architecture to perform the visibility test.

According to our results, this occlusion culling method is applicable in densely occluded scenes where pixel shaders are computationally expensive and specifically if they alter the default depth value of the fragments, like in [4] and [5]. Conversely we found that for scenes with lightweight pixel shaders and no depth overrides, our method performs similar to the GPU built-in Early-Z culling, making it suitable for mixed case scenarios.

As our implementation is based on Shader Model 3, it does not require special hardware requirements, beyond the vertex shader texture lookup capabilities present in most GPUs. However we found that in some older hardware, particularly those without Unified Shader architecture, the vertex

texture lookup may downgrade the performance significantly [24]. It is also important to have some considerations before applying this technique. As all the occludees are sent to the GPU, no matter if they are occluded or not, there is a CPU-GPU bus bandwidth required to transfer the primitives to the graphics adapter. Moreover, as many other similar occlusion culling algorithms, the occluders have to be preprocessed in order to simplify the geometry into simpler conservative volumes.

Among the numerous enhancements to be made to our method, we would like to modify it to overcome the limitation of the 256×256 pixel size occludees and to explore built in hardware options to reduce the UVM, avoiding the current two rendering pass method.

Finally as newer versions of DirectX and OpenGL become available we could explore the option of implementing this method using compute shaders, orienting it to the work presented by Nießner[17] and Rákos[15]. We could also count the number of visible blocks in each occludee and utilize the results to determine some level of detail in geometry and pixel shaders.

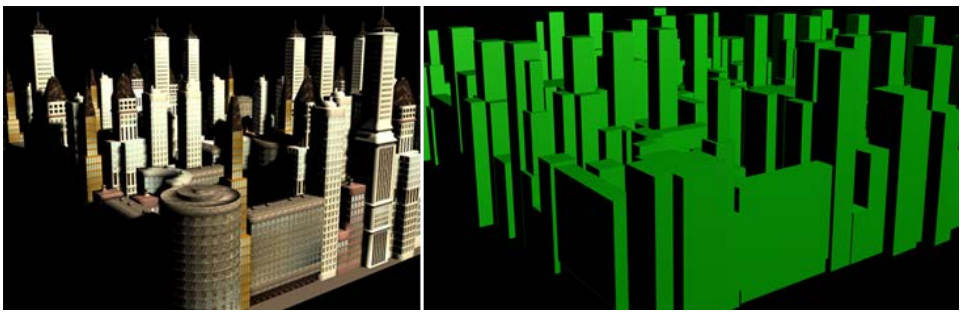


Fig. 7. *Left: The 3D city scene used to test the algorithm. Right: The simplified occluder set used for Occlusion Culling*

AKNOWLEDGMENTS

The authors would like to thank the GIGC Computer Graphics Research Group for supporting this research, and the Department of Information Systems Engineering for providing the support and funding. We also thank Retrovia Project, specially Marta Garcen and Eva Ferrari (English node) for reviewing our work and to the Algebra and Analytic Geometry node to make this contact possible.

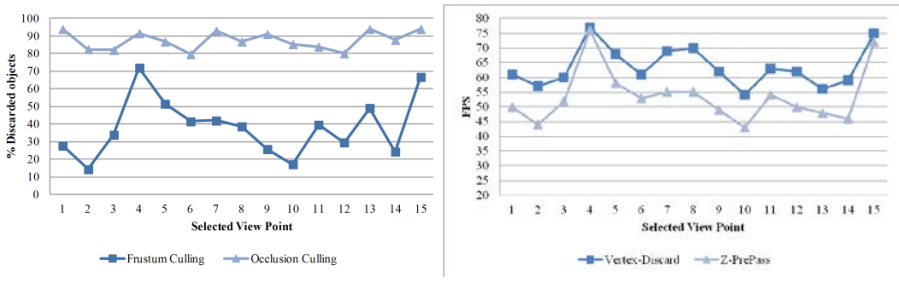


Fig. 8. *Left: Discarded mesh percent, first with only Frustum Culling and then activating Occlusion Culling, at the fifteen different selected view points. Right: FPS rendering performance with Z-PrePass and then with Vertex-Discard Occlusion Culling activated, at the 15 different selected view points.*

References

1. Intel Corporation (2013). "Early Z Rejection", http://software.intel.com/en-us/vcsource/samples/early-z-rejection, Accessed June.
2. Haines, E. and Worley, S. (1996). "Fast, low memory z-buffering when performing medium-quality rendering," J. Graph. Tools, vol. 1, no. 3, pp. 1–6.
3. Riguer, G. (2002). "Performance optimization techniques for ati graphics hardware with directx 9.0", ATI Technologies Inc.
4. Krishnamurthy, V. and Levoy, M. (1996). "Fitting smooth surfaces to dense polygon meshes", in Proceedings of the 23rd annual conference on computer graphics and interactive techniques, ser. SIGGRAPH '96. New York, NY, USA: ACM, pp. 313–324.
5. Lee, A., Moreton, H. and Hoppe, H. (2000). "Displaced subdivision surfaces", in Proceedings of the 27th annual conference on Computer graphics and interactive techniques, ser. SIGGRAPH'00. New York, NY, USA: ACM Press/Addison-Wesley Publishing Co., pp. 85–94.
6. Cohen-Or, D., Chrysanthou, Y., Silva, C. and Durand, F. (2003). "A survey of visibility for walkthrough applications", Visualization and Computer Graphics, IEEE Transactions on Visualization and Computer Graphics, vol. 9, pp. 412–431.
7. Greene, N., Kass, M. and Miller, G. (1993). "Hierarchical z-buffer visibility", Anaheim, CA, pp. 231–238.
8. Zhang, H., Manocha, D., Hudson, T. and Hoff, K. (1997). "Visibility culling using hierarchical occlusion maps". Los Angeles, CA: In Computer Graphics (Proceedings of SIGGRAPH 97), pp. 77–88.
9. NVIDIA Corporation (2013). "Nv occlusion query," http://www.opengl.org/registry/specs/NV/occlusion query.txt, Accessed Mar.

10. Hillesland, K., Salomon, B., Lastra, A. and Manocha, D. “Fast and simple occlusion culling using hardware-based depth queries”, Techinical Report TR02-039, Dept. Comp. Sci., University of North Carolina, 2002.
11. D. Staneker, D. Bartz, and M. Meissner, “Improving occlusion query efficiency with occupancy maps,” in Proceedings of the 2003 IEEE Symposium on Parallel and Large-Data Visualization and Graphics, ser. PVG ’03. Washington, DC, USA: IEEE Computer Society, p. 15, (2003)
12. NVIDIA Corporation (2013). “Nv conditional render,” http://www. opengl.org/, registry/specs/NV/conditional render.txt, Accessed Mar.
13. Microsoft (2013). “ID3D11Predicate interface,” http://msdn.microsoft.com/en-us/library/windows/desktop/ff476577%28v=vs.85%29.aspx, Accessed Mar.
14. Engelhardt, T. and Dachsbacher, C. (2009). “Granular visibility queries on the gpu,” Boston, pp. 161–167.
15. R´akos, D. (2013). “Hierarchical-z map based occlusion culling,” http://rastergrid.com/blog/2010/10/hierarchical-z-map-based-occlusion-culling/.
16. Darnell, N. (2013). “Hierarchical z-buffer occlusion culling”, http:// www.nickdarnell.com/2010/06/hierarchical-z-buffer-occlusion-culling/, Accessed Mar.
17. Nießner, M. and Loop, C. (2012). “Patch-based occlusion culling for hardware tessellation,” in Computer Graphics International.
18. Vale, W. (2011). “Practical occlusion culling in killzone 3”, p. 49.
19. Andersson, J. (2009). “Parallel graphics in frostbite-current & future,” SIGGRAPH Course: Beyond Programmable Shading.
20. Intel Corporation (2013). “Software occlusion culling”, http:// software.intel.com/en-us/articles/software-occlusion-culling/, Accessed Jan.
21. Barbagallo, L. R., Leone, M. N., Banquiero, M. M., Agromayor, D. and Bursztyn, A. (2012). “Techniques for an image based occlusion culling engine”, in XVIII Argentine Congress on Computer Sciences, ser. CACIC 2012, Bahía Blanca, pp. 405–415.
22. Leone, M. N., Barbagallo, L. R., Banquiero, M., Agromayor, D. and Bursztyn, A. (2012). “Implementing software occlusion culling for real-time applications”, in XVIII Argentine Congress on Computer Sciences, ser. CACIC 2012, Bahía Blanca, pp. 416–426.
23. Hey, H., Tobler, R. F. and Purgathofer, W. (2001). “Real-time occlusion culling with a lazy occlusion grid”. London, UK, UK: Springer-Verlag, pp. 217–222.
24. NVIDIA Corporation (2008). “Geforce 8 and 9 series gpu programming guide”, http://developer.download.nvidia.com/GPU Programming Guide /GPU Programming Guide G80.pdf, Accessed Mar.

X

Software Engineering Workshop

Reengineering a Software Product Line: A Case Study in the Marine Ecology Subdomain

NATALIA HUENCHUMAN[1], AGUSTINA BUCCELLA[12], ALEJANDRA CECHICH[1], MATÍAS POL'LA[12], MARÍA DEL SOCORRO DOLDAN[3], ENRIQUE MORSAN[3] AND MAXIMILIANO ARIAS[1]

[1] GIISCO Research Group
Departamento de Ingeniería de Sistemas – Facultad de Informática
Universidad Nacional del Comahue, Neuquén, Argentina
natalia.huenchuman@gmail.com,
{agustina.buccella,alejandra.cechich,matias.polla}@fi.uncoma.edu.ar,
ariasmaxi89@gmail.com

[2] Consejo Nacional de Investigaciones Científicas y Técnicas – CONICET
[3] Instituto de Biología Marina y Pesquera "Almirante Storni"
Universidad Nacional del Comahue – Ministerio de Producción de Rio Negro
San Antonio Oeste, Argentina
{msdoldan,qmorsan}@gmail.com

Abstract. *Software product line and component-based software engineerings follow similar objectives, minimizing costs and effort on the development of new systems by means of reuse techniques as the main tool. Although they have different basis as to the way of carrying out a software development, both engineerings can be combined to improve costs and implement effective quality systems in less time. In this work, we present a reengineering process applied to a product line created for the marine ecology subdomain. This process is strictly oriented to reusable components by using open source tools and following the ISO and OGC standards.*

Keywords: *Software Reengineering, Software Product Lines, Geographic Information Systems, Open Source Tools.*

1. Introduction

Component-Based Software Engineering (CBSE) [7,17] provides methodologies, techniques and tools based on the use and assembly of pre-manufactured parts (developed at different times, by different people with different goals and uses) that may be part of new systems to develop. The main goal of this engineering is to minimize the cost and time of the development of systems, without a detriment of the quality of them. Although the software component technology has started a long time ago, approximately during the 60 decade [12], there are still aspects that are being analyzed to obtain tangible benefits for developing new systems. At the same time, new paradigms have emerged involving similar objectives than CBSE, but focusing on different aspects when building systems. One of these widely

used paradigms, is the Software Product Line Engineering (SPLE) [3,5,15,18] which provides mechanisms for defining common assets together with controlled variability within a particular domain. The main differences between both approaches, product lines and components, lie in two main aspects. First, in the software product lines, the assets are explicitly designed for reuse, ie, they are created with a goal of predictability against opportunism of the component development. Second, the product lines are managed as a whole, instead of components which are designed and maintained separately[1]. However, although the product lines approach does not explicitly require the development of software components, the combined use of these paradigms helps reduce coupling, increase cohesion, and improve modularity and evolution of developed systems [1,2].

In previous works [13,14] we developed a software product line on the marine ecology subdomain in which we defined a platform of common services and their variabilities. The previous SPL was designed by following a methodology which combines advantages of several methodologies widely referenced in academy and industry [3,6,10,15] and extended in order to apply a domain-oriented approach. Although the SPL was used to develop a first product within the domain (created for the Institute of Marine Ecology and Fishery "Almirante Storni"[2]), there were several problems when we had to introduce changes or when new products had to be developed for other organizations. These problems included aspects such as the code highly coupled and complex application of reuse techniques, complex instantiation of variabilities, etc. Therefore, in this work, in order to solve these problems, we propose to redesign the SPL by applying techniques that enable the software components development in order to achieve an effective reuse within the geographical domain. The reengineering process generates changes not only in the underlying technology but also in the development approach. Thus, the process includes coding, because the components are rewritten in another language, and forward engineering, because the design of the line is changed in order to modify some previous services and add new ones.

Regarding to the reengineering process, there exist several works proposing novel methodologies or case studies to restructure software product lines towards a software component approach [5,9,11,16,19]. For example, the methods MAP (Mining Architectures for Product Lines) [20] and OAR (Options Analysis for Reengineering) [16], developed by the Software Engineering Institute[3], are based on mining existing assets of legacy products for reusing them in a product line approach. Both methods can be used together for supporting the reuse of architectures or independent components to be fit in specific architecture constraints[4]. At the same time, other two contemporary methodologies, the PuLSE (Product Line Software

1. A Framework for Software Product Line Practice, Version 5.0. http://www.sei.cmu.edu/productlines/frame_report/pl_is_not.htm
2. http://ibmpas.org/
3. http://www.sei.cmu.edu
4. http://www.sei.cmu.edu/productlines/frame_report/miningeas.htm

Engineering) [21] and RE-PLACE (Reengineering Enabled Product Line Architecture Creation and Evolution) [22], support the reuse of existing assets by proposing specific reengineering activities included in the methodologies. PuLSE is a more general methodology which can be also used for a completely new SPL development (without reusing legacy products). On the contrary, the RE-PLACE method is specific for product line reengineering addressing activities to identify, model, design, and make the transition of a legacy product to an SPL approach. During the last few years, new proposals have emerged focusing on different branches for software product line reengineering. A systematic review of these branches together with particularities of recent works on each of them was presented by Laguna & Crespo [11]. At the same time, in [23] authors extend this review for generating a taxonomy based on three main dimensions of the SPL reengineering process, quality, SPL implementation, and legacy migration. In this work, we propose a reengineering process which combines some of these proposals and adapt them in order to follow a domain-oriented approach.

This paper is organized as follows. The next section presents the reengineering process describing the activities involved in the analysis, design, and development of reusable components. The process is described through the illustration of a case study during the reengineering of a previous SPL in the marine ecology subdomain. Future work and conclusions are discussed afterwards.

2. Reengineering a Software Product Line

In order to perform the reengineering of the SPL we combined the OAR method, presented in [16], together with our domain-oriented methodology described in [4,14]. The OAR method was selected mainly based on its application experiences [5,16] and the feasibility of its application on a legacy SPL. In Figure 1 we show the seven activities of our reengineering process in which we can see the influence of geographic standards to perform some of them. In particular, we use a service taxonomy, containing the standard services defined in the geographic domain and particularly the marine ecology subdomain [4]. This taxonomy is a specialization of the ISO 19119 standard[5] in which the geographic services are classified.

5. Geographic information. Services International Standard 19119, ISO/IEC, 2005.

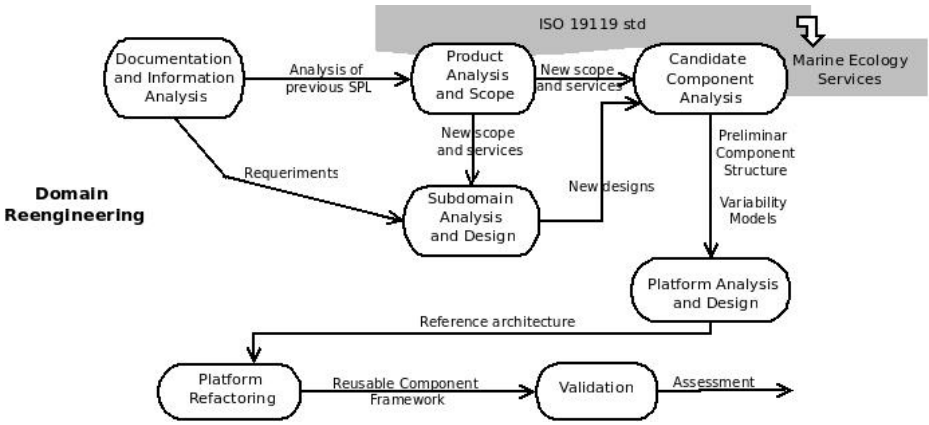


Fig. 1. *Activities of the reengineering process*

Next, we describe each of these activities, together with the works performed on our previous SPL:

- *Documentation and Information Analysis*: In this activity, the first task is to conform the reengineering team including all important stakeholders in the process. This team will be the responsible of delimiting the domain by reviewing all the information provided. This information includes the documentation of the previous SPL and the service taxonomy. Then, with respect to the service taxonomy, the team must analyze the services by understanding the functionality they offer and the interaction among them. The team can also suggests new services (which are not specified) as necessary for the domain.

  To perform these tasks, we firstly conformed the reengineering team which was a multidisciplinary team involving expert users (biologists in general) and informatics (software engineers and developers). Then, the marine ecology domain was analyzed by using the information provided by experts and the documentation provided. With this information was possible to analyze the previous requirements against the new ones resulting in the addition of two new features, the execution of some spatial queries and the generation of statistical data. These requirements were added as new standard services in the service taxonomy.

- *Product Analysis and Scope:* This activity must analyze not only the domain but also the new SPL platform. The team must reanalyze the scope of the platform by considering the new requirements specified in the previous activity and define which of them will be part of the derived products. At the same time, an exhaustive study of implementation issues within the geographic domain and the possibilities of migrating to software components must be fully

analyzed. It is necessary an exhaustive study of new and previous geographic software tools (used in the previous implementation) in order to determine which of them will be applied in the new implementation. The selection of the tools must be based on the particularities of the open source software and the restrictions defined by the service taxonomy and geographic standards.

In our work, the previous SPL was analyzed due to problems emerged when we had to reuse services. Its highly coupled design generated more time and effort when the line was instantiated to create new products. In general, these problems were related to the underlying technology used to develop the SPL platform. The architecture of the previous SPL was developed at modular level by using open libraries and tools for manipulating geographic data. They were: PostGIS[6], for the creation of spatial database; Geoserver[7], as map server to publish data from the main sources of spatial data using open standards; and OpenLayers[8], as lightweight web client, mainly selected because of the ease access to the configuration and adaptation of the source code.

Although many of these tools are widely used in current applications, they are not suitable to construct independent developments that increase the modifiability and evolution capabilities. In our case, these tools allowed developers to implement the components of the previous SPL, but in a partial compliance with the architectural constraints. Thus, in the previous SPL a modular structure was developed which generated coupled code, hard interface separation, etc. Moreover, the most important drawback was that all business logic had to be included in the user interface of each module generating a high coupling in the code in detriment of an effective reuse of the platform services.

The analysis of the tools used in the previous SPL and the selection of the new geographic tools were guided by their viability to implement components which support the development of the architecture defined by the geographic standards. This architecture is composed of a *human interaction layer*, which is responsible for the interaction with the user; a *user processing layer*, which coordinates and implements the functionality required by the user; and an *information management layer* responsible for modeling, storing, and managing the geographic data storage. The main advantage of this architecture is the separation of the functionality into three separate layers that interact through their well-defined interfaces. Considering then this architecture, and after a thorough study of the possible open source tools available on the Web [8], we chose five of them for implementing the requirements of each layer of the architecture. Figure 2 shows these tools within the layer they are involved.

6. http://postgis.refractions.net/
7. http://geoserver.org/display/GEOS/Welcome
8. http://openlayers.org/

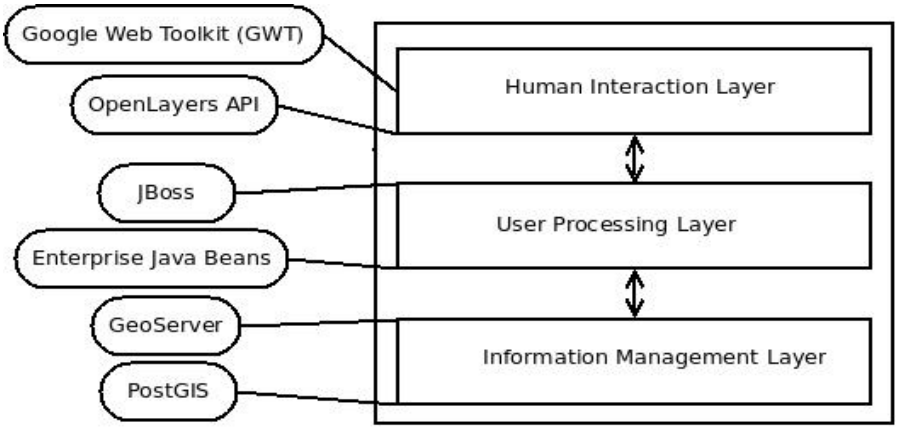


Fig. 2. *Open source tools selected for implemented the architectural layers*

As we can see, for the *information management layer*, we decided to continue with the same spatial database system, PostGIS, due to it has a number of advantages that position it first among the options of this type of software. Among them, we can say that it is free software, GNU General Public License (GPL) licensed, OGC standard compliance, supports spatial types, spatial indexes, and so on. Within the same layer, with respect to geographic server, we also decided to maintain the same, Geoserver, mainly due to maintainability and its easy management setting through a friendly web interface. After selecting the initial tools, we analyzed software tools for the component development, which are implemented as part of the *user processing layer*. Among the various existing tools, we selected the Enterprise Java Beans (EJB) technology[9] because of its extensive use and its simplified process of creating components. This allow programmers to abstract from the general problems of an application (concurrency, transactions, persistence, security, etc..), to focus on developing business logic itself. Then, for the deployment of EJBs was selected the JBoss server[10].

The main challenge was then the analysis of software tools for creating the web interface, ie, web clients for geographic services (corresponding to the *human interaction layer*). There exists many of these tools, each of them with different characteristics. For example, some of them use only client side technology while the others are dependent on server-side functionality. This last set of

9. Oracle, http://www.oracle.com/technetwork/java/javaee/ejb/index.html
10. http://www.jboss.org

tools allow the execution of tasks such as advanced security, user and groups management, spatial analysis and customization of controls and features of graphical user interfaces, among others. Fortunately, the OGC has promoted the use of standards for web mapping services that have helped to establish a common framework for accessing geographic information on the Internet (Web Map Service, Web Feature Service, Web Coverage Service), presenting through styles (Style Layer Descriptor), filtering (Filter encoding), storing, transporting (Geography Markup Language and Keyhole Markup Language) and processing (Web Processing Service). In addition, many existing web clients have dependencies among them, some have disappeared and others are taken as the basis for new developments. Figure 3 shows the dependencies among web GIS clients extracted from OSGeo[11]. In our analysis, we used these set of tools to classify them according to three main aspects: which of them are officially abandoned projects (marked with a red triangle), which do not have a recent version (identified with yellow triangle), that is, with more than a year without a new version, and which of them can be considered as valid today (indicated by a green check mark).

Many of the tools showed in Figure 3 are implemented according two paradigms: *UMN MapServer*[12] and *OpenLayers*. The clients using *UMN MapServer* were created considering the particularities the client offers such as map, scale, reference map, navigation tools, spatial objects identification, etc. Also, it has a Application Programming Interface (API) called *MapScript*[13], which has been implemented in different programming languages. At the same time, some clients optionally use *UMN MapServer* by means of *MapScript*, such as *AppForMap*[14] and *GeoMOOSE*[15].

The new client generation of tools use OpenLayers due to their optimal performance when layers and maps must be visualized by a web interface. Different enterprises contribute to its development and several projects, as *MapBuilder*[16], have finalized to accelerate this process. Also, there are other clients which allow to render maps with this paradigm, such as *i3Geo*[17] and *Flamingo*[18].

Finally, there are clients which are not based on other ones, that is, they were independently originated. Examples of them are *Geomajas*[19], *iGeoPortal*[20] and *Mapbender*[21]. As most of these APIs

11. http://wiki.osgeo.org/wiki/Comparacion_de_clientes_ligeros_web_para_SIG/
12. http://mapserver.org/
13. http://mapserver.org/mapscript/index.html
14. http://appformap.mapuse.net/
15. http://www.geomoose.org/
16. http://www.mapbuilder.net
17. http://www.gvsig.org/web/projects/i3Geo
18. http://www.flamingo-mc.org
19. http://www.geomajas.org
20. http://wiki.deegree.org/deegreeWiki/iGeoPortal
21. http://www.mapbender.org/

for thin clients are written in JavaScript, in a first approximation we analyzed to encapsulate the code by using *OpenLayers* and *GeoExt* in servlets. However, this method still had the disadvantages of the language (which is not completely object-oriented and it is not a typed language). Therefore, thin clients were discarded and we began analyzing different development frameworks.

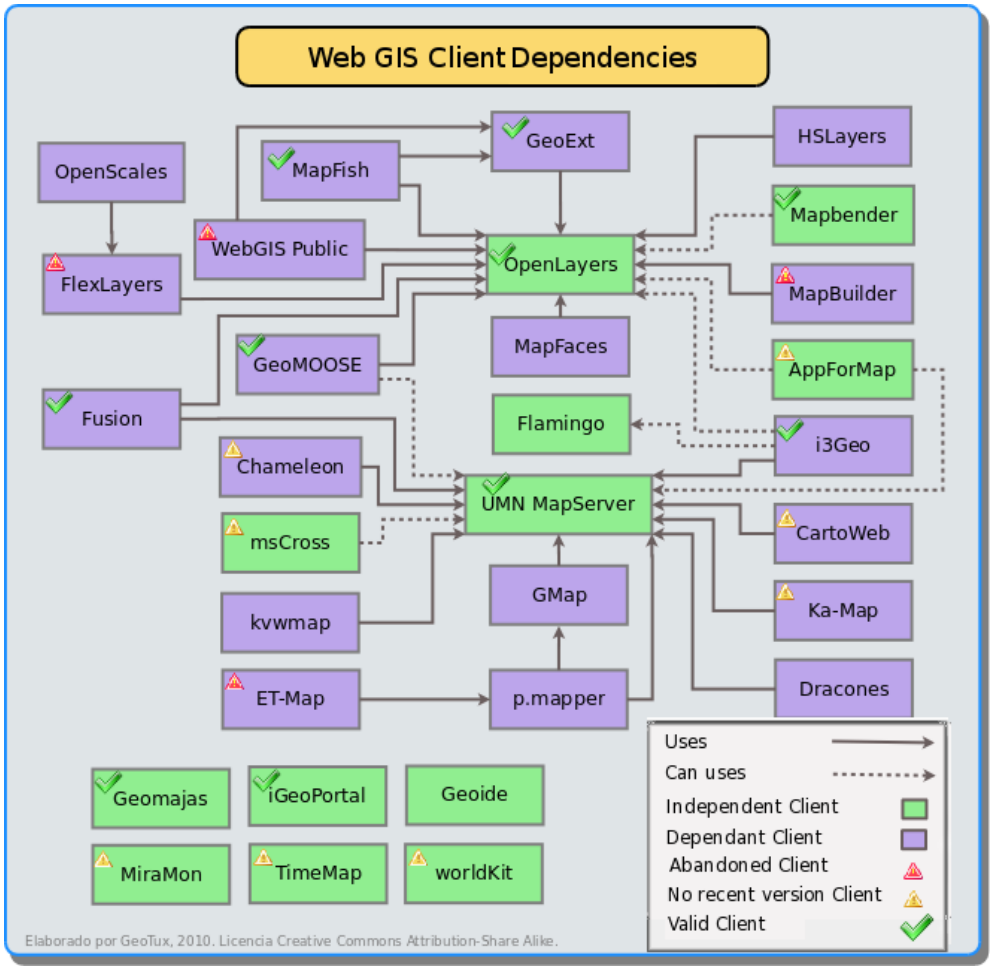


Fig. 3. *Web GIS client dependencies together with their validities*

Among them, we analyzed the framework developed by Google, called *Google Web Toolkit* (GWT)[22], which facilitates the creation of web components in JavaScript. As described above, the development

22. http://code.google.com/intl/es-ES/webtoolkit/

of JavaScript components is often a tedious process as each browser has its own peculiarities making the application programmers validate the operations on each of them. Therefore, *GWT* was seen as a solution due to it allows programmers to implement their web applications in Java, and then it return the equivalent code in JavaScript and HTML. *GWT* is responsible for generated JavaScript code to work properly on different browsers. Moreover, GWT is bound to several projects, including one related to *OpenLayers*, called *GWT-OpenLayers*[23]. This is a Java wrapper for *OpenLayers JavaScript API* that allows *GWT* projects to be used as a library (.jar).
Thus, after extensive testing and analysis tasks, we could finally get the combination of tools to fulfill our architectural constraints and implement the component-based structure (Figure 2). On one hand, we continued using *OpenLayers*, chosen by their easiness, efficiency and use, and on the other hand, we could develop web components using the Java language. Another important advantage of choosing these tools was that *GWT* supports remote method invocation (RPC) enabling a perfect integration with the *EJB* technology.

- *Sudomain Analysis and Design*: The information obtained in the previous activity is used to analyze and organize the functions or services the subdomain must provide together with the tools available to perform them. The domain modeling is needed to find commonalities and variabilities of the range of applications to be derived from the line. As the domain refers specifically to analyze the geographical domain, we should apply the guidelines defined in the ISO 19109[24] standard by using spatial types defined in the ISO 19107[25] standard.
  In this activity we created the final conceptual model according to previous and new requirements obtained in the last activities. This model was made by using the guidelines proposed in the standards.

- *Candidate Component Analysis*: In this activity, by considering the new requirements and the previous components, the new set of components must be defined. This new component structure proposed must consider evolution and modifiability aspects. Also, the variability must be considered and reanalyzed. Thus, the outputs of this activity are a complete reusable component definition together with the variability models involved.
  Here, we performed a module inventory by considering how the specification and implementation was made in the previous SPL. In Table 1 we can see a simplified list of the modules inventoried together with the services provided by each of them.

23. http://www.gwt-openlayers.org/
24. Geographic information. Rules for Application Schema. Draft International Standard 19109, ISO / IEC, 2005.
25. Geographic information. Spatial Schema. International Standard 19107, ISO / IEC, 2003

From this inventory and the new requirements of the subdomain, we performed an analysis to obtain the component candidates. To do so, we used the list of services generated in previous works [4] and derived from the ISO 19119 standard for the marine ecology subdomain. At the same time, we added new services to this list in which we defined the new requirements.

Modules	**Services implemented**
Graphical interface	Show/hide layers – Zoom tools – Scale
Geographic features management	Show and query data about geographic features. Show, query and edit data about geographic features graphically.
Change detection	Find differences among data of the same type into a specific geographic area in different moments.
Proximity analysis	Obtain geographic features around an specific area
Geographic Statistics	Generate statistics of geographic features
Feature Access	Access to geographic features – Manage data of geographic features
Map Access	Access to georeferenced maps

Table 1. Module inventory

After that, we create the new reference architecture for the SPL based on the three layers (Figure 2). In Table 2 we show a simplified list of component candidates together with the services implemented on each of them and grouped according to the architectural layers.

As we can see in this new component structure, we modified and distributed the services of each module in order to improve this structure. For example, in the *human interaction layer*, the *geographic features management* module contained more services including visualization and edition capabilities. In order to maximize the modifiability and evolution, we decided to separate these services into three new components. They are, *geographic viewer* to show the whole geographic space and show/hide layers, *visualization features* to show several types of attributes, and *geographic manipulation tools* to perform geometric figures on the map such as points, polygons, and lines.

In the *user processing* layer we performed some modifications and added new components according to the previous and new requirements. At the same time, some modules were designed as components and remained with the same functionality.

Components	Description	Architectural layer
Geographic viewer	Show the whole geographic space. Show/hide layers	Human Interaction
Attribute viewer	Show attributes of geographic features. Show attributes of specific layers. Show additional data	Human Interaction
Visualization features	Zoom functionality, pan tools, scale, refresh.	Human Interaction
Geographic manipulation tools	Tools for performing geometric figures on the map such as points, polygons, lines, etc.	Human Interaction
Graphical interface	Enter data, visualization of different functions	Human Interaction
Proximity analysis	Obtain geographic features around an specific area	User Processing
Change detection	Find differences among data of the same type into a specific geographic area in different moments	User Processing
Graphical statistics	Generate statistics of geographic features	User Processing
Access to geographic features	Perform queries to a geographic repository – Manage data about geographic features	Information management
Map access	Access to georeferenced maps	Information management

Table 2. Component candidates of the new SPL

For example, the *proximity analysis* component was defined as the module with the same name. Also the same happened with the *change detection* component which implements the same functionality as before. However, the *graphical statistics* component was redefined completely in order to interact with other components which query the database. Finally, in the *information management* layer, the components were the same as before, but they were reimplemented with the new set of tools.

- *Platform Analysis and Design*: This activity must design the final reference architecture based on the components defined in the previous activity. The reusable component structure is refined in order to take final decisions on the variability designs and future implementations. In addition, the components and the reference architecture are reorganized by considering functional and non-functional (quality) needs.
  In our work, we created the final reference architecture based on the component structure and the layers defined. At the same time, due to

the new requirements emerged from the restructuring process, we designed and developed two new components, *Calculus Map* and *Graphical Statistics*. Both implement services for the execution of some spatial queries and the generation of statistical data. For example, the *graphical statistics* component displays some species data in histograms in order to analyze and compare the results of the collection of samples in the different zones and surveys. Figure 4 shows the sequence diagram for modeling the interaction between the objects to perform this functionality. As we can see, from a survey, a year, a zone and a specie, the component obtains the data sizes of individuals (measured in mm). With this information, the data are displayed in a histogram. In this component we can also observe a variability point in order to visualize the results through *histograms* (as continuous lines) or in *bar graphs*.

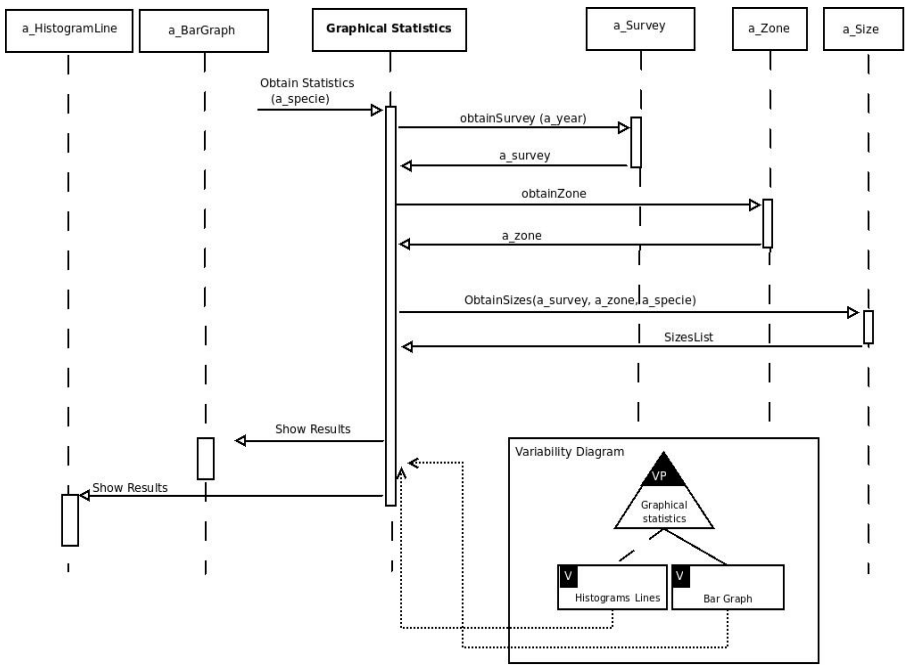


Fig. 4. *Sequence diagram of the Graphical Statistics component*

- *Platform Refactoring:* In this activity all the components of the new SPL platform must be completely rewritten by using the set of tools selected.
  In our work, we reimplemented all the components according to the functionality specified on each of them. For example, in Figure 5 we can see the user interface created by the *graphical statistics*

component showing the result in a histogram. The main goal of this component is to show in a simple and fast way the range of sizes in which species have been found. For example, in the figure we can see that the majority of the individuals found sizes between 60 and 120 mm in the CCII 87 survey of the *Viera Tehuelche* specie.

- *Validation*: In this activity is necessary a regression testing [24] in order to verify the previous functionality is still satisfied and the new one is correct.
  In our work, we performed some tests in order to analyze the new functionality implemented. At the same time, the new SPL was used in order to instantiate two products, one for the IBMPAS and the other for the CENPAT-CONICET[26]. The prototypes of both products are available in http://gissrv.fi.uncoma.edu.ar/SaoProjectUI and http://gissrv.fi.uncoma.edu.ar/CenpatProjectUI, respectively.

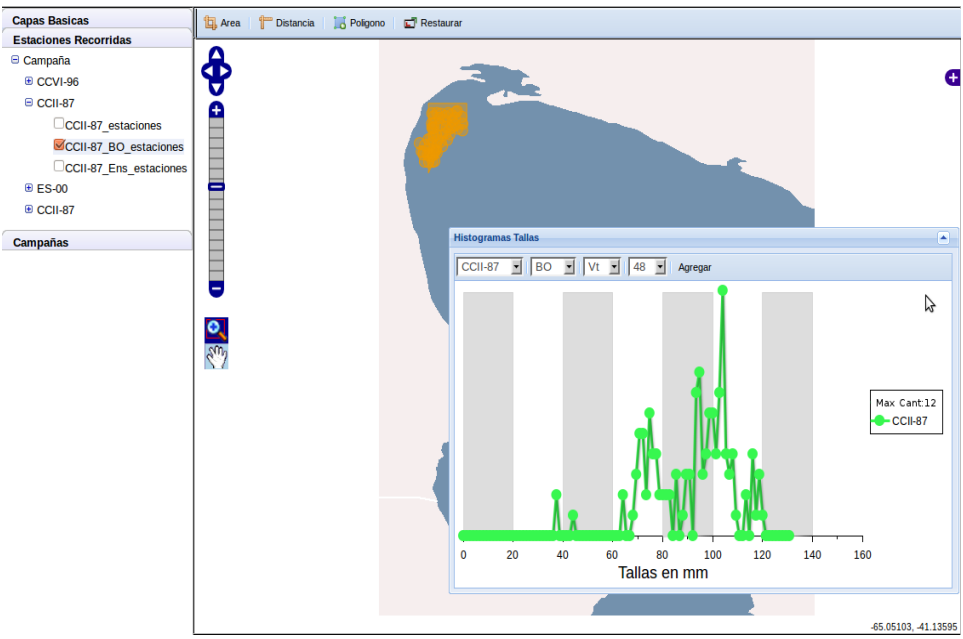


Fig. 5. *User interface with the result of the execution of the graphical statistics component.*

26 http://www.cenpat.edu.ar/

3. Conclusions and Future Work

In this article we have described the reengineering process of an SPL oriented to create reusable software components within a layer-based architecture. At the same time, the components have been developed according to the particularities of the geographic domain and supported by ISO and OGC standards. The process was illustrated by a case study involving a previous SPL in the marine ecology subdomain.

The use of the standards allowed us to delimit the range of services that the domain should offer making possible to be expanded to other geographic subdomains. The reengineering process also allowed to reuse the defined services minimizing the time and effort in early stages. Next, the case study showed several benefits derived from the use of such standards, the specification of components and the influence on the selection of open source tools for the implementation. In this work, the selection of these tools was one of the most complex and long tasks due to we had not only to contemplate that they were suitable for the development of the components according to the architecture defined in the ISO 19119 standard, but also had continuity in their developments, good documentation, active forums, flexibility to be extended, etc.

The final result of our work is a set of highly cohesive and loosely coupled components that maximize modifiability, evolution and specifically reuse. This was achieved because components are available to be treated as separate units which are then assembled to create an SPL platform. Thus, the time and effort in the creation of new products can be decreased due to the simplicity in the use and assembly of the components of the line.

As future work, we must validate the reengineering process by applying it to other SPL reengineering projects. At the same time, we should measure its benefits by applying reuse indicators which show qualitative and quantitative levels of the reuse achieved so much in the creation of a component-based SPL as in the creation of new products derived from it.

References

1. Amin, F., Mahmood, A. K. and Oxley, A. (2011). Reusability Assessment of Open Source Components for Software Product Lines. International Journal of New Computer Architectures and their Applications (IJNCAA), 3(1).
2. Atkinson, C., Bayer, J. and Muthig, D. (2000). Component-based product line development: The kobra approach. 576: 289-309.
3. Bosch, J. (2000). Design and use of software architectures: adopting and evolving a product-line approach. ACM Press/Addison-Wesley Publishing Co., New York, NY, USA.
4. Buccella, A., Cechich, A., Arias, M., Pol'la, M., del Socorro Doldan, M. and Morsan, E. (2013). Towards systematic software reuse of gis: Insights from a case study. Computers & Geosciences, 54(0): 9-20.

5. Clements, P. C. and Northrop, L. (2001). Software Product Lines: Practices and Patterns. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.
6. Czarnecki, K., Helsen, S. and Eisenecker, U. W. (2005). Formalizing cardinality-based feature models and their specialization. Software Process: Improvement and Practice, 10(1): 7-29.
7. Heineman, G. T. and Councill, W. T. (editors) (2001). Component-based software engineering: putting the pieces together. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.
8. Huenchuman, N. (2012). Reestructuracion de una lnea de productos de software para el subdominio de ecologa marina. Tesis de Licenciatura en Ciencias de la Computacion, 2013.
9. Waraporn Jirapanthong. Experience on re-engineering applying with software product line. CoRR, abs/1206.4120.
10. Kang, K., Cohen, S., Hess, J., Nowak, W. and Peterson, S. (1990). Feature-Oriented Domain Analysis (FODA) Feasibility Study. Technical Report CMU/SEI-90-TR-21, Software Engineering Institute, Carnegie Mellon University Pittsburgh, PA.
11. Laguna, M. and Crespo, Y. (2013). A systematic mapping study on software product line evolution: From legacy system reengineering to product line refactoring. Science of Computer Programming, 78(8): 1010-1034.
12. Mcilroy, D. (1969). Mass-produced Software Components. In Proceedings of Software Engineering Concepts and Techniques, pp. 138-155. NATO Science Committee.
13. Pernich, P., Buccella, A., Cechich, A., Doldan, S. and Morsan, E. (2010). Reusing geographic e-services: A case study in the marine ecological domain. In Wojciech Cellary and Elsa Estevez, editors, Software Services for e-World, volume 341 of IFIP Advances in Information and Communication Technology, pp. 193-204. Springer Boston.
14. Pernich, P., Buccella, A., Cechich, A., Doldan, S., Morsan, E. (2012). M. Arias and M. Pol'la. Product-line instantiation guided by subdomain characterization: A case study. Journal of Computer Science and Technology, Special Issue 12(3). ISSN: 12(3): 116-122.
15. Pohl, K., Böckle, G. and Van der Linden, F. J. (2005). Software Product Line Engineering: Foundations, Principles and Techniques. Springer-Verlag New York, Inc., Secaucus, NJ, USA.
16. Smith, D., Liam, O. and Bergey, J. (2002). Using the options analysis for reengineering (oar) method for mining components for a product line. 2379:316-327.
17. Clemens A. Szyperski (1998). Component software: beyond object-oriented programming. Addison-Wesley-Longman.
18. Van der Linden, F., Schmid, K. and Rommes, E. (2007). Software Product Lines in Action: The Best Industrial Practice in Product Line Engineering. Springer-Verlag New York, Inc., Secaucus, NJ, USA.
19. Zhang, G., Shen, L., Peng, X., Xing, Z. and Zhao, W. (2011). Incremental and iterative reengineering towards software product line: An industrial case study. In Proceedings of the 2011 27th IEEE International

Conference on Software Maintenance, ICSM '11, pp. 418-427, Washington, DC, USA. IEEE Computer Society.
20. Stoermer, C. and O'Brien, L. (2001). Map: mining architectures for product line evaluations. In Proceedings of the Working IEEE/IFIP Conference on Software Architecture, WICSA '01, Washington, DC, USA. IEEE Computer Society.
21. Bayer, J., Flege, O., Knauber, P., Laqua, R., Muthig, D., Schmid, K., Widen, T. and DeBaud, J. (1999) Pulse: A methodology to develop software product lines. In Proceedings of the 1999 Symposium on Software Reusability, SSR'99, pp. 122–131, New York, NY, USA. ACM.
22. Bayer, J., Girard, J., Würthner, M., DeBaud, J. and Apel, M. (1999). Transitioning legacy assets to a product line architecture. SIGSOFT Softw.Eng. Notes, 24(6):446–463.
23. Fenske, W.,#

An Experimental Analysis of Application Types for Mobile Devices

LISANDRO DELÍA[1], NICOLÁS GALDAMEZ[1], PABLO THOMAS[1], PATRICIA PESADO[1]

[1] Institute of Research in Computer Science III-LIDI.
School of Computer Science.
National University of La Plata. Argentina
{ldelia, ngaldamez, pthomas, ppesado}@lidi.info.unlp.edu.ar

Summary. *The popularity of mobile devices has generated new challenges for software engineers. The technical capabilities offered, as well as their restrictions, are a fertile, albeit complex, scenario. There are various alternatives when developing applications for mobile devices. In this paper, we discuss the existing software development approaches, their main characteristics, and an experimental case that allows analyzing the advantages and disadvantages of each approach.*

Key words: *mobile devices, native mobile applications, hybrid mobile applications, web mobile applications.*

1. Introduction

Mobile devices are part of everyday life and are increasingly sophisticated; their computation power creates possibilities that were unthought-of until a few years ago.

The growing demand for specific software for these devices has generated new challenges for developers, since this type of applications has its unique characteristics, restrictions and requirements, which are different from those of traditional software development.

Mobile computing can be defined as a computational environment that has physical mobility. Mobile computational environment users will be able to access data, information or other logical objects from any device on any network while they are on the move [1].

The specific features of this environment include: high level of competitiveness, need for short delivery times, and the added difficulty for identifying stakeholders and their requirements.

Applications are generated in a dynamic and uncertain environment. Typically, they are small and non-critical, but not necessarily non-significant. They are targeted at a large number of end users and are released as quick versions to meet market demands [2].

The development of mobile applications is currently a great challenge due to its specific demands and the technical restrictions of mobile environments

[3], such as devices with limited capabilities, but in continuous evolution; various standards, protocols and network technologies, need to operate on different platforms, user-specific requirements, and market time-stringent demands.
These devices have distinct physical characteristics, among which their size, weight, screen size, data input mechanism, and expandability stand out. Also, technical aspects, including processing power, memory space, battery autonomy, operating system, and so forth, play an essential role. All these characteristics have to be carefully considered when developing applications [4].
Throughout the brief history of software development, hardware and software platforms have evolved constantly, but never before has the computational power users hold in their hands been so massive, specifically through the use of mobile devices. This results in new challenges and, with them, the growth of Software Engineering as discipline, accompanying this evolution.
In this paper we present a comparative study of the types of applications for mobile devices, based on an experimental case developed for the educational platform WebUNLP [6]. In Section 2, the most relevant characteristics of the different types of applications for mobile devices are detailed. Then, the development process for these different types of applications is discussed, applied to the experimental case used as reference. Finally, the conclusions and future lines of work are presented.

2. Types of applications for mobile devices

In recent years, the mobile device market, especially that of smart phones, has seen a remarkable growth, both in Argentina and the world. In particular, the platforms that have grown the most in our country are Android and iOS [8] [9].
Each of these platforms has its own development infrastructure.
The main challenge for application providers is to offer solutions for all platforms, but this has a high cost [11].
The ideal solution to this problem is creating and maintaining a single application for all platforms. The purpose of multi-platform development is maintaining the same code base for various platforms. Thus, the development effort and cost are significantly reduced.
In the following sections, we present three approaches for the development of applications for mobile devices: one native approach and two multi-platform approaches (web and hybrid).

2.1 Web Applications

Web applications for mobiles are designed to be executed in the browser of the mobile device. These applications are developed with HTML, CSS and JavaScript, i.e., the same technology used to create websites.

One of the advantages of this approach is that no specific component has to be installed in the device, and no approval from the manufacturer is required for the applications to be published and used. Only Internet access is required. Also, application updates appear directly on the device, since changes are applied on the server and available immediately. In brief, it is fast and easy to implement.
The main advantage of this type of applications is their independence from the platform. There is no need to adapt to any operating environment. Only a browser is required.
On the flip side, this decreases execution speed and the applications could be less attractive than native ones. Additionally, their performance could be lower due to connectivity issues. Finally, this type of applications cannot use all of the hardware elements offered by the device, such as camera and GPS, among others.

2.2 Native Applications

Native applications are those that are conceived to be run on a specific platform, i.e., the type of device and the operating system to be used, and its version, have to be taken into consideration.
The source code is compiled to obtain the executable code, similar to the process used for traditional desktop applications.
When the application is ready for distribution, it is transferred to the specific App Stores (application stores) of each operating system. These stores have an audit process in place to assess if the application meets the requirements of the platform in question. After this step is completed, the application is available to the users.
The main advantage of this type of applications is the possibility of interacting with all the capabilities offered by the device (camera, GPS, accelerometer, calendar, and so forth). In addition to this, Internet access is not strictly necessary. Their execution is fast and they can be run in the background and alert the user when an event requiring their attention occurs.
Clearly, these advantages are paid for with higher development costs, since different programming languages have to be used for the different platforms. Therefore, if the goal is to span over several platforms, an application for each of them has to be produced. This carries greater costs for updating and distributing new versions.

2.3 Hybrid Applications

Hybrid applications combine the best of both of applications before mentioned. Multi-platform technologies such as HTML, JavaScript and CSS are used, but a good portion of the specific capabilities offered by the devices can be accessed.
In brief, they are developed using web technology and run within a web container on the mobile device.

The main advantages of this methodology include the possibility of distributing the application through the App Stores, reusing the code for multiple platforms, and using the hardware features of the device.
One of the disadvantages is that, since the same interface is used for all platforms, the look and feel of the application will not be that of a native application. Finally, execution will be slower than that of native applications.

3. Experimental Case: WebUNLP

3.1 Description of the Problem

WebUNLP is a virtual teaching and learning environment that allows educators to present their educational proposals. Students and educators can meet in this space to share study materials, communicate, and generate a virtual educational experience [6].
Currently, WebUNLP has a web version that focuses on desktop and portable computers, but it is not adapted to be used from mobile devices.
The development presented in this paper consists in expanding WebUNLP by building a mobile application that will allow access to certain system functionalities through mobile devices. The approach proposed includes reviewing the same solution to compare native, web and hybrid developments in order to determine which one is the most convenient.
As with any software development, building a mobile application involves having a clear definition of its purpose and the requirements it has to meet. In particular, in the case of software for mobile devices, it is essential that objectives are more specifically defined than for their desktop counterparts [7].
In the case of WebUNLP in particular, an incremental development of a mobile application was carried out, and this first version was focused on one of its communication tools: the notice board. This tool allows communicating news pertaining to courses such as, for example, changes in schedule, reminders for assignment delivery dates, and so forth [6].

3.2 Analysis

One of the first issues raised was the selection of a platform. In terms of market, the dominant operating systems in Argentina are Android and iOS [8][9], so it was decided to support these two operating systems.
Functional and non-functional requirements were analyzed separately from the platform and then specifically for each of them.
Below, some of the requirements that the application has to meet are listed:

- Users must be able to access the application with the same credentials they use to access the web version.
- Users must be able to access the notice board of all the courses they are taking, either as educator or student.

- Users must receive a notification on their devices when there is news published on their notice boards. This requirement is not possible for the web version that is accessible from desktop and/or portable computers.
- Users must have the same use experience with all operating platforms.
- The existing web application must be synchronized with the mobile application to be developed, which means that any changes made through the mobile application must be reflected on the web version and vice versa.

3.3 Design

To meet the requirements listed in the previous section, with the exception of the one regarding notifications, the design of the web mobile application is a replication of the offerings of WEBUNLP, and only the interface would have to be adapted to the screen size of a mobile device.

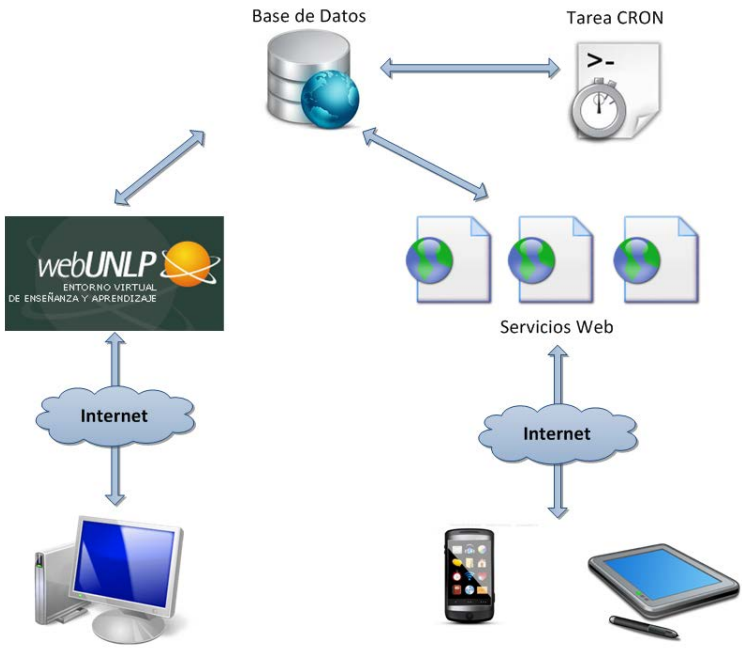


Fig. 1 *- Generic architecture for the native and hybrid applications*

However, the design of the native and hybrid versions of the application is more complex. Figure 1 presents the generic architecture for all the components involved in this development scenario. It shows access from PCs to WebUNLP and access from mobile devices (phones and tablets) to the

information in WebUNLP through web services. For developing these web services, a Restful API (Application Programming Interface) was designed, chosen due to its simplicity, scalability and interoperability [14] [15].
There is also a scheduled task (Cron) that is run intermittently at regular time intervals to notify the corresponding mobile devices when a news event is created on the WebUNLP notice board. Cron identifies the operating system of the device that will receive the notification and generates the notification.
As for the graphic interface, a design that is independent from the platform was proposed to analyze application usability aspects, and then the necessary adjustments were implemented for each type of application.
For the interfaces that were designed, navigation is done serially, in the order shown in the mockup [13] depicted in Figure 2.

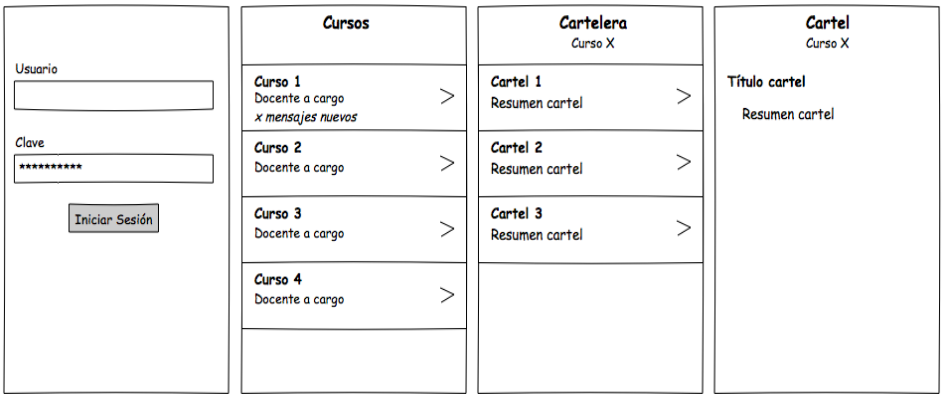


***Fig. 2** - Mockup, independent from the type of application*

3.4 Development

3.4.1 Native Application for Android

For developing applications for the Android platform, a JDK (Java Development Kit) and its programming environment, known as Android SDK (Software Development Kit), are required. The latter provides the libraries and tools needed to build, test, and purge applications for Android.
The development of the WebUNLP application for Android followed the conventions adopted by its community because they apply good practices for interface building and their backing logic, access to data and web services. To meet the requirement for receiving notifications on the corresponding device, Cron generates the notification by means of the GCM (Google Cloud Messaging) service [16]. Figure 3 shows the interfaces of the application developed.

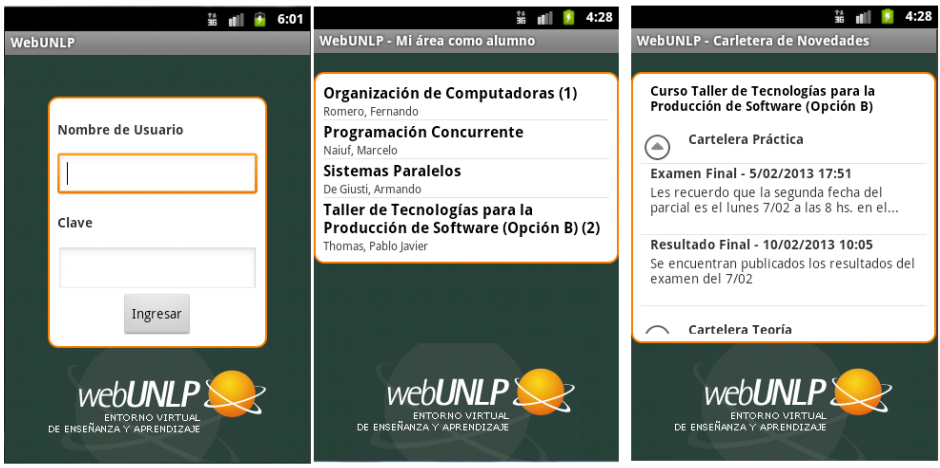


Fig. 3 *- Native application for Android*

3.4.2 Native Application for iOS

The Apple iOS platform is based on a proprietary model, and therefore, for developing a native iOS application, an Apple Mac running OS X with Xcode installed is required. The main programming language is Objective C. Xcode is Apple's development environment for all its devices, and is responsible for providing iOS SDK with the necessary tools, compilers and frameworks. Additionally, Xcode is in-built with simulators for iOS devices (iPhones and iPads) that simplify the testing stages of the system developed.
In relation to graphic interface and user interaction, the conventions proposed by Apple [10] were followed to achieve a better integration between the application and the operating system and improve user experience.
Finally, to meet the requirement for notifications, Cron notifies the corresponding iOS device by means of the APNs (Apple Push Notification Service) service [17]. Figure 4 shows the interfaces of the application developed.

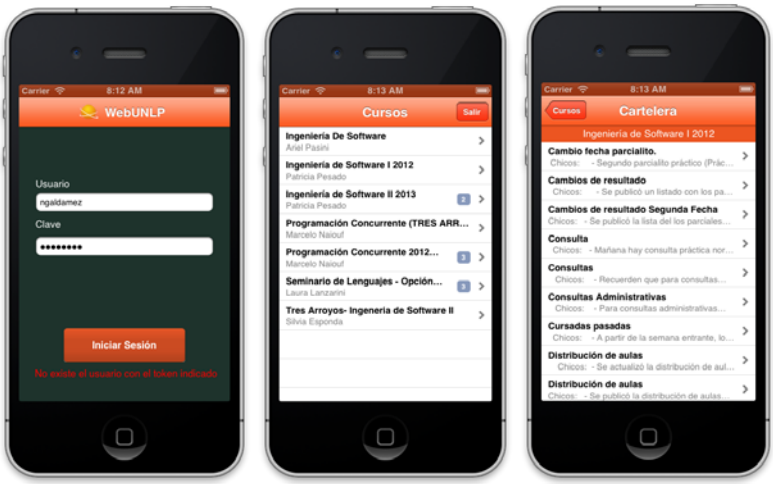


Fig. 4 *- Native application for iOS*

3.4.3 Hybrid Application

For building the hybrid version of the WebUNLP application, the PhoneGap framework [19] was used, which allows developing mobile applications using technologies that are common to all devices: HTML5, CSS and JavaScript.
Also, the JavaScript framework called Jquery Mobile [20] was used to achieve interfaces with consistent aspect and behavior across the various mobile platforms.
For the implementation of the MVC (Model, View, Controller) design pattern, the Backbone.js library [21] was used.
Finally, to meet the notifications requirement, the Pushwoosh plug-in [18] was used.
Figure 5 shows the interfaces of the application developed.

3.4.4 Web Application

Finally, a web application capable of accessing WebUNLP's notice board was developed. This application is available for any mobile device that has a browser that admits the features used for the development: HTML5, CSS3, JavaScript.
Since data transmission/reception speed in a mobile device through WiFi, and 3G in particular, is lower than the speed of a desktop, the web version of WebUNLP's notice board is light and most of the requirements are implemented through Ajax [12] to avoid, in case of changes, the need to reload the entire page.

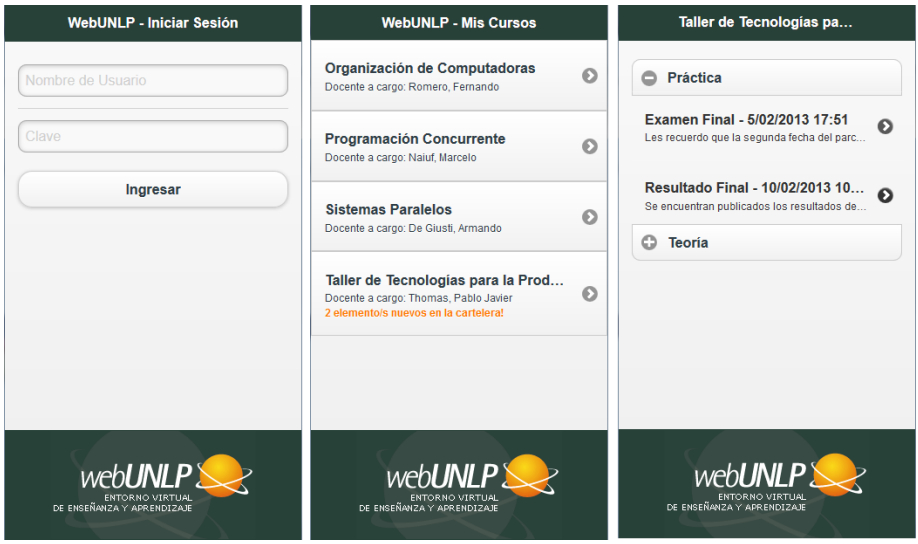


Fig. 5 *- Hybrid application*

4. Conclusions

Initially, mobile devices were conceived and designed with a specific purpose. Through the years, technological growth has allowed adding new functionalities, which has in turn allowed expanding their uses.

Currently, the underlying computational power that a great variety of mobile devices offer has created new possibilities, which results in a challenge for software engineers.

It is hard to make absolute statements in this context of spiraling and continuous evolution. It is clear that, for the time being, there are three possibilities for developing the same application, and an analysis of each of these has allowed drawing some conclusions.

There are certain aspects that are specific to selecting a domain to develop an experimental case. The novelty of having a mobile application is not reason enough to justify such development. A set of clearly established requirements has to be met.

WebUNLP is a virtual teaching and learning environment used by various grade and post-grade courses of the National University of La Plata. Therefore, there is a significant number of potential users that could access this environment from any place.

Thus, this virtual space was selected for replication into a mobile version that not only brings it closer to its users, but also expands its functions, in this particular case the notice board.

Currently, the three types of mobile application (web, native and hybrid) have already been developed with the same purpose and to meet the same set of requirements.

The great simplicity for generating the web version stands out, since the same technological tools as those used for developing any traditional web application are used. The main difference in this sense is the limitation in relation to screen space on the device. However, the greatest contrast is that not all the hardware capabilities offered by the device can be accessed, which prevented the implementation of one of the requirements, probably the most interesting one – the notification of WebUNLP notice board news to the user.
On the other hand, the native versions met all requirements. However, the main disadvantage in this case is non-portability, which results in the need of platform-specific development. In this paper, we presented developments for Android and iOS, the most widely used operating systems in Argentina, with a greater inherent cost.
Finally, the hybrid version combined the simplicity of the web development with the use of all of the capabilities offered by the device. This type of approach is aimed at overcoming the disadvantages of both previous approaches, and is therefore positioned as the prima facie choice, always conditioned by the specific requirements to be met.
As use test, the native version of the application was recently made available to users, who expressed their interest by requesting the addition to it of other functions offered by WebUNLP.

5. Future Work

In the short term, some of the requirements met by WebUNLP that are not included in the mobile version will be expanded, such as, for instance, messaging and forums. Also, it is expected that new requirements will be added, such as a chat service.
From a developmental point of view, alternatives to generate hybrid mobile applications with other frameworks [22] [23] will be studied.

References

1. Talukder, A.K., Ahmed, H. and Yavagal, R. (2010). *Mobile Computing, Technology, Applications, and Service Creation*. Second Edition. Tata McGraw-Hill.
2. Abrahamsson, P. (2005). *Mobile software development - the business opportunity of today*. Proceedings of the International Conference on Software Development, (pp. 20-23). Reykjavik.
3. Hayes, I. S. (2002). *Just Enough Wireless Computing*. Prentice Hall Professional Technical Reference. ISBN:0130994618
4. Abrahamsson P. et. al. (2004). *Mobile-D: An Agile Approach for Mobile Application Development.* OOPSLA'04, Oct. 24–28, Vancouver, British Columbia, Canada.

5. Tracy, K.W. (2012). *Mobile Application Development Experiences on Apple's iOS and Android OS*, Potentials, IEEE.
6. WebUNLP website. http://webunlp.unlp.edu.ar
7. Salmre, I. (2005). *Writing Mobile Code Essential Software Engineering for Building Mobile Applications*. Addison Wesley Professional.
8. http://gs.statcounter.com
9. http://www.mapbox.com/labs/twitter-gnip/brands/#4/-40.43/-63.62
10. *iOS Human Interface Guidelines*, http://developer.apple.com/library/ios/#DOCUMENTATION/UserExperience/Conceptual/MobileHIG/Introduction/Introduction.html
11. Raj R., Tolety S.B. (2012). *A study on approaches to build cross-platform mobile applications and criteria to select appropriate approach*. India Conference (INDICON), Annual IEEE.
12. https://developer.mozilla.org/en/docs/AJAX
13. http://es.wikipedia.org/wiki/Mockup
14. Richardson L., Ruby S. (2007). RESTful Web Services, O'Reilly Media.
15. http://msdn.microsoft.com/es-es/magazine/dd315413.aspx#id0070023
16. http://developer.android.com/google/gcm/index.html
17. http://developer.apple.com/library/mac/#documentation/NetworkingInternet/Conceptual/RemoteNotificationsPG/Chapters/ApplePushService.html#//apple_ref/doc/uid/TP40008194-CH100-SW9
18. http://devgirl.org/2012/12/04/easy-phonegap-push-notifications-with-pushwoosh/
19. http://phonegap.com/
20. http://jquerymobile.com/
21. http://backbonejs.org/
22. Digital Possibilities. Mobile Development Frameworks Overview http://digital-possibilities.com/mobile-development-frameworks-overview/
23. Markus Falk. Mobile Frameworks Comparison Chart, http://www.markus-falk.com/mobile-frameworks-comparison-chart/

Generating a Ranking Algorithm for Scientific Documents in the Computing Science Area

H. KUNA[1], M. REY[1], J. CORTES[1], E. MARTINI[1], L. SOLONEZEN[1]

[1]Computer Science Department. School of Exact, Chmistry and Natural Sciences, National University of Misiones. Argentina.
{hdkuna, m.rey00}@gmail.com

Abstract. *The generation of a ranking algorithm for the sorting out of scientific documents belonging to the computing science area is a fundamental requirement for the developing of Information Retrieval Systems which should be able to work on such kind of elements. These systems aim to optimize the process of content search in the web by means of several tools such as meta-searchers. The same widen the search scope since they are able to use the data bases of several searchers simultaneously; in addition they are able to include several methods for sorting out the documents, which improve the relevance for the user. In this paper it is introduced the developing of a ranking algorithm to sort out the list of results which returns an Information Retrieval Systems for the searching of scientific documents in the computing science area.*

Keywords: *information retrieval, ranking algorithm, web search, bibliometric indicators.*

1. Introduction

1.1 Information Retrieval Systems

An Information Retrieval System (IRS) can be defined as a process which is able to store, retrieve, and keep information [1], [2]. In the existing computing science literature, there are several proposals about the basic structure which should have an IRS. An example of this is the one which considers it as the result of the union of four elements such as [3]:

- The documents which are part of the collection on which the retrieval of information will be performed.
- The queries which represent the needs of information on the part of the users.
- The way in which the documents and queries representations and the existing relationships among them are modelled.
- The evaluation function which determines order of the documents in the results to be presented takes into account the query and some properties to be evaluated in the documents.

Nowadays the main IRS models which run in the internet are directories, web search engines and meta-searchers [4]. Taking into account such a classification, it can be stated that there are several IRS implementations in the web which use different search methods on general or specific contexts as it can be read in different publications [5],[6].

1.2 IRS for Scientific Documents in the Computing Science Area

No evidence has been found about the existence of IRS implementations which are specifically applied to data bases of scientific documents belonging to the computing science area and besides which implement several methods to improve quality of the listing of the results to be presented to the user on the basis of the relevance which they can have according to the query performed.
In the context of the present article, the meta-searchers become more relevant owing to the fact that they enable to use other searchers data base replicating the users' queries on every one of them and to process the results in the most convenient way to generate a unique list of results to be presented to the user.
Generating an IRS which works on scientific documents in the computing science area requires the development of various components among which stands out the algorithm to be used for the evaluation of every document obtained from the searches with the objective of fusing and sorting out the final list of results.

1.3 Metrics for Evaluating Scientific Documents

Owing to the IRS nature discussed above and the ranking algorithm to be generated for same, the methods for the evaluation of scientific papers must be developed in a specific way. In order to do that a series of evaluating characteristics should be considered, such as [7], [8]:

- The quality of publishing source, distinguishing if it is published in a scientific journal or in a scientific conference or similar event.
- The authors' quality, in this case taking into account the number of publications the author himself has released and their relevance, measured by the number of citations that would produce.
- The article quality itself, in this case, measured by the times it has been cited as time has gone by.

For each one of these characteristics there are metrics widely accepted which can be applied. Some of them can be observed in table 1.

Table 1. Metrics assessed for the evaluation of scientific articles

Features to be evaluated	Available metrics		Origin of the metrics
Quality of publication source	Publication in scientific Journals	Impact factor (IF) [9]	Web of Knowledge[1] – Institute for Scientific Information (ISI)
		SCImago Journal Rank (SJR) [10]	Scopus[2] – SCImago Group, Extremadura University, Spain
	Publication in Scientific Conference	CORE Ranking [11]	Computer Research & Education of Australia[3]
Authors' quality	H Index [12]		Scientific Article
	G Index [13]		Scientific article
Quality of the article	AR Index [14]		Scientific Article
	Number of citations		-

In the case of the type of source of publication , there are two indexes which are used to estimate the quality of the source of publication issued in journals, the impact factor (IF) [9] and the SJR index, SCImago Journal Rank [10]. In both cases it is about metrics that evaluate the citations that the articles published receive considering the amount as well as the relevance that the journal published. Meanwhile, in the case that the article was published in a conference or similar event there is a ranking as the one which the Computer Research & Education of Australia (CORE) web generates [11]. Different scientific meetings or conferences are classified in the following four established levels: A*, A, B y C, listing which will be modified and updated in short time as said by the authors of the web.

To estimate an author's production quality, the metrics available are the H index [12] and the G index [13]. These take the amount of citations received by the authors' different publications and the amount of publications to compute a value which represents its influence.

To evaluate the quality of a set of publications throughout the time, an index as the AR [14] can be used which take their period of time published and ponder them using that factor in combination with the amount of citations obtained by each one of the articles which make up the collection; being the last factor, the citation account, another of the available metrics to evaluate the quality of a specific scientific document.

[1] www.wokinfo.com – Accessed: 14/04/14
[2] www.scopus.com – Accessed: 14/04/14
[3] www.core.edu.au – Accessed: 14/04/14

The current research work objective is to develop a ranking algorithm for specific scientific documents for the computing science area so as to generate a component which could be included in an IRS, specifically a meta-searcher whose purpose is the retrieval of the contents of this knowledge area in the web. For such a task it is pretended to include several metrics in an algorithm, which allow the evaluation of a scientific article from different aspects such as the publication source quality, the quality of the authors which subscribe it and the article quality itself.

2. Methods and Materials

2.1 IRS Structure

Owing to the fact that the ranking algorithm should be incorporated to an IRS, it was necessary to consider which its structure would be; being the same made up by modules to perform the following operations, a chart of the modules can be seen in figure 1:

- Handling of the queries introduced by the user to be used on the integrated data sources;
- Carrying out the search, replicating the user's query in the different integrated data sources;
- Capturing, selecting, and unifying the results obtained from the different sources, being this one, the module in which the algorithm to be developed would be incorporated;
- Improving the results to be presented to the user by means of different intelligent techniques.

An issue to be considered was the data sources to which the IRS would access to obtain the scientific documents which matched the user's requirements, taking into account that they would allow to obtain, directly or indirectly, the values corresponding to the metrics which would be determined to be used in the evaluation algorithm. Initially the searchers selected were the Google academic searcher, Google Scholar[4], and the Scopus search engine from the Elsevier publishing company. This selection was done because both tools are continually evolving their components and the documents they access and fulfil with the requirements to possess different metrics which can be used to evaluate the publications which they retrieve, being great quality alternatives to fulfil with the planned objectives[15], [16].
Having the context defined, in which the ranking algorithm would work, the design, the development and later validation was carried out.

[4] www.scholar.google.com - Accessed: 14/04/14

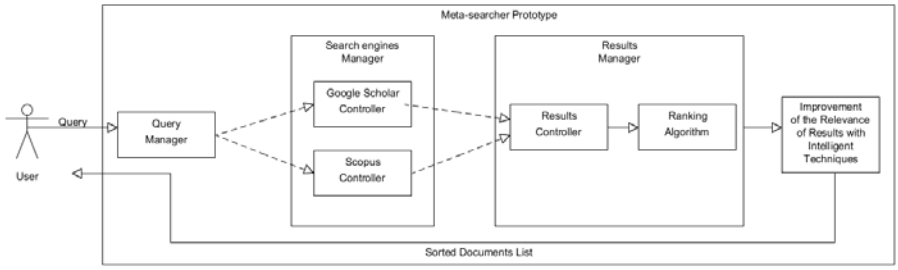


Fig. 1. *Meta-searcher components*

2.2 The Algorithm Design

In the first instance, the metrics to be used in the ranking algorithm were selected, emphasising different aspects in each one, considering again the evaluable characteristics of the scientific documents, the quality of the source of publication, the authors' quality and the article quality itself. The metrics were determined to ponder every result

- For the publication source quality, two factors were considered to evaluate it, depending on the fact that if the article was published in a scientific journal or in the corresponding area of knowledge conference. In the first case, it has been opted to use the SJR index [10] developed by the SCImago research team; this selection was due to the advantages it has as regards the ISI IF [9]. Such as: is open access, the Scopus data base has a larger amount of journals, including those which are not written in English, the citations received in an article not only undergo a quantitative evaluation but also a qualitative one, incorporating the quality of the journal which generates the citation, among others [17], [18]. In the case of conference and scientific meeting articles, the ranking which was used was the one generated by the Computing Research and Education Association of Australia (CORE). In a few words:
  - *If* (publication_type = scientific_journal) *Then* use_SJR
  - *If* (publication_type = scientific_conference) *Then* use_CORE
- In the case of the author's quality, the H index [12] was used since it is widely accepted and used for the evaluation of an certain author's scientific production [7], [8] even considering some critics done about it. The index represents an author's X amount of articles which have received X citations as a minimum.
- In the case of an article quality, it was decided to consider both metrics previously evaluated, the AR index [14] and the amount of citations received by the article emphasizing the need of the first one to be adapted to work on an only one document instead of working on a collection as it was originally exposed.

2.3 Developing a Ranking algorithm

Once the metrics which would make up the algorithm were selected, the development of the algorithm was carried out. For such activity the formulas were initially defined by means of which the corresponding values for every document property would be calculated.

- For the corresponding factor to the property of the quality of the publication source, in case the publication was issued in a scientific journal, the corresponding value is calculated using the logarithm on the base of 10 of the value of SJR index of the journal. This is done with the objective to homogenize this factor values in regards to rest of the algorithm components, since the range of values present in the index is greater than ten in a great number of journals. However if the publication takes place in a conference or scientific event, the classification model which is given by the CORE ranking had to be adapted by transforming the conference classification to a numeric format to be able to work with it. The value corresponding to the factor of the publication source is obtained by means of 1 formula if it is issued in a journal, although if it is issued in a conference, formula number 2 is used.

$$\text{publicationSource} = \log_{10}(\text{SJR}) \; . \tag{1}$$

$$\text{publicationSource} = [A^* = 1;\; A = 0.75;\; B = 0.5;\; C = 0.25] \; . \tag{2}$$

- For the factor corresponding to the authors' quality, the author's H index for the article in evaluation is considered. If it is an article with more than an author, the index value is pondered in function to the position which occupies in the list of authors of the document. Besides, the logarithm on base 10 is used again for the resulting value of the pondered addition of the authors' H index values of the article. The computation of the factor can be seen in formula 3.

$$\text{authors} = \log10(\bullet\ (\text{hIndex}(\text{author}_i))/i) \; . \tag{3}$$

- For the factor corresponding to the document quality, in this case, as the combined approach to use the AR index and annex to the same one the amount of citations received by the publication, it was determined that the factor which would ponder the quality of the publication would be the resulting quotient between both elements: the amount of citations and the amount of years since its publication. Giving origin to formula 4 in which the result of the adaptation done previously can be observed.

$$\text{publicationQuality} = \text{recievedCitations}/\text{yearsFromPublicationDate} \tag{4}$$

Once the corresponding components to every one of the features to be evaluated of a scientific document were determined, an adjusting factor would be added to the final algorithm computation, which would have the function to allow that one of the factors had more importance than the others. The calculation of the factors associated to every property multiplied by the adjusting factors results in a final value which is used to realize the order of the results before presenting them to the user.

- The inclusion of the adjusting factors associated to the components which correspond to the evaluated properties produce the final value which correspond to every document in evaluation by the ranking algorithm, this is displayed in formula 5. The stated values along with the experts in the thematic area, for the adjusting factors, were 0.5, 0.3 and 0.2 respectively.

$$finalValue = \pm * [publicationSource] + {}^{2} * [authors] + {}^{3} * [publicationQuality] . \quad (5)$$

3. Experimentation

3.1 Development of the IRS prototype for the experimentation

With the objective of validating the right proposed ranking algorithm functioning, the same has been included in the meta-searcher prototype, which constitutes the IRS partial implementation described in sections above. The mentioned prototype was developed prioritizing the use of technologies which were based in the Open Source philosophy, such as HTML, PHP and SQL languages, along with the MySQL database engine, using the Apache web server as its implementation environment.
The prototype implementation process falls in the following steps:

1. Development of the methods to access, query and extract the results from Google Scholar and Scopus.
2. Implementation of the ranking algorithm with access to the data sources which store the values of the different metrics involved.
3. Development of the visual prototype components, i.e., the interfaces which capture the user's query and the corresponding one to the presentation of the listing of the unified and sorted results.
4. Integration of all the components in an only one software product.

3.2 Validation of the developed algorithm

The validation process fell into two stages which evaluated the results from two perspectives. First, the results were considered from an expert's view on library sciences. Secondly, the ranking algorithm was evaluated as component of the ISR prototype which is in charge of improving the results relevance to be presented to the final user. To achieve what has been

mentioned above, three experts on web information retrieval have been consulted.
For the first instance of validation whose details can be seen in table 2, several queries have been done by using the meta-searcher prototype described in the prior section, by working on reduced number of documents and exporting the calculation results corresponding to the ranking algorithm to an external file to IRS. Taking into account such data, the expert on the area of library sciences has determined that the metrics employed have been accurately calculated, generating a numeric value which allows to establish an order among the documents which are part of the list of the resulting search, based on the documents relevance evaluated from the selected properties.

Table 2. Results of the first validation instance

Queries performed	Amount of processed results	Effectiveness evaluated by the expert
data mining AND outliers	20 (10 Google Scholar + 10 Scopus)	74%
fuzzy sets AND clustering	20 (10 Google Scholar + 10 Scopus)	87%
alphanumeric data AND outliers	20 (10 Google Scholar + 10 Scopus)	81%
scientific production AND metrics	20 (10 Google Scholar + 10 Scopus)	77%
text mining AND ontologies	20 (10 Google Scholar + 10 Scopus)	96%

Afterwards the validation of the ranking algorithm as a component of the IRS prototype was carried out by the experts. The experimentation details are displayed in table 3 in which the amount of queries and results to obtain were increased. In this case the way in which the experts worked was to evaluate the results quality as regards the user's different requirements. As a result, it has been determined that the results management component, through the developed ranking algorithm, fulfils satisfactorily with the objective of evaluating the results quality for the generation of the final listing to be presented to the user achieving that the listing presents in the first places those scientific articles of greater quality.

Table 3. Results of the second validation instance

Queries performed	Amount of the processed results	Effectiveness evaluated by the experts
data mining AND outliers	100 (50 Google Scholar + 50 Scopus)	72%
fuzzy sets AND clustering	100 (50 Google Scholar + 50 Scopus)	93%
alphanumeric data AND outliers	100 (50 Google Scholar + 50 Scopus)	84%
scientific production AND metrics	100 (50 Google Scholar + 50 Scopus)	90%
text mining AND ontologies	100 (50 Google Scholar + 50 Scopus)	94%
data mining AND systems audit	100 (50 Google Scholar + 50 Scopus)	69%
scientific articles AND ranking algorithms	100 (50 Google Scholar + 50 Scopus)	83%
fuzzy controllers AND robotics	100 (50 Google Scholar + 50 Scopus)	79%
fuzzy sets AND document processing	100 (50 Google Scholar + 50 Scopus)	75%
web agents AND document analysis	100 (50 Google Scholar + 50 Scopus)	86%

4. Conclusions and Future Research Works

The present research work has managed to develop and validate a specific ranking algorithm for the evaluation of scientific documents belonging to the area of the computing science. Several bibliometric indicators have been taken into account with the aim to obtain values for the evaluation of different properties to determine the quality of scientific documents of the computing science area. In addition, the algorithm has been included in the IRS prototype, specifically, a meta-searcher whose field of application are the documents above mentioned, constituting a significant advancement as regards the objectives of the present research work.

What can be mentioned as future research works is to evaluate the incorporation of other bibliometric indicators which can be useful to the ranking algorithm, always considering the own specificity of the area to

which the documents to be evaluated belong; to evaluate the incorporation of fuzzy logic and/or artificial intelligence to automatize the adaptation of the ranking algorithm adjusting factors; incorporate an evaluation of the authors' reputation to the analysis of every resulting article to the thematic sub-area on which the search will be carried on, among others.

5. Bibliography

1. Salton, G., Mcgill, M. (1983). Introduction to Modern Information Retrieval. McGraw-Hill, Inc.
2. Kowalski, G. (1997). Information Retrieval Systems: Theory and Implementation. Kluwer Academic Publishers, Norwell, MA, USA.
3. Baeza-Yates, R., Ribeiro-Neto, B. (1999). Modern information retrieval. ACM press, New York.
4. Olivas, J. A. (2011). Búsqueda Eficaz de Información en la Web. Editorial de la Universidad Nacional de La Plata (EDUNLP), La Plata, Buenos Aires, Argentina.
5. Serrano-Guerrero, J., Romero, F. P., Olivas, J. A., de la Mata, J. (2009). BUDI: Architecture for fuzzy search in documental repositories. Mathw. Soft Comput, 16, 1, 71–85.
6. de la Mata, J., Olivas, J. A., Serrano-Guerrero, J. (2004). Overview of an Agent Based Search Engine Architecture, en Proc. Of the Int. Conf. On Artificial Intelligence IC-AI'04, 62-67. Las Vegas, USA.
7. Bollen, J., Van de Sompel, H., Hagberg, A., Chute, R. (2009). A Principal Component Analysis of 39 Scientific Impact Measures. Plos One.
8. Pendlebury, D. A. (2009). The use and misuse of journal metrics and other citation indicators. Arch. Immunol. Ther. Exp. (Warsz.), 57(1), 1-11.
9. Garfield, E. (2006). The history and meaning of the journal impact factor. JAMA, 295(1), 90-93.
10. Gonzalez-Pereira, B., Guerrero-Bote, V., Moya-Anegon, F. (2009). The SJR indicator: A new indicator of journals' scientific prestige, arXiv:0912.4141.
11. CORE Conference Ranking, Computer Research & Education of Australia, http://www.core.edu.au
12. Hirsch, J. E. (2005). An index to quantify an individual's scientific research output. Proc. Natl. Acad. Sci. U. S. A., 102(46), 16569-16572.
13. Egghe, L. (2006). Theory and practise of the g-index. Scientometrics. 69(1), 131-152.
14. Jin, B. (2007). The AR-index: complementing the h-index. Issi Newsl. 3(1), p. 6.
15. Moya-Anegón, F., Chinchilla-Rodríguez, Z., Vargas-Quesada, B., Corera-Álvarez, E., Muñoz-Fernández, F. J., González-Molina, A., Herrero-Solana, V. (2007). Coverage analysis of Scopus: A journal metric approach. Scientometrics. 73(1), 53-78.

16. Meho, L. I., Yang, K. (2006). A New Era in Citation and Bibliometric Analyses: Web of Science, Scopus, and Google Scholar. arXiv e-print cs/0612132.
17. Falagas, M. E., Kouranos, V. D., Arencibia-Jorge, R., Karageorgopoulos, D. E. (2008). Comparison of SCImago journal rank indicator with journal impact factor. Faseb J. 22(8), 2623-2628.
18. Leydesdorff, L., Moya-Anegón, F., Guerrero-Bote, V. P. (2010). Journal maps on the basis of Scopus data: A comparison with the Journal Citation Reports of the ISI. J. Am. Soc. Inf. Sci. Technol., 61(2), 352–369.

Variants evaluation in a model designed to anticipate the convenience of tracing software projects

JUAN GIRÓ, JUAN VÁZQUEZ, BRENDA MELONI Y LETICIA CONSTABLE

Departamento de Ingeniería en Sistemas de Información
Facultad Regional Córdoba, Universidad Tecnológica Nacional
Maestro López esq. Cruz Roja Argentina, Ciudad de Córdoba
{juanfgiro, jcjvazquez, bemeloni, leticiaconstable}@gmail.com

Abstract. *The evidence shortage indicating that advances in the traceability field are indeed being used in the software industry has motivated model development leading to a better knowledge of the problem and to the ability of anticipating the expected results in projects. In order to do this, it was necessary to identify the greatest impact factors in the success of traceability processes and to propose models that allow us to make predictions from these factors. In this paper we evaluate the results of introducing variants in the metrics associated to those factors in order to facilitate the selection of the most convenient for the best performance of the prediction model. To that end, we use the ROC analysis that, despite its advantages, has had very little diffusion in software engineering so far.*

Keywords: *software engineering, ROC analysis, requirements traceability.*

1. Introduction

The evidences indicating that the progress in the field of requirements traceability of development projects is not being really applied in software industry [1] led to the necessity of better understanding the problem and its causes. Looking over the unfavorable experiences we concluded that they could be split into in three groups: *i)* the ones that were prematurely abandoned or did not satisfy the expectations from a technical point of view, *ii)* the ones that were successful technically with an implementation cost higher than the benefit obtained, and *iii*) the ones that with a high cost led to a poor or null result, that is a combination of the first two groups. This is surprising since nowadays the importance and transcendence of requirements traceability is thoroughly acknowledged as a support of software development processes [2], having been incorporated in every current development standard and model.

Before going on it is necessary to emphasize that when we talk about requirement traceability in software development projects we mean a management that links the various stages of its life cycles, ensuring the

project success, providing the necessary coherence, completeness and rightness guarantee to produced software and enabling its effective corrective and preventive maintenance for the rest of its useful life.

By analyzing the study lines in traceability field, it can be proven that most of them are oriented to the development of new tools and methodologies, with a much lesser effort destined to the study of the results obtained of the application these in the industry and the causes of the said difficulties.

In addition to this, the few documents [1][3] destined to analyze the origin of traceability difficulties address the problem in a qualitative way and most of the times the focus is too general.

Thus the presumption arose that it was not casual that certain projects can be successfully traced and other cannot, so there should be a combination of objective conditions that lead to one result or another and this idea led to formulate an hypothesis that *there are factors that condition the success of traceability processes and that it is feasible to identify them.*

The verification of this hypothesis points toward the activity accomplished in the project "Aseguramiento de la Trazabilidad en Proyectos de Desarrollo de Sistemas de Software" [4], and factors and models were proposed in that framework in order to anticipate the results of traceability processes, that are being progressively improved [5][6][7].

In this work we study variants of the metrics proposed for the evaluation of the said factors in order to identify the most convenient. The organization of this document is the following: Section 2 summarizes the characteristics of the studied model, the traceability factors chosen and the proposed variants for its metrics, Section 3 analyzes the impact of the new metrics in the performance of the model, discussing the results obtained, and Section 4 introduces the conclusions of this work and the future activities.

2. Projects traceability model, its factors and metrics

There are three main entities recognized that are closely related to each other: *a)* the *software product* itself, *b) the project*, that answers to a certain process model and includes the product construction, *c*) the *organization*, that constitutes the scenery in which the project is developed. Thus, it is anticipated that the factors searched will be linked to dimensions of these three entities.

The following criteria related to the selection of factors, how these are evaluated and their interpretation in relation to the addressed problem were also established: a) they must be quantifiable, *b*) a scale multiplier will be applied to them in order to express them in the *zero to five interval, c*) the *increasing* values are more convenient to the traceability of a project and *d*) they must be orthogonal with each other.

Thus we define the eight factors proposed [6][7] to predict the convenience of doing the requirements traceability in software development projects, grouped by entities, which are summarized in Table 1 and Table 2:

Table 1: Definition of entities, factors, associated variables, descriptions and reference variables

Entity	Factor	Variable	Description and reference variable *Vr*	*Vr* Interval
Product	Size	***t***	Function Points ***PF***	100 - 1000
	Validity	***v***	Useful Life ***VU*** [years]	0,5 - 10
	Reuse	***r***	Future reuse ***RE*** [%]	0 - 80
	Reliability	***c***	Reliability Indicator ***CO*** (*)	0 - 5
Project	Term	***p***	Execution Term ***DP*** [years]	0 - 5
	Team	***e***	Team effectiveness ***EF*** (**)	1 - 5
Organization	Maturity	***m***	Maturity Level ***CMMI***	1 - 5
	Dependency	***d***	Autonomy Level ***NA*** (***)	1 - 5

Table 2: Definition of ***CO*** (*), ***EF*** (**), Maturity Level ***CMMI*** and ***NA*** (***) indicators

Reliability ***CO***	
0	Non important
1	Low
2	Medium
3	High
4	Very High
5	Absolute

Effectiveness ***EF***	
1	Poor
2	Low
3	Medium
4	High
5	Very High

Maturity ***CMMI***	
1	Initial
2	Managed
3	Defined
4	Quantitatively Managed
5	Optimizing

Autonomy ***NA***	
1	Independent
2	Own Rules
3	Client Rules
4	Headquarters Rules
5	3 - 4 combined

There is a detail of the scope and justification of the proposed factors and adopted metrics for the presented model in references [6] and [7], and we will call this Model "A" from now on.

The representation of the eight factors by using a radar diagram is shown in Figure 1, where the polygons are examples of: *a*) a traceable project, *b*) a non traceable project, *c*) an atypical project and *d*) the "grey zone" that represents the discriminating border between both cases.

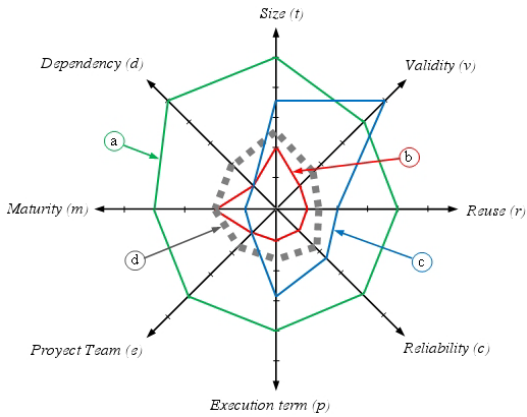


Fig. 1. *Representation of the projects traceability in a radar diagram*

With Model "A" we proposed to assign a variable to each one of the three entities involved: *product* (η_1), *project* (η_2) and *organization* (η_3), considering the orthogonality between the factors when calculating the resultant module of each one of the three variables. This allows reducing the problem dimension and facilitates the visualization of data populations, presenting the expressions of the three variables in Table 3.

Table 3: Reducing the problem to three dimensions

Entity	Variable	Evaluated Expression
Product	η_1	$\eta_1 = \sqrt{\left(t^2 + v^2 + r^2 + c^2\right)}$
Project	η_2	$\eta_2 = \sqrt{\left(p^2 + e^2\right)}$
Organization	η_3	$\eta_3 = \sqrt{\left(m^2 + d^2\right)}$

Starting from this dimensions reduction emerged the idea of using the resultant module of the eight factors, from now on "ρ", as a representative parameter or "indicator" of each considered case:

$$\rho = \sqrt{\left(\eta_1^2 + \eta_2^2 + \eta_3^2\right)} = \sqrt{\left(t^2 + v^2 + r^2 + c^2 + p^2 + e^2 + m^2 + d^2\right)} \qquad (1)$$

The ρ indicator represents the radius of a fraction of a spherical cap in the three dimensions space, or in the hyperspace of eight, and the goal is to determine the best value of $\acute{A}$ so that the untraceable project populations are separated from the traceable ones in the best way possible.
The "study case" data population can be seen over the Cartesian axis system η_1, η_2 and η_3 in Figure 2.a., and in Figure 2.b you can see the distribution of these data as a function of $\acute{A}$.

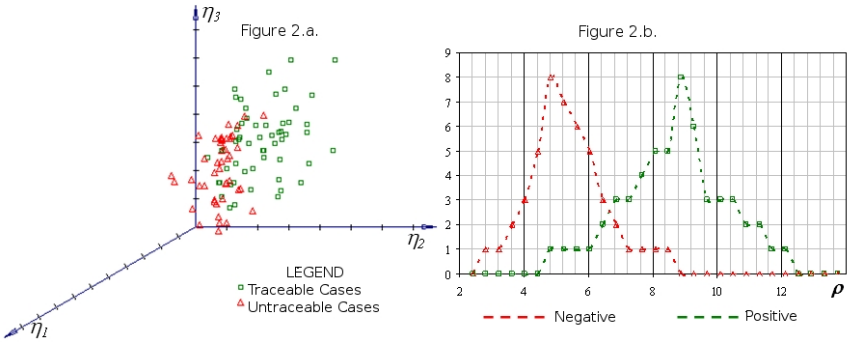


***Fig. 2**: a) Representation of the population of the batch of data from the study case and b) Distribution curves of the data population as a function of "**Á**"*

Even if proposed model "A" proved a good performance [7], the question arose of whether the metrics used to quantify the eight factors were the best ones. Stated in other terms, ¿is it possible to obtain a better discrimination of positive and negative cases if other metrics are used in the definition of ***Á***? If this is the case the populations shown in Figure 2.b. will be further separated and the model will sort them more effectively.

To answer this question we incorporated four variants to the original Model "A", summarized in Table 4:

Table 4: Definition of variable metrics in model "A" and their variants

Variable	Model A (original)	Variant B (linear)	Variant C (linear modified)	Variant D (semi-sigmoid)	Variant E (sigmoid)
t	5****PF*** / 1000	5****PF*** / 1000	5****PF*** / 1000	Sg(5****PF***/1000)	Sg(5****PF***/1000)
v	5****VU*** / 10	5****VU*** / 10	5****VU*** / 10	Sg(5****VU***/ 10)	Sg(5****VU***/ 10)
r	See (*)	1 + ***RE***/20	***RE***/16	Sg(1+ ***RE***/20)	Sg(***RE***/16)
c	1 + 0,16****CO***^{2}	1 + 0,8****CO***	***CO***	Sg(1+0,8****CO***)	Sg(***CO***)
p	***DP***	***DP***	***DP***	***DP***	Sg(***DP***)
e	***EF***	***EF***	***EF***	***EF***	Sg(***EF***)
m	***CMMI***	***CMMI***	***CMMI***	***CMMI***	Sg(***CMMI***)
d	***NA***	***NA***	***NA***	***NA***	Sg(***NA***)

$$(*)\ \boldsymbol{r} = 1 + 0{,}025*(\boldsymbol{RE} + 0{,}0125*\boldsymbol{RE}^2)$$

In "D" and "E" variants a sigmoid expression is adopted in order to quantify the variables of traceability factors. The object of the sigmoidal function is to polarize the results to both ends of the interval. A change in the origin and an

amplification factor were incorporated in order to implement the expression, providing results in the [0,5] interval for values of the argument 0 d x d 5, as represented in Eq.2:

$$Sg(x) = 5 / (1 + \exp(-2*(x-2,5)) \quad (2)$$

The analysis of the metrics presented in Table 2 proves that:

Variable *t*: For the model "A" and variants "B" and "C" we propose a linear formula, while in variants "D" and "E" the result of the linear formula is affected by the sigmoid expression (Eq. 2).

Variable *v*: The same criterion of the previous variable is adopted: a linear formula for the first three cases and a sigmoid correction (Eq. 2) for the last two.

Variable *r*: A polynomial formula is proposed for Model "A", a linear expression providing results in [1..5] interval in variant "B", a linear expression with results in the [0..5] interval in variant "C", the sigmoid correction of variant "B" is assigned to variant "D" and the sigmoid correction of "C" is assigned to variant "E". In figure 3 we represent them as a function of the reuse percentage ***RE*** in the 0 – 80% interval in order to facilitate the interpretation of the expected effect with the different expressions.

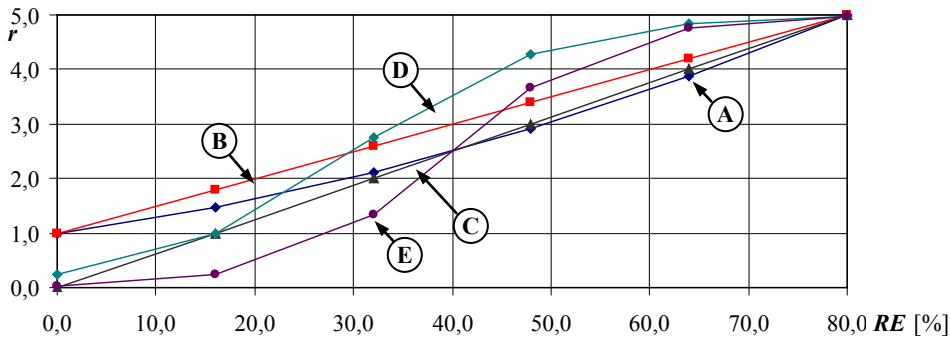


***Fig. 3**: Variable **r** evolution as a function of the reuse **RE** percentage according to the expressions proposed for the metrics of the different variants.*

Two points will be considered to illustrate this; RE = 16 % and RE = 64 %, and the values of ***r*** obtained in each case are shown in Table 5, where E case presents an antisymmetric polarization and D case an asymmetric polarization.

Table 5: Values of *r* obtained with different metrics in two points. (***RE*** = 16% and 64%)

RE = 16 %		***RE*** = 64 %	
Mod.	***r***	**Mod.**	***r***
A	1,48	A	3,88
B	1,80	B	4,20
C	1,00	C	4,00
D	0,98	D	4,84
E	0,24	E	4,76

Variable *c*: In the different models the proposed formulas are similar to the previous case: "A" polynomial, "B" linear in the [1..5] interval, "C" linear in the [0..5] interval and the sigmoid corrections of "B" and "C" in the last two. They are represented in Figure 4.

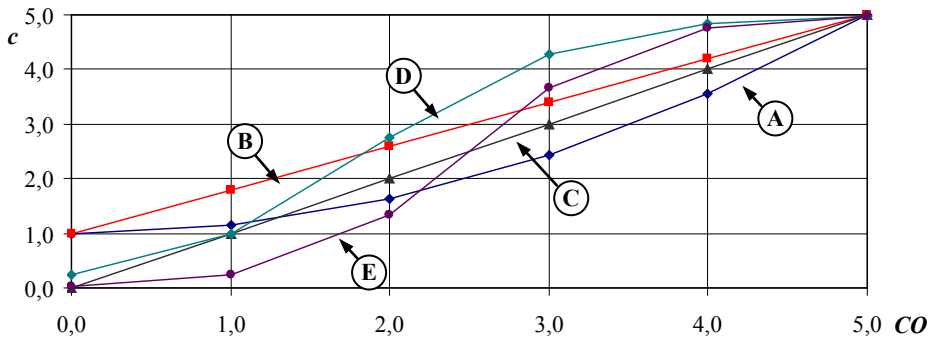


Figure 4*: Variable* ***c*** *evolution as a function of the reliability CO according to the proposed expressions for the metrics of the different variants.*

Variable *p*: For the model "A" and variants "B", "C" y "D", the direct assignment of Project term ***DP*** to ***p*** is proposed, while in variant "E", the result of applying the sigmoid expression to the ***DP*** term is assigned. (Eq. 2).

Variable *e*: The same criterion than for variable ***p*** is applied, in this case for ***EF***.

Variable *m*: The same criterion than for variable ***p*** is applied, in this case for the maturity level ***CMMI.***

Variable *d*: The same criterion than for variable ***p*** is applied, in this case for autonomy level ***NA.***

Once the variants to be considered are established, the next step was to define the comparing tool in order to identify the best model. The ROC

analysis is advisable for this case, which is a technique destined to evaluate dichotomous sorter and recently has experienced an important diffusion in much varied fields such as bioengineering, automatic learning and data mining, even though it is still rarely used in software engineering. The ROC analysis allows: *i)* to be able to objectively choose the best between several sorting models and *ii)* to optimize the tuning of the chosen model. In this case it is used with the first goal in mind.

If we consider the populations of positive and negative data, such as the ones represented in Figure 2.b., when defining a cutoff value for $\acute{A}$ indicator (called $\acute{A}_c$) four groups of data are immediately established, which are represented in Table 6. These groups result in the definition of the parameters that are the basis of the ROC analysis, as shown below.

Table 6: Results obtained with a sorter and parameter definition.

Sorting Model		Real Condition	
		Positives	**Negatives**
Results with cutoff value $\acute{A}_c$	**Positives**	**True positives (*VP*)**	**False positives (*FP*)**
	Negatives	**False Negatives (*FN*)**	**True Negatives (*VN*)**

$$Sensitivity = FVP = \frac{VP}{VP + FN} \quad (3)$$

$$Specificity = FVN = \frac{VN}{VN + FP} \quad (4)$$

$$Specificity = FVN = 1 - FFP \quad (5)$$

$$Accuracy = \frac{VP + VN}{VP + FN + VN + FP} \quad (6)$$

The ROC curves represent the sensitivity (Eq. 3) as a function of ***FFP*** (Eq.5) and the best model is the one that encloses more area under this curve (***AUC***) [9][10][11]. It should be noted that the ROC curves present the advantageous property of being insensitive to changes in the proportion of positive and negative instances that may be in a batch of data. The area under the curve ROC (***AUC***) also exhibits important statistical properties of its own.

3. Presentation and discussion of the obtained results

The comparison between the basic model and its variants was established based on the areas under de ROC Curves; and the accuracy of each model were also considered (Eq. 6). The "study case" used [7] consists in a batch of 102 samples, including 55 projects successfully traced and 47 project with negative results. This data were already represented in Figures 2.a and 2.b. In Figure 5 we present the curves of accuracy and the ROC curves from the basic model and its variants.

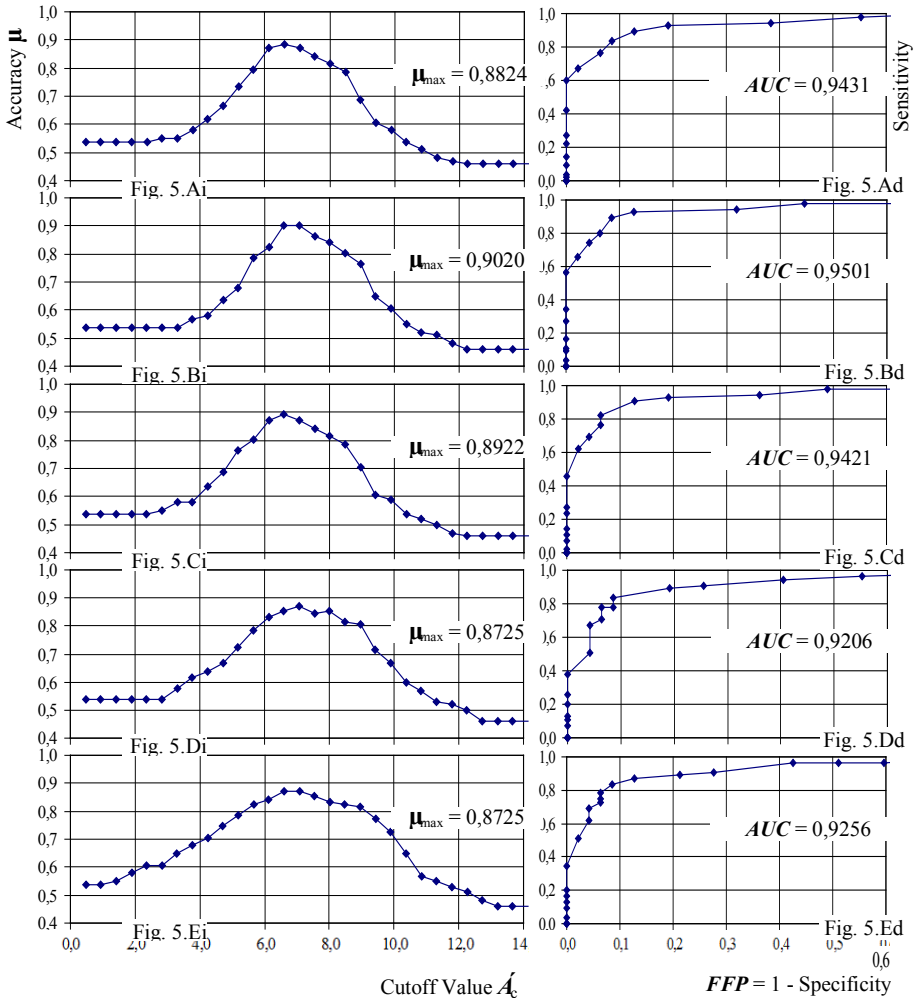


Fig. 5*: Left, for the basic model and its variants, representation of the accuracy* **μ** *as a function of the cutoff value* $\acute{A}_c$ *(5.Ai, 5.Bi, 5.Ci, 5.Di, 5.Ei) and right, partial representation of the corresponding ROC curves (5.Ad, 5.Bd, 5.Cd, 5.Dd, 5.Ed) for the same models.*

Finally, the Gini coefficient (***G***) is adopted as a comparison indicator, which is a measure of statistical dispersion proposed by the Italian Corrado Gini and is directly related to the area under the ROC curve (***AUC***):

$$G = 2.AUC - 1 \tag{7}$$

The ***G*** coefficient is applied to the study of inequalities in several fields, such as sociology, economics and engineering, among others, where the greater values of ***G*** are associated to best performances, being the possible maximum 1 (0 d ***G*** d 1).

A summary of the obtained results is presented in Table 7 in order to facilitate the comparison of the models: Area under the ROC curves ROC (***AUC***), Gini coefficient ***G***, maximum accuracy μ_{max} and cutoff value $\acute{A}_c$ corresponding to each case. The second and fourth column values (***AUC*** and μ_{max}) are also shown in Figure 5 graphs.

Table 7: Summary of sorting models performance.

Model	*AUC*	*G*	μ_{max}	$\acute{A}_c$
A	0,9431	0,8862	0,8824	6,5997
B	0,9501	0,9002	0,9020	6,5997 - 7,0711
C	0,9421	0,8842	0,8922	6,5997
D	0,9206	0,8412	0,8725	7,0711
E	0,9256	0,8512	0,8725	6,5997 - 7,0711

The following considerations are made by analyzing the obtained results with the different models on the study case:

a) In ***AUC*** the greatest difference is between variants B and D, being 3,1%. In accuracy the greatest difference is 3,3% and is presented by comparing the B variant with D and E variants. This means that the important differences in the proposed metrics has a lesser significant impact in the models performance.

b) The best cutoff value $\acute{A}_c$ is found between 6,5997 y 7,0711 in every case, which represents an environment of 6,7 % considering the greater value as a reference. If we consider that $|\acute{A}_c|_{max} = 14$, we can conclude that the different models define the cutoff value with a dispersion of 3,4 %. Thus metric variants also exhibited a scarce impact in the prediction of the best cutoff value.

c) Both the accuracy curve and the ROC curve of Model A present the softest evolutions, without measurable discontinuities in their gradients.

d) On the contrary, variants B to E lead to ROC curves with sudden changes of gradient, which is frequent in ROC curves of fairly small populations as the one considered.

e) B and E Models present two cutoff values with the same accuracy, the first with the highest accuracy and the second with one of the two lowest.

f) The models that are not affected by the sigmoid function (A, B and C) are the ones that present the greater areas ***AUC*** in relation to variants D and E that have their variables partially or totally affected by the sigmoid function.

g) B and C variants, that have linear metrics for the eight variables, are the ones that present the highest accuracy. B variant does it in a broader $\acute{A}_c$ interval.

h) It can be seen that the effect of the sigmoid function of tending to polarize the values of the arguments did not have the expected effect in this kind of model, since it had been anticipated that it would contribute to sort more easily both classes of projects populations (traceable and untraceable), which is not proven by the results.

i) Polynomial metrics of model A did not have an important effect either, considering that they provided expectative when chosen in the development of this first model [6], but were overcome by linear metrics.

j) Given its direct relation with ***AUC***, the Gini coefficient ***G*** does not provide any new information but represents a traditional indicator of the sorters efficacy.

5. Conclusions and future work

When attempting to understand the causes that block the effective application of traceability in software industry we entered in a complex and thrilling world that, in a certain way, answers to the strictness of math and, at the same time, is affected by the conduct, frequently ambiguous, of human beings.
In this context certain factors were identified that were considered determinant, metrics were proposed to quantify them and models have been developing to try to reproduce the scenarios in which traceability systems and their results apply. As soon as the first models were tested the question arose about which were the best metrics in order to prove the efficacy of this tools in the prediction of software projects traceability. In this work we proposed five sets of metrics and we compared their performances with a study case adopted as a reference, arriving at the conclusion that linear metrics are the most convenient. The next steps will be oriented to corroborate these conclusions with other study cases, for which we are side working in obtaining more real case data, in the necessary quantity, quality and variety. The possibility of having an effective prediction model of project traceability amply justifies the effort that is being done.

References

1. Blaauboer, F., Sikkel, K., Aydin, M. (2007). Deciding to adopt requirements traceability in practice. Proc.of the 19th Int. Conf. on Advanced Infor. Systems Engineering. Springer-Verlag.
2. Kannenberg, A., Saiedian, H. (2009). Why Software Requirements Traceability Remains a Challenge. CrossTalk: The Journal of Defense Software Engineering. July/August, 14-19.
3. Ramesh, B. (1998). Factors influencing Requirements Traceability Practice. Communications of the ACM. 41(12), 37–44.
4. Giró, J., Vazquez, J., Meloni, B., Constable, L., Jornet, A. (2010). Aseguramiento de la Trazabilidad en Projectos de Desarrollo de Sistemas

de Software. Projecto de Investigación, Secretaría de Ciencia y Tecnología, Código SCyT 1214/10.
5. Giró, J., Vazquez, J., Meloni, B., Constable, L., Jornet, A. (2011). Modelos para anticipar la factibilidad de que un proyecto de desarrollo de software sea trazable. Workshop de Ingeniería de Software, CACIC 2011. Universidad Nacional de La Plata, 837-846.
6. Giró, J., Vazquez, J., Meloni, B., Constable, L., Jornet, A. (2012). Hacia una respuesta al interrogante de si será factible trazar un cierto proyecto de desarrollo de software. Informe Técnico 2012/01, Proyecto 1214, SCyT, FRC, UTN.
7. Giró, J., Vazquez, J., Meloni, B., Constable, L., Jornet, A. (2012). Uso del Análisis ROC para anticipar la conveniencia de trazar proyectos de software. Workshop de Ingeniería de Software, CACIC 2012. Universidad Nacional del Sur, Ciudad de Bahía Blanca.
8. Powers, D. (2007). Evaluation: From Precision, Recall and F-Factor to ROC, Informedness, Markedness & Correlation. Technical Report SIE-07-001, School of Informatics and Engineering, Flinders University, Adelaide, Australia.
9. Fawcett, T. (2006). An introduction to ROC analysis. Elsevier ScienceDirect, Pattern Recognition Letters, 27 - 861–874.
10. Shin Y., Huffman Hayes J., Cleland-Huang J. (2012). A framework for evaluating traceability benchmark metrics. TR: 12–001, DePaul University, School of Computing.
11. Biggerstaff, B. (2000). Comparing diagnostic tests: a simple graphic using likelihood ratios. Statistics in Medicine 19 (5) 649–663.

Model for context-aware applications (MASCO): A case study for validation

EVELINA CAROLA VELAZQUEZ[1], ARIEL NELSON GUZMAN PALOMINO[1], MARÍA DEL PILAR GALVEZ DÍAZ[1], NÉLIDA RAQUEL CACERES[1]

Universidad Nacional de Jujuy – Facultad de Ingeniería,
San Salvador de Jujuy,
Jujuy, Argentina
{mdpgalvezdiaz,nrcaceres}@fi.unju.edu.ar

Abstract. *This paper presents the implementation of a case study to control in an automated way the functioning of a greenhouse using the MASCO model for context-aware applications. The greenhouse includes sensors and actuators as well as the values of context variables considered for a normal crop development show dependence. A simulation work is performed, a mathematical formula is derived to handle the interdependence of context variables, and design patterns are used to address the complexity. In conclusion the design patterns applied allow to maintain the structural integrity and the flexibility feature of the model.*

Keywords: *Software engineering – Models – Patterns –Context Aware*

1. Introduction

Context -aware applications allow determining what happens between the system and its environment, determining what, how and when external events arise and to which the system must respond. In this context, the implementation of a process of automatic control of a greenhouse climate is presented using the MASCO model, with the aim of determining the model adaptability to the case put forward and its flexibility to reach a stable state in the system. In section 2 the MASCO model is described, in section 3 the case study: Greenhouse is described, in section 4 the characteristics of the implementation performed and the design patterns used to give a solution to decision making are specified, in section 5 the conclusions are put forward, and in section 6 the references are presented.

2. MASCO Model

The MASCO model (Model providing services for context- aware applications) originated in an extension of the model presented in Gordillo [1] which considers the variable of context location and the user profile as

sensitive services, with reference to the framework Context Toolkit based on Widgets [2] and the CIM automation model-Computer Integrated Manufacturing- [3] which is a reference model that supports industrial applications.
In context- aware applications, in which there is more than one context variable, the system must manage different behaviours or offer different services based on the value or state change of one or more context variables or their combination. Besides, an entity object or context variable object of the application may have to relate to one or more objects each of them representing a context variable or entity. [4]
The hardware for sensing evolves constantly. The sensing rules vary according to the hardware capabilities. The sensed data must be interpreted and the application should appear transparent to these processes. This is achieved by decoupling the sensors from their logic and what is concerned with the application [1]. MASCO, that takes into consideration all the situations put forward, is presented in Fig. 1, the shaded areas represent the components presented in [1] and the dotted areas represent the modifications performed throughout the works presented in [4], [5], [6], [7].
MASCO is a layered model, in which five layers, which are described below, are identified: [6]

- Aplication Layer: objects of the application domain are found.
- Context Layer: encloses the objects required for processing the context data.
- Service Layer: encloses the objects required for providing external as well as internal services to the system.
- Sensing Concern Layer: deals with the interpreting or translating of the data originated in the Hardware Abstractions Layer.
- Hardware Abstractions Layer: in this layer the objects representing sensors and actuators are grouped.

3. Case study: Greenhouse

The case study corresponds to the implementation of a climate control system for a greenhouse whose processes are automated, which performs the monitoring of the interest parameters through sensors, checks greenhouse internal environment conditions based on the sensed values, and corrects them using actuators upon installed automatic devices (side windows, drip irrigation, etc.), so that the weather conditions are optimum for the suitable development and growth of crops located there. For this a unique process consisting in the control of interest parameters centralized in the monitoring of the four main variables of the plant photosynthetic process is established, which constitute context variables for the MASCO model which must be checked and corrected when outliers occur. These variables are: luminosity, temperature, carbon dioxide, and relative humidity.
Luminosity: It is the amount of radiation that is projected by an energy source. In the case study it is provided by the sun or an artificial source which

ensures that the plant receives the required amount of radiation to optimize the process of photosynthesis.

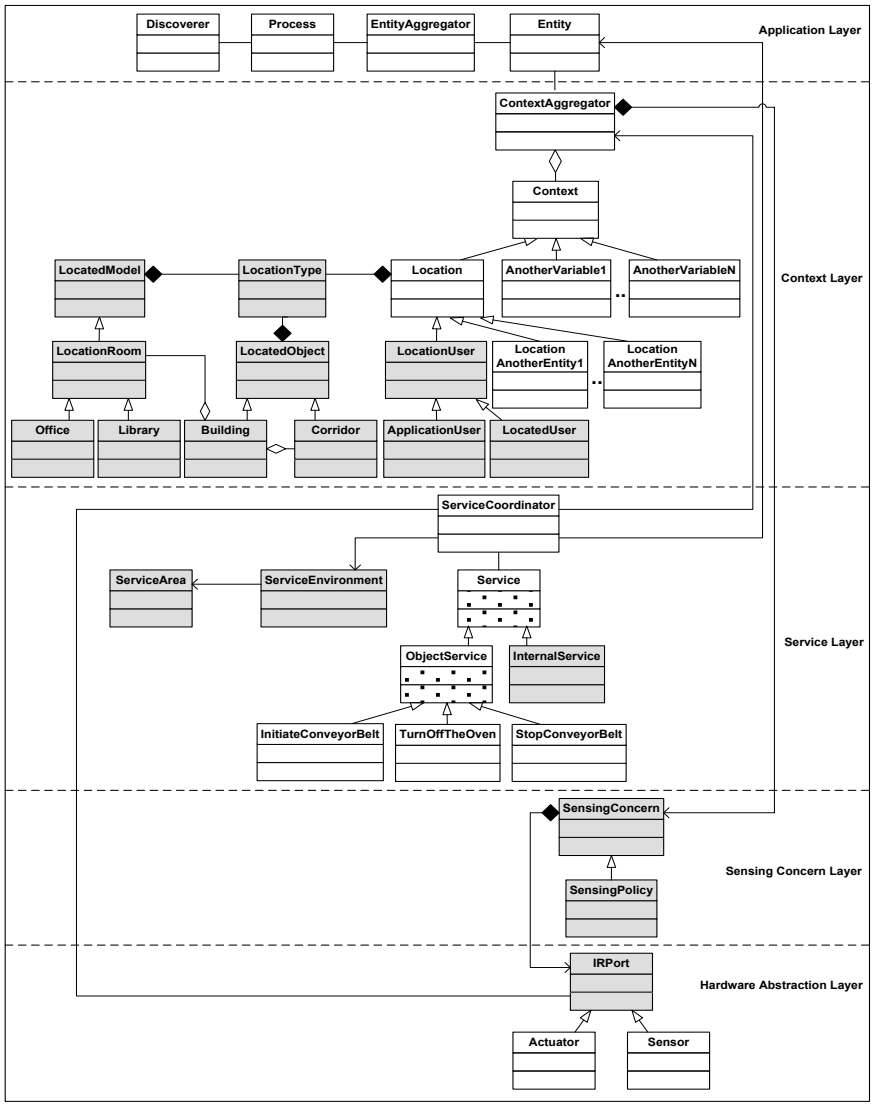


Fig. 1. *MASCO model*

Temperature: this magnitude influences on the growth and development of plants. The general optimum temperature for plants is between 10 and 20º C. For its handling it is important to know the needs and limitations of the crop species. Besides, the values to be reached for optimum crop growth and its limitations must be taken into account:

- Lethal minimum temperature: that below which injuries are caused to the plant.
- Biological maximum and minimum temperatures: they indicate values, above or below which respectively, it is not possible for the plant to reach a certain vegetative phase, such as flowering, fruiting, etc.
- Nighttime and daytime temperatures: they indicate the recommended values for a suitable plant development.

The temperature inside the greenhouse is according to solar radiation, ranging between 200 and 4000 W/m^2; when solar radiation is insufficient to maintain the required temperature, the decision to activate the artificial energy source respecting the required values of the other context variables is made. A dependency relationship between Temperature and Luminosity is established.
Carbon Dioxide: the atmosphere carbonic anhydride (CO_2) is the essential raw material of plants chlorophyll function. The normal CO_2 concentration in the atmosphere is 0.03%. This index should be increased to 0.1-0.2%, limits, when the rest of the plant production factors are optimum, if maximization of plants photosynthetic activity is desired. Concentrations over 0.3% are toxic for crops. The recommended CO_2 levels depend on the species or crop, solar radiation, ventilation, temperature and humidity. The optimal assimilation is between 18°C and 23º C, descending over 23-24º C. With regard to luminosity and humidity, each plant species has a different optimal value.
However, one cannot speak of a good photosynthetic activity without an optimal luminosity. Light is a limiting factor, and thus, the CO_2 uptake rate is proportional to the amount of light received, besides depending as well on its own CO_2 concentration available in the plant atmosphere. It can be said that the most important period for carbon enrichment is at noon, since it is the time at which the maximum luminosity conditions occur. Here a dependency relation between carbon dioxide concentration and luminosity is also set.
Relative Humidity (HR): humidity is the water mass in volume unit or in air mass unit, amount of water in the air, according to the maximum that would be able to contain the same temperature. An important feature for the case study is the inverse relationship between Temperature and Relative Humidity: if the temperature is high the HR decreases, otherwise the HR increases, which involves finding out the balance between both quantities in order to optimize plant photosynthesis.
Relative Temperature of air is a climatic factor that can modify the final crop yield. When it is excessive plants reduce transpiration and decrease their growth, floral abortions occur due to pollen compactness and a larger development of cryptogamic diseases. Conversely, if too low, plants transpire in excess, may dehydrate, in addition to curd problems.

3.1 Relationship between context variables

Upon analyzing the behaviour of context variables considered for the case study, a strong link among them was observed, this determines the control

system condition that handles the interaction of variables to maintain the greenhouse stable conditions, this link is required to achieve a successful photosynthesis in plants. This interaction and the way to perform the control are being studied in this work.

Binding relationship of context variables:

- Luminosity:
  - Temperature. (Proportionally).
  - Carbon Dioxide. (Proportionally).
  - Relative Humidity. (Proportionally).
- Temperature:
  - Carbon Dioxide. (Proportionally).
  - Relative Humidity. (Inversely proportional).

This relationship can be generalized as follows:

$$t = f(l) \quad (1)$$

Where t (temperature) is a function f of l (luminosity) and:

$$c = g(t,l)\ a\ c = g(f(l), l)\ a\ c = g(l) \quad (2)$$

Where c (carbon dioxide) is a function g of t (temperature) and l (luminosity), but taking into account that t is also a function of l we can conclude that carbon dioxide depends on l and t, but with higher priority depends on l.

$$h = h(t,l)\ a\ h = (f(l), l)\ a\ h = h(l) \quad (3)$$

Where h (relative humidity) is a function h of t (temperature) and l (luminosity), as t depends on l, so h depends on l and t, but with higher priority on l.

From (1), (2) and (3) we deduce that there is a priority order of effects among context variables link: Luminosity, Temperature, Carbon Dioxide and Relative Humidity.

3.2 Sensors

To perform the monitoring of the values of context variables, simulated sensor values are used in a predefined period of time, which depend on the time at which a change is made on the value thereof so that it is representative for the expert's analysis and assessment. The saving of values takes place in the same period for the four context variables so as to combine them for their assessment and subsequent corrective action. The corresponding types of sensors were used to monitor the values of the four context variables, for the present work the use of a sensor for each of them is considered.

It is established that the entity to be considered is a plant from a particular species.

3.3 Actuators

Actuators are devices of objects automatic handling, that correct outliers of one or more context variables, thus for example, a side window, which is an object to reduce the temperature inside the greenhouse, has one or more

actuators, motors, allowing its regulated opening or closing. For the case study, actuators correspond to motors, servomotors, and other devices which command the objects used to correct values.

- Temperature: Side Windows, Refrigerating Fan, Refrigerating Paper, Heater Fan and Pump associated to a water heater tank to increase temperature.
- Relative Humidity: electric valves Pump, Irrigation Peak, Drip Irrigation.
- Lighting: Photocell with internal clock controller and light group.
- Carbon Dioxide Concentration: Ventilation dampers.

The greenhouse environment control is performed by simulating the sensing process, generating values for the four context variables from functions that respect their dependency relationship, thus for example, for the luminosity variable, values are generated according to the following function:

$$y = 0{,}0026x6 - 0{,}2115x5 + 6{,}6352x4 - 98{,}616x3 + 683{,}09x2 - 1599{,}1x + 1584 \quad (4)$$

Where y is a polynomial function grade 6, which is a trend line performed according to the actual graph drawn for current luminosity saved values recorded per hour in an autumn day. Thus, for saved values according to Table 3, we will have a graph and a trend line as shown in Fig. 2.

Table 3. Luminosity Values.

Time	0:00	1:00	2:00	3:00	4:00	5:00	6:00	7:00
Luminosity	500	500	1000	1000	2000	2000	3000	3000
Time	8:00	9:00	10:00	11:00	12:00	13:00	14:00	15:00
Luminosity	3000	3000	3000	3000	3000	3000	3000	3000
Time	16:00	17:00	18:00	19:00	20:00	21:00	22:00	23:00
Luminosity	3000	3000	3000	3000	3000	3000	1000	1000

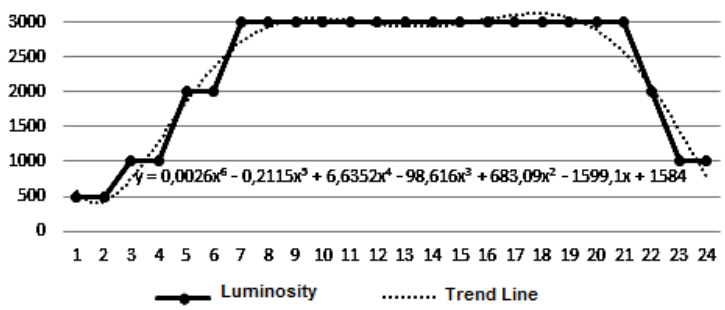


Fig. 2. *Trend Luminosity Function Graph of an autumn day based on actual sensed values.*

Likewise, the other 3 context variables and actuators usage, whose utilization affects the climatic conditions inside the greenhouse, are simulated.

3.4 Optimal conditions for plant growth

In general, optimal conditions for the better growth of plants demand a higher luminosity, which increases temperature, relative humidity and carbon dioxide.
However, the increase in temperature reduces relative humidity, making it necessary for some device to compensate these conditions so that the optimal values can be produced. Optimal conditions include higher luminosity, which is found naturally by solar radiation, being considered in amplitudes between 12 and 16 hours depending on the Season of the year and the region where the greenhouse is situated; also a high temperature between 10ºC and 20ºC, high relative humidity values, and carbon dioxide concentrations between 0.1 and 0.2%, which are established in Table 4. Nevertheless, it is worth highlighting that these are general conditions, since each plant of a particular species has its own optimal conditions.

Table 4. General optimal conditions for the process of photosynthesis

Sensed values	Optimal values for photosynthesis	Actuators
Luminosity → Higher	Higher	Turn on the luminaire when it gets dark.
Temperature → High	High	Verify if the temperature does not exceed 20ºC. The allowed range is [10ºC, 20ºC].
RH → Normal	High	Reduce temperature within the allowed range [10ºC, 20ºC], for relative humidity increase.
CO_2 → Normal	High	Increase temperature between [18ºC, 23ºC] for CO_2 increase.

4. Implementation features

For the case study implementation using the MASCO model, it was necessary to adapt it to the features mentioned in section 3. The work was performed layer by layer and patterns to support the decision making process about service invocation to check outliers were used, based on the context current conditions, that is, on the context variables analyzed as a group due to the existing dependency among them mentioned in section 3.1, besides looking for low coupling in the whole monitoring process, control, checking up, decision making, and carrying out of correction services. Next the patterns used in each MASCO adapted layer to the case study providing solutions to the problems that came up in each case are specified.

4.1 Hardware Abstraction Layer

The classes representing sensors to monitor the four context variables: Luminosity, Carbon Dioxide, Relative Humidity and Temperature, as well as the required actuators for the devices which regulate them increasing or reducing their values, as shown in Fig. 3 are specified.

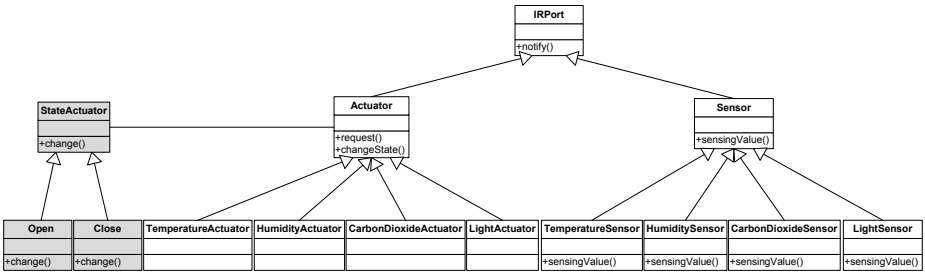


***Fig. 3**. Hardware Abstraction Layer*

The State pattern was implemented in the StateActuator class, shaded areas in Fig. 3, to determine the state in which each actuator is. Thus, their availability is established in order to call services upon them and perform regulating actions on devices such as windows, drip irrigation, etc. It was decided to use this pattern for it simplifies the state determination on actuators. Otherwise, it would mean putting sensors to the actuators adding complexity and mixing the saved values devices of context variables with those regulating them.
Open and Close classes determine the two possible values of actuators, this shows if the regulating device upon which it works is currently open or closed, allowing to call for correction functions in the classes corresponding to actuators. Besides, a legacy over the sensor class was specified, which allows to add different types of sensors.

4.2 Sensing Concern Layer

The policies to convert the data obtained by the sensors into data which can be processed and understood for later decision making are aspecified (Fig. 4). The SensingConcern class receives the sensors values by means of an Observer pattern implementation, which is determined from the MASCO model. SensingConcern does not know the time at which the values of the context variable are monitored, the Hardware Abstraction Layer sends the notification when the values are monitored, this is done at the same time for the four context variables to assure the joint evaluation of the environment condition. Besides, in order to apply a suitable value conversion policy to each monitored context variable a Strategy pattern is applied, thus, the SensingPolicy class determines the transformation rules.

Once the transformation of values is performed, the ContextAggregator class of the Context Layer is notified.

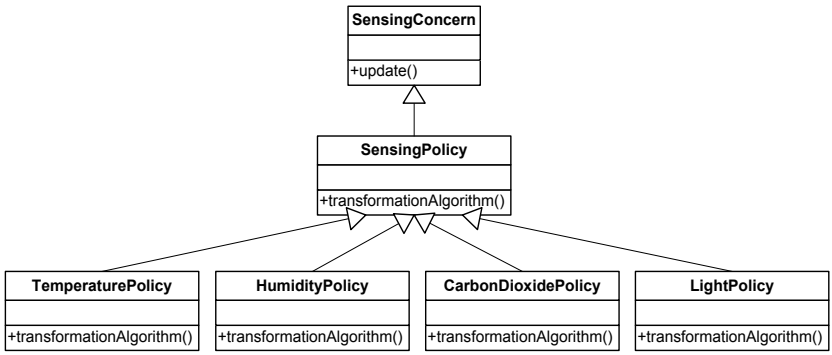


Fig. 4. *Sensing Concern Layer*

4.3 Context Layer

This layer makes the decision on which is the context variable over which service request will be invoked, leaving to the Service layer the decision of which services to request. To carry out this decision the State and Template Method patterns were implemented in combination, in addition to the patterns implemented by the MASCO model for the ContextAggregator class: Observer to receive the notification when the monitored values of context variables are transformed from the Sensing Concern layer, and Mediator to combine the values of context variables due to their dependence. In this layer the values of each context variable in their reference values are verified individually, through the Mediator pattern variables are combined and the whole context is checked by means of the Template Method pattern which is implemented in the ContextShapeDefinition class, this is because it is possible that the context variables coming across outliers are able to present a whole anomalous context corresponding to the most critical case, besides, a unique context according to a single context variable with outlier, whereas the opposite context occurs when the Temperature and Relative Humidity variables, which are inversely proportional, have outliers and their correction involves finding out a balance between two opposing values, according to what was stated in section 3.1. In order to determine whether a context variable has an outlier, the State pattern was used, changing its state after independent verification, and only if there are outliers the ContextAggregator class is notified for it to define the context shape and to call services according to the current anomalous context case. If no outliers come up services are not called and the next monitoring period is awaited to verify the system state. The added State and Template Method patterns are shaded in Fig. 5.

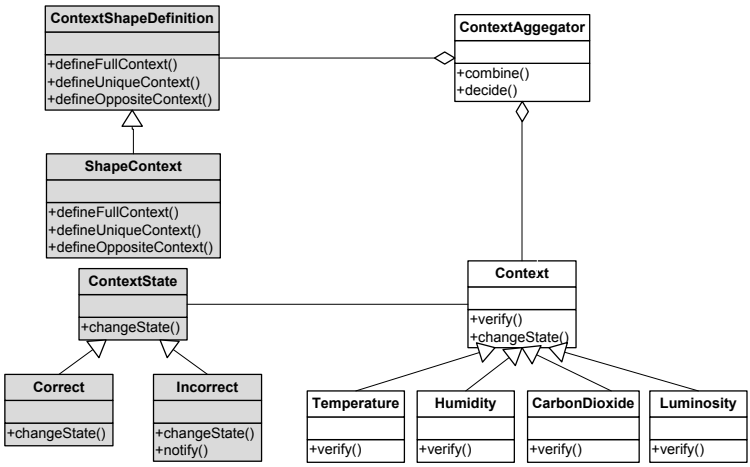


***Fig. 5**. Context Layer*

4.4 Service Layer

In this layer the process of deciding which services must be requested based on the decision made in the Context layer occurs. This is based on the higher priority context variable with outliers, since the dependency relationship ensures that altering a context variable alters the others according to section 3.1. The Strategy pattern, indicated in grey in Fig. 6 to perform the call for the specific services of increasing or reducing the values of that context variable was implemented. For this the ServiceCoordinator class makes the decision based on the Actuators state, as was described in section 4.1.

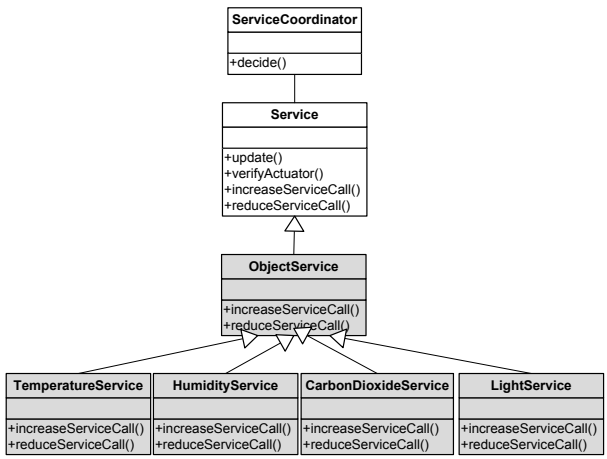


***Fig. 6**. Service Layer*

Once the services are called the Actuators state is changed in the HardwareAbstraction layer, and the next monitoring period is awaited where the action performed is validated. Should there occur a critical state of complete unchecked context in three iterations must be passed to manual control; because there is a risk of crop loss due to an extreme condition which cannot be corrected in an automated way.

5. Conclusions

Applying patterns to the model provided solutions to two major problems which came up when implementing the case study: the interaction among dependent variables and the decision about which services to request in order to correct outliers. Its use ensured the structural integrity of the model, enabling to adapt the layers incorporating the required classes, showing MASCO flexibility when applied to the case study.

The performance, which could have been an inconvenience when presenting the request for services which are chained to a lower layer from the application, was solved by providing direct communication among layers, as in the case of services requested from the Context Layer to the Service Layer, maintaining the particular features of a layered model.

6. References

1. Gordillo, S., Rossi, G., Fortier, A. (2006). Engineering Pervasive Services for Legacy Software. Proccedings of the 1st International Workshop on Software Engineering for Pervasive Services, Lyon.
2. Dey, A. K. (2001). Providing architectural support for building context aware applications. PHD Thesis. Georgia Institute Technology, USA.
3 García, M. (1999). Automation of Industrial Processes. Politechnic University of Valencia, Spain.
4. Quincoces, V.E., Gálvez, M.P., Cáceres, N.R., Vega, A.A., Ramos, H. O. (2009). Extending of a layered model providing services for context aware applications. Researches at NOA Faculties of Engineering, ISBN 978-987-633-041-1.2009, Vol I, pp. 35-40, Cap. IV. EUNSa, Salta.
5 Quincoces, V.E., Gálvez, M.P., Cáceres, N.R., Vega, A. A. (2010). A Model providing services for context aware applications: Validation at early stages. Researches at NOA Faculties of Ingeneering, pp. 481-486, EdiUNJu, Argentina.
6 Gálvez, M.P., Quincoces, V.E., Cáceres, N.R., Vega, A.A. (2010). Refining of a layered model providing location services for context aware applications. III International Congress on Telecommunications, Information and Communication Technologies, Quito.
7 Gálvez, M.P., Brouchy,C., González, O., Cáceres, N.R., Quincoces, V. E. (2011). A Model providing services for context aware applications

(MASCO): Interaction among entities. Researches at NOA Faculties of Ingeneering, pp. 1103-1109. Científica Universitaria, UNCa, Argentina.

X

Database and Data Mining Workshop

A Novel Language-Independent Keyword Extraction Method

GERMÁN AQUINO[1], WALDO HASPERUÉ[1,2], CÉSAR ESTREBOU[1] AND LAURA LANZARINI[1]

[1]III-LIDI. School of Computer Science. UNLP. Argentina
[2] CONICET scholarship
{gaquino, whasperue, cesarest, laural}@lidi.info.unlp.edu.ar

Abstract. *Obtaining the most representative set of words in a document is a very significant task, since it allows characterizing the document and simplifies search and classification activities. This paper presents a novel method, called LIKE, that offers the ability of automatically extracting keywords from a document regardless of the language used in it. To do so, it uses a three-stage process: the first stage identifies the most representative terms, the second stage builds a numeric representation that is appropriate for those terms, and the third one uses a feed-forward neural network to obtain a predictive model. To measure the efficacy of the LIKE method, the articles published by the Workshop of Computer Science Researchers (WICC) in the last 14 years (1999-2012) were used. The results obtained show that LIKE is better than the KEA method, which is one of the most widely mentioned solutions in literature about this topic.*

Keywords: *Text Mining, Document characterization, Back-propagation, WICC.*

1. Introduction

Text mining presents interesting challenges to solve, since the lack of structure in the texts analyzed makes it difficult to extract information from them. Nowadays, given the large number of texts that are published each day, be these scientific articles, books, journals, periodicals or web pages, facing these challenges can prove to be interesting, as well as developing strategies that allow obtaining information from relevant texts.

One way of briefly describing the topic of a document is by means of a list of keywords. The keywords in a document are of the utmost importance, since they allow carrying out several tasks, such as searching for a specific topic, classifying documents, clustering [1], summarization [2] [3] [4], etc.

Even though most of the times the author of the document is the one in charge of proposing the list of keywords, as in the case of scientific publications, there are other times when this list is not present at all and, therefore, it would be interesting to have an automated method that can propose a list of keywords by analyzing the text of the document.

Within text mining, there have been various alternatives proposed for the task of extracting keywords. There are statistical methods that typically do not have prior training with the documents; in such cases, only statistical information is collected from the words that are present in the document to identify which of them can be chosen as keywords. The most widely used statistical methods include TF-IDF [5] [6] [7] [8], word co-occurrence [9], etc.

On the other hand, there are machine learning-based methods that, from a given corpus, carry out a training process and generate a model that allows performing classifications afterwards or, in the case of keyword extraction, establishing which words in the document are candidates to be chosen as keywords. In these cases, each document in the initial corpus must have a list of keywords that are used as positive cases during training. Some of the machine learning methods used in this type of tasks are Naïve Bayes [8] [10], Support Vector Machine [11], etc.

The methods that analyze the linguistic aspects of the documents are those that offer the most interesting solutions, since they combine lexical analysis, syntactic analysis, etc. [12] [13]; however, their disadvantage is that they are strongly dependant on the language used to write the documents.

One of the main concepts pertaining to the specific task of extracting keywords from documents is that of n-grams. Any word within a document is a unigram, while any sequence of two or more words forms an n-gram, where *n* indicates the number of words in the sequence. When extracting keywords, any n-gram in the document is a potential keyword for that document. Most of the techniques that carry out this task perform calculations and measurements on each n-gram in the document, and then process them by means of a machine learning technique [14] or by assigning a given score [15] to obtain a model that can be used as predictor for future documents.

In order to extract keywords, some methods require a corpus from which to generate a first model [7] [8], while others do so from a single document [9]. There are techniques that carry out numeric and/or statistical calculations for all n-grams in the document [16] [17], while others exploit certain linguistic information [12] [13]. Most of the existing strategies pre-process the documents with stemming and word filtering techniques by means of a stop-word list.

In this paper, a novel method is proposed, called LIKE (Language Independent Keyword Extraction), which uses texts from documents from a given corpus to obtain a model that can be used to extract keywords. To this end, it uses a three-phase algorithm. The first phase consists in extracting any n-grams that are detected as candidates for keyword, the second phase calculates a set of numerical features for each n-gram that was detected, and the third and final stage uses those features to produce a model by training a feed-forward network. This trained network is used as model to decide, given a new document, possible keywords. LIKE is independent from the source language of the documents, provided that the same language is maintained throughout each individual document, since it does not carry out the usual pre-processing steps of stemming or word filtering using a stop-word list.

This article is organized as follows: in Section 2, LIKE is described; in Section 3, the results obtained in the experiments carried out are presented; and in Section 4, conclusions and future work are presented.

2. LIKE

The method proposed in this paper, called LIKE, is a three-phase method that allows extracting a list of keywords by analyzing the text in a document. LIKE is independent from the source language of the documents, since it does not carry out the pre-processing steps of stemming or word filtering using a stop-word list. LIKE analyzes the documents in a document corpus to train a back-propagation network that will then be used as model to determine the list of keywords for new documents.

LIKE starts by identifying all existing n-grams in each document in the corpus. Since the number of all possible n-grams would be excessively high, a strategy is required to reduce this number. In this proposal, the method presented by [18] is used, since it allows identifying a much lower number of n-grams.

During the second stage, each n-gram obtained in the previous stage is transformed into a features numeric vector; these vectors are labeled as one of two classes of data. One class includes the vectors corresponding to those n-grams that are part of the list of keywords proposed by the author(s) of the document, while the other class is formed by the vectors corresponding to all remaining n-grams. The third stage consists in using these two data classes to train a back-propagation network.

2.1 Phase 1. N-gram Extraction

The first phase of the method proposed consists in identifying the n-grams in the corpus. For each document, all existing n-grams are extracted. An n-gram is considered to be valid if it is formed by consecutive words within the same sentence with no punctuation marks between any of them.

In general, the number of existing n-grams is excessively high. For the tests carried out for this work, which has a corpus of 96 documents, more than 580,000 n-grams can be extracted. Therefore, a strategy to identify a lower number of n-grams is required. In this proposal, the algorithm presented in [18] is used. This strategy, inspired in the Apriori algorithm, builds sets of elements from other smaller sets. In this algorithm, the maximum n value (number of words in the n-gram) and the minimum occurrence frequency for each n-gram have to be determined. N-grams are built from the (n-1)-grams that meet the requirement of a minimum specified frequency. To do this, it is assumed that an n-gram whose frequency is k is built from the intersection of two (n-1)-grams whose frequency is at least k, i.e., an n-gram cannot be more frequent than its parts. For each n-gram, the first n-1 and last n-1 words are taken, and it is checked that these (n-1)-grams meet the minimum allowed

frequency criterion. If this criterion is not met, the n-gram is discarded. Finally, there are *n* runs on the text, first to obtain 1-grams, then 2-grams based on these, then 3-grams, and so forth.
The use of this strategy in LIKE allows identifying a low number of n-grams in each document. In the experiments that were carried out for this work, the total number of n-grams for the entire corpus was reduced to little more than 70,000.
The result of this phase is a list of n-grams, which are then labeled. Once this list of keywords is known for each document, a label is assigned to each n-gram to indicate if the n-gram is a keyword or not. Thus, a two-class set of data is generated.

2.2 Phase 2. N-gram Characterization

The purpose of this phase is converting each of the n-grams that were identified in the previous phase into a features vector. In this article, we propose that the eight features detailed below are calculated.

i) TF (Term Frequency): TF is perhaps, together with IDF, one of the descriptors most widely-used to characterize n-grams. Term Frequency is the number of times the n-gram occurs in the document divided by the total word count of the document.

ii) IDF (Inverse Document Frequency): It is the ratio between the number of documents in the corpus that include the n-gram $d(g)$ and the total number of documents D.

$$IDF(g) = \log\left(\frac{D}{d(g)}\right)$$

iii) First occurrence of the term: This represents the relative position of the first occurrence of the n-gram. It is calculated as the number of words before the first occurrence of the n-gram divided by the total word count of the document.

iv) Position within the sentence: This is the relative position of the n-gram in the sentence that contains it. The same as the previous one, it is calculated as the number of words before the occurrence of the n-gram in the sentence divided by the total word count of the sentence itself. If the same n-gram appears several times in different sentences in the same document, then all n-gram occurrences are averaged.

v) Part of the title: This feature is a binary value that indicates if the n-gram appears in the title of the document or not.

vi) Part of the n-gram present in the title: This feature (only valid for n-grams with two or more words) counts the number of words in the n-gram that also appear in the title, regardless of the order of the words in the title. This number of occurrences is normalized by the number of words in the n-gram. In the case of

unigrams, the same as the previous feature, this is a binary value that indicates either presence or absence.

vii) NSL (Normalized Sentence Length): This is the length of the sentence where the n-gram appears divided by the length of the longest sentence in the document. If the n-gram appears in more than one sentence in the document, all occurrences are averaged.

viii) Z-score: This is a statistical measure that normalizes the frequency of the n-gram. It requires knowing the mean and standard deviation of the frequency for each n-gram.

$$Z\text{-}score(g) = \frac{freq(g) - \mu}{\sigma}$$

If the n-gram appears in more than one sentence in the document, all occurrences are averaged.

Of the eight features proposed, only two (IDF and Z-score) require the corpus in order to be calculated.

The result of this phase is a features vector for each of the n-grams identified in the previous phase.

2.3 Phase 3. Creating the Model

The third phase of the method proposed consists in creating a model that can learn from a given corpus and allows classifying n-grams from new documents as possible keywords or not. The prediction model is built by training a back-propagation network.

The problem that arises when trying to use the set of vectors obtained in the previous phase as data for training the back-propagation network is that the classes in this data set are not balanced, since the "not a keyword" class has a lot more elements than the "is a keyword" class. In the corpus used to carry out the experiments presented here, the ratio of elements in both classes was approximately 150 to 70,000.

When there is a data set with unbalanced classes, training a back-propagation network is not an easy task, since the training set prevents the generation of a model that can accurately predict the data in the minority class. In light of this problem, several solutions have been proposed ([19] [20] [21]). In particular, the solution described in [21] proposes that, before the training process, a clustering operation is performed on the data in the larger class in order reduce its number of elements. In this work, the idea in [21] is used – the data in the larger class are clustered using the k-means algorithm.

Be u the number of data present in the minority class, the value of k is then established as $k=u/10$. A clustering of k clusters is performed, and 10 random elements are extracted from each resulting cluster. These $10*k$ elements thus selected form a new data set that replaces the original data from the majority class. Following this methodology, the back-propagation network can be trained using a data set whose classes have similar numbers of elements.

To train the back-propagation network, the classic algorithm is used. After several tests and empirical observation, a decision was made to use seven neurons for the hidden layer, the logsig function as transfer function for the hidden layer, the tansig function for the output layer, an alfa of 0.25, and a maximum of 1,000 iterations. The best results were obtained with this configuration.
The result of this training process is a model that can predict keywords for new documents. The procedure to establish the keywords for a new document is as follows: first, the n-grams are extracted as described in Section 2.1; then, features vectors are calculated for each n-gram as explained in Section 2.2; and finally, these new vectors are presented to the trained network to determine if a given n-gram is a keyword for the document or not.

3. Results

The method proposed here was tested using as corpus all papers submitted to WICC (http://redunci.info.unlp.edu.ar/wicc.html) between 1999 and 2012. Only those articles written in English were included in the corpus (96 articles). The rationale for using only articles written in English was that, at a later stage, the results obtained with this method would be compared with those obtained with other keyword extraction method that is widely used in the literature: KEA [8]. Even though KEA can be adapted to work with languages other than English, since it depends on a stemmer and a stop-word list, those developed by the authors were used, which are in English.
KEA [8] is an automated keyword extraction algorithm that identifies candidate words by using lexical methods to calculate a set of features, and then apply an automated learning algorithm that allows predicting which candidates are good keywords.
The same as LIKE, KEA builds a prediction model using a training corpus with specified keywords, and it then uses this model to extract keywords from new documents.
KEA allows the free extraction of keywords, as well as the extraction of keywords using a vocabulary list that is controlled by means of a thesaurus. For these tests, the first mode was used, establishing as parameter a number of three keywords per document. In order to train KEA, a stop-word list and a stemmer are required. The stop-word list contains words of low semantic content (conjunctions, articles, prepositions, etc.) that should not be considered as keyword candidates.
The first step in the KEA method consists in filtering out the words that appear in the stop-word list, and then apply a stemming process to reduce to their syntactic root all n-grams that were not filtered out. The next step is to calculate the features of all candidate words, which include: TF-IDF, the initial position of the n-gram in the text and the length of the n-gram (the number of individual words that form the n-gram). Based on this representation, KEA uses Naïve Bayes as learning algorithm.

Both LIKE and KEA were trained using the same corpus. From that corpus, some documents were selected randomly for the training stage and the rest were used for testing. For each test, accuracy, recall and f-measure are calculated.

Both methods were run with the 10-fold cross-validation procedure, and average accuracy, recall and f-measure were obtained. The 10-fold cross-validation procedure was run 30 separate times with both methods in order to measure the statistical significance of the various results obtained.

One of the greatest disadvantages of LIKE (also present in KEA) is that, for each n-gram, two features are calculated whose result depends on the entire corpus (IDF and Z-score). Depending on a corpus for calculating features is not desirable, so two versions of the LIKE method were run – LIKE-8, which uses the eight features proposed in this article (see Section 2.2), and LIKE-6 which uses only the six features that do not depend on the corpus (i.e., all but IDF and Z-score).

Table 1 shows the average accuracy, recall and f-measure for the 30 separate runs with LIKE-8, LIKE-6 and KEA. With these results, a statistical test was carried out to determine the statistical significance for LIKE-8 vs. KEA, LIKE-6 vs. KEA and LIKE-8 vs. LIKE-6 (Table 2). As it can be seen in Table 2, both LIKE-8 and LIKE-6 achieved better results than KEA, while the version that used all eight attributes improved only accuracy and f-measure results when compared to the version that used only those six that are not corpus-dependent.

Table 1. Average precision, recall and f-measure for LIKE-8, LIKE-6 and KEA (standard deviation indicated between parentheses).

	LIKE-8	LIKE-6	KEA
Precision	0.76 (0.051)	0.65 (0.101)	0.52 (0.006)
Recall	0.75 (0.094)	0.72 (0.141)	0.37 (0.004)
f-measure	0.74 (0.053)	0.68 (0.116)	0.43 (0.005)

Table 2. Results of the statistical significance for precision, recall and f-measure for LIKE-8 vs. KEA, LIKE-6 vs. KEA and LIKE-8 vs. LIKE-6. For α=0.01 the "+" sign indicates that the result is statistically significant, while the "–" sign indicates that there is no statistical significance (p-value indicated between parenthesis).

	LIKE-8 vs. KEA	LIKE-6 vs. KEA	LIKE-8 vs. LIKE-6
Precision	+ (5.48×10^{-22})	+ (5.19×10^{-08})	+ (4.12×10^{-06})
Recall	+ (1.50×10^{-19})	+ (4.08×10^{-14})	- (0.4832)
f-measure	+ (2.18×10^{-24})	+ (2.51×10^{-12})	+ (0.0096)

4. Conclusions and Future Work

The novel automated method LIKE for extracting keywords from the text of a set of documents has been presented. This method extracts n-grams from

the documents and then calculates a series of features to convert them into numeric vectors. It then uses these vectors as data to train a back-propagation network and thus obtain a model that works as predictor and that can be used to extract keywords from new documents.
In this paper, the calculation of eight features is proposed for each n-gram, with only two of these being dependent on the entire corpus. LIKE was trained using the eight features, and then a second test was carried out using only the six features that do not depend on the corpus. Articles written in English submitted to the WICC between 1999 and 2012 were used for the experiments. The results obtained were compared with KEA, and it was shown that both the six-feature and the eight-feature LIKE models were better. When comparing the results obtained with both versions of LIKE, using all eight features turned out to be superior than using just six when calculating precision and f-measure, while for the recall parameter, neither of the versions appeared to be better than the other.
The main advantage of the method presented here is that it does not depend on the language of the texts analyzed, since it does not pre-process them because it does not use stemming or a stop-word list.
As future work, it would be interesting to study in detail the n-grams that are negatively classified so as to determine their nature and analyze the possibility of detecting grammar structures that help improve the performance of the method. Also, if less candidate n-grams are identified, the majority class of negative cases would be reduced and this could possibly lead to being able to omitting the clustering stage before training the network.
Another aspect to be studied in relation to the method proposed here is the possibility of assigning keywords from a list of controlled vocabulary. Different authors may choose different key words in articles dealing with the same topic, so it would be interesting if an automated assignment method were available to assign key words from a list of controlled vocabulary. This would ensure that documents on related topics would have the same key words, which would in turn improve the results obtained in future searches, classifications or statistical analyses.

References

1. Tonella, P., Ricca, F., Pianta, E., Girardi, C. (2003). Using keyword extraction for Web site clustering. In: Conference Using keyword extraction for Web site clustering, pp. 41 - 48.
2. D'Avanzo, E., Magnini, B., Vallin, A. (2004). Keyphrase Extraction for Summarization Purposes: The LAKE System at DUC-2004. Proceedings of the 2004 Document Understanding Conference.
3. Wan, X., Yang, J., Xiao, J. (2007). Towards an Iterative Reinforcement Approach for Simultaneous Document Summarization and Keyword Extraction. In: Conference Towards an Iterative Reinforcement Approach for Simultaneous Document Summarization and Keyword Extraction, pp. 552-559.

4. Zha, H. (2002). Generic summarization and keyphrase extraction using mutual reinforcement principle and sentence clustering. In: Conference Generic summarization and keyphrase extraction using mutual reinforcement principle and sentence clustering, pp. 113-120.
5. Islam, M.R., Islam, M.R. (2008). An improved keyword extraction method using graph based random walk model. 11th International Conference on Computer and Information Technology 225-229.
6. Kaur, J., Gupta, V. (2010). Effective Approaches For Extraction Of Keywords. International Journal of Computer Science Issues 7.
7. Liu, Y., Ciliax, B.J., Borges, K., Dasigi, V., Ram, A., Navathe, S., Dingledine, R. (2004). Comparison of two schemes for automatic keyword extraction from MEDLINE for functional gene clustering. Proc IEEE Comput Syst Bioinform Conf 394-404.
8. Witten, I.H., Paynter, G.W., Frank, E., Gutwin, C., Nevill-Manning, C.G. (1999). KEA: practical automatic keyphrase extraction. In: Conference KEA: practical automatic keyphrase extraction, pp. 254-255.
9. Matsuo, Y., Ishizuka, M. (2003). Keyword Extraction From A Single Document Using Word Co-Occurrence Statistical Information. In: Conference Keyword Extraction From A Single Document Using Word Co-Occurrence Statistical Information, pp. 392-396.
10. Frank, E., Paynter, G.W., Witten, I.H., Gutwin, C. (1999). Domain-specific keyphrase extraction. proc. Sixteenth International Joint Conference on Artificial Intelligence 668--673.
11. Wu, C., Marchese, M., Wang, Y., Krapivin, M., Wang, C., Li, X., Liang, Y. (2009). Data Preprocessing in SVM-Based Keywords Extraction from Scientific Documents. Fourth International Conference on Innovative Computing, Information and Control (ICICIC), pp. 810 - 813.
12. Csomai, A., Mihalcea, R. (2008). Linguistically Motivated Features for Enhanced Back-of-the-Book Indexing. Proceedings of the 9th SIGdial Workshop on Discourse and Dialogue.
13. Kireyev, K. (2009). Semantic-based estimation of term informativeness. Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics 530-538.
14. Hulth, A. (2003). Improved automatic keyword extraction given more linguistic knowledge. Proceedings of the 2003 conference on Empirical methods in natural language processing 216-223.
15. Wang, C., Zhang, M., Ru, L., Ma, S. (2008). An Automatic Online News Topic Keyphrase Extraction System. IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology 1, 214-219.
16. HaCohen-Kerner, Y., Gross, Z., Masa, A. (2005). Automatic extraction and learning of keyphrases from scientific articles. Proceedings of the 6th international conference on Computational Linguistics and Intelligent Text Processing 657-669.
17. Hu, X., Wu, B. (2006). Automatic Keyword Extraction Using Linguistic Features. Sixth IEEE International Conference on Data Mining Workshops 19 - 23.

18. Fürnkranz, J. (1998). A study using n-gram features for text categorization. Austrian Research Institute for Artifical Intelligence 3, 1-10.
19. Castro, C.L., Braga, A.P. (2013). Novel Cost-Sensitive Approach to Improve the Multilayer Perceptron Performance on Imbalanced Data. IEEE Transactions on Neural Networks and Learning Systems 24, 888-899.
20. Lin, M., Tang, K., Yao, X. (2013). Dynamic Sampling Approach to Training Neural Networks for Multiclass Imbalance Classification. IEEE Transactions on Neural Networks and Learning Systems 24, 647 - 660.
21. Zhang, Y.-p., Zhang, L.-N., Wang, Y.-C. (2010). Cluster-based majority under-sampling approaches for class imbalance learning. 2nd IEEE International Conference on Information and Financial Engineering (ICIFE) 400-404.

VIII

Architecture, Nets and Operating Systems Workshop

Preprocessing Database Fingerprints for Indoor Positinioning Systems

CARLOS KORNUTA[1], NELSON ACOSTA, JUAN TOLOZA[1]

CONICET, INCA/INTIA - School of Exact Sciences
UNICEN, TANDIL - Argentina
{ckornuta, nacosta, jmtoloza}@exa.unicen.edu.ar

Abstract. *Indoor positioning systems calculate the position of a mobile device (MD) in an enclosed environment with relative precision. There are various techniques of positioning, where the most widely used parameter is the RSSI (Received Signal Strength Indicator). In this paper, we analyze the fingerprinting technique to calculate the error window obtained with the Euclidian distance as main metric. Build variations are presented for the Fingerprint database analyzing various statistical values to compare the precision achieved with different indicators.*

Keywords: *Indoor positioning, Indoor localization, RSSI, Fingerprint.*

1. Introduction

At present, it is necessary to have mechanisms that allow determining the location of a MD within a building. Some examples include interactive maps of malls and museums, college campus guided maps, patient monitoring systems in hospitals and/or nursing homes for the elderly [1]. The use of GPS is not possible, since it does not work in enclosed spaces because it requires a direct and unobstructed line of sight between the receiver and a minimum of three satellites [2] [14].

Calculating the relative position of a MD is the process through which information is obtained about the position of such device in relation to landmarks on a predefined space [3]. The techniques used to calculate a position in an enclosed space are classified, based on the sensor technology used, in: Time of Arrival (ToA), Angle of Arrival (AoA), and Received Signal Strength Indicator (RSSI). Within the latter, the most commonly used algorithm for estimating the position is Fingerprint [4].

In 2000, the RADAR system [5] obtains a mean accuracy within 2-3 meters. In 2003, the LEASE system [6] offers a framework based on the Fingerprint technique that achieves an accuracy of 2.1 m. In 2007 the Fingerprint technique is used [7] to estimate the position of the DM in conjunction with a Bayesian network algorithm, achieving an accuracy of 1.5 m. In [8] a system

using the method Fingerprint and Euclidean distance with enhancement algorithm using fuzzy logic, in the first instance to obtain an accuracy of 4.47 m and then fuzzy logic is presented 3 m. Ekahau positioning system [9] based on the RSSI parameter attains a precision of 1-5 m depending upon the conditions of the environment. In [10] an algorithm based on using neural networks is presented system, achieving precision 1-3 m. In this article various variants of the construction of the database are analyzed Fingerprint.

This paper is organized as follows: Section 2 presents the technique used, Section 3 details the experiments carried out, Section 4 analyzes data, Section 5 analyzes the metrics used and the results obtained during positioning. Finally, Section 6 describes the conclusions and future work.

2. Location using FINGERPRINT

The method is based on Fingerprint each position within an enclosure has a unique signature, consisting of a tuple (P / L), where P contains information about the unique pattern and L information on the position within the building. The information on the position can be represented in a coordinate tuple format or a proxy. This technique requires training, where the sampling of each of the signatures is made [11].
First, a radio map [6] must be designed, which is a pattern map containing the specific positions within an enclosed space and an RSSI strengths vector with all the strengths or intensities of the APs reached at each position. Creating a radio map involves:

1. At each position, signal strength (RSSI) values are shown, putting together a strength vector whose dimension depends on the number of visible APs.
2. For each sector in the area that can receive the signal from N access points (AP), an RSSI vector is obtained from each AP.
3. To link signature and location information a deterministic method is used to find the position of the closest vector in many cases the Euclidean distance is used.

To calculate the position of the MD, the values from all visible APs are captured from that same position. The acquired values are then compared with the values stored in the database to obtain the coordinates that represent the location of the device [12].
Fingerprint data base is a summary of the data of the Radio Map, which facilitates location and minimize the calculation error reduces. Estimation algorithms correlate the values obtained from the location information and the Fingerprint database to determine the relative position of the DM. The best known deterministic method is the "nearest neighbor", where the mean vectors are used, which contain the average of the RSSI values of each AP at each point on the map.

3. Experiments and design of the database

The experiment is performed in the field of research institute fellows INTIA / INCA, Faculty of Exact Sciences, National University of Central Buenos Aires Province. The area has an approximate size of 36 m2. For measurement and data capture the corresponding area is divided into a coordinate axis (row, column) (Figure 1), each region of the map has a spacing of 90 cm from the previous point.

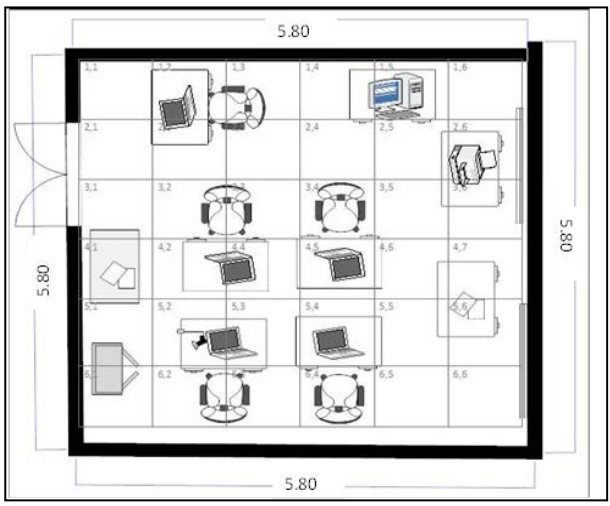


Fig. 1. *Location of map coordinates*

Data were captured with IWLIST on Ubuntu 8.04. The data capture process for building the radio map (Figure 1) is as follows:

1. Positioning of the MD at a coordinates point in the map.
2. Scanning and capturing RSSI values for 180 seconds to stabilize sampling signal.
3. Scanning, capturing and storing the RSSI and SSID values corresponding to the signal of the various APs that are within reach at that position for 90 seconds.
4. Moving the device to the next coordinates point in the map, and repeating step (1) if it is not the last location.

In Figure 2 the distribution of AP is displayed. The sector is INTIA Fellows AP 4 and the building is 58 m long. For each sampling point 100 RSSI parameter vectors are obtained, containing 11 values for available APs in range of the DM. By convention, when an AP is out of range, the value 0 is assigned.

With information on the Radio Map of Fingerprint data base (consisting of the average values) is constructed, and has also studied other values to represent each AP at each position:

- **Mean RSSI:** the arithmetic mean of all observations of AP.
- **Interquartile Dupla:** Considering the total values for each AP, the data is sorted in ascending order, then divided into 4 sets with equal number of elements. Quartiles ends are removed, and quartiles is calculated**:**
  - Average and
  - Mode arithmetic**.**
- **Mode:** with the numbers of samples taken is calculated.
- **Average Dupla Interquartile:** the average values and interquartile fashion Dupla are averaged.

Fig. 2. *Distribution of AP, called 1: farm, 2: default, 3: inca, 4: INCA2 5: intia, 6: ISISTAN-2, 7: PLADEMA-2, 8: PLADEMA-invited, 9: slab, 10: unicen2, 11: wlbiolab.*

4. Data analysis

Taking AP 3 as reference, which is approximately 15 m (4 brick walls and a wallboard) from the location where data capture is done along with the sampling of values corresponding to the signal of the APs found, signal strength variations, analyzed by row, are as follows:

- From the initial position (1,1), in a straight line, signal strength decreases by 2 dBm every 180 cm. At position (1,5), it returns to its normal value, and then it decreases again. Therefore, it fluctuates between -86 and -90.
- If we consider the second row of coordinates, signal strength decreases by 2 dBm after 270 cm.

- In the third row, signal strength decreases by 2 dBm in 360 cm, and then increases to -89 at the following point, to return to its starting value at the next pair of coordinates.
- On row 4, the value is constant, with no significant variations until the last point, where signal strength increases to -79.
- On row 5, after 90 cm from the initial position, signal strength decreases to -86 and then remains stable until reaching point (5,4), where it increases by 3 dBm and then decreases to -92, and remains stable at the initial values.
- On row 6, signal strength increases by 5 dBm after 180 cm and then decreases, reaching a value of -91, to then go back to the initial values.

By comparison with the values obtained for AP 4, which is within the same sampling sector, signal strength values are as follows:

- It starts at (1,1) with an initial value of -54, and signal strength fluctuates between +- 9 dBm, with the exception of point (1,6).
- The fluctuations and variations observed on row 2 are less significant than for the previous point – signal strength varies within a 3 dBm range, with the exception of point (2,5), where it decreases 10 dBm and ends with an approximate value of -55 dBm.
- On row 3, it varies between -41 and -17 dBm, and is becomes stable at values that are similar to the initial ones.
- On row 4, it varies between -43 and -8 dBm at (4,3), while at position (4,5) it stabilizes again at its initial values at 180 cm.
- On row 5, it varies between -47 and -55 dBm and between -47 and -55 dBm, oscillating at 8 dBm
- On row 6, it varies between -49 and -59 dBm, with a variation of +- 10 dBm.

After analyzing the data, it can be inferred that signal strength values fluctuate within a wider spectrum if the AP is at a smaller physical distance from the sampling point. If the AP is further away from the reference point, signal strength does not present significant changes, with variations of +- 3 approximately.
Table 1 shows Wi-Fi signal absorption values for different materials [13]; these values affect RSSI degradation.

Table 1. Wi-Fi signal strength attenuation caused by materials at 2.4 GHz:

Obstacle	Additional Loss (dB) (approx.)
Non-metal window (glass)	3
Metal window	5 to 8
Thin wall	5 to 8
Medium wall	10
Thick wall (15 cm)	15 to 20
Very thick wall (30 cm)	20 to 25
Floor or thick ceiling	15 to 20
Floor or very thick ceiling	15 to 25

Table 2 identifies 4 main coordinates within the map that correspond to certain points where there could be a discrepancy between the values and the set of detected APs; three APs are selected as way of example, identifying average, maximum and minimum RSSI.

Table 2. AP variation analysis

Coordinates	AP	Average	Maximum	Minimum
1.1	3	-89	-81	-97
	7	-75	-71	-79
	6	-64	-57	-69
1.6	3	-91	-77	-97
	7	-72	-71	-79
	6	-63	-57	-71
6.1	3	-86	-77	-95
	7	-73	-69	-77
	6	-63	-53	-71
6.6	3	-87	-79	-93
	7	-75	-71	-81
	6	-52	-45	-71

5. Result analysis

The device is positioned at a point on the map (Figure 1) and the strengths vector for the visible APs is obtained. Then, with this pattern vector and each of the stored strengths vectors (mean, interquartile pair, mode, pair average), the Euclidian distance is calculated, obtaining the distances to each point of coordinates. The approximate position is determined as the lowest value that meets the equation, that is, the shortest distance between the training set obtained (*Fingerprint* database) and the input data pattern.

5.1 Position (1,5):

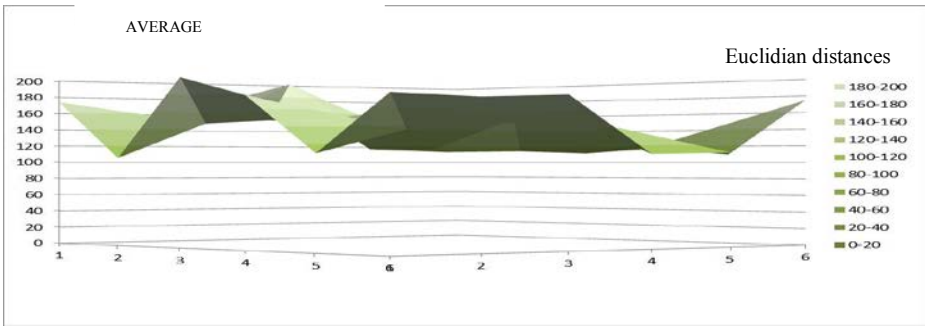


Fig. 3. *Surface chart, AVERAGES for position (1,5)*

Considering Figure 3 with the pattern [0, 0, 0, -89, -61, -59, -75, -75, -67, -89, 0], and using the average vectors for calculations, the section that minimizes distance is coordinate position (1,2). In this case, it can be seen that there is an error of 1.80 m which, considering Table 1, can be due to AP signal fluctuations, caused by strength weakening due to the presence of a wall next to the sampling point, which would cause the RSSI to decrease and prevent the visibility of an AP 3.

5.2 Position (5,3)

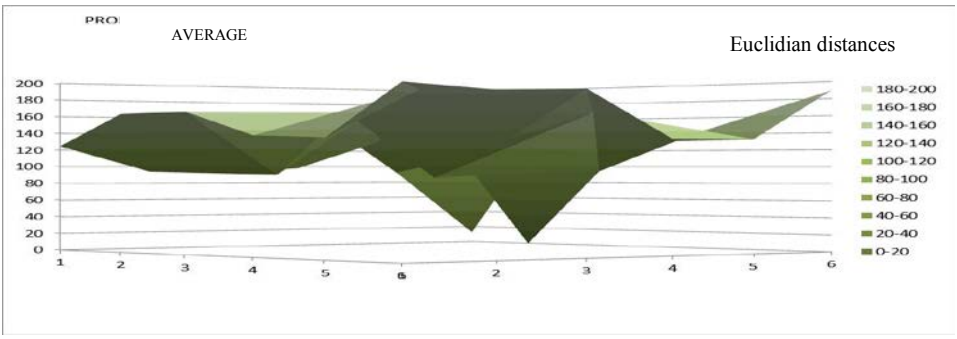


Fig. 4. *Surface chart, AVERAGES for position (5,3)*

Figure 4 shows the "*pattern vector*" for position (5,3): [0,0, -91, -45, 0, -63, -81, -83, -69, 0, -93]. The shortest distance obtained by the averages is position 5,3 exactly, which is represented as a minimum at that point. There are some vectors that show variations in the signal of the closest APs (3, 6 and 9) in a range between +- 2/4 dBm; accuracy is reduced, and the location of the MD at point (4,5) is determined with an error of 1.7 m.

If we look at the map presented in Figure 1, it can be seen that there are no walls adjacent to the capture point in this area, e.g., (1,5) in Figure 3.

5.3 Position (4.5)

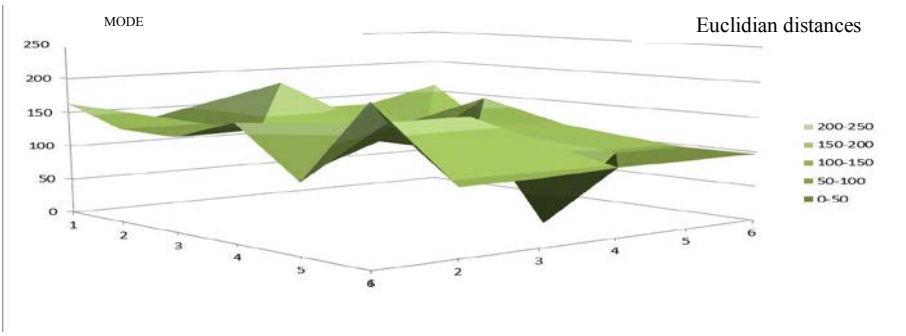


Fig. 5. *Surface chart, MODE for position (4.5)*

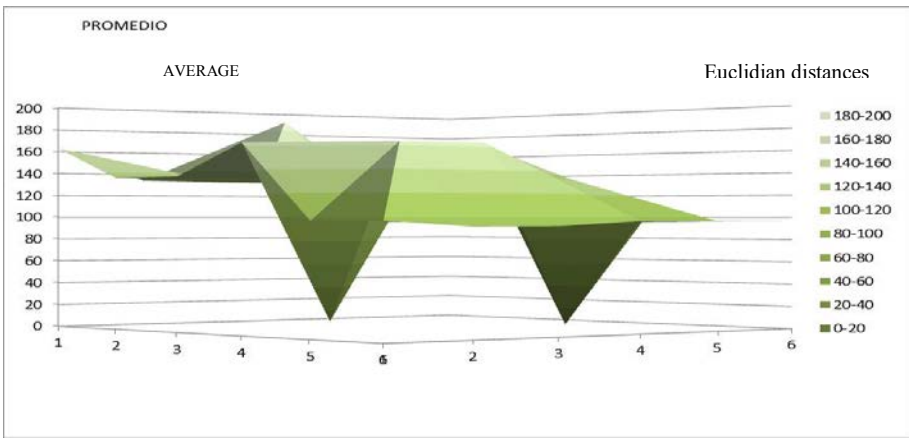


Fig. 6. *Surface chart, AVERAGES for position (4.5)*

Figures 5 and 6 present the pattern [-95, 0, -93, -43, 0, -57, -77, -75, -65, -89, 0], corresponding to position (4,5). Both charts show the same position, but the one in Figure 5, obtained by calculating the distance of the mode vectors stored in the database, offers a better representation, with a minimum absolute value at the positioning point. The result depicted in Figure 6, with calculations based on averages, is different, although quite similar. Neither of these examples have adjacent obstacles, such as walls or windows, that result in an increased signal absorption.

6. Conclusions and Future Work

To analyze the error rate, 60 tests were carried out at each point of the map (Figure 1), obtaining the pattern vector for estimating the position of the MD. From the total of tests carried out, in 70 % of them the distances presented in the chart shown in Figure 7 were obtained. As it can be observed, the error margin is smaller at the central coordinates in the map, obtaining a minimum value of 1.2 m and a maximum value of 2.4 m. When analyzing the results obtained for the lower and upper sections, it was corroborated that the error range varies between 2.4–3.6 m.

Thus, it can be inferred that the greater error margins are recorded in those sections where there are certain adjacent obstacles (walls, windows, and so forth) that result in an increased signal absorption and a greater multi-trajectory effect.

An experience that allowed locating a MD with a lower error by working with average and mode values has been documented, using a *Fingerprint* database with variations.

Considering the entire analysis area, an average maximum error of 3.6 meters is achieved for positioning the devices. In the central areas, approximately 0.90 meters from walls, the maximum error achieved is 1.7 meters.

There is promise in this technology, and we will continue our work to reduce error. Our next approach will be to include a voting method for automatically selecting the best technique, add a feature to take into account Wi-Fi signal travel time, carry out tests in various offices, and carry out tests in open spaces.

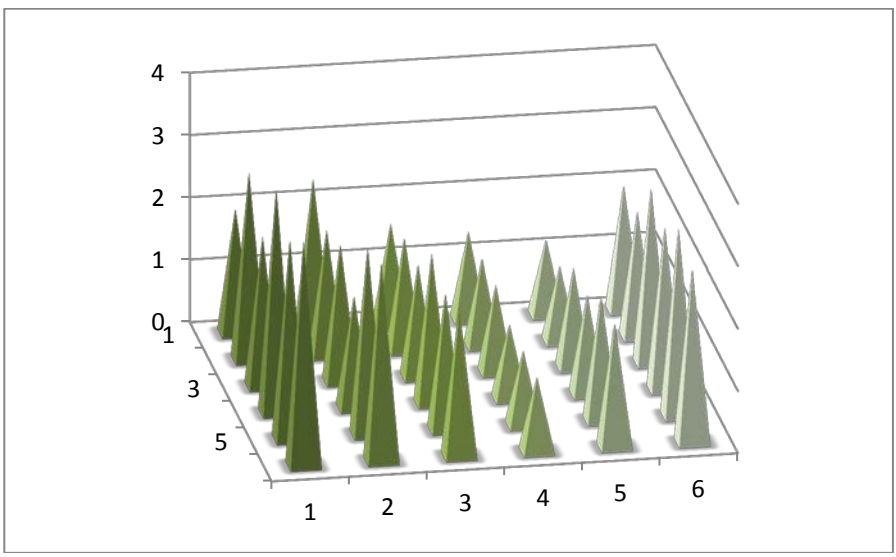


Fig. 7. *Positioning errors obtained, showing how error increases when there are nearby walls.*

References

1. Ladd, A. M., Bekris, K. E., Marceau, G., Rudys, A., Kavraki, L. E. and Wallach, D. S. (2002). Robotics-based location sensing using wireless ethernet. ACM International Conference on Mobile Computing and Networking (MOBICOM'02), New York.
2. Djuknic, G. M. and Richton, R. E. (2001). Geolocation and assisted GPS, *IEEE Computer*. Vol. 34, Nro. 2, pp:123-125.
3. Rappaport, T. S., Reed, J. H. and Woerner, B. D. (1996). Position location using wireless communications on highways of the future, IEEE Communic. Vol. 34, Nro.10, pp: 33-41.
4. Pahlavan, K., Li, X. and Makela, J. P. (2002). Indoor geolocation science and technology, *IEEE Communications Magazine.*, Vol. 40, Nro. 2, pp: 112-118.
5. Bahl, P. and Padmanabhan, V. N. (2000). RADAR: An in-building RF based user location and tracking system, *IEEE Infocom 2000*. Vol.2, Nro. 1, pp: 775-784.
6. Youssef, M. A., Agrawala, A. and Shankar, A. U. (2003). WLAN location determination via clustering and probability distributions, in *Proc. IEEE International Conference on Pervasive Computing and Communications, Fort Worth*.
7. Teuber, A., Eissfeller, B. (2006). WLAN indoor positioning based on Euclidean distances and fuzzy logic, *Proceedings of the 3rd Workshop on Positioning, Navigation and Communication (WPNC'06)*, Munich, Alemania.
8. Ekahau (2012). *Ekahau positioning engine 2.0; 802.11 based wireless LAN positioning system*, Ekahau Technology, Internal Report, www.eukahau.com.
9. Battiti, R., Lee Nhat, T., Villani, A. (2002). Location-aware computing: a neural network model for determining location in wireless LANs.
10. Pahlavan, K., & Krishnamurthy, P. (2002). Principles of Wireless Networks - A Unified Approach, Prentice Hall. ISBN-10: 0130930032
11. Brachmann, F. (2012). A comparative analysis of standardized technologies for providing indoor geolocation functionality, Symposium on Computational Intelligence and Informatics (13th CINTI), 2012 IEEE, Hungary, Budapest.
12. Enge, P. and Misra, P. (1999). Special issue on gps: The global positioning system, Proceedings of the IEEE.Pp: 3-172.
13. Najnudel, M. (2004). Estudo de propagação em ambientes fechados para o planejamento de wlans, Universidad Católica de Rio de Janeiro. Tesis.
14. Toloza, J., Acosta, N., de Giusti, A. (2012). An approach to determine magnitude and direction error in gps system. Asian Journal of computer science And Information Technology. Vol. 2, Nro. 9, pp: 1-5.

Radio Communication Solutions in Small and Isolated Communities: the IEEE 802.22 Standard

A. ARROYO ARZUBI[1], A. CASTRO LECHTALER[1, 3 Y 5], A. FOTI[4], R. FUSARIO[1, Y 4], J. GARCÍA GUIBOUT[2] AND L. SENS[4]

[1] Escuela Superior Técnica - Facultad de Ingeniería del Ejército - IESE, C1426, Buenos Aires; [2] Instituto Tecnológico Universitario - Universidad Nacional de Cuyo, M5500, Mendoza; [3] Universidad Nacional de Chilecito, F5360, Chilecito, La Rioja; [4] Universidad Tecnológica Nacional, C1042, Universidad de Buenos Aires, C1120, Buenos Aires; República Argentina.

{A. Arroyo Arzubi, arroyoarzubi@iese.edu.ar, A. Castro Lechtaler, acastro@iese.edu.ar, A. Foti, foti.antonio@gmail.com, R. Fusario, rfusario@speedy.com.ar, J. García Guibout, jgarcia@itu.uncu.edu.ar, L. Sens, lsens@frba.utn.edu.ar}

Abstract. *In recent years the use of wireless communications has increased significantly. Rural communities without cable network communication in Argentine have found a solution in wireless technologies. Based on previous fieldwork, this paper analyzes software development of integration based technologies for communication equipment. It focuses on the feasibility of the IEEE 802.22 standard as a solution to the wireless problem in our country.*

Keywords*: IEEE 802.22, White Spaces, Cognitive Radio, Rural Communications, Digital TV Broadcast.*

1. Introduction

In the framework of the Project ***Communitarian Private Networks*** [1], different technologies providing links to small and isolated communities in Argentine have been analyzed and compared. These communities, with low population densities, hold no commercial interest to service providers [2], [3], [4]. Notwithstanding, several rural facilities maintain operations in these isolated areas, providing significant quantities of food products at different stages of manufacturing. They supply not only nearby cities, but also constitute an important source of export commodities and revenue for many countries.

The geographic dispersion of these facilities interfere with cable communications –either with copper pairs, coaxial or optic fiber cables – due to high costs and maintenance problems. Consequently, the solution consists of establishing full duplex links via radio waves at a 30 to 70 km distance between antennas and at frequencies not restricted by government regulations [5], [6].

Towards the end of the 90s and beginning of this century, technical problems evolved side by side with their solutions. The process lead to the approval, on July 1st, 2011, of the standard IEEE ***802.22 - Cognitive Wireless RAN Medi-***

um Access Control (MAC) and Physical Layer (PHY) Specifications Policies and Procedures for Operation in the TV Bands [7].
The present work analyzes the use of this type of links in the same area where our group is conducting fieldwork.

2. Previous Fieldwork and Testing of New Technologies

The Project *Communitarian Private Networks* [5] explores different technologies providing communications to small and isolated rural communities in Mendoza state in Argentine, with low population densities and without telephone services, whether these may be landlines or cellular. Hence, these communities do not have access to voice, data or internet networks.
A small community which met the requirements of the Project was searched: an isolated and distant town where experiences can be appropriately implemented. The community Corral de Lorca, in the department of General Alvear, province of Mendoza was finally selected. Its location is shown in Figure 1.

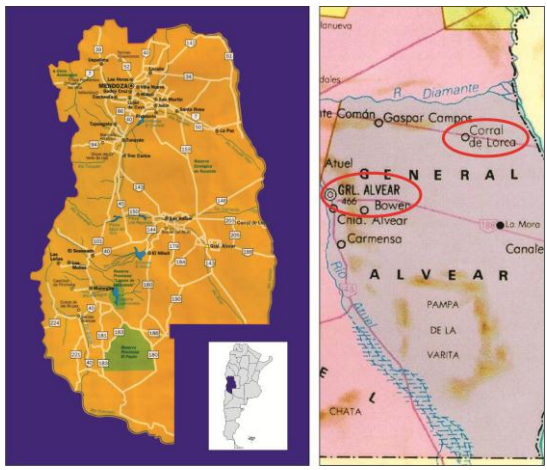


Fig. 1*. Geographic Location of Corral de Lorca*

The technologies under study were: Power Line Communications (PLC) [2, 3] and the standard 802.11 [4]. Both experiences show that PLC technology is not recommended for outdoor links under the required conditions.
The considered solution involves establishing a point to point link where one endpoint is located in the department´s main city, General Alvear, with access to the PSTN network, and the other in the Corral de Lorca community, located in the southwest of the province of Mendoza, Argentina.
Two Motorola Canopy platforms are used at bandwidths of 2.4 GHz and 5.7 GHz (similar to WiMax). At the Corral de Lorca node [6], phone services can

be implemented through VoIP as well as 802.3 to the Local Area Network and 802.11 for Wireless Internet Services.
The link is placed at a critical distance. Corral de Lorca is 70 kilometers from General Alvear in a straight line over a desert but with dense vegetation close to the base station area. To analyze the propagation issues the freeware ***Radio Mobile*** developed by Roger Coude [7] is used.
The outcome is not entirely satisfactory. Significant attenuation is registered due to the distance of the trans-horizon link and to the strong attenuation resulting from the dense vegetation at the outskirts of General Alvear. These factors contribute to a low value in the signal/noise ratio at the reception end in Corral de Lorca.
The conclusions are:

- The link is studied as a typical case to solve the general problem of rural populations. Hence, it could be generalized later for the application of analogous situations. In these cases, the distance to cover should range between 60 to 70 kilometers.
- Coordinates and terrain conditions are detailed between the two endpoints. Estimates for different bandwidths are established, i.e. 2.4 GHz and 5.7 GHz.
- The distance between the endpoints (60 km) is greater than regular distances contemplated in the theory for the application of standard 802.11.
- In addition, the analysis focuses on the fact that, in practice, transmitted signals in rural areas behave differently from those in urban settings because the former suffer less from noise spectrums. Radio Mobile software is considered to be a valuable tool in the design of radio electric links.

As a result of these experiences, continuing work is focusing on the new 802.22 standard.

3. Technical Problems and New Solutions

3.1 Introduction

The boundaries between the fields of communications and computer science have merged over time. Several concepts used in the field of telecommunications are now encountered in computer science and vice versa. Also, methodological practices of computer science are an integral part of telecommunications nowadays.
Consequently, today we refer to both disciplines as Information and Communications Technology - ICT[1] or as Teleinformatics, as referred to by European and American scholars.

[1] Also known as TICs in Spanish

Moreover, by the end of the 90s and beginning of this century, wireless communications increased exponentially. For instance, currently the total number of mobile phone users exceeds the number of existing landlines.
The pervasive use of mobile communications presents several technical difficulties, which in turn lead to the development of their consequent technical solutions. In the following section, the main changes of the advance in wireless technology are outlined.

3.2 White Space and Congestion Bands

Modern societies are increasingly relying on radio spectrum use. The pervasiveness of wireless services and communication devices (mobile phones, police communications, Wi-Fi and digital TV broadcasting) are examples of this dependency. It has become one of the most necessary resources of modern times [9].
Global demand growth for mobile data traffic has increased between 2011 and 2012 at a rate over 100%. The expected growth rate of this demand for the period 2008 and 2013 is estimated to average at 131% per year [10], exceeding 2.000.000 Terabytes per month by the end of the current year. The intense spectrum use, up to 5 GHz, and more specifically at the coverage below 1 GHz, has lead to a thorough review of regulatory policies, along with a renewed interest in ***White Space*** research [2] [11].
Possible solutions to the increasing traffic, especially below 1 GHz, are: review and redesign of the regulatory framework, reduction of wireless services broadcasting, improved compression standards, replacement of various services by satellite or cable, dynamic spectrum access, and development of cognitive radio technologies. The latter is oriented to take advantage of under-utilized frequencies, temporary voids of primary signals, and different types of white space.
The CEPT, ***European Conference of Postal and Telecommunications Administrations***, has defined ***White Space*** as "a label indicating a part of the spectrum, which is available for radio communication applications (service or system) at a given time in a given geographical area on a non-interfering or non-protected basis with regard to other services with a higher priority on a national basis" [12]. Currently, several research efforts from different organizations, national and international, are working on white space.
Cognitive Radio Technology (CRT) is considered another possibility to address the rising spectrum shortage. When fully operational, CRT could provide technologies for a variety of applications (rural broadband, public safety and emergency response, and urban frequency use). This technology will also have significant consequences for dynamic detection and spectrum management.

[2] Or white holes.

3.3 Software Defined Radio

With the exponential growth of the ways and means by which people need to communicate through wireless communications, modifying radio devices easily and cost-effectively has become critical.

The technology ***Software Defined Radio (SDR[3])*** provides flexibility and profitability, as well as grants end users with comprehensive benefits from service providers and product developers *[13]. The* ***Wireless Innovation Forum*** defines ***Software Defined Radio*** as "radio in which some or all of the physical layer functions are software defined."

The radio is a device which transmits or receives wireless signals using a portion of the radio spectrum. Traditional radio devices exclusively based on hardware (e. g.: mixers, filters, amplifiers, modulators/demodulators, and detectors) are limited because their features can be modified only by physical intervention.

On the other hand, a ***Software Defined Radio (SDR)*** is implemented by means of software on a computer or embedded system. The concept is not new, but the rapidly evolving capabilities of digital electronics render practical many processes which used to be only theoretically feasible before [13].

Under this technology, the software proves to be efficient at a relatively inexpensive cost, with multimode and multiband wireless devices which can be continuously improved with software updates. In some cases, the software manages some or all of the functions to operate the radio equipment (including those of the physical layer processing).

3.4 Cognitive Radios [4]

At the end of the decade of the 90s, Joseph Mitola[5] and Gerald Maguire, researchers from the Royal Institute of Technology[6] developed what they called ***Cognitive Radio,*** an improvement of their previous work on ***Software Defined Radio*** technology [14] [15],

While Software Defined Radio offers great potential, it also requires arduous processing, limiting its flexibility and adequacy of network response.

Cognitive Radio embedded in communications software, such as ***Radio Knowledge Representation Language - RKRL***, can be considered an intelligent and efficient system for radio communications and protocols. Basically, it provides mechanisms based on the use of smart technology to optimize the spectrum.

---

[3] Also known as **S*oftware Radio***.

[4] Mitola defines ***cognitive*** as ***the mix of declarative and procedural knowledge in a self-aware learning system***.

[5] Joseph Mitola III received his doctorate in the Royal Institute with his thesis ***Cognitive Radio: An Integrated Agent Architecture for Software Defined Radio***.

[6] Located in Stockholm, Sweden.

As mentioned in 3.2, the allocation of frequencies in a saturated spectrum is not optimal, originating ***White Space***. A special range is assigned to the operators for the use of ***Digital TV Broadcasting***.

Those were the reasons which led to develop Cognitive Radio for wireless communications: ***to detect the parts of the radio frequency spectrum used inefficiently and to allow reuse without causing interference with the services assigned to them.*** The solution of these problems by variable frequency allocation, allows others to take advantage of unused parts of the spectrum.

Using intelligent software, Cognitive Radio periodically scans the spectrum in search of white holes, detects the use given to each of them, and then determines whether it is reusable.

The system operates by changing the transmitter parameters based on interaction with the environment. It has the ability and the technology to capture or sense the information from other radio equipment, providing spectrum awareness whereas reconfigurability enables the radio to be dynamically programmed.

It can be programmed to transmit and receive on a variety of frequencies and to use different transmission access technologies supported by its hardware design.

These operating procedures show the interaction between hardware design and application software development. They also represent a typical teleinformatics application, as characterized by Minola in his thesis.

3.5 Digital TV Broadcasting

Frequency spectrum use for TV broadcasting has varied since the first black and white broadcasts to the current digital high definition systems. Two bands are used: VHF (54 to 88 and 174 to 216 MHz) and UHF (512 to 806 MHz).

Fig. 2a. *Province of Buenos Aires*

Fig. 2b. *Rest of the country*

In several South American countries, the Japanese standard **Integrated Services Digital Broadcasting (ISDB)**[7] has been adopted with a few variants, such as the replacement of the compressing system MPEG-2 for MPEG-4. It was developed by the ***Association of Radio Industries and Businesses,*** known as ***ARIB,*** *w*hich promotes the efficient use of the spectrum.
ISDB include four standards depending on the used medium: ISDB-S (satellite), ISDB-T (terrestrial), ISDB-C (cable) and 2.6 GHz mobile broadcasting. All of them are based on multiplexing with a transport stream structure and are capable of High-Definition Television (HDTV) and standard definition television. The name of the standard was chosen for its similarity to ISDN (Integrated Services Digital Network). Both allow the simultaneous transmission of multiple channels of data through the multiplexing method.
In most cases, broadcasting stations have antennas reaching about 150 meters high, with significant coverage areas.
In the case of Argentina, more that 50 broadcasting stations have been set up as of July of 2013, covering a significant area of the country. The plan is to cover practically all of the populated areas, giving service to 90% of the population. Figures 2a and 2b illustrate the cities where these stations have been installed.

3.6 IEEE 802.22 as a Solution for Rural Areas

In Argentina, as in many countries with large rural areas, most cities are located within a range of 40 to 80 km apart in average.
The Project *Communitarian Private Networks* focuses on evaluating solutions to the communication problems of rural areas, in particular, isolated communities with low population density.
In our countries, the intensive use of spectrum and saturation in many of its frequency bands is due to wireless communications which has been the only feasible solution.
The 802.22 standard aims at using the vacancies in the TV spectrum. These frequencies are particularly suitable for remote areas where cables signal transmission are expensive or difficult to implement. Cables could only be replaced by costly satellite services. Thus, to implement a link using spare frequencies in these bands may be a practical and inexpensive solution.
In our country, the TV on the VHF band will be eliminated in 2016 (analogic blackout), liberating most of the UHF band, considering that the ***Argentine National Authority for Broadcasting Services - AFSCA***[8] has licensed only a few channels in the main cities (22 to 36).
As the new digital TV technology allows several standard definition programs in the same bandwidth of one high definition channel, there is a significant spectrum saving, and we still can get lots of free frequencies (channels 38 to 69), mainly in small cities.

[7] ***International Services Digital Broadcast***, Terrestrial Brazilian version ISDB-T_B.
[8] ***Autoridad Federal de Servicios de Comunicación Audiovisual***.

It is an opportunity for this IEEE 802.22 standard to be considered in the spectrum reallocation under study by the ***National Argentine Spectrum Authority - CNC***[9].

4. IEEE 802.22. Cognitive Wireless RAN Medium Access Control (MAC) and Physical Layer (PHY)

4.1. Introduction

On July 1st 2011, the standard ***IEEE 802.22: Cognitive Wireless RAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Policies and Procedures for Operation in the TV Band*** was approved under the sponsorship of the LAN / MAN Standards Committee.
The standard aims to set up criteria for the deployment of multiple interoperable products of the 802.xx series[10], offering fixed broadband access in various geographical areas, including especially those of low population density in rural areas, and avoiding interference to services working in the television broadcasting bands.
The standard, commonly known as ***Wireless Regional Area Networks – WRANs***, has been developed to operate primarily as broadband access to data networks in rural areas.

4.2. General features

The standard includes cognitive radio techniques to moderate interference to other existing operators, to grant geolocation capability, to provide access to a database of incumbent services, and to detect the presence of other services through spectrum-sensing technology, such as different WRAN systems or IEEE 802.22.1[11] wireless beacons.
The WRAN systems involve the use of channels ranging from 54 to 862 MHz in the VHF and UHF bands. The use of cognitive radio technologies scans for spare frequencies while avoiding interference with TV stations operating in the same bands.

[9] ***Comisión Nacional de Comunicaciones***.
[10] Wireless.
[11] ***IEEE 802.22.1: Standard to Enhance Harmful Interference Protection for Low-Power Licensed Devices Operating in the TV Broadcast Bands.*** 2010.

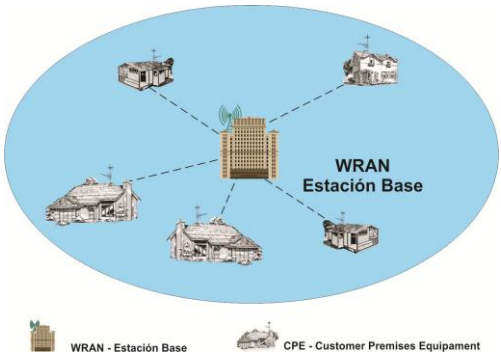


***Fig. 3**. 802.22: Working scheme*

Figure 3 illustrates a typical design. Assuming different quality of service (QoS), a Base Station - BS complying with the standard provides high-speed Internet services of up to 512 ***Customer Premise Equipments - CPEs***, fixed or portable, or groups of devices.

4.3. Cognitive Radio Capability

The cognitive radio capabilities supported by the standard are required to meet regulatory requirements for protection of frequency of incumbent's operators as well as to provide for efficient operation. They include: spectrum sensing, geolocation services, database access, registration and tracking of channel set management [8].
In areas where a computer with the IEEE 802.22 standard is intended to operate, the detection of operational channels which could be subject to interference includes the following:

- Television broadcasts.
- Wireless microphone transmissions.
- Transmissions from protecting devices, such as a Wireless Beacon[12].
- Other transmissions such as medical telemetry that may need to be protected in the local regulatory domain.

4.4. Topology

The standard topology is point-to-multipoint. The protocol works in a master/slave procedure, so that each CPE requires approval from the BS to transmit.
The system functions with a ***Base Station - BS*** and multiple ***Customer Premise Equipment - CPEs***. The base station controls the whole link, as well as its own performance and the CPE stations. It executes media access control,

[12] IEEE 802.22.1.

modulation of the RF transmission, coding, and selection of operating frequencies.
The CPE uses an antenna system as shown in figure 4. It has a directional antenna similar to those used for transmitting/receiving TV signals, one sensing antenna that surveys the spectrum to determine which frequencies are available and a GPS antenna to determine the exact location of the transmitting station [8].

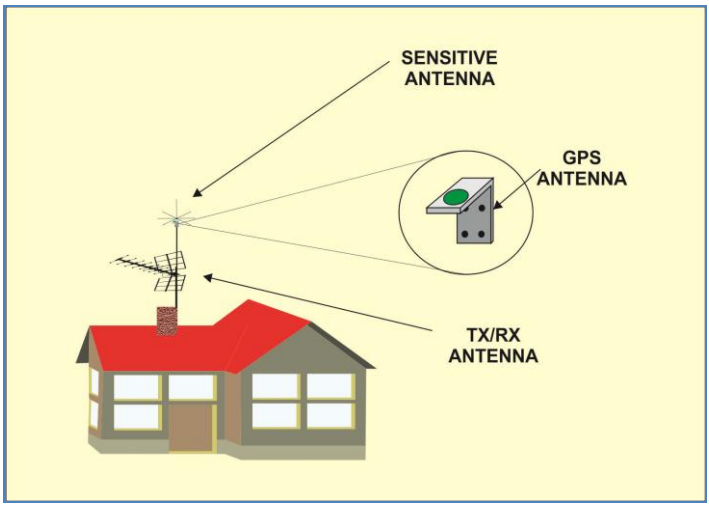


Fig. 4. *Customer Premise Equipment Antennas*

When the sensing antenna detects a band of the spectrum in use, the cognitive radio system changes the transmission features to avoid interference while granting priority to TV operators.
The GPS determines the exact location of the detected signal, so that the system searches the database service of the regulatory agency and find free frequencies for frequency hopping. According to the received information, the base station changes or not the parameters of transmission/reception.

4.5. The IEEE 802 LAN/MAN Committee: Family of Wireless Standards

The ***IEEE 802 LAN/MAN Standard Committee*** has developed a large and diverse family of wireless data communication standards. Since the first 802.3 version to the present, they have dealt with different requirements in wireless communications.
Figure 5 illustrates the most significant wireless standards and the relative position of the 802.22.

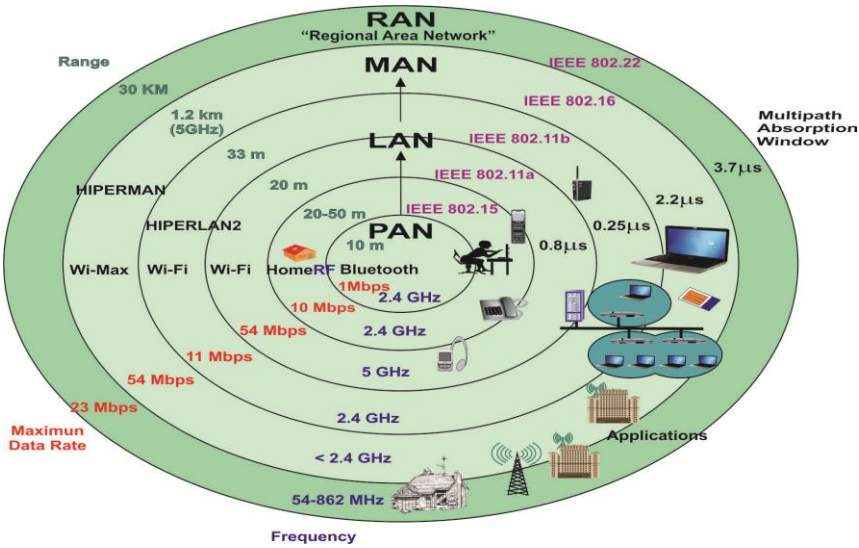


Fig. 5. *Different Wireless Standards Developed by the IEEE 802 Committee*

The standard provides wireless broadband access in rural areas within a range of 30 up to a maximum of 100 km from a base station.

4.6. Physical Layer - PHY

Similarly to the ***Asymmetric Digital Subscriber Line – ADSL***, the IEEE 802.22 standard provides broadband access at a data transfer rate of 1.5 Mbps for downlink and 384 kbps for uplink (Figure 6).

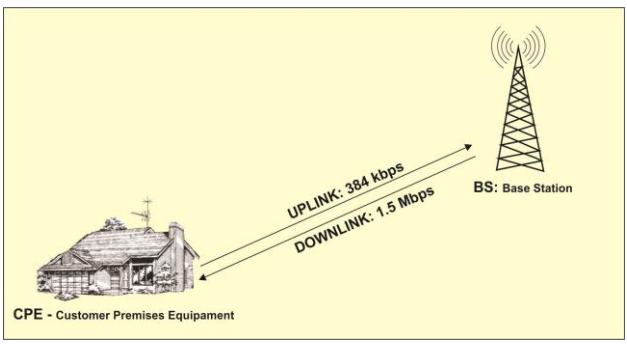


Fig. 6. *Different Wireless Standards Developed for the IEEE 802 Committee*

It works with multiplexing ***Orthogonal Frequency Division Multiple Access - OFDMA*** and defines twelve combinations of three modulations: QPSK -

Quaternary Phase Shift Keying, 16-QAM, and 64-QAM Quadrature Amplitude Modulation; and convolutional coding for error handling with the procedure ***Forward Error Control - FEC***.

Parameters	Specification
Frecuency range	54-862 MHz
Bandwidth	6 MHz, 7 Mhz, 8 Mhz
Payload modulation	QPSK, 16-QAM, 64-QAM
Transmit effective isotropic radiated power	Default 4 W for CPEs
Multiple access	OFDMA
Cyclic prefix modes	1/4, 1/8, 1/16, 1/32
Duplexing	TDD

Fig. 7. *Details the Different Features of the Standard*

4.7. Medium Access Control Layer - MAC

The MAC layer supports cognitive capabilities. Thus, it must have mechanisms for flexible and efficient data transmission. It must guarantee the reliable protection of services in the TV band and should be allowed to coexist with other 802.22 systems.
This layer is applicable to any region in the world and does not require country-specific parameter sets.
It is ***connection-oriented*** and provides flexibility in terms of QoS support. It also regulates downstream medium access by TDM, while the upstream is managed by an OFDMA system. The BS manages all the activities within its cell and the associated CPEs are under the control of the BS.

5. Conclusions

Societies today have become highly dependent on the radio spectrum with the intensive use of wireless devices and communication services. Cognitive Radio, using intelligent software and taking advantage of white holes, may be a solution to spectrum saturation.
The Project Communitarian Private Networks has focused on evaluating solutions to the communication problems of rural areas. It has concluded that wireless communications may be among the feasible solutions.

Taking advantage that Argentina has a plan to cover a significant area of its territory with a TV broadcasting system, the conditions may be the ideal to introduce simultaneously the 802.22 standard to the problem of rural communications.
The Project ***Communitarian Private Networks*** continues its work on this line of research.

6. Acknowledgements

The financial support provided by Agencia Nacional para la Promoción Científica y Tecnológica (Project PICTO 11- PICTO 11-18621 is gratefully acknowledged.

References

1. Castro Lechtaler, A. (Director). PICTO 11-18621. Redes Privadas Comunitarias. Proyecto FONCyT, ANPCyT. Working Paper.
2. Garcia Guibout, J., García Garino, C., Castro Lechtaler, A., Fusario, R. and Sevilla, G. (2007). Physical and Link Layer in Power Line Communications Technologies. Proceedings of 13th of Argentine Congress on Computer Science. ISBN 978 - 950 – 656 – 109 – 3. pp. 56 a 67. Corrientes.
3. García Guibout, J., García Garino, C., Castro Lechtaler, A., Fusario, R. and Sevilla, G. (2007). Power Line Communications in the Electric Network. Proceedings of 13th of Argentine Congress on Computer Science ISBN 978 - 950 – 656 – 109 – 3. pp. 68 a 79. Corrientes.
4. García Guibout, J., García Garino, C., Castro Lechtaler, A. and Fusario, R. (2008). Transmission voice over 802.11. Proceedings of 14th of Argentine Congress on Computer Science. ISBN 978 - 987 - 24611 - 0 - 2. pp. 307 a 318. Chilecito.
5. Castro Lechtaler, A., Foti, A., Fusario, R., García Garino, C. and García Guibout, J. (2009). Communication Access to Small and Remote Communities: The Corral de Lorca Project. Proceedings of 15th of Argentine Congress on Computer Science. ISBN 978 - 897 - 24068 - 4 - 1. pp. 1.117 a 1.126. Jujuy.
6. Castro Lechtaler, A., Foti, A., García Garino, C., García Guibout, J., Fusario, R. and Arroyo Arzubi, A. (2010). Proyecto Corral de Lorca: Una solución de conectividad a grupos poblacionales pequeños, aislados y distantes de centros urbanos. Proceedings de la Novena Conferencia Iberoamericana en Sistemas, Cibernética e Informática: CISCI 2010. - Volume III - ISBN - 13: 978 – 1 – 934272 – 96 - 1. PP. 121 a 127. Orlando, USA.
7. http://www.cplus.org/rmw/index.html (Radio mobile software).
8. IEEE 802.22 - Cognitive Wireless RAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Policies and Procedures for Operation in the TV Bands.

9. Cordeiro, C., Challapali, K. and Sai Shankar, D. B. (2006). N. IEEE 802.22: An Introduction to the First Wireless Standard based on Cognitive Radios Journal of Communications, Vol. 1, N° 1.
10. Gómez, C. (2013). Spectrum Regulation and Policy Officer Radiocommunication Bureau. www.itu.int/ITU-D/asp/CMS/Events/.../ITU-APT-S3_Cristian_Gomez.pdf. ITU. Apia, Samoa.
11. CEPT Report 24. (2008). A preliminary assessment of the feasibility of fitting new/future applications/services into non-harmonized spectrum of the digital dividend (namely the so-called "***white spaces***" between allotments. Report C from CEPT to the European Commission in response to the Mandate on: Technical considerations regarding harmonization options for the Digital Dividend.
12. http://www.wirelessinnovation.org/introduction_to_sdr
13. Dillinger, M., Madani, K., Alonistioti, N. (2003). Software Defined Radio: Architectures, Systems and Functions. Ed. Wiley & Sons.
14. Mitola, J., Maguire, G. (1999). Cognitive radio: making software radios more personal. IEEE Personal Communications Magazine, vol. 6, Nr. 4, pp. 13–18.
15. Mitola, J. (2000). Cognitive Radio: An Integrated Agent Architecture for Software Defined Radio. Dissertation submitted in partial fulfillment of the degree of Doctor of Technology. Royal Institute of Technology (KTH) - Teleinformatics. ISSN 1403 - 5286. Sweden.

V

Innovation in Software Systems Workshop

Knowledge Management: An Approach Applied to Public Administration

SEBASTIÁN PARDO[1], JUAN ENRIQUE CORONEL[1], RODOLFO BERTONE[2], PABLO THOMAS[2]

[1] Honorable Accounts Court of the Province of Buenos Aires – Argentina
[2] Institute of Research in Computer Science III-LIDI,
School of Computer Science.
National University of La Plata - Argentina

{spardo, jcoronel} @htc.gba.gov.ar
{pbertone, pthomas}@lidi.info.unlp.edu.ar

Abstract. *Knowledge management encompasses the set of activities carried out to use, share and develop knowledge in an organization and the individuals that work in it, helping them to better achieve their goals. In this paper, we present an approach to knowledge management and then analyze knowledge management applied to various environments within the public administration. From a practical standpoint, a software tool that promotes knowledge management has been developed; this tool was implemented in a specific public entity, the Honorable Accounts Court of the Province of Buenos Aires (HTC). By applying the theories that support knowledge management, existing non-tangible resources can be viewed, shared and used in various ways, which can help public organizations progress and modernize.*

Keywords: *Knowledge Management - Information Unit - Honorable Accounts Court Province Buenos Aires - Electronic Governance.*

1. Introduction

Knowledge allows making decisions and taking action [1]. Knowledge management can be defined as the set of activities carried out to use, share and develop knowledge in an organization and the individuals that work in it, helping them to better achieve their goals [2].
Knowledge management generates resources for organizations, the so-called intellectual capital, as non-tangible and lasting element for an efficient and sustainable management in time. Through knowledge management, organizations encourage individuals to go further in their jobs by contributing ideas while avoiding knowledge drain through people leaving the organization. This implicit or subconscious concept of knowledge retention is an innovation in the field of administration, and it is essential nowadays.

Focus has to be redirected to the need of intelligent and efficient States: relating and unifying criteria, discovering that the public administration has service consumers (citizens), competencies and providers, as well as regions and the need for integration among all these components.
The objective proposed in this paper is the development of a collaborative computer platform that will allow the systematic management of the Information Units of the Honorable Accounts Court of the Province of Buenos Aires.
This platform will be innovative in that it will incorporate to its design the concepts of participation and knowledge management through the interrelation of the Information Units, and it will allow users to weigh each Information Unit and add comments related to the unit in question, links to websites, and bibliographic references.
In the following section, a conceptual approach to knowledge management is presented; then, knowledge management in the public administration is described, and after that, the software system developed for the HTC is discussed. Finally, the results obtained are detailed, conclusions are drawn, and future work is presented.

2. Knowledge Management: A Conceptual Approach

2.1 Knowledge and Organizations

Davenport and Prusak differentiated between three concepts: data, information and knowledge [6]. Data are the minimum semantic unit, and they correspond to the primary information elements that are by themselves irrelevant as support for decision making. Information can be defined as a set of processed data that have a meaning (relevance, purpose and context) and, therefore, are useful for decision makers, since they help reduce uncertainty. Knowledge is a mixture of experience, values, information and know-how that acts as a framework for the incorporation of new experiences and information, and is useful for taking action. It originates in and applies to the mind of those having the knowledge [3].
From the perspective of knowledge management, one of the most relevant epistemological aspects is that of the process to generate and acquire knowledge.
Davenport and Prusak define knowledge as a fluid mixture of accumulated experience, values, contextualized information and the intuition of the expert, which combined create a reference framework for assessing, and incorporating, new learning and information. Within organizations, knowledge is found in repositories, but also in routine organizational processes, practices and guidelines [4].
Nonaka and Takeuchi generated a conceptual development where knowledge is created in reality when the different knowledge types are converted from one to another, through organizational levels, starting with the individual and

then moving up to the group, organizational and finally interorganizational levels, thus creating a spiral that results in innovation not only for products and technologies, but also for organizational processes and strategies [5]. This approach shapes the dominant conception on this issue nowadays.

2.3 Goals, Objectives and Cornerstones for Knowledge Management

The primary goal of knowledge management is defined as the improvement of the organizational services through the incorporation of individuals to obtain, share and apply collective knowledge to make optimal decisions in real time, the latter being understood as the time available for making the decision and carrying out the action that will materially affect the result.
As regards objectives, the following can be mentioned:
1. Getting institutions in general and companies in particular to act as intelligently as possible to ensure their viability and global success.
2. In other cases, being aware of the best value of their knowledge assets.
To achieve these goals, organizations build, transform, organize, deploy and effectively incorporate their knowledge assets [6].
A knowledge management system must combine three essential cornerstones for management:

- Staff and culture.
- Institutional management.
- Technology (portals, groupware, electronic communication tools, data warehouses and data mining, and support infrastructure).

3. Knowledge Management in Public Administration

It would be useful to make a distinction between the concepts of information society and knowledge society. Information society refers to the growing technological ability to store more information and transmit it faster and with greater broadcasting capacity. Knowledge society refers to the critical and selective incorporation of information by citizens who know how to take advantage of information [7]. Public institutions are founded upon these societies.
It can be said that public institutions are large producers and consumers of knowledge. As opposed to the case of private companies, public administration does not have to worry about profitability, but it must rather focus on two essential aspects [1]:

- Being highly efficient in collecting and appropriately spending its resources.
- Improving the quality of life of its citizens through the specialized services they provide.

However, to achieve these objectives, public institutions also have serious difficulty in tackling two other aspects:

1. Describing in detail the results they promise and, more specifically, management indicators that reflect their efficiency in keeping these promises.
2. Identifying the critical knowledge that affects the most the achievement of such results.

There are also political factors, such as:

- Internal problems: authoritarianism, corruption, social imbalance, inefficiency, insufficient organization of public institutions, and so forth.
- External problems: institutional centralization of Governments, laws and regulations that limit counties and states, political and party-political issues related to budget aspects, etc.

Public organizations are basically knowledge organizations and, to carry out their functions, the raw material they have to work with is basically information, and the service they provide to their customers is purged knowledge. It can be said that currently, most organizations lack a defined strategy to manage their most important asset [8].

4. Software System for Knowledge Management in the Context of Public Administration in the Prov. of Bs. As.

4.1 Application Context

The Honorable Accounts Court of the Province of Buenos Aires (HTC) is a constitutional body. Its powers, detailed in Provincial Law No. 10869 (Organic Law of the Accounts Court), are [9]:

1) To review public income collections and investment accounts, both provincial and municipal, approve them or reject them and, if rejected, indicate the official(s) responsible for the account in question, as well as the corresponding amount and cause(s) for rejection.

2) To inspect provincial and municipal public offices that are responsible for the administration of public funds, and take all necessary measures to prevent any formal irregularities in accordance with the procedure determined by law.

4.2 Project Genesis

The HTC had various decentralized systems for the treatment of precedents, case law and verdicts, which are specific issues processed by the Legal Office, the Queries Office and the General Office, respectively, all of them reporting to the Presidency of the entity. These systems were implemented in obsolete programming languages, with database engines that used different technologies and whose technological life cycles were already over. For example, there were three implementations developed in Visual Basic 6, whose support ended in 2005 and was extended until 2008. As regards

databases, there were implementations in Microsoft SQL Server 7 and Microsoft Access.
These implementations required constant corrective maintenance, and there were also new functional requirements based on current technological needs, such as, for example, the centralization and cataloging of publications, agility for loading, and statistics.
User requirements and queries started growing rapidly, in addition to the need for change typical of any systems department. This led to an initial survey and brainstorming process, and two alternatives were considered as possible solutions:

1. Conventional solution: generating a new system or module for each Office, in accordance to priorities to be analyzed. This solution would perpetuate the decentralized paradigm of organizational information.
2. Solution based on knowledge management: generating a system that, after surveying the requirements of each of the Offices, would offer a comprehensive solution for the systematic management of all publications, based on knowledge management principles. This solution requires a comprehensive analysis of the Offices, their operation, information flow, needs and proposals, as well as a survey and analysis of their strengths, opportunities, weaknesses and threats of the systems that are currently implemented, in order to receive and take advantage of previous experiences. Technically, the implementation of this type of systems requires longer times to go into production due to the normal difficulties related to the unification of criteria and requirements, the time required for analysis, and the difficulties associated with the implementation of generic systems.

In both cases, as much information as possible has to be migrated from the obsolete systems to the new implementations.
After a final review, the second solution was selected, considering the costs related to any integral solution.

4.3 Summun: Integral Solution Based on Knowledge Management

Summun, the software product developed, is defined as a participative construction tool that is transversal, scalable and acts as the foundation for an organizational intelligence and learning strategy [10]. For the development, the technology called "Symfony 2" was used as framework based on free software and designed to optimize the development of web applications based on the MVC (Model, View, Controller) pattern.
For the database, an object-oriented data model was defined, and DBMS MySql was used, through Doctrine's ORM, which allows associating objects to a relational database [12].

4.4 Development Phases

Three development phases were defined:

Phase I: Individual Management of Information Units
In this phase, the concept of "Information Unit" (IU) is defined, which is the basic or primary information component to be managed by the system. Management includes additions, deletions, modifications, simple and advanced searches, status management (draft, loaded, authorized, internal access, public access), keywords, user subdivision (load and authorization) and precedent traceability, case law, verdicts, verdicts by the Accounts Court, and regulations. Units whose status is "Public Access" must be accessible to the general public through the website of the Organization.

Phase II: Participative Management and Knowledge Building
During this phase, knowledge management concepts are defined. The following elements are included [10]:

- Relations: they allow linking units to various modes.
- External links: they allow linking any given unit to one or more hyperlinks.
- Bibliographic references: they consist in quoting, in as much detail as possible, references in books, manuals, treaties, registries or any other type of physical compendium that is available at the Organization.
- Comments: this is a text field that any registered user can use to leave comments to increase the value of the unit.

Phase III: Integrated Management
The third phase is more complex and plans on generating "digests" that will integrate various IU for specific purposes. It may include implementations related to quality management at the Honorable Accounts Court.

4.5 Summun Implementation

On the home page, shown in Figure 1, there is a dashboard with statistical data and information regarding the IU more recently used. The "Information Unit" is the basic or primary information component to be managed by the system. IU management is the data feed for the software, the source of information that will provide future systems, supported by Summun, with the information required for making decisions.
The dashboard represents an innovation in the field of computer systems at the HTC. This software development concept consists in concentrating in a single page and in real time all of the aspects that are considered significant for management. It is presented as an executive and management tool that shows important information in a centralized manner.
It includes elements such as: last units loaded, comparative amount chart, ranking of most visited units, units created, ranking of creators, and marker list.

In its IU menu, Summun presents a list with the various units to be managed, namely: case law, query precedent, verdict precedent, HTC verdicts, and regulations.

Fig. 1: *Summun's home page*

The design style of all IU types follows the same concept – a screen with all information units corresponding to the type of unit in question, and a set of functionalities, represented in Figure 2:

Fig. 2: *ABM Module of query precedent*

For the example, there is a main page organized in such a way that the available information units are shown as a grid, together with their characteristics, as well as a set of actions for each unit. There are also shortcuts to perform searches.

Summun incorporates to its design the concepts of "participation" and "knowledge management". This is achieved by interrelating the IUs and allowing actions related to knowledge management definitions such as weighing, comments, relations and references.

This combination of data loading, relations, comments and references creates a virtuous cycle for knowledge management by the Honorable Accounts Court that is supported by the computer system presented here. Each Information Unit has its on characteristics in relation to knowledge management.
Knowledge management functionalities allow accessing a query interface for anything pertaining to knowledge management in relation to the Unit. Integration is depicted in Figures 3 and 4, where the knowledge management functionalities associated to a unit are shown.

Fig. 3: *Links and relations of an IU*

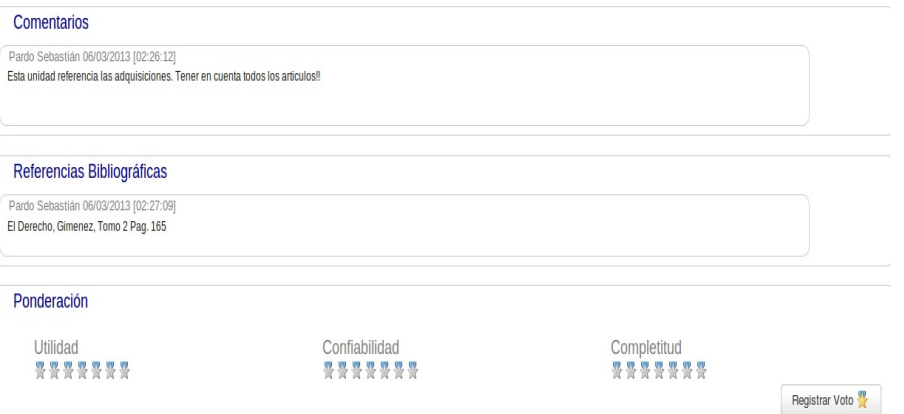


Fig. 4: *Comments, references and weight of an IU*

The figures show links, relations, comments and bibliographic references, aspects related to the knowledge management menu. This concept is implemented within the context of Summun's stage 2. It is subdivided into four functionalities, each of them building the idea of relations, comments, links and references as key principles for knowledge management at the Honorable Accounts Court.
Information Units can also be weighed based on their usefulness, reliability and completeness. Weighing is a concept inspired on Wikipedia ratings, and

in the future it will allow generating a dashboard which could include rankings with the "best" information units based on each criterion.

5. Results Obtained

After developing a special module, documentary units and physical files were migrated with an efficacy of 100% - exactly 14,504 Information Units obtained from the database management systems PostgreSQL, SQL Server and MySql. In 6 months using the system, approximately 800 Units were loaded, which highlights the significance of migration in relation to data volume.
As of July 3, 2013, 14,756 public visits and 1087 internal visits have been recorded, showing the high involvement level of society.
The HTC is ISO 9001:2008 certified, which considers the implementation of "non-conformities" as tool for the users to expose non-compliance with requirements. As of July 3, 2013 there are no non-conformities related to the Summun system.
Similarly, the IT Department offers a support software application that allows users to send in their technical issues, needs, requirements or opinions. Based on data from the IT manager and phone user support records, Figure 5 shows the requests and their distribution as of March 12, 2012:

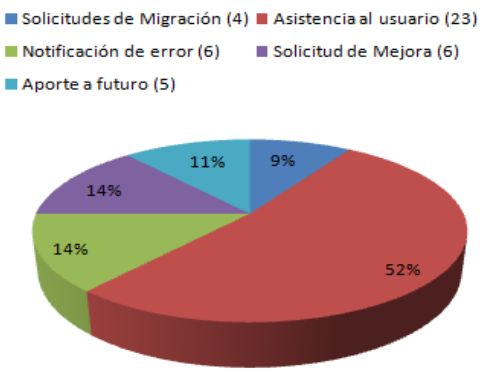


Fig. 5: *Distribution of requests as of March 12, 2012*

6. Conclusions

Knowledge management is a discipline, not a technology that can be bought and sold. It is an interaction between processes, individuals and organizations supported by technology. For the success of this interaction in an organization, a modern organizational culture is required, one that is promoted by the Management, favors a stimulating environment for

collaboration, and also provides the methods and tools for members to be able to share their explicit knowledge in an efficient manner.
This paper focuses on the application of knowledge management to the public administration. In this regard, it can be concluded that, even though public organizations do not have to worry about profitability, they do need to comply with their budgets and collections, and above all, they need to overcome issues related to environmental and political factors.
The development of the tool called "Summun" was presented as an example of a practical case, and from the experience obtained it can be concluded that this is an ambitious but realistic and practical project, whose goals were achieved and resulted in an information system used throughout the HTC. An innovative concept was introduced – knowledge management, and its advantages were highlighted and promoted through training using Summun for practical hands-on experience.
Finally, it can be said that knowledge management needs to be implemented as state policy, understanding the significance of technological and conceptual innovation as the fundamental foundation upon which public policies should be based, fostering management efficacy, equity and transparency.

7. Future Work

User experience indicated the need for a comprehensive search function that is independent from the IU. That is, the search function should be able to find keywords regardless of the type of unit, and show all data on a single screen.
The analysis and development of stage 3 of the project is being considered, planning the generation of "digests" that will be part of different units for specific purposes. Additionally, Summun will be linked to aspects that are directly related to quality management at the Honorable Accounts Court.
Finally, there is a project to add user recommendations, supported by algorithms of the "closest neighbor" type, currently used in websites such as Amazon, Netflix, Reddit, and so forth [11].

References

1. Catenaria, http://www.catenaria.cl/img/pdf/conocimiento.pdf
2. Bustelo Ruesta, Iglesias, A. (2001). Gestión del Conocimiento y Gestión de la Información. INFORAREA S.L., year VIII, n. 34.
3. Seminario USAC, http://seminario1usac.wordpress.com/2011/05/08/business-intelligence/
4. Moore, Bresó Bolinches (2001). El desarrollo de un sistema de gestión del conocimiento para los institutos tecnológicos.Revista Espacios, Vol.22.
5. Lopez, Cabrales, Schmal (2005). Gestión del Conocimiento: Una Revisión Teórica y su Asociación con la Universidad.Universidad de Talca, Chile.

6. Del Moral Bueno, Anselmo y otros (2007). Gestión del Conocimiento. Paraninfo.
7. Wikipedia, Sociedad de la Información y del Conocimiento, http://es.wikipedia.org/wiki/Sociedad_de_la_informaci%C3%B3n_y_del_conocimiento
8. Rabinovitch, J. (2009). Gestión del Conocimiento y Gobierno Electrónico: Mitos y Realidades. Departamento de Asuntos Económicos y Sociales (ONU), UNDESA.
9. Gob Prov. De Bs. As., http://www.gob.gba.gov.ar/legislacion/legislacion/l-10869.html
10. Flores, Coronel, Pardo, Groizard (2012). "Summun. Conocimiento Organizacional". Presentación en Honorable Tribunal de Cuentas de La Provincia de Buenos Aires.
11. Wikipedia, Sistema Recomendador, http://es.wikipedia.org/wiki/Sistema_recomendador
12. Databases and Doctrine, http://symfony.com/doc/current/book/doctrine.html

IV

Signal Processing and Real-Time System Workshop

Detection of pathological respiratory signs in productive poultry populations by digital processing of acoustic signals

CRISTIAN KÜHN AND CÉSAR MARTÍNEZ [1, 2]

[1] Laboratorio de Cibernética, Facultad de Ingeniería,
Universidad Nacional de Entre Ríos, Ruta 11, Km 10, Oro Verde, Entre Ríos
[2] Centro de Investigación en Señales, Sistemas e Inteligencia Computacional (SINC (I))
Departamento de Informática, Facultad de Ingeniería,
Universidad Nacional del Litoral, Santa Fe, Argentina
cristian.kuhn20@gmail.com, cmartinez@bioingenieria.edu.ar

Abstract:
The need for an early detection of the presence of a health problem in poultry production significantly improves the possibilities for its control. Therefore, this work presents the design and development of an automatic method for the task of an early recognition of the presence of respiratory disease signs related to the poultry production. The system starts with a signal recording in productive chickens' sheds, preprocessing for signal conditioning, measurement of parameters of interest (energy, pseudo-spectrum) and generating a signal detection which indicates the presence of pathological findings in the studied population. The results were satisfactory, having the system been able to detect the signs in different experimental conditions, from the study of a single sick individual to the mixture of healthy and diseased individuals.

Keywords:
Acoustic analysis, pathological respiratory signs, pseudo-spectrum, poultry population.

1. Introduction

The poultry industry is one of the most important productive chains in the country, having it consolidated as one of the most dynamic that the agricultural production has. In this industry, respiratory diseases of chickens are a matter of public health importance in a production setting since they have a high morbidity (80-100%) and mortality ranges develop between 5 to 20%, as the type and severity of the outbreak. Knowing the presence of respiratory signs in the poultry population is very important to take an early action on a future presence of a chronic disease. Today, even the knowledge of the authors, there is no reliable and easy application system that provides

this information. One of the main problems is the acquisition and automatic classification, because the continuous audiovisual recording is subjective, complicated and it is sensitive to make mistakes.
The digital signal processing provides tools that have been successfully applied to various tasks, giving the possibility of implementing workable systems in the environment of the animal production. In this context, various applications have been reported on the acoustic analysis closely related to the one presented in this document, such as the analysis of vocalizations of mammals [2], bat communication [3], or the repertoire of sounds of whales [4]. A working line previously scanned by the authors consists on the acoustic analysis of chewing sounds of ruminants to automate the feeding behavior [5, 6].
State of the art shows that the acoustic spectral analysis is attractive because of its simplicity, speed and relative robustness or noise. That is why in this paper the design and development of a method for detecting arises respiratory signs of disease patterns in sound. This is to maintain a relatively low computational complexity, which provides a system to be used to detect in real time the presence of abnormal signs in the poultry production.
The rest of the paper is organized as follows. Section 2 details the design of the proposed solution. In Section 3 the results obtained in different experimental conditions are shown. Finally, Section 4 summarizes the conclusions of this work and future work are outlined.

2. The proposed method and materials

The problem of classification of respiratory signs is similar to many problems in pattern detection and classification. The entire process consists of the following steps, implemented in the mathematical software MatLab:

- The data collection process includes the acquisition of audio using recording devices. The records are obtained in controlled environments in terms of background noise.
- The feature extraction is based on the spectrum-temporal examination of sound recordings. It basically involves the use of measurements pseudospectrum major peaks in the signal segments. From observing patterns a set of parameters that will be obtained then your discrimination.
- The recognition consists of the measurement of the above parameters on pseudo - stationary intervals audio. The method assumes an allowable variability patterns, to add robustness to the system and better fit the reality of the problem.

Figure 1 shows a detailed diagram in blocks of the whole process. In the nex sections, each stage is explained, exemplifying with resulting signals of each process given the novelty of the solution in the approached task.

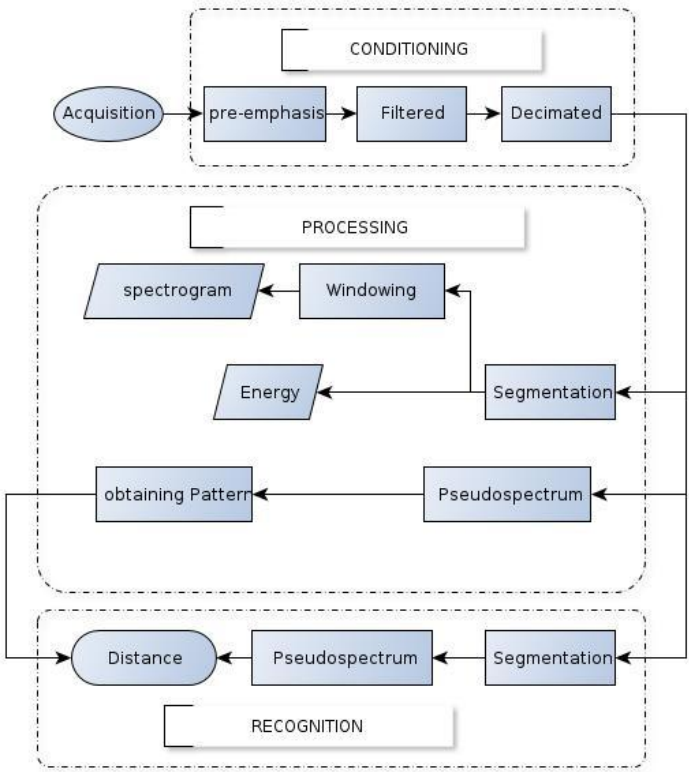


Fig. 1: *block diagram of the proposed system*

2.1. Acquisition and conditioning

Acquisition. Initially, the spectral characteristics of the pathological signs were unknown, so the maximum resolution available was used: 16 bits per sample, and a sample rate of 44100 Hz. The recordings used the microphone on board of an I-EEE Asus Netbook. Figure 2 shows an example of a sonogram, which removes the interval of time between 20 and 30 s, in order to prevent various noises: between 10-30 s. chicken riot sound is identified until it managed to stabilize it against the setup of acquisition, and between 30-50 s. interference from a car is registered.

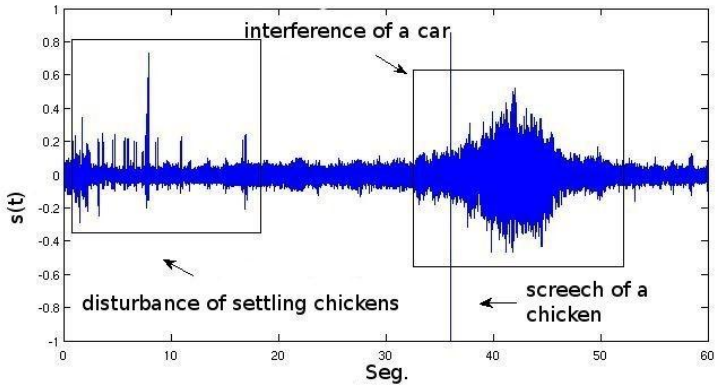


Fig. 2: *example of a sonogram with some interferences*

Pre - emphasis. The digitized $x(n)$ signal is subjected to digital filtering of low order (typically a first-order FIR filter), to flatten the spectrum signal as:

$$x = x(n) - ax(n-1); for a \in R.$$

Filtering and decimation. The implementation of this stage is given by the need to limit the signal $x(n)$ in band, leaving only those frequencies containing respiratory signs.
For this task two filters Butterworth were implemented, a high pass and low pass, respectively applied in that order. The determination of the parameters of the filters was experimentally made by inspection of the spectrogram of the signal, resulting in the necessary cuts and step frequency. The decimation is applied for sub-sampling the signal by an integer factor in the form $s(n) = x(nK)$, preserving the original signal from a sample of every moment nK.
Figure 3 shows the sonogram already filtered and decimated along with spectrogram cut in the frequency band of interest (0-2500 Hz). In both studies are evidenced continuous pathological breathing patterns (22 s., s. 24, etc.), immersed in a background white noise.

2.2. Processing of the signal

This block seeks to isolate patterns of interest which were evidenced in the signal. To do this, the pseudoespectrum of the signal will be calculated, and a characteristic pattern of the respiratory sign in the signal will be determined.
For the following processes, the preprocessed signal $s(n)$ is windowed in blocks of N samples with overlapping of 50% using Hamming windows.

Energy. A measure that helps to discern the blocks with respiratory signs is the energy of the signal, calculated as $E=\| s(n) \|_2^2$. Figure 4 shows a 2 example of analysis, where the peaks at the location of the events of interest can be seen.

Pseudoespectrum. The estimation of the frequency components of the signs respiratory immersed in the noise signal is the basis for the classification. The MUSIC (*Multiple Signal classification*) algorithm obtains the estimation of the pseudoespectrum of the conditioned signal [7].

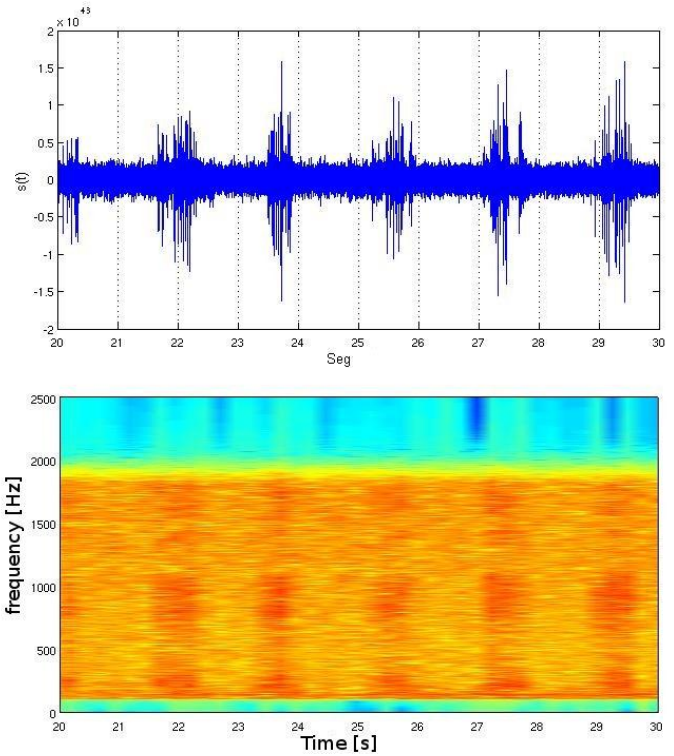


Fig. 3: *sonogram and spectrogram of a pre-processed segment.*

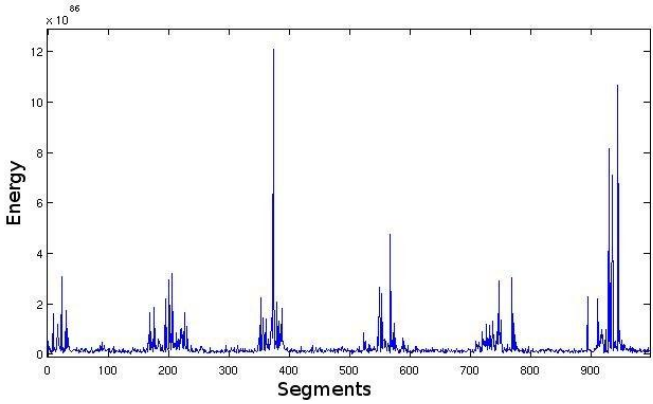


Fig. 4: *energy signal.*

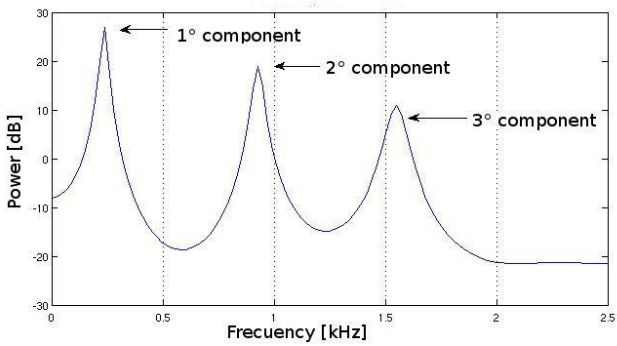


Fig. 5: *pseudo-spectrum obtained by the MUSIC algorithm.*

The MUSIC algorithm manages to estimate the frequency content of a pure signal contaminated with white Gaussian noise, through a breakdown in values and eigenvectors, to what is called pseudoespectrum [8]. The location of the peaks of the estimated function is based on the detection of respiratory signs in signal. Figure 5 shows an example of pseudoespectrum calculated on the test signal.

2.3. Recognition of respiratory signs

Once the characteristic pattern of the respiratory sign is determined, its presence within the audio signal of a complete realization has to be evaluated. So, it is necessary to carrry out the previous pre-processing

of the signal, segmentation into blocks and the calculation of the pseudoespectrum to each one. Finally, the signal detection D is obtained, which consists of the comparison -in each segment- of the pseudoespectrum obtained with respect to the characteristic pattern of the pathological sign. The generated signal is binary, indicating the presence (1's) or absence of the sign detected (0's) according to whether the euclidean distance d_j for the segment j is less or greater than a threshold of reference, respectively, according to:

$$d_j = \sqrt{\sum_{i=1}^{N} (p_i - q_i)^2} \ ,$$

where N is the number of samples of the segment, p is the pseudoespectrum pattern of the respiratory sign and q the pseudoespectrum calculated on the segment. The task of the respiratory sign recognition allowed minor differences at a threshold of maximum permissible difference in way of discerning between artifacts in signal (noise of cars, etc.).

3. Experiment and results

In order to be able to evaluate the performance of the recognition system, different situations were proposed, in order to observe and compare aspects of its operation. The response to a variation in the number of tested chickens will be observed, from a production batch of 15,000 chickens, whose age is of 25 days, which gave evidence of cases in which there were individuals with early signs of respiratory condition (poultry farm located in the province of Entre Ríos, Argentina) .

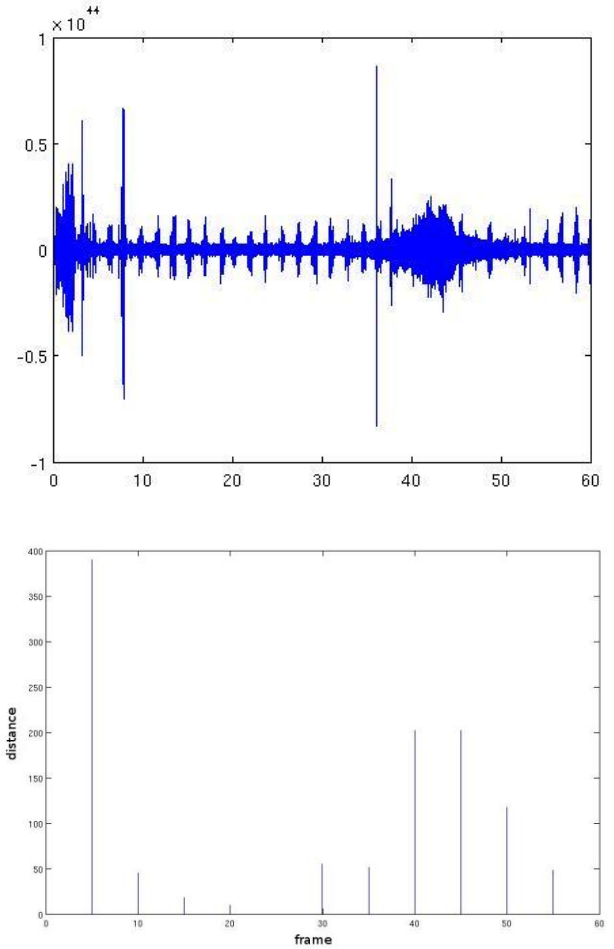


Fig. 6. *Analysis of distance between pseudoespectrums. Sonogram of the analyzed signal (top); calculated distances indicated in the center of each considered frame without thresholding (bottom).*

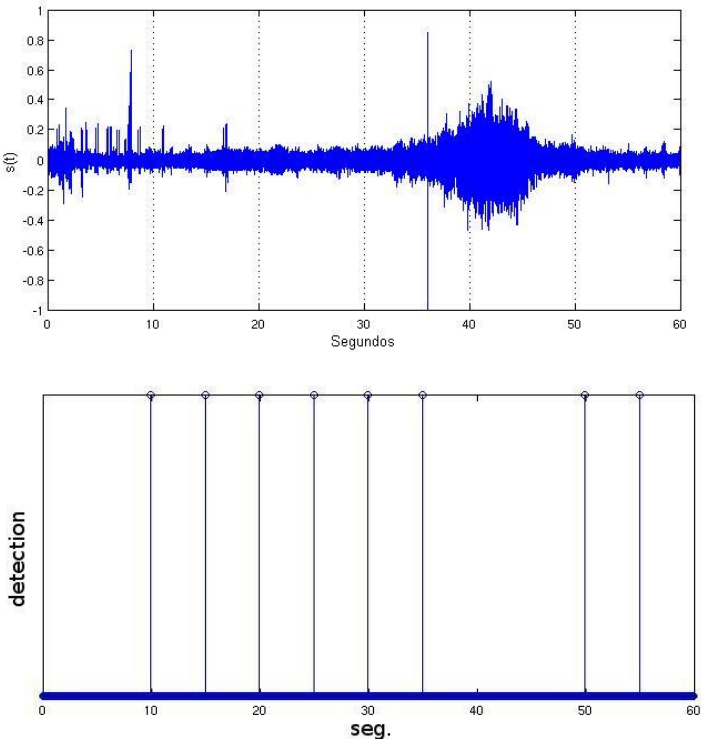


Fig. 7. *Raw sonogram of the signal of 1 sick chicken (top) and detection signal of respiratory signs (bottom).*

3.1 Test with a single sick chicken

In this stage, a chicken with pathological respiratory signs was isolated in a room away from the shed which houses the production batch. The distance between the microphone and the chicken was around 10 cm. Figure 7 shows an example of the results obtained. It can be seen how the system recognizes the presence of the respiratory sign only at those intervals where there is no noise, in this case ignoring noises at the beginning (about the first 5 s.) and at the end the disturbance by noise of automobile (approximately to the 40 s.).

3.2 Tests with several sick chickens.

Figure 8 shows the signal acquired of a group of 4 sick chickens and how the system recognizes the respiratory sign in those intervals without the presence of abnormal noises. Differing it from the earlier case, here there is a bigger periodicity in the respiratory signal events,

as well as there is also an increase in breadth. This is due to a partially synchronized breathing by small subgroups of chickens, a particular feature of the disease.

3.1. Test with mixtures of sick and healthy chickens

Figure 9 shows the case of a sign belonging to a multitude of 7 sick chickens, which was mixed with healthy individuals (approximately 10 birds/m2), recorded with a microphone hanging 10 cm. above the chickens.

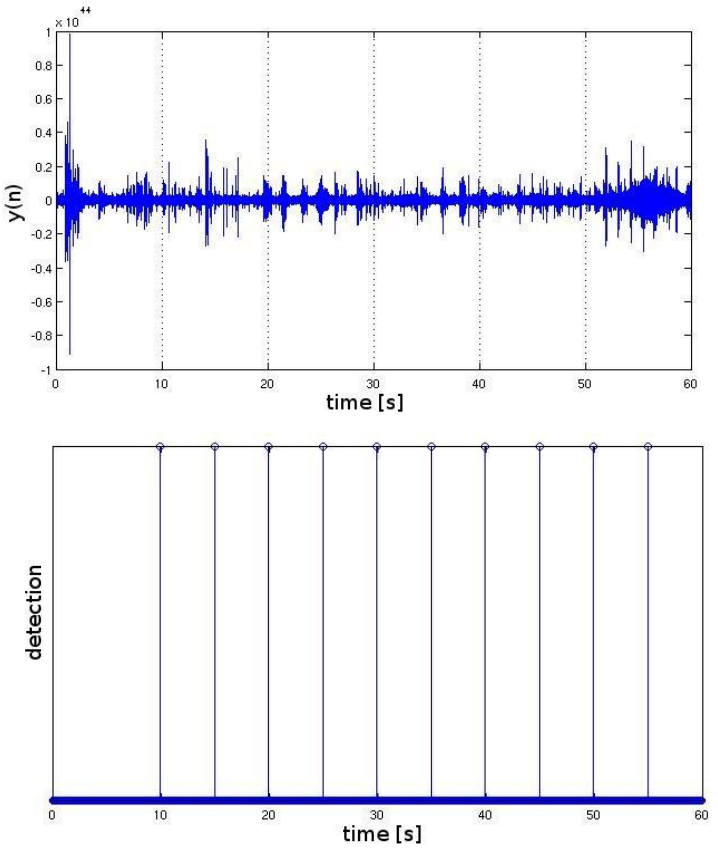


Fig. 8. *Sonogram of the signal without having been processed of 4 sick chickens (top) and signal detection of respiratory signs (bottom).*

4. Conclusions

In this work, the design and development of a computational technique of audio signal processing has been presented, which proved to be useful for poultry production providing an automated tool for an early detection of pathological respiratory signs.

From the acoustic recording of chickens in barn, using techniques of filtering and frequency estimation, the morphology of pathological respiratory signs could be identified. Also, the system provides the identification of such signs in individuals of the productive environment.

A line of continuation of this work, subsequent to the detection of the pathological sign, is the statistical quantification of the incidence of the disease is in the population. This analysis would serve to determine, through a sampling of the population of a shed, if it presents signs and also to establish different levels of involvement. On the other hand, it is necessary to expand the experimentation towards higher populations inside the sheds, making adjustments in the system to increase robustness in the natural environment of the production.

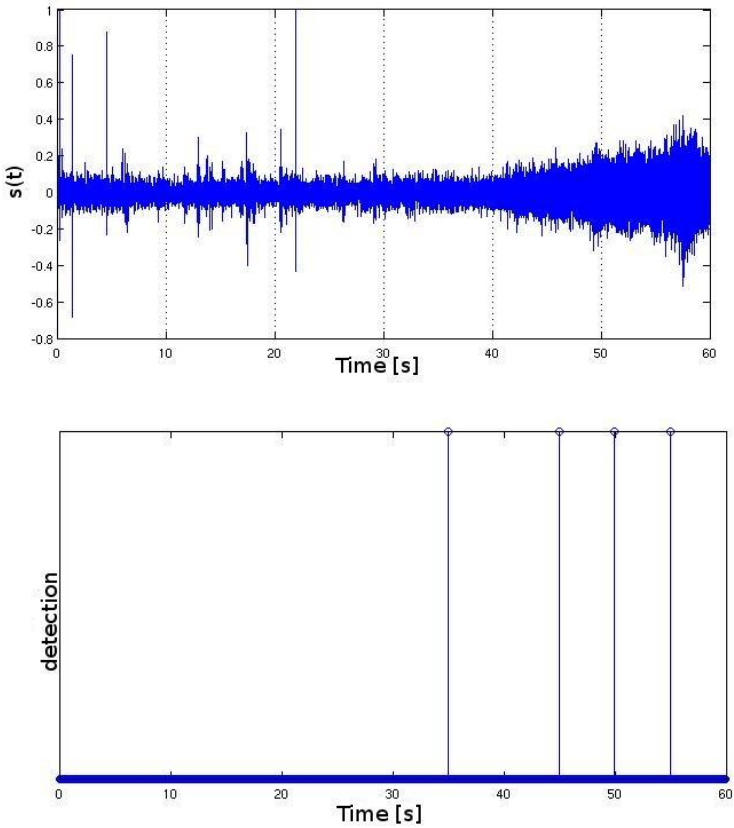


***Fig. 9**. Sonogram of the signal without having it been processed of a mixture of sick and healthy chickens (above) and signal detection of respiratory signs (down)*

Acknowledgements

The authors wish to thank Agencia Nacional de Promoción Científica y Tecnológica (under project PAE 37122) and the Universidad Nacional del Litoral (PACT #58, CAI+D 2011 #58-511, #58-525).

References

1. SENASA. (2003). Plan nacional de sanidad avícola.
2. Schrader, L. and Hammerschmidt, K. (1997).Computer-aided analysis of acoustic parameters in animal vocalizations: a multi-parametric approach. Bioacoustics, 7(4):247–265.
3. Kanwal, J. S., Matsumura, S., Ohlemiller, K. and Suga, N. (1994). Analysis of acoustic elements and syntax in communication sounds emitted by mustached bats. The Journal of the Acoustical Society of America, 96:1229.
4. Clark, C. W. (1982). The acoustic repertoire of the southern right whale, a quantitative analysis. Animal Behavior, 30(4):1060–1071.
5. Milone, D. H., Galli, J., Martínez, C. E., Rufiner, C. E., Laca, E. and Cangiano, C. (2008). Reconocimiento automático de sonidos masticatorios en rumiantes. In Anales de las 37 Jornadas Argentinas de Informática, JII-Agroinformática, pages 372–384, Santa Fe, Argentina, september 8-12.
6. Milone, D. H., Galli, J., Cangiano, C., Rufiner, H. L. and Laca, E. (2012). Automatic recognition of ingestive sounds of cattle based on hidden markov models. Computers and Electronics in Agriculture, 87:51–55, sep.
7. Schmidt, R. (1986). Multiple emitter location and signal parameter estimation. Antennas and Propagation, IEEE Transactions on, 34(3): 276–280.
8. Marple, L. (1987). Digital Spectral Analysis with Applications. Prentice-Hall.

II

Computer Security Workshop

Improving versatility in keystroke dynamic systems

ENRIQUE CALOT, JUAN MANUEL RODRÍGUEZ, JORGE SALVADOR IERACHE[1]

Laboratorio de Sistemas de Información Avanzados,
Facultad de Ingeniería, Universidad de Buenos Aires,
Buenos Aires, Argentina
{ecalot,jmrodri}@fi.uba.ar,jierache@yahoo.com.ar

Abstract. *Keystroke dynamics is a biometric technique to identify users based on analyzing habitual rhythm patterns in the way they type. In order to implement this technique different algorithms to differentiate an impostor from an authorized user were suggested. One of the most precise method is the Mahalanobis distance which requires to calculate the covariance matrix with all that this implies: time processing and track each individual keystroke event. The hypothesis of this research was to find an algorithm as good as Mahalanobis which does not require track every single keystroke event and improve, where possible, the processing time. To make an experimental comparison between Mahalanobis distance and euclidean normalized, a distance which only requires calculate the variance, an already studied dataset was used. The results were that use normalized euclidean distance is almost as good as Mahalanobis distance even in some cases could work better.*

Keywords*: Keystroke Dynamics, Web based authentication, Mahalanobis distance, Biometrics, typing biometrics*

1. Introduction

The variables that help make a handwritten signature a unique human identifier also provide a unique digital signature in the form of a stream of latency periods between keystrokes. The handwritten signature has a parallel on the keyboard. The same neurophysiological factors that make a written signature unique are also exhibited in a user's typing pattern[1].
Password typing is the most widely used identity verification method in World Wide Web based electronic commerce. Due to its simplicity, however, it is vulnerable to impostor attacks. Keystroke dynamics and password checking can be combined to result in a more secure verification system[2].

[1] This paper was done with Cloodie R&D Support

This authentication is fragile when there is a careless user and/or a weak password. Biometric characteristics are unique to each person and have advantages as they could not be stolen, lost, or forgotten[3, 4].
The biometric technology employed in this paper is the typing biometrics, also known as keystroke dynamics. Typing biometrics is a process that analyzes the way a user types at a terminal by monitoring the keyboard inputs in attempt to identify users based on their habitual typing rhythm patterns[5, 4].
Even though WWW keystroke dynamic systems may run locally on the web browser, due to security measures it should be ran on the webserver layer. This paper discusses an approach to reduce data transmission size.
Using a know dataset[6] we designed an experiment to compare three methods to compute the keystroke dynamics of users and compare them with impostors.
Our hypothesis is that one of the most used and precise method –the Mahalanobis distance– is as successful as the second method –normalized euclidean distance–. Ignoring the success rates, there are some advantages that the normalized euclidean distance has over the Mahalanobis distance, so if the hypothesis is confirmed using this method should prove to be a more useful way to calculate keystroke dynamics.
Some advantages of the normalized euclidean distance ar the lesser transferred information, processing time and the bigger versatility when changing passwords.

2. Current implementations

There are different methods to compare keystrokes, all based on measuring the distances between two strokes, a negative result is found when both differ more than a threshold. One of the best methods is the Mahalanobis distance[2, 6].
Three methods are shown below, each method is a generalization of the previous one.

2.1 Euclidean distance

The time a user press a key or the time between one key and the other may result in a vector of events ($\vec{\Gamma}$). Each event represent a key hold time or the elapsed time between two keys. Since in the training phase an event may occur several times, the vector is a list of the expected values of every event time.
Calculating the euclidean distance between two vectors works as a comparison algorithm, with relatively high success rates[6].

$$d(\vec{\Gamma}_1, \vec{\Gamma}_2)^2 = \| \vec{\Gamma}_1 - \vec{\Gamma}_2 \|^2 = \sum_{i=1}^{N} (\Gamma_{1,i} - \Gamma_{2,i})^2 \quad (1)$$

Where $\vec{\Gamma}_1$ is the vector of training event times and $\vec{\Gamma}_2$ is the vector of testing event times.
To optimize calculation timings the squared norm is actually used.

2.2 Normalized euclidean distance

A disadvantage of the former method is that important information is ignored. The variance of each event time should be taken into consideration, and that is exactly what the normalized euclidean distance does: adding the variance (s_i^2) of each event time.

Using the inverted variance of the training set ($\vec{\Gamma}_1$) as a weight factor, the normalized euclidean distance is defined as

$$d(\vec{\Gamma}_1, \vec{\Gamma}_2)^2 = \sum_{i=1}^{N} (\Gamma_{1,i} - \Gamma_{2,i})^2 s_i^2 \quad (2)$$

where s_i^2 is the variance of each element of $\vec{\Gamma}_1$.

2.3 Mahalanobis distance

Again, the former method is skipping information, this time is the covariance between events.
Mahalanobis distance is defined as

$$d(\vec{\Gamma}_1, \vec{\Gamma}_2)^2 = (\vec{\Gamma}_1 - \vec{\Gamma}_2)^T S^{-1} (\vec{\Gamma}_1 - \vec{\Gamma}_2) \quad (3)$$

Where S^{-1} is the inverted covariance matrix corresponding to all events in the training set $\vec{\Gamma}_1$ [7].

3. Problems of Mahalanobis distance in keystroke dynamics

Translating each method to a kernel matrix it turns out that in equation 3 the matrix S is the identity in the euclidean distance, a diagonal with the variances in the normalized euclidean distance and the covariance matrix in the Mahalanobis distance.

3.1 Distance kernel matrix size

To generate the covariance matrix for the Mahalanobis distance all key-press events and their respective timings should be transmitted to the server while training –or at least the covariance matrix and the expected event timings–. But to generate the diagonal matrix for the normalized euclidean method is it possible to send only three integer numbers per event or two floats.

Using the property $Var[X] = E[X^2] - E[X]^2$ it is possible to generate the variance using only the sum of squares $SS = \sum_{i=0}^{n} \Gamma_{1,i}^2$, the sum $S = \sum_{i=0}^{n} \Gamma_{1,i}$ and the total n since $E[X^2] = \frac{SS}{n}$ and $E[X] = \frac{S}{n}$. All three numbers are natural and may be combined in an $\mathbf{N}^3$ vector which supports commutative addition properties. This method allows parallelized variance calculus[9].

Table 1. Different parameters to be sent to the server

Distance Method	Variables
Euclidean	$(S,n) \in \mathbf{N}^{2\times n}$ or $\Gamma = E[X] \in \mathbf{R}^n$
Normalized euclidean	$(S,SS,n) \in \mathbf{N}^{3\times n}$
Mahalanobis	$\Gamma = E[X] \in \mathbf{R}^n$ and $Cov[X] \in \mathbf{R}^{n\times n}$

There are several ways to send the data depending on the algorithm to be used. Table 1 compares them.

For example, when 20 events are used, the covariance matrix has $20 \times 20 = 400$ values and the $\Gamma = E[X]$ vector has 20 values. Normally a $\mathbf{R}^{n\times n}$ matrix can be encoded with n^2 numbers, but as

$Cov[a,b] = Cov[b,a]$, covariance matrix is symmetric and therefore it can be encoded with $\frac{n(n+1)}{2}$ numbers. Assuming a real number and an integer has the same size, the transmission would be of $\frac{20 \times 21}{2} + 20 = 230$ numbers while the normalized euclidean only transmits $3 \times 20 = 60$ numbers. Generalizing, the data transmission of Mahalanobis distance is $\frac{n(n+1)}{2} + n$ reals, that is $O(n^2)$, normalized euclidean is $3n$ integers, that is $O(n)$ and euclidean is $2n$ integers or n reals, that is also $O(n)$.

3.2 One password algorithm

Another problem is that training is done with only one password. A new password should require the user to re-train all the covariance matrix with Mahalanobis.
Normalized euclidean distance may reuse the variances of the common keys between two different passwords while Mahalanobis distance may not.

3.3 Backspace eliminating digraphs

When the user trains it is possible that mistakes a character and use backspace to correct it, in this case one event will be missing. For example the word "train" has 5 characters so the events will be t.hold, t.up-r.down, r.hold, etc. The problem occurs when "te[backspace]rain" is typed, the event t.up-r.down was not recorded because there were two keys in the middle "e" and [backspace].
Having a variable number of events per key is a problem to calculate the covariance matrix, but allows to process backspaces in passwords (sacrificing the success rate due to lesser information available) and free text.
Table 2 shows an example of Mahalanobis method with three pairs of events and normalized euclidean with three and two times per event respectively.

3.4 Processing times

Calculating the covariance matrix and inverting it should take a considerable amount of time for the Mahalanobis method, the time here is expected to be $O(n^2)$.

Table 2: Example of how timing counts are dependent on the event in Mahalanobis distance

Method	Key	Times	Matrix S	Inverse S^{-1}
Mahalanobis	$!A.hold!$ $!A.up-B.down!$	$\{90\}, \{99\}, \{97\}\{161\}, \{171\}$	$\begin{bmatrix} \frac{67}{3} & \frac{175}{6} \\ \frac{175}{6} & \frac{139}{3} \end{bmatrix}$	$\begin{bmatrix} \frac{556}{2209} & -\frac{350}{2209} \\ -\frac{350}{2209} & \frac{268}{2209} \end{bmatrix}$
Normalized euclidean	$!A.hold!$ $!A.up-B.down!$	$\{90\}, \{99\}, \{97\}\{161\}, \{171\}$	$\begin{bmatrix} \frac{67}{3} & 0 \\ 0 & 50 \end{bmatrix}$	$\begin{bmatrix} \frac{3}{67} & 0 \\ 0 & \frac{1}{50} \end{bmatrix}$

Normalized euclidean should also take time to compute the variances, but this procedure is $O(n)$. Inverting the matrix lacks of relevant costs due to the properties of the diagonal matrices. Euclidean distance should be the fastest algorithm because of its simplicity.
It is important to remark that due to parallelized calculation of the variance, part of the calculating time in training mode for the normalized euclidean distance may be done while reading the keyboard by the user machine.
The experiment also intends to measure algorithms processing time.

4. Experimental comparison

We use an already studied dataset for two main reasons, one is because it was collected in a controlled laboratory environment, the second reason is because 14 detection algorithms were tested using this dataset[6] and that give us a big framework to start our research. The data was collected from 50 different users along 8 days or sessions –totalizing 400 cases per user–, in each session the users typed always the same string: ".tie5Roanl" which represents a reasonable secure password. When any error in the sequence was detected, the subject had been prompted to retype the password. To make this a laptop was set up with an external keyboard to collect data and a Windows application was developed to prompt the subjects to type the password. The application displays it in a screen with a text-entry field. In order to advance to the next screen, the subject must type the 10 characters of the password correctly in sequence and then press Enter. The data set contains the hold time of each key, the time between two consecutive keys were pressed and

the time since one key was released and the next was pressed. One of the three values depends linearly of the other two. Due to preconditions of covariance one value was dropped away leaving two values per key.
From the 400 cases per user, the first 200 cases were used to train the detection algorithm and the second 200 cases were used to validate it, also the first 5 cases of all the other users were taken to generate an impostor dataset in order to validate negative cases. This data set and schema was taken from Killourhy and Maxion[6].
We performed 19 tests, the first using two events (the first two values of the Γ vector) and increasing the number of events until the last one, using all twenty. We expected to have a best success rate in the last test because it counts with more information. We ran the three mentioned algorithms in each test.
Finally we calculated the area under the receiver operating characteristics (ROC) curve –a performance measure for machine learning algorithms commonly used in systems that learns by being shown labeled examples[8]–. This method, known as AUC, was chosen because it is a classifier performance evaluator independent of the decision threshold chosen on the keystroke distances.

5. Results

With the one key test case we obtained in one sample user $\vec{\Gamma} = [98.98, 166.905]$, where first value corresponds to the expected key-hold time and the second to the expected elapsed time until the next key was pressed. Both times are expressed in milliseconds.

$$S^{-1}_{Mahalanobis} = Cov\left[\vec{\Gamma}\right]^{-1} = \begin{bmatrix} 341.29 & 282.19 \\ 282.19 & 5464.9 \end{bmatrix}^{-1} = \begin{bmatrix} 0.0031 & -0.00016 \\ -0.00016 & 0.0002 \end{bmatrix} \quad (1)$$

$$S^{-1}_{normalizedEuclidean} = \begin{bmatrix} 341.29 & 0 \\ 0 & 5464.9 \end{bmatrix}^{-1} = \begin{bmatrix} 0.0029 & 0 \\ 0 & 0.00018 \end{bmatrix} \text{(⊡ SEQ equation\n * MERGEFORMAT ⊡2⊡)}$$

$$S^{-1}_{euclidean} = S_{euclidean} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \text{(⊡ SEQ equation\n * MERGEFORMAT ⊡3⊡)}$$

Note that $S_{Mahalanobis}$ and $S_{normalizedEuclidean}$ have the same diagonal values but this is not the case with their inverses.

Table 3: Experimental results

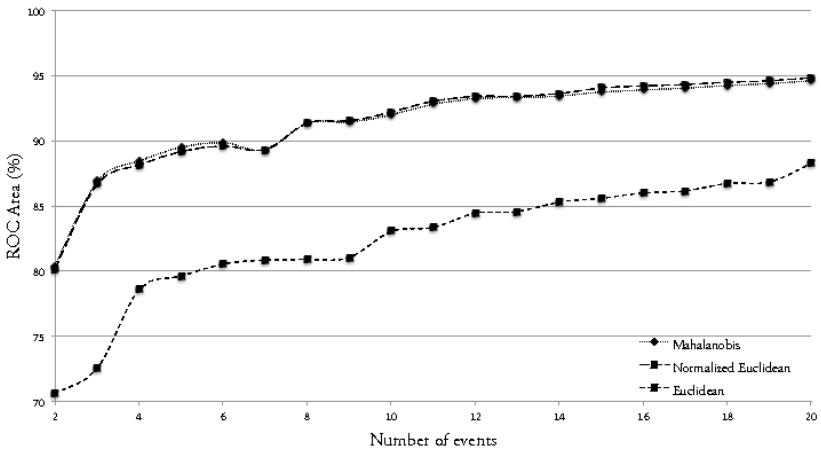
N	Method	Total Errors	ROC	Zero-miss	False-Alarm	Time
2	Mahalanobis	0.01887	80.43%	7461 of 12750	769 of 10200	1.356s
2	Normalized euclidean	0.01899	80.17%	7451 of 12750	788 of 10200	1.300s
2	Euclidean	0.02240	70.61%	9030 of 12750	649 of 10200	0.872s
20	Mahalanobis	0.00970	94.60%	5576 of 12750	464 of 10200	1.896s
20	Normalized euclidean	0.00919	94.84%	5581 of 12750	428 of 10200	1.764s
20	Euclidean	0.01440	88.27%	6853 of 12750	844 of 10200	1.704s

Table 3 shows the results of the 3 methods with 2 and 20 timing events respectively. Each method shows the area under ROC curve in percentage among with zero-miss and false-alarm rates. It is also shown the total processing time of training, testing all the 12750 positive and 10200 negative sets and calculating the results. In the last test –with 20 events–, normalized euclidean distance method performed even better than Mahalanobis.

As expected, our hypothesis that in the test with 20 timing events is better than the test with 2 was confirmed and that Mahalanobis and normalized euclidean distance are both superior than euclidean distance was confirmed too. Processing is, as expected, bigger for Mahalanobis and decreasing for normalized euclidean and finally, the fastest method, euclidean distance.

As expected our hypothesis that 20 is better than 2 was confirmed and that Mahalanobis and normalized euclidean distance is superior than euclidean distance was confirmed too. Processing are, as expected, bigger for Mahalanobis and decreacing for normalized euclidean and finally, the fastest method, euclidean distance.

Fig. 1. Distance methods compared in success versus amount of information



In Figure 1 it is possible to appreciate how similar are the normalized euclidean and Mahalanobis distances compared to the euclidean.

6. Conclusions

Normalized euclidean distance and Mahalanobis distance are almost the same in all ran tests. In the case of 20 events the results varied 0.24%. Normalized euclidean was faster than Mahalanobis distance for $132ms$ but slower than euclidean for only $60ms$. Versatility in normalized euclidean is also an advantage, passwords may be changed and the already-trained keys be kept in the new training. Those results lead to the conclusion that normalized euclidean distance is strong enough to be used and its advantages in data sizes and versatility are considerably important to be chosen against Mahalanobis distance and its success rate suggests that it should be employed against euclidean distance.

6.1 Future lines of research

We are exploring the way users may vary keystroke dynamics over the time. Using variance parallelization principle[9] there is a way to "forget" the training, making it autoadaptive with this time-wise learning technique. We are also exploring new fields on keystroke dynamics that include user emotional state detection.

Acknowledgments

This paper acknowledges support from Cloodie R&D.

References

1. Joyce, R., Gupta, G. (1990). Identity authentication based on keystroke latencies. Commun. ACM 33, 2 (February 1990), 168-176. http://doi.acm.org/10.1145/75577.75582
2. Cho, S., Han, C., Han., D. H., Kim, H. I. (2000). Web based Keystroke Dynamics Identity Verification using Neural Network. Journal of Organizational Computing and Electronic Commerce, Vol. 10, No. 4, pp. 295–307.
3. Polemi, D. (1997). Biometric Techniques: Review and Evaluation of Biometric Techniques for Identification and Authentication, Including an Appraisal of the Areas Where They are Most Applicable. Institute of Communication and Computer Systems, National Technical University of Athens, Athens, Greece. Retrieved on 2013-07-01: ftp://ftp.cordis.lu/pub/infosec/docs/biomet.doc, EU Commission Final Rep.
4. Araujo, L. C. F., Sucupira, L. H. R., Lizarraga, M. G., Ling, L. L., Yabu-Uti, J. B. T. (2005). User authentication through typing biometrics features. Signal Processing, IEEE Transactions on, vol. 53, no. 2, pp.851–855.
5. Monrose, F., Rubin, A. D. (2000). Keystroke dynamics as a biometric for authentication. Future Gen. Comput. Syst., vol. 16, no. 4, pp. 351-359.
6. Killourhy, K. S., Maxion, R. A. (2009). Comparing Anomaly-Detection Algorithms for Keystroke Dynamics. In International Conference on Dependable Systems & Networks (DSN-09), pp. 125–134, Estoril, Lisbon, Portugal, 29 June to 02 July 2009. IEEE Computer Society Press, Los Alamitos, California.
7. Mahalanobis, P. C. (1936). On the generalised distance in statistics. Proceedings of the National Institute of Sciences of India 2 (1): 49-55. Retrieved on 2013-07-01: http://www.new.dli.ernet.in/rawdataupload/upload/insa/INSA_1/20006193_49.pdf
8. Bradley, A. P. (1997). The use of the area under the ROC curve in the evaluation of machine learning algorithms. Pattern Recognition, Volume 30, Issue 7, July 1997, Pages 1145–1159, ISSN 0031-3203, http://dx.doi.org/10.1016/S0031-3203(96)00142-2.
9. Chan, T. F., Golub, G. H., LeVeque, R. J. (1979). Updating Formulae and a Pairwise Algorithm for Computing Sample Variances. Technical Report STAN-CS-79-773, Department of Computer Science, Stanford University. Retrieved on 2013-07-01: ftp://reports.stanford.edu/pub/cstr/reports/cs/tr/79/773/CS-TR-79-773.pdf

II

Innovation in Computer Science Education Workshop

Challenges and Tools for the Early Teaching of Concurrency and Parallelism

LAURA DE GIUSTI[1], FABIANA LEIBOVICH[1], MARIANO SÁNCHEZ[1], FRANCO CHICHIZOLA[1], MARCELO NAIOUF[1], ARMANDO DE GIUSTI[1,2]

[1] Instituto de Investigación en Informática LIDI (III-LIDI)
Facultad de Informática – Universidad Nacional de La Plata
[2] Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET)
Argentina
{ldgiusti, fleibovich, msanchez, francoch, mnaiouf, degiusti}@lidi.info.unlp.edu.ar

Abstract. *The early introduction of basic concurrency and parallelism concepts is discussed, following curricular trends promoted by technological changes.*
This paper discusses issues pertaining to the introduction of multiple core processors and the new architectures associated with clusters, multiclusters and clouds based on multicore/GPGPU architectures.
A tool combining an interactive visual environment for concurrent programming with the use of demonstration robots, especially for communication and synchronization concepts, is presented.
Finally, applications that expand the scope of the environment developed are discussed, as well as their use in various courses and future R&D lines on this topic.

Keywords*: Concurrency, Parallelism, Curricula, Environment, Multirobot, Concurrent and Parallel Algorithms.*

1. Introduction

Concurrency has been a central issue in the development of Computer Science, and the mechanisms used to express concurrent processes that cooperate and compete for resources have been in the core curriculum of Computer Science studies since the seventies, in particular after the foundational works of Hoare, Dijkstra and Hansen [HOA78][HOA85] [DIJ65][DIJ78][HAN77].
These concepts were traditionally taught starting with the availability of a single central processor, which could partially exploit the concurrency offered by any given algorithm, based on the available physical architecture (even with specific hardware such as co-processors, peripheral controllers, or vector schemes that would replicate arithmetic-logical computational units).
Parallelism, understood as "real concurrency" in which multiple processors can operate simultaneously on multiple control threads at the same point in time, was for many years a possibility that was limited by available hardware technology [HWA84][HWA93][DAS89].

Classic Computer Science curricula [ACM68][ACM78][ACM99] included the concepts of concurrency in various areas (Languages, Paradigms, Operating Systems), and parallelism was almost entirely omitted, except to introduce the concepts of distributed systems.
The advent of the ADA language [OLS83] in the mid-eighties marks a milestone in the evolution of this area, since the different mechanisms for expressing concurrency are clearly specified in a language while offering the possibility of associating processes ("tasks" in ADA) to different physical processors.
The current processor architectures, which integrate multiple "cores" within one physical processor [GEP06][MCC08][GPG], have had a notorious impact on the development of Computer Science, resulting in a reformulation of a processor's "base model". This has resulted in the replacement of the "Von Neuman machine" [GOL72] concept, with just one control thread, with a scheme as the one shown in Figure 1 that integrates multiple "cores," each with one or more control threads and several memory levels that are accessible in a differentiated manner [AMD09].

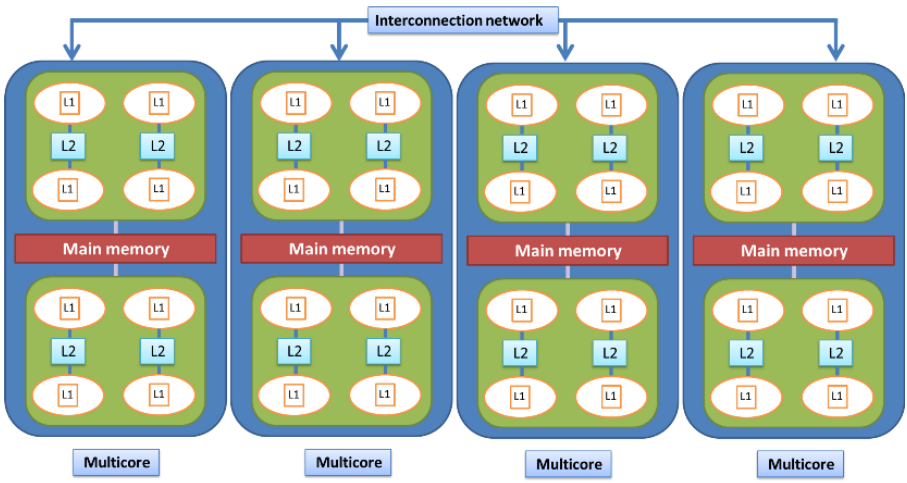


Fig. 1. *Scheme of a basic processor nowadays.*

At the same time, changes in technology have produced an evolution of the major topics in Computer Science, mainly due to the new applications being developed from having access to more powerful and less expensive architectures and communications networks [DEG13][HOO13].
This has led international curricular recommendations [ACM04] [ACM08][ACM13] to mention the need to include the topics of concurrency and parallelism during the early stages of student education, since all architectures and real systems with which they will work in the future will be essentially parallel.
However, it is at this point that one of the most significant issues arises, since parallel programming (and the essential concepts of concurrency) is more

complex for students who are starting their studies. New tools are required to allow teaching this topic at an early stage [CAR03][DEG12a].

2. Purpose of the Multirobot Environment Being Developed

In this paper we present CMRE (Concurrent Multi Robot Environment) as a tool aimed at introducing the concepts of concurrency and parallelism, with a visual and interactive approach, combined with the use of physical robots to demonstrate the concepts and development examples.

2.1 Visual Environment in Concurrency and Parallelism

Our starting point was the work carried out at III-LIDI [DEG12b][DEG12c] for teaching the concepts of concurrency in a CS1 course, with the idea of expanding it along the following lines:

- Being able to declare "processors" or virtual robots that represent the "cores" of a real multiprocessor architecture. These virtual robots can have their own clock, and different times for carrying out their specific tasks.
- Having the ability to declare shared or exclusive resources, including the possibility of selective mutual exclusion.
- Establishing virtual objects that represent basic data that can be counted and handled easily using virtual robot primitives.
- Having communication primitives available through synchronous and/or asynchronous messages.
- Having synchronization primitives through shared memory.

2.2 Addition of Physical Robots to the Visual Environment

In addition to the points mentioned above, real-time communication with physical robots of the Lego Mindstorms EV3 line [LEGa][LEGb] is added, so as to be able to run parallel algorithms in the environment with direct effect on the physical robots that replicate on the land the behavior defined by the algorithms.
This demonstration model helps students understand certain problems, such as the concepts of *fairness, deadlock or starvation* [AND00].

3. Concurrency/parallelism Architecture and Primitives in CMRE. Examples

CMRE is the evolution of the Visual da Vinci environment, whose main purpose had been to solve problems that specified the behavior of a *single robot,* which could move in a city formed by 100 avenues (vertical) and 100 streets (horizontal) and could distinguish objects (flowers and papers) and

perform operations on them (pick them up and/or put them down). The robot can also "count" and "inform" results. Table 1 summarizes the metaphor at which the new environment aims.

Table 1. Analogy between CMRE and the concepts of Concurrency and Parallelism

Concepts of Concurrency and Parallelism	CMRE
Multiple processors/cores	Multiple robots (implemented with one process per robot)
Shared memory	Shared areas of the city
Distributed memory	Exclusive areas per robot
Shared and distributed memory	Partially shared areas
Message communication among processes	Incoming and outgoing messages among robots.
Mutual exclusion on shared resources	Blocked city corners.
Selective mutual exclusion	Access to partially shared areas.
Synchronous execution model	Synchronous virtual clock.
Heterogeneous architectures	Assigning specific time frames to each robot actions.
Local or global data	Countable objects in the city (flowers/papers).

Figure 2 schematically defines the general structure of a program in CMRE upon which the corresponding primitives will be based.

```
programa areas1
areas      {definition of the structure of the city}
  nameArea1: typeArea(Coordinate0, Coordinate1, Coordinate2, Coordinate3)
  nameArea2: typeArea(Coordinate0, Coordinate1, Coordinate2, Coordinate3)
robots       {definition of the behavior for each type of robot}
  robot type1
  comenzar
    {body}
  fin
  robot type2
  comenzar
    {body}
  fin
variables      {robot creation}
   nameVariableRobot1: type1
   nameVariableRobot2: type2
comenzar
  {Assignment of exclusive areas to each robot}
  AsignarArea (nameVariableRobot1, nameArea1)
  AsignarArea (nameVariableRobot2, nameArea2)
  Iniciar(nameVariableRobot1, PosAv, PosSt)
  Iniciar(nameVariableRobot1, PosAv, PosSt)
fin
```

Fig. 2. General structure of a program in CMRE.

As mentioned above, the capabilities of CMRE can be summarized as follows:

- There are multiple processors (robots) that carry out tasks and that can co-operate and/or compete.
- The environment model ("city") where the robots carry out their tasks supports exclusive areas, partially shared areas and fully shared areas. An exclusive area allows only one robot to move in it, a partially shared area specifies the set of robots that can move in it, and a fully shared area allows all robots defined in the program to move in it.
- If only one robot is used in an area that encompasses the entire city, the scheme used in Visual Da Vinci is repeated.
- When two or more robots are in a (partially or fully) shared area, they compete for access to the corners on their runs, and the resources found there. For this, they must be synchronized.
- When two or more robots (in a common area or not) wish to exchange information (data or control), they must use explicit messages.
- Synchronization is done through a mechanism that is equivalent to a binary semaphore.
- Mutual exclusion can be generated by stating the areas reached by each robot. Entering other areas in the city, as well as exiting them, is not allowed.
- The entire execution model is synchronous and allows the existence of a cycle virtual clock which, in turn, allows assigning specific times for the operations, simulating the existence of a heterogeneous architecture.
- The environment allows executing the program in a traditional manner or with step-by-step instructions, giving the user detailed control over program execution to allow them controlling typical concurrency situations such as conflicts (collisions) or deadlocks.
- In the step-by-step mode, the effect of the operations can be reflected on physical robots, communicated through Wi-Fi.
- In the environment, each robot has an associated status that shows the contents of its bag (number of flowers and papers in the model), the corner it is occupying, and its current state: if it is executing an operation, waiting for a message, or waiting for a corner to be freed.

3.1 Area Statements

The area statement process starts with the keyword **areas** and ends when the keyword **robots** appears. An area in the city is a rectangular subset of city corners through which robots can move. They can be classified into three types:

- Shared area (areaC): this is the default type of area; it corresponds to any region in the city that can be freely accessed, i.e., any robot can move within it.
- Exclusive area (areaP): this type of region allows the presence of only one robot in it. Any robot attempting to enter an exclusive area

corresponding to another robot will generate a run time error. It should be noted that exclusive areas allow an implicit mutual exclusion mechanism between robots.

- Partially shared area (areaPC): this type of regions allows access to one or several robots, with the constraint that they must have been previously authorized. It should be noted that partially shared areas allow a selective mutual exclusion mechanism between robots.

Each area statement starts with a name, followed by a colon and the keyword areaC, areaP or areaPC (to indicate its type) plus four parameters. These represent the bottom left and upper right coordinates that the area will have within the city. Each type of area is associated to a color, as shown in Figure 3.

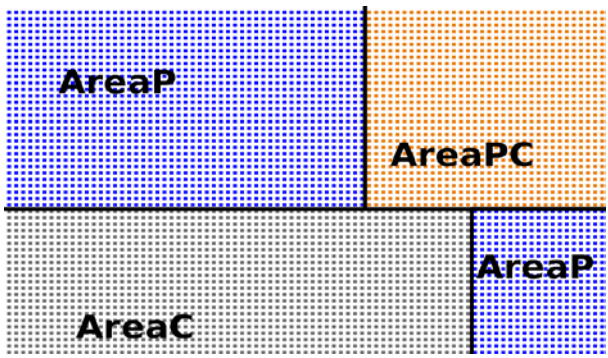


Fig. 3. *Representation of the city with the definitions of the different areas.*

The scope and visibility of the areas are applicable to the entire program, and they must be assigned to the robots that will be allowed to access each area before they start executing their tasks.

3.2 Robot Statements

The robot statement process starts with the keyword **robots** and ends when the keyword **comenzar** appears.

Robots have a structure that is almost identical to that of the main program or subprocesses, including header, statements and body.

The header starts with the keyword **robot** followed by a name. Local process and variable statements follow the same rules as the statements from the main program, with the difference that no new areas or robots can be declared.

Thus, robot creation is always explicit. A feature to be taken into account is that, even if the environment will be used to simulate a Visual Da Vinci type of environment, the single robot will have to be created from the main program.

The body of a robot is a sequence of sentences delimited by the keywords **comenzar** and **fin.** These sentences correspond to the same ones in Visual Da Vinci, and they are used to define robot behavior in the city.

The possibility of defining types of robots allows reusing the code for different robots that have the same behavior, taking into account that care must be taken mainly in relation to the use of absolute locations, since robots may share city areas or not.

3.3 Main Program Body

From the main program, robots have to be assigned to their areas and then they have to be "started" by using the directive *Iniciar*, which requires the name of the robot and its initial location.
This sentence requires compilation time verifications so that two or more robots do not attempt to occupy the same corner, taking also into account the type of area to which the corner in question belongs. For example, a robot cannot occupy a corner that belongs to an area that belongs exclusively to other robot.
A new subset of sentences, called concurrency sentences, is added.

3.4 Handling of Collisions

Conceptually, the "shared resources" found in this model can be reduced to accessing a corner on which other objects may be present.
To avoid "collisions" on the corners is the basic synchronization problem in CMRE.
To handle it, the language has directives that allow blocking and releasing resources:

- *bloquearEsquina* (BE): indicates that the robot is asking for exclusion in order to occupy a corner (which also allows it to pick up or put down objects).
- *liberarEsquina*(LE): this indicates that the robot frees the resource (the corner that it was occupying).

3.5 Communication / Synchronization

As already mentioned, there are multiple robots working on the city. In many cases, they will have to collaborate to solve a problem.
This requires communication and synchronization. An explicit asynchronous message passing mechanism with two directives is used:

- *enviarMensaje* (EM): this allows a robot to send a message to another robot (identified by name). After sending the message, following the asynchronous model, the robot continues with the next sequential instruction without waiting for the message to be received.
- *recibirMensaje* (RM): this indicates that a robot will wait until it synchronizes with the message sent by another robot. Upon reception, the name of the robot from which a message is expected is indicated.

Finally, two problems that are widely used for teaching the concepts of concurrent and parallel programming were selected.
Figure 4 shows the code corresponding to a "master/worker" problem that uses message passing. To solve the problem, 4 exclusive areas and 4 robots (1, 2 & 3 workers, 4 master) are declared, where each of them operates within its exclusive area, picking up all flowers, and, when they complete this task, robots 1, 2 and 3 send their results to robot 4 for it to add them up and report the grand total.

```
programa example1

procesos
  proceso avenida(ES f:numero)
  comenzar
    repetir 49
      {runs through the avenue, picking up
        flowers and adding them to
        parameter f}
  fin

areas
  {area definition}
  area1: areaP(1, 1, 50, 50)
  area2: areaP(1, 51, 50, 100)
  area3: areaP(51, 1, 100, 50)
  area4: areaP(51, 51, 100, 100)

robots
{behavior for each type of robot}
  robot worker
  variables
    f:numero
  comenzar
    f:=0
    repetir 49
      avenida(f)
      Pos(PosAv+1,PosCa-49)
    avenida(f)
    enviarMensaje(f, robot4)
  fin
```

```
robot master
  variables
    f:numero
    total:numero
  comenzar
    f:=0
    repetir 49
      avenida(f)
      Pos(PosAv+1,PosCa-49)
    avenida(f)
    total:=f
    recibirMensaje(f, robot1)
    total:=total+f
    recibirMensaje(f, robot2)
    total:=total+f
    recibirMensaje(f, robot3)
    total:=total+f
    informar(total)
  fin

variables
  {creation of robot variables}
  robot1: worker
  robot2: worker
  robot3: worker
  robot4: master

comenzar
  {Assignment of areas to each robot}
  AsignarArea(robot1, area1)
  AsignarArea(robot2, area2)
  AsignarArea(robot3, area3)
  AsignarArea(robot4, area4)
  iniciar(robot1, 1, 1)
  iniciar(robot2, 1, 51)
  iniciar(robot1, 51, 1)
  iniciar(robot2, 51, 51)
fin
```

Fig. 4. *Example of program using message passing.*

Figure 5 shows the code corresponding to a problem that uses shared memory. In this case, the entire city is declared as shared by 2 robots (1 and

2), and they must coordinate to remove the flowers from corner (1,1) until it is empty. This coordination is required to ensure that the robots are not at the same time in the same corner and, therefore, picking up the same flower. Every time a process picks up a flower, it moves it to the next corner (different for each robot).

```
programa example2
procesos
  proceso girar(E cant:numero)
  comenzar
    repetir cant
      derecha
  fin
  proceso depositarUnaFlor
  comenzar
    mover
    liberarEsquina(1,1)
    depositarFlor
  fin

areas

areas
  {area definition}
  area1: areaC(100, 100, 100, 100)

robots
  {comportamiento de c/tipo de robot}
  robot tipo1
  variables
    seguir:boolean
  comenzar
   seguir:=V
    girar(2)
    bloquearEsquina(1,1)
    mover
    si ~(HayFlorEnLaEsquina)
       seguir:=F
    mientras(seguir)
      tomarFlor
      girar(2)
      depositarUnaFlor
      girar(2)
      bloquearEsquina(1,1)
      mover
      si ~(HayFlorEnLaEsquina)
         seguir:=F
    girar(2)
    mover
    liberarEsquina(1,1)
  fin
```

```
robot tipo2
  variables
    seguir:boolean
  comenzar
    seguir:=V
    girar(3)
    bloquearEsquina(1,1)
    mover
    si ~(HayFlorEnLaEsquina)
      seguir:=F
    mientras(seguir)
      tomarFlor
      girar(2)
      depositarUnaFlor
      girar(2)
      bloquearEsquina(1,1)
      mover
      si ~(HayFlorEnLaEsquina)
        seguir:=F
    girar(2)
    mover
    liberarEsquina(1,1)
  fin

variables
  {creation of robot variables}
  robot1: tipo1
  robot2: tipo2

begin
  {Assignment of areas to each robot}
  AsignarArea(robot1, area1)
  AsignarArea(robot2, area1)
  iniciar(robot1, 1, 2)
  iniciar(robot2, 2, 1)
end
```

Fig. 5. *Example of shared memory program.*

3.6 Current Development Progress

CMRE is fully developed in Java and is being used experimentally at the UNLP. The physical robots are in the process of being purchased, although they do not represent an additional complexity (in the current state of development). The features chosen for the physical robots are related to adding new possibilities to the environment, as mentioned in the future lines of work.

4. Conclusions and Future Lines of Work

An environment has been presented for the early teaching of the concepts of concurrency and parallelism using virtual and physical robots in an interactive and flexible programming environment.
We are currently studying the generalized use of CMRE in applications for which the robots acquire information in real time and the algorithms defined in the environment make decisions dynamically. This is particularly relevant for subjects that deal with Real Time Systems and even Intelligent Systems.

5. References

1. [ACM04] ACM/IEEE-CS Joint Task Force on Computing Curricula. "Computer Engineering 2004: Curriculum Guidelines for Undergraduate Degree Programs in Computer Engineering". Report in the Computing Curricula Series (2004).
2. [ACM08] ACM/IEEE-CS Joint Interim Review Task Force. "Computer Science Curriculum 2008: An Interim Revision of CS 2001". Report from the Interim Review Task Force (2008).
3. [ACM13] ACM/IEEE-CS Joint Task Force on Computing Curricula. "Computer Science Curricula 2013". Report from the Task Force (2013).
4. [ACM68] ACM Curriculum Committee on Computer Science. "Curriculum '68: Recommendations for the undergraduate program in computer science". Communications of the ACM, 11(3):151-197 (1968).
5. [ACM78] ACM Curriculum Committee on Computer Science. "Curriculum '78: Recommendations for the undergraduate program in computer science". Communications of the ACM, 22(3):147-166 (1979).
6. [ACM99] ACM Two-Year College Education Committee. "Guidelines for associate-degree and certificate programs to support computing in a networked environment". New York: The Association for Computing Machinery (1999).
7. [AMD09] AMD. "Evolución de la tecnología de múltiple núcleo". http://multicore.amd.com/es-ES/AMD-Multi-Core/resources/Technology-Evolution (2009).

8. [AND00] Andrews G. (2000). “Foundations of Multithreaded, Parallel, and Distributed Programming”. Addison Wrsley.
9. [CAR03] Carr S., Mayo J., Shene C. (2003). “Threadmentor: a pedagogical tool for multithreaded programming”. ACM Journal of Educational Resources, 3:1–30.
10. [DAS89] Dasgupta S. (1989). “Computer Architecture. A Moder Synthesis. Volume 2: Advanced Topics”. Jhon Wilet & Sons.
11. [DEG12a] De Giusti A. E., Frati F. E., Leibovich F., Sánchez M., De Giusti L. C., Madoz M. C. (2012). “Concurrencia y Paralelismo en CS1: la utilización de un Lenguaje Visual orientado”. Proceeding del VII Congreso de Tecnología en Educación y Educación en Tecnología.
12. [DEG12b] De Giusti L. C., Frati F. E., Leibovich F., Sánchez M., Madoz M. C. (2012). “LMRE: Un entorno multiprocesador para la enseñanza de conceptos de concurrencia en un curso CS1”. Revista Iberoamericana de Tecnología en Educación y Educación en Tecnología. pp. 7 - 15.
13. [DEG12c] De Giusti A. E., Frati F. E., Sánchez M., De Giusti L. C. (2012). “LIDI Multi Robot Environment: Support software for concurrency learning in CS1”. Proceeding of IEEE International Conference on Collaboration Technologies and Systems. pp. 294-298.
14. [DEG13] De Giusti A. E. (2013). “El cambio tecnológico como motor de la Investigación en Informática”. Conferencia inaugural del Workshop de Investigadores en Ciencia de la Computación (WICC2013).
15. [DIJ65] Dijkstra E. W. (1965). “Solution of a problem in concurrent programming control”. Communications of the ACM, 8(9):569.
16. [DIJ78] Dijkstra E. W. (1978). “Finding the Correctness Proof of a Concurrent Program”. In Program Construction, International Summer Schoo, Friedrich L. Bauer and Manfred Broy (Eds.). Springer-Verlag, 24-34.
17. [GEP06] Gepner P., Kowalik M.F. (2006). “Multi-Core Processors: New Way to Achieve High System Performance”. In: Proceeding of International Symposium on Parallel Computing in Electrical Engineering 2006 (PAR ELEC 2006). pp. 9-13.
18. [GOL72] Goldstine H. H. (1972). “The Computer”. Princeton University Press,
19. [GPG] GPGPU. “General-Purpose Computation on Graphics Processing Units”. http://gpgpu.org.
20. [HAN77] Hansen P. B. (1977). “The Architecture of Concurrent Processes”. Prentice Hall.
21. [HOA78] Hoare C. (1978). “Communicating Sequential Processes”. Communications of the ACM, 21(8): 666-677.
22. [HOA85] Hoare C. (1985). “Communicating Sequential Processes”. Prentice Hall.
23. [HOO13] Hoonlor A., Szymanski B. K., Zaki M. J., Thompson J. (2013). “An Evolution of Computer Science Research”. Communications of the ACM.
24. [HWA84] Hwang K., Briggs F. A. (1984). “Computer Architecture and Parallel Processing”. McGraw Hill.

25. [HWA93] Hwang K. (1993). “Advanced Computer Architecture: Parallelism, Scalability, Programmability”. McGraw Hill.
26. [LEGO] "Lego Education". http://www.legoeducation.us/eng/characteristics/ProductLine~LEGO%20MINDSTORMS%20Education%20EV3.
27. [LEGOb]"LEGO Mindstorms EV3 Announced". http://brickextra.com/2013/01/10/lego-mindstorms-ev3-announced/
28. [MCC08] McCool M. (2008). “Scalable Programming Models for Massively Parallel Multicores”. Proceedings of the IEEE, 96(5): 816–831.
29. [OLS83] Olsen E. W., Whitehill S. B. (1983). “Ada for Programmers”. Prentice Hall.

III

ETHICOMP LatinAmerica

Complexity is Free, but at What Cost?

A Survey of the Current Uses of 3D Printers and the Ethical Concerns that Will Arise from Their Continued Use

KELLY GREMBAN

Department of Computing Sciences
Villanova University
Villanova, Pennsylvania 19085, U. S. A.
kgremb01@villanova.edu

Abstract. *As 3D printing becomes more widespread, ethical decisions must be made in regards to how the technology should be used. I discuss various ways 3D printers are currently being used, and how they may be used in the future. Three ethical concerns are addressed: 1) intellectual property rights, 2) the printing of plastic firearms, and 3) the printing of living body parts.*

Keywords: *3D printing, additive manufacturing, bioprinting, printable guns*

1. Introduction

3D printers have been around for nearly three decades, but they are mostly used for commercial manufacturing and were made available to consumers only in recent years. As the technology becomes more versatile and affordable, it is increasingly apparent that 3D printing will be the next invention to revolutionize societies and economies worldwide. 3D printers can create everything from customizable prosthetic limbs that fit better than generic models for a fraction of the cost, to lifelike action figures, to replacement parts for out of production items. Theorists predict that they could "revamp the economics of manufacturing and revive … industry as creativity and ingenuity replace labor costs as the main concern around a variety of goods" [1]. But with technology that promises more uses than can even be comprehended at this point, there are a lot of questions to be answered, such as whether 3D printers will positively or negatively affect society, and what limitations will or should be placed on their use.

This paper will first examine the promises of 3D printing technology to revolutionize the manufacturing industry and economy, and then address two ethical concerns that will come up as this technology advances: intellectual property infringement and the use of 3D printers to create ethically debatable items, such as body parts and firearms.

2. A Brief Background

3D printing is revolutionary because it combines computer-generated ideas with effective and easy manufacturing to create products previously thought impossible. To create with a 3D printer, the user starts with a Computer Aided Design (CAD) which is a digital model of the object. These can be made by creating the design through a CAD program or by using a 3D scanner to create a model of a real-life object [2]. The CAD software then slices the model into minute cross-sections that are fractions of a millimeter thick. The printer takes these cross-sections and applies them through a process called Additive Manufacturing (AM) in which each layer of material is deposited and fused with the layer below it [3]. Printers currently on the market are mostly restricted to printing with plastics, although larger industrial printers can work with metals. According to Hod Lipson of Cornell University, "any material you can squeeze, melt or generate into a powder, you can print" [4]. There are numerous unique variations on the typical plastic or metal printers:

- The "candyfab" uses granulated sugar to print candy [5].
- The "Burritob0t" can print customized burritos in less than five minutes [6].
- The "D-Shape" prints sandstone to create houses [7].
- Researchers have printers that use living cells to print "cartilage, meniscus of the knee … spinal disks and heart valves" [4].

One benefit to 3D printing is that it is more eco-friendly than traditional methods of manufacturing. AM is revolutionarily efficient, both in terms of environmental impact and production cost. First, AM requires as little as one-tenth the amount of material as conventional approaches. Whereas traditional manufacturing must remove excess, AM builds up materials until it forms a whole [8]. Second, taking the manufacturing out of the factory also means that objects can be created anywhere, thereby cutting down on shipping requirements. Third, producing only when required removes the need for an economic system based in mass production that leads to thousands of surplus products being wasted [4]. 3D printing is beneficial because it allows for manufacturing physical objects on-site with minimal waste.

The second benefit of 3D printing is cost-efficiency for individual businesses because it streamlines the production process. Wohlers Associates, a consulting company that pays special attention to 3D printers, estimates that businesses using these devices can reduce costs by 50% and time requirements by nearly 70% [9]. The driving factor behind the cost reduction is that "complexity is free" [4]. It used to be that fabricating businesses spent most of their time creating and re-creating prototypes, and the more complex an object the more time, personnel, and money it required. With 3D printing, the major expense for companies is now just the amount of material needed to build the object [10]. 3D printing also cuts out assembly lines because a 3D printer can print moving parts at the same time, already assembled [2]. Daniel O'Conners demonstrated this by printing "a spinning gyroscopic

thingumabob complete with moving ball bearings" in one session which moved freely after being removed from the machine [11].
The ability to handle complexity leads to the third benefit: innovation. With 3D printers, manufacturers and even at-home amateurs can create structures that would be impossible with any other approach. For example, a 3D printer can create a complete bike chain, printed with the links already connected. And with reduced barriers to participate in 3D manufacturing anyone with access to a printer can contribute [8]. This change has begun to bring about the democratization of manufacturing, which, as it continues, will "allow local entrepreneurs to solve all kinds of problems, both big and small" [12]. People will not have to rely on off-the-shelf products but will be able to customize existing items or create entirely new products to find more efficient solutions. The most important function of 3D printers is the fact that they allow an entirely fresh generation of ideas to come into being. In Lipson's words, "it's not about how you duplicate things that you make today with other techniques, but it's how you explore, as we said, the new frontiers of design, making things you can't imagine today" [4]. 3D printers are not simply going to change how products are made, but will widen the definition of what it is possible to make.
As with the arrival of any revolutionary technology, the changes that come about may be difficult to embrace at first. 3D printers will change the way we think about modern manufacturing practices, and as a result could render many of them obsolete. Businesses that rely on the current way of doing things – like assembly lines and mass production – may have a hard time keeping up, but eventually this revolution will bring about new opportunities. "As businesses, industries, and jobs go away, new ones appear, and historically the new ones more than make up for the old ones that have vanished" [5]. One possible outcome is the strengthening of small businesses. Currently, it is difficult for locally-owned shops to compete with mega-store corporations. Small businesses cannot stock the same variety of products or rely on a national or global infrastructure to get cheaply produced goods. But with 3D printing, creativity will quickly surpass mass productivity in economic importance. In terms of the effects 3D printers will have on businesses and the economy, the outlook is positive.

3. Intellectual Property Concerns

Because 3D printers are so efficient at production and reproduction, there are several ethical concerns that must be addressed in the upcoming years. The first is that of intellectual property. Like the printing press, photocopier, VCR, and DVR before it, the 3D printer will be the center of a debate between individuals protecting fair use and open sources, and companies protecting copyright and patents.
The 3D printer of today is comparable to the computer in its formative years: this technology has the potential to revolutionize the creation and distribution of physical objects just as computers revolutionized the creation and

communication of ideas. However, the same pitfalls that the computer industry went through have the potential to affect the 3D printing industry before it really gets started. Michael Weinberg, an attorney for Public Knowledge, refers to laws like the Digital Millennium Copyright Act that restricted the rights of the general public on the internet before the general public even knew they had those rights. He says that unless people actively learn about and defend their rights to fair use and open source materials in regards to 3D printing, they may lose them as corporations and industries that feel threatened by innovative technology try to protect themselves by restricting usage. [2]

Just as with the computer industry, the rise of the 3D printer will most likely expand the manufacturing industry but there will be strife before this can happen because it goes against most of the prevalent business models. Entrepreneurs and hobbyists looking to make use of 3D printers will have to compete with established industries protecting their interests. Patent holders will try to put restrictions on CAD files to prevent users from either scanning and reproducing copyrighted products, or creating products that infringe upon established patents. Currently, there are multiple sites where users can freely share CAD files in peer-to-peer communities. If the files become legally restricted then these open source communities may be destroyed by those who assume that any CADs shared are pirated, in the same way that Napster and other peer-to-peer sites were taken down.

In his essay, "It Will Be Awesome If They Don't Screw It Up", Weinberg advises 3D manufacturers on how to practice their rights without infringing on copyrights, patents, or trademarks. The best way to keep 3D printing technology from being restricted is by knowing how to use it without violating intellectual property rights in the first place. However, it is still going to be difficult to maintain the right to freely create and share in the face of large industries that feel threatened. The fact that there is no way to prove the benefits 3D printing will have does not make this problem any easier, because "policymakers and judges will be asked to weigh current concrete losses against future benefits that will be hard to quantify and imagine" [2]. It is likely that this case will go the way of its predecessors, photocopiers and VCRs, and be settled in favor of the new technology, but given the counter-examples of the computer industry's heightened restrictions it would be prudent to be vigilant about the public's rights to fair use of products and open source sharing of creative material.

4. Issues Arising from Printed Weapons?

While the right to creation via 3D printing should be preserved, there are some scenarios in which advanced home manufacturing could cause a real danger to the public. One benefit to the current system of centralized manufacturing is that it can be regulated. Dangerous objects like firearms are supposed to be made and distributed only by certain people and only in accordance with specific guidelines. Decentralized 3D manufacturing can

avoid these regulations entirely by allowing individuals to print their own weapons, or at least enough of the component parts to avoid regulation.

In the United States, there are currently several layers of law enforcement surrounding the creation, distribution, and purchase of firearms at both the state and federal level. Specifically, "anyone 'engaged in the business' of manufacturing, importing or dealing in firearms is required to become a federal firearm licensee" and when any gun is sold, the distributor must run a background check on the buyer and record the serial number of the gun which must be included by the manufacturer. However, once you go beyond that the regulations become more complicated. For example, since the component parts of guns can be sold separately, the piece that is legally considered the "firearm" is the central frame, also known as the lower receiver, because it allows for the combination of the other pieces. Additionally, there are restrictions about how a gun may be made or what materials must be used. For example: "the Undetectable Firearm Act of 1988 requires that all major gun components generate accurate depictions in x-ray machines and also requires assembled firearms to trigger metal detectors". In this way, the distribution of firearms is restricted by limiting who can buy or sell guns as well as by specifying how gun parts must be made and ways to track them. [3]

The current system of regulating gun access and use is not perfect, but 3D printing is poised to upset any efficacy of the regulations. This is in part because of two current trends: 3D printing is becoming more advanced and widely available, and the firearm industry is beginning to use more polymer materials in weapons design, specifically in the design of the frame – the one regulated component. While there are still metal components that would have to be purchased from an arms manufacturer, the frame could be printed at home without having to adhere to any regulations. Additionally, 3D printers soon will have the capability of printing the highly-regulated parts that can alter a semi-automatic rifle into a fully automatic one. These abilities to make alterations at home circumvent laws restricting the use of highly dangerous weapons by the public. [3]

Americans have never been explicitly prohibited from creating their own firearms, but historically being able to make these weapons required the dedication to learn metalworking first. Now 3D printers are making it so that "a person with little to no understanding of firearms will nonetheless be capable of wielding a weapon in [a] short matter of time" [13]. This year, the Texas-based group Defense Distributed, headed by Cody Wilson, successfully fired their 3D printed handgun, "The Liberator," and put the CAD files online for others who have 3D printers to use. The gun is entirely plastic except for a firing pin and the ammunition – the plans do include a piece of steel that would set off metal detectors, but it is an enhancement that can be omitted without affecting the functionality [14].

This 3D printing innovation has set authorities scrambling to counteract the effect of do-it-yourself, undetectable firearms. New York Congressman Steve Israel called for the renewal of the Undetectable Firearms Act after hearing the news, while New York Senator Charles Schumer suggested banning 3D-printed guns entirely. The Australian police force released a statement

warning people that using the Liberator would put their personal safety at risk, explaining that they had tested it and the gun exploded on the second round. Police Commissioner Andrew Scipione attributes the "catastrophic failure" to a lack of standards for homemade weapons that endanger the gun owners as much as their targets [15]. The general political tone seems to be leaning towards restriction, but in America at least, forbidding the personal production of weapons may be constitutionally impossible.

Although there is a legitimate threat to the public involved with this system of printing, any legal action in America restricting access to or use of 3D printers may go against the Constitutional right to bear arms. In fact, there is an argument that allowing 3D printed guns would actually enrich Second Amendment protections. Currently, the right to bear arms does not apply to those who are handicapped and cannot use generically-produced weapons to defend themselves. With infinitely customizable design options, 3D printers could extend this right by creating unique guns that compensate for the user's limited abilities [3]. Additionally, the recent Supreme Court case of the District of Columbus v. Heller upheld firearm rights, and explained that the continued right to be able to resist tyranny is one of the key reasons for upholding the Second Amendment even in the modern age. It is arguable that the "ability to make one's own weapons, spare parts and ammunition would be essential to sustain protracted resistance against tyranny or to obtain meaningful protection in times of anarchy" [3]. If this issue goes to court in the United States, it is reasonable to expect that this argument will be made and supported by those who view their right to bear arms as inalienable.

As with all technology, there are ways to use it for dangerous purposes, but that must be weighed against the improvements and greater rights that it provides as well. That being said, protecting lives should be held above protecting rights. Until firearm regulations and police enforcement are prepared to handle the possibility of homemade weaponry, these uses for 3D printers should be pursued carefully.

5. Printing the Biological World

Printed weapons are a concern that is being addressed currently, but there are benefits to looking ahead and giving consideration to applications of 3D printing technology that are not yet affecting mainstream culture. Researchers in the medical field are using printers in ways that will revolutionize health and wellness. So far, results are still experimental, but intentions and predictions for where this technology will go next range from improving the quality of life to altering the construction of the human body.

The 3D printing of biological material, or bioprinting, uses live cells and specially designed cultures as the "ink" in their printers. This field of study shares many of the same principles as AM, but has several differences and difficulties that come about from using living material. The first is that the cells settle and readjust after being printed. For this reason, a CAD made from a scanned organ cannot be printed as-is; "the organ blueprint must be

larger and probably have a slightly different shape" due to "postprinting remodeling associated with tissue fusion, tissue compaction and tissue maturation processes" [16]. The second major difference is having to prevent damage from happening to the cells during and after the printing process. Vladimir Mironov, the director of the Advanced Tissue Biofabrication Center at the Medical University of South Carolina, explains, "[f]rom an engineering point of view, high temperature and toxicity (typical for rapid prototyping technologies and processes) are not acceptable for the bioprinting process" [17]. Every step of the process of printing puts strain on the cells, from being stored in cartridges, to being ejected, to surviving in lab conditions afterwards.

Despite these complications, scientists have already had success with their bioprinting experiments:

- Laurence Bonasar of Cornell University used a modified Fab@Home printer to print cartilage directly onto a bone [17].
- A team of researchers, also using a Fab@Home 3D printer, used cartilage from calves and silver wire to print a pair of functioning bionic ears which continued to perform for more than ten weeks [18].
- The University of Bordeaux was the first to work on printing bone tissue [17].
- A group from the Wake Forest Institute for Regenerative Medicine successfully printed skin onto live animals and showed that the procedure cut the healing time of wounds by more than half [17].
- The research company Organovo printed a functioning, miniature human liver using a proprietary 3D printer, NovoGen [19].

Much of this bioprinting is focused on one of two goals: printing entire organs for transplant or printing functional tissue for medical research. Achieving the goal of printed organs will be very difficult, but steps are already being made. For example, Mironov and his co-authors state that the most challenging step is managing to print the system of arteries necessary for maintaining cell life [16], but Mironov himself goes on to claim in a later paper that several universities, including his own, have data to show that this is feasible [17]. Eventually scientists want to reach the point where they can collect a patient's cells and print a new organ directly into the body, which would have numerous benefits. Most notably, it would "once and forever eliminate patient waiting lists for organ transplantation," thus saving countless lives which would otherwise be lost simply due to lack of resources [16]. Additionally, being able to collect and print with the patient's own cells would eliminate the dangers of the body rejecting the new organ or developing tumors [17]. This achievement will allow for a higher quality of life for a greater number of people, without requiring sacrifice or endangering the patient unnecessarily.

Bioprinting tissue for medical research is likely to be happening sooner than made-to-order organs. It may not be as accurate as in-depth clinical trials with real patients, but it is expected to be "more predictable than small or

even large animal testing" and simultaneously "reduce the costs of drug development and improve drug safety" [16, 17]. Overall, this will be a benefit to the medical community. Researchers will be able to have similarly if not more useful information from testing with human tissue, all without the ethical conundrum of weighing benefits against testing on sentient animals.

Since bioprinting is still in its formative years, there is a great deal of thought being put towards how it will be used in the future, and how it may even have an influential role in the shaping of the future. Mironov speculates that being able to make body parts to order with your own cells will lead to two outcomes: on one hand it can extend the length of human life as each part that wears out is replaced, and on the other hand it may create a culture of what he terms "body fashion" as people with the means to do so design and print custom body enhancements for themselves [17]. This one example demonstrates how a single piece of technology can have such far-reaching effects as to influence both the quality of life and changes in culture.

Along with expanding the length of our lives, bioprinting and 3D printing can help to expand the physical capabilities of humans. This invention could be what makes long-term space exploration possible: the ability to travel with a full hospital and manufacturing facility. If that does not sound enough like science fiction, there are some people who are interested in bioprinting for even more futuristic reasons. The group that created the bionic ears out of cartilage has explained that their goal is to develop "a unique way of attaining a seamless integration of electronics with tissues to generate 'off-the-shelf' cyborg organs" [18]. These possibilities are even spreading into the art world. Heather Dewey-Hagbog has created a work called "Stranger Visions" in which she collects discarded DNA from public places, analyzes the samples, and then uses the genetic information to 3D print a face [20]. While these are not totally accurate resemblances – no one has recognized themselves in her work yet, at least – and she only prints in plastic, she believes that this is just a precursor to being able to clone a person from a bit of hair or skin. These predictions may seem like something from a strange tale, but researchers are working every day on turning them into reality.

6. Conclusion

3D printing is the next pivotal step in advancing technology, and will disrupt the established systems as thoroughly as the computer and the invention of the Internet did mere decades ago. 3D printing has the potential to greatly improve our quality of life and expand our ability to practice our inalienable rights. But, at the same time, there are dangers and new methods of misuse that must be anticipated and prevented. Despite the pitfalls that arise with any new piece of technology, 3D printers will benefit the quality of life. While precautions must be made to prevent dangerous and illegal use, they should not include restricting the distribution and creative freedom that will bring about innovative advancement. As enthusiasts start experimenting with strange and exciting new uses "the best improvements will spread fastest, in a

process akin to Darwinian natural selection" [5]. For that reason, it is important to encourage a "diversity of approaches and strong competition among different approaches" in order to ensure superior results going forward [16]. The next several years will be crucial in the formation of 3D printing rights and restrictions, and hopefully lawmakers, industry leaders, and everyday users will work to create the most creatively supportive community possible and allow the new possibilities it opens up to develop.

Acknowledgments

Many thanks to Dr. William Fleischman, who encouraged me to submit this paper and offered support and advice for its development.

References

1. Vance, A. (2013). "3-D Printing Spurs a Manufacturing Revolution." *The New York Times* 13 Sept. 2010: n. pag. Web.
2. Weinberg, M. (2013). *It Will Be Awesome If They Don't Screw It Up*. *Www.publicknowledge.org*. Public Knowledge, Nov. 2010. Web.
3. Jensen-Haxel, P. (2013). "3D Printers, Obsolete Firearm Supply Controls, and the Right to Build Self-Defense Weapons under Heller." *Golden Gate University Law Review* (2012): n. pag. *LexisNexis*. Web.
4. Wohlers, T., Pettis, B., and Lipson, H. (2012). "Can 3D Printers Reshape the World?" Interview by Ira Flatow. *Science Friday*. NPR. 22 June. Radio. Transcript.
5. Easton, T. A. (2013). "The 3D Trainwreck: How 3D Printing Will Shake Up Manufacturing." *Analog Science Fiction & Fact*.Nov. 2008. Web. 20 Jun.
6. Cheshire, T. (2012) "BurritoB0t: the 3D printer that creates Mexican snacks in five minutes." Wired.co.uk. 16 Aug. Web. 18 Jun. 2013.
7. Steadman, I. (2013). "The race to build the first 3D-printed building." Wired.co.uk. 4 Jun. Web. 21 Jun. 2013.
8. "Print Me a Stradivarius." *The Economist* 10 Feb. (2011): n. pag. *The Economist*. Web. 15 Apr. 2013.
9. Quittner, J. (2010). "How 3D Printing Is Saving This Jewelry Design Business." *Crain's New York Business*. N.p., 20 Oct. Web. 15 Apr. 2013.
10. Graham-Rowe, D. (2008). "3-D Printing for the Masses." *MIT Technology Review* (n.d.): n. pag. 31 July. Web. 12 Apr. 2013.
11. O'Connor, D. (2013). "Thingi Thursday: Spinning Gyroscope." Weblog post. *Www.prsnlz.me*. N.p., 13 Jun. Web. 16 Jun. 2013. http://www.prsnlz.me/blogs/daniel-oconnors-blog/thingi-thursday-spinning-gyroscope/.

12. MacDonald, C. (2012). "3D Printing and the Ethics of Value Creation." Web log post. *The Business Ethics Blog*. N.p., 1 Dec. Web. 10 Apr. 2013.
13. O'Neill, K. J. (2012). *Is Technology Outmoding Traditional Firearms Regulation? 3-D Printing, State Security, and the Need for Regulatory Foresight in Gun Policy*. *Social Science Research Network*. N.p., 3 May. Web. 16 Apr. 2013. <http://papers.ssrn.com/sol3/papers.cfm?abstract_id=2186936>.
14. Greenberg, A. (2013). "Meet The 'Liberator': Test-Firing The World's First Fully 3D-Printed Gun." *Forbes*. N.p., 5 May. Web. 28 June 2013.
15. Clark, L. (2013). "Australian Police: Exploding 3D Printed Gun Will Kill You and Your Victim." *Wired.co.uk*. Wired, 24 May. Web. 25 June 2013.
16. Mironov, V., Kasyanov, V., Drake, C. and Markwald, R. R. (2008). "Organ Printing: Promises and Challenges." *Regenerative Medicine* 3.1: 93-103. Web. 25 Jun 2013.
17. Mironov, V. (2011). "The Future of Medicine: Are Custom-Printed Organs on the Horizon?" *The Futurist* 45.1: 21-24. *Proquest*. Web. 25 June 2013.
18. Mannoor, Manu S., Ziwen Jiang, Teena James, Yong Lin Kong, Karen A. Malatesta, Winston O. Soboveio, Naveen Verma, David H. Gracias, and Michael C. McAlpine. (2013). "3D Printed Bionic Ears." *Nano Letters* (2013): 2634-639. *Pubs.acs.org*. 1 May. Web. 27 June 2013.
19. Organovo. (2013). *Organovo Describes First Fully Cellular 3D Bioprinted Liver Tissue*. *Ir.organovo.com*. N.p., 22 Apr. Web. 27 June 2013.
20. Dewey-Hagborg, H. (2013). *Stranger Vision*. N.p., 7 Mar. Web. 24 June 2013. <http://deweyhagborg.com/strangervisions>.

Esta edición de 150 ejemplares
se terminó de imprimir en Estudiocentro,
Bolívar, Buenos Aires, Argentina,
en el mes de octubre de 2014.

Its objectives are:
“Coordinate academic activities related to the improvement of the teachers' training as well as the curricular update and the use of shared resources to assist the development of both the Computer Sciences careers and the Technology careers in Argentina" and "To establish a cooperative framework for the development of Postgraduate activities in Computer Sciences and Technology, in order to optimize the assignation and use of the resources”.

RedUNCI:

This Network was formally created through an Agreement signed in November 1996 by five National Universities (UNSL, UBA, UNLP, UNS y UNCPBA), during the second edition of CACIC.
Actually 56 argentine Universities are active members of this network.

Regular Activities of the RedUNCI

- Arrangement of an Annual Congress on Computer Science (CACIC) since 1995.
- Arrangement of an Annual Workshop for Researchers on Computer Science (WICC) since 1999.
- Meetings for university professors of Computer Science, for Postgraduate Dissertators and for specialists in certain areas, to promote the debate of common interest topics.
- Publication of *the Journal on Computer Science &Technology* by agreement with ISTEC (Iberoamerican Science and Technology Education Consortium).
- Annual Congress on Technology in Education and Education in Technologies (TE&ET) since 2006.
- Publication of the *Iberoamerican Journal of Technology in Education and Education in Technology*, since 2007.

ISBN 978-987-1985-49-4