INTERNATIONAL JOURNAL

of

COMPUTERS, COMMUNICATIONS & CONTROL

With Emphasis on the Integration of Three Technologies

IJCCC

Year: 2010 Volume: V Number: 1 (March)

Agora University Editing House

CCC Publications

www.journal.univagora.ro

Ştefan I. Nitchi
Department of Economic Informatics
Babes Bolyai University, Cluj-Napoca, Romania
St. T. Mihali, Nr. 58-60, 400591, Cluj-Napoca
nitchi@econ.ubbcluj.ro

Shimon Y. Nof
School of Industrial Engineering
Purdue University
Grissom Hall, West Lafayette, IN 47907, U.S.A.
nof@purdue.edu

Stephan Olariu
Department of Computer Science
Old Dominion University
Norfolk, VA 23529-0162, U.S.A.
olariu@cs.odu.edu

Horea Oros
Department of Mathematics and Computer Science
University of Oradea, Romania
St. Universitatii No. 1, 410087, Oradea, Romania
horos@uoradea.ro

Gheorghe Păun
Institute of Mathematics
of the Romanian Academy
Bucharest, PO Box 1-764, 70700, Romania
gpaun@us.es

Mario de J. Pérez Jiménez
Dept. of CS and Artificial Intelligence
University of Seville
Sevilla, Avda. Reina Mercedes s/n, 41012, Spain
marper@us.es

Dana Petcu
Computer Science Department
Western University of Timisoara
V.Parvan 4, 300223 Timisoara, Romania
petcu@info.uvt.ro

Radu Popescu-Zeletin
Fraunhofer Institute for Open
Communication Systems
Technical University Berlin, Germany
rpz@cs.tu-berlin.de

Imre J. Rudas
Institute of Intelligent Engineering Systems
Budapest Tech
Budapest, Bécsi út 96/B, H-1034, Hungary
rudas@bmf.hu

Athanasios D. Styliadis
Alexander Institute of Technology
Agiou Panteleimona 24, 551 33
Thessaloniki, Greece
styl@it.teithe.gr

Gheorghe Tecuci
Learning Agents Center
George Mason University
University Drive 4440, Fairfax VA 22030-4444, U.S.A.
tecuci@gmu.edu

Horia-Nicolai Teodorescu
Faculty of Electronics and Telecommunications
Technical University "Gh. Asachi" Iasi
Iasi, Bd. Carol I 11, 700506, Romania
hteodor@etc.tuiasi.ro

Dan Tufiş
Research Institute for Artificial Intelligence
of the Romanian Academy
Bucharest, "13 Septembrie" 13, 050711, Romania
tufis@racai.ro

Lotfi A. Zadeh
Department of Computer Science and Engineering
University of California
Berkeley, CA 94720-1776, U.S.A.
zadeh@cs.berkeley.edu

Short Description of IJCCC

Title of journal: International Journal of Computers, Communications and Control
Acronym: IJCCC
International Standard Serial Number: ISSN 1841-9836, E-ISSN 1841-9844
Publisher: CCC Publications - Agora University
Starting year of IJCCC: 2006
Founders of IJCCC: I. Dzitac, F.G. Filip and M.J. Manolescu
Editorial Office:
R&D Agora Ltd. / S.C. Cercetare Dezvoltare Agora S.R.L.
Piaţa Tineretului 8, Oradea, jud. Bihor, Romania, Zip Code 410526
Tel./ Fax: +40 359101032
E-mail: ijccc@univagora.ro, rd.agora@univagora.ro, ccc.journal@gmail.com
Website: www.journal.univagora.ro
Number of issues/year: IJCCC has 4 issues/odd year (March, June, September, December) and 5 issues/even year (March, September, June, November, December). Every even year IJCCC will publish a supplementary issue with selected papers from the International Conference on Computers, Communications and Control.
Coverage:

- Beginning with Vol. 1 (2006), Supplementary issue: S, IJCCC is covered by Thomson Reuters - SCI Expanded and is indexed in ISI Web of Science.
- Beginning with Vol. 2 (2007), No.1, IJCCC is covered in EBSCO.
- Beginning with Vol. 3 (2008), No.1, IJCCC, is covered in SCOPUS.

Scope: IJCCC is directed to the international communities of scientific researchers in universities, research units and industry. IJCCC publishes original and recent scientific contributions in the following fields: Computing & Computational Mathematics; Information Technology & Communications; Computer-based Control.
Unique features distinguishing IJCCC: To differentiate from other similar journals, the editorial policy of IJCCC encourages especially the publishing of scientific papers that focus on the convergence of the 3 "C" (Computing, Communication, Control).
Policy: The articles submitted to IJCCC must be original and previously unpublished in other journals. The submissions will be revised independently by at least two reviewers and will be published only after completion of the editorial workflow.

Contents

Editorial – Special Issue on Collaboration Support Systems (CSS)
H.S. Ko, S. Y. Nof **6**

Introducing Collaborative Practices in Small Medium Enterprises
D. Antonelli, P. Chiabert **8**

A Software System Development Life Cycle Model for Improved Stakeholders' Communication and Collaboration
S. Cohen, D. Dori, U. de Haan **20**

Mining Authoritativeness of Collaborative Innovation Partners
J. Engler, A. Kusiak **42**

Coordinating Aerial Robots and Unattended Ground Sensors for Intelligent Surveillance Systems
E. Pignaton de Freitas, T. Heimfarth, R. Schmidt Allgayer, F. Rech Wagner, T. Larsson, C. E. Pereira, A. Morado Ferreira **52**

Reference Architecture for Collaborative Design
C.Y. Huang, T.T. Yang, W.L Chen, S. Y. Nof **71**

Design of Protocols for Task Administration in Collaborative Production Systems
H. S. Ko, S. Y. Nof **91**

Gaze, Posture and Gesture Recognition to Minimize Focus Shifts for Intelligent Operating Rooms in a Collaborative Support System
J. P. Wachs **106**

Swarming Models for Facilitating Collaborative Decisions
C.B. Zamfirescu, F. G. Filip **125**

Author index **138**

Int. J. of Computers, Communications & Control, ISSN 1841-9836, E-ISSN 1841-9844
Vol. V (2010), No. 1, pp. 6-7

Editorial – Special Issue on Collaboration Support Systems (CSS)

H.S. Ko, S. Y. Nof

Recent and emerging advances in computer and information science and technology have realized a powerful computing and communication environment. It enables effective interactions and collaboration among groups of people and systems (and systems-of-systems) beyond traditional restrictions of time and space. The evolution in hardware (e.g., pervasive computing devices, wireless sensor networks, nano-electronics) and software (e.g., multi-agent systems, workflow and information integration, interaction models and protocols) technology, and their flexible teaming have further enabled diverse forms of collaboration approaches. It has been observed during the last few decades that numerous collaboration methodologies, tools and applications in various domains have emerged to provide better quality services, helping to solve domain-specific, highly complex problems. The development of collaboration tools and methodologies has increased the domain knowledge that can be discovered and shared by individuals, and the level and intensity of interactions and collaboration that can dramatically decrease problem complexity and increase solution quality. At the same time, inefficient interactions, task and information overloads, and ineffective collaboration are prevalent.

In spite of the considerable progress in collaboration tools and methods, a sound foundation of collaboration science is yet to be established; lacking such foundation is a major obstacle in leveling up sophistication and anticipated benefits of collaboration methods. Recently, however, Collaboration Control Theory (CCT) models and techniques have been proposed as the foundation of designing Collaboration Support Systems (CSS), and new features of CSS are being investigated in various areas based on CCT. In order to provide readers with a significant opportunity of investigating novel and forthcoming problems in collaboration research, the guest editors have invited authors from various disciplines and focused on collaborative design and modeling features in the respective areas. After careful and rigorous review and revision processes, eight articles have been selected for their special quality and relevance to CSS. The selected articles are organized along two main topics in CSS: 1) CSS Models and Theories; and 2) CSS Methods and Applications.

In the area of *CSS Models and Theories*:

- **"Swarming models for facilitating collaborative decisions"** – Zamfirescu and Filip introduce the use of swarming models (stigmergic mechanisms) to build collaborative support systems for complex cognitive tasks, exemplifying them through an experiment for group decision processes (GDP) in e-meetings.

- **"Design of Protocols for Task Administration in Collaborative Production Systems"** – Ko and Nof investigate the design of task administration protocols for collaboration support in production system, where those protocols, as control mechanisms, can manage complicated events in the collaborative task workflow environment and overcome limitations of coordination protocols.

- **"Mining authoritativeness of collaborative innovation partners"** – Engler and Kusiak, present a novel approach to automatically determine the authoritativeness of entities for collaboration and demonstrate the use of mining schema for identifying collaboration partners over the Internet.

- **"Reference architecture for collaborative design"** – Huang, Yang, Chen, and Nof present reference architecture for collaborative design (CD) as a framework for analyzing and supporting CD, then describe and illustrate dimensions forming the architecture as a cube of design aspect, design stage, and collaboration scope.

The next group of articles focuses on *CSS Methods and Applications*:

- **"Coordinating aerial robots and unattended ground sensors for intelligent surveillance systems"** – de Freitas, Heimfarth, Allgayer, Wagner, Larsson, Pereira, and Ferreira present a system solution to enable interoperability and coordination support for heterogeneous sensor networks composed of low-end ground sensor nodes and mobile sensors carried by autonomous aerial robots.

- **"Introducing collaborative practices in small medium enterprises"** – Antonelli and Chiabert propose a methodology to evaluate the possibility of using PLM as a framework which exploits collaboration links within an enterprise, based on an exhaustive analysis of the PLM impact on different aspects of the enterprise.

- **"A software system development life cycle model for improved stakeholders' communication and collaboration"** – Cohen, Dori, and De Haan describe a collaborative software system development life cycle model using Object Process Methodology (OPM), which includes various stakeholders and variables, and considers multiple aspects in collaboratively developing off-the-shelf software.

- **"Gaze, posture and gesture recognition to minimize focus shifts for intelligent operating rooms in a collaborative support system"** – Wachs describes the design of an intelligent, collaborative system framework, which involves the integration of machine vision, voice recognition and computer graphics-projection techniques to improve operation rooms for surgery by highly intuitive, natural and multimodal interactions.

These articles are presented based on related scientific investigations around the world and reflect well on the broad and challenging area of CSS. The guest editors wish to thank all the contributing authors, the referees, and the editorial office colleagues who have all endeavored to bring this special issue on CSS to you.

H.S. Ko and S.Y. Nof
West Lafayette, Indiana, USA
Special Issue Guest Editors

Int. J. of Computers, Communications & Control, ISSN 1841-9836, E-ISSN 1841-9844
Vol. V (2010), No. 1, pp. 8-19

Introducing Collaborative Practices in Small Medium Enterprises

D. Antonelli, P. Chiabert

Dario Antonelli, Paolo Chiabert
Politecnico di Torino
Department of Production Systems and Economics
Corso Duca degli Abruzzi 24, 10129 - Torino, Italy
E-mail: {dario.antonelli,paolo.chiabert}@polito.it

Abstract: In an enterprise, collaborative working schemes are obtained only by adopting a suitable organization of the enterprise functions. At the informative level collaboration can be enabled by using suitable project management tools for the exchange of information that is the basis of collaboration. One of these tools is *Product Lifecycle Management* (PLM). On the basis of an exhaustive analysis of the PLM impact on the different aspects of the enterprise, the paper proposes an original methodology to evaluate the possibility of using PLM as framework in which exploit collaboration links within the enterprise. The methodology is not founded on the formal declared organization but on the actual data flows that are induced by the relations among the documents used to develop a project. Data are collected on the field using questionnaires. The links are made explicit by applying the hierarchical clustering with single linkage agglomerative technique. The attitude to the implementation of PLM is then analysed with respect to the organization and to the informative system. Such methodology is general and can be profitably applied to assist the PLM implementation in the enterprises.

Keywords: Product Life-Cycle Management, Concurrent Engineering, Business Process Management, Cluster analysis, Collaboration.

1 Introduction

Small and Medium Enterprises (SMEs) are companies with a turnover of less than 50MEuro and a number of employees below 250, as for the EU Recommendation 2003/361/EC. In the European countries they are the backbone of the economic system [7]. *Concurrent Engineering* (CE) is an effective strategy to shorten 'time to market' of innovative products. The key for its accomplishment stays in the collaboration among all the enterprise personnel appointed to the project. The goal is accomplished by the parallel development of different activities obtained by the integration of several company functions in a multidisciplinary group in order to anticipate the constraints generated by the industrial dynamics [14]. The core aspect of CE is therefore the creation of tight co-operation links among all the industrial actors involved in the definition of product and process. A significant tool to assist the CE effectiveness is PLM. Unfortunately, PLM implementation inside SMEs is scarce. The reason goes beyond implementation difficulties or excessive costs of IT systems. It derives from the poor conformity of the ordinary SME organization of work to the requirements of CE in term of strategy, organization and technology [13].

It is useful to remark that often in literature the term collaboration is given the same significance as the term co-ordination. Collaboration means working with the others for the success of an activity. Coordination can refer to optimisation of the work by finding the best division of activities to be performed by humans and machines. The definition put forth by Malone and Crowston [15] is: "Coordination is managing dependencies between activities". Coordination also, recalling the theory of systems, implies the presence of a coordinator, then some hierarchical shape on the organisation [18]. Instead, collaboration is the process of various individuals, groups or systems working together on a voluntary basis without the need for a manager or a work program [16]. Obviously collaboration works only if some behaviour rules are respected, as the ones proposed in [21] in the case of collaborative software agents.

In the organization oriented to CE, different coordination mechanisms have been adopted [10]. They differ mainly on the instant of application of the coordination: predetermined before starting the work or determined during the work as a result of a negotiation. A discussed list of co-ordination systems is given in [20]: Information workflow, Negotiated objectives, Milestones, Professional rules, Mapped parameters, Design space sharing, Joint parameters, Data coherency. Some coordination systems need the active participation of a coordinator, e.g. the project manager. Conversely other systems like Design space sharing and Data coherency must be enforced by the use of appropriate software tools, e.g. the PLM.

A comprehensive definition of Product Lifecycle Management is "a strategic business approach for the effective management and use of corporate intellectual capital" where "corporate intellectual capital" consists of products definition, products history and best practices" [6]. Several PLM systems already operates in large companies demonstrating their effectiveness. They range from the Ford project C3P (CAD/CAE/CAM and Product Information Management) started in 1996 and based on SDRC's I-DEAS and Metaphase software to the Boeing 7E7 Dreamlinear airplane entirely designed in the Dassault Systemes PLM software platform, to the Siemens Teamcenter Suite largely adopted in automotive industries. Presently new lighter software suites, based on the "out of the shelf" commercial policy, make PLM systems affordable to SMEs. Despite the availability of software, PLM do not have an adequate diffusion among them.

In authors' opinion the most important justification of enterprises' interest in PLM should be the competitive edge provided by collaboration. As a matter of fact, PLM assures the most effective control of enterprise's projects by increasing the efficiency in information management along three directions:

- Increase of efficiency in the management of technical product information. Product data represent the core of manufacturing enterprise. They should be protected against unauthorized accesses but, at the same time, should be promptly shared for design activities as well as for many other functions in the enterprise.
- Easiness of product data exchange by establishing proper interfaces towards enterprise functions (manufacturing, maintenance ...) and, vice versa, by defining communicational channels to gather any relevant information for the upgrading.
- Management of the enterprise knowledge database for storing and retrieving historical products. The carry over process, widely adopted to speed up the time to market of new products, benefits from the use of PLM tools.

The reasons of its inadequate diffusion are uniformly spread over enterprise organization and information technology system:

- Accessing product data from enterprise functions not directly involved in product development is a functionality provided by IT tools like ERP that are seldom present inside the SMEs.
- Product/process development is historically a self-governed function, with human resources, activities and hardware/software tools sharply separated by other industrial functions.
- The implementation of IT tools linking different enterprises areas requires a clear definition of decisional and informational processes.

The idea behind this research is that the work organization inside a SME is based on spontaneous collaboration that overcomes the attempts to organize the activities by introducing coordination mechanisms, as in the case of PLM implementation. Spontaneous collaboration works well as far as the dimensions of the firm are small enough to allow everyone to know where to find the information they need. As far as the enterprise grows a technology like PLM becomes necessary to assist the information retrieval. To assess the validity of this supposition we need to find a way to measure the extent of collaboration. The paper illustrates an original approach to the analysis of management processes of SME's product data. Using simple data-driven procedures, it is possible to forecast the impact of collaboration on SME organization without inferring formal models in any case available in literature [4]. In detail, the proposed analysis evaluates the SME readiness to PLM implementation, by matching the collaborative network with the functional organization of the enterprise.

2 Product and process information in SME: dataflow

It is useful to distinguish among products and processes to develop products. Innovative and competitive products increase enterprise's market share whilst efficient and robust industrial processes allows the enterprise emerging over the competitors. As a consequence, the information on enterprise products and processes, in terms of data and metadata (Table 1), has to be as efficient as possible along the whole product lifecycle.

Table 1: Enterprise information

	Product data	**Processes**
Business	Metadata (STEP, PDM)	Enterprise content management
Engineering	Data (GPS, STEP)	Manufacturing planning

PDM/PLM systems were born to control data proliferation in design activities, therefore their primary function is the control of product data in terms of integrity, protection and modification. A second important function regards data availability: user-friendly, simultaneous and multi-point access to product data [20].
PLM systems do not limit their function to database repository, but actively support product data development by tracking design activities, maintaining configuration of evolving products, and allowing the re-use of the data in the development of new products [9].
The enlargement of the scope of product data, which are no more property of design function but become capital of the whole enterprise, requires a redefinition of their accessibility and modifiability, with regard to other IT systems like ERP. The redistribution of the responsibilities on product data among enterprise functions represents the most innovative aspect of PLM systems. Obviously it is also the most difficult aspect to be addressed in PLM implementation and it justifies the lateness of PLM diffusion in the SMEs.
PLM systems have an impact on the entire enterprise and require a structured organization where informative flows, interfaces and decision makers are well identified. This is normal in large companies, but it is not predictable in SMEs, where the same person often provides different functions and where decision makers are individuated more on the basis of capital share than on the basis of organizational role and technical competence.
The intrinsic complexity of PLM systems and the informal organization of small companies represent a real difficulty in the implementation of PLM in SMEs [1].

2.1 SME organizational model

The technical literature widely illustrates case studies related to the implementation of PLM in large and well structured organizations. Unfortunately few experiences are available on PLM implementation in SMEs, where informal organizational structure does not allow the application of standard methodologies, based on the development of a detailed ontological model (SAP, UML, ...), in order to perform an efficient description of enterprise processes [8], [17], [25].
A preliminary analysis of enterprise organization should rely on the generic product lifecycle functions [2] schematized in Figure 1:

- Direction: The overall project management, this function is transversal to the others.
- Design concept: Idea for new product or, more often, product design enhancement is recognized, based on market knowledge.
- Market demand analysis: Manufacturer studies need for the new product design and estimates demand and feasibility of meeting demand.
- Engineering: Design engineers create the product design using all available information from the PLM system, including after-market factors, manufacturability data, customers' needs/preferences, and more.
- Sourcing: Procurement carry out preliminary work required to acquire parts, materials, components, equipment, and anything else is required to manufacture the product.
- Production: Product is built to design specifications established by engineering and using parts and materials acquired by sourcing. Conformance to specification is checked through quality control/assurance or process control methodologies.
- Distribution: Product is shipped to either distributors, who store it until order is received from customer, or directly to final customer.
- After market: Products are maintained, serviced, or repaired under warranty or as a value-added service. Using a unique repository of various after-market data ensures that after-market factors are taken into account in subsequent design projects, increasing the value of the product to the customer.

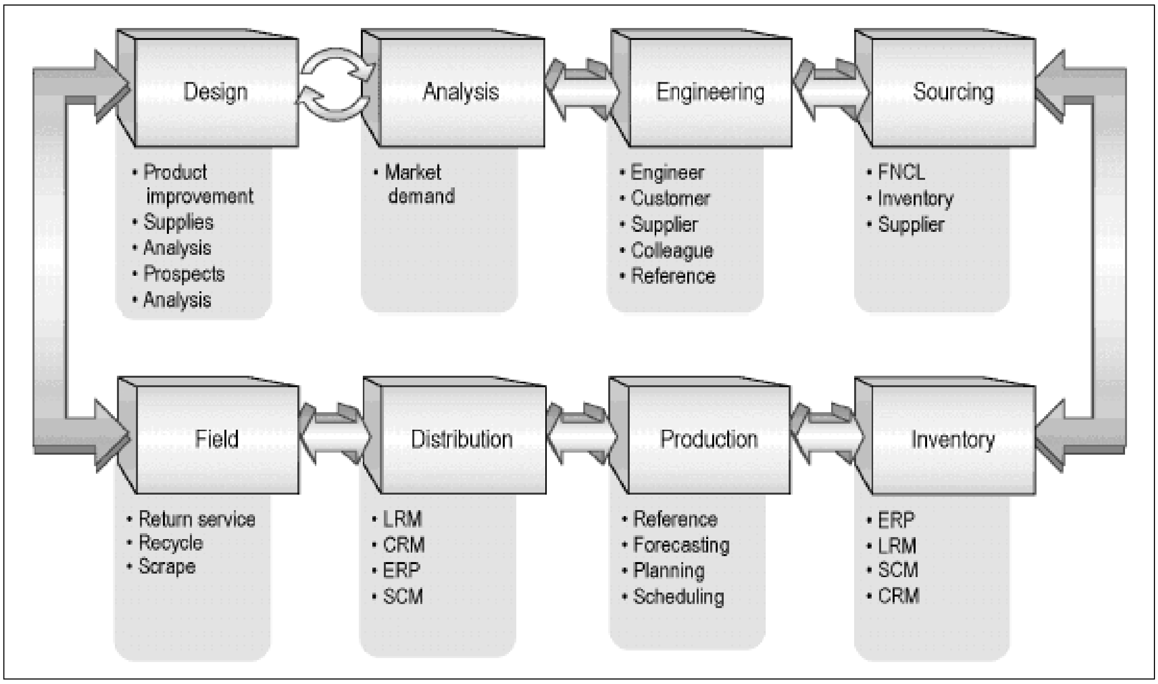


Figure 1: Product lifecycle analysis (from Aberdeen [1]).

The experimental data used in the paper were collected from a SME which designs, manufactures, installs and maintains industrial choppers all around the world. The enterprise structure, according to a functional point of view, adheres to the proposed model.

2.2 SME investigation procedure

There is a widespread commonplace saying that every SME works without any notable organization. Actually, the organization exists but it is usually not explicitly formalized: it is a natural organization based on significance relationships and spontaneous collaboration.
Therefore, it makes no sense to look at classical organization devices, like organization charts, information systems, team management tools (PERT, Gantt) [23]. There are four basic principles which make the organization of a SME efficient:

- Process-based cooperation, without a fixed separation of tasks.
- Extended sharing of knowledge, unfortunately transmitted in a informal way.
- Small teams with continuous communications and interchanges of information.
- Potential for outside development by having recourse to the networks of enterprises.

These same principles represent the hardest obstacle to an efficient description of the enterprise data, functions, processes and related managing tools.
Researchers adopt a simple data-driven approach focused on the investigation of enterprise documents in order to overcome the obstacle. A set of questionnaires submitted to enterprise people are used to collect information on produced and consulted documents thus providing an unbiased objective description of the true organization, processes and hierarchies operating in the enterprise.
The questionnaires play a fundamental role in diminishing the noise involuntarily introduced by interviewed people: the focus on documents and their management and not on sensible data regarding the enterprise organization, defines the search field and the required information.
Researchers submitted questionnaires to enterprise personnel during the development of a new project and transformed the enterprise in a living laboratory where researchers interacted with employers during the questionnaires filling, observed the product development process and analyzed the documents identifying their dependence relationships as well as their format.

2.3 SME information reorganization

Several operations applied to the information collected by questionnaires, provide a more structured description of enterprise and its organization, functions, processes and hierarchies:

- Analysis and evaluation of PLM functionalities in the enterprise context;
- Analysis and definition of product data: quantity, relationships, applicability, supporting media;
- Definition of enterprise functions and analysis of their role in product data management;
- Analysis of enterprise's acquaintance with informatics;
- Development of the DLSM (Data Lifecycle Simulation Matrix) representing the lifecycle of product data and their dependencies within the enterprise functions;
- Development of the DISM (Data Interchange Simulation Matrix) representing the interfaces used to communicate the product data among the different functions;
- Creation of a metric to exploit the enterprise position, according to technological and organizational aspects, against the best practice.

The DLSM and DISM matrices provide a complete picture of product data management across the enterprise functions and along the product lifecycle. On this basis it is possible to evaluate the organizational and informational structure by detailing the interactions with each enterprise function. Moreover, analyzing the DLSM and DISM matrix it is possible to extrapolate useful indicators which address the reorganization of the enterprise.

3 Reorganizing the data: DLSM and DISM Matrices

The core of the methodology is the building of a DLSM (Data Lifecycle Simulation Matrix), a square matrix containing a complete description of the enterprise organization based on its documents. The DLSM matrix represents the dependence relationship among the data/documents inside different enterprise functions.
The rows of the DLSM matrix, contain the enterprise documents classified according to the enterprise function responsible for their production and maintenance. The columns of the DLSM matrix contain the same documents presented in the same order. The dependence relationships among documents are made explicit by activating the cells at the intersection of the row and of the column corresponding to the associated documents. When a cell is activated, the document along the column will make use/reference to the document identified in the row. A formal description of DLSM matrix is:

$$
\begin{aligned}
DLSM(i, j = 1) &= \{\text{Enterprise documents - Source}\} \\
DLSM(i = 1, j) &= \{\text{Enterprise documents - Access}\} \\
DLSM(i, j) &= \begin{cases} 1 & \text{when document } i \text{ is accessed by document } j \\ 0 & \text{otherwise} \end{cases}
\end{aligned}
\tag{1}
$$

Figure 2 shows a pattern view of the 160x160 sparse DLSM matrix representing the documents required to describe product lifecycle within the analysed case study. The grey areas placed on the matrix diagonal highlight the enterprise functions: Direction, Marketing, Administration, Design, Planning, Supply, Manufacturing and assembling, Inventories, After sale services.

Figure 3 shows the upper and left corner of DLMS matrix including some documents produced by the two functions Direction and Marketing. Some interesting results can be extracted from simple operations performed on the DLSM matrix.

The DISM (Data Interchange Simulation Matrix) is a square matrix having the same structure of DLSM, but now the cells are activated when the format and the support used to save two inter-dependent documents are the same.
The core of the method is the transformation of the DLSM into the DISM. For every intersection activated in the DLSM, the document formats and supports are compared: if they agree (i.e. both MS Word format, or Excel format, or paper support, ...) the cell in the DISM is activated. Obviously there are quite less occurrence in the DISM with respect to DLSM.
Figure 4 represents the upper and left corner of the DISM matrix including some documents produced by the

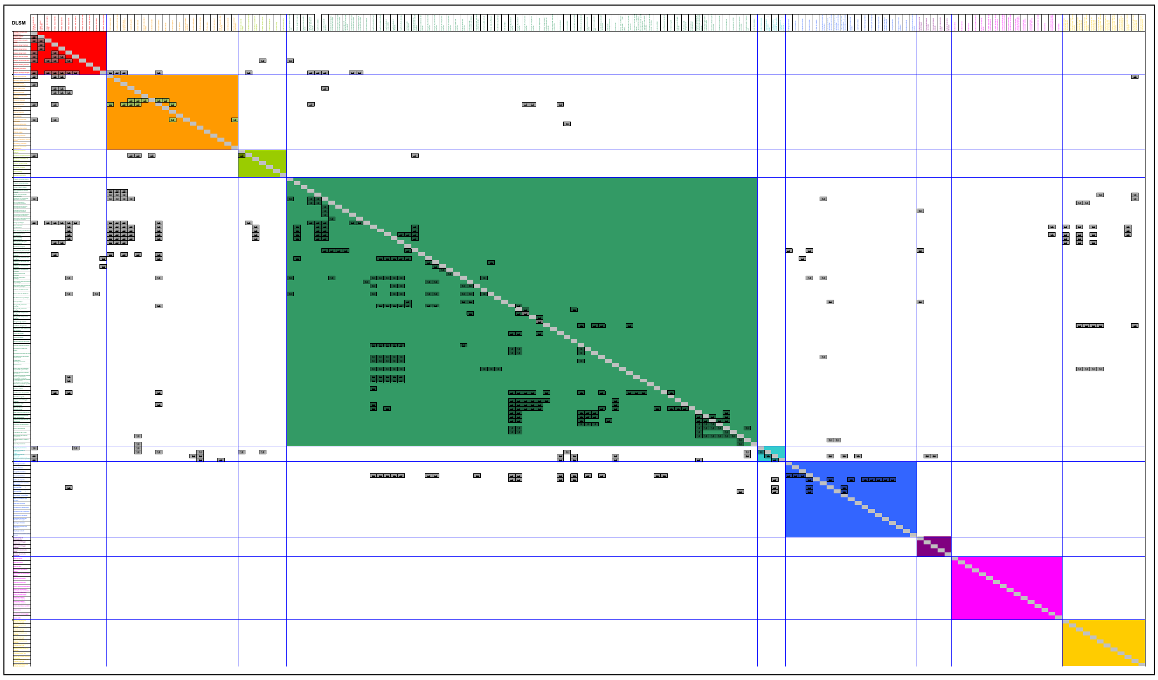


Figure 2: Overview of the complete DLSM.

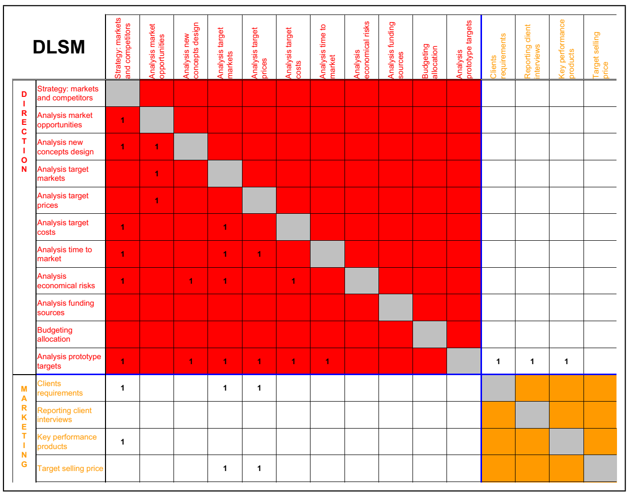


Figure 3: Sample of a portion of DLSM including some documents of the function "Direction".

enterprise function "Direction". For every intersection activated in the DISM, the compatibility of the adopted applications and their publication on the firm's network is verified and the sum over the rows and the columns occurrence are generated in order to deduce the integration in the product data flow.

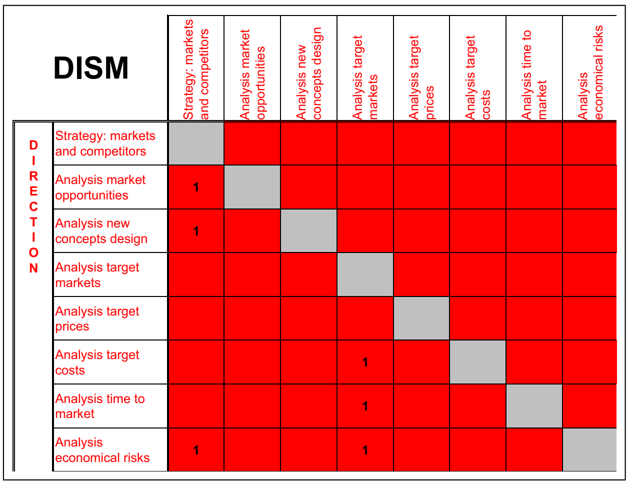


Figure 4: Sample of a portion of DISM including some documents of the function "Direction".

The activated cells in the DISM play a relevant role from a PLM perspective when the document format and the support are electronic. Taking in account only the interfaces based on electronic format, the distance of DISM from DLSM addresses the investments in the informational infrastructure the enterprise needs.

3.1 Analysing the DLSM matrix

The sum α and β of the occurrences along the rows and the columns of DLSM allows to determine quantitatively the amount of functional dependencies among the documents in the information flow. Table 2 presents the α and β values for the document "Analysis target prices" produced by the enterprise function "Direction":

Table 2: Dependencies of "Analysis target prices" document.

Document	α	β
Price target	1	7

The document is consulted by only another document ("Analysis of market opportunities", produced by the function Direction) and requires the existence of other seven documents belonging to the Direction, Marketing and Design functions.
It is noteworthy that the same document can be shared among different functions inside the enterprise, therefore another important parameter is the percentage of occurrences of every document (i) inside every function (j), associated to the matrix $DLSM(i,j)$.
As an example Figure 5 shows the percentage use of all the documents generated by the function "Direction". In the collaboration network, the function "Direction" behaves as a supplier, the other enterprise functions are then the clients.

The analysis can be profitably reverted in Figure 6 that shows the percentage use of documents, generated by other functions, through the function "Direction", which now acts as the client for the other functions. Different analyses of DLSM structure bring to light unpredicted outcomes.

4 Clustering the DLSM: multivariate analysis

The scope of this section is illustrating the motivations behind the choice of the clustering method used to group the project documents. Clustering of the documents is the main application of the DLSM matrix. Produced documents have already a taxonomy, belonging to one of the product lifecycle functions, that correspond directly

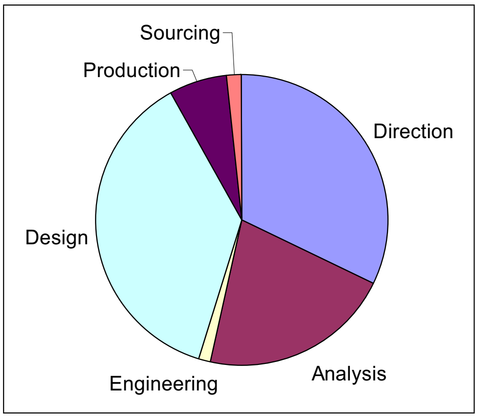
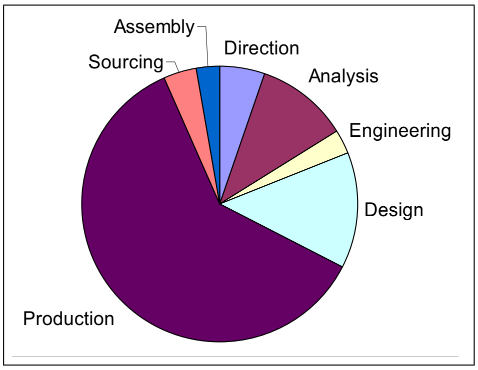


Figure 5: Use of Direction documents by different enterprise functions including Direction (labels represents the destinations).



,

Figure 6: Use of enterprise data by the Direction function (labels represents the origins).

to the enterprise departments. Nevertheless, a typical SME structure is by far different and usually simpler than what appears from the organizational chart. Therefore it is noteworthy to test if the clustering of the document based on their correlations matches the official taxonomy.

We remark that the clustering performed here has nothing to do with the known study field of Information Science that goes under the name of Document Classification. This latter can be defined in the following way [24]: "Document clustering is a fundamental operation used in unsupervised document organization, automatic topic extraction, and information retrieval. It provides a structure for organizing large bodies of text for efficient browsing and searching".

In our case, document clustering, based on the DLSM matrix of correlations, is only a pretext to analyse the enterprise organization from an unbiased point of view. In common with the ordinary Document Classification is the choice of unsupervised clustering methods. As there are reasonable doubts that the real structure of the collaboration network would match the enterprise structure it is convenient not to use the knowledge about the existing organization and leave the classification of the documents to an unsupervised method: the Statistical Cluster Analysis.

Cluster analysis investigates grouping by minimizing a suitable distance measure among the data. The distance function should give a way to measure the similarity between two documents. Euclidean distance is obviously improper for measuring the similarity in the correlations among documents. For the research aims the best suited is the Hamming distance that calculates the percentage of attributes that differ between two objects. Given the DLSM, the element *DLSM*(*i*,*j*) represents the attribute *j* of the document *i*. The number of documents and of attributes is the same, *N*. For every two documents *r*, *s* the Hamming distance d_H is:

$$d_H = \frac{DLSM(r,j) - DLSM(s,j)}{N} \qquad (2)$$

Hierarchical Clustering is the more appropriate technique for working with ordinal metrics. It creates a cluster tree, which is not a single set of clusters, but rather a multi-level hierarchy, where clusters at one level are joined as clusters at the next higher level. The decision of the most appropriate level of clustering is found by choosing the threshold of the inconsistency coefficient for each link of the hierarchical cluster tree. There are different agglomerative techniques [3], among which the nearest neighbour or single linkage. Two clusters are joined together on the basis of the minimum distance between the two nearest elements among all the existing clusters. If cluster A is made of documents a_i (i=1,...,m) and cluster B is made of documents b_j (j=1,...,r), the distance between the two clusters is:

$$d_{sl} = min(d_h(a_i, b_j))\forall i, \forall j \tag{3}$$

Single linkage is seldom selected among the techniques because it has the drawback of creating elongated cluster (in a spatial visualization of the distance measure) if there is a long chain of documents each one differing from the successive for some attribute. In present study the attributes are the dependencies of documents and it is reasonable to expect that every document be the starting point for a successive document.
Before to apply the clustering, a pre-processing has been executed on the DLSM to reduce the search space [5]. After a binary sorting, the void rows and columns have been excluded from the clustering. These represent documents which have been produced independently the ones from all the others. It is possible that most of them be useless documents and be requested by internal bureaucratic procedures.
The result of the application of the hierarchical clustering to the modified DLSM is the dendrogram graph represented in Figure 7. The dashed line represents a threshold inconsistency value chosen to group the documents in ten clusters which correspond to the nine divisions actually present in the enterprise.

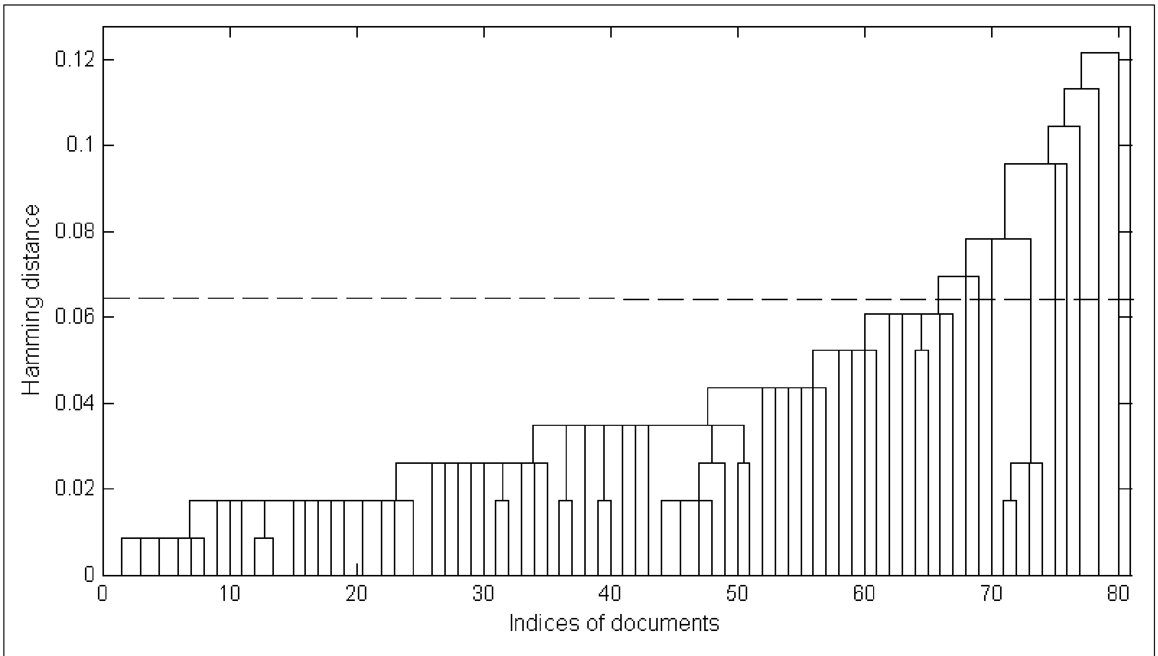


Figure 7: Dendrogram of the hierarchical tree for the DLSM matrix.

A number of considerations can be drawn on the result of this clusterization:

- One cluster (document index from 1 to 67) is by far larger than the others and includes documents belonging to nearly all the divisions;
- With the exception of the fourth cluster (document index from 71 to 74), all the others have only one document inside;
- These one-document clusters have an high unconsistency value, therefore they cannot be attributed to a larger one changing the agglomerative technique.

The first consideration describe a positive situation. The majority of documents have strict reciprocal dependencies. A possible explanation is that the SME applies implicitly a concurrent engineering strategy in the

management of new projects. As a matter of fact every project is carried on by an unstructured working team made usually of few persons if not just one, who are responsible for the production of the key documents which are used by all the divisions to produce their own documents.
Second and third considerations indicate the presence of dead end documents, i.e. documents that get their data from other documents but are no longer used to produce new documents. This is not necessarily an issue as it seems natural that every division produces a final report on the project done. The fourth cluster instead represents a small number of documents totally independent from the others. The issue is not in the presence of a cluster but in the high value of the inconsistency index (measured by the length of the dendrogram branches) that denotes an excessive distance between this cluster and every other.
From a PLM perspective the results of the DLSM clustering suggests the need of investments in the sharing of documents. Under the hypothesis of a working team operating in a concurrent engineering framework is essential to increase the accessibility of documents by all the enterprise functions.
In a SME, the issues related to the condivision of product data are strictly related to the possibility of access to the different database and data formats. Inside the large majority of the SMEs, the integration of the enterprise documentation is not considered a critical issue [11].

5 Evaluating the attitude to change

The questionnaires provide a picture of product data management across the enterprise functions and along the product lifecycle. Such information can be reorganized according to the DISM in order to gain a better knowledge on enterprise organization. On this base it is possible to evaluate the impact of PLM on the informational structure by detailing the effects on each enterprise function.
The evaluation of attitude to collaborative working in the CE context should be expressed in terms of costs and benefits, which depend on enterprise readiness and propensity to accept innovation [19]. The test case focused on the analysis of two attitude indexes: attitude to organizational changes and attitude to informational changes.
The attitude to innovation assumes different characteristics depending on enterprise functions. Authors extracted from technical literature a set of criteria to be applied to the most relevant functions, as identified on the basis of DSLM and DISM. Different evaluating scales and different weights have been applied according to criteria nature [17].
The results of evaluation of both organizational and informational indexes for the most relevant enterprise functions (P1 - Marketing, P2 - Design, and P3 - Supply) are summarized in Figure 8, which represents a pictorial description of the enterprise global attitude to innovation.

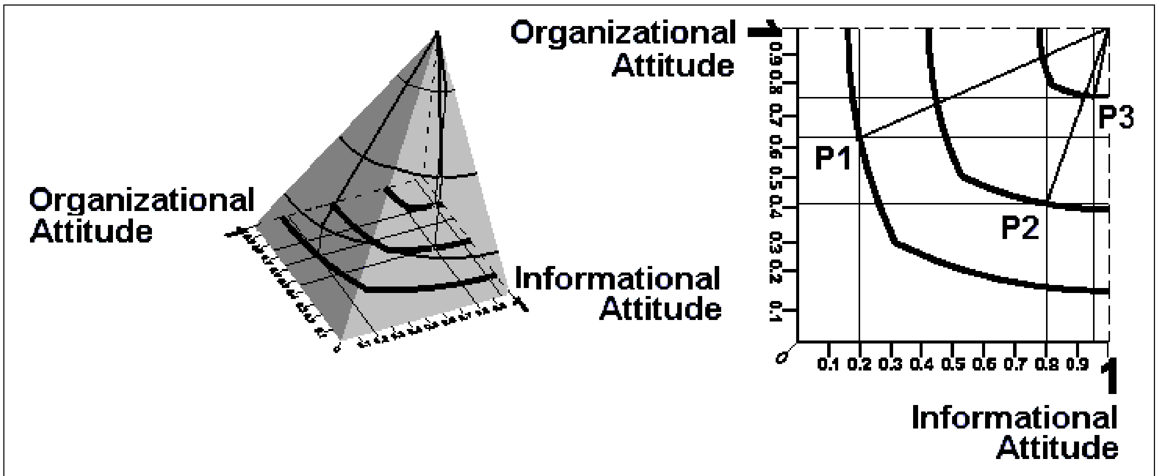


Figure 8: Global attitude index for the test case.

6 Conclusions

The paper focuses on the development of a method for the evaluation of enterprise readiness to collaborative product and process design, with particular attention to the implementation of PLM in the SMEs.
Data collection relies on questionnaires focused on the document management process adopted in the enterprise. The questionnaires are submitted to enterprise personnel while researchers integrates knowledge and analysis of decisional processes.
In order to evaluate the amount of Business Process Management required to upgrade the enterprise organisation to PLM pre-requisites, the authors developed an analysis tool consisting of a classification of product data flows and their supporting formats. Subsequent refinements of the gathered information allow to benchmark the enterprise propensity to the implementation of PLM.
The simple approach proposed in the paper seems to solve some of the problems related to the introduction of information technology systems in SMEs: insufficient awareness of organizational issues, insufficient involvement of end users, inadequate traininig of users.

Bibliography

[1] Aberdeen, "PLM - It's for Any Manufacturer Striving for Product Excellence", *An Executive White Paper*, Aberdeen Group, Inc., www.aberdeen.com, 2002.

[2] R.W. Bagshaw, S.T. Newman, "Structured approach to the design of a production data analysis facility Part 1: Conceptual design and information requirements", *Int. J. Prod. Res.*, Vol. 39, no. 10, pp. 2121-2141, 2001.

[3] A. Cevahir, "Clustering of Correlated Documents into Designated number of Clusters: A Practical Approach", www.cs.bilkent.edu.tr, 2005.

[4] D. Chen, G. Doumeingts, "Developing a Theory of Design - A Tentative Approach Using Mathematics", *Studies in Informatics and Control*, Vol.11, No.1, 2002, ICI, Bucharest.

[5] G. Czibula, G. Serban, "Hierarchical Clustering Based Design Patterns Identification", *Int. J. of Computers, Communications & Control*, Vol. III, pp. 248-252, 2008, CCC Publications.

[6] Datamation Limited, "Understanding Product Lifecycle Management", *Rep. No. PLM-11*, Rev. No. 1.0, September 2002, Cambridge, UK.

[7] European Commission, *SME and cooperation, Observatory of European SMEs*, 2003, Enterprise Pubblication, Brussels.

[8] B. Eynard, T. Gallet, P. Nowak, L. Roucoules, "UML based specifications of PDM product structure and workflow", *Computers in Industry*, Vol.55, pp. 301-316, 2004.

[9] B. Fagerström, L. Olsson, "Knowledge Management in Collaborative Product Development", *Systems Engineering*, Vol. 5, No. 4, pp. 274-285, 2002.

[10] F.G. Filip, "Decision support and control for large-scale complex systems", *Annual Reviews in Control*, Vol.32, No.1, pp. 61-70, 2008.

[11] F. Franceschini, *Advanced quality function deployment*, Boca Raton, St. Lucie Press, 2002.

[12] H. Kaschel, L.M. Sánchez y Bernal, "Importance of Flexibility in Manufacturing Systems", *Int. J. of Computers, Communications & Control*, Vol. I (2), pp. 53-60, 2006, CCC Publications.

[13] J. Krauth, Introducing information technology in small and medium sized enterprises, *Studies in Informatics and Control*, Vol.8, No.1, 1999, ICI, Bucharest.

[14] A. Kusiak, *Concurrent Engineering: Automation, Tools, and Techniques*, Wiley, USA, 1992.

[15] T.W. Malone, K. Crowston, "The interdisciplinary study of coordination", *ACM Computing Surveys* (CSUR), Vol. 26, No. 1, pp.87-119, 1994.

[16] S.Y. Nof, G. Morel, L. Monostori, A. Molina, F. Filip, "From plant and logistisc control to multi-enterprise collaboration", *Annual Reviews in Control*, Vol.30, No.1, pp. 55-68, 2006.

[17] K. Schmid, I. John, "A customizable approach to full lifecycle variability management", *Science of Computer Programming*, Vol.54, pp. 259-284, 2004.

[18] M.G. Singh, A. Titli, Systems Decomposition, *Optimisation and Control*, Pergamon Press, 1988.

[19] R. Sudarsan, S.J. Fenves, R.D. Sriram, F. Wang, "A product information modelling framework for product lifecycle management", *Computer Aided Design*, Vol.37, pp.1399-1411, 2005.

[20] J. Stark, *Product LifeCycle Management - 21st Century Paradigm for Product Realisation*, 2005, Springer-Verlag, London, UK.

[21] S.N. Talukdar, "Collaboration rules for autonomous software agents", *Decision Support Systems*, Vol. 24, pp.269-278, 1999.

[22] S. Tichkiewitch, D. Brissaud, "Co-ordination Between Product and Process Definitions in a Concurrent Engineering Environment", *Annals of CIRP*, Vol. 49, pp. 75-78, 2000.

[23] A. Tiwana, B. Ramesh, "A design knowledge management system to support collaborative information product evolution", *Decision Support System*, Vol.31, pp.241-262, 2005.

[24] C. Xiaohui, G. Jinzhu, T.E. Potok, "A flocking based algorithm for document clustering analysis", *J. of Systems Architecture*, Vol.52, pp. 505-515, 2006, Elsevier.

[25] S. Zina, M. Lombard, L. Lossent, C. Henriot, "Generic Modeling and Configuration Management in Product Lifecycle Management", *Int. J. of Computers, Communications & Control*, Vol. I (4), pp. 126-138, 2006, CCC Publications.

Dario Antonelli Italy 1966, holds a M.S. Degree in Mechanical Engineering from the Politecnico di Torino, 1990. He worked at Fiat Research Center until 1992. He is currently Associate Professor at the Department of Production Systems and Economics of Politecnico di Torino. His scientific activity is mainly related to numeric Finite Element Simulation of metalworking processes, to experimental identification of process parameters and to Supply Chain Management.

Paolo Chiabert (b. June 28, 1963). M.Sc. Degree in Electronic Engineering, Politecnico di Torino (1993), IBM Consultant (1994), Researcher in Design and Methods of Industrial Engineering at the Politecnico di Torino (1995-2003), Associate Professor in Technology and Manufacturing Systems (2004). According to the Product Lifecycle Management perspective, the research interests range from Design methods, to Manufacturing technologies, to Production planning and control, to Quality management. Since 1997 Italian expert in the ISO Technical Committee 213 - Geometrical Product Specifications and Verification, he is member of the GPS Technical commission at UNI.

Int. J. of Computers, Communications & Control, ISSN 1841-9836, E-ISSN 1841-9844
Vol. V (2010), No. 1, pp. 20-41

A Software System Development Life Cycle Model for Improved Stakeholders' Communication and Collaboration

S. Cohen, D. Dori, U. de Haan

Shalom Cohen, Uzi de Haan
Technion, Israel Institute of Technology
Haifa, Israel
E-mail: {shalom1,uzid}@tx.technion.ac.il

Dov Dori
1. Technion, Israel Institute of Technology
Haifa, Israel
2. Massachusetts Institute of Technology
Cambridge, MA, USA
E-mail: dori@ie.technion.ac.il

Abstract: Software vendors and entrepreneurs, who try to introduce an innovative software product to a specific organization or an entire market, enter a long and tedious process. During this process, the market and various organizations evaluate the product from different perspectives, such as software robustness, manufacturer reliability, and corporate need for the product. The vendors and entrepreneurs engaged in this process encounter decision crossroads for which no relevant guidance exists in the literature.

The research closely monitored the processes associated with the introduction and assimilation of an innovative off-the-shelf (OTS) software product into five different organizations in different vertical market segments. Observations were carried out to assess organizational and marketing processes and to document and analyze what the software product undergoes before it is accepted for acquisition or full implementation within the organization.

The research outcomes offer a unified, collaborative multi-tier System Development Life Cycle (SDLC) framework and methodology for packaged OTS software products that greatly improves communication and collaboration among the stakeholders. Each tier addresses a different force or stakeholder involved in the software market: vendor, customer, consultants and integrators. All stakeholders refer to the same time-line thus; tasks of various stakeholders are streamlined. Adherence to the unified time-line brings about an increased amount of stakeholder interaction, communication and collaboration.

Newly found tasks that improve communication and collaboration among stakeholders include (1) offering of the OTS software product together with personnel as a bundle, (2) an improvisation-intensive iterative task of weaving potential customers' requirements into the prototype, and (3) a third sale milestone, representing the successful diffusion of the product. The significance of this interdisciplinary research stems from its unique position at a crossroad between software engineering, marketing, and business administration, which has not yet been sufficiently explored or cultivated.

Keywords: collaboration, system development life cycle model, stakeholders.

1 Introduction

Two major trends dominate the software development world today. The first is the shift of organizations from fulfilling their own software requirements in-house to buying it on the market, either as an off-the-shelf packaged software product or from a company tailoring a specific solution.[1] The second trend is the shift from developing tailor-made software to purchasing packaged software from vendors either in stores or directly from the vendors.[2] Here we assume that acquisition of packaged software is done by an organizational consumer.[1]

When relating to a packaged software product, which may be seen as a system by its own accord, one should make a clear distinction between a software product and an Information System (IS). [3] An IS is made up of a number of software products or modules put together.[1] This research examines an off-the-shelf (OTS) packaged software product as a system[2] which goes through the various stages of System Development Life Cycle (SDLC)[3]. Many software development processes and models use stages outlined in SDLC.[4] We relate to SDLC not only in its traditional "waterfall" sense, but also to other models outlining the stages in software development. Since these models, including the spiral and Rapid Application Development (RAD) are not as broadly known as the "waterfall" model and are less useful for explaining the market effects on software development delineated hereafter [1], we do focus on the waterfall model as a reference.

The lifecycle of an information system includes the various phases that a software product goes through starting with its conception all the way to the stage when it is no longer available for use.[5] The software lifecycle, depicted in Figure 1, typically includes the following phases: requirements, analysis, design, construction (or coding), testing (validation), installation, operation, maintenance, and the less emphasized retirement.[6]

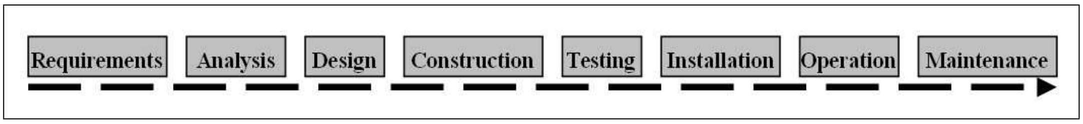


Figure 1: The traditional phases of the System Development Life Cycle Model

These basic phases have also been adopted for IS acquisition purposes. Although the names of the phases were changed where appropriate, the basic structure and timeline have been kept. The phases of IS acquisition, shown in Figure 2, are project justification, financial evaluation of the project, preparations for acquisition, Request For Proposals, vendor evaluation, contract negotiations and, implementation and maintenance.[7]

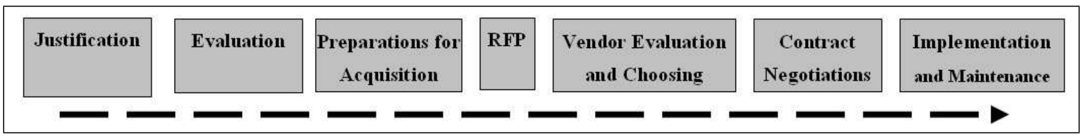


Figure 2: The SDLC model adapted to the acquisition process of Information Systems

The literature research, summarized below, indicates that no significant attempt has been made to extend the SDLC model to other situations encountered by many software vendors and software developing entrepreneurs.[2] Software vendors and entrepreneurs, who try to introduce an innovative software product to a specific organization or an entire market, enter a long and tedious process. During this process, the market and various organizations evaluate the product from different perspectives, such as

[1] The scope of the research is limited to describing the organizational consumer and not the private home user which is a discussion in its own right and differs in many ways from that outlined hereafter.

[2] "The System," "OTS software product," and "Packaged Software Product" will be used from now on interchangeably

[3] The acronym SDLC will denote here the system development lifecycle model as it relates to software products. Its implementation with regards to OTS software products will be discussed here.

software robustness, manufacturer reliability, and corporate need for the product. The vendors and entrepreneurs engaged in this process encounter decision crossroads for which no relevant guidance exists in the literature.

This lack of guidance is somewhat surprising, since information systems development is best understood as a market phenomenon. It is a perspective which highlights how software is developed, who performs the development, who sells the related products, and how they are introduced to users.[1]

2 Research Goal and Objectives

The goal of this research is to develop and evaluate a collaborative multi-tier lifecycle development model for packaged off-the-shelf (OTS) software products. The proposed model accounts for market and organizational factors and the way they are woven into the traditional phases of software development. To this end, the research has monitored, outlined, characterized, defined, and mapped specific phases which OTS software products typically go through. The resulting comprehensive model relates to the development, marketing, assimilation, and other organizational aspects of the OTS software product. The research has identified and defined new, modifiable software lifecycle processes, the adoption of which might benefit various stakeholders under various marketing conditions. Our hope that the prevailing model will entail this aim by creating a task-based learning community which is a group of people who are organized around a task i.e. stakeholders, collaborating for a specified period of time to produce a product.[8]

Here, we attempt to create a unified exhaustive SDLC framework on one timeline with a number of tiers, creating a new collaborative multi-tier system development life cycle methodology. Each tier addresses a different force or stakeholder involved in the software market, such as producers, consumers, consultants, and integrators.[1] The basic time frame of the SDLC, especially the beginning (inception) and end (implementation and maintenance) is kept. The various milestones along the SDLC time line indicate an appropriate task for each tier and an explanation of that task. Tasks on the same vertical axis are to be performed concurrently and collaboratively.

Figure 3 depicts a possible scheme of the proposed collaborative multi-tier market- and organization-oriented SDLC model to be fleshed out as a result of the field study outcomes. The list of stakeholders stacked in Figure 3 is by no means exhaustive.

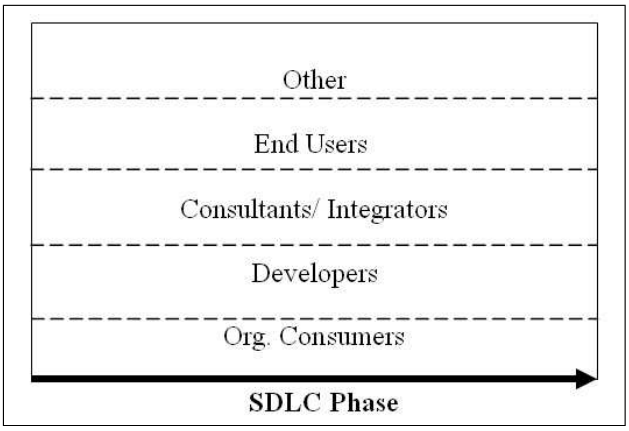


Figure 3: A possible scheme of the proposed collaborative multi-tier market- and organization-oriented SDLC model to be fleshed out as a result of the experiment outcomes

The main novelty of this research is that it is a first field-based study that is aimed at the establishment of a collaborative multi-tier SDLC model and a methodology based upon it. In addition, in most IS field studies, researchers have access to a limited amount of evidence and observations in participating organizations. In contrast, this research takes advantage of the fact that the software that was examined has been developed at the Technion and was fortunately being positioned at the point of time required for this research. Due to the special ties between the Technion and the software vendor, this study has had access to evidence and observations that are normally out of reach to researchers. The validity of these unique findings was tested against the more abridged findings of the control case studies.

3 Literature Review

The only academic development model incorporating any type of market effects is that of Carmel and Becker.[9] They developed a process model for packaged software development which was partly empirical.

Carmel and Becker [9] point to a few market-related actions necessary to be performed in some, but not all of the described stages. Actions like "assessing product differentiation considerations" are attached to the "Initial Screening Stage" of the "Requirements loop" with no explanation how they can be achieved.

In summary, Carmel and Becker [9] were the first to attempt a complete process model which adds marketing tasks. Their model, however, was only partially based on empirical findings, and instead of having the market define software market needs and SDLC phases, they suggested them a priori. Moreover, as Cusumano et al.[10] noted, their reliance and justification of a pivotal "freeze specification" stage is problematic in a highly volatile market.[10]

As Carmel [2] noted, no major study had been conducted on market introductory effects on packaged software innovations before 1995.[2] Nevertheless, the idea of introducing a market-based perspective into Information Systems development was introduced later on.[1] He juxtaposed a market-oriented approach with a simplification of the traditional "waterfall" model. At the basis of his idea is a separation of the traditional SDLC model (from development to user introduction) into two separate parallel models, one for the software developer and the other for the software consumer. In addition Sawyer had several interesting assertion as to the growing importance of additional stakeholders in the development process.

For example:

- Third parties (consultants, vendor representatives, etc.) have an increasing role in the initial stages of SDLC. Consultants/integrators are now also part of the Information System Development (ISD) process, as they enable and mediate the software market. This contributes to widening the chasm between users and vendors. This chasm is bridged only by indirect links between customers and developers via intermediaries or customer surrogates.[11]

- System installation requires a third party in charge of installing the product, customization, and training.

- The development process is of smaller importance to the consumer than the final product.

Although most of the assertions in Sawyer's model may make sense, they are in no way based on empirical evidence and do not have a direct connection to an SDLC model currently in use. His model lacks due reference to the producer's side, an aspect which this research has elaborated on.

In order to cover a large number of organizational and market-related SDLC influencing factors, we searched for academic and professional models in seven domains. We began by looking at the above mentioned few existing market based IS/Software Development models to see how an innovative OTS software product is produced in the market. We then continued by looking at works on software cycles and structured development studies to uncover the new OTS software manufacturing methods. Leaving the IS domain, we followed Moore's technology diffusion theory to look for models on technology adoption, innovation introduction and marketing diffusion theory which may relevantly describe the diffusion process of an OTS software product too.[12] We then reviewed research in the cross domain of organizational decision making on IS/Software related issues to learn how the common IS/Software related decision-making processes in various organizations are performed. The maturity of the software product, as well as that of the organization, is a matter of much interest to Industrial Engineers and Business Administrators and it influences entrepreneurial vendors tremendously and therefore reviewed here too. The relatively young academic field of Entrepreneurship was searched for adequate models and research on innovation, innovation-exploration and entrepreneurship in the software market. Finally, we surveyed the market for best practices and existing methodologies for OTS software development by entrepreneurial vendors. Table 1 summarizes the main studies related to this research topic under the various categories. From the above literature review we learnt about the possible variables and added them to the examined research's model as described in the following section.

Tabel 1. Summary of main studies SDLC and related subjects.

Aspect	Article	Empirical
Market Based IS/ Software Development	Carmel and Becker[9]	Very partial
	Keil and Carmel[11]	Yes
	Sawyer[1]	No
	Sing and Kotze[34]	Partial
Software Cycle and Structured Development	Cusumano et al.[10]	Yes
	Cusumano[35]	No
	Carmel[2]	Yes
	Clark and Wheelwright[36][37]	Partial
	Wheelwright and Clark[38]	Partial
	Boehm and Bose[39]	Yes
	Fine[40]	No
	Avison and Fitzgerald[4]	No
	Ebert[42]	Yes
Technology/ Innovation Introduction/ Diffusion Theory Marketing[4]	Mustonen-Ollila and Lyytinen[43]	Yes
	Lucas and Spitler[44]	Yes
	Davis[45]	Yes
	Moore and Benbasat[46]	Partial
	Brancheau and Wetherbe[47]	Yes
	Cooper and Zmud[48]	Yes
	Fichman and Kemerer[25]	Yes
IS related Decision-Making and Software Acquisition Processes	Verville and Halingten[49]	Yes
	Nelson et al.[50]	Yes
	Iivari and Ervasti[51]	Yes
Software Product and Organization Maturity	Paulk[52]	No
	Nordman [53]	Report
	Lee and O'Connor[54]	No
	Montaguti et al.[55]	No
Hi Tech Entrepreneurship	Shane[56]	Partial
	Shane[57]	No
	Murray and Tripsas[58]	Yes
	Baker et al.[23]	Yes
	Vera and Crossan[27]	Yes
Best Practices	NIH Matrix[59]	Best Practice
	Agile[61]	Best Practice

[4]For the most part, the writings in this discipline have not distinguished between the more general definition of an IT product and a specific IS/software-like product.

4 Methods and Experiments

In this section, we first provide a short explanation to the research method used (a), and then we describe the case study sites selected (b). Data collection efforts are described in (c) and finally in (d) we cover the preliminary research model.

a. Case Study Methodology

Yin identified three main types of case studies based on the purpose for which they are used [13]:

(1) Explanatory - A case study intending to explain the casual links in real-life interventions that are too complex for survey or experimental strategies.

(2) Descriptive - A case study that emphasizes the formation of hypotheses of cause-effect relationships, where a descriptive theory must cover the depth and scope of the case under study.[14]

(3) Exploratory - A case study in which the fieldwork and data collection may be undertaken prior to definition of the research questions and hypotheses. The framework of the study must be created ahead of time to maximize what can be learned, knowing that time is limited. The selected cases should be easy and should include willing subjects.[14]

The research strategy utilizes a combination of exploratory and descriptive case studies. The study does not explain an existing theory, so it cannot be categorized as explanatory. Rather, it tries to describe and explore emerging software development and marketing processes. Fieldwork and data collection were done prior to exact definition of the research questions and hypotheses generations.[15]

The research includes multiple exploratory/descriptive case studies, using replication logic. Replication logic is a logic by which case studies are selected to create a multiple-case design. Cases are selected so that they can either produce typical, negative or disconfirming results or exceptional/discrepant results. This form of case selection is also known as a theoretical sampling of cases as opposed to the normal sampling logic used in quantitative methods.

The outcomes of this design are improved theory, generalization ability and cross-case analysis. The latter is achieved by the use of two additional case studies which serve as control or baseline studies. Each case study is treated as an independent experiment, and the entire study is comprised of and based upon a sequence of multiple experiments.[16] When a case study strategy is agreed upon, it permits for both qualitative and quantitative sources of evidence to be collected and analyzed.[15] The collection and analysis of these two complementary forms of evidence has enabled triangulation. Various methods of data collection and fact retrieval were utilized, as described in section (c) of this chapter.

The field-study strategy, by which this research obtained insights into the processes that innovative software products go through, is an empiric study with multiple case studies. The OPCAT software was introduced into five organizations operating in mostly different vertical market sectors, so that the lessons learned from them cut across sectors. The list of sectors included banking, military, avionic, software and banking-software. One additional off-the-shelf software product (Product B) was introduced to a telecom company, and the final product (Product C) was introduced to a software organization. The number of case studies chosen (7) reflects a practical balance between the need for sufficient ground for generalization of the findings and the research time and capacity constraints. The number corresponds to the recommended range of 4 to 10 cases for theory building purposes.[15]

b. Case Study Sites

As the choice of organizations in which to perform the case studies is not a pure random sample, we tried to compensate for this by using theoretical sampling [13] designed to cover a broad spectrum of sectors, company sizes and locations, as depicted in table 2.

The customer organizations chosen for the case study sites were:

Org.1-E is a large Airborne Avionics Systems Manufacturer that employs over 3000 employees.

Table 1: Profile of firms in the exploratory/descriptive multi-firm study

Company	Sector	Size(Employees)	Location	Introduced Product
Org.4-B	Banking	>10.000	Israel	OPCAT
Org.1-E	Military/Avionics	>10.000	Israel	OPCAT
Org.2-EL	Military	<1.500	Israel	OPCAT
Org.5-Q	Software/Cellular	-100	Israel	OPCAT
Org.3-S	Software/Banking	-1.000	Singapore	OPCAT
Org.6-C	Telecom	>1.000	Israel	Non OPCAT
Org.4-B	Software	>7.000	Israel	Non OPCAT

Org.2-EL is a medium sized division (<1000 employees) of a high-technological military products manufacturer of ground, air and space-related products.

Org.3-S: Six of the seven case studies were held in the same country and the remaining study took place abroad, where the sale procedure to a large Asian banking software developer employing around 1000 people was followed.

Org.4-B is one of the largest banks in Israel, employing around 10,000 employees. The attempt was made to the bank's business software applications division.

Org.5-Q is a small software company with less than 100 employees, developing software for the cellular phone industry.

In addition to introducing OPCAT into five organizations, two additional case studies - Org.6-C, Org.7-BM - were held in which similar OTS software products were followed by means of their introductory phases into the market. These two additional case studies served as baseline case studies and assisted with both building the validity and analysis of the findings from the first five studies and with building a more robust and accurate SDLC model[4].

The case study sites were monitored periodically according to the type of evidence that was collected. For routine correspondence and product-related documentation, ongoing collection was used. For researcher observations, such as meeting attendance with adopting organizations, they were held according to the case study's natural timetable. Evidence collection sessions that is pushed by the researcher, such as questionnaires/surveys and interviews, were held at fixed time intervals across all case studies so that a matrix of observations - period versus company - was created. These mixed monitoring methods enabled the evaluation of evidence versus specific reference points in time and the description of continuous events as they were unfolding.

c. Data Collection

The software products that were introduced into each of these organizations, and the processes that they underwent thereafter until successful adoption installation and acquisition, or possibly rejection by the organization, were monitored and meticulously documented. Four different types of evidence collection were utilized for the monitoring and documentation of the above- mentioned processes: direct passive observations, documentation collection, open-ended and focused interviews and physical artifacts i.e. generated computer code or diagrams.

Table 3 summarizes the data collection efforts in the five main case studies held with the OPCAT vendor. The table also clearly indicates that the most extensive case studies as far as data collection was concerned were Org.1-E and Org.3-S. Two of the remaining 3 studies were shorter studies, mainly as they represent failed attempts of implementation by OPCAT, and thus spanned a shorter life-cycle.

[4]Due to strict non-disclosure restrictions, the information regarding the two baseline case studies, as well as the products, organizations, and customers examined has been kept confidential notwithstanding its use for hypothesis building and generalization purposes.

Table 2: General - data collection types and data collection summary statistics

Case site	**Duration (Months)**	**Inter-views**	**Direct Passive Meeting Documen-tation**	**Question-aires**	**Documentation Emails/ Hard copy/ PPTs/Other**	**Other Physical Artifacts**
Org.1-E	14	4	12+5 day brainstorming session	-	2/ 5/ 10/ 5	3
Org.2-EL	14	4	7	1	22/ 3/ 6/ -	30
Org.3-S	20	6	12+5 day session Course	0	157/ 3/ 8/ 3	8
Org.4-B	9	2	8	1	12/ -/ 4/ -	2
Org.5-Q	6	2	5	1	10/ -/ 3/ -	2

The evidence gathered from these case studies was then compared and analyzed and eventually enabled the identification and definition of common phases that the software went through in the various organizations and industries.

d. The Preliminary Model - A General View

Per the research method described above we defined our own set of a priori basic constructs for this research. The various preliminary variable groups and their interaction are modeled using Object Process Methodology (OPM), which provides a variety of complexity management tools that help diagram the model clearly and efficiently.[17] The top-most diagram, seen in Figure 4, demonstrates the main process of successful OTS software product implementation, which this research examined.

This process and its impact on the SDLC of OTS software product innovations was our dependent variable. This process is handled and impacted by the various stakeholders in today's OTS software product market, i.e., the vendor, adopting organization, third party integrators/consultants and, indirectly, other market and industry effects that constitute the intervening and contextual variables of this model, respectively.

The stakeholders interact via external non-systemic, environmental social networking process, in which they exchange leads, assign projects, etc. The impact of this process as a whole was of concern to this research, but its internal components and intricacies were not further elaborated, as the issue of social networking has already attracted extensive writing and research.[18]

The independent variable of this model is the OTS software product. For the sake of simplicity, the object representing the product includes only four basic states: specified, developed, acquired, and implemented. These are the most important states in a software product's lifecycle from the initial undeveloped product state, i.e., product in specification format only, to the successfully implemented product by an adopting organization.

Research model links in general represent possible hypotheses resulting from these relationships. Thus, the bidirectional effect links connecting the various attribute groups in Figure 4 mark the possible influence each group may have on others. The bidirectional links generalize unidirectional and bidirectional influences and suggest the variable group undergoes certain changes once the process is performed.

The preliminary model spanned 38 variables brought together from the various IS, OR and marketing domains discussed in Section 3. For the sake of brevity we do not bring here a full discussion regarding the reasons for their inclusion, and the variables comprising each variable group|footnoteThe full expla-

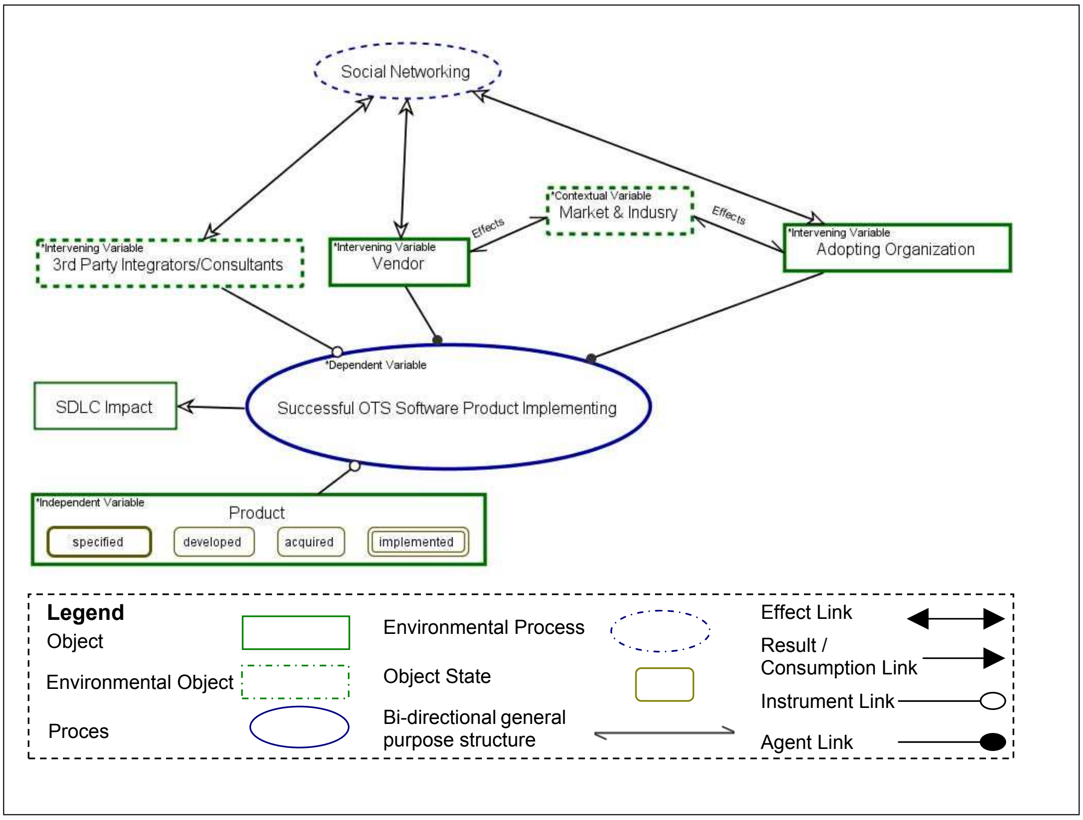


Figure 4: The proposed preliminary logical model

nation is readily available from the authors.. Furthermore, the full list of initial variables, as well the final ones, is given in table 4.

5 Intermediate Findings and Conclusions

The findings of this research include the final collaborative multi-tier SDLC research model and specifications of how the original research model has been modified throughout the research by omitting, adding and merging variable groups and variables. The aim here was to show how the model and its variables were validated through the various case studies conducted. A table is used to show the original model's variables vs. the final model's variables, and how each variable gained or lost validity based on evidence from the case studies.

Therefore, in this section, we begin by describing the changes to the original research model in (a), and continue with explaining about Lead-Driven Development (LDD) described in (b) brought as a partial downscaled example of the much larger and full Collaborative Multi-Tier System Development Lifecycle model. We end the section by giving a short explanation as to the contents of the full Collaborative Multi Tier System Development Lifecycle model in(c).

a. Changes to the Original Research Model

Following the guidelines of the case study research methodology [16], we entered the case sites using a preliminary suggested research model described in Section 4. As a case study proceeds, the research model is often updated by adding previously missed-out variables and deleting unnecessary or irrelevant ones. Table 4 lists the variables in the original and final models, their inclusion or exclusion in the preliminary and final models, the case studies upon which the exclusion or inclusion were based, and the nature of the impact of the variable. The nature is depicted using three symbols: +, - and OT, as explained next. The plus symbol is used to denote a positive influence on the successful sale and implementation of the OTS software product. The minus sign is used to denote a negative influence on the successful sale and implementation of the OTS software product. OT, which denotes "Other" reasons, is used when the impact is of a compound or qualitative nature. The last "Based on" column provides a supporting reason from the literature for the inclusion or exclusion of a specific variable. Explanation of the unique impacts and findings of the findings mentioned above:

The point in time where the implementation process is completed has been found to be, in the discussed product type, the third sale point. This is so as an initial first sale is either an impulsive buy or an exploratory attempt and is bundled with a human implementer. The second sale is a post-sale buy to try and achieve the product's associated benefits independently of external vendor human resources attached. The third sale constitutes reconfirmation of the product's benefits to the adopting organization and is characterized by purchase of licenses and a long-term support plan.

The political factors were observed in only one case study which accidentally took part during the Second Lebanon War, which took place in Northern Israel during the summer of 2006. However, we attributed this to coincidence and did not otherwise find any political issues effecting the market or industry and therefore deleted these two variables from the final model.

A unique influence was found in two of the larger case study sites, i.e., the partially-government-owned organizations. In these sites, a very strong influence of outsourced personnel, sometimes even positioned within the adopting organization's decision making units, was noticed.

The addition of the outsourced human resources as a descriptive characteristic of the customer's users was done in tandem with the addition of the same variable in the vendor's descriptive variables. This is possible, as in many closed and highly specialized industries, many of the organizational employees today are outsourced employees, who are often sent from a common pool of HR outsourcing companies and employees.

b. Lead-Driven Development

Based on the case studies carried out as part of this research, a new approach to software development for off-the-shelf (OTS) products of entrepreneurial vendors has been identified. The new model, called Lead Driven Development (LDD) includes detailed guidelines for entrepreneurial vendors developing OTS software. These include directions for pure development procedures (at the coding level) along with organizational steps to be held in conjunction with the coding process to support successful product implementation. This model relies on and revolves around an innovative procedure of improvisation, which is new to this industry. Improvisation counters many current trends which state that increased formality yields successful implementation.

As Table 5 indicates, LDD may be highly beneficial to the vendor. Examining table 5, we see that in all the five case studies, some form of LDD was followed. The classification level of LDD correspondence of each vendor per each case study site was scored on a scale of 1 to 5, where 5 represents complete correspondence. The classification was made by a number of uninvolved parties who checked for a clear-cut correspondence of the software introduction and development process to LDD. Since we examined the development process performed by the vendors and did not follow other stakeholders, we isolated the benefit associated with the use of LDD to be the influence of LDD on OTS product sales in the corresponding case study. Benefit was therefore observed as the influence of LDD in achieving a preliminary sale with an organization (first sale). A higher level of benefit was achieving a second sale and the highest level - a third sale. As explained later, a third sale is a measure of successful implementation. Since the observed vendor is of an entrepreneurial character, associated benefits of an efficient development process, such as shorter coding times or increased flexibility were not accounted for.

Table 4. Original vs. final research model variables

Top Group	Variable Group	Variable	Original Model	Final Model	Case Studies	Nature of Impact/ Signi-ficance	Based on
Product	SDLC Maturity	Introduction Stage	✓	✓	EL,E,S,B,Q	-	
Product	SDLC Maturity	Growth Stage	✓	✓	EL,B,Q	+	
Product	SDLC Maturity	Maturity Stage	✓	✓	NA	+	
Product	SDLC Maturity	Decline Stage	✓	✓	NA	-	
Product	Trialability		✓	✓	EL,B,Q	+	
Product	Complexity		✓	✓	EL,E,S,B,Q	-	
Product	Compatibility		✓	✓	EL,E,S,B	+	
Product	Relative Advantage	Functional	✓	✓	EL,E,S,B,Q	+	
Product	Relative Advantage	Economic	✓	✓	EL,E,S,B,Q	?	
Product	Relative Advantage	Emotional	✓	×	Not in any	NA	
Product	Whole Product Factor		✓	✓	EL,E,S,B	+	
Product	Specification Flexibility		✓	✓	E,S	+	
Vendor	HR Structure	DMU/ Stakeholders	×	✓	EL,B,Q	OT	
Vendor	HR Structure	Personnel/ Outsourced HR	×	✓*	EL,B,Q	OT	[22]
Vendor	Service		✓	✓	EL,B,Q	+	
Vendor	Business Model	Lock-in	✓	✓	EL,E,S,B,Q	+	
Vendor	Business Model	Novelty	✓	✓	EL,E,S,B,Q	+	
Vendor	Business Model	Efficiency	✓	✓	EL,E,S,B,Q	+	
Vendor	Business Model	Complemen-tarities	✓	✓	EL,E,S,B,Q	+	
Vendor	Marketing Strategy-4Ps	Price	✓	✓	EL,E,S	-	
Vendor	Marketing Strategy-4Ps	Promotion	✓	✓	EL,E,S,B,Q	+	
Vendor	Marketing Strategy-4Ps	Place	✓	✓	EL,E,S,B,Q	OT	
Vendor	Marketing Strategy-4Ps	Product	✓	*✓	EL,E,S,B,Q		

Table 4. Original vs. final research model variables (cont.)

Top Group	Variable Group	Variable	Original Model	Final Model	Case Studies	Nature of Impact/ Signi-ficance	Based on
Market and Industry	PEST	Political	✓	×	EL	NA	[19]
Market and Industry	PEST	Economic	✓	✓	EL,E,S,B,Q	+	
Market and Industry	PEST	Sociological	✓	×	Not in any	NA	[19]
Market and Industry	PEST	Technological	✓	✓	EL,E,S,B,Q	+	
Market and Industry	Industry Type		✓	✓	EL,E,S,B,Q	OT	
Adopting Organization	Type	Innovators	✓	✓	EL,E,S	+	
Adopting Organization	Type	Early Adopters	✓	✓	B,Q	+	
Adopting Organization	Type	Early Majority	✓	✓	NA	+/-	
Adopting Organization	Type	Late Majority	✓	✓	NA	-	
Adopting Organization	Type	Laggards	✓	✓	NA	-	
Adopting Organization	Users	Change Resistance	×	✓	E,B	-	[20]
Adopting Organization	Users	Profession	✓	✓	EL,E,S,B,Q	+	
Adopting Organization	Users	Position	✓	✓	EL,E,S,B,Q	+	
Adopting Organization	Users	Learning Curve	×	✓	E,B	+	[21]
Adopting Organization	Users	Outsourced HR*	×	✓	B,E	OT	[22]
Adopting Organization	DMU	Key Events	✓	✓	EL,S	OT	
Adopting Organization	DMU	Time	✓	✓	EL,E,S,B,Q	+	
Adopting Organization	DMU	Power Position of Employees	✓	✓	EL,E,S,B	+	
Adopting Organization	DMU	No. of Decision Makers involved in process	✓	✓	EL,E,B,Q	-	
3nd Party Integrator/ Consultant	Outsourced HR*		×	✓	B,E	OT	[22]

Table 5: Level of Lead-Driven Development implemented by vendors vs. Benefit in sales.

Site	LDD Level	Sale	2nd Sale	3nd Sale
Org.1-E	4	+	+	+
Org.2-S	3	+	+	-
Org.3-EL	5	+	not yet	not yet
Org.4-Q	1	-	-	-
Org.5-B	1	-	-	-

We entered the more comprehensive table figures of table 5 into a statistical software tool and found that the correlation between sales, second sales and third sales (i.e. vendor benefit) and the level of LDD implementation is clearly significant, positive and high.

After establishing that the use of LDD is beneficial for the vendor we carefully documented this process and generalized it over all the case studies. The description of the full LDD process now follows.

The hereunder elaborated emerging process for software development includes 12 main steps of which at least 4 include some form of improvisation. Moreover, the most unique phase of this suggested model, the lead gathering task is improvisation intensive. In addition, the model may serve as a strict continuous model similar to the Waterfall model or may be used as a Spiral model involving repetitive tasks.

At the core of this model, are 12 steps as follows:

Step 1: Initiation - This stage is a formal stage in the regular standard SDLC model. However, with entrepreneurial firms this step tends to be an informal one with no accurate start point in time. This stage includes structuring the will and intent to begin with the project and giving the go-ahead instruction as well as providing the limited resources necessary to start exploring the venture.

Step 2: High Level Concept Development - This second step includes forming the high level concept of the product, the problem which it comes to solve, its associated benefit etc. Depending on the scope of the project/system this development phase may be fulfilled using limited resources, spare time, and sometimes even academic resources. A substantial level of improvisation is used at this early stage as well - as part of the founding process.[23] Improvisation is carried out in the development of the suggested product in a quick and result-oriented fashion while using minor or no documentation and testing at all.

Step 3: Prototype - The first important milestone of the entrepreneurial vendor is the ability to deliver a functional prototype. The prototype should convey clearly the problem it is solving, its abilities and associated benefits in an easy and understandable manner with a friendly user interface. The number of moderate bugs, missing features as well as load balancing issues is not of much importance at all at this stage as the product shall be used mostly for demo and pilot purposes in the near future. This first prototype release is called by us a "Bugged Release". The prototype should further include a number of working examples from various domains.

Step 4: Minor Testing in Non Profit Environments, Academic Demo and Use - After completing the prototype, which should by this time be a powerful demonstration tool, the vendor should strive to demonstrate the tool in non-profit environments. The aim of these demonstrations is finding a limited installation bed for the product. These installations provide the developers with important feedback on bugs, missing features and general use of the tool - a preliminary focus feedback group for the tool. The academic scenery is extremely beneficial for these purposes as it also hosts great uncovered commercial potential through conventions, conferences and to a vast number of current or to be professionals. See also penetration attempts in academia by established firms like Philips, IBM and SAP.

Step 5: Market Introduction, Benefit Oriented Demonstrations and Mini Pilots - An additional task which is improvisation intensive is demonstrating the tool to potential customers. The suggested form of product demonstration which we call "benefit oriented demonstration" is a special type of marketing method unused so far in the world of software. The equivalent in the non-software world is that of a vacuum cleaner demonstration in the customer's home to show him immediate benefits of the product.[24] Thus, in this situation we suggest vendors demonstrate their new tools by implementing a form of improvisation at the customer's site. The vendor should use the tool to perform an on the spot real work task brought in by the customer for which neither the vendor nor the customer were prepared. If this session exceeds one meeting it may be considered as a mini pilot.

It is also in this step that the main improvisational task of the entire proposed model is undertaken. From meeting to meeting the vendor's marketing representatives must try to anticipate - using preliminary talks, phone conversations, social networking ties or emails - the needs of the potential customer as well as his existing environment. This highly informal improvisational task is used to build software requirements for the development coding team.

The requirements gathered are for features which will be required in the marketing sessions with the potential customers. These requirements are then addressed and coded immediately by the development team. The new features developed are neither documented nor tested thoroughly as they will be mainly used for demonstration purposes and may be ultimately dropped. However, a mentioning and documentation of the added features is required at least in a "What's New" file accompanying the product.

This step is of a strict repetitive improvisational nature and involves the gathering of lead requirements between marketing meetings and converting them into semi-operational software features.

Step 6: Offer OTS + HR = Project. After one of the leads materializes the vendor is asked to prepare a formal proposal for sale. Many of our case studies have indicated that OTS products for the professional organizational realm are rarely sold if they come from unknown vendors due to risk factors mainly of product abandoning. Hence, offering the OTS with Human Resources (an implementer) which will assist the adopting organization, prepare the initial material and then tutor its users is usually beneficial to reduce the uncertainty in these situations. After the customer agrees upon the terms of the project (and not only the product) contract engaging commences and the customer is now considered as the baseline customer. Hence, we call this sale the "first sale" en route to successful implementation.

Step 7: Bug fixing due to baseline customer requirements and marketing requests - After a certain amount of work is done using the tool at the customer's site, either by the customer's users or by the HR which was coupled to the OTS, important feedback regarding the products begins to accumulate. This information enables bug fixing and tool robustness improving. In addition, important missing features required by the baseline customer and marketing department are added to the software and provide for the First Commercial Release of the software. This release is still highly saturated by bugs but is already a commercial useable - "non-frustrating" - version of the product.

Step 8: Constrain Features, Further Commercial Releases and Support Plan - The unbridled adding of new features, in the format suggested in steps 5 and 7 above, creates overwhelming monstrous software. At this point the vendor should start to funnel out some features which cater to a smaller audience and which have not been found to be part of the vendor's targeted audience needs. Furthermore, the vendor should try to find a common thread or theme connecting and guiding all other features. This decision enables further product releases each containing additional noteworthy features, bugs correction and feature enhancements. With the continuous use of the product in at least one baseline customer and before the move to the next implementation step, a support plan (or plans) for the product should be created.

Step 9: Develop complexity management tools, Train integrators personnel, Interface with customer's software, Find additional benefit oriented projects, Embed within organizational deliverables - Within the baseline customer's everyday work, issues of model complexity very quickly arise. These issues, which are different from testing the product or load-balancing it, should be addressed and solved early on. In addition, this is also the time to deepen the roots within the customer by both trying to embed the product deliverables within the customer's overall deliverables, interfacing (physically) with the company's organizational ISs and by finding additional projects within the customer's company to be involved with. Deepening the vendor's role within the customer's site is a highly improvisational task in nature and hence requires adequate skills.

Step 10: Second Sale - Licenses: The first sale is by no means any indication of a successful implementation or diffusion of the innovation within the adopting organization.[25] Moreover, the coupling of the HR with the OTS does not really enable a real diffusion of the independent product. Hence, a second sale to the same organization marks an important future commitment of the organization to drop the tutoring relationship and proceed to license purchases for independent use. License purchases signify that the organization now associates positive benefit to the use of the product.

Step 11: Maintenance, Begin User Training, Tutoring, Software Support - After a second sale is made, the relationship between the vendor and customer moves into the maintenance phase. As the previous phase was coupled by HR this is where the customer will be taking his first steps with the software alone. These first steps include: formal user training sessions, one on one user tutoring sessions and general software support.

Step 12: Third Sale, Support - The earliest point one might consider as the point of successful diffusion or implementation of the innovation is, as seen in this study, the point of third sale. After the second sale, the customer independently used the software by himself, learnt the product's advantages and disadvantages and may now associate self benefit to product more accurately. Therefore, a third sale is the first true mark that the customer is truly realizing the benefits of the product and is preparing to use it in the long term. Support of the product continues now on an annual basis with milestones for new upgrades and releases.

Given that a vendor adopts this new 12 stage development process including and especially the improvisation-intensive stages, the question which should arise is how does the organization build and enhance the skills required for improvisation and what are these skills?

In the context of our study we identified three main factors which influence improvisational skills:

Teamwork skills - The ability of the entrepreneurial team to communicate with one another and relay timely information, get things done easily and quickly with no inhibitors and outside impeding factors.

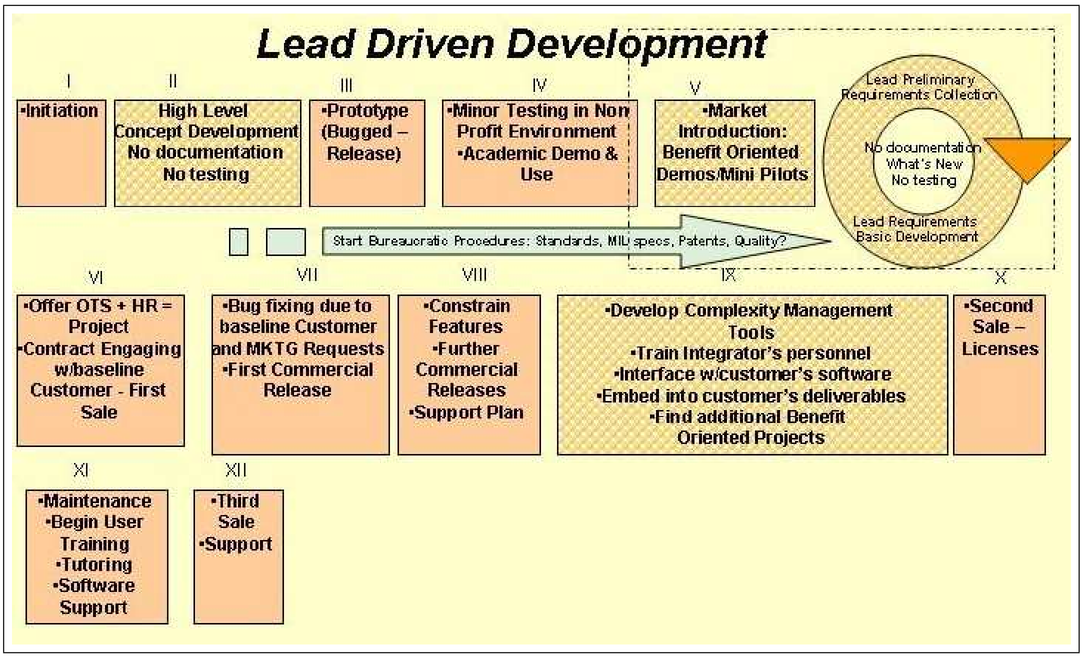


Figure 5: Lead Driven development: the 12 step timeline

Experience - The entrepreneurial team members' experience in similar circumstances and their memory to recall their right and wrong doings there.

Experimental culture - The culture of the team, which encourages trying out many a time risky and/or innovative solutions.

These three main factors, measured and calibrated according to characteristics described in [26], coincide with improvisational skills factors in the literature. For example, Vera and Crossan [27] create a theoretical framework based on improvisation and innovative performance in teams. Identifying variables from improvisational theatre, they tested the impact of the 16 different related variables on improvisational skills in an environment of a local municipality. They found 4 of the 16 variables to be of higher influence than others. The four factors they isolated were: expertise, teamwork skills, experimental culture and real-time information and communication.

In the context of our study we can therefore translate and apply their insights as follows: Expertise - Gaining a higher level of software expertise in the intricacies of the development environments enables software developers to find out of the sleeve solutions and bypasses for many software unpredicted difficulties encountered.

Teamwork skills - In general software teams, with a rather higher sense of collaboration usually tend to innovate more. To further clarify this point, we can propose the software teams the following teamwork skills which we encountered: development collaboration, information sharing via email, shared drives, knowledge management portals, inner group dynamics and communication etc.

Experimental culture - The experimental culture includes the ability to import new ideas and procedures from the World Wide Web, forums, groups and software development associates and try them out. Furthermore, experimentation in the software industry, which is not usually backed up by management should be backed up by top management and should also include experimentation on code developed using a number of alternate mechanisms.

Real-time information and communication - The need for real-time information and communication in the software industry is ever more compelling than any other industry. This is so because the software industry is built upon and relies heavily on the backbone of internet. Therefore, when improvising it is crucial to gain real-time updated information over the LAN or internet and have a variety of channels for communicating with the customer and other team members. Each of these channels specializes in a different type of content that may be passed: audio, video, documents, emails etc.

c. The Collaborative Multi-Tier System Development Life Cycle

The Lead-Driven Development (LDD) paradigm described above represents a list of tasks from the vendor's point of view. This list helped build the vendor's tier in the complete Collaborative Multi-Tier System Development Life-Cycle task matrix that caters to all stakeholders. The Collaborative Multi-Tier model takes into consideration, through the nature of the tasks suggested, pure IS development tasks (e.g., prototyping, bug fixing), market influences (e.g., market introduction techniques for entrepreneurial OTS software vendors, such as offering the first sale of such a product as a combined project with human resources), and organizational recommendations aimed mainly to avoid customer internal organizational obstacles. The Collaborative Multi-Tier System Development Life-Cycle (CMSDLC) model, which combines common tasks from the various case studies, is consistent as it avoids contradicting tasks. One of its unique features is that it makes a distinction between benefit-oriented tasks and standard waterfall-type formal tasks and milestones.

When a task or a set of tasks is performed in an iterative manner it is marked as an on-going task or specifically mentioned in the explanation section as one that needs to be performed iteratively. One such example is the task dealing with lead requirements gathering from potential customers for the purpose of prototype and feature building.

The CMSDLC model is also suitable for mature organizations that seek to develop a new OTS software product. It is even more suitable when the mature organization separates this entire operation from its existing core operations through various methods, such as founding a new subsidiary. This is akin to an entrepreneurial firm from the development and market perspectives. However, the financial backing of the parent organization and its reputation may shorten the duration of many of the tasks, and may make them more easily achievable. In cases where a mature organization seeks improvement to existing product development methods, the suggested model may not be as applicable, since the level of uncertainty such an organization encounters regarding market and organizational effects (especially regarding the customers), is lower.

The chart of the full model depicts in a single, poster style view the multi-tier model. The timeline of the system development lifecycle and the tasks for each stakeholder are stacked on separate horizontal lines. Since all the stakeholders share one timeline, the interaction between the stakeholders and the interdependency of tasks is clearly visible. Only through extensive collaboration can the effort proceed as a whole. For each task, a list of case studies which the task was based upon is provided. A list of one-letter abbreviations corresponding to the case study sites which demonstrated the use of the specific task appear below each task.

The logic behind gathering all the tasks into the model includes checking tasks for contradictions in the various case studies and merging similar tasks to unifying generic tasks.

6 Conclusion and Future Research

At this point we would like to introduce the hypotheses which were derived from the findings of our case studies. The hypotheses suggest a plethora of possible future research to be held on the process of validating them.

The first hypothesis which was emerging was based on research question 2. We found that the two parts of an OTS software product, the product and its underlying methodology, played an important role in all the case studies and therefore justified the hypothesis. We therefore phrased the first hypothesis as follows.

H1: *The more a customer is inclined to adopt a methodology, the more he or she is inclined to adopt the OTS software tool associated with that methodology.*

In other words, we will be looking for a possible correlation on a "bundling" relationship between the methodology and its supporting tool. We can explain this phenomenon by looking both at the marketing literature regarding characteristics of products and at the IS literature which explains and recommends various modes for IS product sale.

The marketing literature [28] and [12] and [29] defines clearly the "whole product concept" which was introduced by Theodore Levitt.[30] Levitt defined the whole product as follows: A product is, to the potential buyer, a complex cluster of value satisfactions.

The whole product factor denotes the completeness of the product being marketed at a certain point in time with regard to a complete solution. The components of the whole product include the core product, the tangible product, augmented product and total solution. The relationship between the OTS product used in our study and its methodology typifies the product as being close to an augmented product.

The second hypothesis involved a distinction between organizations using a formal software development procedure and others who use improvisation-intensive development techniques. As we started noticing in the case studies, this distinction is related to the level of uncertainty the vendor organization runs into. Hence we defined the following hypothesis:

H2: *The higher the level of uncertainty is, the higher the vendor's use of improvisation intensive development methods.*

Improvisation has been shown to be used by entrepreneurial vendors when faced by time pressure, complexity, and uncertainty.[31] Future research may extend this assertion to include and emphasize its relevance in the development process too.

The third hypothesis concerns the direct benefits associated with utilizing improvisation within the software development process, i.e., the time-to-market of the product and increased sales. The hypothesis was defined as follows.

H3: *When uncertainty is high, the more a vendor uses improvisation the more his market response time shortens and his ability to make a first sale improves.*

The organizational change resistance to new technologies in general and software in particular was found in our case studies to be solved by applying a marketing technique which couples a human resource implementer to the product.[32] The human resource implementer escorts the implementation and even performs most of the initial work for the customer using the tool. This triggered the following hypothesis:

H4: *The higher the level of HR participation in the OTS software sale attempt to the change resistant customer is, the higher are the chances for successful implementation.*

Similarly to the entrepreneurial vendor's efforts to overcome change resistance within the adopting organization users, the vendor has to build its legitimacy with the adopting organization's DMU.[33] We started noticing that this legitimacy buildup was being done via affiliation with accredited scientists/academics and/or through established third party integrators.

Stinchcombe [33] also specified three reasons for the impediments companies have from entering into a business relationship or buying a product from a new organization. He calls this effect the "Liability of Newness". The reasons cited for this liability are Lack of Experience, Lack of Size and Lack of Legitimacy. The latter was addressed by OPCAT, who built its legitimacy in all the case studies through the use of a reputable scientist to improve its lack of legitimacy and external reputation. Furthermore, some legitimacy build-up was achieved through the use of large third party integrator with proven reputation and experience in the industry.

This gave rise to the following hypothesis:

H5: *The more a new vendor firm affiliates with a distinguished scientist, or an established third party, the more the chances for successful implementation are higher.*

7 Summary and Recommendations

We have proposed and evaluated a software system development life cycle model which aims to improve successful system implementation and adoption by use of communication and collaboration amongst stakeholders. The new model for software development that emerged - Lead-Driven Development - was discovered, validated against the case studies, and refined via observations in five industry case studies and two additional control studies regarding successful implementations of OTS products of entrepreneurial developers.

The proposed Lead-Driven Development model accounts for market and organizational factors and the way they are woven into the traditional phases of software development. It offers the basis for the unified, comprehensive multi-tier SDLC framework and methodology that contributes to improved stakeholders' communication and collaboration through the use of a common reference model for all stakeholders. Each tier addresses a different force or stakeholder involved in the software market: vendor, customer, consultants and integrators.

The model is potentially beneficial for improving communication and collaboration among life cycle stakeholders in that it embeds action items from the IS, marketing and organizational realms. Many of these action items are performed using improvisational skills.

To excel in Lead-Driven Development in general, and in software development improvisation in particular, entrepreneurial vendors should enhance their improvisational skills. In line with previous studies [27], we found three main factors that influence improvisational communication skills: experience, teamwork skills, and experimental culture. Focusing on these facets, organizational training should provide a clear positive effect on improvisational skills and hence on innovation abilities.

Bibliography

[1] S. Sawyer, A market-based perspective on information systems development, *Communications of the ACM* (44:11), pp.97-102, 2001.

[2] E. Carmel, Cycle Time in Packaged Software Firms, *Journal of Product Innovation*, (12), pp.110-123, 1995.

[3] D. Dori, *Object Process Methodology*, Springer-Verlag, 2002.

[4] W. Royce, Managing the development of large software systems, In *Proceedings of IEEE WESCON*, pp.1-9, 1970.

[5] *TechTarget*, viewed on http://searchvb.techtarget.com/sDefinition/0,,sid8_gci755068,00.html, February, 2004.

[6] D. Howe (Ed.), *The Free On-line Dictionary of Computing*, http://foldoc.doc.ic.ac.uk/ foldoc/-foldoc.cgi? query=lifecycle, 1996.

[7] *GAO, GAO/IMTEC-8.1.4 Assessing Acquisition Risks*, http://www.gao.gov/ special.pubs/im814.pdf, 1992.

[8] B. Krogstie, and B. Bygstad, Cross-Community Collaboration and Learning in Customer-Driven Software Engineering Student Projects, *CSEET Proceedings of the 20th Conference on Software Engineering Education & Training*, 2007.

[9] E. Carmel, and S. Becker, A process model for packaged software development, *IEEE Transaction Engineering Management*(41: 5), pp.50-56, 1990.

[10] M. Cusumano, A. MacCormack, C. F. Kemerer, and B. Crandall, Software Development Worldwide: The State of the Practice, *IEEE Software*, pp.28-34, 2003.

[11] M. Keil, and E. Carmel, Customer-developer links in software development, *Communications of the ACM* (38:5), pp.33-44, May 1995.

[12] G. Moore, *Crossing the Chasm*, Harper Collins, 1999.

[13] R. K. Yin, The Abridged Version of Case Study Research: Design and Method, In L.Bickman and D. J. Rog *Handbook of Applied Social Research Methods*, Thousand Oaks, CA: Sage, pp.229-259, 1998.

[14] W. Tellis, Introduction to Case Study, *The Qualitative Report* (3:2) July 1997. (http://www.nova.edu/ssss/QR/QR3-2/tellis1.html)

[15] K. M. Eisenhardt, Building Theories from case Study research, *Academy of Management Review* (14:4), pp.532-550, 1989.

[16] R. Yin, *Case Study Research: Design and Methods*, Thousand Oaks, London, New Delhi: Sage, 1984.

[17] D. Dori, I. Reinhartz-Berger, and A. Sturm, OPCAT - A Bimodal CASE Tool for Object-Process Based System Development, *Proc. IEEE/ACM 5th International Conference on Enterprise Information Systems* (ICEIS 2003), École Supérieure d Électronique de l Ouest, Angers, France, pp.286-291, April 23-26, 2003.

[18] C. Typaldos, *Social Networking*, viewed on http://www.typaldos.com/, 2003.

[19] *PEST*, viewed on http://www.marketingteacher.com/Lessons/lesson_PEST.htm, November 2004.

[20] K. R. Stam, J. M. Stanton, I. R. and Guzman, Employee Resistance to Digital Information and Information Technology Change in a Social Service Agency: A Membership Category Approach, *Journal of Digital Information*, vol. 5 Issue 4, 2004.

[21] C. F. Kemerer, How the Learning Curve Affects CASE Tool Adoption, *IEEE Software*, v. 9, pp.23-8, 1992.

[22] M. Lacity, and L. Willcocks, Information Technology Sourcing Reflections, *Wirtschaftsinformatik*, Special Issue on Outsourcing, vol. 45(2), pp.115-125, 2003.

[23] T. Baker, A. S. Miner, and D. T. Eesley, Improvising firms: bricolage, account giving and improvisational competencies in the founding process, *Research Policy*, vol. 32, Issue 2, pp.255-276, 2003.

[24] *Vacuum marketing strategy*. Viewed on http://www.marketingsurvivalkit.com/advertising-sales-strategy.htm, October 2006.

[25] R. G. Fichman, and C. F. Kemerer, The illusory diffusion of innovation: An examination of assimilation gaps.Information Systems Research, *Information Systems Research* (10:3), pp. 255-275, 1999.

[26] S. Cohen, *A Multi-Tier System Development Life Cycle Model for Off-the-Shelf Software with Market and Organizational Effects. Ph.D. dissertation*, Faculty of Industrial Engineering and Management, Technion, Haifa, Israel, 2007.

[27] D. Vera, and M. Crossan, Improvisation and Innovative Performance in Teams, *Organization Science*, vol. 16, No. 3, pp. 203-224, May-June 2005.

[28] A. N. Alderman, Implementing the Whole Product Concept in Strategic Sector Marketing, *Proceedings of the Fifteenth Annual IEEE Applied Power Electronics Conference and Exposition*, New Orleans, LA, USA, pp.27-30, 2000.

[29] *JWR*, viewed on: http://jwr.strategictechnology.com/ pages/ChoosingSoftware.pdf, November 2005.

[30] T. Levitt, Marketing Intangible Products and Product Intangibles, *Harvard Business Review*, vol. 59 No. 3, pp.94-102, 1981.

[31] S. L. Brown, and K. M. Eisenhardt, The Art of Continuous Change: Linking Complexity Theory and Time-Paced Evolution in Relentlessly Shifting Organizations, *Administrative Science Quarterly*, vol. 42, No. 1, pp. 1-34, 1997.

[32] S&M, *How software firms spend their S&M dollars?*, http://www.softletter.com/PDFs/FHp10-p13.pdf, 2006.

[33] A. L. Stinchcombe, *Social structure and organizations In James G. March (ed.), Handbook of organizations*, Chicago: Rand McNally, 1965.

[34] S. Singh, and P. Kotze, An overview of systems design and development methodologies with regard to the involvement of users and other stakeholders, *Proceedings of SAICSIT*, pp.37-47, 2003.

[35] M. Cusumano, *The Business of Software: What Every Manager, Programmer, and Entrepreneur Must Know to Thrive and Survive in Good Times and Bad*, Free Press, 2004.

[36] K. B. Clark, and S. C. Wheelwright, *Managing New Product and Process Development: Text and Cases*, New York: Free Press, 1993.

[37] K. B. Clark, and S. C. Wheelwright, *The Product Development Challenge: Competing through Speed, Quality, and Creativity*, Boston: Harvard Business School Press, 1995.

[38] S. C. Wheelwright, and K. B. Clark, Accelerating the Design-Build-Test Cycle for Effective Product Development, *International Marketing Review* (11:1), pp.32-46, 1994.

[39] B. Boehm, and P. Bose, A Collaborative Spiral Software Process Model Based on Theory W, *Proceedings, 3rd International Conference on the Software Process, Applying the Software Process*, IEEE, 1994.

[40] C. H. Fine, *Clockspeed: Winning Industry Control in the Age of Temporary Advantage*, Perseus Publishing, 1999.

[41] D. Avison, and G. Fitzgerald, Where Now for Development Methodologies?, *Communications of the ACM*, vol. 46, no. 1, pp.78-82, 2003.

[42] C. Ebert, Requirements Engineering: Understanding the Product Life Cycle: Four Key Requirements Engineering Techniques, *IEEE Software*, pp.19-25, May/June 2006.

[43] E. Mustonen-Ollila and K. Lyytinen, Why organizations adopt information system process innovations: a longitudinal study using Diffusion of Innovation theory, *Information Systems Journal* (13:3), pp.275-297, July 2003.

[44] H. C. Jr. Lucas, and V. Spitler, Technology Acceptance and Performance: A Field Study of Broker Workstations, *Decision Sciences Journal*, (30:2) 1999.

[45] F. D. Davis, Perceived Usefulness, Perceived Ease of Use, and User Acceptance of Information Technology, *MIS Quarterly*, pp.319-339, 1989.

[46] G. C. Moore, and I. Benbasat, Development of an Instrument to Measure the Perceptions of Adopting an Information Technology Innovation, *Information Systems Research* (2:3), pp.192-220, September 1991.

[47] J. C. Brancheau, and J. C. Wetherbe, The Adoption of Spreadsheet Software: Testing Innovation Diffusion Theory in the Context of End-User Computing, *Information Systems Research* (1:2), pp.115-142, 1990.

[48] R. B. Cooper, and R. W. Zmud, Information Technology Implementation Research: A technological Diffusion Approach, *Management Science* (36:2), pp.123-139, 1990.

[49] J. Verville, and A. Halingten, An investigation of the decision making process for selecting an ERP software: the case of ESC, *Management Decision* (40:3), pp.206-216, 2002.

[50] P. Nelson, W. Richmon, and A. Seidmann, Two Dimensions of Software Acquisition, *Communications of the ACM*, (39:7), pp.29-35, 1996.

[51] J. Iivari, and I. Ervasti, The impact of alternative IS acquisition options upon the IS implementation and success, in *Proceedings of the 1992 ACM SIGCPR conference on Computer personnel research*, Cincinnati, Ohio, United States, pp.338 - 349.

[52] M. C. Paulk, Structured Approaches to managing Change, *Crosstalk: The Journal of Defense Software Engineering* (12:11) November, pp.4-7, 1999.

[53] Nordman, *Commercialization Success in Early Stage Technology Companies*, viewed on www.rocketbuilders.com/ commercialization/ RB_Commercialization_Presentation_Jun2004.pdf, 2004.

[54] Y. Lee, and G. Colarelli O'Connor, New Product Launch Strategy for Network Effects Products, *Journal of the Academy of Marketing Science*, (31:3), pp.241-255, 2003.

[55] E. Montaguti, S. Kuester, and T. S. Robertson, Entry Strategy for radical product innovations: A conceptual model and propositional inventory, *International Journal of Research in Marketing*, (19), pp.21-42, 2002.

[56] S. Shane, and S. Venkataraman, The promise of entrepreneurship as a field of research, *Academy of Management Review*, vol. 25, No. 1, pp.217-226, 2000.

[57] S. Shane, *Entrepreneurship: A Process Perspective*, South-Western College Pub; 1st edition, January 2004.

[58] F. Murray, and M. Tripsas, The Exploratory Process of Entrepreneurial Firms: The role of purposeful experimentation, *Advances in strategic management*, vol. 21, pp.45-75, 2004.

[59] *NIH System Development Life Cycle (SDLC) IT Security Activities Matrix*, viewed on http://irm.cit.nih.gov/ security/nih-sdlc.html, December 2006.

[60] DOJ, *The Department of Justice Systems Development Life Cycle Guidance Document*, viewed on http://www.usdoj.gov/ jmd/irm/lifecycle/table.htm, January 2003.

[61] *Agile software development: a definition from Whatis.com* - Viewed on:http://whatis.techtarget.com/ definition/0,,sid9_gci936457,00.html, November 2006.

Shalom Cohen (b. December 16, 1973) completed his MSc in Operations Research at Tel Aviv University, Israel and his PhD in the Faculty of Industrial Engineering and Management at the Technion, Israel. In addition to Lecturing at the Technion on Information System related topics Shalom holds the position of Chief System Architect in a high tech company in the homeland security sector.

Dov Dori (b. September 2, 1953) is information and systems engineering Professor at the Faculty of Industrial Engineering and Management, Technion, Israel, and Research Affiliate with the Engineering Systems Division at Massachusetts Institute of Technology. He received his BSc in Industrial Engineering and Management from Technion in 1975, MSc. in Operations Research from Tel Aviv University in 1982 and PhD in Computer Science from Weizmann Institute of Science in 1988.

Uzi de Haan (b. September 5, 1943) completed his MSc in Aeronautical Engineering at the University of Delft, Holland and his PhD in the Faculty of Industrial Engineering and Management at the Technion, Israel. He joined the Technion as a professor in the area of Strategic Management and Entrepreneurship at the Faculty of Industrial Engineering after many years in the high-tech industry.

Int. J. of Computers, Communications & Control, ISSN 1841-9836, E-ISSN 1841-9844
Vol. V (2010), No. 1, pp. 42-51

Mining Authoritativeness of Collaborative Innovation Partners

J. Engler, A. Kusiak

Joseph Engler
Adaptive Systems Rockwell Collins, Inc.
Cedar Rapids, IA 52498 USA
E-mail cjengler1@msn.com

Andrew Kusiak
Mechanical and Industrial Engineering
3131 Seamans Center University of Iowa
Iowa City, IA 52242-1527 USA
E-mail: andrew-kusiak@uiowa.edu

Abstract:
The global marketplace over the past decade has called for innovative products and cost reduction. This perplexing duality has led companies to seek external collaborations to effectively deliver innovative products to market. External collaboration often leads to innovation at reduced research and development expenditure. This is especially true of companies which find the most authoritative entity (usually a company or even a person) to work with. Authoritativeness accelerates development and research-to-product transformation due to the inherent knowledge of the authoritative entity. This paper offers a novel approach to automatically determine the authoritativeness of entities for collaboration. This approach automatically discovers an authoritative entity in a domain of interest. The methodology presented utilizes web mining, text mining, and generation of an authoritativeness metric. The concepts discussed in the paper are illustrated with a case study of mining the authoritativeness of collaboration partners for microelectromechanical systems (MEMS).
Keywords: Innovation, web mining, text mining.

1 Introduction

Innovation most often occurs in one of two forms, incremental or radical. Radical (discontinuous) innovation assumes focuses on a completely new concept that is radically different from the existing ideas. This type of innovation occurs rarely and is not easily predictable. Incremental, or continuous, innovation builds upon previous concepts and therefore it is easier to be quantified. Kusiak [1] defined innovation as an iterative process aimed at the creation of new products, processes, knowledge or services by the use of new or even existing technology. This definition summarizes the typology of incremental innovation.

Innovation has further been quantified into five generational models. The first generational model is linear, where innovation is unidirectionally pushed from the research phase to the commercial application phase [2], [3]. The second model, the pull model, holds the consumer as the main focus of innovation as opposed to the designer [4]. Feedback forms the third model and utilizes the consumer's responses to an initial product/service offering to perform incremental innovation on that product and/or service [5]. The fourth model is known as the strategic model in which innovation lines up directly with the company's strategy [2].

The model pursued in this paper is the fifth model, also known as the networked model. In this model extra-enterprise and cross-discipline organizations form a network to innovate. The term Open Innovation [6] is often used when describing this model. Collaborative networking involves a detection of the optimal sources of collaboration. This can often be viewed as a challenge and often results in the local optimum as the choice of the collaborative source rather than the unseen, and most often unknown, global optima (e.g., the most authoritative person/company).

The success of collaborative innovation depends greatly on the quality of the collaboration sources. The probability of innovation success, as measured by the market, can be considered proportional to the quality of the collaborative sources. Therefore, it is incumbent upon a company, in pursuit of innovation, to seek out the optimal sources for collaboration. A wealth of information is available upon the World Wide Web (WWW) for identifying the optimal sources of collaboration (e.g., white-papers written by authorities of specific domains).

Various researchers have begun to investigate possible means by which collaboration produces effective results. Unfortunately, the literature is lacking solid systematic methodologies by which collaboration authoritativeness may be determined apriori of collaboration inception. Chapman et al. [22] proposed a process model for collaborative data mining in an electronic manner but fail to address the need for authoritativeness when determining the collaborative partner selection process. Lavraĉ et al. [23] surveyed various methodologies for collaboration but fail to report a systematic methodology of determining the most authoritative entity for collaboration. Gajda [24] investigated assessment measures for determining the success of a collaborative partnership on a project. Unfortunately, these measures are considered upon the completion of the project rather than to determine the effective partners for collaboration prior to entering in to a collaborative agreement.

Other researchers have posited criteria and methodologies for collaboration partner selection. Geringer [25] proposed task-related selection criteria for international joint ventures but failed to present a systematic methodology for automatically determining the authoritativeness of a collaborative partner. Hitt et al. [26] investigated resource-based and organizational learning for collaborative partner selection. Again, the methodologies in the literature fall short of the goal of automated systematic determination of authoritativeness for collaboration.

This paper presents machine learning algorithms to extract collaborative innovation relationship information from various sources including the WWW. Utilizing machine learning algorithms to discover valuable knowledge from disparate sources has been presented in the literature. Chen [27] posited utilizing natural computing techniques, such as swarm intelligence, to foster a collective intelligence in a virtual learning environment. Grebla et al. [28] presented a Bayesian belief network for mining data from various databases to assist in predicting arteriosclerosis and cardiovascular disease.

This paper offers a novel methodology by which the optimal authoritative sources of collaborative partnership may be discovered. Through the use of web mining, text mining, and the creation of an authoritativeness matrix, users may determine the optimal authority with which to perform collaborative innovation. Of course, optimality may depend upon more than just the most authoritative partner on a given subject. Other factors such as the availability of the collaborator or the cultural background of the collaborator (e.g., defense systems collaboration) may be involved. Thus, this paper advocates the creation of an authoritativeness matrix as opposed to simply defining global optima for collaboration.

The remainder of this paper proceeds as follows. Section 2 discusses the focused mining of the World Wide Web to discover authoritative sources for collaboration. The distillation of these sources to form an authoritativeness matrix is discussed in Section 3. An authoritativeness metric is presented in Section 4. Section 5 discusses clustering of the mined sources of collaboration. Section 6 offers a case study for determining the leading authorities on microelectromechanical systems (MEMS). Finally, Section 7 offers concluding remarks.

2 Focused Mining of the Web

The first major step in forming a collaborative innovation relationship is to seek out and choose partners for the collaboration process. The World Wide Web (WWW) presents a proven search space for multiple concepts. A natural inclination is to manually search the internet for such sources of collaboration. Some companies hire business development teams to perform this task. Manually searching the web is a time intense process that often yields sub-optimal results. Sometimes searches can even present misguided or influenced results due to the ability for parties to influence their rank among the various search engines [7].

Increasing the difficulty of the manual search, many search engines utilize payer based rankings which facilitate assigning a higher position in the search results. Additionally, many websites have multiple internal links thus boosting certain search engine ratings. Such forms of ranking manipulation may provide false results and thus the most suitable collaborative candidates could be missed.

To overcome the limitations of a manual search of the internet for collaboration resources, a focused web miner is presented. The focused web miner used for this process includes user inputs of a specified phrase which become the search criteria. The focused web miner then proceeds, following standard focused web crawling methodologies as presented by Liu [8], to traverse the WWW in search of white papers, articles, and journal entries related to the search criteria. The presented web miner can easily be extended to handle other sources, e.g., information about companies which have reached Phase II funding from Small Business Innovation Research (SBIR) programs.

The presented version of the focused web miner does not attempt to mine the web blog data or the standard html files. Rather, this version of the focused web miner seeks content mimicking academic writings. Thus, the focused web miner spends a fair amount of time searching academic web sites, scientific communities, and trade journals. It is from these types of internet resources that the, often academic, writings are extracted.

The discovery of web pages containing white papers is a significant task involving crawling the internet and classifying the web pages that are examined as a review or non-review page. The standard approach to performing the crawl is to utilize a focused web crawler. A focused web crawler targets a specific corpus of web pages. Standard crawling does not consider a specific topic of inquiry; rather its job is to index all pages available on the internet.

Even with the assistance of algorithms such as PageRank developed by Google [17], successful standard crawling requires massive hardware and bandwidth. This drawback prevents most corporations from performing this type of crawl internally. Focused crawling requires far less hardware and bandwidth but does require some sophistication of algorithms to weed out the undesirable links as they relate to the given query. The algorithms useful to focused web crawling involve basic classification algorithms.

There is a great variety of classification algorithms for determining web pages containing white papers. Shih et al. [18] suggested the use of web page content structure as parameters of classification. The authors in [18] indicated that content providers tend to choose URLs and page layouts that coherently structure their content. This html structure may be useful in determining the likelihood that a web page contains reviews. Kules et al. [19] extended this idea to limit the features used for classification to items such as web page titles, URLs and text snippets. Jin et al. [20] took a different approach from the previous two and utilized a data-mining algorithm called Hidden Nad've Bayes. Their methodology considered a large corpus of web pages and calculated the probability that each page fit into a particular category.

Fortunately for the focused web miner proposed here, document type is often the best indicator of a possible fit. Given the query string provided by the user and a set of allowable document types (e.g., pdf files only) the need for classification is reduced greatly. The need for classification increases dramatically when white papers other than standard academic content is searched. In such case the use of a combination of Bayesian classification and the web page structure algorithm of [18] is suggested (see [21]).

Once the focused web miner has discovered a document of the standard academic writing, it attempts to download this document to a central repository that is dedicated to the search criteria. This repository often becomes extremely large, in the range of a terabyte or more, but is central to the process for collaboration resource detection. No partitioning of this repository takes place at the time of download; rather, all documents are placed into the same location. It is upon these documents that the process of determining the most authoritative collaboration partners is performed. While this form of repository may seem excessive, the reader is reminded of the low cost of data storage. Additionally, a great deal of information will be gleaned from this repository over time.

3 Authoritativeness Matrix

An authoritativeness matrix is generated from the documents that were obtained from the focused web mining process. Utilizing standard text mining techniques the documents are deconstructed to gain the information necessary for the generation of the authoritativeness matrix. A text-mining algorithm extracts from each document the author's names and the references, other authors, cited by that particular paper.

To assist in discovery of the authors who wrote, and are cited in, the papers a list of first names is utilized. The list of first names, freely available on many internet sites, allows for the detection of document patterns within the corpus such that most often names of authors are placed within a given context within the document (e.g., author names at the beginning, authors who are cited at the end). The text mining algorithm utilizes these patterns to classify portions of the document which will have the information extracted from. Additional, text-mining algorithms may be utilized to detect the sense, positive or negative, in which a citation appears within the document.

From the information mined by the text-mining algorithm the authoritativeness matrix may be constructed. The authoritativeness matrix is a two dimensional matrix, or table, made up of columns representing individuals who have been referenced by the papers and rows representing paper authors. The authoritativeness matrix forms a concise but sparsely populated representation of the given, or presented, authorities in the documents.

Figure 1 presents an example authoritativeness matrix for three documents. The rows represent the authors of the documents while the columns represent authorities who are cited as references in these documents. A cell of the matrix is 1, if the author in the row containing the cell has referenced an authority in the column containing the cell. Otherwise the cell is 0. Each row represents a single author for a single paper, thus there may exist multiple rows for a single paper.

From the example of Figure 1 it is easily determined that JR Koza is the most authoritative person with which to conduct collaborative innovation for the example domain. This is due to the fact that JR Koza is the most cited

	Howit, P.	Benkler, Y.	Stokic, D.	Von Hippel, E.	Nolan, R.L.	Hansen M.T.	Koza, J.R.	Document year
Kusiak, A.	1	1	0	1	1	0	1	2006
Lin, G.	0	1	0	0	1	1	1	2004
Stokic, D.	1	0	1	0	0	0	1	1999

Figure 1: Example authoritativeness matrix for three documents

author within the tiny corpus of documents for the example. It will be demonstrated later in this section that other factors contribute to this outcome.

It should be noted that an author may reference his/her own writings as well as be referenced by others. Thus, it is possible for this matrix to hold entities that are in both the rows and columns of the matrix. In fact, it is possible, although highly improbable that the authoritativeness matrix holds exactly the same authors in its rows as it does references in its columns. As will be explained in Section 4, there some caution needs to be exercised when an author often references their own work while others do not. This caution is the motivation for the storage of the document year in the authoritativeness matrix as will be explained.

The representation of the authorities of the documents discovered by the focused web miner by the authoritativeness matrix ensures ease of storage and traversal. The authoritativeness matrix is compact enough to be stored in main memory especially given the sparseness of the matrix. This allows for efficient processing when determining the true authorities for the collaboration process as presented next.

4 Authoritativeness Metric

To determine which entities in the authoritativeness matrix represent the optimal, or most, authoritative entity, within the search criteria, an authoritativeness metric is used. This metric accounts for the number of documents which were written by the authority, the number of times the authority is referenced in the body of work, discovered by the focused web miner, as well as the average age of the documents written by the authority. Additionally, the sense, positive or negative, in which the author is portrayed in the document can be collected.

Thus, the first step in calculating the authoritativeness metric is to scan the authoritativeness matrix and calculate a number of measures. These measures will be stored in various hash tables for efficient referencing. During the scan of the authoritativeness matrix a hash table representing the authors, or rows, of the matrix is created. Each time an author is encountered in the scan of the matrix that author is either added to the hash table as a key value pair of <AuthorName, 1> or the value of the index of the author in the hash table is incremented. The same action is performed for the columns, or referenced authorities, in the matrix. Additionally, a hash table is created for the purpose of obtaining the average age of the documents for each author, or row, in the matrix. The use of the three hash tables makes for an efficient scanning methodology for the authoritativeness matrix since the matrix is scanned only a single time. A forth table may be required to represent sense.

The hash table which represents the number of documents written by the authors, or rows, of the authoritativeness matrix is used to obtain the number of out-links of each author. Out-links are documents written by the author. Similarly, the hash representing the number of times an author is referenced is used to obtain the number of in-links for each authority. In-links are documents written by other authors referencing the given author, thus implicitly conveying authority on, or detracting from in the case of a negative sense, the referenced author.

The conveyance of authority on an author by referencing their work is held here in a context similar to that of the PageRank algorithm discussed in [9]. Conveyance of authority plays a vital role in the determination of authoritative collaborative sources. Thus, the authoritativeness metric proposed weights the in-link measure higher than the out-link measure. Additionally, should sense be included, the in-link could have the ability to decrease the author's authoritativeness.

The initial authoritativeness metric is defined in (1). In (1), λ is a user defined parameter which allows for changing the weight of the out-links based on the average age of the documents written by author a_i. This parameter assists in controlling the conveyance of authority to an author who is a prolific writer but perhaps not often cited by other authors. Additionally, the λ parameter allows for decreasing the weight of older documents if only recent documents are desired. If sense is utilized, the in-link, in_i, can be a negative number. It should be noted that determination of the final authoritativeness metric measure is an iterative process as will be explained.

$$A_i = \ln(\lambda^{t'}(out_i) + in_i) \tag{1}$$

where:
out_i is the number of out-links;
in_i is the number of in-links;
t' is the average age of the documents of author a_i in years from the current year;
λ is a user parameter in the range of [0, 1].

Thus, the initial authoritativeness of an author ai is given by A_i. Once the initial individual authoritativeness metrics of the entire authoritativeness matrix is calculated, the iterative process of boosting the authoritativeness is performed. Similar to the methodology used in the PageRank algorithm [10], it is desirable to instill more authority to an author who is referenced by another author of high authority. Thus, if one author is considered a leading authority, deemed so by the authoritativeness metric, and that author references a second author, the second author's authoritativeness metric measure should be increased.

The iterative process of authoritativeness boosting is performed using the average of the in-links pointing to the current author. In-links of high authority contribute to the boosting of the current author's authoritativeness, while in-links of less authoritative authors are not detrimental to the current author's authoritativeness. Thus the average of the in-links during the authoritativeness boosting process is calculated by including the authoritativeness of the author who referenced the current author as shown in (2).

$$\overline{in_i} = \sum_{i=1}^{N} \frac{e^A}{N} \tag{2}$$

where:
A_i is the authoritativeness of author a_i;
N is the number of in-links to a_i.

At each iteration of the calculation of the final authoritativeness metric the average of the in-links is used to calculate the new authoritativeness metric measure A_i for each author a_i. Equation (3) describes how the new in-link measure is calculated iteratively.

$$in_i' = \sum_{j-1}^{N} \begin{cases} e^A - \overline{in_i} & \text{if } e^A > \overline{in_i} \\ 1 & \text{if } e^A \leq \overline{in_i} \end{cases} \tag{3}$$

Thus, at each iteration of the boosting process, ini of (1) is replaced with in_i for the calculation of the authoritativeness of each author. The boosting process is continued for n iterations, as set by the user, or until the order of the authorities remains unchanged which is the preferred method.

With the final authoritativeness metric at hand for each author it is easy to determine which author and/or entity is the most authoritative in the subject matter of the search criteria. It is useful to determine the top k authorities in the subject matter to ensure that a good collaborative resource can be found available and willing to collaborate on the innovation at hand. As such it is useful to set a user defined parameter λ which is a threshold below which authoritativeness is discounted. This threshold is utilized in determining the authorities of the clustered documents as explained next.

5 Document Clustering

Often the size of the search space or the generality of the search criteria can result in a document set of varied type which is large in size. To ensure that the collaborative partner chosen by the authoritativeness metric is the one that is most appropriate for the specific collaboration it is helpful to cluster the documents into similar categories. Once the clustering has been performed, a cluster that is most similar to, or most represents, the specific innovation topic is chosen. From that cluster it is the possible to determine the best collaborative source for the innovation. Note, the collaborative authority of a specific cluster may not be the authority whose overall authoritativeness metric is the highest. Rather, the cluster authority, or authorities, will be those who are most advantageous for the specific innovation subject.

Clustering of the documents mined via the focused crawler begins with the generation of a word frequency matrix for the documents. The word frequency matrix represents the counts of each word in the individual documents. Each row of the matrix represents a single document; while each column of the matrix represents a single word. There exists columns for every document word, which is not a stop word, thus the matrix can be somewhat sparse. Many words, known as stop words, do not assist in properly classifying the documents. Stop words are most common in everyday language and thus not specific to the topic. Words such as "the", "in" and "here" are removed from the word frequency matrix prior to clustering. Further, it is often favorable to generate the root of words as opposed to the actual words for this frequency matrix. Thus, words such as "innovation", "innovate", and "innovativeness" would all be placed in the root word frequency cell for the word "innov". Figure 2 below represents a partial frequency word matrix.

	Ability	Able	Absolute	Abstract	Accelerate	Accept
00006009.pdf	1	5	0	0	0	4
00104286.pdf	0	0	0	0	0	0
00183750.pdf	2	0	2	1	0	0
00263768.pdf	1	0	2	0	0	0
00267882.pdf	0	0	0	1	0	0
00522535.pdf	1	0	0	1	21	0
00540568.pdf	0	0	1	1	0	0
00608091.pdf	0	0	0	0	1	1
00653292.pdf	0	1	0	1	0	2

Figure 2: Word frequency matrix

Once the word frequency matrix is obtained it is important to reduce the dimension of the matrix to ensure efficient clustering. Dimensionality reduction techniques, such as singular value decomposition, that are used for standard data mining are especially helpful here. The word frequency matrix before dimensionality reduction can easily include thousands of words or attributes. Rarely are all the attributes of value to the clustering. Thus, by performing a dimensionality technique such as singular value decomposition, the attribute set can be reduced down to a size that is more manageable, typically of size 100 or less [11].

Once the dimensionality reduction has been performed, the reduced word frequency matrix is clustered with simple k-means clustering algorithm described in [11]. Thus, a brief review of the cluster centroids will help to determine which cluster most resembles the subject matter of the specified innovation.

The authoritative collaboration partner(s) can easily be determined from those entities that have contributed work to the cluster that most resembles the subject of the innovation. Section 4 presented a threshold measure by which authorities could be weeded out of the collaborative search process. Following the Apriori Property discussed in [11] and [12], those authorities that are not authoritative for the entire group should not be considered authoritative for a subsection of that group. Therefore, only authorities with the authoritativeness metric higher than the user defined threshold should be sought within the clusters.

6 MEMS Case Study

This section presents a case study on the discovery of the most authoritative person to perform collaborative innovation with for the domain of microelectromechanical systems (MEMS). In this study 2403 papers were mined from the internet on the subject of MEMS

Simon [13] describes MEMS as a monolithically integrated device used for microwave applications such as switches, distributed phase shifters and BPSK modulators. Other applications for MEMS have also surfaced. In fact, according to Maeda et al. [14] MEMS is expected to be one of the most promising areas of research and development contributing to future success of electronics businesses.

After the 2403 papers were mined from the internet using the focused web miner, the author's names and references were parsed from the documents as described in Section 3 above and the authoritativeness matrix was generated. The authoritativeness metric, described in Section 4, was applied with λ set to 0.80 to slightly discount the average age of the documents. Figure 3 illustrates the top ten authorities after this initial calculation of authoritativeness. Figure 3 illustrates the results of running the algorithms presented in this paper prior to the iterative boosting discussed in Section 4. Thus, the results in Figure 3 are more indicative of a rapid manual search.

Author	Authoritativeness Metric
GM Rebeiz	4.5218
S Eshelman	4.1897
CL Goldsmith	4.0073
R Langer	3.8286
TA Desai	3.8067
A Malczewski	3.7136
MJ Cima	3.5835
DJ Beebe	3.5553
B Pillans	3.5553
J Ehmke	3.5264
JB Muldavin	3.4965

Figure 3: The top 10 authors in the non-boosted authoritativeness metric list for MEMS

As seen in Figure 3, GM Rebeiz is indicated as the leading authority on MEMS. A quick search of the internet with the name GM Rebeiz justifies his rank as the top authority in this non-boosted list. GM Rebeiz is a professor at the University of Michigan in the College of Engineering and leads a team of 8 PhD students in a focus on RF-MEMS [15].

Utilizing the authoritativeness matrix, discovered in the mining of the 2403 documents which generated the non-boosted results of Figure 3, the iterative boosting of authoritativeness is applied. Upon the application of the boosting of the authoritativeness the list changes in order as can be seen in Figure 4. Boosting has the effect of attributing higher authority to those whose papers have been cited by authors of higher authority. Thus, this is the list that a person for collaborative practices in the field of MEMS should be sought from.

Author	Authoritativeness Metric
CL Goldsmith	19.1892
M Sarantos	18.1891
H Fudem	18.1891
B Pillans	18.1849
S Eshelman	17.9906
RF Lohr	17.6873
D Strack	17.67790
F Kuss	17.6779
E Niehenke	17.6779
E A Sovero	17.4917

Figure 4: Top 10 authors in the boosted authoritativeness metric list for MEMS

From the boosted authoritativeness it is easy to see that CL Goldsmith is the authoritative figure one would wish to collaborate with. In fact, with a quick search of the internet it is found that CL Goldsmith is the president of a company called Memtronics and received his PhD from the University of Texas [16]. The list contains other potential candidates who may be sought after should CL Goldsmith not be available for collaboration.

For this case study, the focused web miner ran for approximately 18 hours to gather the 2403 documents. The parsing of the author's references and document age took less than 2 minutes. The initial authoritativeness was then calculated from the matrix in approximately 1.5 minutes. The boosting of the authoritativeness took 16 iterations before order no longer changed and took less than 10 minutes to achieve (see Figure 5). Thus, overall, the process of mining the leading authority in the field of MEMS, based upon these documents, took less than 18.5 hours and required very little of the user's time to perform. It is easily seen that this is a marked improvement upon a manual search.

Activity	Time/Quantity
White papers mined	2403
Focused miner runtime	18 hours
Parse authors, references, age	2 minutes
Initial authoritativeness calculation	90 seconds
Number of boosting iterations	16
Boosted authoritativeness calculation	10 minutes
Total algorithm run time	18 hours 13.5 minutes

Figure 5: Summary of the algorithm run

The case study illustrates the effectiveness of the authoritativeness metric presented in this paper. Further, the case study highlights the differences between boosted and non-boosted authoritativeness. From the perspective of a company that is seeking a collaborative partner, the boosted authoritativeness offers a list of highly respected candidates. Deriving this list in the short unmanned time frame of 18.5 hours offers companies a great benefit in discovering the most authoritative person(s) to perform collaborative innovation with.

The effectiveness of the authoritativeness metric is further explained in a recent scenario that was encountered by an electronics manufacturer who required expertise with legacy 16 bit PCMCIA PC cards. Due to confidentiality the details of this episode cannot be related although a summary of the scenario can be provided. The electronics manufacturer, a government contractor, was contracted to design a laptop integrated testing device for a piece of electronic equipment for a foreign concern. One of the requirements for this testing device was for it to integrate with the laptop through a legacy 16 bit PCMCIA PC card. The contractor lacked the domain knowledge to effectively and rapidly design the testing device with this form of legacy interface. Therefore, the author's were asked to apply the authoritativeness metric to determine which entities to best collaborate with on this issue. The results of the running of the algorithms, described herein, a list of authoritative entities was generated. The second entity on this list was eventually utilized to solve the domain issue. The first entity on the list was not completely suited for the task due to security restriction.

The above presented scenario lends additional support towards the effectiveness of the authoritativeness metric. It is shown that the authoritativeness metric is applicable not only to academic, research, and scientific activities but also to integration of various domain expertise in a corporate setting as well. Further, it is shown that the list of authoritative entities is crucial for selection of collaborative partners due to various external constraints (e.g., security, geospatial reasons) that cannot be accounted for within the authoritativeness metric. By producing a list of authoritative entities the end user is capable of filtering for these external constraints while still achieving the results of finding the optimal collaborative partner.

7 Conclusions and Future Works

Open innovation is the means by which companies seek external entities with which to collaborate to form innovation. This paper illustrated that finding the best source of collaboration for a given innovation in a manual fashion is sub-optimal. This paper presented a novel methodology for the automation of collaboration partner detection for the purpose of collaborative innovation. Furthermore, a process by which the authoritativeness of the collaborative partner is ensured to be optimal was presented. Using data mining, clustering, and analysis of the documents related to the innovation domain increases competitiveness of companies.

Novel to this paper is the use of boosted authoritativeness. The iterative process of increasing, or decreasing the authoritativeness of possible candidates for collaborative innovation extends the search process, and represents an automated methodology for determining the best candidate entity (company, person) for collaborative innovation. Future research should includes increasing the efficiency of document detection during the web mining process as well as increasing the rate at which document classification takes place.

Bibliography

[1] A. Kusiak, Innovation: A Data-Driven Approach, *International Journal of Production Economics*, Vol. 122, No. 1, pp. 440-448, 2009.

[2] G. Berkhout, P. van der Duin, Mobile Data Innovation: Lucio and the Cyclic Innovation Model, *Proc. of the 6th Intl. Conf. on Electorinc Commerce*, Delft, Netherlands, pp. 603-608, 2004.

[3] Y. Sawatani, F. Nakarmura, A. Sakakibara, M. Hoshi, S. Masuda, Innovation Patterns, *Proc. of the 2007 IEEE Intl. Conf. on Services and Computing*, Salt Lake City, UT, pp. 427-434, July 2007.

[4] J.B. Zhang, Y. Tao,The Interaction Based Innovation Process of Architectural Design Service, *Industrial Engineering and Engineering Management 2007 IEEE Intl. Conf.*, pp.1719 - 1723, Dec. 2007.

[5] A.W. Ulwick, Turn Customer Input Into Innovation. *Harvard Business Review*, Vol. 80, No. 1, pp. 91-97, 2002.

[6] L. Collins, Opening up the Innovation Process, *Engineering Management Journal*, Vol. 16, No. 1, pp. 14-17, 2006.

[7] A. Langville, C. Meyer, Deeper Inside PageRank, *Internet Mathematics*, Vol. 1, No. 3, pp. 335-380.

[8] B. Liu, *Web Data Mining*, Springer, Heidelberg, 2007.

[9] T. Haveliwala, Topic-Sensitive PageRank: A Context-Sensitive Ranking Algorithm for Web Search, *IEEE Transactions on Knowledge and Data Engineering*, Vol. 15, No. 4, pp. 784-796, 2003.

[10] S. Kamvar, T. Haveliwala, C. Manning, G. Golub, Extrapolation Methods for Accelerating PageRank Computations, *Proc. Of the 12th Intl. Conf. on World Wide Web*, Budapest, Hungary, pp. 261-270, 2003.

[11] I. Whitten, E. Frank, *Data Mining, Practical Machine Learning Tools and Techniques*, Morgan Kauffman, New York, 2005.

[12] J. Han, Y. Yin, G. Dong, Efficient Mining of Partial Periodic Patterns in Time Series Databases, *Proc. of the 15th IEEE Intl. Conf. on Data Engineering*, Sydney, Australia, pp. 106-115, March 1999.

[13] S. Simon, Modeling and Design Aspects of the MEMS Switch, *Proc. Of the 2003 IEEE International Semiconductor Conference*, Sinaia, Romania, September 28 - October 2, pp. 128-132, 2003.

[14] R. Maeda, M. Takahashi, S. Sasaki, Commercialization of MEMS and Nano Manufacturing, *Proc. Of the 6th IEEE Intl. Conf. On Polymers and Adhesives in Microelectronics and Photonics*, Tokyo, Japan, pp. 20-23, January 2007.

[15] G. M. Rebeiz, Homepage $http://www.eecs.umich.edu/rebeiz/rebeiz.html$.

[16] C. L. Goldsmith, Homepage $http://www.memtronics.com/page.aspx?page_id = 10$.

[17] Y. Zhai, B. Liu, Web Data Extraction Based on Partial Tree Alignment, *Proc. of the 2005 International World Wide Web Conference*, May 10-14. Chiba, Japan, pp. 76-85, 2005.

[18] L. Shih, D. Karger, Using URLs and Table Layout for Web Classification Tasks, *Proc. of WWW 2004*, May 17-22, New York, pp. 193-202, 2004.

[19] B. Kules, J. Kustanowitz, B. Shneiderman, Categorizing Web Search Results into Meaningful and Stable Categories Using Fast-Feature Techniques, *Proc. of JCDL'06*, June 11-15, pp. 210-219, 2006.

[20] X. Jin, R. Li, X. Shen, R. Bie, Automatic Web Pages Categorization with ReliefF and Hidden Nad've Bayes, *Proc. of SAC '07*, March 11-15, pp. 617-621, 2007.

[21] J. Engler, A. Kusiak, A. Mining the Requirements for Innovation, *Mechanical Engineering*, Vol. 130, No. 11, pp. 38-40, 2008.

[22] P. Chapman et al., *Step-by-step Data Mining Guide*, CRSIP-DM Consortium, CRISP-DM 1.0, 2000.

[23] N. Lavraĉ, H. Motoda, T. Fawcett, R. Holte, P. Langley, P. Adriaans, Introduction: Lessons Learned from Data Mining Applications and Collaborative Problem Solving, *Machine Learning*, Vol. 57, No. 1-2 , pp.13-34, 2004.

[24] R. Gajda, Utilizing Collaboration Theory to Evaluate Strategic Alliances, *American Journal of Evaluation*, Vol. 25, No. 1, pp. 65-77, 2004.

[25] M. Geringer, Strategic Determinants of Partner Selection Criteria in International Joint Ventures, *Journal of International Business Studies*, Vol. 22, No. 1, pp.755-786, 1991.

[26] M. Hitt, M. Dacin, E. Levitas, J. Arregle, A. Borza, Partner Selection in Emerging and Developed Market Contexts, *Academy of Management Journal*, Vol. 43, No. 3, pp. 440-467, 2000.

[27] Z. Chen, Learning about Learners: System Learning in Virtual Learning Environment, *International Journal of Computers, Communications and Control*, 3(1):33-40, 2008.

[28] H. Grebla, C. Cenan, C., Distributed Machine Learning in a Medical Domain, *International Journal of Computers, Communications & Control*, 1(S):245-250, 2006.

Int. J. of Computers, Communications & Control, ISSN 1841-9836, E-ISSN 1841-9844
Vol. V (2010), No. 1, pp. 52-70

Coordinating Aerial Robots and Unattended Ground Sensors for Intelligent Surveillance Systems

E. Pignaton de Freitas, T. Heimfarth, R. Schmidt Allgayer, F. Rech Wagner, T. Larsson, C. E. Pereira, A. Morado Ferreira

Edison Pignaton de Freitas and Tony Larsson
Halmstad University
School of Information Science, Computer and Electrical Engineering
Halmstad, Sweden
E-mail: {edison.pignaton, tony.larsson}@hh.se

Tales Heimfarth and Flávio Rech Wagner
Federal University of Rio Grande do Sul
Institute of Informatics
Porto Alegre, Brazil
E-mail: {theimfarth, flavio}@inf.ufrgs.br

Rodrigo Schmidt Allgayer and Carlos Eduardo Pereira
Federal University of Rio Grande do Sul
Electrical Engineering Department
Porto Alegre, Brazil
E-mail: {allgayer, cpereira}@ece.ufrgs.br

Armando Morado Ferreira
Military Institute of Engineering
Defense Engineering Graduate Program
Rio de Janeiro, Brazil
E-mail: armando@ime.eb.br

Abstract:
Sensor networks are being used to implement different types of sophisticated emerging applications, such as those aimed at supporting ambient intelligence and surveillance systems. This usage is enhanced by employing sensors with different characteristics in terms of sensing, computing and mobility capabilities, working cooperatively in the network. However, the design and deployment of these heterogeneous systems present several issues that have to be handled in order to meet the user expectations. The main problems are related to the nodes‘ interoperability and the overall resource allocation, both inter and intra nodes. The first problem requires a common platform that abstracts the nodes’ heterogeneity and provides a smooth communication, while the second is handled by cooperation mechanisms supported by the platform. Moreover, as the nodes are supposed to be heterogeneous, a customizable platform is required to support both resource rich and poorer nodes. This paper analyses surveillance systems based on a heterogeneous sensor network, which is composed by low-end ground sensor nodes and autonomous aerial robots, i.e. Unmanned Aerial Vehicles (UAVs), carrying different kinds of sensors. The approach proposed in this work tackles the two above mentioned problems by using a customizable hardware platform and a middleware to support interoperability. Experimental results are also provided.
Keywords: Sensor Networks, Unmanned Vehicles Systems, Wireless Communication, Heterogeneous Platforms

1 Introduction

New applications based on the interaction between autonomous robots and static sensor nodes are emerging. Part of the growing interest in using this interaction comes from the potential benefits it can provide, such as distributed processing and data gathering enrichment. Moreover, the deployment of such systems is becoming a tangible reality due to advances in small and efficient processors, sensor devices, and wireless networking. The integration of static and mobile sensor nodes enhances capabilities of the overall sensor network. It enables new applications in which fixed sensor nodes provide an active response capability, while unmanned mobile nodes acquire a spatial vision of the region, allowing monitoring of larger areas. Military and civilian applications, such as borderline patrol, search and rescue, area surveillance, communications relaying, and mapping of hostile territory, can take advantage of this kind of enlarged sensor networks. Since these tasks may be repetitive, tedious, and dangerous, they are ideal for autonomous unmanned vehicles [1].

Indeed, applications of sensor network can greatly profit of using different kinds of mobile and sophisticated sensors in addition to static ones, which typically are simpler and more resource constrained. The cooperation between these different types of sensors provides advanced functionalities that were not feasible before [2] and thus also opens a vast range of new application scenarios.

Wireless sensor networks are usually developed to perform specific applications. Sensor nodes have usually a small footprint including only the resources that are necessary to meet specific application requirements. However, reconfigurable and customizable architectures are important to make both sensor nodes and sensor networks more flexible. Depending on the application, sensor nodes vary from resource-constrained to high performance computing nodes, resulting in a heterogeneous network composed by a variety of sensor node types.

The main issues in developing heterogeneous sensor networks are: (i) support for cooperation among heterogeneous nodes; and (ii) customization of sensor nodes. The former is related to concerns such as message exchange synchronization, QoS requirements management, task (re-) allocation, and network adaptation. The later is related to the diversity of node platforms, which may be built upon very distinct hardware components controlled by very different pieces of software.

Considering (i), the use of a middleware is a suitable approach to address the mentioned concerns, since it can integrate the technologies used in different nodes by means of a common communication interface and cooperation support. Regarding (ii), customizable architectures can be very useful to build platforms for different sensor network nodes, from the very simple to the more sophisticated ones. This kind of architecture can provide a common base capability for all nodes. However, for nodes that need more advanced capabilities, the required resources can be incorporated. Hence, even though all nodes should have the same base capability, some of them could be equipped with additional resources, thus making the sensor network more powerful due to this allowed heterogeneity.

This paper presents a flexible and adaptable platform infrastructure intended to support heterogeneous sensor network applications composed by static sensor nodes on the ground and mobile sensors carried by autonomous aerial robotic platforms, such as autonomous UAVs. It is based on the proposal of (i) a flexible middleware [3] and, (ii) on a customizable hardware architecture aimed for sensor nodes, called FemtoNode [4]. The key idea is to use this customizable platform to deploy different kinds of sensor nodes, from very tiny and resource constrained up to more sophisticated ones. Both types of nodes run a common middleware in order to provide the desired interoperability that will allow the cooperation among different sensor nodes. Nodes may be built upon the FemtoNode architecture and alternatively upon nodes with other hardware platforms, such as SunSPOT [5] or similar ones. Moreover, the middleware provides services to allow autonomous decision making by the nodes, in order to perform reflection about the runtime conditions and adapt the system according to the current needs. These features allow the mobile nodes to take decisions about their movements and all nodes, i.e. also the fixed nodes, to take decisions about network parameters, such as QoS and resource usage. In addition, this paper presents a bio-inspired coordination mechanism that uses artificial pheromones used to provide information about the position of the mobile sensor nodes, facilitating the cooperation among these nodes and the others that make part of the network.

The remaining of this paper is organized as follows: Section 2 presents related work in the area. In Section 3, the application scenario is highlighted, characterizing the network heterogeneity. Section 4 presents an overview of the proposed middleware, while Section 5 presents details about the key issues addressed by the middleware. The pheromone-based coordination among static and mobile sensors is the focus of Section 6. In Section 7, the FemtoNode customizable hardware architecture is described. Section 8 presents a case study while Section 9 provides results for some of the presented features of both middleware and customizable hardware. Finally, Section 10 draws concluding remarks and gives directions for future work.

2 Related Works

AWARE [6] is a middleware whose goal is to provide integration of the information gathered by different types of sensors, including low-end sensor nodes in a wireless sensor network and mobile robots equipped with more sophisticated sensors. Our proposal not only addresses heterogeneous sensors and their coordination, but also concerns like QoS and runtime reflection in order to cope with changes in the environment and in the network, which is missing in [6]. Moreover, in our approach, the autonomous decisions taken by the mobile nodes characterize a higher intelligence degree of our mobile robots if compared with the proposal presented in [6].

Batalin et al. [7] propose an approach that uses a sensor network to guide a mobile robot with limited sensor capabilities. This work presents aspects of cooperation among different sensors, but it does not address the same problem as proposed in our work. In that approach, the mobile node has limited sensor capabilities and only gathers data from the sensor network to guide the robot's movement. In our work, mobile nodes are equipped with sophisticated sensors that can provide data, which may be merged with data that come from the other nodes in order to achieve more refined decisions to guide the movement of the mobile nodes.

Schmidt et al. [8] present an approach that uses artificial intelligence techniques to configure an underlying middleware. Their approach uses the concepts of missions and goals to plan the allocation of tasks in a network of homogeneous nodes. The handling of heterogeneous nodes is one of the differences between our work and their approach. Additionally, in that work, the intelligence is outside the middleware and influences it by just sending commands or adjusting its parameters. In our approach, the decision making support is an integral part of the middleware, spreading intelligence over the network.

Jin et al. [9] provide a very consistent proposal to handle the problem of balance between target search and response by a team of UAVs. The work evaluates the tradeoff between search and response within the framework, presenting a predictive algorithm that provides a good balance between these tasks. The first difference between our approach and this work is that we handle only the alarm response, abstracting the concern about the UAVs movement planning to perform the search for new targets. This difference is due to the peculiarity of the distinct missions addressed in the current paper and in [9]. We focus on area surveillance, while they focus on target acquisition. In our case, the whole area must be covered, which may not be true in the target acquisition they address. Another difference is that we use the UAVs in coordination with ground sensor nodes. Besides, the assumption of a centralized information base considered in their work is not used in our proposal. Their initial centralized off-line task assignment is another premise that is not valid in our work.

In [10], an approach using digital pheromones to control a swarm of UAVs is presented. The method proposed by the authors uses digital pheromones to bias the movements of individual units within a swarm toward particular areas of interest that are attractive, from the point of view of the mission that the swarm is performing, and away from areas that are dangerous or just unattractive. In the large sense, the pheromone-based strategy used in our work has a similar goal, driving the UAVs to areas of interest. However, differently from their approach, we use the pheromone traces to localize the UAVs when an alarm is issued by a ground sensor node informing an event of interest and then drive the UAVs to the location where the event happened.

3 Application Scenario Overview and Network Heterogeneity

In the following, heterogeneity means that nodes in the network may have different sensing capabilities, computation power, and communication abilities. Additionally, it means that they run on different hardware and operating systems. Therefore, such sensor networks are made up of low- and high-end nodes. Moreover, sensor nodes may have fixed positions or be able to move, being carried by mobile aerial robots (UAV platforms), which can also vary from very small, as in [11], up to huge aircraft platforms, like GlobalHawk [12].

Low-end sensor nodes are those with constrained capabilities, for instance piezoelectric resistive tilt sensors, with limited processing support and communication resource capabilities. High-end sensor nodes include powerful devices like radar, high definition visible light cameras, or infrared sensors, which are supported by moderate to rich computing and communication resources.

Mobility, as mentioned, is another important characteristic related to the heterogeneity addressed in this work and requires special attention. Sensor nodes can be statically placed on the ground or can move on the ground or fly at some altitude over the target area in which the observed phenomenon is occurring. Figure 1 graphically represents the idea of the three heterogeneity dimensions considered in this work, in which each axis represents one of the considered characteristics.

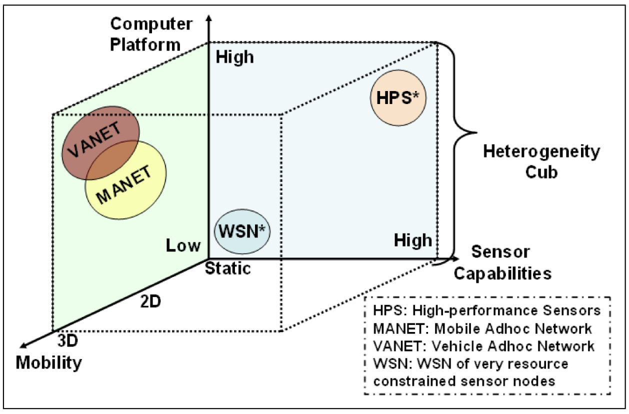


Figure 1: Heterogeneity Dimensions.

The reason for heterogeneity in the sensor nodes is to support a large range of applications that deal with very dynamic and changing scenarios, which require different types of sensor capabilities. Moreover, these different scenarios require also adaptations in the network, in terms of choosing suitable sensors for the tasks at hand as well as feasible QoS parameters, among others.

In order to illustrate the above idea, suppose that a network has the mission of providing a certain kind of information during a given period of time. The set of sensors selected in the beginning of a mission may not be the most suitable one during the execution of the whole mission. The network must be thus able to choose a better alternative, among the set of all available options, in order to accomplish the mission. For example, an area surveillance system may receive the mission to observe if certain types of vehicles that are not allowed to pass through the surveyed area make any such violation and report if that is the case. To perform this in an efficient way ground sensors are set to alarm in the presence of unauthorized vehicles. Suddenly, an alarm is triggered by one of these sensors. Then, in order to verify the occurrence, UAVs equipped with visible-light cameras are commanded to fly over the area where the ground sensor has issued the alarm. Due to a sudden change in the weather (e.g. the area becomes foggy or cloudy), visible-light cameras become useless. However, the mission must still be accomplished. Therefore, nodes must be coordinated to accomplish the mission by sending UAVs equipped with other kinds of sensors that can provide the required information in bad weather conditions, such as infrared cameras.

4 Middleware Support Overview

One of the key ideas of the proposed work is to drive the use of sophisticated sensors carried by UAVs using data from low-end nodes. UAVs' equipped with intelligent behavior uses such data to decide the trajectory direction during the system runtime. Consequently, the proposed middleware must be able to provide cooperation among the different sensors and still fit in both the low-end and rich sensor nodes.

The middleware must thus be lightweight and provide enough customization in order to address the needs of both types of sensor nodes. These goals are achieved by using aspect- and component-oriented techniques, as in [13] and [14], and also a mobile multi-agent approach, as discussed in [3]. Figure 2 depicts an overview of the middleware layers, whose description is provided in the following.

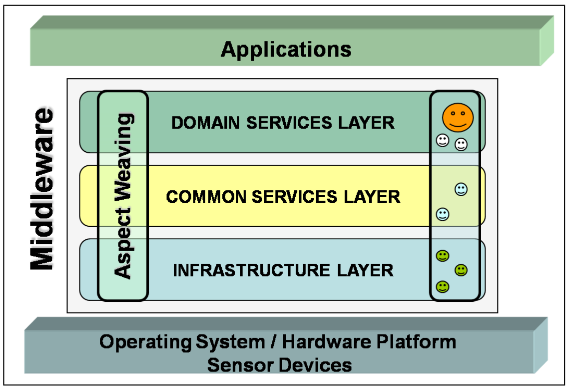


Figure 2: Overview of the Middleware Layers.

The bottom layer is called *Infrastructure Layer*. It is responsible for the interaction with the underlying operating system and also for the management of node resources, such as available communication and sensing capabilities, remaining energy, etc. This layer is also responsible for coordination of necessary resource sharing.

The intermediate layer is called *Common Services Layer*. It provides services that are common to different types of applications, such as QoS negotiation, quality of data assurance, and data compression. Other concerns also handled within this layer are: deadline expiration alarms; timeouts for data transmissions; number of retries and delivery failure announcements; resource reservation negotiation among applications (based on priorities established by missions and operation conditions); bindings; and synchronous/asynchronous concurrent requests.

The top layer is called *Domain-Services Layer*, whose goal is to support domain specific needs, including data fusion support and specific data semantic support, in order to allow the production of application-related information from raw data processing. Fuzzy classifiers, special types of mathematical filters (e.g. Kalman Filter), and functions that can be reused among different applications in the same domain are found in this layer.

"Smile faces" in Figure 2 represent autonomous agents that can provide specific services in a certain node at a given moment during system runtime. The *Domain-Services Layer* hosts a special agent (called *planning-agent*), which performs a special task related to the reasoning about the missions that a node is responsible for performing.

Concerns that affect elements in more than one middleware layer, such as security and real-time requirements control, are represented as cross-layer features. These crosscutting concerns are addressed by the aspect-oriented approach presented in [13].

A. Planning-agent Model Overview

As an important element in the proposed approach, the model of the planning-agent is briefly presented.

The model used for the planning-agent is a cognitive one, based on the model of mental attitudes, known as BDI model (Belief-Desire-Intention), presented in [15]. The motivation is that the BDI model appears to suite well to the surveillance problem addressed, as some decisions that need to be taken by the planning-agent require cognitive skills to evaluate if certain actions are adequate to achieve a desired result. These decisions are based on knowledge about conditions that may interfere on the performance of those actions. In the current problem formulation, it is desired to obtain information by means of sensing activities, which are the goals of a sensing mission. This knowledge is the ”belief” that the node has about the relevant conditions, and the intentions are translated into the actions required to retrieve the desired information.

As an example, when a UAV receives an alarm, it will consult its beliefs in order to decide if it is able to respond to that alarm. If so, it will take the responsibility for the respective alarm and include the incoming information from the alarm into its beliefs, as well as include the accomplishment of the alarm handling into its desires. Based on that, the UAV will then intend to fly to the place where the alarm was issued in order to execute the respective sensing over the target.

B. Middleware Services Usage Example

Taking the application scenario given as example in Section 3 and the middleware layers described above, this subsection provides some examples of the utilization of middleware services and of the interaction between nodes.

Assuming the described example, in the case that alarms are issued by several ground sensor nodes, the *Domain Services Layer* provides data aggregation, for fusion of data from many sensors in an area, which can provide richer information, such as the direction of a crossing vehicle, or handle problems such as alarm duplicity. The delivery of such information has an expiration time, since after a given time threshold, it is probable that the vehicle may have changed its trajectory, making the previous collected data useless. Therefore, the *Common Services Layer* associates QoS with the delivery of messages as well as a guarantee mechanism to assure that an alarm has being correctly delivered and in time to a UAV. The *Infrastructure Layer* uses these QoS parameters to manage resources utilization.

In the UAVs, the middleware makes the complementary task, providing data fusion of images with matching alarm messages received from low-end nodes, i.e. fusion of position information included in the alarm messages with images that are being taken by the UAVs, in the *Domain Services Layer*. This is the case, for example, when more than one alarm has been issued in a given location. The data fusion helps in distinguishing the source of the alarms. QoS verification of incoming messages is performed by the *Common Services Layer*, which checks if either stored data can be used by the application or a request for fresh data must be sent. This is the case, for example, if the ground sensor network experiences problems and takes a too long time to deliver an alarm message to a UAV. It may happen that, when this alarm comes to the UAV, the object that triggered the alarm is not anymore at the location where it was detected. At this point, a request for confirmation can be sent by the UAV to another UAV in the region around the location of that alarm, requiring certain levels of QoS.

Depending on the results from data processing, as described above, the UAVs autonomously decide their own placements over the surveillance area, by means of the reasoning mechanisms of their respective *planning-agents*. Additionally, other factors influence this decision, e.g. specific needs of the current situation and also sensible data that must be sent to the base station or to another UAV. In this case, data segregation and network utilization control play a crucial role in the efficiency of the system. Important data must be prioritized. Thus, network resources are allocated to data transmission according to the established priorities. Possibly, the specific needs faced by the system in a given situation may require adjustments in the resource usage policy, which influences the services provided by the *Common Services Layer*. For instance, a change to a "Mandatory Priority" may take place in order to assure that data about a detected object or phenomena arrive within a given deadline at the base station via relayed communication through other UAVs.

In order to consider all the mentioned factors, necessary support for reflection about the network conditions and mission needs must be available. This support is provided by the *planning-agent* installed in the *Domain Services Layer*, which will, for instance, analyze the current conditions and requirements and, taking into account the information provided by other sensors, decide the best placement of the UAV in order to meet the system needs, thus selecting the next steps to move.

5 Addressing Key Issues of Sensor Networks with the Proposed Middleware

The proposed middleware uses the publish-subscribe paradigm and is inspired by the *Data Distribution Service for Real-time Systems* (DSS) specification, from OMG [16]. Some nodes publish their capabilities and the offered data, while others subscribe to data in which they are interested.

Although largely being inspired by the OMG DSS standard, the proposed middleware does not follow the whole specification. As it is intended to fit in both low-end nodes (based on simple and constrained platforms) and more sophisticated ones (carried by the UAVs), it must not only be lightweight but also provide capabilities for customizations to cope with the needs of the different sensor nodes. Consequently, the middleware uses a minimalist approach, being kept as simple as possible in each node. Required features are included by adding components or weaving aspects to handle real-time and other non-functional crosscutting requirements in the minimal middleware. Using aspects to tune the middleware is out of the scope of this paper, and for more information readers are referred to [13] and [17].

The following subsections show how the proposed middleware addresses some of the main platform needs in heterogeneous sensor networks, enabling the intelligent behavior of the mobile nodes and providing the required data with real-time guarantees.

A. Flexibility

The middleware provides full communication control, i.e. it does not use underlying mechanisms available in the nodes' network layer. Instead, it provides its own communication control. This means that all parameters related to communication are controlled by the middleware, which uses only basic connectionless communication services offered by the nodes' network layer. The middleware handles parameters such as number of retries, message priority, memory utilization for buffering, and timing. This provides more flexibility, with direct impact in the reduction of message delivery latency.

B. Network Reflection

Reflection over the state of the network is a feature that enables the sensor nodes to take decisions regarding their participation in accomplishing a specific mission or sub-mission. In the mobile sensor nodes, this feature is responsible for the decision regarding the movement of the robot platform that carries the sensor, in order to take it to a place or the area related to a mission. The reflection considering the network conditions is performed inside the middleware by the planning-agent, which schedules the activities that should take place in order to accomplish a given mission.

C. Dynamicity

When a node gets into the network, its services are announced using the publish-subscribe paradigm. Then, all interested nodes can subscribe for those services. This eliminates the need for a dedicated server node that centralizes all available services in the network. Additionally, this approach reduces latency in acquiring data because there is no intermediary node between the data producer and consumer.

D. Minimum Message Exchange

The publish-subscribe paradigm by itself already reduces the number of exchanged messages due to the elimination of intermediate nodes (such as brokers). However, bandwidth for control messages is still required. This bandwidth need can be further reduced with the use of smart techniques such as QoS contracts and attaching freshness timestamps to data, thus avoiding unnecessary request for data re-send. Another important characteristic for minimizing bandwidth requirements is the selection of routing strategies that optimize the use of the communication channel, finding the best path to be followed to arrive at the destination node of a message. An example of such strategy is a routing mechanism based on pheromone traces left by the mobile nodes, which will be explained in the following section and highlighted when presenting the experimental results.

E. Multicast Communication

The middleware uses multicast communication to reach selected destination nodes. This type of communication positively influences the latency and throughput, as data is sent at the same time to several nodes without unnecessary broadcast and delays, which would occur in unicast communication. A negative-acknowledgement (NACK) strategy (controlled by timeout expiration) is adopted in order to reduce acknowledgement messages in the network. However, very sensible data may require a positive acknowledgement to assure their delivery. Hence, positive acknowledgement is also made available as a middleware service and can be used when required.

F. Network Resources Usage Control

In order to improve the overall system performance, the control of the use of communication media and of transmission buffers is crucial. The middleware performs this task by taking into account two factors: (i) the priority associated to each application; and (ii) the resource sharing policy adopted in the system. There are three available resource sharing policies: (a) **Fair Sharing:** priorities are not considered and thus all applications have the same right to use the resources in a round-robin scheme, which is organized in an incoming FIFO queue; (b) **Soft Priority Sorted:** the priorities are taken in account, but in a relaxed way. If a higher priority application needs a resource already used by a lower priority one, it must wait until the resource is released. Due to its higher priority, it will get access to the resource before other applications, which may be waiting for the resource; (c) **Mandatory Priority:** higher priority applications can preempt lower priority ones in order to access the desired resources. Priority inversion issues are handled by a priority inheritance mechanism.

G. QoS Control

QoS control is performed through a contract between the provider and the requester of data. When a node publishes a data service, it informs about the QoS (level) offered. Nodes interested in the published data service should accept the offered QoS and subscribe to the service. However, if a node is interested in the data but does not agree with the offered QoS, it has two alternatives: (a) if the application that is requiring the data has a priority lower than any other one using the same service, the requester node looks for another data provider; (b) if its priority is higher than all other applications, the requester node negotiates with the data provider node, in order to obtain the desired QoS. This renegotiation occurs in spite of undesired consequences that may affect other lower priority applications, which need to look for another data provider if the QoS could not be accepted anymore.

H. Use of Cached Values

The use of cache in both data providers and requesters may avoid unnecessary data communication. When the measurement device gathers a new value, the data provider publishes the new value, thus updating its subscribers. If the data size is large, requiring many packets to be transmitted, a differential value can be sent instead of the whole data value in order to reduce packets transmission. This option is arranged in advance, at the time when the nodes are negotiating the QoS contract.

I. Data Segregation

There are two kinds of data exchanged among nodes in the network: control data and application data. Control data is small and can usually not experience latency or unexpected delays to achieve their destination. Thus, control data are separated from application data by receiving higher priority to be forwarded. Moreover, data from different types of applications are handled according to their type and semantics, e.g. the communication of a video stream is handled differently from the communication of character strings. Data from a specific application can be handled with higher or lower priority, depending on its semantics.

J. Synchronous and Asynchronous Calls

The middleware is intended to support both synchronous and asynchronous calls. Synchronous calls are time bounded in order to avoid unpredictable waiting periods. The waiting time and number of retries are configurable and negotiated during the QoS negotiation. Asynchronous calls are also provided in order to support the handling of external unpredictable event.

6 Pheromone-based Coordination Strategy

The coordination strategy used in this work to make mobile sensor nodes cooperate with static sensor nodes is based on pheromone traces handed over by the mobile sensors to the static ones. Artificial pheromones are usually applied to distributed coordination by means of stigmergy, the indirect communication using environment cues [17]. A pheromone trail is deposited in the environment when the entities are moving. The pheromone provides information to other entities when they pass over it. Artificial pheromone also looses its strength along the time, modeling the evaporation of the real pheromones. In the UAV research field, pheromones are used to guide the movement of UAV swarms, for instance in surveillance and patrolling applications [18] [19].

Differently from other existing approaches, in our work pheromones are used to guide the selection and assignment of a suitable UAV to handle an alarm issued by a ground sensor node. When an alarm is issued by the detection of a target, the network is responsible for selecting an appropriate UAV to respond to the alarm. This is performed by routing a given alarm to the UAV that has the strongest pheromone trace over the area. Having this information, the UAVs will base their movement decisions in a way to respond to the received alarms. This strategy is called here heuristic-P.

Following the above outlined principles, the UAVs that are not engaged in the handling of any target will leave pheromone traces over the area which they cross. This pheromone trace is represented by a piece of information that is taken by the ground sensor nodes that are deployed in the area through which the UAVs have passed. When a target is detected by a ground sensor node, an alarm is issued, as already mentioned. The decision about which UAV that will handle the potential target indicated by the issued alarm will be taken by the ground sensor nodes, by routing the alarm in the direction that points to the UAV which has the strongest pheromone trace over that area of the network. This process just considers the pheromone trace handed over by the UAVs to ground sensor nodes. This means that the only parameter taken into account is the time interval since a UAV passed by that specific location. Heuristic-P is inspired in [20], which presents a pheromone-based strategy to migrate services in a sensor network, in which the pheromone concentration determines the places where the services are required. In heuristic-P, instead of services, alarms are moved through the network following the pheromone concentration. Figure 3 presents a scenario that illustrates the strategy. A ground sensor node in the left border of the area detects a target. Then it issues an alarm, which is received by its neighbors. However, only those which have pheromone information about a UAV stronger than that of the alarm issuer will forward the alarm. This way, the alarm will follow a path to the closest UAV, which is represented in the figure by the shaded sensors, until the alarm delivery.

Figure 4 illustrates the choice of the strongest pheromone trace to be followed by an issued alarm. It is possible to observe that the alarm follows the strongest trace, which corresponds to UAV-A, until its delivery to this UAV. The arrows illustrated besides each sensor node represent how strong the pheromone

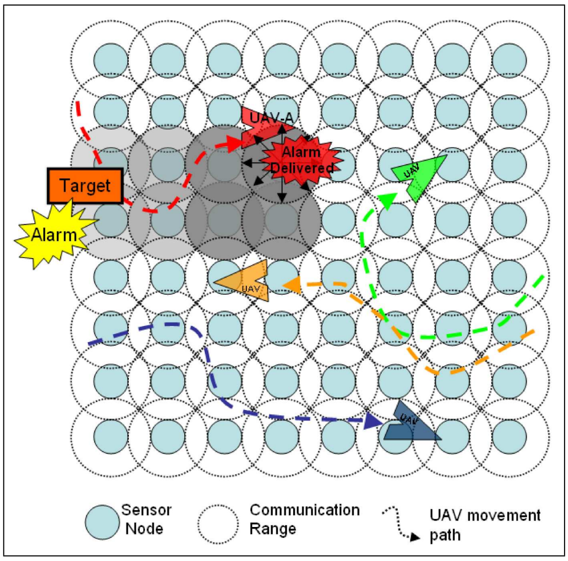


Figure 3: Illustrative scenario for the pheromone strategy.

of each UAV is. As it is possible to see, the pheromone level of UAV-A is increasing to the left, while the pheromone level of UAV-B is increasing to the right.

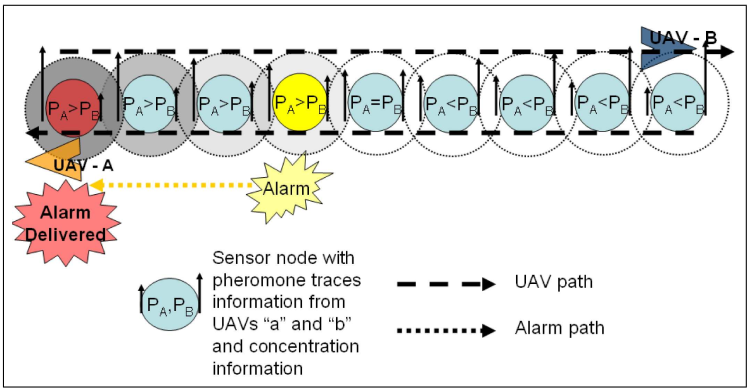


Figure 4: Choice of a UAV based on the pheromone strategy.

When an alarm reaches the UAV indicated by the strongest pheromone trace, if this UAV is not engaged in the handling of another alarm it sends a confirmation message to the node that had delivered the alarm. If the suggested UAV is already engaged in another alarm, the current alarm follows the second strongest pheromone trace to find another UAV to engage.

When an idle UAV detects a new target, it takes the responsibility for handling it. In case that the UAV is already busy with another alarm response mission, it relays the incoming alarm that will be routed to another UAV, according to the pheromone-based heuristic-P strategy explained above.

In order to increase the robustness of the proposal, in case an alarm is issued by a node that has no pheromone trace, a direction is randomly chosen and the alarm is sent in that direction until it finds a pheromone trace. When the trace is found, it follows the trace as explained above. This situation is more likely to occur in the initialization of the system, especially in cases in which the number of UAVs deployed in the system is very low with regard to the area under surveillance.

When a UAV receives an alarm and is not able to perform the task, it may send the alarm back to the network, which will try to find another UAV following the traces, or hand it over directly to another UAV. This situation may occur when the type of the sensor that the UAV carries is not appropriate to handle the event that was detected.

7 FemtoNode - Wireless Sensor Architecture

The architecture of a sensor node aims at efficiently supporting specific application needs. It requires a dedicated processing module, including a wireless communication interface, which meets both energy and performance requirements, as well as respects footprint constraints. The fact that application requirements as well as environment and other operational conditions may change during system run time imposes a major challenge [21]. In this context, the use of reconfigurable hardware [22] appears as an interesting alternative. Therefore, a customizable sensor node called FemtoNode is proposed. It contains a customizable ASIC and a wireless communication interface, which are configured according to application requirements.

The nodes use the RT-FemtoJava processor [23], a stack-based microcontroller that natively executes Java byte-codes. It implements an execution engine for Java in hardware, through a stack machine that is compatible with the specification of Java Virtual Machine. The customized application code is generated by the Sashimi design environment [24]. The code also includes a VHDL description of the processor core and ROM (programs) and RAM (variables) memories. The Sashimi environment has been extended to incorporate an API that supports concurrent tasks, implementing the RTSJ standard [25].

As RT-FemtoJava is customizable, its code can be optimized according to the application requirements, reducing the occupied hardware area and also the energy consumption and dissipation. The customizable hardware architecture of the FemtoNode allows the use of the sensor node as either a low- or high- end node. If the application requires higher performance resources to handle more complex data, such as image processing, additional resources can be included in the FemtoNode implementation. However, if the application is aimed at processing simple data, such as those from presence sensors, a reduced set of resources is used in the processor. This feature is important for the sensor node, because energy consumption is a great concern in wireless sensor networks, due to the nodes' limited energy resource. Besides, reducing the unused resources during its synthesis the sensor node architecture allows its implementation in reconfigurable circuits with fewer available logical units, which is a feature that provides a larger application portability between different reconfigurable architectures with fewer available resources.

In the current implementation, the FemtoNode includes a wireless transceiver of Texas Instruments CC2420, which utilizes the IEEE802.15.4 standard communication protocol targeted to wireless sensor network applications with a low data rate. A module adapter described in VHDL implements the interface with the wireless transceiver. The module uses data and address buses to communicate with the processor, performing the exchange of data and allowing the transceiver parameters configuration.

As the data transfer rate from the wireless transceiver is low, compared to the processor frequency, the wireless communication module implements a buffer to store data, preventing delays while providing the necessary data to the processor. The module uses an interrupt system to inform the processor when a reception occurred.

To facilitate the use of the wireless communication module by the application developers, a communication API has been developed. The Wireless-API abstracts details of the communication media between the sensor nodes, offering a simplified form for the configuration of the data transfer module.

The Wireless-API is used by the middleware communication services in the *Common Services Layer* and also by resource management services in the *Infrastructure Layer*, in order to implement end-to-end communication with the desired QoS and reliability control.

8 Case Study Description

In order to illustrate the use of the proposed platform infrastructure, including the customizable FemtoNode and the adaptable middleware, an area surveillance application is studied. In this application, low-end sensors nodes are scattered on the ground along a borderline. In case an unauthorized vehicle crosses the borderline limit, the sensors issue an alarm which will trigger the use of Unmanned Aerial Vehicles (UAVs), which are equipped with more sophisticated sensors, such as radars or visible light cameras, in order to perform the recognition of the vehicle. Figure 5 presents this scenario.

Figure 5: Area Surveillance Application Scenario.

Sensor nodes with two different architectures compose the described surveillance system, one based on the FemtoNode and another one on the SunSpot. Each of them includes all necessary resources to meet the requirements of their utilization. Thus, based on the application specifications, a customization of the FemtoNode architecture was implemented. The UAVs' architecture is a FemtoNode with a large set of resources, capable of processing a large amount of data. On the other hand, both FemtoNodes and SunSpots were used to populate the sensor network on the ground. The FemtoNode architecture used in the ground sensor nodes is simpler and more constrained in terms of available resources, as the SunSpot one, and hence only capable of processing simple data, like those produced by piezoelectric sensors or accelerometers, which inform only if an object over a certain weight threshold passed over its sensing area.

According to the coordination strategy based on pheromones presented in Section 6, the sensor nodes on the ground route the alarms according to the pheromone trace left by the UAVs, choosing the strongest trace to follow. When the alarm achieves a node close to the UAV, the alarm is delivered. This mechanism addresses several problems related to the communication between nodes, such as controlled delay, delivery assurance, and alarm duplicity handling.

9 Results

Several simulations of the scenario described in Section 8 have been performed. They do not cover all aspects mentioned in the above description, but focus on the behavior of the system using the coordination strategy described in Section 6. These simulations were conducted using ShoX [26], a powerful wireless network simulator implemented in Java.

Additional results related to the customizable FemtoNode working in a heterogeneous network together with SunSpot nodes are also presented. These results were gathered from the deployment of a laboratory-size testbed demonstrator, which shows the applicability of the platform.

The simulations provided results in an "ideal-like" operation condition environment for the proposed technique, while the demonstrator was built in order to assess if these results provided by the simulations are possible to be achieved in a real deployment.

For more results related to the mechanisms that support the other features mentioned in Section 5, interested readers are referred to [27], [28], and [29].

A. Simulation Results

The metrics evaluated in the simulations were: 1) the mean response time to the alarms generated in the system, and 2) the number of alarms lost, due to communication failures.

The simulation setup was the following: The surveillance area has dimensions 10 Km x 10 Km, in which 20,000 ground sensor nodes are randomly deployed with independent uniform probability (homogeneous Poisson point process in two dimensions, which generates a geometrical random graph). This distribution gives more than 70% probability that the nodes in the network will form a connected graph [30], for a communication range of 500 meters. Six UAVs of three different types, equally distributed, patrol the area, having a communication range of 1.5 Km and flying at speeds from 100 Km/h up to 120 Km/h. Three different runs were simulated, with one, three, and five targets respectively. The targets can further be of five different types, randomly chosen, with speeds from 50 Km/h up to 80 Km/h.

Figure 6 presents the simulation results in terms of the mean time required to respond to the alarms. Both raw data from each run (total of 20 runs for each number of targets) and the average value (lines with squared dots) are plotted in the figure. It is possible to observe that, in the worst case, the mean time to find a UAV that is idle to engage in the handling of an alarm is around 4 seconds, in the scenario with the maximum number of targets. On the other hand, in the best case, when there is just one target, the time needed to find a UAV is in average less than 1 second. An explanation for this behavior is that it is more probable to find an idle UAV when the number of targets is smaller. This may happen because, when there are more targets, an alarm message may follow a pheromone trace of a UAV that has just engaged in handling a target announced by another alarm, so the alarm must be retransmitted to the network and follow another trace. However, the solution does scale, as the increase in the mean time to find an idle UAV is linear with the increase in the number of targets, as can be concluded by taking the average values for all runs for each number of targets.

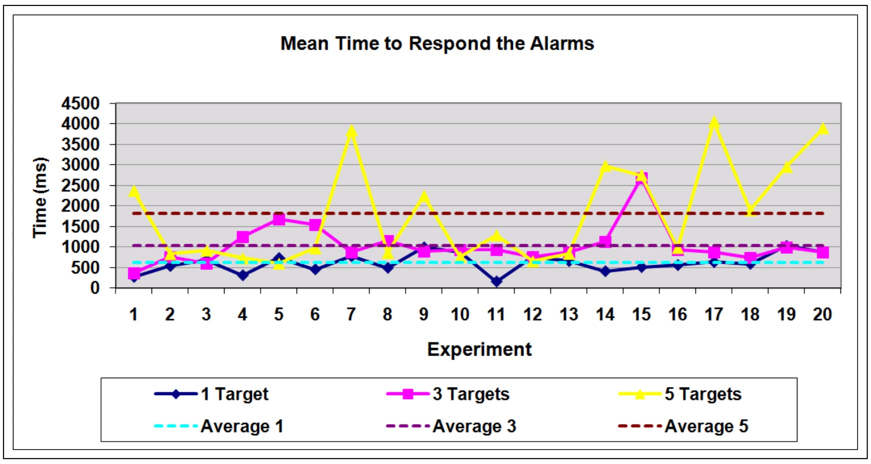


Figure 6: Alarm Response Time Achieve by the Simulation.

The second metric evaluates the system efficiency in terms of detecting a target and correctly routing the alarm message to an idle UAV. For all simulation runs, no alarm was lost, which means that the system had 100% efficiency for the simulated scenario and correctly found an idle UAV at all occasions when an alarm was issued.

B. Demonstrator Results

The simulations performed in ShoX showed that the approach proposed in this paper works well in the described scenario. However, wireless communications are very sensible to interferences and unpredictable variations. This means that simulation data, such as communication reachability and delays, are not always confirmed in real deployments. This fact motivated the deployment of a demonstrator to assess the properties of a network used to evaluate part of the system presented in this paper.

The deployed demonstrator is composed as a network consisting of sixteen static ground sensor nodes (nine SunSpots and the others FemtoNodes) and one mobile node (FemtoNode). The ground sensor nodes are equally distributed in a grid in an area of 225 square meters. The mobile FemtoNode, moved manually, represents a UAV that ”flies” over this area leaving pheromones over the ground sensor nodes via a periodic beacon message sent to the network. Upon the occurrence of an alarm, the nodes route it in the direction of the nodes with stronger pheromone traces, until it arrives at a node which has communication with the UAV. Figure 7 presents the demonstrator setup. The radio in the nodes was adjusted to provide a communication range of 5 meters, such that the nodes are capable of communicating only with their immediate vertical and horizontal neighbors, which are 5 meters apart, but not with their diagonal neighbors or any other node in the grid. The mobile node, representing the UAV, has the same communication range configuration as the static nodes.

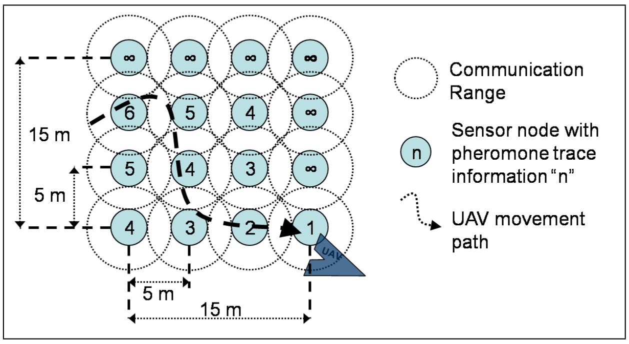


Figure 7: Demonstrator Setup.

The pheromone traces in the nodes are represented by the numbers in the center of the circles representing the ground sensor nodes in the figure. The smaller the number is, the stronger the pheromone. This translates the idea of the time past since a ground sensor node received the last pheromone beacon from a UAV. When a ground sensor node receives this pheromone beacon, it sends this information to its neighbors with a pheromone one point weaker (a number one unit greater than the one representing the node’s pheromone information). This is an indirect beacon that helps the other nodes find the traces to route the alarms. The nodes that receive the indirect beacons do not forward it. The symbol ∞ means that the node has no pheromone trace, i.e. the last beacon (directly from a UAV or indirectly from another ground node) was received a long time before, above a tunable threshold. The number representing the pheromone is periodically incremented, indicating that the pheromone trace becomes weaker when time elapses, until disappearing (become ∞). Figure 8 presents an example of how an alarm issued by a sensor node (Figure 8-A) is routed through the network, following the pheromone traces (Figure 8 from A to D), until it is delivered to a UAV (Figure 8-E).

Twenty runs were performed. In each of them, an alarm was generated by one of the static nodes, randomly chosen, which had to be routed to the UAV according to the pheromone mechanism described in Section 6 and implemented as described above. Some of the middleware features presented in Section 5 were installed in the static nodes, such as message priorization and QoS control, in terms of delay to forward an alarm message [31]. In order to stress the network and test these mechanisms, random messages were generated by the static nodes, which competed with the beacon and alarm messages for

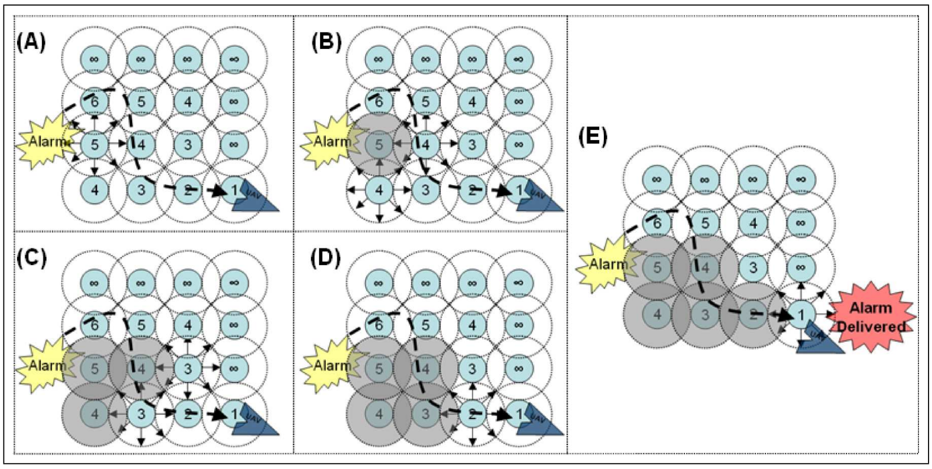


Figure 8: Alarm Routing and Delivery.

the utilization of the communication resources.

The evaluated parameter with the described testbed was the time to respond to the alarms generated in the system. By obtaining this metric, the delay of one hop communication was calculated and compared with the one achieved in the simulation results described before. Figure 9 presents the time taken by the system to deliver the alarm to the UAV.

The average number of hops to deliver the alarm was 5 hops for the 20 runs of the testbed. Taking the average of the time to deliver an alarm, 538.85 ms, and the average number of hops, we get an average delay of 107.77 ms in each hop.

Considering the simulation results, taking the worst case scenario, the one with 5 targets, in average, the number of hops for an alarm to be delivered was 13.78. Taking the average of worst case scenario, 1,821.65 ms to deliver an alarm, we get a 132.14 ms delay for an alarm to be forwarded among the static nodes in each hop.

Comparing the delays obtained from the simulation runs and from the demonstrator, it is possible to observe that they are very close to each other. The delays obtained with the demonstrator are even better than the ones achieved by simulation, which shows the applicability of the approach described in this paper.

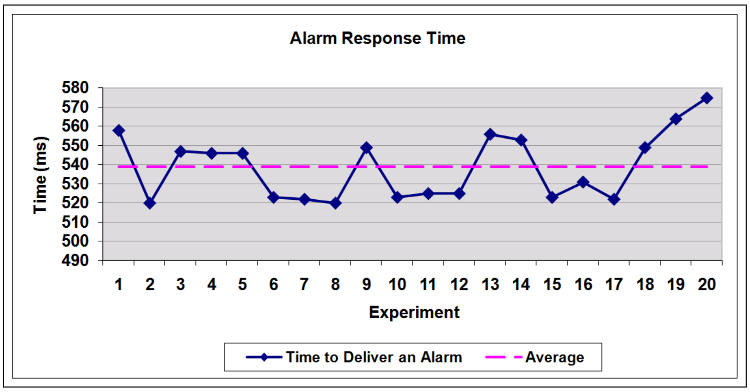


Figure 9: Alarm Response Time Achieved by the Demonstrator.

10 Conclusion and Future Work

This paper presented a system solution to provide interoperability and coordination support for heterogeneous sensor networks composed by ground static sensor nodes and mobile sensors carried by autonomous aerial robots. This solution is based on customizable sensor nodes and an adaptive middleware to manage different resources available in each customized sensor node. The FemtoNode platform provides a support to the development of different types of nodes, customized according to specific requirements. The middleware gives support to the network heterogeneity, enables adaptations to fulfill requirements that may change during the system run-time, and also promotes the necessary coordination among nodes.

The coordination and cooperation among distinct nodes in the network allow intelligent behavior of the unmanned vehicles, resulting in autonomous decisions regarding their movement and also how they can complement the work performed by other nodes. This intelligent behavior is based on data exchanged between nodes that then are aggregated and analyzed in order to support the autonomous decisions.

Simulation and testbed results were provided. These results assessed the suitability of the pheromone coordination mechanism, presenting delay time results for the delivering of alarm messages that drive the movement of the mobile sensor nodes in the system.

Additional simulations are planned in order to assess the effectiveness of other features of the middleware that were not yet explored, such as the use of cached values and multicast communication. In order to validate the overall proposal, a larger scale demonstrator is also planned to be deployed, with enlarged communication range and real UAVs controlled by FemtoNodes.

Acknowledgments

E. P. Freitas thanks the Swedish Knowledge Foundation and the Brazilian Army for the grant to follow the PhD program in Embedded Real-time Systems at Halmstad University in cooperation with UFRGS in Brazil.

Bibliography

[1] D. A. Schoenwald. AUVs: In Space, Air, Water, and on the Ground. *IEEE Control Systems Magazine*, Vol. 20, No. 6, pp. 15-18, 2000.

[2] D. Culler, D. Estrin, and M. Srivastava. Overview of sensor networks. *IEEE Computer*, Vol. 37, No. 8, pp. 41-49, 2004.

[3] E. P. Freitas, M. A. Wehrmeister, C. E. Pereira, and T. Larsson. Reflective middleware for heterogeneous sensor networks. *Proceedings of 7th Workshop on Adaptive and Reflective Middleware*, ACM, pp. 49-50, 2008.

[4] R. S. Allgayer, M. Götz, and C. E. Pereira. FemtoNode: Reconfigurable and Customizable Architecture for Wireless Sensor Networks. *Proceedings of 10th International Embedded Systems Symposium (IESS'09)*, Langenargen, Germany, pp. 302-309, 2009.

[5] Sun Microsystems. SunSPOT, www.sunspotworld.com

[6] A. T. Erman, L. Hoesel, and P. Havinga. Enabling Mobility in Heterogeneous Wireless Sensor Networks Cooperating with UAVs for Mission-Critical Management. *IEEE Wireless Communications*, Vol. 15, Issue 6, pp. 38-46, 2008.

[7] M. A. Batalin, M. Hatting, and G. S. Sukhatme. Mobile Robot Navigation using a Sensor Network. *IEEE Intl. Conf. on Robotics and Automation*, 2004.

[8] J. S. Kinnebrew, A. Gupta, N. Shankaran, G. Biswas, and D. C. Schmidt. A Decision-Theoretic Planner with Dynamic Component Reconfiguration for Distributed Real-Time Applications. *Proceedings of 8th International Symposium on Autonomous Decentralized Systems*, pp. 461-472.

[9] Y. Jin, Y. Liao, A. A. Minai, and M. M. Polycarpou. Balancing Search and Target Response in Cooperative Unmanned Aerial Vehicle (UAV) Teams', *IEEE Transactions on System, Man, Cybernetics-Part B: Cybernetics*, vol. 36, No. 3, pp. 571-587, 2006.

[10] B. Walter, A. Sannier, D. Reinerss, and J. Oliver. UAV Swarm Control: Calculating Digital Pheromone Fields with the GPU. *In The Interservice/Industry Training, Simulation & Education Conference (I/ITSEC)*, vol. 2005 (Conference Theme: One Team. One Fight. One Training Future), 2005.

[11] MSB Co. web site. Submeter-scale aircraft, http://spyplanes.com

[12] R. Leonard, and J. Drezner. Global Hawk and Darkstar. Santa Monica, California : RAND Corporation, vol. 4, 2002.

[13] E. P. Freitas, M. A. Wehrmeister, C. E. Pereira, F. R. Wagner, E. T. Silva Jr., and F. C. Carvalho. DERAF: A High-Level Aspects Framework for Distributed Embedded Real-Time Systems Design. Springer, pp. 55-74, 2007.

[14] E. P. Freitas, M. A. Wehrmeister, C. E. Pereira, and T. Larsson. Using Aspects and Component Concepts to Improve Reuse of Software for Embedded Systems Product Lines. *Proceedings of 13th Early Aspects Workshop at SPLC-08*, pp.105-112, 2008.

[15] M. E. Bratman. Intention, Plans, and Practical Reason. Cambridge, MA, 1987.

[16] Object Management Group (OMG). Distribution Service for Real-time Systems (DSS) Specification, Vol. 1.2, 2007.

[17] E. Bonabeau, M. Dorigo, and G. Theraulaz. Swarm Intelligence: From Natural to Artificial Systems. *Oxford University Press*, Santa Fe Institute Studies in the Sciences of Complexity, NY, 1999.

[18] P. Gaudino, B. Schargel, E. Bonabeu, and B.T Clough. Swarm Intelligence: a New C2 paradigm with an Application to Control of Swarms of UAVs. *Proceedings of 8th International Command and Control Research and Technology Symposium*, 2003.

[19] J. A. Sauter, R. Matthews, H. V. D. Parunak, and S. A. Brueckner. Performance of Digital Pheromones for Swarming Vehicle Control, *Proceedings of 4th International Joint Conference on Autonomous Agents and Multi-Agent Systems*, ACM Press, pp. 903-910, 2005.

[20] T. Heimfarth, and P. Janacik. Experiments with Biologically-Inspired Methods for Service Assignment in Wireless Sensor Networks, *IFIP Intl Federation for Information Processing*, Vol. 268, Eds. Boston: Springer, pp. 71-84, 2008.

[21] H. Hinkelmann, P. Zipf, and M. Glesner. A Domain-Specific Dynamically Reconfigurable Hardware Platform for Wireless Sensor Networks, *Int. Conf. on Field-Programmable Technology*, pp. 313-316, 2007.

[22] P. Garcia et al. An overview of reconfigurable hardware in embedded systems, *EURASIP J. Embedded Systems*, New York, NY, USA, n.1, pp.13-13, 2006.

[23] S. A. Ito, L. Carro, and R. P. Jacobi. Making Java Work for Microcontroller Applications, *IEEE Design and Test of Computers*, Los Alamitos, Vol. 18, No. 5, pp. 100-110, 2001.

[24] Sashimi Manual, www.inf.ufrgs.br/~lse/sashimi/, 2006.

[25] M. A. Wehrmeister, C. E. Pereira, and L. B. Becker. Optimizing the generation of object-oriented real-time embedded applications based on the real-time specification for Java, *Proceedings of The Design, Automation, and Test in Europe Conference*, Belgium, pp. 806-811, 2006.

[26] J. Lessmann, T. Heimfarth, and P. Janacik. ShoX: An Easy to Use Simulation Platform for Wireless Networks, *Proceedings of 10th International Conference on Computer Modeling and Simulation*, pp. 410-415, 2008.

[27] E. P. Freitas, T. Heimfarth, M. A. Wehrmeister, F. R. Wagner, A. M. Ferreira, C. E. Pereira, and T. Larsson. Using Link Metric to Improve Communication Mechanisms and Real-time Properties in an Adaptive Middleware for Heterogeneous Sensor Networks, *Advances in Information Security and Assurance*, LNCS, 5576, Springer Berlin, Heidelberg, 422-431, 2009.

[28] E. P. Freitas, T. Heimfarth, F. R. Wagner, A. M. Ferreira, C. E. Pereira, and T. Larsson. An Agent Framework to Support Sensor Networks - Setup and Adaptation, *Proceedings of International Multiconference on Computer Science and Information Technology*, pp. 533?540, Mragowo, Poland, 2009.

[29] E. P .Freitas, T. Heimfarth, F. R. Wagner, A. M. Ferreira, C. E. Pereira, and T. Larsson. Evaluation of Coordination Strategies for Heterogeneous Sensor Networks Aiming at Surveillance Applications, *Proceedings of 8th IEEE Sensors*, pp. 591?596, Christchurch, New Zealand, 2009.

[30] C. Bettstetter. On the Minimum Node Degree and Connectivity of a Wireless Multihop Network, *Proceedings of the 3rd ACM International Symposium on Mobile Ad Hoc Networking & Computing*, ACM, New York, NY, USA, 2002, pp. 80-91.

[31] E. P. Freitas, M. A. Wehrmeister, C. E. Pereira, and T. Larsson. Real-time Support in a Reflective Middleware for Heterogeneous Sensor Network. *In Proceedings of the 29th IEEE Real-Time Systems Symposium: Work-in-Progress Proceedings*, Barcelona, Spain, pp. 57-60, December 2008.

Edison Pignaton de Freitas - has a position as Computer Engineer at the Brazilian Army. He is currently a PhD student at Halmstad University, Sweden, and Federal University of Rio Grande do Sul (UFRGS), Brazil, in the area of sensor networks, performing his research in the embedded systems groups of both universities and has published several papers in this area. He got his Bachelor degree in Computer Engineering from the Military Institute of Engineering, Brazil, in 2003, and his MSc degree in Computer Science from UFRGS, in 2007. In 2001 and 2002 he participated in an interchange program for Engineering students between Brazil and France, studying at the Genie Informatique Industriel program of the Institut National des Sciences Appliques, in Toulouse, and performing an internship at the Systems Department of the AIRBUS, working in the A380 project.

Tales Heimfarth - received his PhD degree in Computer Science from the University of Paderborn, Germany, in 2007. His PhD dissertation was entitled "Biologically Inspired Methods for Organizing Distributed Services on Sensor Networks". He received a MSc degree in Computer Science in 2002, with the thesis "Real-time Communication Platform over an SCI Cluster", and a Bachelor degree in Computer Science in 2000, both from UFRGS. Since 2008 he holds a post-doc position at UFRGS, performing research in the area of sensor networks. He is author of several papers published in international conferences in the area of sensor networks, especially with focus on basic software and auto-organizing protocols. He participates of the ShoX project, which is a open-source, event-oriented simulator for ad-hoc networks, led by the University of Paderborn.

Rodrigo Schmidt Allgayer - is currently a PhD student at UFRGS, Brazil, in the area of wireless sensor networks, embedded systems and reconfigurable architectures, performing his research in the Control, Automation and Robotics Group. He got his Bachelor degree in Electrical Engineering in 2005, and his MSc degree in Electrical Engineering in 2009, both from UFRGS. His research interests include wireless sensor networks, reconfigurable architectures, and embedded systems.

Flávio Rech Wagner - received a BSc degree in Electrical Engineering (1975) and an MSc degree in Computer Science (1977), both from UFRGS, Brazil. He received a PhD degree in Computer Engineering from the University of Kaiserslautern, Germany, in 1983. In 1992 and 2002 he held post-doc positions at INPG (Institut National Polytechnique de Grenoble), France. He was invited professor at the University of Tübingen, Germany, in 1994. He is currently professor at UFRGS, position that he holds since 1977, and since 2006 he is the Director of the Institute of Informatics of UFRGS. He has been President of the Brazilian Computer Society, from 1999 to 2003. He is associate editor of the Design Automation for Embedded Systems journal and has been member of more than 30 program committees of international conferences, having published 3 books, 7 book chapters, and more than 120 papers in conference proceedings and journals.

Tony Larsson - became a Member of IEEE in 1990. He received a MEng degree in 1974, a Tech.Lic. 1986 in Computer Systems and a PhD degree in Computer Science in 1989, all at the Institute of Technology at Linköping University, Sweden. He worked for Ericsson AB from 1974 to 2002, in several different positions, such as engineer, manager, and expert (the latter in system design methods and architecture), in areas such as testing, computer aided design, radio network control, and distributed computer systems platforms for dependable telecommunication applications. He then worked for the Swedish defense material administration in the area of network based defense and is since 2003 Professor in Embedded Systems at Halmstad University.

Carlos Eduardo Pereira - received a BSc degree in Electric Engineering from Escola de Engenharia (1987), Brazil, and an MSc degree in Computer Science from UFRGS in 1990. He received a PhD degree in Electrical Engineering from Technische Universitat Stuttgart, Germany, in 1995. In 2000 he held post-doc positions at United Technologies Research Center (UTRC). He is a professor in the Department of Electrical Engineering and Science Computing at UFRGS and since 2009 he is the Deputy Director of the Engineering of UFRGS. He is a senior member of the IEEE and IFAC Technical Committee on Real-Time Programming. Has experience in Electrical Engineering, focusing on Electronic Automation of Electric and Industrial Processes, acting on the following subjects: Real-Time Systems, Object-oriented Programming, Industrial Automation, Industrial FieldBus and Software Engineering.

Armando Morado Ferreira - received BSc (1990) and MSc (1997) degrees in Mechanic Engineering, both from Military Institute of Engineering. He received his PhD degree in Mechanical Engineering from University of Delaware (2001), and MSc and PhD degrees in Military Sciences from Brazilian Army's Command and High-Staff School (2004 and 2009) Lieutenant-Colonel Military Engineer, professor at Military Institute of Engineering and instructor at Brazilian Army's Command and High-Staff School, he has knowledge in Mechanical Engineering with focus in robotization, concentrating his work in the following themes: trajectory generation, non-linear control and unmanned vehicles control and modelling, besides a great interest in the area of Information Science and Technology Management applied to Defence.

Int. J. of Computers, Communications & Control, ISSN 1841-9836, E-ISSN 1841-9844
Vol. V (2010), No. 1, pp. 71-90

Reference Architecture for Collaborative Design

C.Y. Huang, T.T. Yang, W.L Chen, S. Y. Nof

Chin-Yin Huang
Department of Industrial Engineering and Enterprise Information
PO Box 985, Tunghai University, Taichung 407, TAIWAN
E-mail: huangcy@ie.thu.edu.tw

Tung-Ting Yang
HTC Corporation
No.23, Singhua Rd., Taoyuan City, Taoyuan 330, TAIWAN
E-mail: anne7349@hotmail.com

Wu-Lin Chen
Department of Computer Science and Information Management
Providence University, Taichung 407, TAIWAN
E-mail: wlchen@pu.edu.tw

Shimon Y. Nof
School of Industrial Engineering
Purdue University, 315 N. Grant Street, W. Lafayette, IN 47907-2023, USA
E-mail: nof@purdue.edu

Abstract:
Issues and themes of Collaborative Design (CD) addressed by research done so far are so extensive that when running a project of collaborative design, people may lack directions or guidelines to support the whole picture. Hence, developing reference architecture for CD is important and necessary in the academic and the empirical fields. Reference architecture provides the systematic, elementary skeleton and can be extended and adapted to diverse, changing environments. It also provides a comprehensive framework and enables practices implemented more thoroughly and easily. The reference architecture developed in this re-search is formed along three dimensions: decision aspect, design stage, and collaboration scope. There are five elements in the dimension of decision aspect: (1) participant, (2) product, (3) process, (4) organization, and (5) information. The dimension of design stage includes three stages: (1) planning and concepting, (2) system-level design and detail design, and (3) testing and prototyping. The dimension of collaboration scope includes three types of collaboration: (1) cross-functional, (2) cross-company, and (3) cross-industry. Because of the three reference dimensions, a cubic architecture is developed. The cubic reference architecture helps decision-makers in dealing with implementing a CD project or activity. It also serves as a guideline for CD system developers or people involved in the design collaboration to figure out their own responsibility functions and their relations with other members. Demonstration of how to use the reference architecture in developing design collaboration activities and specifying the details for cross-company CD is also provided in this research.

Keywords: collaboration reference model, collaboration support, communication, cross-company design collaboration, design management, design stage, information system.

1 Collaborative Design

1.1 Definition and Issues

Research about collaboration activities has become popular from the 1990s with the emergence of the Internet and following the increasing requirements and benefits of concurrent design [20] [24]. Collaboration covers an extensive domain that includes a large variety of concerns and issues, from product development collaboration to customer relations management, e-service and e-commerce [23] [10] [50] [7] [32]. Specifically, the core arena of the research reported in this article focuses on Collaborative Design (CD) under the key subject of product development collaboration. *Collaboration is a process in which entities share information, resources and responsibilities to jointly plan, implement, and evaluate a series of activities to achieve a common goal. It implies a group of entities that work together and enhance the capabilities of each other. Collaboration involves mutual engagement of participants to solve a problem together* [31]. Related research topics are summarized in Table 1.

Table 1 Research Topics of Collaborative Design

Topic of CD	References
Methods and models developed for CD	Klein [21]; Brelinghoven [4]; Witzerman and Nof [52]; Huang and Nof [17] Khanna et al. [20]; Jin and Lu [18]; Abdalla [1]; Mahesh et al.[29]; Panchal et al.[34]; Wang et al. [51]
Collaborative environment developed to facilitate or support CD	Lara and Nof [25]; Hao et al. [13]; Li and Su [26]; Pappas et al.[36]; Robin et al. [39]; Alvares and Ferreira [2]; Tian et al. [45]; Tseng et al. [47]
Knowledge/ intelligent capital of CD	Thomas and Baker [44]; Chiang et al. [8]; Huang [16]; Panchal et al. [35]; Chira et al. [9]
Management issues of CD (including social/organizational aspects)	Noori and Lee [33]; Ceroni and Velasquez [6]; Pilemalm et al. [37]; Shiau and Wee [41]; Slimani et al. [42]
Applications or case examples of CD	Tang [43]; Hao et al.[13]; Ceroni and Nof [7] Trappey and Hsiao [46]; Chung and Lee [12]

1.2 Development Process

Traditional Computer-Aided-Design (CAD) system improves the efficiency and effectiveness of design information interchanging and speeds up the concept design visualization process, but the way of it only supports individual designers in their design activities and does not support the trend and business surroundings. Evolved collaborative CAD system conquered this problem enabling multiple designers to work on a design together, such as system called Computer Supported Cooperative Work (CSCW). With IT such as Internet support and Web 2.0 etc., web-based CAD systems advance to facilitate extensive application of CD. Web-based CAD system such as C-DeSS [22] integrates web-based multimedia tools with web-based model display. This communication platform allows multiple users from geographically distributed locations to share their design concepts, innovations, and models through Internet technology [27].

1.3 Components

An interaction model between factors influencing the design system was developed by Robin et al. [40]. The key interest of their model is to present and define all elements which influence a design system and interactions between them. The objective is to support engineering management according to structuring of decisions making. From this fundamental map, it describes the context of product design system and integrates all these elements, including actor, process, organization, product, knowledge, and environments. Eight kinds of links within all of the six elements compose research contents of design topics.

1.4 Challenges

Reference architecture is a necessity among organizations that collaborate at multiple levels. Examples such as the Boeing 787 Dreamliner project design complexities and challenges could be cited to illustrate the need for a useful, well-defined collaboration architecture to avoid costly mistakes in the design process. Eight issues that need to be conquered regarding CD framework are [35]:

1. Adaptability to network architecture changes or malfunction,
2. Usability on heterogeneous platforms with heterogeneous operating systems,
3. Heterogeneous languages for different agents (semantic interoperability),
4. Capability to transmit message and data changes (semantic interoperability),
5. Rapid configuration of the product realization environment (considering reasons like time-to-market, and so on),
6. Minimizing the impact of agent service changes,
7. Readiness for future expansion, and
8. Readiness for discrepancy of process information.

The above eight issues can be used as a checklist for the completeness of the reference architecture developed. In addition, the reference architecture is viewed as an approach to establish collaboration requirements planning, the first principle of Collaborative Control Theory, CCT [32].

2 Reference Architecture and Factors

Functions of reference architecture include: (1) showing the evolution of development, (2) guiding all the parties from each of the different disciplines involved, (3) incorporating several different views, (4) presenting a method for the breakdown of all system functions to their inherent generic functions and tasks [38]. To develop reference architecture for CD, factors of three layers are considered: *elementary factors*, *qualified factors*, and *advanced factors*.

2.1 Elementary Factors

Two substances taken into consideration are industry characteristics and the infrastructure of information system. The former indicates that involved participants focus more on the product and process aspect in improving efficiency and effectiveness of design process. The latter one means level or completeness the infrastructure of IT can support. The IT infrastructure primarily aims at fundamental information system such as PDM or Collaborative Product Definition Management (CPDM) and adoption of software such as CAD or CAPP to facilitate the design process.

2.2 Qualified Factors

Based on the elementary factors, four CD qualified factors are considered in this research:

1. Data repository or database with profound contents: It stores the entities of the virtual world together with their attributes, and the database should include functions such as information storage and data retrieval.

2. 3D virtualization tool for product design: Tool such as Dynamic UI Generation, CAD-CAE System, and 3D editor. This tool is a virtual product manager that provides users with a local as well as collaborative desktop GUI.

3. Connecting systems for communication and interchanges: These are systems connected with one another via LAN or Internet. Therefore Internet, web browser or process diagram tool may be practical enablers of CD, used to model a product realization process and then invoke the available agents integrated into the collaboration support framework. The cooperative support platform is another example of a connecting system [28].

4. Powerful Application Service Provider (ASP) Server or core application: This service provides functionality for dynamic and stable loading, processing and saving the design-related objects of the virtual world, which may originate from diverse collaborating designers and their agents.

2.3 Advanced Factors

With basic equipment as elementary and qualified factors, discussions of CD will be advanced to another connected frame. The CD Cube, which is composed of three key dimensions, as defined in the next section, focuses more on the advanced factors which enable the systematic implementation of effective CD. In the decision dimension, five deciding elements are specified; in the design stage dimension, different characteristics of CD follow the classification of each design stage; finally, in the collaboration scope dimension, models of CD are defined and discussed with various boundaries set for each scope.

3 Reference Architecture for Collaborative Design

First, a conceptual depiction of the contents and dimensions to model a CD activity is introduced (Figure 1). To structure a CD reference architecture, the situation of proposed CD is first clarified, including the actors involved, the target objectives for collaboration, available competences in the CD activity, and available resources. Next, CD process model, corresponding CD organization type to carry out the project, information interchange mechanism, and the collaborative design stage and scope which are involved have to be specified. Considering all of these issues, the CD framework containing taxonomy of each composing elements, principles for CD initiator, and other subjects can be developed.

3.1 CD Framework

Based on the previously discussed background, a 3*5*3 CD framework is developed (Figure 2). With which, one can figure out the current status of collaboration, or see if all relevant details have already been taken into considerations. For example, once the CD of product concept design is decided to be originated across two extended companies, then the circumstances taken into considerations should fall on to the left bar of middle surface of the cube. Furthermore, if the CD is carried out through the product design lifecycle from planning to production ramp-up, then the middle level surface would be the proposed boundary.

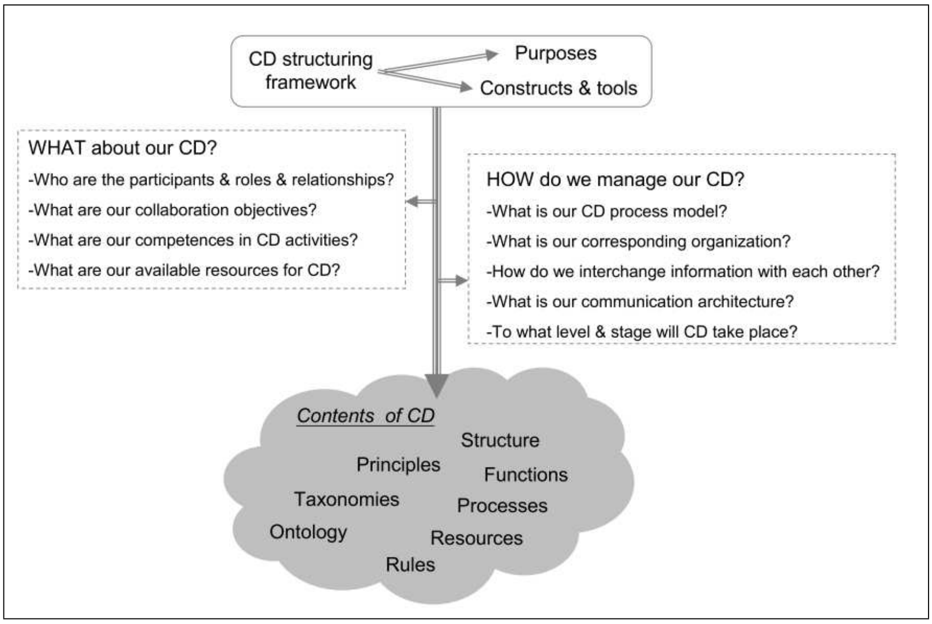


Figure 1: Content of CD Reference Architecture

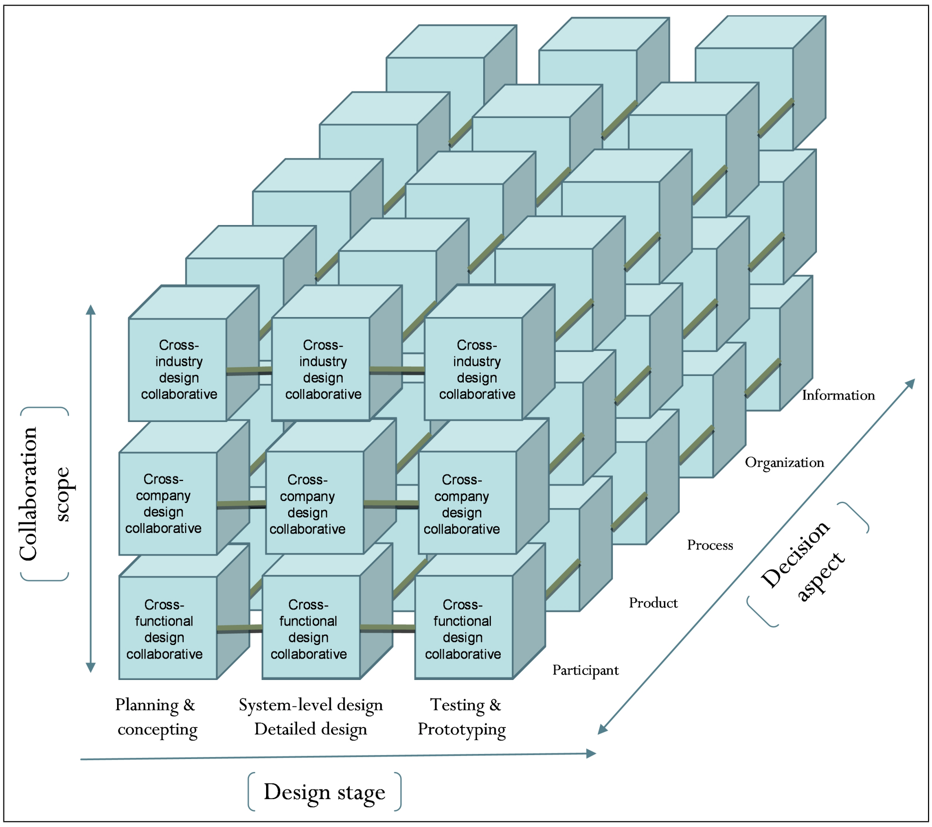


Figure 2: Reference Architecture of Collaborative Design

3.2 Decision Aspect

In the dimension of decision aspect, five elements are specified in a CD project: participant, product, process, organization, and information.

Participant

With the horizontal axis of deepness and vertical axis of extensiveness in collaborative activities, a two-by-two matrix is produced. This matrix clarifies the participant types involved in CD in a view of primary actors for decision making (Figure 3 (a)). Based on the two axes, four types of CD participants, which are major, niche, compatible, and minor players, are specified.

Major Player represents deep connection of cooperators and advanced extensiveness of the proposed collaboration. This type of participant focuses on fundamental CD or NPD (new product development/ design), and is prone to adopt the manner of integrated corporation, joint venture (JV), or extended enterprise (EE).

Niche Player represents deep connection of cooperators but limited extensiveness of proposed collaboration. This type of participant possesses certain dominant or specific skills or technology in which primary actors are interested. For this reason, corresponding collaboration target would be professional New Product (NP) or under the mature competence market applying the manner of VE (virtual enterprise), VO (virtual organization), etc.

Compatible Player represents shallow connection of cooperators but advanced extensiveness of proposed collaboration. This type of participant is capable of providing non-fundamental but customized or domain-specific NP with manner of contractual agreement, which is the same manner applied by Minor Player (below). The Compatible and Minor Players, however, can be discriminated by the product or service they collaboratively design. Consider the example of Foxconn(R), although Foxconn seems like a Minor Player providing fundamental OEM jobs to many other potential competitors for their customers, it performs with solid status in the market for its own specific or customized capability to fulfill the needs of customers. Hence, it is considered as a Compatible Player.

Minor Player represents shallow connection of cooperators with limited extensiveness of proposed collaboration. This type of participant has the most potential risks for being replaced by competitors easily for lack of its own sustainability with others or for lack of close relations or extraordinary capability. Minor Players usually participate in general/routine NPD or take the OEM role to their ODM/OBM customers.

Product

The vertical axis represents the market status consisting of current market and new market, and the horizontal axis represents the current technology and new technology. First, it is hypothesized that the product is in the status of current market using current technology. Then, four main product types can be defined in view of product development projects under this structure, as shown in Figure 3 (b). When corporation strategy is the current market-current technology, incremental improvement to existing products would be the product type. When corporation strategy is the new market-current technology, derivative of existing product platforms would be the product type. When corporation strategy is the current market-new technology, new product platforms or new product generation/upgrade would be the product type. In this region, it is possible to be still in the same product family but using different product platform in product realization. On the other hand, new product generation/upgrade type can also be introduced in this corner, for it also to be under the same product family with different or update technology. When corporation strategy is the new market-new technology, fundamentally new products would be the product type.

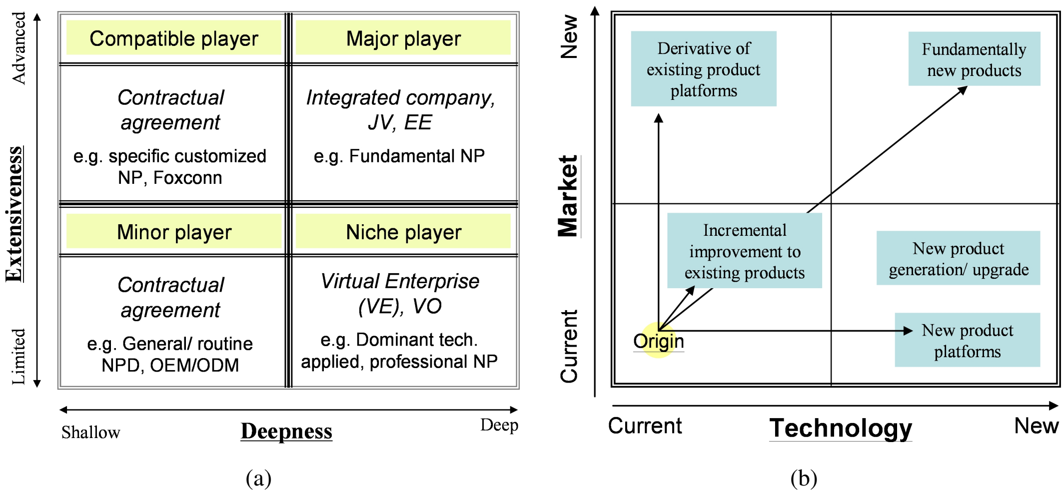


Figure 3: (a) CD participant classification matrix, (b) Strategy of product type

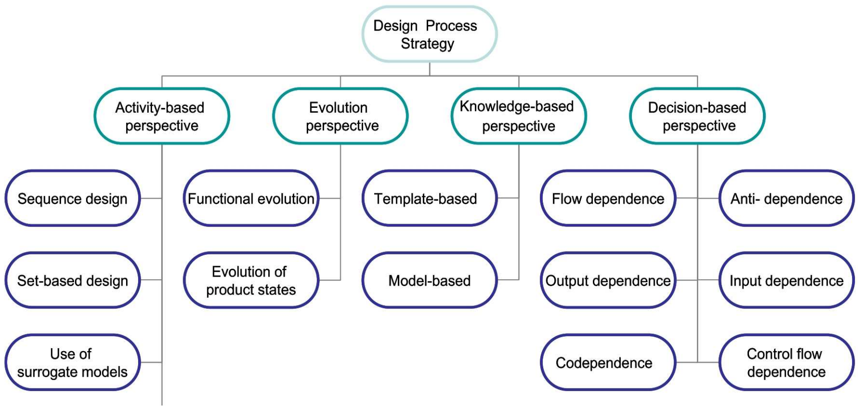


Figure 4: Process Taxonomy

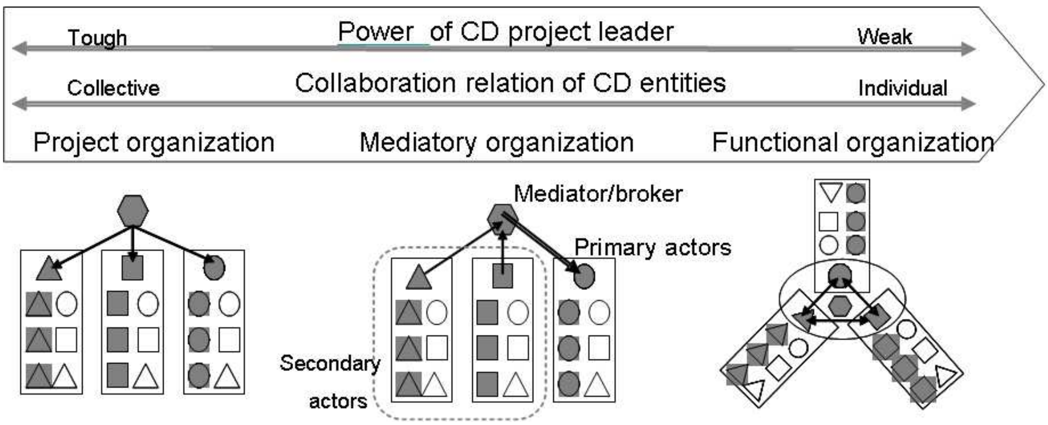


Figure 5: Organization type taxonomy

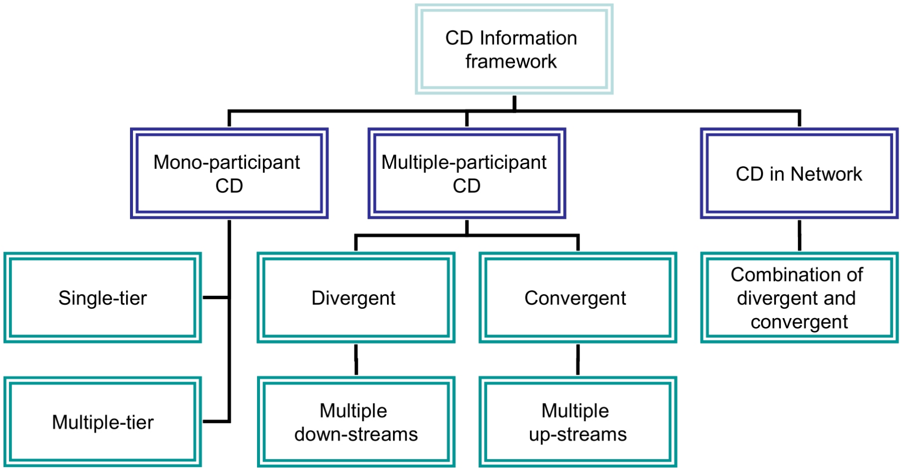


Figure 6: CD Information framework

Process

After reviewing research regarding process types of design, product development and collaboration, an overview aspect of design process strategy is developed. Under the design strategy domain, four perspectives are concluded: activity-based, evolution, knowledge-based, and decision-based. These four comprise the design process taxonomy of CD (Figure 4).

Organization

A structured organization can facilitate design communication and consequently contribute to the success of the design project [11]. With different types of CD cooperation, organizational changes are necessary to be adopted for better fitness of design process rearrangement and coordination between entities of collaboration. Combining theories proposed by Hayes [14], Ulrich and Eppinger [48] and Chiu [11], taxonomy of CD organization is presented in Figure 5. Three organization categories in the taxonomy can explain considered CD situations and provide a reference. Extended from classical project management organization classification proposed by Ulrich and Eppinger [48], the taxonomy of organization types includes project, mediatory, and functional organization (Table 2).

Information

By utilizing the organization and participants structures as the interpreting settings, CD can take place in three circumstances, namely mono-participant CD, multiple-participant CD, and network CD (Figure 6).

3.3 Design Stage

Design process can be defined as networks of information transformations from one state to another. The state of information refers to the amount and form of that information that is available for design decision-making [35]. Stages/processes of product design and development can be basically defined as process of Idea/Requirement, Concept Design, Design Build, System Integration, Product Validation, Manufacture Validation, and Mass Production. Ulrich and Eppinger [49] also have similar view on stages/processes of product design and development. They define six steps: planning, concept development, system-level design, detail design, testing and refinement, and production ramp-up. Three significant types of design processes are proposed by Holt [15]:

1. The analytical design process: used when there is little uncertainty about the alternatives, and the outcome is only a modification of something that already exists.

2. The iterative design process: This process is best suited to medium-risk projects such as radical improvements and adopted innovations.

3. The visionary design process: in which the problem cannot be defined precisely and is, perhaps vague at best.

A more precise viewpoint on activities of design is given by Borja de Mozota [3]. They consider a creative process, divided into three main phases: an analytic stage of widening the observation field; a synergistic stage of idea and concept generation; and a final stage of selecting the optimal solution. The creative process comprises of five phases, each of which has a different objective and corresponds with the production of increasingly more elaborate visual outputs.

Table 2 Organization type taxonomy

	Organization type		
Characteristics	Project organization	Mediatory organization	Functional organization
Candidate of CD leader	From one of the entities	Primary actor	From one of the entities or the third party
Hierarchical power	Tough, demanding, powerful	Clear hierarchical relation	Weak/Harmony hierarchical relations
Final decision maker	CD leader	CD leader (primary actor)	All of the entities
Power of entities	Dependent to each other in decision making	Independent but dependent to each other in cross-decision making	Independent but has to be responsible to individual belonging organization
Collaborative relation	Collective-oriented	Partnership or supporting relationship	Individual-oriented
Interface of the entities	Mutually agreed protocols	Mediator or broker or software agent	Process integration

Planning and Concepting

In the first design stage, Planning and Concepting, collaboration is basically focusing on the preparation stage ahead of the precise designing practical tasks prior to the next stage. Issues of this stage are primarily about the coordination related to CD project/process planning and product concept development. These issues include design tasks such as identification of user needs, technical factors, the diverse requirements of the operating environment, product exploration, and concept development. Detailed, interleaved tasks include initial design idea collection, information pooling from the present market, confirmation of customer needs, assessment of new technologies and needs, assessment of market demands, application for forming the design project, gaining permission for triggering following activities, designating product platform and architecture, and proposals of product concept designing.

Keinonen [19] demonstrates the concept design process which consists of a series of stages, containing information gathering, brainstorming, scenario creation, concepts formation, formalization, evaluation, and final integration with project planning. Basically, the planning and concepting stage is composed of three sub-categories, namely background research, concept generation, and concept evaluation.

System-Level Design and Detailed Design

In the second design stage, subjects being focused are primarily about the specified design tasks regarding product structure and architecture. CD issues of this stage would be complex for various kinds of coordination related to the design activities occur among the participants, products, process, and organization, through information systems. The second stage contains major design tasks such as product material and technology defined, new product design specifications, system-level design, detailed designs, resource allocation, selected concept design confirmation, current product analysis and market survey, etc. Because various interactions and coordination occur in the CD activities, conflicts and arguments could occur. Based on Liu et al. [27], there are three types of conflicts: (1) conflicts of a single task (2) conflicts between tasks, and (3) conflicts of system level with different disciplines.

The first type of conflict includes two indications. One indicates the conflicts occur when the design task with different objectives, properties, requirements, or even sharing same resources such as machines for making components, material for manufacturing, etc. The other indicates the conflicts occur when participants updating design description simultaneously, i.e, *codependence* [20]. For instances, when both of sales A and B are trying to book the same original material C for their own WIP, WIP a and WIP b, with the constraints of limited original material C, conflicts may occur if there is no decision rule or user authority restrictions of the material booking system.

The second type of conflict is related to coordination between activities of design process or agent communication. A typical example could occur for the engineering systems of engine design and air-conditioning of Boeing 747. The interaction between these two systems should be taken into consideration together during system-level design to avoid the situation of incompatibility. The solution of the conflicts have been proposed by Khanna and Nof [20] based on a canonical model for the task dependences. The third type of conflict is related to the communication issues of standalone systems. This type of conflict has two indications: interface incompatibility and loss of associated information under the situation of design changes [30]. For example, in the same system-level design stage, electrical engineers and mechanical engineers are both working on the product architecture modeling. They have to deal with the interface incompatibility of geometrical information and the capability of interpreting meta-information of interchanges.

Testing and Prototyping

In the third stage, major interleaved activities involve preempt production, manufacturing validation, marketing experiments, prototyping production, product evaluation, product refinement, etc. This is the last step along the product design process, but does play an important role to fill up the existing gaps between design and manufacturing stages. To facilitate the integration of different experts and enhance the efficiency of the iterative phases, prototypes are used as cost-efficient visual models. The use of virtual prototypes is especially important in the early phases of product development, to enable time- and cost-efficient decision making [5]. Prototyping can be categorized, however, into physical and digital activities [5].

Physical prototyping is well-known by the name rapid prototyping (RPT) for making it possible to produce physical artifacts directly from CAD model. The most common techniques today, including stereo-lithography (STL), selective laser sintering (SLS), solid ground curing (SGC), and fused deposition modeling (FDM), are mainly used to produce design or geometrical prototypes. To accelerate the development process, technical and functional prototypes are of great importance. Rapid tooling offers the possibility of building functional prototypes, and it is possible to build tools rapidly and inexpensively for prototypes in parallel with the product development process.

Physical prototypes, however, are often time- and cost-intensive and thus need to be reduced to a minimum. By integrating CAD technologies, rapid prototyping, virtual reality, and reverse engineering, prototypes can be produced faster and less costly than before. The digital demonstration allows early modification and optimization of the prototype. Furthermore, it leads to cost-saving increase in the variety of prototypes. Additionally, faults concerning fabrication of the product itself can be detected in the early development phase and thus be eliminated without unnecessary expenditures.

An important component of digital prototyping is the digital mock-up (DMU), a purely digital test model of technical product. The objective of the DMU is the availability of multiple views of product shape, function, and technological coherences. This DMU forms the basis on which the modeling and simulation (testing) can be performed and communicated for an improved configuration of the design. The primary digital design model is called the virtual product. The idea is to test the prototype regarding design, function, and efficiency before producing the physical prototype. An enormous advantage of the DMU is the shortening of iteration cycles. Employing the DMU considerably reduces the time-to-market.

3.4 Collaboration Scope

The last dimension in the CD reference model is the collaboration scope. Dealing with the scope of CD, we put it into three segments: cross-functional CD, cross-company CD, and cross-industry CD.

Cross-Function

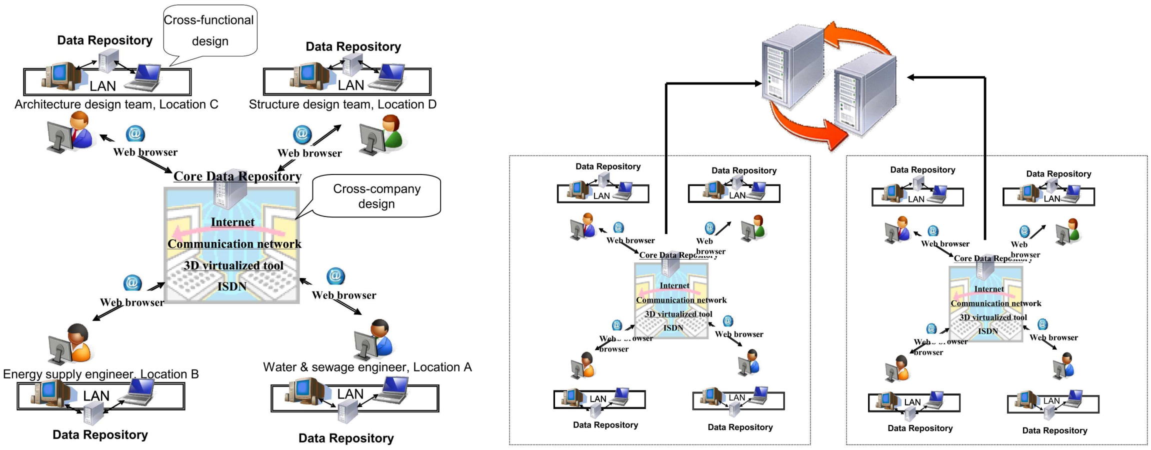
Cross-functional CD, which is commonly utilized, represents CD activities taking place within the collaboration scope of one individual enterprise. CD may occur within the design collaboration team composed of designers only, or among multiple-discipline departments/functions containing designers and non-designers. In general, essential participants of cross-functional CD may include project manager, designers, project supporters (non-designers), and system administrator/coordinator.

Cross-Company

In cross-company CD, CD activities occur between two enterprises which may have relation with one another, or are completely individual. Liu et al. [27] define the main characteristics of CD within the scope of cross-company as containing groups of designers, manufacturers, suppliers, and customer representatives, which can be seen as the extended case of cross-functional CD. It should be noted in their definition that the actors involved in this scope of CD are counted by groups rather than as individuals. Vertical integration and horizontal integration are two views to examine in the cross-company relationships.

In view of vertical integration, CD could be carried out by two enterprises within the same supply chain; members of CD may be responsible for different tasks of the supply chain such as CD of two cooperators with the relationship of ODM and OBM. Or, CD could be carried out by two enterprises of two unrelated supply chains. In this case, CD can be executed by members of the same function in two different supply chains, or by members of complementary functions of two different chains. In more complicated situations, members of CD may include actors of the same and complementary functions from both the same chain and different chains.

In view of horizontal integration, members of CD do not only belong to different supply chains but also have different industry attributes. These differences render them unable to have complementary collaborative relationships. They may get together not for permanent cooperation but to capture certain instant opportunities, which apply with the concept of virtual enterprises.



(a) (b)

Figure 7: (a) Communication reference framework (b) Communication reference framework: cross-industry collaboration by a third-party repository

Cross-Industry

In cross-industry CD, the model is regarded as the extended horizontal integration CD of cross-company. Considering the product City Storm(R) bicycle, which is the CD achievement of Giant(R) and DEM(R), as an example, Giant (a bicycle builder) and DEM (an artifact design company) belonged initially to two different industries, but they cooperate with each other and both focus on their own core competence. Eventually, through CD, both companies create a successful collaborative business.

The relevant communication reference framework is defined to indicate the elementary components of a fundamental CD (Figure 7 (a)). In the framework, there are four CD teams in this network, which are located around the globe, and each of them is responsible for specific tasks of the CD project, which are architecture design, structure design, energy supply engineering, and water & sewage engineering. To configure CD activity, each of the CD members should be equipped with technology and infrastructures including data repository, web browser, process diagram tool, interface mapping/integration tool, and Dynamic User Interface (UI) Generation (application tools). The scenario described above is an example of CD taken with cross-company scope.

In Figure 7 (b), the cross-industry collaboration can be regarded as the extended case of cross-company collaboration. The most significant difference between Figures 7 (a) and (b) of CD is the interfaces or collaboration platforms on which they communicate with each other. In Figure 7 (b), participants of CD which come from two different supply chains of diverse industry characteristics may collaborate with each other under the CD environment supported by a third-party repository, which represents certified criteria for communication among cooperators. In addition to the certified criteria for communication, integration, and exchange, the third-party repository may protect the privacy and security of the cooperating parties which belong to different companies and different industries.

4 Cross-Company Application of the Reference Architecture

To demonstrate the CD application of the reference architecture, a cross-company reference model is developed. By horizontally slicing the reference architecture in Figure 2, a cross-company reference model (Figure 8) may be developed for a specific application domain.

In the participant element, two main relation types of CD participants are designer-to-designer and

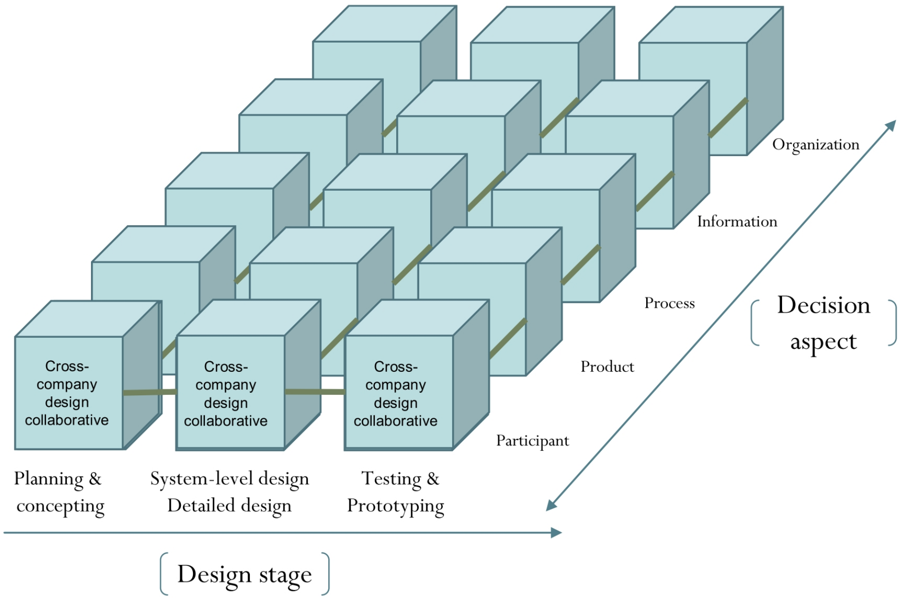


Figure 8: Cross-Company Reference Model

designer-to-non-designer. In the former case, the involved participants can be classified by the CD participant classification matrix in Figure 3 (a). For instance, if the case of the target CD product is Dominant Technology Required, then Niche Player of possessing particular techniques would be included. Examples are an up-stream IC-chip manufacturer such as Intel, or a transmission corporation such as Shimano (R), whose products are the most critical components of bicycle, to all ODM, even OBM such as Giant.

In the product element, CD with complex and multiple-disciplines team members is more commonly found to be a cross-company CD. Fundamentally New Products and New Product Platforms are those cases. Considering Product Structure and Product Extensiveness, product of cross-company CD falls into the Product Structure of Integration, not limited to the Product Extensiveness (i.e., specialized and general usage of the current product). For example, even with product types of highly complex design architecture, such as airplanes, jet engine is not necessarily designed by CD. Such products can consider continuing the use of fine-tuned existing technology or extended by present platform, but without costly cross-company CD. On the other hand, complex product types such as ERP software packages or operating systems (e.g., Microsoft Vista), while they belong to General Products, they may still have the need for cross-company CD to ensure enough multidisciplinary technologists involved in the product development to make the new product as complete and innovative as possible.

In the organization element, Project, Mediatory, and Functional organizations are potentially applicable. For information element, there are two alternative directions. One is in view of horizontal integration CD, and the other is vertical integration CD. The former one represents roles involved in performing the same or complementary functions within the same supply chain tier. For example, a system-level design can be carried out by three participants who belong to different corporations. The three corporations are around the globe and each takes the responsibility of product structure design, architecture design, and system integration assessment, respectively. All of them perform the tasks of the system-level design jobs at the same design stage and tier (not up-stream and down-stream relationship) but collaborate with each other. The latter one can apply the CD Information framework introduced in Figure 6, namely multiple-participant CD and CD in network. For instance, CD such as Derivative of Existing Product Platforms can be carried out by co-working with the manufacturing partners. If the participants include members from both multiple up-stream and down-stream companies, then it is the case of Network

Cross-Company CD.

4.1 Case Study

In the case of Cross-Company CD, suppose the two dimensions of reference model are followed, decision aspect and design stage, to demonstrate how the reference model can help guide and present the CD activities among collaborating companies. In the following sections, the corresponding decision aspects are depicted for each design stage from incubation to verification stage under the scenario proposed by Chung and Lee [12].

Incubation Stage (Planning and Concepting)

Participants & Product : in the first stage, participants involved are only the customer company, the injection molding company, and the mold company. Partnership selection process may take place within each of these three parties. In this case, however, the product is prone to be Product Platform Improved or Upgrade. Therefore, the injection molding company here may play the role of Compatible Player to the customer company, whereas the mold company may be the Minor Player.

Process : In Product Planning and Concepting, although the mold company is actively involved in the activity, only the customer and injection molding companies are responsible for the product concept development and specification definition. If it is the case of OEM, the injection molding company does gain the specific design drawing and specification information from the costumer company. If it is ODM or OBM, the role the injection molding company would play is relatively more important as specification/code definer. In the latter case, communication between the customer and the injection molding companies would be more frequent and intense.

Organization : In this stage, the mold company does not play significant role yet in the CD activity. Two main participants, the customer and the injection molding company cooperate more as a Functional organization, each playing their respective duty without dominating each other.

Information : As mentioned by Chung and Lee [12], information exchange platform is formed by XML, and the CD system architecture follows web-based mechanism. In the incubation stage, the information framework would follow the situation of Multiple-Tier (here it is two-tier) Mono-Participants (one company in each tier) CD.

Proceeding Stage (System-Level Design and Detail Design)

Participants : During the system-level and detail-level design, the mold company plays an important role in this stage. In this stage the injection molding and the mold companies have closer interaction. From the point of view of the customer, the mold company may now be considered a Minor Player, since it only executes the manufacturing activities. But it could be considered a Compatible Player by the injection molding company.

Product : Products in this stage would be plastic parts manufactured by the mold company with the specification and design drawings provided by the customer company, but may be verified or trimmed by the injection molding company which is authorized by the customer company to do so.

Process : through CORBA and platform formed by XML, the customer and the injection molding companies may cooperate with each other to analyze and verify the specification and design drawings to consider their manufacturability, quality, cost, etc.

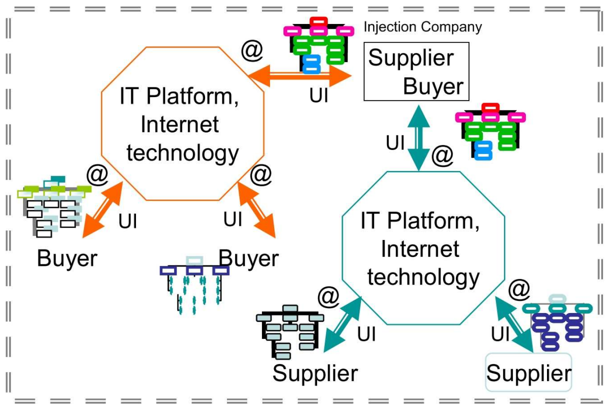


Figure 9: Information framework of CD in network

Organization : Organization type of this case under the system-level and detail-level design stage would fall into the Mediatory organization type, and the injection molding company plays the Mediator role.

Information : During the design static part, technologies used include XML to share information among design participants and application programs, RAMDES (RApid Mold Design Expert System) for mold design tool, PCIA (Parameter Connectivity Information Administrator) for evaluating the validities of parameters used in different dimensions and companies, client module and server module. In addition, the information system does provide a display of design information served by similar functions such as PDM or PLM system.

During the design operation, the CD system in this case relies on the four qualified factors mentioned in Section 2.2, which are data repository supported by XML, 3D virtualized tool for product design among various clients, connecting system by PCIA, and ASP sever to verify CAD system customized for mold design. Besides, the situation faced here will fall into the category of CD in network. The injection molding company plays the middle layer character facing both the layers of the customer (buyer) and the mold (supplier). Although Chung and Lee [12] do not deliver the detailed information regarding corresponding authority of each participant using the information sharing platform, the information gathered from the customer and the mold companies should be filtered by the injection molding company for its own purposes, or for other confidential considerations. Information exchange media (IT platform and Internet technology) would follow the mechanism depicted in Figure 9. Figure 9 indicates that the injection molding company plays both buyer and supplier roles relative to the mold company and to the customer company, respectively. It could co-design with the suppliers/buyers through IT platform and Internet technology.

Verification Stage (Testing and Prototyping)

Participants : In this stage, participants involved are basically the manufacturing and business divisions of the mold company, and may include some analysis and design divisions of the injection molding company. Meanwhile, design and analysis office of the mold company would support them to assure successful mold production. From another perspective, however, manufacturing and business offices play secondary roles to support design and analysis divisions to fulfill the order from the injection molding company.

Product : Product at this stage is basically the improved design objects and routine product development tasks that should be carried out by the manufacturing office of the mold company and be delivered on-time.

Process : The divisions of the mold company have to cooperate with each other to fulfill the quantity and quality specifications of the request orders from the injection molding company. The fulfillment of the process can be delivered by the mutually agreed protocols.

Organization : Organization applied here may be Project organization (Table 2).

Information : Divisions within the mold company can co-work on the same IT system or intranet platform to interchange information within different departments. Data repository such as product tracking system or PDM system and connecting system such as intranet communication platform may provide a channel for information sharing and update among the divisions. Although concurrent engineering could be implemented in the CD, certain sequential activities still take place within the CD activities.

This case study has demonstrated the application of the CD reference architecture for the case of three participating companies. The three companies together, and each for its own purposes, can systematically use this CD reference architecture to analyze their respective and mutual requirements for the purpose of most effective collaboration. It is anticipated that effective collaboration would reduce conflicts and costs, support the quality of the design on which they collaborate, and also foster further productive collaboration among them. The case study demonstrates the proposed reference architecture can successfully conquer the challenging issues (Nos. 1, 2, 4, 5, 6 and 7) in Sections 1.4.

5 Conclusions and Future Works

Reference architecture for collaborative design has been defined and implemented. This reference architecture provides a new perspective for analyzing requirements for successful and effective CD, and dimensions to be considered when initiating a CD project. This architecture was designed to help decision-makers in dealing with implementation of a CD project or activity. The sliced CD reference model also serves as a guideline map for collaborating software system developers or people involved in the design collaboration to figure out their own functions and current progress of the group. Three dimensions included in the reference architecture are:

1. **Decision aspect**: five elements, including Participant, Product, Process, Organization, and Information.

2. **Design stage**: Planning and concepting, system-level design and detail design, and testing and prototyping

3. **Collaboration scope**: Cross-functional CD, Cross-company CD, Cross-industry CD

The architecture by demonstrating a case study already meets several major challenges facing collaborative design, including:

1. Adaptability to architecture changes;
2. Usability on heterogeneous information systems;
3. Capability to transmit message and data changes;
4. Rapid configuration;
5. Minimizing the impact of service changes;
6. Readiness for future expansion.

An application of the CD reference architecture specified in the scope of cross-company is also developed in this research. This research demonstrates how to use the reference model (as a portion of the reference architecture) in developing design collaboration activities and how to specify the details of CD contents. As for future research, details specifications of each cube and the cube-cube relationships in the reference architecture are important issues. Detailed checking to understand whether the CD reference architecture meets all the challenges presented in Section 1.4 are also required.

Acknowledgments
The authors are thankful for financial support from research project NSC 98-2221-E-029-019, National Science Council, Taiwan. Special thanks to Purdue University for pioneering the Purdue Enterprise Reference Architecture, PERA, which inspired the development of this research.

Bibliography

[1] J.A. Abdalla. A communication model for structural design objects. II: Performatives and protocols. *Advances in Engineering Software*, 37 (6): 393-405, 2006.

[2] A.J. Alvares, J.C.E. Ferreira. A system for the design and manufacture of feature-based parts through the Internet. *nternational Journal of Advanced Manufacturing Technology*, 35 (7-8): 646-664, 2008.

[3] B. Borja de Mozota. Design management: using design to build brand value and corporate innovation. New York: Allworth Press, 2003.

[4] A. Brelinghoven. Cooperative concurrency control for design environment. Proceedings EURO-DAC '95. Brighton, Great Britain: IEEE Computer Society Press, 1995: 308-313.

[5] H.-J. Bullinger, J. Warschat, J. Leyh, T. Cebulla. Planning and integration of product development. In: Salvendy G., editor. Handbook of industrial engineering: technology and operations management. New York: Wiley, 2001:1283-1294.

[6] J.A. Ceroni, A.A. Velasquez. Conflict detection and resolution in distributed design. *Production Planning & Control*, 14 (8): 734-742, 2003.

[7] J.A. Ceroni, S.Y. Nof. Task parallelism in distributed supply organizations: a case study in the shoe industry. *Production Planning & Control*, 16(5): 500-513, 2005.

[8] T.A. Chiang, A.J.C. Trappey, C.C. Ku. Using a knowledge-based intelligent system to support dynamic design reasoning for a collaborative design community. *International Journal of Advanced Manufacturing Technology*, 31 (5-6): 421-433, 2006.

[9] O. Chira, C. Chira, T. Roche, D. Tormey, A. Brennan. An agent-based approach to knowledge management in distributed design. *Journal of Intelligent Manufacturing*, 17 (6): 737-750, 2006.

[10] C. Chira, O. Chira. A Multi-Agent System for Design Information Management and Support. *International Journal of Computers, Communications and Control*, 1, Suppl. Issue, 2006.

[11] M.-L. Chiu. An organizational view of design communication in design collaboration. *Design Studies*, 23 (2): 187-210, 2002.

[12] J. Chung, K. Lee. A framework of collaborative design environment for injection molding. *Computers in Industry*, 47 (3): 319-337, 2002.

[13] Q. Hao, W.M. Shen, Z. Zhang, S.W. Park, J.K. Lee. Agent-based collaborative product design engineering: An industrial case study. *Computers in Industry*, 57 (1): 26-38, 2006.

[14] R.H. Hayes, S.C. Wheelwright, K.B. Clark. Dynamic manufacturing: creating the learning organization. New York: Free Press, 1988.

[15] K. Holt. The nature of design process. In: Oakley M., editor. Design management: a handbook of issues and methods. Oxford, UK ; Cambridge, MA., USA: Blackwell Reference, 1990:195-198.

[16] C.C. Huang. A multi-agent approach to collaborative design of modular products. *Concurrent Engineering-Research and Applications*, 12 (1): 39-47, 2004.

[17] C.Y. Huang, S.Y. Nof. Development of integrated models for material flow - Design and control - a tool perspective. *Robotics and Computer-Integrated Manufacturing*, 14 (5-6): 441-454, 1998.

[18] Y. Jin, S.C.Y. Lu. Agent based negotiation for collaborative design decision making. *CIRP Annals-Manufacturing Technology*, 53 (1): 121-124, 2004.

[19] T. Keinonen. Introduction to concept design. In: Keinonen T., Takala R., editors. Product concept design: a review of the conceptual design of products in industry. London: Springer, 2006:1-31.

[20] N. Khanna, J.A.B. Fortes, S.Y. Nof. A formalism to structure and parallelize the integration of cooperative engineering design tasks. *IIE Transactions*, 30 (1): 1-15, 1998.

[21] M. Klein. Integrated support for cooperative design coordination: managing processes, conflicts and memories. In: Nof S.Y., editor. Information and Collaboration Models of Integration. Boston: Kluwer, 1994:435-459.

[22] M. Klein. Capturing Geometry Rationale for Collaborative Design. it 6th Workshop on Enabling Technologies Infrastructure for Collaborative Enterprises (WET-ICE '97), pp. 24-28, 1997.

[23] H.S. Ko, S.Y. Nof. Design of Collaborative e-Service Systems. In: Salvendy, G., W. Karwowski, editors, Introduction to Service Engineering, John Wiley & Sons, Inc., 2009.

[24] A. Kusiak. Concurrent engineering: Issues, models, and solution approaches. In: Dorf, R.C, A. Kusiak editors. Handbook of Design, Manufacturing, and Automation, Wiley Inter-science, 1994: 35-49.

[25] M.A. Lara, S.Y. Nof. Computer-supported conflict resolution for collaborative facility designers. *International Journal of Production Research*, 41 (2): 207-233, 2003.

[26] J.S. Li, D.Z. Su. Support modules and system structure of web-enabled collaborative environment for design and manufacture. *International Journal of Production Research*, 46 (9): 2397-412, 2008.

[27] X. Liu, S. Raorane, M. Leu. A Web-based Intelligent Collaborative System for Engineering Design In: Li W.D., McMahon C., Ong S.K., Nee A.Y.C., editors. Collaborative Product Design and Manufacturing Methodologies and Applications. Springer London, 2007:37-58.

[28] Y. Luo. Cooperative Design in Building Construction In: Li W.D., Ong S.K., Nee A.Y.C., McMahon C., editors. Collaborative Product Design and Manufacturing Methodologies and Applications. London: Springer, 2007:93-108.

[29] M. Mahesh, S.K. Ong, A.Y.C. Nee. A web-based framework for distributed and collaborative manufacturing. In: Li W.D., Ong S.K., Nee A.Y.C., McMahon C., editors. Collaborative Product Design and Manufacturing Methodologies and Applications. London: Springer, 2007:137-148.

[30] F. Mervyn, A.S. Kumar, A.Y.C. Nee. Leveraging design process related intellectual capital-a key to enhancing enterprise agility. In: Li W.D., Ong S.K., Nee A.Y.C., McMahon C., editors. Collaborative Product Design and Manufacturing Methodologies and Applications, London: Springer, 2007: 71-90.

[31] S.S. Msanjila, H. Afsarmanesh. Trust analysis and assessment in virtual organization breeding environments. *International Journal of Production Research*, 46: 1253-1295, 2008.

[32] S.Y. Nof. Collaborative control theory for e-work, e-production, and e-service. *Annual Reviews in Control*, 31(2): 281-92, 2007.

[33] H. Noori, W.B. Lee. Collaborative design in a networked enterprise: The case of the telecommunications industry. *International Journal of Production Research*, 42 (15): 3041-3054, 2004.

[34] J.H. Panchal, H.J. Choi, J.K. Allen, D.L. McDowell, F. Mistree. A systems-based approach for integrated design of materials, products and design process chains. *Journal of Computer-Aided Materials Design*, 14: 265-293, 2007.

[35] J.H. Panchal, M.G. FernĂĄndez, C.J.J. Paredis, J.K. Allen, F. Mistree. Leveraging Design Process Related Intellectual Capital. A Key to Enhancing Enterprise Agility. In: Li W.D., Ong S.K., Nee A.Y.C., McMahon C., editors. Collaborative Product Design and Manufacturing Methodologies and Applications. London: Springer, 2007:201-208.

[36] M. Pappas, V. Karabatsou, D. Mavrikios, G. Chryssolouris. Development of a web-based collaboration platform for manufacturing product and process design evaluation using virtual reality techniques. *International Journal of Computer Integrated Manufacturing*, 19 (8): 805-814, 2006.

[37] S. Pilemalm, P.O. Lindell, N. Hallberg, H. Eriksson. Integrating the Rational Unified Process and participatory design for development of socio-technical systems: a user participative approach. *Design Studies*, 28 (3): 263-288, 2007.

[38] H. Li, T.J. Williams. A Formalization and Extension of the Purdue Enterprise Reference Architecture and the Purdue Methodology, Report No. 158, Laboratory for Applied Industrial Control, Purdue University, http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.25.6659&rep=rep1&type=pdf, 1994.

[39] V. Robin, B. Rose, P. Girard. Modelling collaborative knowledge to support engineering design project manager. *Computers in Industry*, 58 (2): 188-198, 2007.

[40] V. Robin, S. Sperandio, S. Blanc, P. Girard. Interactions modeling between factors influencing performance of the design process. International Conference on Engineering Design, ICED 2005. Melbourne, 2005.

[41] H.Y. Shiau, H.M. Wee. A distributed change control workflow for collaborative design network. *Computers in Industry*, 59 (2-3): 119-127, 2008.

[42] K. Slimani, C.F. Da Silva, L. Medini, P. Ghodous. Conflict mitigation in collaborative design. *International Journal of Production Research*, 44 (9): 1681-1702, 2006.

[43] D.B. Tang. An agent-based collaborative design system to facilitate active die-maker involvement in stamping part design. *Computers in Industry*, 54 (3): 253-271, 2004.

[44] J.P. Thomas, K.D. Baker. Knowledge representation for design. 1991 IEEE International Conference on Decision Aiding for Complex Systems. Charlottesville, VA 1991:1827 - 1832.

[45] L. Tian, J.Z. Chen, Q.Y. Wang, W.T. Hao, B.S. Tong. CoDesign Space: a collaborative design support system in a network environment. *International Journal of Computer Integrated Manufacturing*, 20 (2-3): 265-279, 2007.

[46] A.J.C. Trappey, D.W. Hsiao. Applying collaborative design and modularized assembly for automotive ODM supply chain integration. *Computers in Industry*, 59: 277-287, 2008.

[47] K.C. Tseng, H. Abdalla, E.M. Shehab. A Web-based integrated design system: its applications on conceptual design stage. *International Journal of Advanced Manufacturing Technology*, 35 (9-10): 1028-1040, 2008.

[48] K.T. Ulrich, S.D. Eppinger. Product design and development. McGraw-Hill/Irwin: Boston, 2004.

[49] K.T. Ulrich, S.D. Eppinger. Product design and development. McGraw-Hill Higher Education: Boston, 2008.
[50] J.D. Velasquez, S.Y. Nof. Collaborative e-Work, e-Business, and e-Service. In: Nof, S.Y., editor, Handbook of Automation, Springer, 2009, 1549-1576.
[51] C. Wang, K. Li, H. Ghenniwa, W. Shen, Y. Wang. Real Time Distributed Shop Floor Scheduling: An Agent -Based Service-Oriented Framework. In: Li W.D., Ong S.K., Nee A.Y.C., McMahon C., editors. Collaborative Product Design and Manufacturing Methodologies and Applications. London: Springer, 2007:175-197.
[52] J.P. Witzerman, S.Y. Nof. Tool Integration for Collaborative Design of Manufacturing Cells. *International Journal of Production Economics*, 38 (1): 23-30, 1995.

Chin-Yin Huang is Associate Professor in the Department of Industrial Engineering and Enterprise Information at Tunghai University, Taiwan. He also serves as Director of the Executive Master Program for Health Administration. He received his Ph.D. from Purdue University, USA. His research interests include agent-based integrated systems, distributed production planning, process optimization in plastic injection molding, and medical data analysis. His recent book, *Industrial Engineering and Management* was published in 2004. He is a member of CIIE, IFAC, IFPR-APR, and SOCOLNET.

Tung-Ting Yang is a Manager of Product Manufacturing in HTC. She received her BA degree from National Chengchi University and an MSIE degree from Tunghai University. Her research interests include new product development and financial management.

Wu-Lin Chen is Assistant Professor in the Department of Computer Science and Information Management, Providence University, Taiwan. He received his Ph.D. from Purdue University. His research interests include supply chain optimization, stochastic systems, and manufacturing process optimization.

Shimon Y. Nof is Professor of Industrial Engineering and director of the NSF-industry supported PRISM Center (Production, Robotics and Integration Software for Manufacturing & Management) at Purdue University. He has held visiting positions at MIT and universities in Chile, EU, Hong Kong, Israel, Japan, and Mexico. Received his B.Sc. and M.Sc. in Industrial Engineering & Management (Human-Machine Systems), Technion, Haifa, Israel; Ph.D. in Industrial & Operations Engineering (Production Analytics), University of Michigan, Ann Arbor. Among his honors: Inaugural member of Purdue's Book of Great Teachers (1999); the Engellerger Medal for Robotics Education (2002), Fellow of IIE, recent President and former Secretary General of IFPR, recent Chair of IFAC CC-Mfg. & Logistics Systems. Dr. Nof has published over 400 articles on production engineering and information technology, and is the co-author/editor of ten books, including the *Handbook of Industrial Robotics* (Wiley, 1985; 2nd Edition 1999) and the *International Encyclopedia of Robotics* (Wiley, 1988), both winners of the "Most Outstanding Book in Science and Engineering" award by the Association of American Publishers, *Industrial Assembly* (Chapman & Hall, 1997, co-authored with H-J Warnecke and W. E. Wilhelm), and the *Handbook of Automation* (Springer, 2009).

Int. J. of Computers, Communications & Control, ISSN 1841-9836, E-ISSN 1841-9844
Vol. V (2010), No. 1, pp. 91-105

Design of Protocols for Task Administration in Collaborative Production Systems

H. S. Ko, S. Y. Nof

Hoo Sang Ko
School of Industrial Engineering,
Purdue University 315 N Grant St,
West Lafayette, IN 47907, USA
E-mail: ko@purdue.edu

Shimon Y. Nof
School of Industrial Engineering,
Purdue University 315 N Grant St,
West Lafayette, IN 47907, USA
E-mail: nof@purdue.edu

Abstract:
Customer-focused and concurrent engineering service systems process tasks more effectively as a result of the power of collaboration among multiple participants. In such environments, however, complex situations might arise that require decisions beyond simple coordination.Task Administration Protocols (TAPs) are designed as a control mechanism to manage complex situations in collaborative task environments. This article presents the design of TAPs for collaborative production systems in which tasks are performed by the collaboration of multiple agents. Three component protocols are found to constitute TAPs and are triggered at appropriate stages in task administration: 1) Task Requirement Analysis Protocol, 2) Shared Resource Allocation Protocol, and 3) Synchronization & Time-Out Protocol. A case study with TAPs metrics for task allocation in a collaborative production system is investigated to compare performance under TAPs, and under a non-TAP coordination protocol (which is considered to be simpler). In terms of task allocation ratio, the case study indicates that performance under TAPs is significantly better (up to 10.6%) than under the non-TAP coordination protocol, especially under medium or high load conditions. The advantage of TAPs can be explained by their design with relatively higher level of collaborative intelligence, addressing more complex control logic compared with non-TAP coordination protocols.
Keywords: rules, figures, citation of papers, citation of books, examples.

1 Introduction

In highly distributed networked systems, tasks arrive at agents in the system, and then are processed by the collaboration of agents that have the skills and capabilities to process the entire task or parts of it. The achievement of goals depends on how effectively each individual agent coordinates tasks with others to solve the given problems in a collaborative manner. There have been numerous research studies regarding collaborative problem solving in a decentralized way by multiple agents, e.g., optimization by collaborative swarm intelligence [22], and collaborative negotiation in global supply networks [29]. As systems and systems-of-systems become more complex, especially when attempting to improve performance via collaboration, the dynamic complexity of interactions among agents requires a higher level of collaborative intelligence. Thus, there should be an effective control mechanism to rationalize, coordinate, and harmonize tasks by exchanging information and decisions among collaborating participants.

Coordination protocols have been developed to achieve effective control by managing a given, known type of dependence among tasks, e.g., producer/consumer relationships, as defined in coordination theory [1, 2]. Application of coordination protocols in collaborative service systems includes multi-robot systems [3, 30], supply chain management [4, 29], and agent-based manufacturing systems [10, 28], to name a few. The operation of these systems is usually represented by collaborative work in multi-agent systems [6, 23], and various coordination protocols have been developed to provide the agents in the systems with the rules and the interaction procedures on how to cooperate and coordinate effectively. Many coordination protocols have been developed based on the framework of contract net protocol (CNP) [7], a market-based approach in which tasks are announced to agents and allocated to the agent that provides the best bid. Such decentralized approaches have been developed to achieve effective resource allocation in distributed systems and have shown good performance in terms of communication and computational efficiency, scalability, and flexibility [5, 8, 9], compared to centralized approaches [2, 7].

A critical limitation of coordination protocols, however, is that there exist events that cannot be handled by merely coordination protocols. These situations include:

1. The priority of tasks must be dynamically changed; and
2. There are complicated situations requiring decisions beyond simple coordination.

For example, consider tasks that are close to their deadline and need to be handled quickly with high priority over other, less urgent tasks. Resources need to be allocated for the urgent tasks first in order to meet the deadline. Later, if a more urgent task, e.g., resource failure requiring repair, or an emergency task, is generated, the previously scheduled tasks may be preempted by the new one so that it will not cause a significant damage to the system. Coordination protocols, however, do not handle such situations since they only allow fixed task allocation, i.e., the winner to whom the task was allocated is deemed to commit itself to the awarded task until it is completed. In order to overcome such limitations of coordination protocols, the protocols need to be able to indentify repeatedly the current state of the system and take proper actions to deal with complicated conditions. Such protocols, which assume the responsibility of making decisions actively and triggering timely actions so that the overall system's performance can be further improved, are defined as Task Administration Protocols (TAPs) [11, 20].

The purpose of this article is to design TAPs and analyze TAPs' advantage over non-TAP coordination protocols. This is more important when more distributed and decentralized networks of activities require collaboration, which increases the complexity of the problem. In this article, the design of TAPs for effective task allocation and administration is developed by addressing three basic elements in task administration: 1) Task, 2) Resource, and 3) Time. Three main reasons require the development of TAPs with abilities beyond conventional coordination protocols, as follows.

(1) Dynamic tasks re-prioritizing

When tasks enter the system, they need to be analyzed and sorted based on their priority, so that important or urgent tasks can be handled first. Priority of tasks is an essential consideration in task administration. With coordination protocols, priority is evaluated by managerial decision or prior assignment based on task type [1], or by pricing based on market-based approach [8, 9]. Tasks with higher priority receive more monetary value, so the priority is reflected in the level of funding. The limitation in the coordination protocols is that the priority of a task is predetermined and fixed by its price and deadline in the task description at the moment the task arrives. While this approach may be adequate in some cases, in real situations, however, the priority may change due to emergent events. Thus, dynamic change of priority along time should be considered.

(2) Resource availability

Appropriate resources need to be allocated by considering the task requirements, the status of resources and their schedules. A bidding procedure can be processed to obtain good resource allocation solutions using the CNP framework (e.g., [8, 10]).

(3) Time-out conditions
A resource serving a task needs to be monitored. If a task occupies (engages) a resource excessively for certain reasons, e.g., resource failure, or abnormal tasks, the task access to the resource should be timed-out so that the overall performance of all tasks' services does not seriously degrade by the given task over-occupying the resource. In order to prevent such wasteful situations, a time-out protocol with an appropriate threshold needs to be used. In several studies and applications, time-out protocols have been developed and proved to be effective [12, 14].

In order to deal with the three basic elements of administration, TAPs' design model requires three components as developed in this study. The rest of the article is organized as follows. Section 2 describes the definitions and structures of TAPs with the three component-protocols. Each of the components is described in detail in section 3. In order to illustrate the advantage of TAPs over simple coordination protocols, a case study of applying TAPs for a collaborative production system is presented in section 4. Finally, section 5 concludes the article.

2 Task Administration Protocols (TAPs) in A Coordination Network

An active middleware model of coordination network [15, 16] is used to define the model of TAPs. A coordination network (Co-net) is a network of autonomous agents that enables collaboration among the agents. Each agent exchanges information and execution to achieve its objectives while it tries to maximize the system (or system-of-systems) goals. The Co-net is defined as:

$$\mathit{Co\text{-}net} = < \pi, \tau, \alpha, \sigma > \tag{1}$$

where π: a set of agents in *Co-net*; π = {A1, A2,..., $A_{|\pi|}$};τ a set of tasks to be processed in *Co-net*; τ = {T1, T2,..., $T_{|\tau|}$};α: a set of activities performed by π to fulfill τ; $\alpha = \{Activity_1, Activity_2, ..., Activity_{|\alpha|}\}$; and σ: a set of control mechanisms used by π.

Coordination is defined as the process of managing dependencies between activities [1]. In the *Co-net*, dependence (δ) is a relationship between τ, π, and α, and defined as follows:

$$\{T_i \times \pi(T_i) \times \alpha(T_i) | T_i \in \tau\} \rightarrow \delta \in \triangle \tag{2}$$

where π(Ti): a set of agents who can process Ti; α(Ti): a set of activities or sub-tasks required to process Ti; $\triangle$: a set of dependence types. A particular dependence δ can be decided by analyzing tasks and evaluating certain factors in the system, e.g., task priority and time-out threshold.

In the *Co-net*, coordination protocols (CP) are designed to manage a particular dependence (δ) between tasks and resources. A coordination protocol is defined as follows:

Definition 1.

$$P_j = \{\delta, I, R, PA, S\} \in CP \tag{3}$$

where P_j: a particular coordination protocol for handling T_i (j=1,..., J; J is the number of CP); I: a set of initiators which initiate coordination activities; $I \subset \pi$; R: a set of responders which respond to initiators' requests; $R \subset \pi$ and $I \cap R = \emptyset$; PA: a set of parameters of the coordination protocol; S: a set of decision logic in each transition stage of the protocol; and CP is a set of coordination protocols.

TAPs are defined as a set of protocols which assume the responsibility of making decisions actively and triggering timely actions so that these decisions and triggers can improve the coordinated performance [11, 20]. When a task (T_i) or, in a broader sense, an event (e) occurs in the system, TAP mechanism should identify the dependencies between tasks and activate the appropriate protocol (P_j) between agents with proper parameter values (PA) for the protocol. In the Co-net, therefore, a TAP is defined as follows:

Definition 2.

$$TAP = \{\sigma \cup P_j | \sigma(e|\delta \subset \triangle) \to P_j\} \tag{4}$$

where $\sigma$is a mechanism to select an appropriate protocol to handle the upcoming events; e is an event (e $\in$ E; E: a set of events), e.g., arrival of a task, failure in processing a task, etc. In other words, the TAP is a control mechanism to handle an event by analyzing dependence between tasks and involved initiators and responders, as well as the parameters and decision logic required to deal with the event. According to the type of event and systems of interest, different task administration schemes need to be applied.

Theorem 1.(TAP-Set). CP is a subset of but not identical to TAP.

Proof. According to *Definition 1*, CP is a set of P_j's which handles tasks under certain dependence $\delta \subset \triangle$. By *Definition 2*, TAP includes P_j's as its elements but also contains the mechanisms (σ) to dynamically activate appropriate P_j to handle the upcoming events e based on the current dependence $\delta \subset \triangle$. Moreover, CP cannot handle all types of events in task administration shown in Table 1, unlike TAP.

Theorem 2. (TAP-Performance). Better TAP yields better performance than CP.

Proof. Let θ be a performance metric. Since CP can handle only predefined δ, even though the dependence changes from δ_t to δ_{t+1} along time t, P^t (δ_{t+1}) = P^{t+1} (δ_{t+1}), where P^t is a protocol activated at time t. Since TAP can trigger a more appropriate protocol and its parameters under the current dependence and event, one can find P^{t+1} such that $\theta(P^t\ (\delta_{t+1})) < \theta\ (P^{t+1}\ (\delta_{t+1}))$. Therefore, the overall performance $\sum_t \theta\ (P^t\ (\delta_t))$ under TAP (P^t variable) will be better than under CP (P^t fixed) along time. This will be illustrated in a case study in section 4.

There are three general elements in task administration: task, resource, and time. The general administration elements, their properties and protocol solutions, which has been found from existing literatures, are presented in Table 1. TAPs are composed of three component-protocols, each of which will be activated by the TAP mechanism under certain dependence and deals with the events corresponding to an administration element. When a task arrives at the system and is inserted to task queue, the task requirement analysis protocol (TRAP) is activated to analyze the task and assign its priority and dependence to other tasks and resources. Upon arrival at task queue, a task must be assigned to the best resource, which is decided by the shared resource allocation protocol (SRAP), based on the current status of resources. While processed, a task may occupy a resource excessively, preventing other tasks from being processed by the resource. If the time taken in the resource is beyond a certain threshold, the task needs to be returned to task queue so that other tasks can be processed at the resource. Sometimes, a task cannot be performed by its due date under the current schedule. In this case, the task may need to seek a resource which can complete the task by preempting the other task currently being processed at the resource. These procedures are controlled by the synchronization & time-out protocol (STOP). Each of the three protocols is explained in the subsequent sections.

The theorems presented above are illustrated with three examples in Table 2. In the examples, both TAPs and CPs are used to control multi-agent interactions in certain distributed production systems, but TAPs are more intelligent because of inclusion of one or more TAP components which are missing in CPs. In all the three examples, the performance under TAPs is better than under CPs as they are designed to dynamically handle complex tasks. Even though the TAPs include all three components, there could be a difference in performance between different TAP designs, as in the first example [21]. The three component protocols are explained in the next section.

Table 1: Administration elements, their related events and dependence, and TAP solutions and examples

Admin. Elements	**Task** (τ)	(Resource (R)	**Time (t)**)
Events (e)	Task arrival	Resource allocation	Excessive process time, urgent tasks
Dependence Analyses (δ)	1. Prioritize tasks 2. Identify task requirement/dependence (δ)	1. Optimal/effective allocation 2.Decentralized decision making	1. Synchronization 2. Uncertainty in processes & products
Protocols P_j	Task Requirement Analysis Protocol (TRAP)	Shared Resource Allocation Protocol (SRAP)	Synchronization & Time-Out Protocol (STOP)
Examples [Ref's]	Pricing of tasks [8-10, 17-18]; Priority assignment protocol [13,21]	Market-based resource allocation [8-10,16-18]	Time-out protocol [12-14]

Table 2: Illustration of TAP-Set and TAP-Performance

Example $[Ref's]$	TAP	TAP Component	CP	Performance Metrics (θ)	Results
[21]	TestLAN protocol 1) TAP1: adaptable 2) TAP2: non- 2) TAP2: non-adaptable	TRAP, SRAP, STOP	TestLAN protocol w/ FCFS	waiting time, flowtime	θ(TAP1) $>$ θ(TAP2) $>$ θ(CP)
[14]	Time-out protocol	STOP	Non-timeout protocol	flow time, service time	θ(TAP) $>$ θ(CP)
[16]	Viability-based Resource allocation protocol	SRAP	Resource allocation protocol w/o viability	profit, number of un-allocated tasks	θ(TAP) $>$ θ(CP)

3 Components of Task Administration Protocols

3.1 Task Requirement Analysis Protocol (TRAP)

A key objective in task administration is to allocate incoming tasks to appropriate resources continuously. When this allocation is needed, TRAP, which is a component-protocol of TAPs, is activated to handle the event. This protocol finds the dependence between tasks by their priority relationships. Whenever a new task T_i arrives, a task agent (TA) calculates the priority of T_i and previous tasks still in the system at time t by using a dynamic task priority evaluation function (pf(T_i, t)). If the priority of T_i is less than the priority of previous tasks still in the system, it is just added to the end of the task queue as in First In, First Out regime. Otherwise, the tasks in the queue need to be sorted by their current relative priority. From time to time, the task currently being processed may need to be preempted by an urgent task with relatively higher priority.

Each task has different requirements, such as the type of task, quantity (volume), deadline, estimated cost, etc. In general, a task can be defined as follows:

$$T_i = < type_i, gty_i, dd_i, v_i, PR_i(t) > \tag{5}$$

where $type_i$ is the class of T_i that requires a certain skill of a resource agent (RA); qty_i is the task amount involved in T_i that engages the capacity of a RA; dd_i is the latest time by which T_i must be served by an RA; v_i is the estimated value per unit of T_i; and PR_i(t) is the priority of T_i at time t. PR_i(t) represents the relative importance of T_i and the task that has higher priority should be served first. PR_i(t) is dynamically evaluated by using the priority evaluation function pf(T_i, t). Priority evaluation function needs to be of different form to correspond with specific applications. In its generic form of supply-demand networks, the priority evaluation function is defined as:

$$pf(T_i,t) = w_1 s_i + w_2 \frac{gty_i(v_i - c_i)}{\sum_j gty_j(v_j - c_j)} + w_3\{1 - \frac{dd_i - t}{\sum_j (dd_j - t)}\} + w_4(1 - \frac{gty_0^t}{gty_o}) + w_5 \sum_k (1 - pe_k) \tag{6}$$

where w_n: weight of each factor ($0 \le w_n \le 1$; $\sum w_n = 1$; n = 1, 2,..., 5); j: index of a task previously assigned to the agent; 0: index of the current task; s_i = 1 if $type_i = type_0$, 0 otherwise; c_i: estimated unit cost to perform T_i by a server; qty_0^t: remaining quantity of task T_0 at t; k: index of tasks that will be decommitted due to T_i $(PR_i(t) > PR_k(t))$; pe_k: penalty of decommitment of T_k; $0 \le pe_k \le 1$. The implication of the five factors is as follows:

1) If T_i is the same type of task as the current task, its priority is high since setup is minimized or not needed. In some case, however, the opposite is required.

2) The relatively larger T_i's quantity and profit (value) are, the higher its priority.

3) A task relatively closer to its deadline has higher priority.

4) If the current task is not finished yet but almost is close to being done, it has a relatively higher priority.

5) After its priority re-assignment,$T_i$may cause some tasks, which have already been assigned to resources, to be decommitted if they have a lower priority. The penalty due to the decommitment needs to be considered for fair evaluation of T_i's priority.

To sum up, TRAP is triggered upon arrival of a new task at a TA, which receives tasks and identifies task type and analyzes task requirements, i.e., due date, sub-tasks and required resources. TA assigns a priority value to each of the tasks based on the priority evaluation function. Tasks will be sorted and re-sorted in task queue by their relative priority. After sorting its queue, TA announces the tasks to the RAs who are capable of processing the tasks, and a resource will be allocated to each task one by one by SRAP, which is explained in the next section. The overall procedure in TRAP can be summarized as follows:

1)Task agent (TA) receives a task T_i.

2)Calculate priority of tasks by Eq. (6).

3)Sort the tasks in task queue by PR_i(t).

4)Activate SRAP.

3.2 Shared Resource Allocation Protocol (SRAP)

After tasks are analyzed and their priorities are assigned, SRAP is activated to find the best resources for the tasks. Each of the resources in the system is managed by a resource agent (RA). TA announces the first task (with the highest priority) in task queue to RAs which are capable of processing the task. Each RA calculates the bid based on expected waiting time in the queue for the resource. TA collects the bids and selects the RA with the best bid (the lowest cost). The steps in SRAP are as follows:

1) TA announces T_i to RA_r where r is the index of RA which is capable of processing T_i.

2) RA_r sorts its queue including T_i by the priority of the tasks in the queue.

3) RA_r calculates the bid for T_i by using the following equation:

$$b_{r^i} = c_r \sum_{k=1}^{m_r} \mu_{pt_k} + cs_r \tag{7}$$

where b_{r^i} is the bid by RA_r for T_i; c_r is the cost of RA_r per unit time; k is the index of tasks in the queue of RA_r (k = 1,..., m_r);μ_{pt_k} is mean processing time of T_k; and cs_r is setup cost of RA_r, which is added only if the next task is of different class from the previous task.

4) RA_r submits b_{r^i} for T_i to TA.

5) TA selects RA_{r^*} where r^* = arg min_r b_{r^i}.

6) TA assigns T_i to RA_{r^*}. T_i enters the queue of RA_{r^*} with expected cost $b_{r^{*i}}$.

The policy applied in the above protocol pursues earliest completion time, which is reasonable when the effective cost of tasks is higher compared to the effective (variable) cost of resources. If the effective cost of resources is higher, relatively high utilization of resources may be preferred. In this case, a different policy, i.e., minimizing the total idle time of required resources, needs to be applied.

3.3 Synchronization and Time-Out Protocol (STOP)

From time to time, the current task being served by a resource needs to be timed-out and return to task queue of TA. This situation is triggered and handled by STOP, which is activated in the following cases.

A. Excessive resource occupation

Since a task with faults needs to be reworked by the resource, it may occupy the resource excessively even while idling it, and cause other tasks to be delayed. STOP checks if the current task uses the resource more than certain time-out threshold, which is calculated as follows [13]:

$$to_i = \mu_{pt_i} + 2\sigma_{pt_i} \tag{8}$$

where to_i is the time-out threshold for T_i, and σ_{pt_i} is standard deviation of processing time for T_i. (Two standard deviations are assumed for simplicity here, but another coefficient can be used.) Even if the task has been occupying the resource beyond the time-out threshold, the task remains at the resource's service in the following cases:

1) No other task is waiting in the queue.

2) The current task T_i may be late if timed-out. If dd_i - t $<$ ept_i, where t is the current time and ept_i is the extra processing time needed for T_i, T_i remains in the resource.

B. Preemption by an urgent task

Even though the current task Ti has been served by the resource for less than the time-out threshold, a task Tk in the queue may be late if it is not served until the current process is completed. This problem may be solved by preempting Ti and serving the urgent task Tk first. The protocol logic can be summarized as follows:

1) Check $st_k = dd_k$ - t - μ_{pt_i} of the T_k waiting in the queue of the resource, where st_k is the slack time for T_k, i.e., the remaining time until dd_k.

2) If $st_k < 0$ and $dd_i > ept_i$, stop T_i and process T_k.

3) Once T_k is completed, resume T_i.

To sum up, TAPs are composed of three inter-related protocols: TRAP, SRAP, and STOP. Upon arrival of tasks, TRAP is activated to analyze the task requirements and assign their priority. Next, SRAP is activated to select the best resources based on their current workload. During processing tasks, STOP is activated to monitor if the current task in service needs to be timed-out because of its excessive use of the service or preempted by another urgent tasks. The overall protocol logic is shown in Figure 1. The following case study illustrates their application.

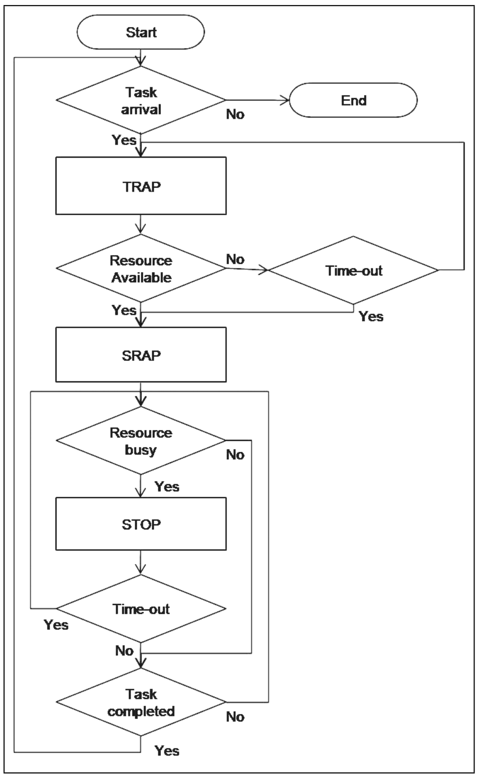


Figure 1: Overall logic of task administration protocols with three component-protocols

4 A Case Study: Collaborative Production Systems

In order to analyze the advantages of TAPs over non-TAP coordination protocols, this section illustrates a case study of TAPs application for task administration in a collaborative production system. A collaborative production system is defined as a distributed network of production resources in which resources can communicate and coordinate with each other to process tasks in the system in an effective and efficient manner, based on collaborative resource sharing. A good example of collaborative production systems is TestLAN (Testing Local Area Network), which is a local area network integrating distributed test operations in manufacturing facilities [13]. TestLAN was developed to increase throughput and reduce the waiting time for testing by integration and communication of distributed testing servers and clients. In TestLAN, products are tested by shared resources, and those with faults are reworked and retested until all the faults are eliminated. Efficient testing resource allocation is difficult since the testing process is highly variable due to design changes, new quality control requirements, and occasionally faulty manufacturing processes.

In collaborative production systems like TestLAN, priority of tasks can be evaluated by their slack time. In TestLAN, assume only the third term in Eq. (6), regarding due date of the task, is effective in priority evaluation, and for simplicity assume every task is of the same importance and the objective is to complete as many tasks as possible in time. Hence, priority of a task in TestLAN is evaluated by a time-based cost of each task as follows:

$$st_i = dd_i - E[tt_i] \tag{9}$$

where tt_i is testing time of T_i. $E[tt_i]$ can be calculated by using the following equation [13]:

$$E[tt_i] = E[pt_i + \sum_{r=1}^{rn_i} rt_{ir}] = \mu_{pt_i} + \mu_{rn_i} \cdot \mu_{rt_i} \tag{10}$$

where pt_i is processing time of T_i; rn_i is number of reworks for T_i; rt_{ir} is rework time for r-th faults occurring in T_i; and μ_{pt_i} , μ_{rn_i} , and μ_{rt_i} are mean of processing time, number of rework, and rework time for the tasks of the same class as T_i, respectively. In order to reflect the rework in TestLAN, Eq. (7) and (8) need to be modified as follows:

$$b_r^i = c_r \cdot \sum_{k=1}^{m_r} (\mu_{pt_k} + \mu_{rn_k} \cdot \mu_{rt_k}) + cs_r \tag{11}$$

$$to_i = \mu_{pt_i} + 2\sigma_{pt_i} + \mu_{rn_i}(\mu_{rt_i} + 2\sigma_{rt_i}) \tag{12}$$

TAPs for the TestLAN function as follows. TRAP assigns a higher priority to a task with a lower slack time. Tasks are sorted in task queue by their priority and a resource who submits the best bid will be allocated to each task one by one by SRAP. In SRAP, each testing resource agent can sort the tasks in its own queue plus the new task to find the best bid. During processing the current task by each resource, time-out conditions are checked by STOP and the previously assigned tasks may be preempted if any of the two time-out preemption conditions are met. For this case study, two TAPs are developed: TAP1 and TAP2. Both TAP1 and TAP2 are composed of the three component-protocols. TAP1 considers, however, only time-out condition B in section 3.3 above, while TAP2 considers both time-out conditions A and B. A coordination protocol, CP, which is considered to be a subset of TAP, is used to compare performance under the various protocols. The logic of CP is similar to the one of SRAP, except the procedure of re-sorting tasks in resources' queue in SRAP is not included, as typical in traditional coordination protocols.

A simulation analysis is designed with two classes of tasks with different levels of urgency. Tasks are generated randomly and processed by three servers. The simulation parameters are shown in Table 3. Performance under the three protocols is measured by 1) task allocation ratio, TAR, and 2) weighted TAR, WAR, defined as follows:

$$TAR = 1 - \frac{\overline{T}}{|\tau|} \tag{13}$$

Table 3: Simulation parameters

Parameter	Value
Number of servers	3
Task class	Normal: dd_i = 600 sec, pt_i = 60 sec (80%) Urgent: dd_i = 180 sec, pt_i = 60 sec (20%)
Interarrival time	low: exp(μ = 25 sec); medium: exp(μ=20sec); high: exp(μ = 15 sec) where ? is the mean of interarrival time
Weight of pf(T_i, t)	w_3 = 1
Simulation length	T_E = 8 hrs
Rework rate	5%
Warm-up time	T_0 = 10 min
Replication	rep = 10
Treatment	TAP1 = TRAP, SRAP, STOP1; TAP2 = TRAP, SRAP, STOP2; CP SRAP
Performance measures	TAR and WAR

Table 4: Java classes in TIE/TAP simulator

Sim	Supports the entire simulation, e.g., simulation initialization, termination, simulation clock, data collection
Model	Simulation modeling and execution, e.g., TestLAN model and parameters
Source	Random task generation
TA	Task agent which collects and announce tasks
RA	Resource agent which bids for and processes tasks
Task	Task definition and requirements
TRAP	Task Requirement Analysis Protocol
SRAP	Shared Resource Allocation Protocol
STOP	Synchronization & Time-Out Protocol

$$WAR = \frac{\sum_{i=1}^{|\tau|} \sum_{r=1}^{R} PR_i \cdot T_i^r}{\sum_{i=1}^{|\tau|} PR_i} \tag{14}$$

where $\overline{T} = |\tau| - \sum_{i=1}^{|\tau|} \sum_{r=1}^{R} |T_i^r|$;, r is the index of RA, $r = 1,\ldots,R$; T_i^r= 0 when T_i is neither assigned to RA_r nor completed within dd_i; 1 otherwise. While TAR does not consider the priority of tasks, WAR is used to measure the performance with consideration of task priority, i.e., assign more weights to relatively higher priority tasks when evaluating TAR.

The simulation is developed with a protocol evaluation tool, called TIE/TAP, which is implemented with Java, so as to compare the performance under three protocols: 1) TAP1, 2) TAP2, and 3) CP. The Java classes in TIE/TAP and their descriptions are listed in Table 4.

TAR results (Figure 2 and Table 5) are shown under low operational load (μ = 25; system capacity is sufficient to process all the tasks), medium (μ = 20; the number of tasks is close to the system capacity), and high (μ = 15; the number of tasks is beyond the system capacity). WAR results (Figure 3 and Table 5) are shown for the same load conditions. The observed results were tested for statistical difference by t-test as shown in Tables 6 and 7. In every case, it is found that TAP2 performs significantly better

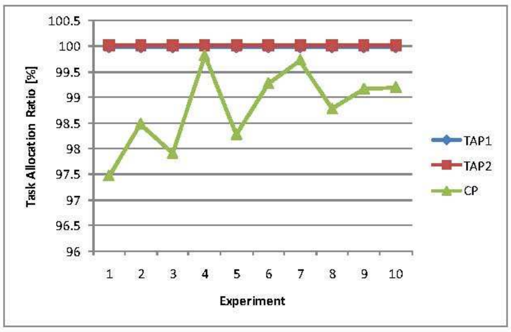
Table 5: Simulation results

Protocol	Task Allocation Ratio low/medium/high[%]	Weighted Task Allocation Ratio low/medium/high[%]
TAP1	100/95.7/72.8	100/90.5/52.1
TAP2	100/97.4/74.6	100/94.2/54.0
CP	98.8/92.8/73.0	98.1/83.6/49.9

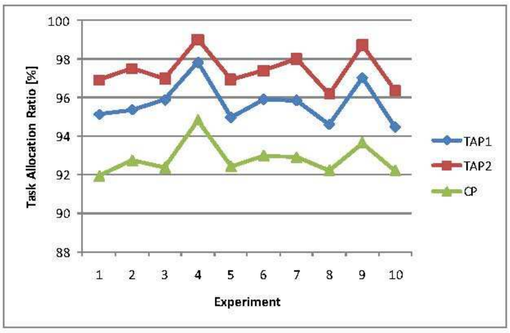
than TAP1 and than CP. Except for one case (TAR under high load condition), TAP1 is also significantly better than CP.

Under the low load condition, all three protocols studied perform similarly (TAR = 100% for TAP1 and TAP2, and 98.8% for CP). Under the medium load, the difference between the protocols increases. TAP2 yields the best TAR (97.4%), TAP1 – 95.7%, CP – 92.8%. Under the high load, TAP2 is also best (TAR = 74.6%), CP – 73.0%, TAP1 – 72.8%. Under every condition, TAP2 performs better than TAP1 and CP. TAP1 performs better than CP under the low and the medium load, but slightly worse under the high load. This situation results from the fact that TAP1 (and TAP2) considers priority of tasks by TRAP, tries to allocate the tasks with higher priority first, and even preempts the previously assigned tasks by STOP for the sake of the urgent tasks. Thus the TAR, which does not consider the priority of tasks, can be reduced under TAP1 (and TAP2) due to the failed tasks with low priority.

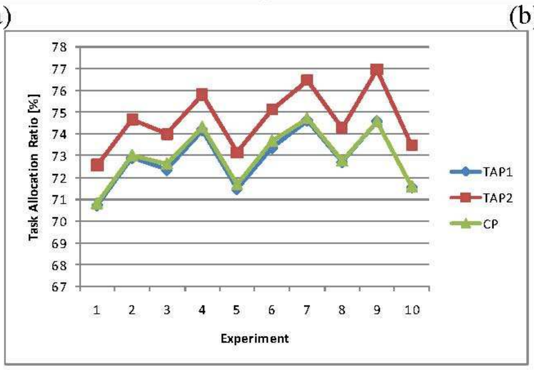
When WAR is considered (Table 5 and Figure 3), however, the performance of TAP1 is significantly better than CP under the high load. TAP2 is again the best (WAR = 54.0%), and TAP1 (WAR = 52.1%) also performs significantly better than CP (WAR = 49.9%). Under the medium load, WAR is 94.2%, 90.5%, and 83.6% for TAP2, TAP1, and CP, respectively. Therefore, when WAR is used to measure performance with consideration of task priority, the difference between TAPs and CP becomes larger under all conditions. This difference is expected from the fact that when tasks' priority changes dynamically, CP are not designed to track and respond to those changes.



(a)



(b)



(c)

Figure 2: TAR under (a) low (μ = 25), (b) medium (μ = 20), and (c) high (μ = 15) load condition

Table 6: t-test for *WAR* of TAP2 and CP

Treatment	Mean		Std. Dev.		t		$t_{1-\alpha,v}$
	$\mu = 20$	$\mu = 15$	$\mu = 20$	$\mu = 15$	$\mu = 20$	$\mu = 15$	$\alpha = 0.05, v = 18$
TAP2	0.942	0.540	0.0198	0.0127	11.444	7.3414	1.734
CP	0.836	0.499	0.0192	0.0112			

Table 7: t-test for WAR of TAP1 and TAP2

Treatment	Mean		Std. Dev.		t		$t_{1-\alpha,v}$
	$\mu = 20$	$\mu = 15$	$\mu = 20$	$\mu = 15$	$\mu = 20$	$\mu = 15$	$\alpha = 0.05, v = 18$
TAP1	0.905	0.521	0.0219	0.0101	3.774	3.641	1.734
TAP2	0.942	0.540	0.0198	0.0127			

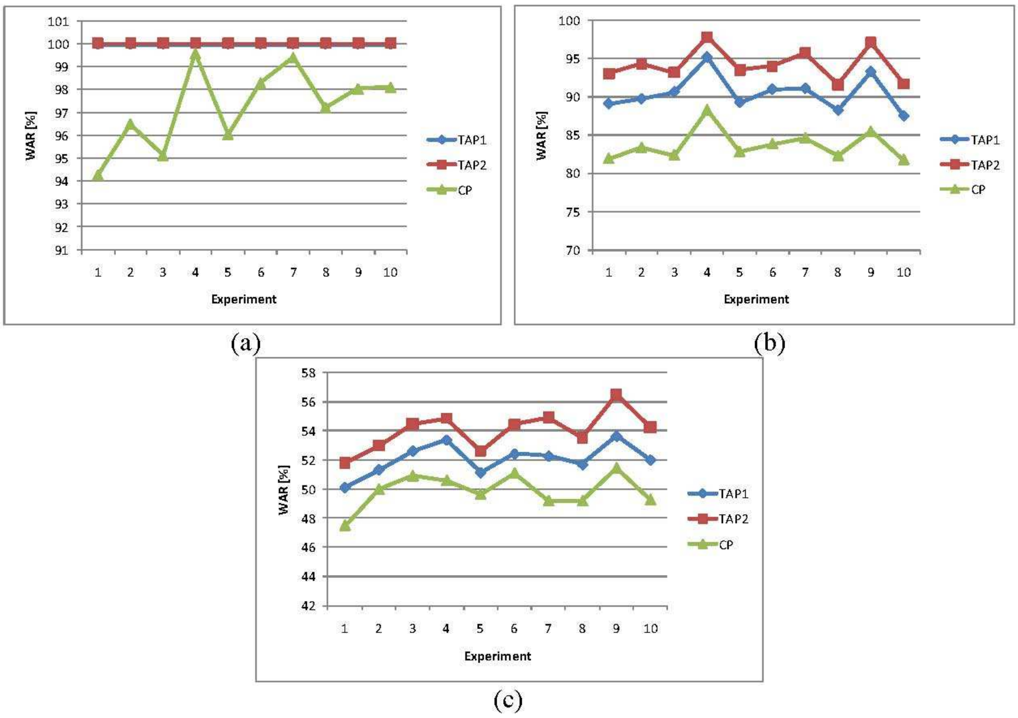


Figure 3: WAR under (a) low (μ = 25), (b) medium (μ = 20), and (c) high (μ = 15) load condition

In the studied case, the illustrated observations imply: (1) systems operating under TAPs perform better than under non-TAP CP, since TAPs can effectively handle the complicated situations that CP cannot (θ(CP) $<$ θ(TAP); see Table 6); and (2) even among TAPs, there can be better designed TAPs depending on the applications and logic applied (θ(TAP1) $<$ θ(TAP2); see Table 7).

5 Conclusions

The design of TAPs for task administration in a collaborative production system is investigated in this article. TAPs are designed as a control mechanism that can manage complicated situations in the collaborative task workflow environment. In order to overcome certain limitation of coordination protocols, TAPs are designed with three component-protocols, i.e., TRAP, SRAP, and STOP, each of which deals with inter-related aspects of task administration, including task, resource, and time. A case study of applying TAPs for TestLAN, as an example of collaborative production system, is developed to illustrate design of TAPs and show the advantage of TAPs over non-TAP CP. The simulation, implemented based on the TIE/TAP Java-based simulator, show that TAPs perform significantly better than other non-TAPs, in particular under medium or high load conditions (up to 10.6% in terms of WAR). This advantage results from the fact that the three protocol combinations in TAPs dynamically interact to consider the dynamic priority of tasks and the current situations and conditions in tasks and resources. Thus, TAPs can address a higher level of collaborative intelligence compared to non-TAP CP. Finally, the results imply that (1) TAPs are better than non-TAP CP under certain conditions; and 2) there can be better design of TAPs even among TAPs to increase system performance.

The logic in the protocols is designed to fit the given case study. Although the general structure of protocols can be followed in order to handle the complicated situations in collaborative tasks/resources networks, for better effectiveness the protocol logic needs to be context-specific. Different applications require different decision policies, heuristics and logic, and they should be reflected in the protocol logic, as illustrated in the case study. In order to obtain better performance, the logic and the parameters in the protocol should be carefully selected and modified. For example, the system performance in the case study can be significantly affected by parameters such as time-out threshold. This recommendation is one of the challenging issues in protocol design, and some research has tried to address this issue by service-oriented protocol adaptation [21]. Further research is currently ongoing to develop and improve TAP logic with protocol adaptation for other applications, e.g., TAPs for collaborative intelligence in research activities [19], in which TAPs should be developed to manage collaborative workflow between distributed research groups and enhance their collaboration. Other similar, potential areas of TAPs application include collaborative intelligence in e-Learning environment [24], knowledge management in enterprise portal [25], and e-Service collaboration protocols [26].

Bibliography

[1] T.W. Malone, K. Crowston, The Interdisciplinary Study of Coordination, ACM Computing Surveys, 26(1):87-119, 1994.

[2] K. Crowston, J. Rubleske, J. Howison, Coordination Theory: A Ten-Year-Retrospective, Human-Computer Interaction and Management Information Systems: Foundations, P. Zhang, D. Galletta (eds.), M. E. Sharpe, Inc., 2006.

[3] B. Gerkey, M.J. Mataric, Sold!: Auction Methods for Multi-robot Control, IEEE Transactions on Robotics and Automation, 18(5):758-768, 2002.

[4] R.Y.K. Fung, T. Chen, A Multiagent Supply Chain Planning and Coordination Architecture, International Journal of Advanced Manufacturing Technology, 25(7):811-819, 2005.

[5] W. Shen, D.H. Norrie, Dynamic manufacturing scheduling using both functional and resource related agents, Integrated Computer-Aided Engineering, 8(1):17-30, 2001.

[6] M. Wooldridge, N.R. Jennings, Intelligent Agents: Theory and Practice, The Knowledge Engineering Review, 10(2):115-152, 1995.

[7] R.G. Smith, The Contract Net Protocol: High-level Communication and Control in A Distributed Problem Solver, IEEE Transactions on Computers, 29(12):1104-1113, 1980.

[8] S.S. Fatima, M. Wooldridge, Adaptive Task and Resource Allocation in Multi-agent Systems, Proceedings of the Fifth International Conference on Autonomous Agents, Montreal, Canada, 537-544, 2001.

[9] M.P. Wellman, E. Walsh, P.R Wurman, J.K. MacKie-Mason, Auction Protocols for Decentralized Scheduling, Games and Economic Behavior, 35(1-2):271-303, 2001.

[10] K. Ertogral and S. D. Wu, Auction-theoretic coordination of production planning in the supply chain, IIE Transactions, 32(10):931-940, 2000.

[11] C.Y. Huang, S.Y. Nof, Evaluation of Agent-based Manufacturing Systems Based on A Parallel Simulator, Computers and Industrial Engineering, 43(3):529-552, 2002.

[12] K. Esfarjani, S.Y. Nof, Client-server Model of Integrated Production Facilities, International Journal of Production Research, 36(12):3295-3321, 1998.

[13] N.P. Williams, Y. Liu, S.Y. Nof, TestLAN Approach and Protocols for The Integration of Distributed Assembly and Test Networks, International Journal of Production Research, 40(17):4505-4522, 2002.

[14] J. Peralta, J., P. Anussornnitisarn, S.Y. Nof, Analysis of A Time-out Protocol and Its Applications in A Single Server Environment, International Journal of Computer Integrated Manufacturing, 16(1):1-13, 2003.

[15] P. Anussornnitisarn, Design of Active middleware protocols for coordination of distributed resources, PhD dissertation, Purdue University, 2003.

[16] P. Anussornnitisarn, S.Y. Nof, O. Etzion, Decentralized Control of Cooperative and Autonomous Agents for Solving The Distributed Resource Allocation Problem, International Journal of Production Economics, 98(2):114-128, 2005.

[17] A.J. Arauzo, J.M. Galan, P. Javier, A. Lopez-Paredes, Multi-agent Technology for Scheduling and Control Projects in Multi-project Environments. An Auction Based Approach, Inteligencia Artificial, 42:12-20, 2009.

[18] Y.H. Lee, S.R.T. Kumara, K. Chatterjee, Multiagent Based Dynamic Resource Scheduling for Distributed Multiple Projects Using A Market Mechanism, Journal of Intelligent Manufacturing, 14:471-484, 2003.

[19] H.S. Ko, S.Y. Nof, Design of Collaborative e-Service Systems, Introduction to Service Engineering, G. Salvendy, W. Karwowski (eds.), John Wiley & Sons, Inc., 2009.

[20] J.D. Velasquez, S.Y. Nof, Collaborative e-Work, e-Business and e-Service, Springer Handbook of Automation, S.Y. Nof (ed.), Springer, 2009.

[21] N.P. Williams, Y. Liu, S.Y. Nof, Analysis of Workflow Protocol Adaptability in TestLAN Production Systems, IIE Transactions, 35(10):965-972, 2003.

[22] R.I. Lung, D. Dumitrescu, Collaborative Optimization in Dynamic Environments, International Journal of Computers, Communications, and Control, 1(Suppl), 2006.

[23] C. Kolski, P. Forbrig, B. David, P. Girard, C.D. Tran, H. Ezzedine, Agent-Based Architecture for Interactive System Design: Current Approaches, Perspectives and Evaluation, Lecture Notes in Computer Science, 5610:624-633, 2009.

[24] I. Moisil, A Model of the Student Behaviour in a Virtual Educational Environment, International Journal of Computers, Communications and Control, 3(Suppl):108-115, 2008.

[25] M. Guran, Knowledge Management using Intranets and Enterprise Portals, International Journal of Computers, Communications and Control, 3(Suppl):75-81, 2008.

[26] G. Kramler, E. Kapsammer, W. Retschitzegger, G. Kappel, Towards Using UML 2 for Modelling Web Service Collaboration Protocols, Interoperability of Enterprise Software and Applications, D. Konstantas et al (eds.), Springer, 2006.

[27] F.G. Filip, G. Neagu, D.A. Donciulescu, Job Shop Scheduling Optimization in Real-time Production Control, Computers in Industry, 4(4):395-403, 1983.

[28] J.M. Frayret, S. D'Amours, B. Montreuil, Coordination and Control in Distributed and Agent-based Manufacturing Systems, Production Planning and Control, 15(1)42-54, 2004.

[29] J. Jiao, X. You, A. Kumar, An Agent-based Framework for Collaborative Negotiation in The Global Manufacturing Supply Chain Network, Robotics and Computer-Integrated Manufacturing, 22(3):239-55, 2006.

[30] L. Iocchi, D. Nardi, M. Piaggio, A. Sgorbissa, Distributed Coordination in Heterogeneous Multi-robot Systems, Autonomous Robots, 15(2):155-168, 2003.

Hoo Sang Ko is a Senior Researcher of PRISM Center and a doctoral candidate in the School of Industrial Engineering at Purdue University. He received his B.S. in 1999 and M.S. in 2003 in Mechanical Engineering at Seoul National University in South Korea. After graduation, he worked as an R&D engineer at Digital Appliance Business of Samsung Electronics. His research interests include design of collaboration support systems, task administration protocols for control of collaborative e-Work, and facility sensor networks.

Shimon Y. Nof is Professor of Industrial Engineering and director of the NSF-industry supported PRISM Center (Production, Robotics and Integration Software for Manufacturing & Management) at Purdue University. The PRISM Center was established in 1991, and its motto is "Knowledge through information, wisdom through collaboration." In 2001, PGRN, PRISM Global Research Network was established with affiliate labs and centers throughout the world. Professor Nof was awarded an Honorary Doctorate in Engineering from the University of Sibiu, Romania in 2007, when Sibiu served as the European Capital of Culture.

Int. J. of Computers, Communications & Control, ISSN 1841-9836, E-ISSN 1841-9844
Vol. V (2010), No. 1, pp. 106-124

Gaze, Posture and Gesture Recognition to Minimize Focus Shifts for Intelligent Operating Rooms in a Collaborative Support System

J. P. Wachs

Juan P. Wachs
Purdue University
Department: School of Industrial Engineering
Address: 315 N. Grant St., W. Lafayette, IN 47907-2023 USA
E-mail: jpwachs@purdue.edu

Abstract: This paper describes the design of intelligent, collaborative operating rooms based on highly intuitive, natural and multimodal interaction. Intelligent operating rooms minimize surgeon's focus shifts by minimizing both the focus spatial offset (distance moved by surgeon's head or gaze to the new target) and the movement spatial offset (distance surgeon covers physically). These spatio-temporal measures have an impact on the surgeon's performance in the operating room. I describe how machine vision techniques are used to extract spatio-temporal measures and to interact with the system, and how computer graphics techniques can be used to display visual medical information effectively and rapidly. Design considerations are discussed and examples showing the feasibility of the different approaches are presented.
Keywords: posture recognition, behavior analysis, intelligent rooms.

1 Introduction

Intelligent systems can assist in improving safety and performance during surgery in many ways. Intelligent operating rooms assist the surgeon in time- and safety-critical situations. Multimodal cues from humans and the environment can be used to extract behavior, which in turn can be used to derive contextual and event information. Some of these behaviors include, for example, whether there are distractions, whether the subject is busy with certain tasks or how frequently the doctor switches his focus of attention.

1.1 Previous research

Hansen [1] studied focus shift in the operation room and he used the terms focus spatial offset and movement spatial offset. Focus spatial offset is the change in the doctor's gaze as a result of focusing in a new spot. Conversely, movement spatial offset results from the doctor's change in position. It is possible to perform basic behavior analysis of body language in order to determine which type of offset of attention occurs in any point in time.

1.2 Hypothesis

With behavioral information the following questions could be answered:

1. How can surgeon's behavioral cues be leveraged to improve the Operating room (OR) layout dynamically?

2. How can innovative channels of interactions in this enhanced layout minimize unnecessary focus shifts?

3. How can the doctor and the operating room collaborate to display time sensitive visual information effectively?

Our hypothesis is that a reconfigurable smart operating room can organize the displayed information in such a way that unnecessary focus shifts are minimized. This has an impact on the surgery performance time. Moreover, with the adoption of natural modalities of interaction, unnecessary instructions to the nurses will be eliminated. This also has an impact on the task completion time.

1.3 Previous works

Previous research has demonstrated the negative effects of attention shift on the performance of cognitive and motor tasks [2], [3]. A study conducted by Godell et al., [4] looked at virtual reality laparoscopic surgical simulation tasks designed to replicate the levels of cognitive and motor demands in surgical procedures, and found that there was a 30-40% increase in task completion time in the distracted vs. undistracted condition.

Recent advances have been proposed to counteract unnecessary distracting behavior through the integration of doctor behavior analysis and context awareness into the operating room [5], [6].

The analysis of body language is critical in determining when the surgeon is operating or analyzing medical imagery or just chatting with the staff. Providing the intelligent operating room with the ability to understand the doctor's behavior and the context of the situation allows the projection of patient imagery in the area that allows the least shift of focus and the most optimal interaction space. Thus, focus shift is reduced and task performance is improved.

For example, when a surgeon interacts with a particular display to obtain information which must be cross checked with magnetic resonance imaging (MRI) images obtained previously, in the current operating room's layout, she will need to move to the main control (see Fig 1). This causes focus and spatial shifting and hence distraction and unintended delay. An intelligent operating room can use the surgeon's body language to determine that she is currently interacting with a control and can then determine the best orientation and position to project the patient MRI imagery.

A doctor's assistance system mitigates shift of focus and distractions because it senses the center of the surgeon's attention through context and uses this knowledge to display relevant patient imagery at this attention point. The system also provides a sterile interface for interaction with the visual information in the spot where the surgeon is already standing, thereby avoiding spatial shift. In this paper, torso and head orientation can be used to find the focus of attention and the context of the event, and how gestures and voice can be combined to interact with medical imagery is discussed. This concept is depicted in Figure 1. Also, by extracting the dynamics of head pose (eye gaze) and hand gestures, it may be possible to identify those attention shifts which are associated with premeditated or task-oriented attention shifts. For example, a hand gesture oriented to the display combined with gaze directed towards it may indicate that the attention shift was goal-oriented. If the surgeon is bored or distracted he would likely not point to the display. Intelligent environments could use this semantic information to assist doctors and improve the safety and comfort of their interactions and surroundings.

Examples from two different real-time working systems to support collaborative work: real-time pose, posture and head orientation recognition, and a sterile hand gesture recognition system for displaying medical imagery.

The remainder of the paper is organized as follows. Section two covers previous work, and a description of the methodology proposed for displaying imagery and interacting with the intelligent room are presented in section three. A concluding discussion appears in section four.

2 Related Work

A large amount of research has recently been conducted on intelligent rooms based on the detection of human behaviors and supporting collaborative work with the room. Many of these examples use machine vision techniques to extract important clues, such as head-gaze orientation and posture patterns. They also use computer graphics techniques to display the visual information with high detail and accuracy using special displays or projectors. To interact with the room, voice, body and hand gestures are the prominent channels of natural communication.Below, an overview of related research pertaining to the health-care scenario is presented.

2.1 Intelligent Rooms

Traditional intelligent rooms are multi-modal environments where audible and visible clues are captured and analyzed exclusively from people's actions. These systems typically run in real-time, and utilize embedded sensors small enough as to be unnoticeable by people. Intelligent rooms in collaborative environments, however, allow human-machine interaction.

Recent research has supported the incorporation of sensors that look inside a living or working space (the room) to observe inhabitant behavior and potentially infer his intent. One of the pioneering works is the DigitalDesk [7], [8] where a computer display is projected onto the desk, and video cameras pointed down at the desk feed an image-processing system that can sense what the user is doing. This application allows users to construct a mixed paper and electronic drawing device. Not surprisingly, some of the ideas first presented in the DigitalDesk system can now be found in more sophisticated applications. One example is the multi-touch screen interaction on the current Microsoft Surface [9] device. Current approaches used in intelligent rooms combine robotics and vision technology with speech understanding algorithms and distributed agent-based architectures to provide information services in real-time for users engaged in everyday activities [10].

Real-time three-dimensional people-tracking by a network of video cameras using a rectilinear video array tracker and an omnidirectional video array tracker (O-VAT) in order to develop a real-time system for tracking the head and face of a person in an intelligent room is described in [11]. In the GlobeAll system, [12], a modular four-component prototype for a vision-based intelligent room is developed. The prototype consists of a video input from an electronic pan-tilt-zoom camera array, which performs tracking and interpretation. An intelligent room for welfare applications called the "percept-room" was developed in [13]. The system is capable of detecting human faces and classifying hand postures in Japanese Sign Language in color images. In the Human Interaction Loop (CHIL) smart room project [14] perceptual and situation-modeling components are used to provide context-aware services to the users of the smart room.

Anthropocentric Interfaces based on intuitive and natural aspects is undergoing which is expected to improve the usability of current systems based on multimodal interaction [15].

The intelligent room reverses the traditional model of teaching the user to interact with computers. Instead the intelligent room supports natural interaction, enabling the user to feel as if the computer weren't there. The smart room research is particularly encouraging: it enabled people to use home appliances and perform every-day tasks that would be more awkward to do in other ways. Our goal is to bring these achievements into the operating room in the context of a collaborative support system. The steps involved in this transition are described in the next sections.

2.2 Smart Displays and Projectors

Smart projectors allow such applications as augmented reality and immersive displays for three dimensional video-conferencing, real-time annotation and simulation and complex and high volume imagery display. Increasing computing power, smaller projectors and fast graphic cards make projector-based interaction an attractive feature for intelligent rooms. For example, a patient's basic information and condition could be displayed in his field of view, or an MRI image projected over a patient's head could help a surgeon determine more accurately a brain tumor's location. Projection in side walls is the method adopted in this work.

Wide screen projection using multiple projectors is a challenging problem since it requires undistorting the image, projector calibration and a setting a position and orientation for the projectors, shadow elimination, and because the image surface is of unknown geometry. If the surface to be projected over is uneven the problem becomes even more complex. [16] presents an image projection method that allows arbitrary observer-projector-screen geometries, relying on a robust structured light approach which can be easily adapted to multi-projector configurations.

An interactive projector automatic calibration process for multi-projector-camera environments is demonstrated in [17]. This method does not require calibration rigs nor does it assume fixed display positions; however it requires the cameras to be to be pre-calibrated. This is done through detecting self-identifying tags projected in freely moving targets. Overlapping tiled projected images are obtained from multiple steerable projectors in [18]. The projectors operate simultaneously and misalignments on the projected image are corrected through a preliminary process. In addition the authors present a virtual synchronization method based on the dynamic memory architecture for the projectors. A high resolution video display system using multiple projectors to build a surround video display system is presented in [19]. Bhasker et al. [20] suggested a registration technique allowing severely distorted projectors to be used which opens up the possibility of mounting inexpensive lenses on projectors. This technique can handle imperfect and uncalibrated devices in planar multi-projector displays. More specifically, it can correct for geometrically imperfect projectors and photometrically uncalibrated cameras. Approaches to solve occlusion and the resulting blinding light are reported in [21] related to front projection. An algorithm based on a distributed calibration framework for multi-projector displays where the projectors cooperatively re-estimate the poses of all projectors during actual display use is discussed in [22].

2.3 Hand Gesture Recognition in Healthcare

Natural hand poses and gestures are used to control, teach, treat and manipulate systems in diverse areas of the healthcare environment. Gestures can be used to control the distribution of resources in a hospital, to interact with medical instrumentation, visualization displays, to help handicapped users as an alternative interface to computers and as part of rehabilitation therapy. When the hands are attached to sensors to provide haptic (tactile and force) feedback, a surgeon's gestures can be used to perform long distance surgeries with the help of telerobots. Additional systems use voice, gaze and gestures together, profiting from the combined advantages of these modalities to convey richer and redundant information.

Some gesture concepts have been exploited for improving medical procedures and systems. The "come as you are" requirement is addressed in FAce MOUSe [23], where a surgeon can control the motion of the laparoscope by simply making the appropriate face gesture, without hand or foot switches or voice input. Current research to incorporate hand gestures into doctor-computer interfaces has appeared in Graetzel et al.[24]. They developed a computer vision system that enables surgeons to perform standard mouse functions (pointer movement and button presses) with hand gestures while addressing the "intuitiveness" requirement. A hand gesture tracking device for browsing MRI images in the operating room (OR), called "Gestix" was developed in [25] and it was validated in a real brain biopsy (see Figure 1). "Gestix" addressed both the "come as you are" and "intuitiveness" requirements by providing a natural effective interface. The "comfort" requirement is fulfilled in "WearIT@work" [26], a RFID reader is used to identify the patient and to interact with the hospital information system (HIS) using gestures to fill out exams and prescriptions. This project ensures sterility. However, since this is an encumbered interface, the "come as you are" requirement is violated.

From the patient side, the most prominent requirements in a hand gesture interface system are "User

Figure 1: A surgeon using Gestix to browse medical images

adaptability and feedback" and "come as you are" because impaired users may be limited in the classes of gestures that they can learn and the devices that they can wear. In this context, wheelchairs as mobility aids have been enhanced with robotic/intelligent vehicles able to recognize the user's commands indicated by hand gestures [27]. The Gesture Pendant [28] is a wearable gesture recognition system that can be used to control home devices and provides additional functionality as a medical diagnostic tool. The "user adaptability and feedback" requirement is addressed in Staying Alive[29], which is a virtual reality imagery and relaxation tool which allows cancer patients to navigate through a virtual scene. A haptic glove attached to the hand was used to rehabilitate post-stroke patients in the chronic phase in [30].

These reviewed systems indicate that hand gesture interfaces in medical domains still represent a novel area of research and that few systems are currently in use in healthcare environments. Nevertheless, current works highlight the potential of gestures as a natural modality for assisting in the advancement of medical research and surgery, and indicate the need for additional research and evaluation procedures so that such systems can be widely adopted.

3 Methods

In this section, the philosophy behind the design of our system is presented, and how to develop an "intelligent" operating room based on off-the shelf hardware (a servo controlled projector, four pan-tilt-zoom cameras, four microphones, and a controlled connected to a dedicated computer) is described. The system consists of four sub-systems: (a) steerable projectors, (b) focus of attention determination, (c) hand gesture interaction and (d) speech interaction. First the surgeon's posture, pose and orientation are tracked and detected. This information is sent to the steerable projector, which controls a servo-mirror where the projector ray is deflected. The head orientation is used to determine the closest wall in front of the surgeon where the medical imaging can be reflected. The projection system is activated only when the surgeon is standing straight and staring at a wall, and also evokes a command by saying "computer-project-now". Once the image is projected, the surgeon can interact with the images using hand gestures and voice commands. The system is turned off when the surgeon performs a specific gesture command. These functionalities are described in the schema presented in Figure 2.

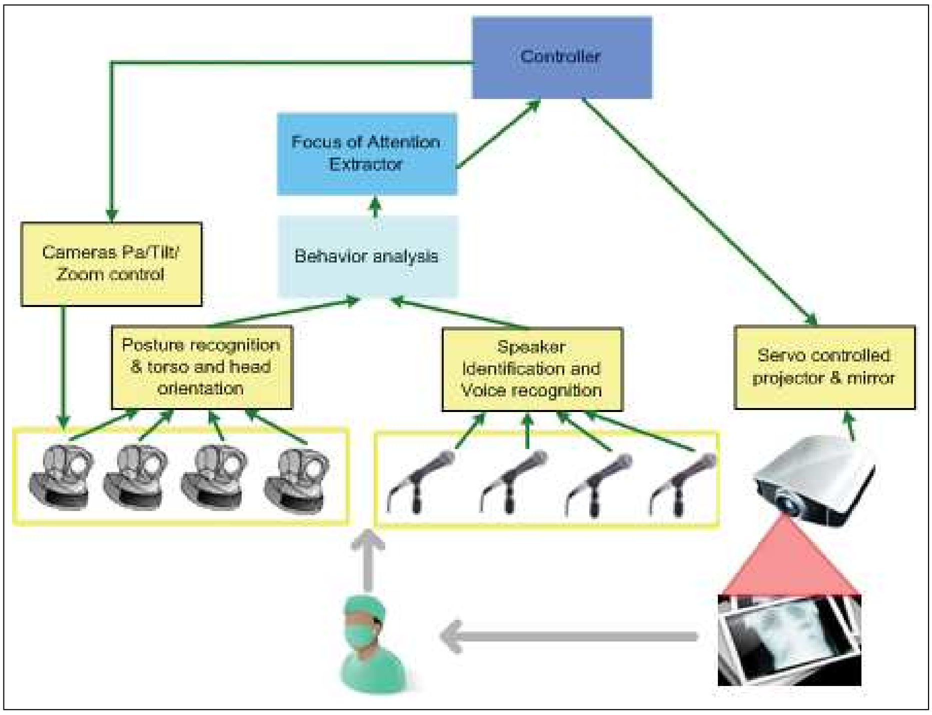


Figure 2: Architecture for the Intelligent Operating Room

In the following sub-sections the subsystems are described in greater detail. Most emphasis is given to focus of attention and hand gesture interaction because those are the most difficult challenges of this environment as showed in previous research. Steerable projectors and speech recognition are the focus of future work.

3.1 Steerable Projectors for non-uniform projection surfaces

A steerable projector allows us to move the projected image in real time such that the projected image is in front of the surgeon, perpendicular to his torso. The implementation of the steerable projector proposed here is similar to [31] where a standard digital projector is combined with a mirror attached to a pan-tilt unit. This approach is more compact and much cheaper than standard steerable projectors. This model includes the internal parameters of the projector, the projector pose, the pose of the pan-tilt mirror system, and the internal details of the mirror system. By changing the pan and tilt angles programmatically, the mirror rotates about the pan and tilt axes respectively and the projected image changes in turn.

The transformation between the projector pose and the mirror pose can be calculated through a calibration process. Since the pan-tilt angles will have an impact on the registration process between the projected image and the camera model of the 3D view, the use of a fixed set of projector poses is suggested, one for each wall, to reflect the surgeon's possible orientations: upper abdominal procedures are usually performed with the surgeon standing in the midline; central and pelvic operations are generally performed with the surgeon on the lateral side of the patient [32].

Therefore, a set of four reflected projector poses is calculated, one for each mirror pose. Each pair of θ and φ values results in a different mirror pose and hence a different set of projected corner points on the surface. The rigid transform from the reflected projector position for each of those mirror poses is

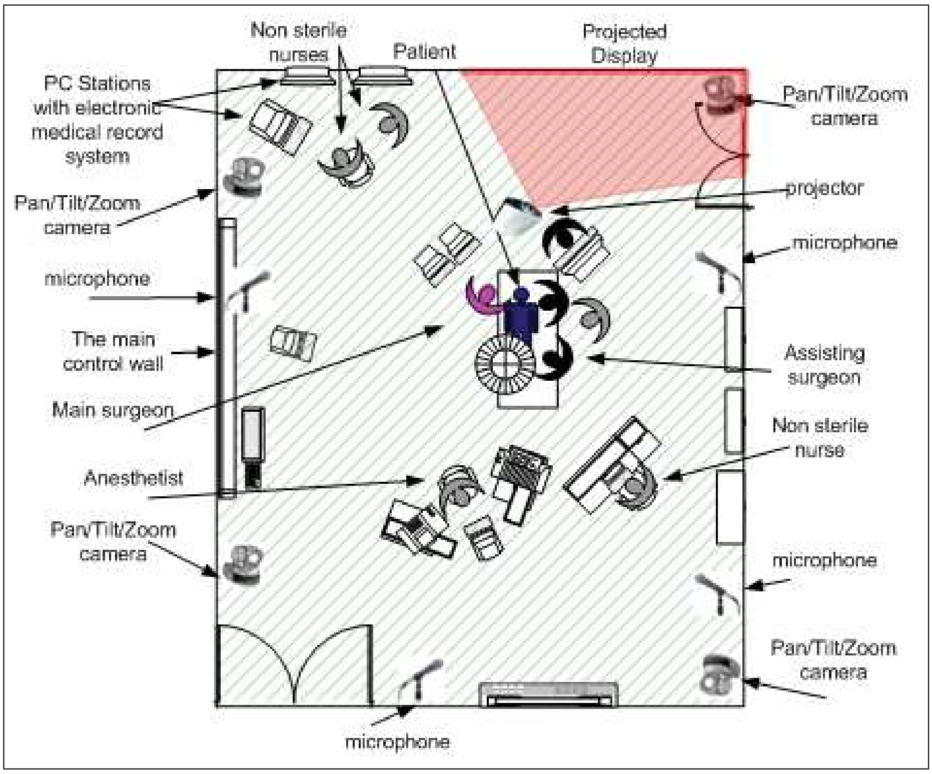


Figure 3: Concept image from the intelligent operating room

calculated using the procedure detailed in [33]. This requires the use of a camera to capture the projected image on the display surface, and assumes that the display surface is uneven. Since in our case the display surface could potentially be any wall around the surgeon, a pan/tilt/zoom camera dedicated to the projector is allocated, in addition to those used to capture the surgeon postures. Let the display surface be represented in a 3D mesh. I try to find the transform that relates a given 2D pixel in the projector image plane (Z), to a 3D pixel in the display surface (K), given that the place of the camera is so that the same point (V) appears in the camera image as Z'. See Figure 4.

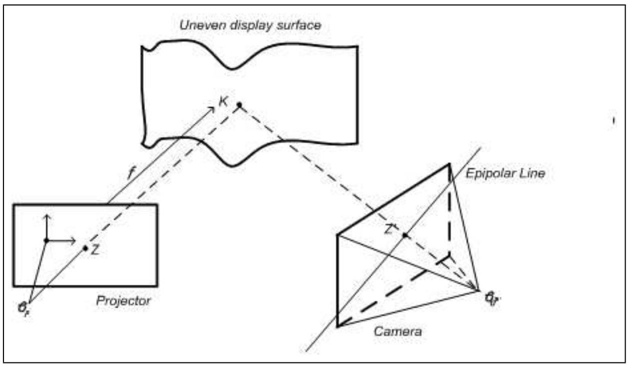


Figure 4: . Camera and Projector 3D view geometry representation

Each point $x = (x, y, w)^T$ in the uneven display surface is a result of a ray originated in the center of the projector O_p traversing the projector plane in point $Z = (\hat{u}_p, \hat{v}_p)$, which in turn appears on the image captured by the pan/tilt camera as point $Z' = ({}_c, \hat{v}_c)$. The goal is to find the static parameter f for every point relating the projector plane and the display surface. Knowing the internal parameters of the projector, and the internal parameters of the camera and the homogeneity matrix, [33] show that each sample point K in the 3D mesh follows can be found using (1):

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = p^{-1} \left(-\hat{p} + f \begin{bmatrix} \hat{u}_p \\ \hat{p}_p \\ 1 \end{bmatrix} \right) \tag{1}$$

where p is the rotation 3x3 matrix, $\hat{p}$ is the translation 3x1 vector, and f is the parametric scalar value. It is possible to show that the parameter f can be estimated using traditional correspondence approaches, or using predictive methods such as particle filtering or Kalman filters. In [33] a bank of Lalman-filters were used to find the correspondences between the projector pixel Z and the projected point K, one filter for each point Z. To reduce the complexity of the process, in every frame only one point Z is selected for matching. Once a point is selected using a certain method (pseudo-randomly, distance based, feature detector), a patch around the neighboring pixels is selected. This patch is cross-correlated with the predicted location on the camera's image, and close to the epipolar line. Once the new coordinate is found on the camera's image, the parameters for the 3D mesh are updated. This process corrects the projected image over the uneven surface so that curves and discontinuities are not perceived by the viewer, however errors (maximum 6.78mm and mean 2.41mm) similar to [33] are expected. Sensitivity studies based on these errors are the focus of future work.

3.2 Focus of attention determination

In this section, it is shown that by extracting the surgeon's head orientation (which will be used as a proxy for eye gaze) and torso posture and orientation (which are indicators of the action being performed), it may be possible to identify focus spatial offset and movement spatial offset which are associated with attention shifts. In the first case, gaze orientation helps us to know where to project the medical imagery (which wall to use as the display surface). In the second case, torso position (standing straight or bent) indicates the surgeon's possible intention: interaction with the patient, interaction with the medical imaging display or other activity.

Our method to extract head and torso position, posture and orientation is to look at each body configuration as a different class: for example, the following notation describes the current state of a surgeon: s1 = standing, torso 90 degrees, facing 0 degrees. Thus, in this section, a multi-class classifier based on parts-based models is described to find each of these configurations. The approach is briefly described for object detection using single and multi-class detectors, in the context of US marine detection, as a case study only, but it is clear that the same approach can be applied to any person. First, it is described the feature extraction process from patches (or parts) and then the basic and shared classifiers are discussed.

3.2.1 Dictionary Creation

Initially, a dictionary is created from square sub-region patches extracted from a set of images per class, similar to [34]-these are also called "features". Each image is convolved with a filter from a bank of filters, grayscale normalized and re-scaled to a standard scale of 128x48 for the standing and 64x48 for marines kneeling. Then patches are selected in *x, y* locations found using an interest point detector. In those locations patches are extracted from all the filtered images. Each patch is associated with the place from where it was extracted, relative to the center of the object. This location information is stored in two vectors containing the *x, y* offset distances respectively l_x, l_y, after applying a blurred delta function to them. Hence, each entry i in the dictionary has the form v_i={filter, patch, l_x, l_y, image no.}. If 8 images per class are used to create the dictionary, 20 patches are extracted per image, a bank of 4 filters is used, and by classifying into 8 classes, a total of 640 entries is obtained. The procedure is shown in Figure 5.

3.2.2 The feature vectors

The training set is created from a sample set of images excluding those used for the dictionary creation. For each of the eight (classes) objects I found all the images that include that type of object. In each image, feature vectors are obtained using the following method:

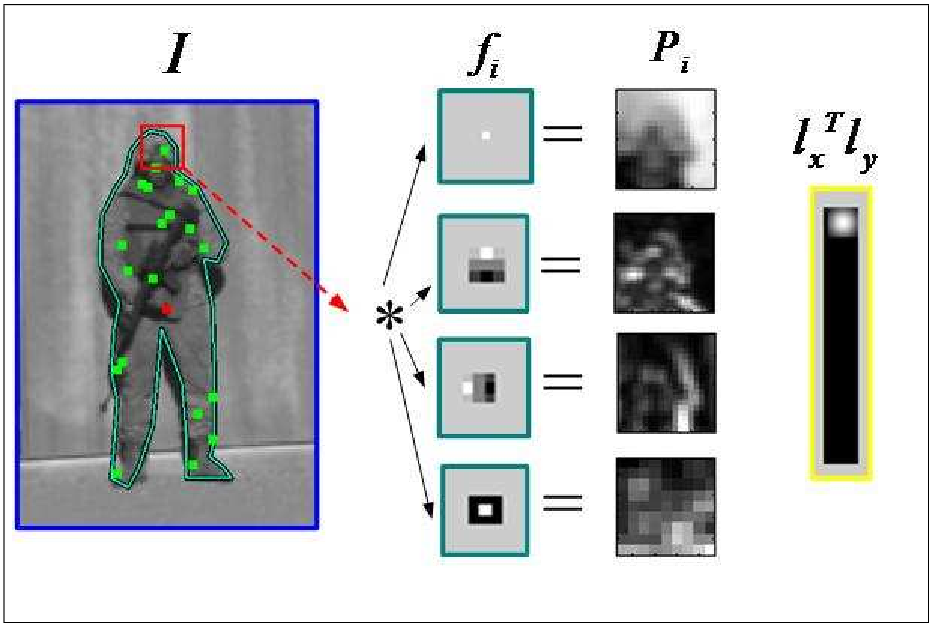


Figure 5: . . Dictionary entries: patches selected randomly (on the left image) are convolved with a bank of filters. The position of the patches is represented by the location matrix (right). Since the red patch is at almost the same horizontal position and at the top, relative to the center, the position matrix has a bright spot

1. Scale all the images in the training set so the object of interest is bounded by a rectangle of size 128x48 and 64x48 (region of interest, ROI) for standing and kneeling respectively, and the images are not larger than 200x200.

2. For each image j normalized in scale, each entry i of the dictionary is applied to it: this means that this image is convolved with the filter in entry i, and convolved again with a Gaussian to smooth the response. Next, it is cross-correlated with the patch in entry i, yielding a strong response where this patch appears in the filtered image. Finally, the 1D filters l_x and l_y are applied to the cross-correlated image, effectively "voting" for the object center. This is summarized in (2):

$$v_i(x,y) = [(I^* f_i) \otimes P_i]^* l_x^T l_y \tag{2}$$

Where * is the convolution operator, $\otimes$ is the normalized cross correlation operator, $vi(x,y)$ is the feature vector entry i, f is a filter, P is a patch, and l_x and l_y are the x,y location vectors with respect to the center of the image respectively.

Each training feature vector is coupled with a class label (1 to 8) and -1 for negative samples. For a sample set of 25 images per class, 4000 negative and 200 positive samples are obtained, with 640 features, see Figure 6.

3.2.3 Sharing the features effectively

In this section I briefly describe the joint boosting algorithm used for multi-class multi-view object detection. For a more detailed discussion, refer to [35]. A boosting algorithm is an additive model where weak learners are sequentially added to form a strong classifier. For the multiclass case, the strong learner is defined as:

$$H(v,c) = \sum_{m=1}^{M} h_m(v,c) \tag{3}$$

Where v is the input feature vector, M is the number of boosting iterations, c is a specific class and

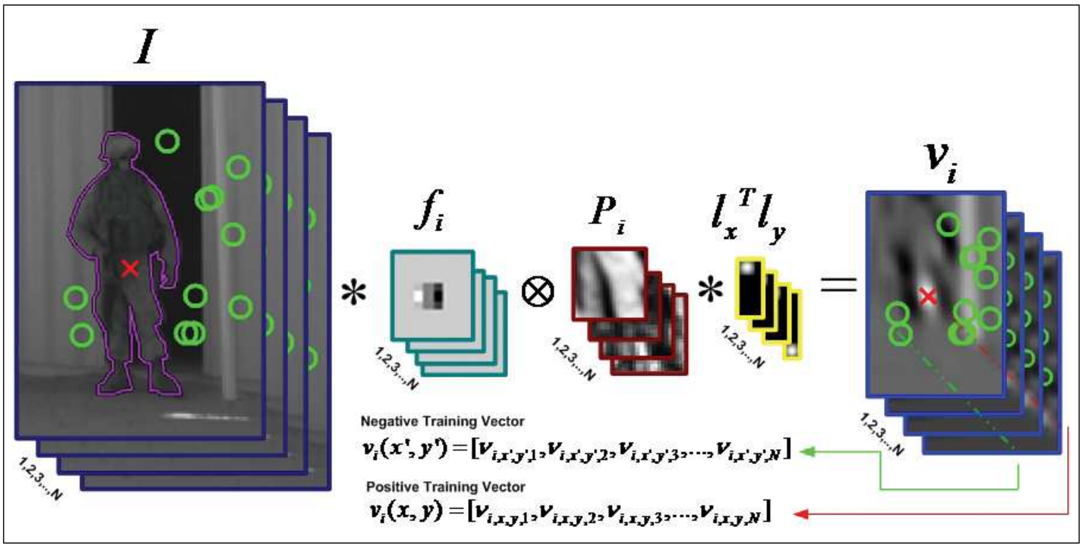


Figure 6: . Positive and negative vector set creation using the dictionary entries and sampling the center out of the silhouette points. Each sampled point, is a vector, where an entry j in the vector represents the number of votes assigned by patch P_i

H(v,c)=log P(zc=1|v)/P(zc = -1|v) is the logistic function where z is the membership label (ą1). When the expectation is replaced by an average over the training data, the cost function can be written as:

$$J_{wse} = \sum_{c=1}^{C} \sum_{i=1}^{N} w_i^c (z_i^c - h_m(v_i, c))^2 \tag{4}$$

Where N is the number of training vectors,w_i^C are the weights for sample i and for class c,z_i^C is the membership label for sample i for class c (± 1). The weak shared learner, also called, regression "stump" is defined for the multiclass in (5):

$$h_m(v,c) = \begin{cases} a_S & \text{if} \quad v_i^f > \theta andc \in S(n) \\ b_S & \text{if} \quad v_i^f \leq \theta andc \in S(n) \\ k_S^c & \text{if} \quad c \notin S(n) \end{cases} \tag{5}$$

where v_f is the component f^{th} from the vector v, θ is a threshold, δ is the indicator function, a_S and b_S are regression parameters. S(n) is a subset of the classes labels. Each round of boosting consists of selecting the shared "stump" and the shared feature f that minimizes (3), from the subset of classes S(n), in the following stated procedure: Pick a subset of classes S(n). Search all the components f of the feature vector v, for each component, search over all the discrete values of θ and for each couple {f, θ}, find the optimal regression parameters a_S and b_S using (6-8). Finally, select {f,θ, a_S, b_S} that minimizes (4).

$$a_s(f,\theta) = \frac{\sum_{c \in S(n)} \sum_i w_i^c z_i^c \delta(v_i^f > \theta)}{\sum_{c \in S(n)} \sum_i w_i^c \delta(v_i^f > \theta)} \tag{6}$$

$$b_s(f,\theta) = \frac{\sum_{c \in S(n)} \sum_i w_i^c z_i^c \delta(v_i^f \leq \theta)}{\sum_{c \in S(n)} \sum_i w_i^c \delta(v_i^f \leq \theta)} \tag{7}$$

$$k^c = \frac{\sum_i w_i^c z_i^c}{\sum_i w_i^c} \tag{8}$$

Therefore a shared weak learner is associated with a set of 6 parameters {f, θ, a_S,b_S, k_c, S_n} of the subset of classes selected. It is more efficient to keep a pointer to the entry in the dictionary from which f was obtained rather than keeping the whole feature vector (Figure 7 displays all the entries in the dictionary). This will also provides us with the patch, filter and location vectors entries in the dictionary which will be used for the detection stage. This new weak learner is added to the previous accumulated learner, for each training example: H(v_i, c)= H(v_i, c)+h_m(v_i, c) where h_m is computed for the optimal subset of classes. The optimal subset of classes is the one that minimize the misclassification error by selecting a feature shared by those classes. Finally, the chain of weak learners is stored in the accumulated learner.

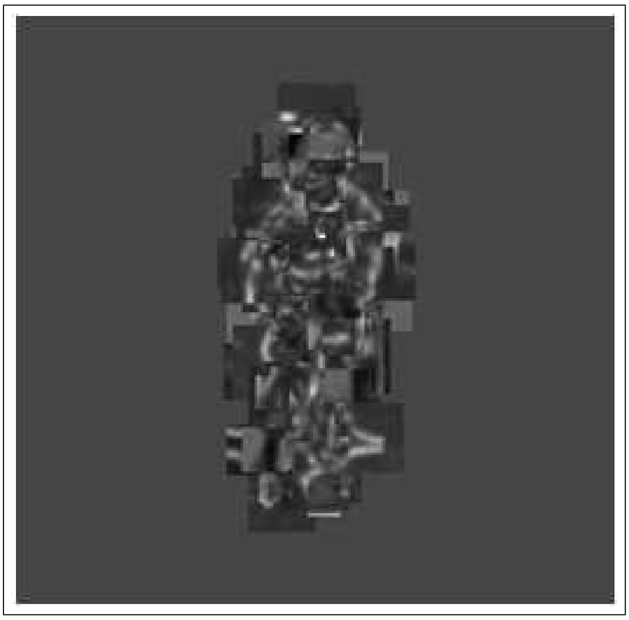


Figure 7: Dictionary entries selected by the multiclass Adaboost

3.2.4 Detection

To detect an object of class c in a test image I need to compute the score for every pixel in the image, provided by the strong classifier H(v,c) evaluated in all the pixels. If the score exceeds some threshold the object is detected. In order to calculate H(v,c) I use the following procedure.

I find all the shared weak learners that shares class c, and for each sharing weak learner:

1. Obtain the 4-tuple {f, θ, a_s, b_s} from the weak learner. Since f is associated with an entry in the dictionary, I retrieve the corresponding filter, patch and vectors L_x, L_y from the dictionary, and apply them to the test image using (1).

2. Calculate h_m(v)= a$\delta(v_f > \theta)$+b where V_f is the image obtained in the previous step. Finally I add up all the weak learners. Each weak learner votes for the center of the object sought, and it is expressed by a grayscale image obtained in step 2. The accumulated image will have bright pixels where the weak learners "agreed" about the center of the object in the "voting space". A maximum in the accumulated image indicates the probability to find the object in that location.

Each strong detector of a different class outputs an accumulated image. Thus, it is possible that more than one strong detector will vote for the same (or very close) pixel coordinates. This situation is not rare since some postures are very similar. To solve this conflict, peaks that are closer than a given radius are clustered together, and the resulting class of the detection is the one from the class with the highest maximum.

3.2.4.1 Torso and head orientation

The maximum voting schema, from the strong detector results, is one class from the four possible classes. Each class is associated with a torso orientation. Class 1 to 4 corresponds to 0, 90, 180, 270 degrees of rotation in the azimuth axes of the torso, accordingly. Once the class is selected, a bounding box with the size equal to the average size of the dictionary annotated boxes for that class, and which center corresponds to the peak of the maximum votes. A color bounding box with an arrow in the direction of the torso is plotted on the image tested to represent the direction where the torso is rotated towards. For head orientation, which is our proxy for gaze direction (this assumption holds when the object visualized is far enough from the observer, a different multiclass detector was trained around the head of the subject. This additional multi-class classifier was trained to detect the head in four orientations. The torso detection is performed first and determines a limited search region for the head. Experiments yielded the top 1/7th of the body detection area with added margins above the top to be a sufficient head search region. Should multiple body postures have been detected, a heuristic increases the search region, taking nearby body detections into account.

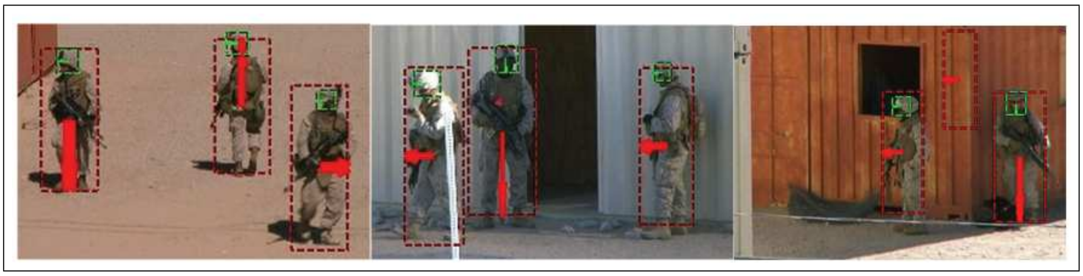


Figure 8: . Examples of head and torso recognition procedure in different scenarios

3.3 Hand Gesture Interaction

A hand gesture interaction system used in the OR for doctors/surgeons should follow the following specifications [36]: (1) Real time feedback and operation; (2) Low fatigue; (3) High intuitiveness; (4) Robust to unintended action; (5) Robust recognition; (6) Easy to use and to learn; (7) unencumbered (no wired attached). Following this considerations, our approach is described next.

3.3.1 Approach

Four pan/title/zoom network cameras placed in the vertices of the operating room's ceiling captures a sequence of images of the hand. The hand is tracked by a tracking module which segments the hand from the background using color and motion information. To clean the main object, morphological image processing operations are used. The location of the hand is represented by the 2D coordinates of its centroid which is sampled continuously resulting in a trajectory. These trajectories are mapped into a set of commands. For example, a flick gesture is the rapid movement of the hand from a neutral position to a specific direction, and return to the original position. 'Flick' gestures are used to navigate through the projected image on the walls of the OR. The doctors/surgeons intended actions/commands are recognized by extracting features from the spatio-temporal data of the gestures. Using the corresponding commands to which the gestures are associated, doctors can bring up X-rays images, select a patient record from the database or annotate a region on the image. A two layer architecture is used. The lower level provides tracking and recognition functions, while the higher level manages the user interface.

As opposed to field conditions, may raise challenges related to limited lighting conditions and unfixed environments. We plan to address this problem using simulation under operating room conditions to obtain a better assessment of system's performance.

3.3.2 Hand Segmentation and Tracking

A 2D histogram is generated in real-time during 'calibration' from the doctor's hand. The calibration process is initiated when the user places his hand slowly into a boundary without touching the screen. This, in turn, is used to build a hand color distribution model. A pixel at location (x, y) is converted to the probability that the pixel is classified as a hand (or gloved hand), in any frame using the 2D histogram lookup table created earlier (Figure 9).

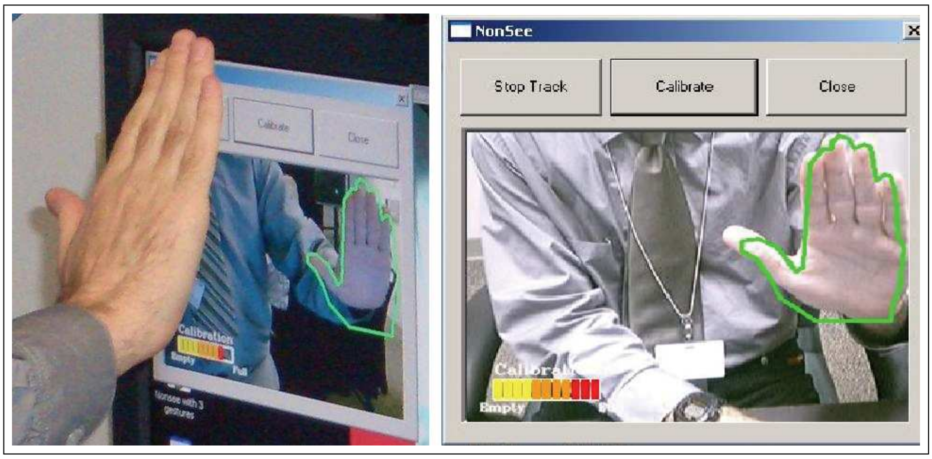


Figure 9: . User hand skin color calibration

In order to avoid false motion clues originated by non-hand motion in the calibration phase, a background maintenance procedure was developed. First, an initial image of the background is created. Changes are detected background differencing. When this difference is coherently significant, I assume that the present object is a hand. The background stored image is updated using a running smoothed average (9).

$$B_k(i,j) = \lambda \times f(i,j) + (1-\lambda) \times B_{k-1}(i,j) \tag{9}$$

Where, B_k is the updated stored background image at frame k, B_{k-1} is the stored background image at frame k-1, λ is the smoothing coefficient (regulates update speed), f(i, j) is the current background image at frame k.

To track the hand, CAMSHIFT is used [37]. It uses a probability distribution image comprised of pixels representing hand colors. This hand image is created from a 2D hue-saturation skin color histogram [38]. A histogram is used as a look-up-table to convert the acquired camera images of the hand into corresponding hand pixels, a process known as back projection. In the original CAMSHIFT algorithm the probability of a pixel belonging to the hand is determined by the grayscale value of the pixel only. In lieu of using color probability alone, I modify it with motion information according to (2) to represent a hand pixel probability. The relative weights between color and motion are shifted according to the amount of motion in the scene resulting in an adaptive fusion system. Using the centroid and size of the hand pixel an iterative procedure based on a generalization of the mean shift algorithm [39]. is used to update the tracking window at each frame. Thresholding to black and white followed by morphological operations is used to obtain a single component for further processing to classify the gestures.

3.3.3 Operation

The gesture interface can be used to browse medical databases and manipulate the projected imagery such as X-rays and MRIs. The finite state machine in Figure 10 illustrates the operational modes with the gesture interface. After the calibration procedure dynamic gestures are used to browse images and hand poses are used to switch between modes of operation. The central area in the middle of the frame is called the "neutral area", see Figure 11.

When a doctor decides to perform a specific operation on a medical image, he/she places the hand in the 'neutral area' momentarily, which will be followed by the zoom posture or rotation pose gesture.

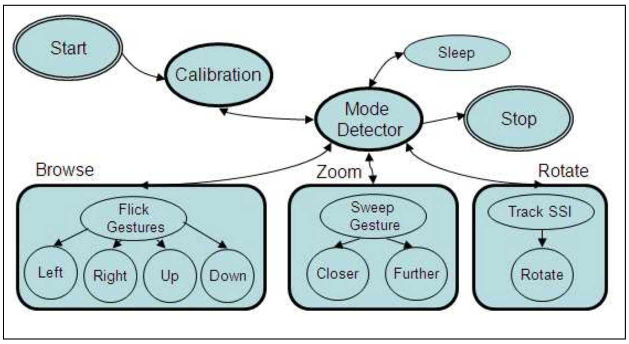


Figure 10: State machine for the gesture-based medical browser

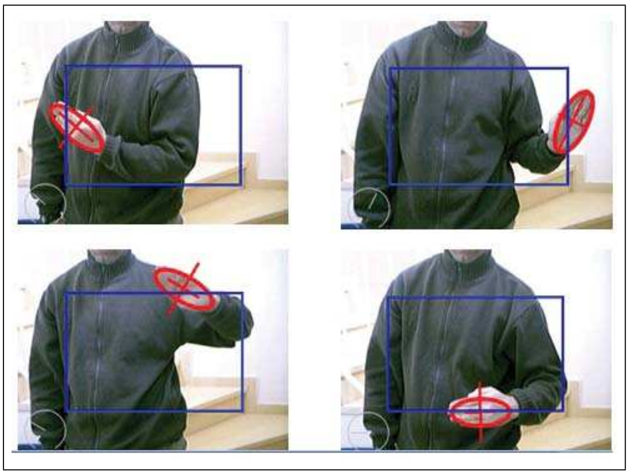


Figure 11: Four quadrants mapped to cursor movements

Navigation gestures are designed to browse through a medical data browser projected on the wall. The immersion sense is created by representing 3D objects, where each image is a side of a cube and arranged in numerous levels. The cube can be rotated CW and CCW, and moved up and down in the vertical direction to exhibit various levels on the screen. Hence, any image on the screen can be accessed directly by four navigation commands. A 'flick' gesture is performed when a doctor/surgeon moves the hand out from a 'neutral area' toward any of four directions, and then back to the neutral area. Interaction is designed in such a way that the gestures commands are carried out only when the doctor's body posture is standing still, instead of bending towards the patient (which may indicate that the surgeon is operating).

Zoom is used to change the size of an image. When the zoom-mode is triggered, the size of image changes according to the proximity of the palm to the camera. To go back to the normal mode, the hand is moved out from the neutral area to any of the 4 directions. The "zoom mode" is activated, when the hand is in the neutral area, rotates suddenly CCW by 90ş.

Rotation is achieved through the rotation of a sterilized straight instrument hold on the doctor's hand. In rotation mode, the angle to which the medical image is rotated is determined by the angle made by the instrument and the horizontal axis, in increments of 90ş. The orientation's instrument can be found using the Probabilistic Hough Transform (pHT). Only lines on the nearby proximity of the hand are considered (2.5 times around the interaction window).

To avoid the recognition of unintentional gestures, when the doctor wants to stop temporarily the operation of the hand gesture recognition system, he can either gaze down towards the patient, or user moves the hand to the lowest part of the screen, keeping it within the screen boundaries. To return to the "normal mode" a wake up gesture is used whereby the user waves the hand over the small rectangle in the upper left corner of the screen. This is useful when the doctor wishes to discuss details on the projected image without being "tracked" by the system.

3.4 Speech Interaction

One of the main goals regarding the intelligent operating room is twofold: (a) identify the user (ID tagging) automatically and (b) to recognize spoken utterances. The first goal, on one hand grants the user (either a nurse or the main surgeon) access to digital patient records and medical imaging tools according to the privileges that have been assigned to him, and on the second hand allows the room to keep track of the subject when ambiguity occurs due to light changing, occlusions and merging of several subjects. If I want to allow only the surgeon to manipulate and annotate the patient's medical images during surgery, the operation will be activated only when spoken utterances are recognized by the room as belonging to the surgeon's voice. To achieve this goal, the users will be required to say the word "login" [38] and compared to prototype feature vectors using some distance measure, such as maximum likelihood. For every participant in the OR, a profile is created and matched to the existing profiles in order to assess the level of permission that needs to be granted.

The second goal involves the recognition of words and sentences using off-the-shelf voice recognition software, called "DragonDictate", which can explicitly build continuous speech and context-free recognition grammars [39]. To transmit the speech to the recognition module, multiple microphones are used. Since I want to keep the sterility requirements, the microphones are not attached to the doctors. Instead, the microphones are set-up in key locations on the OR's ceiling.

Voice commands are used to evoke functions that are very difficult to map to hand gestures since there is no natural association between them. For example, to retrieve medical images of the patient being operated, the surgeon can say the patient's name. To discriminate between environment noise, which can mistakenly being recognized as a command, the user has to start the command by saying "system" shortly followed by the command to be carried out. This approach was suggested in [39] in the context of smart rooms, where the vision recognition and audio recognition modules are independent, and hence it fits the requirements for the operation room. Environment noise can still be too high and interfere with the interaction. We plan to test these scenarios in further work.

4 Conclusions and Future Works

This work presents one possible application for smart rooms: the intelligent, collaborative operating room. By monitoring the surgeon's activities while performing surgery, the system can collaborate with her by displaying relevant medical imaging information in a convenient location in the OR. The OR depends on the surgeon's body language as the essential key to understanding his focus of attention. One goal of the suggested approach is to use computer vision techniques to detect body postures and gaze to determine focus of attention. Smart projectors combined with computer graphics techniques are used to project medical imaging in front of the surgeon, based on the knowledge provided by the posture recognizer. Person-detection with simultaneous human body posture recognition is achieved using parts-based models and a multiclass boosting approach: each body part is matched with the image captured by the camera, and each part votes for the center of the person. Focus of attention is assessed by simultaneous detection of the surgeon's posture and orientation. Detection and classification are possible since features of different human postures have shared subspaces as opposed to the non-person class. One challenge here is the focus on postures that cannot be easily distinguished by their aspect ratio or silhouette, but rather require a bottom-up approach. Parts-based detection does not require explicit models, nor the labeling of individual body parts. The detection and posture classifications are performed in a single pass over the image, and the strength of the recognition is proportional to the ensemble of votes from parts patches.

A vision-based system is used for interaction with the medical images,. It recognizes the surgeon's gestures in real-time which are used to browse, zoom and rotate the images projected on the wall in front of him. The system is user independent since it is calibrated using a multi-modal two step procedure: first the user's ID is recognized using a voice identification system, then cameras are used to model the gamut of colors of the surgeon's hand. Camshift is used to track the hand, which allows dynamic hand gesture navigation control. The decision to use hand gestures as the main modality of interaction is based on previous work done by the author [25], where it was shown that hand gesture interaction in the operating room is the preferred modality of interaction by the doctors in the OR, due to their proficiency at using the hand as their main tool of work. Hand gestures offer the following benefits: (i) Ease of use: - Surgeons are already quite proficient in their use of hands as a primary work tool, (ii) Rapid reaction: - hand gesture commands are intuitive and fast, (iii) Unencumbered: - does not require the surgeon to be wired to any device, and (iv) Sterility: - non contact interaction.

Issues related to image processing performance algorithm under unfixed environments require further analysis and more robust vision algorithms. For example, in [25] we used only one frontal camera, and in the current research I implement four. It is not likely that the surgeon will be in front of any of these cameras. To correct for this, the image need to be re-projected using the homography matrix found in the calibration process. Still some areas may remain occluded. In future work these challenges will be addressed and a simulation framework is going to be adopted in order to quantitatively validate the hypothesis suggested in this work.

Surgeons must command a high level of cognitive and motor skills in order to complete their tasks successfully. Previous research has shown that during the performance of a specific task, most gaze changes are related to the task-oriented visual search. Distractions may have an adverse effect on task performance since they compete with the mental resources allocated for that task. I specifically address the problem of the distraction to the surgeon, (or shift of focus) required for closer examination of medical imaging. I alleviate this situation by displaying the medical imaging in a convenient size right in front of the surgeon, automatically detecting his current orientation.

In more general terms, I believe that hand gesture, body posture and voice recognition can be used to help an intelligent system understand the context of the work being performed in an operating room.

Intelligent rooms equipped with hand gesture, body posture and voice recognition capabilities can assist surgeons in the execution of time-or-safety critical tasks, while providing him/her with natural, intuitive and easy to use interaction modalities.

Bibliography

[1] T.R. Hansen ,*Focus Shift Analysis in the operating theatre*,. 2004. Source:*http : //www.pervasive − interaction.org/publications/Focus_shift.pdf*.

[2] J. Graydon, M. Eysenck, *Distraction and cognitive performance*, Eur J Cogn Psychol.;1(2):161-179, 1989.

[3] V. Weerdesteyn, A. Schillings, G. Van Galen, D. Duysens, Distraction affects the performance of obstacle avoidance during walking, J Motor Behavior. 35(1):53-63, 2003.

[4] K. H. Goodell, C. G. Cao, S. D. Schwaitzberg, *Effects of cognitive distraction on performance of laparoscopic surgical tasks*, J Laparoendosc Adv Surg Tech A. Apr;16(2):94, 2006.

[5] R. M. Satava, *Disruptive visions*: The operating room of the future. Surgical endoscopy, vol. 17, no1, pp. 104-107, 2003.

[6] J. V. Sutherland, W. J. Van den Heuvel, T. Ganous , M. M. Burton, A. Kumar, Towards an Intelligent Hospital Environment: *Adaptive Workflow in the OR of the Future, Stud Health Technol.* Inform.118:278-312, 2005.

[7] P. Wellner,The DigitalDesk Calculator: *Tangible Manipulation on a Desk Top Display, Proceedings of UIST'91*. pp.27-33. 1991.

[8] W. Newman, P. Wellner, *A Desk Supporting Computer based interaction with paper, Proceedings of the Conference on Human Factors in Computing Systems* (CHI'92). p587- 592. 1992.

[9] Microsoft Surface. Available online: www.microsoft.com/surface/

[10] R. A. Brooks, *The Intelligent Room Project*, Proceedings of the Second International Cognitive Technology Conference (CT'97), Aizu, Japan, August 1997.

[11] K. S. Huang and M. M. Trivedi, Video Arrays for Real-Time Tracking of Persons, Head and Face in an Intelligent Room, *Machine Vision and Applications*, Special Issue, Jun. 2003.

[12] M. Nicolescu and G. Medioni, *Electronic pan-tilt-zoom*: a solution for intelligent room systems, Proc. ICME'2000, 2000.

[13] J. C. Terrillon, A. Piplr, Y. Niwa, K. Yamamoto, *Robust Face Detection and Japanese Sign Language Hand Posture Recognition for Human Human-Computer Interaction in an "Intelligent" Room*, In Proc. Int'l Conf. Vision Interface, 2002.

[14] N. Dimakis, J. K. Soldatos, L. Polymenakos, P. Fleury, D. Curin, Jan, J. Kleindienst, *Integrated development of context-aware applications in smart spaces, Pervasive Computing, IEEE 7* (4), 71-79, 2008.

[15] T Skordas, G Metakides, Major Challenges in Ambient Intelligence, *Studies in Informatics and Control*, Vol.12, No.2, June 2003 Page 3. 2.3.

[16] J. P. Tardif, S. Roy, M. Trudeau, Multi-projectors for arbitrary surfaces without explicit calibration nor reconstruction, In Proceedings of Fourth International Conference on 3-D Digital Imaging and Modeling. 3DIM 2003.. pp. 217-224. 2003.

[17] A. Griesser and L. V. Gool, Automatic Interactive Calibration of Multi-Projector-Camera Systems, Proceedings of the 2006 Conference on Computer Vision and Pattern Recognition Workshop, 8, 2006.

[18] I. Mitsugami, N. Ukita, M. Kidode,.'Displaying a Moving Image By Multiple Steerable Projectors', IEEE International Workshop on Projector-Camera Systems (Procams2007), 2007.

[19] Z. Jiang, Y. Mao, B. Qin and B. Zang, A High Resolution Video Display System by Seamlessly Tiling Multiple Projectors, IEEE International Conference on Multimedia and Expo, 2007.

[20] E. Bhasker, R. Juang, A. Majumder, Registration Techniques for Using Imperfect and Par tially Calibrated Devices in Planar Multi-Projector Displays, IEEE Transactions on Visualization and Computer Graphics, 13:6, 1368-1375, 2007.

[21] J. Summet, M. Flagg, TJ. Cham, JM. Rehg and R. Sukthankar, Shadow Elimination and Blinding Light Suppression for Interactive Projected Displays,IEEE Transactions on Visualization and Computer Graphics, 13:3,508-517, 2007.

[22] T. Johnson, G. Welch, H. Fuchs, E. La Force, H. Towles, A Distributed Cooperative Framework for Continuous Multi-Projector Pose Estimation in Proceedings IEEE VR 2009.

[23] A. Nishikawa, T. Hosoi, K. Koara, D. Negoro, A. Hikita, S. Asano, H. Kakutani, F. Miyazaki, M. Sekimoto, M. Yasui, Y. Miyake, S. Takiguchi, and M. Monden. FAce MOUSe: *A Novel Human-Machine Interface for Controlling the Position of a Laparoscope*, IEEE Trans. on Robotics and Automation, 19:5, 825-841, 2003.

[24] C. Graetzel, T.W. Fong, S. Grange, and C. Baur, *A non-contact mouse for surgeon-computer interaction, Technology and Health Care*, 12:3, 245-257, 2004.

[25] J. Wachs, H. Stern, Y. Edan, M. Gillam, C. Feied, M. Smith, and J. Handler, A hand gesture sterile tool for browsing MRI images in the OR, *Journal of the American Medical Informatics Association.* 15(3), 2008.

[26] P. Lukowicz, A. Timm-Giel, H. Lawo, O. Herzog, WearITwork: *Toward Real-World Industrial Wearable Computing*, IEEE Pervasive Computing. 6:4,8-13, 2007.

[27] Y. Kuno, T. Murashima, N. Shimada, and Y. Shirai, Intelligent Wheelchair Remotely Controlled by Interactive Gestures, In Proceedings of 15th Intl. *Conf. on Pattern Recognition*, 4,672-675, 2000.

[28] T. Starner, J. Auxier, D. Ashbrook, and M. Gandy. The Gesture Pendant: A Self-illuminating, Wearable, Infrared Computer Vision System for Home Automation Control and Medical Monitoring, In 4th Intl. Symposium on Wearable Computers, 87-94, 2000.

[29] D. A. Becker and A. Pentland, Staying Alive: *A Virtual Reality Visualization Tool for Cancer Patients*, Proc. of the AAAI'96 Workshop on Entertainment and Alife/AI, 1996.

[30] R. Boian, R. Sharma, C. Han, A. Merians, G Burdea, S. Adamovich, M. Recce, M. Tremaine, H. Poizner, *Virtual reality-based post-stroke hand rehabilitation,Studies in Health and Technology Information*, 85, 64-70, 2002.

[31] M. Ashdown and Y. Sato, Steerable projector calibration, Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), 2005.

[32] E. Moschos, R. Coleman, Acquiring laparoscopic skill proficiency: Does orientation matter?, *American Journal of Obstetrics and Gynecology*, 191:5, 1782-1787, 2004.

[33] R. Yang and G. Welch.,Automatic and Continuous Projector Display Surface Estimation Using Every-Day Imagery, 9th International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision, 2001.

[34] K. Murphy, A. Torralba, D. Eaton, W. T. *Freeman,Object detection and localization using local and global features*, Lecture Notes in Computer Science. Sicily workshop on object recognition, 2005.

[35] J. Wachs, D. Goshorn and M. Kölsch, *Recognizing Human Postures and Poses in Monocular Still Images*, In Proceeding of the International Conference on Image Processing, Computer Vision, and Signal Processing. (IPCV09) Las Vegas, Nevada. July 2009.

[36] J. Wachs, H. Stern, Y. Edan, M. Gillam, C. Feied, M. Smith, and J. Handler, *A Real-Time Hand Gesture Interface for a Medical Image Guided System*, International Journal of Intelligent Computing in Medical Sciences and Image Processing, 1:3:1, 175-185, 2008.

[37] G. R. Bradski, *Computer vision face tracking for use in a perceptual user interface*, Intel Technical Journal, 1-15, 1998.

[38] S. T. Shivappa, M. M. Trivedi, and B. D. Rao, *Hierarchical audio-visual cue integration framework for activity analysis in intelligent meeting rooms*, IEEE CVPR 2009, 2009.

[39] M. Coen, *Design principles for intelligent environments*, In Proc. of AAAI, Madison, Wisconsin, 1998.

Juan Wachs is an assistant professor at Purdue University in the School of Industrial Engineering. He completed a postdoctoral training at the Naval Postgraduate School's MOVES Institute in the area of computer vision. His research interests are machine and computer vision, robotics, tele-operations, human factors, assistive technologies and health support systems. Juan Wachs is a member of IEEE and the Operation Research Society of Israel (ORSIS). He has published in journals including IEEE Trans. Systems, Man, and Cybernetics, Journal of American Medical Informatics, Journal of Image and Graphics, and the International Journal of Semantic Computing. He has been a visiting professor at the Buenos Aires University, Argentina in the area of Image Processing. He has taught digital electronics, multimedia and web development at ORT and Intel Colleges, Israel. He received his B.Ed.Tech in Electrical Education from the ORT Academic College in Jerusalem, his M.Sc and Ph.D in Industrial Engineering and Management, Information Systems and Intelligent Systems tracks, respectively, from the Ben-Gurion University of the Negev.

Int. J. of Computers, Communications & Control, ISSN 1841-9836, E-ISSN 1841-9844
Vol. V (2010), No. 1, pp. 125-137

Swarming Models for Facilitating Collaborative Decisions

C.B. Zamfirescu, F. G. Filip

Constantin-Bala Zamfirescu
Lucian Blaga University of Sibiu,
Faculty of Engineering, Department of Computer Science and Automatic Control
Romania, 69121 Sibiu, 17 Emil Cioran
E-mail: zbc@acm.org

Florin Gheorghe Filip
1. Romanian Academy - INCE and BAR
Romania, 010071 Bucharest, 125 Calea Victoriei, and
2. Nat. Institute for Informatics -ICI
Romania, 011455 Bucharest, 8-10 B-dul Maresal Al. Averescu
E-mail: filipf@acad.ro

Abstract:
The paper highlights the computational power of swarming models (i.e., stigmergic mechanisms) to build collaborative support systems for complex cognitive tasks such as facilitation of **group decision processes** (GDP) in e-meetings. Unlike traditional approaches that minimize the cognitive complexity by incorporating the facilitation knowledge into the system, stigmergic coordination mechanisms minimize the complexity by providing the system with emergent functionalities that are shaped by the environment itself through the possibility to structure it in terms of high-level cognitive artefacts. This is illustrated by conducting a socio-simulation experiment for an envisioned collaborative software tool that acts as a stigmergic environment for modelling the GDP. The results show superior results when the users are allowed to increase the representational complexity of a GDP model with cognitive artefacts that support guidance and action in the conceptual problem space.

Keywords: collaborative working environments, group decision support systems, facilitation, social simulation, stigmergy, swarming models of computation.

1 Introduction

The simplicity of swarming models is becoming ever more popular in the design of decentralized systems that are developed to run in open, dynamic, and unknown environments. Inspired from the behaviour of social insects, marine animals, birds, and even humans, the concept covers a broad spectrum of mechanisms able to generate an intelligent collective behaviour. These mechanisms are simply identified as *stigmergic coordination*[1].

The most cited example of swarming models is the food foraging behaviour in ant colonies [2]. Each ant senses the signs (pheromones) in its environment and acts in accordance with them without any direct communication with other ants from the colony. If there is no sign in the environment the ant executes a randomized search for food. When an ant discovers a food source, it drops a smelling chemical substance (pheromone) on its way back to the nest while carrying a bit of food. Thus the ant creates a pheromone trail between nest and food source. When an ant senses the pheromone trails, it will follow the most intense one to the food source, the intensity of the pheromone signifies the shortest path toward the food that was discovered until that point in time; when the ant arrives at the food source, it will return with food, while depositing more pheromones, intensifying the pheromone trail. The above simple reactive behaviour of each ant results in an emergent intelligent behaviour of the colony that is able to find the shortest path from the nest to the food source without any central coordination. This behaviour is constrained by the limited sensorial aptitude of an ant to sense the local pheromone trails with no mental plan on how to find the shortest path or knowledge about the environment in which they act. Consequently, the ant's behaviour is an emergent property induced at the same time by two exogenous factors: 1) the environment (the surface of the terrain and the evaporation mechanism for the pheromones), and 2) the ants' actions over the environment (the pheromone trails).

Despite its behavioural simplicity (i.e., stimuli-response rules), a single stigmergic agent can emulate any Turing machine and can execute any symbolic or sub-symbolic algorithm proposed by the AI (artificial intelligence) research mainstream [3]. Nevertheless, most of the computer applications are outside the domain of human cognitive abilities, serving merely as cognitive support systems through an active and semantically rich environment [4]. Therefore swarming models of computation are seen as a feasible approach to construct systems that are not limited to the classical optimization problem of finding the shortest route [5], but are able to support human decisions as well [6, 7, 8].

One application domain with a high level of cognitive complexity is the facilitation of GDP in e-meetings. The complexity associated with the construction, coordination and execution of GDP is well recognized in the research field of **Group Decision Support System** (GDSS) [9]. For this reason GDSS has seldom been a full success story and a widely adopted technology as it has been foreseen by its pioneers. All the applications developed to support the group facilitation in e-meetings follow the traditional centralized approach where the system explicitly codifies the facilitation knowledge. Examples include software tools that embed knowledge about the collaborative patterns of interaction [10] and workflows [11] for the most frequently used GDPs. These applications basically suffer from the same obstacles met in the traditional AI mainstream such as [12]: 1) the restrictions to codify the human's knowledge into the computing system; 2) the lack of self-development capabilities for this knowledge; 3) the black-box perspective over a system disconnected from the environment where the relevant knowledge are extracted.

In contrast to the traditional approach that codifies the facilitation knowledge into the GDSS, this paper illustrates how the stigmergic mechanisms of swarming models may be employed to build emergent and self-organizing functionalities that support group facilitation in e-meetings. In this particular case, the stigmergic coordination mechanisms are implemented over the conceptual environment of the GDP modelling space which is exploited by the users of an e-meeting system. Unlike conventional approaches that minimize the cognitive complexity associated with the construction and execution of a GDP by incorporating the facilitation knowledge into the system, stigmergic coordination mechanisms minimize it by structuring a shared conceptual environment populated with the cognitive artefacts that represents the basic skills of conducting an e-meeting. In this way the users are collectively constructing and interpreting the facilitation knowledge through the successive uses of the system.

The remaining part of this paper is organized as follows. The next section presents a brief analysis of the innate relationship between distributed cognition and stigmergy as it has been presented by many authors within the Web 2.0 technology mainstream. Section 3 describes the main components of an envisioned collaborative software tool that act as a stigmergic environment for modelling the GDP: the structure of the semantic environment, the low-level behaviour of the users in interacting with this environment, and the high-level of cognitive constructs that may be employed by the users to structure the GDP. These components are implemented and tested in a socio-simulation experiment which is described in Section 4. The experimental results show clear self-organizing capabilities, but simultaneously high dependability of system's performance on the user's ability to structure the stigmergic environment. From the engineering standpoint of constructing purposeful facilitation tools for e-meetings, these results are discussed and summarized in the last section.

2 Stigmergy and cognition

Most of the research in cognitive science is rooted in the basic assumption that cognition is purely an internal process of representation and manipulation of knowledge disconnected from the environment. Nevertheless, recent studies in cognitive sciences reveal the essential role of the environment (physical or artificial) in mediating the knowledge by facilitating their external representation and information exchange among these representations [13]. It becomes clear 'that individuals are socially and culturally situated and that the environment needs to be considered in order to understand cognition' [14].

By admitting the critical role of the environment in cognition, the researchers became increasingly aware of the relationship between cognition and stigmergy. Even if the term stigmergy has been primarily used for typically reactive (non-rational) agents, its relationship with cognition was investigated for the first time by Susi and Ziemke [15]. The authors conclude with the assertion that the conceptual framework of stigmergy offers a common denominator for the social sciences theories (i.e., activity theory; situated and distributed cognition). For example a cave painting is an emblematic case of stigmergic coordination where people used the physical environment to indirectly communicate their knowledge in ancient human society. The relevance to cognition of the stigmergic coordination mechanisms in human society is illustrated by Parunak in several examples of social activities [6].

Albeit the use of stigmergic mechanisms may be observed in various social activities, it is more evident in the digital world realm. In a comprehensive study of collaborative support systems, Elliot [16] found that stigmergy is a coordination mechanism inherent not only in collaborative processes over physical environments, but also in a range of collaborative support systems. For instance the plethora of applications that are considered to be Web 2.0 technology (especially media such as wiki and community blogging) is generally recognized to be stigmergic systems. Moreover, applications such as Google's PageRank system, eBay's online auctioning, Amazon's recommender systems [6], Wikipedia.org, open source software and multiplayer social environment Second Life [17] are employing the stigmergic coordination mechanisms to exhibit the functionalities of an intelligent collective behaviour.

In these stigmergic systems the users exploit their digital environment through the use of engineered artefacts that may be annotated with symbolic information representing the human's cognition [7]. Given that the essential capability of any stigmergic system is to transfer the cognitive complexity from the humans to the environment [18], the problem-solving capabilities of the users decisively depend on how the problem is represented in the digital environment. A standard representation of the problem in the environment is realized as a composition of cognitive artefacts linked in a weighted graph. Basically, this graph signifies a navigation map that supports the cognitive effort to find and reach any artefact from the place where it is needed. The artefacts commonly stand for the possible states of the problem, while the links are the set of possible actions that guide the decision process (the conceptual navigation) from one state to another of the problem space. Similar with the intensity of the pheromone trails in the case of real ants, these actions are weighted in order to discriminate the most effective ones. As a result, the improvement of problem-solving ability requires two corresponding processes [8]: 1) the augmentation of the environment with additional states and actions to increase the accuracy of problem representation, and 2) the improvement of the preference function for an action in order to compensate the expansion of the exploration space.

The wide employment of the stigmergic patterns of interaction in collaborative working environments has been triggered by its fundamental advantage of preventing the humans' cognition to be exposed to the complexity of the environment [19]. As users interact only locally, there is no need for tasks allocation, the tasks being preferentially performed by the most expert since they are the most attracted to act and finalize the task with minimal effort [20]. Moreover there is no need for prediction since the environment records actions in the problem space and the unexpected events are automatically traced through the outcome of the users' actions over the environment. All these advantages make the stigmergic models of computation a suitable approach to support the cognitive complexity of facilitating GDP, a process that runs in a dynamic, open and uncertain environment.

3 A simulation model for group decisions in e-meetings

The wide range of tools that support group decisions in e-meetings falls under the GDSS general umbrella term. GDSS is defined as an interactive computer-based environment that supports a concerted and coordinated team effort towards completion of joint tasks [21]. A GDSS is composed of a set of highly configurable collaborative "tools" (e.g., brainstorming, voting and ranking, multi-criteria analysis, etc.) that requires a high level of expertise for an effective use for complex decisions [22]. The strong relationship between the GDP outcome and the presence of a skilful facilitator to direct the joint decision process is thoroughly presented in many field studies of GDSS research [23]. The inaccessibility of many organizations to a well-trained GDSS facilitator is recognized to be one of the main obstacles which limit the adoption of GDSS technology [24]. To reduce the dependence on the facilitator, the participant-driven GDSS was proposed as the most promising research direction to leverage the skills and abilities of each group member [25]. However, this approach is highly constrained by the cognitive complexity associated with the construction, coordination and execution of GDP by inexperienced users.

To overcome the problem of cognitive complexity Briggs and de Vreede [24] introduced the **thinkLet** (TL) concept as a discrete facilitation unit that integrates a specific tool, its configuration and a script to use it - a predefined interaction protocol among users that is enforced and mediated by a specific collaborative tool. This concept was anticipated by the declarative **model of the experienced decision maker** (MEDM), which was proposed in late 80's to help the user of a DSS to build the model, select the appropriate solver, and evaluate various solutions provided by the computerized algorithms in so called "mixed knowledge" DSS [22, 26]. The TLs are considered to be the smallest piece of essential knowledge to design collaborative processes. Examples include [24]: StrawPoll (Evaluate) - used to reveal the agreements or disagreements within a group; Leafhopper (Diverge) - used when the participants does not known in advance the topics for discussion or they have different interests or level of expertise; ExpertChoice (Organize) - used when the ideas of the group are organized by a single participant; ReviewReflect (Converge) - used when a different reviews from group members for a document should reach an agreement etc. With TLs, the conceptual model for a GDP takes the form of a shared plan of collaborative actions [24, 27]. Each collaborative action is an interaction protocol embodied in a TL. As any plan, the model for a GDP may be hierarchically decomposed in sub-plans at different levels of abstraction. This conceptual structure of a GDP model is acknowledged in any application domain of the GDSS technology, such as project management [28], user requirement elicitation [29], crisis response management [30], scenario design [31], risk identification [32], etc.

In the view of envisioning a collaborative software tool that acts as a stigmergic environment for modelling the GDP (in the same way in which a collaborative CAD software acts as a stigmergic environment for architectural design [33]), we developed a socio-simulation model that mimics the users' conceptual 'navigation' over the semantic structure of the problem space for facilitating the e-meetings. As for any stigmergic system the simulation model entails the description of agents' behaviour and the structure of the shared environment where the agents are localized and moved over it.

For the GDSS domain, the agents are the users responsible to define, execute and evaluate a GDP model which is a path through the conceptual space of the available TLs. The environment is the collaborative facilitation tool that supports the conceptual representation of the problem space comprising all the TLs discovered and documented by the users' community (so far there are over 70 TLs acknowledged in literature [34]).

3.1 The semantic environment for facilitating the e-meetings

According to Parunak [6], a stigmergic environment assumes the definition of three main components: 1) topology, 2) states, and 3) processes. Structurally, the *topology* may be viewed as a fully connected weighted graph that codifies the facilitation knowledge of group decision in e-meetings. This knowledge presumes correlated information among the users and the TLs, reflecting the users' evaluation of the performance for a TL (a node in the graph) relative to a problem type. The performance is stored for each problem type in a variable associated with each edge of the graph. The problem type is simply codified through a unique *ID* to distinguish among different performances when they are read, during the modelling phase of the GDP, or modified, after the GDP has been executed and evaluated by agents. Evaluation of a GDP model entails a subjective assessment of the model against some performance criteria after its execution. Regularly, GDP's performance may be quantified in terms of efficiency, effectiveness and users' satisfaction, as illustrated in several studies from the GDSS research field [35].

The performance from all the graph's edges describes the state of the environment over time. Usually, the environment executes a set of processes on the variables (as aggregation end evaporation in the case of ants). For our case, we apply a simple weighted additive rule to simulate the aggregation of performances:

$$P_{jk}(TL_k,t) = P_{jk}(TL_k,t-1) + UP_{jk}(TL_k)/w \tag{1}$$

where: t represents the temporal component of the model which is incremented by one for each successive use of the GDSS; k is the TL's identification index from the set of TLs used to model the GDP;$UP_{jk}(TL_k)$ - is the user's performance of the k-th TL evaluated from the side of TL j at moment t;$P_{jk}(TL_k,t)$ and $P_{jk}(TL_k,t-1)$ are the new and previous values of the (collective) performance stored on the edge between the TLs j and k; w is a tuning parameter, arbitrarily chosen, to weight the impact of the last evaluation.

3.2 The agents' behaviour over the semantic environment

The *agents* are the users who interact with the envisioned collaborative tool to model the GDP. Conceptually, in any point in time an agent is "located" in a node (TL) of the cognitive environment of the problem space, performing one of the following basic actions: 1) evaluates the preference for the next possible TL (or TLs) that are going to be executed given the current execution context of the GDP; 2) selects the next best TL (or a group of TLs) for further completing the GDP model; 3) executes the TL (or the group of TLs) from the model, and finally; 4) evaluates the performance for the executed TLs. The evaluation activity is simulated using the formula (1), while the first three actions with Luce's selection axiom [36]:

$$p_{jk} = \mathrm{e}^{p_{jk}(TL_k)/T} / \sum_{i=1}^{m} \mathrm{e}^{p_{ji}(TL_i)/T}, \tag{2}$$

where p_{jk} represents the preference for an alternative TL, i.e. the selection probability of the TL k from the TL j; i is the index of TLs connected from the side of node j (in fact all the m TLs available in the problem space as long the graph is fully connected); and T is a parameter used to define the deviation from a pure rational behaviour.

The above formula is the most common model of stochastic decisions due to its correlation with the psycho-social observations of human behaviour in several domains. As a result of normalization, the preferences for the unexploited TLs are diminishing after each performance update. This mechanism replicates the pheromone evaporation process of the real ants (e.g., even if a TL has been positively evaluated after an execution of a GDP model, the associated preference will decrease once a better alternative is discovered and more frequently used). The uncertainty associated with the construction of preferences is generally modelled in equation (2) with the parameter T that range between 0 (when selection is deterministic as is the ideal case of a perfectly informed decision) and 1 (when the selection is completely random as in the case of a completely irrational decision). Note that Luce's selection axiom does not specify the reasons of uncertainty which for the modelling of GDP may cover any aspect of complexity, unfeasibility, cost or even refusal to evaluate the performance of a TL after its execution.

3.3 Navigation strategies over the semantic environment

The *agents*, who are reflecting the users in modelling a GDP, are engaging in means-ends reasoning activities to achieve the group decision goal. During the execution of the model they must be able to adapt to the changes and uncertainties associated with both the execution and the decision goal. Moreover, when the decision problem is not completely clear or too complex to be fully tackled, the users are defining intermediate sub-goals to be subsequently achieved. As a result, the design of a GDP model is most often done incrementally and interleaved with its execution.

Table 1: Different modelling strategy to construct a GDP model

Design strategies (DS)	**Execution**	**Problem's goal**	**Codification**
DS1	Certain	Stable	The problem type is codified through a unique ID
DS2	Uncertain	Stable	The problem type is codified as a variation from the current state of execution to the desired one
DS3	Uncertain	Unstable	The problem type is codified as a variation from the current state of execution to any future possible state

In Table 1 we have summarized three basic **design strategies** (DS), with direct implications on the way the GDP model is decomposed on different levels of abstractions. These are:

- *DS1* which corresponds to the traditional use of GDSS when the GDP model is predefined by the facilitator. In this case, the user is providing a complete structure of the GDP model, the facilitator having a complete vision over the execution plan for the GDP. It includes all the necessary collaborative actions (in the form of TLs) together with their precedence constraints. This design strategy relates to the hypothetical conditions when the execution context remains stable in time as regards the TL execution's outcome and the decision's objectives. Thus, each problem type is identically codified in all edges that connect the TLs used in modelling the GDP and relates to a low level of semantically structured environment.

- *DS2* which corresponds to problem types with stable objectives but uncertain TL execution's outcome from the GDP model. In this case, after the execution of each TL, adjusting the remaining GDP is needed. This design strategy is codified in the conceptual graph of GDP modelling with different *IDs* for each sub-problem that corresponds to the variance from the current state of execution to the desired one.

- *DS3* which relates to complex plans of actions for the GDP model, where both the outcome of an activity and the decision's objectives are unstable during the execution. In this case, when the decision problem is not entirely clear or too complex for designing an entire GDP model, the users are defining intermediate sub-goals that are subsequently dealt with. This design strategy is codified in the conceptual graph for the GDP modelling with different *IDs* for each sub-problem that corresponds to the variance from the current state of execution to any future possible one.

4 Experimental results

To evaluate the design strategies for modelling the GDP we conducted a virtual experiment following the research methodology proposed by Carley [37], implementing in the Netlogo multi-agent simulation environment [38] the model described in the previous section. In the experiment the users ("turtles") are to engage themselves in the facilitation of e-meetings, trying to define the GDP model for a problem type by moving in the conceptual graph of TLs (the nodes and edges are implemented also as "turtles"). The number of TLs that compose the graph is arbitrary chosen from the interface (the "num-TLs" variable in Figure 1), while their utilities for a certain problem type is predefined with random values between 0 and 1 when the experiments are initialized. Note that the NetLogo implementation includes some additional variables required to analyse if the model presents similar results to those reported in the traditional ethnographical studies. This issue is beyond the scope of this paper and is detailed in [39].

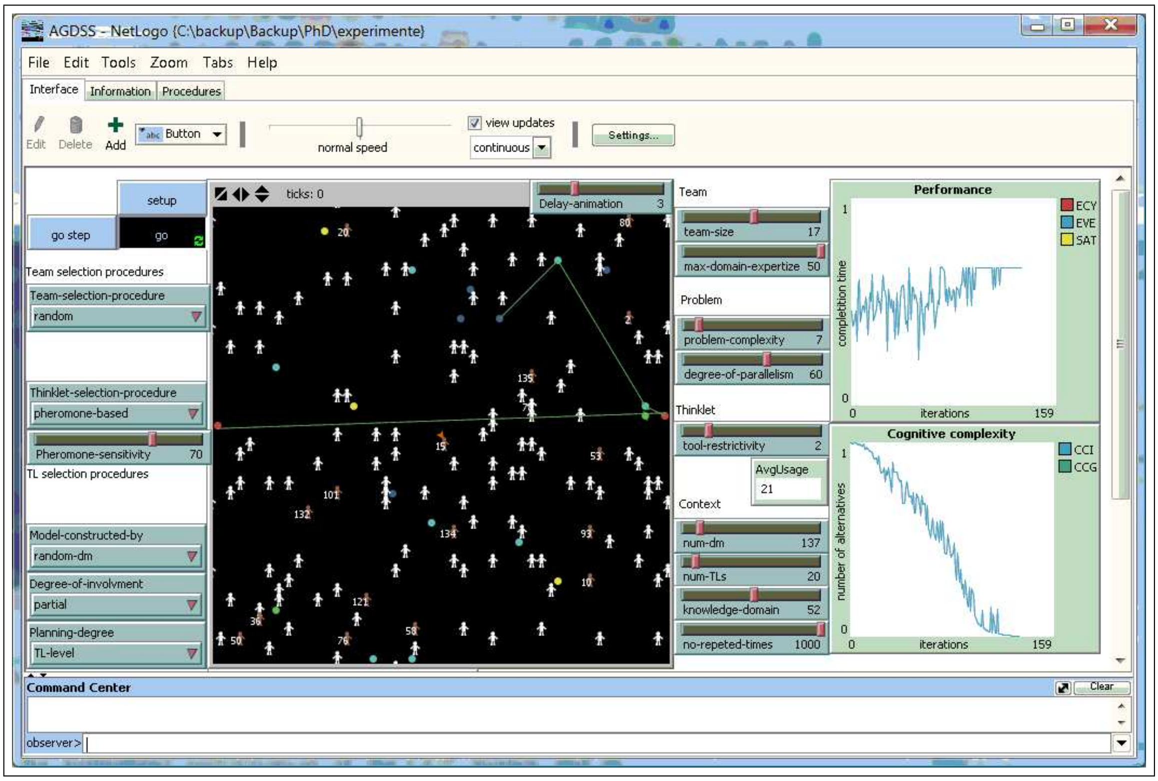


Figure 1: The interface of the model in the Netlogo environment

In the next sections the normalized performance of the GDP models and its associated entropy for 100 successive explorations (iterations) in the three design strategies (the "Planning-degree" variable in Figure 1): *DS1*, *DS2* and *DS3* are presented. An exploration stands for a complete execution cycle of a GDP. It includes three consecutive phases: 1) finding a suitable model through the successive selection (using the Equation (2)) of TLs that compose the GDP for the given problem type; 2) executing the identified model and assessing its performance by reading and averaging the predefined utility values of all the TLs that compose the GDP model; 3) evaluating the model by updating the performance value (using the Equation (1)). The statistics are aggregated from 30 experiments for a relatively simple problem type of 5 successive TLs. The parameter T from Equation (2) is set to 0.7 (the "pheromone-sensitivity" variable from Figure 1 to favour a faster convergence rate in finding a suitable solution in the problem space composed of 70 TLs.

From the engineering viewpoint, the design strategies have direct implications on the way in which the cognitive environment for modelling the GDP is structured. As described before, the structure of the environment is simply reflected in the possibility to semantically decompose the problem in sub-problems on different levels of abstractions. This design issue implies in our implementation to record with specific *IDs* the performance for each sub-problem type that emerges from the decomposition process.

4.1 The impact of the modelling strategies over the model's performance

In Figure 2 the aggregate performance (a relative number between 0 and 1 as resulted from averaging the predefined utility values of all the TLs that compose the GDP model) from 30 experiments of a GDP modelling process for the defined design strategies *DS1*, *DS2* and *DS3* are shown. As may be expected, the performance fits an exponential function, a typical behaviour for a stigmergic system [40, 41, 42].

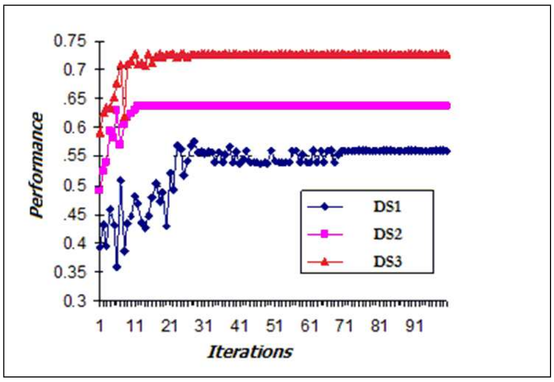


Figure 2: The GDP models' performance for the defined modelling strategies

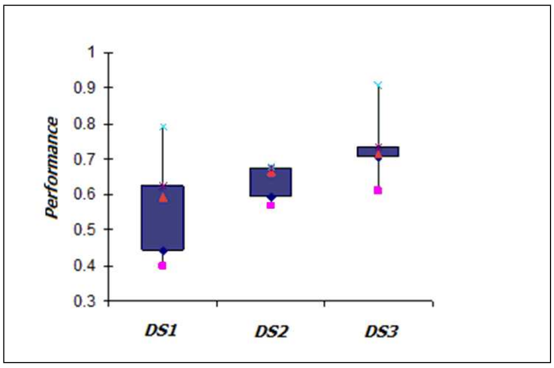


Figure 3: The distribution of GDP models' performance for the three modelling strategies

Like any heuristic model, the stigmergic coordination mechanisms do not guarantee finding an optimal solution, but a near-optimal or acceptable one. As a consequence, from one experiment to another there are some variations in performance for the convergence values. Figure 3 illustrates in Whiskers diagrams the distribution of performance for the same experimental data that are depicted in Figure 2.

The three strategies show different performance and convergences to an optimal solution. Contrasting with *DS1*, *DS2* takes benefit from the prior experiences not only in relation with the entire problem type but also from the intermediate solutions to model the sub-problems in which the initial problem has been decomposed. In addition, *DS3* increases the granularity by adding the opportunity to decompose the problem's objective. As a consequence, the figures show the influence of problem type decomposition on the GDP model's performance.

As may have been expected, *DS3* shows the best performance results and a low inconsistency among the identified solution for a GDP model. But, *DS2* converge faster to an optimal solution and a lower inconsistency among the feasible solutions (the deviation of the average solution from those with maximal and minimal performances). This can be explained by the additional constraints of having stable objectives during the modelling process.

4.2 The cognitive complexity associated with each modelling strategy

The auto-organization of relations between TLs (i.e., the performance update after successive evaluations) entails a decrease of freedom due to the emergence of contextual constraints that reduce the probability to select some TLs (i.e., the preference for the available TL as defined in Equation (2)). For a problem type, the degree of freedom corresponds to the probabilistic distribution of preferences for the selection alternatives that is equivalent with the Shannon normalized entropy [40, 43]. The Shannon normalized entropy for the selection of a TL is given by:

$$E(p_{\mathrm{jk}}) = -\sum_{k=1}^{m} p_{\mathrm{jk}} \cdot \ln(p_{\mathrm{jk}})/\ln(m) \tag{3}$$

where p_{jk} - represents the preference, the selection probability of the TL k from the TL j; k - is the index for the TLs connected from the node j (in fact, all the m TLs available in the problem space).

When the recorded performance is equal for all the available modelling alternatives, the user is considering the entire problem space when he selects a feasible TL (the probabilities from Equation (3) being equally distributed entail an entropy equal with 1). Contrary, when the recorded *performance* favours a single alternative, the user will have no freedom in the selection of the best TL (all the probabilities from formula 2 being 0 except the best alternative that is 1, entails an entropy equal with 0). Thus, the entropy associated with TL's selection is a measure of cognitive complexity for modelling the GDP. Moreover, it is a local metrics that can be computed for each TL's selection activity for modelling the GDP.

Figure 4 shows the cognitive complexity associated with GDP modelling for the design strategies: *DS1*, *DS2* and *DS3*. The data are obtained for the same experimental settings as introduced in the beginning of this section. Because this measure is computed on the basis of the local data for each selection action (the performances available on the edges from the current TL), the figure corresponds to the average of entropies for all the TL selection actions needed to complete the GDP model (5 successive TLs, for this case).

Similar to the distribution of GDP models' performance, comparing with *DS1* the normalized entropy for *DS2* and *DS3* converge faster to 0. For *DS3* the gradual increase in the complexity of the representations for a GDP model in the conceptual problem space (through the definition of different sub-plans on different semantic levels) makes the design of the GDP process more manageable in the sense that it reduces the overall complexity of representing the GDP model. Nevertheless to reduce the cognitive complexity for modelling the GDP, the users should be supported in using this representation for guidance and action in the conceptual problem space. This entails the reflection of the abstract representations into the conceptual environment through an increased semantic complexity (by adding new problem types, codified with distinct IDs in our implementation) which facilitates the synergy among partially overlapping GDP. Only in this way the users may move from one sub-plan to another or from low level detailing with TLs to high level of plan and to be fully supported by the stigmergic coordination mechanisms at the same time.

5 Summary and conclusions

Stigmergic mechanisms are widely employed in human society, from the classical examples of large-scale complex adaptive systems such as stock markets, supply logistics and cultural memes to the more

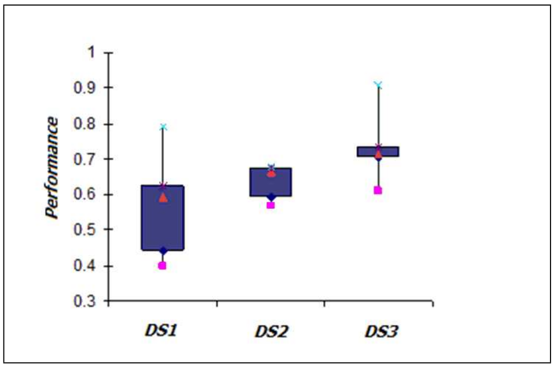


Figure 4: The normalized entropy of the GDP modelling for the three design strategies

recent range of collaborative working environments such as wiki, Google, social networking and open-source software.

Continuing our work in achieving inter-paradigmatic synergy between symbolic and sub-symbolic reasoning for large-scale complex systems [44, 45], the paper highlighted the computational power of stigmergic coordination mechanisms to build collaborative support systems for complex decisions such as facilitation of GDP in e-meetings. Unalike conventional approaches that minimize the cognitive complexity by incorporating the facilitation knowledge into the system, stigmergic coordination mechanisms minimize it by offering emergent functionalities that are made up not only by the user's actions (through the selection of the right modelling components) but also by the environment itself (through the possibility to structure it in terms of high-level cognitive artefacts such as plans or sub-plans). This was demonstrated by implementing and testing in a socio-simulation experiment an envisioned collaborative software tool that act as a stigmergic environment for modelling the GDP. The results show superior results when the users are supported to gradually increase the complexity of the representations for a GDP model which is reflect on its turn in the relational complexity of the conceptual problem space.

Bibliography

[1] P.P. Grassé. La theorie de la stigmergie: essai d'interpretation du comportement des termites constructeurs, *Insectes Sociaux*, 6: 41-81, 1953.

[2] P. Valckenaers, H. Van Brussel, Hadeli, O. Bochmann, B. Saint Germain, C. Zamfirescu On the design of emergent systems: an investigation of integration and interoperability issues. *Engineering Applications of Artificial Intelligence*, 16(4): 377-393, 2003.

[3] H.V.D. Parunak, S.A. Brueckner. The Cognitive Aptitude of Swarming Agents http://www.newvectors.net/staff/parunakv/CASA.pdf, 2009.

[4] M.A. Rodriguez. Faith in the Algorithm, Part 1: Beyond the Turing Test, http://faithinthealgorithm.net/, 2009.

[5] S.C. Negulescu, C.V. Kifor, C. Oprean. Ant Colony Solving Multiple Constraints Problem: Vehicle Route Allocation, *International Journal of Computers, Communications and Control*, 3(4): 366-373, 2008.

[6] H.V.D. Parunak. A Survey of Environments and Mechanisms for Human-Human Stigmergy, *Lecture Notes on Artificial Intelligence*, 3830: 163-186, Springer, 2006.

[7] A. Ricci, A. Omicini, M. Viroli, L. Gardelli, E. Oliva. Cognitive Stigmergy: Towards a Framework Based on Agents and Artifacts, *Lecture Notes in Computer Science*, 4389: 124-140, Springer, 2007.

[8] F. Heylighen, J. Bollen. The World-Wide Web as a Super-Brain: from metaphor to model, *Proceedings of Cybernetics and Systems '96*, Austrian Society for Cybernetics, 917-922, 1996.

[9] G.J. de Vreede, J.A. Boonstra, F. Niederman. What Is Effective GSS Facilitation? A Qualitative Inquiry Into Participants' Perceptions, *Proceedings of the 35th Hawaiian Internal Conference on System Sciences*, IEEE Computer Society Press, 2002.

[10] G.L. Kolfschoten, W. Veen. Tool Support for GSS Session Design, *Proceedings of the 38th Annual Hawaii International Conference on System Sciences*, IEEE Computer Society Press, 2005.

[11] A.V. Lopez, Q. Booker, N.S. Shkarayeva, R.O. Briggs, J.F. Nunamaker Jr.. Embedding Facilitation in Group Support Systems to Manage Distributed Group Behavior, *Proceedings of the 35th Hawaii International Conference on System Sciences*, IEEE Computer Society Press, 2002.

[12] R.A. Brooks. Intelligence Without Representation, *Artificial Intelligence Journal*, 47: 139-159, 1991.

[13] E. Hutchins. *Cognition in the wild. Cambridge*, MIT Press, 1995.

[14] J. Rambusch, T. Susi, T. Ziemke. Artefacts as Mediators of Distributed Social Cognition: A case study. *Proceedings of the 26th Annual Conference of the Cognitive Science Society*, 1113-1118, Mahwah, NJ: Erlbaum, 2004.

[15] T. Susi, T. Ziemke. Social Cognition, Artefacts, and Stigmergy: A Comparative Analysis of Theoretical Frameworks for the Understanding of Artefact-mediated Collaborative Activitym, *Cognitive Systems Research*, 2(4): 273-290, 2001.

[16] M.A. Elliott. *Stigmergic Collaboration: A Theoretical Framework for Mass Collaboration*, PhD Thesis, Centre for Ideas, Victorian College of the Arts, The University of Melbourne, 2007.

[17] M.A. Elliott. Stigmergic Collaboration: The Evolution of Group Work,*M/C Journal*, 9(2), http://journal.media-culture.org.au/0605/03-elliott.php, 2006.

[18] H.V.D. Parunak, T.C. Belding, R. Bisson, S.A. Brueckner, E. Downs, R. Hilscher. Stigmergic reasoning over hierarchical task networks. *Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems*, 1195-1196, 2009.

[19] I. Dzitac, B.E. Barbat, Artificial Intelligence + Distributed Systems = Agents, *International Journal of Computers, Communications and Control*, 4(1):17-26, 2009.

[20] F. Heylighen. Accelerating Socio-Technological Evolution: from ephemeralization and stigmergy to the global brain. in: *Globalization as an Evolutionary Process: Modeling Global Change*, edited by George Modelski, Tessaleno Devezas, and William Thompson, London: Routledge, 286-335, 2007.

[21] G. DeSanctis, B. Gallupe. A Foundation for the study of Group Decision Support Systems, *Management Science*, 589-609, 1987.

[22] F.G. Filip, Decision support and control for large-scale complex systems, *Annual Reviews in Control*, 32(1): 61-70, 2008.

[23] F. Niederman, C.M. Beise, P.M. Beranek. Issues and Concerns about Computer-Supported Meetings: The Facilitator's Perspective, *MIS Quarterly*, 20(1): 1-22, 1996.

[24] R.O. Briggs, G.J. de Vreede, J.F. Nunamaker, Jr.. Collaboration Engineering with ThinkLets to Pursue Sustained Success with Group Support Systems, *Journal of Management Information Systems*, 19(4): 31-63, 2003.

[25] J.H. Helquist, J. Kruse, M. Adkins. Developing large scale participant-driven group support systems: An approach to facilitating large groups, *Proceedings of the First HICSS Symposium on Field and Case Studies of Collaboration*, IEEE Computer Society Press, Los Alamitos, CA, 2006.

[26] F.G. Filip. System analysis and expert systems techniques for operative decision making. *J. of Syst. Anal. Model. Simul.* 8(2): 296-404, 1990.

[27] C.B. Zamfirescu. An Agent-Oriented Approach for Supporting Self-Facilitation in Group Decisions, *Studies in Informatics and Control*, 12(2): 137-148, 2003.

[28] R.J. Harder, J.M. Keeter, B.W. Woodcock, J.W. Ferguson, F.W. Wills. Insights in Implementing Collaboration Engineering, *Proceedings of the* 38^{th} *Hawaiian Internal Conference on System Sciences*, IEEE Computer Society Press, 2005.

[29] M. de Hengst, E. de Kar, J. Appelman. Designing Mobile Information Services: User Requirements Elicitation with GSS Design and Application of a Repeatable Process, *Proceedings of the* 37^{th} *Hawaiian Internal Conference on System Sciences*, IEEE Computer Society Press, 2004.

[30] J.H. Appelman, J. Driel. Crisis-response in the Port of Rotterdam: can we do without a facilitator in distributed settings? *Proceedings of the* 38^{th} *Hawaiian Internal Conference on System Sciences*, IEEE Computer Society Press, 2005.

[31] B. Enserink. Creating a Scenariologic - Design and Application of a Repeatable Methodology, *Proceedings of the* 36^{th} *Hawaiian Internal Conference on System Sciences*, IEEE Computer Society Press, 2003.

[32] G.J. de Vreede, R.O. Briggs. Collaboration Engineering: Designing Repeatable Processes for High-Value Collaborative Tasks. *Proceedings of the* 38^{th} *Hawaiian Internal Conference on System Sciences*, IEEE Computer Society Press, 2005.

[33] L.R. Christensen. The Logic of Practices of Stigmergy: Representational Artifacts in Architectural Design, *Proceedings of the 2008 ACM Conference on Computer Supported Cooperative Work*, ACM, New York, 559-568, 2008.

[34] G.J. de Vreede, R.O. Briggs, G.L. Kolfschoten. ThinkLets: A Pattern Language for Facilitated and Practitioner-Guided Collaboration Processes, *International Journal of Computer Applications in Technology*, 25: 140-154, 2006.

[35] R.O. Briggs, B.A. Reinig, G.J. de Vreede. Meeting Satisfaction for technology-Supported Groups: An Empirical Validation of a Goal-Attainment Model, *Small Group Research*, 37, 2006.

[36] D. Luce. *Individual Choice Behaviour*, Wesley, New York., 1959.

[37] K.M. Carley. On Generating Hypotheses Using Computer Simulations. *Proceedings of the 1999 International Symposium on Command and Control Research and Technology. Evidence Based Research*, Vienna, 1999.

[38] U. Wilensky. *NetLogo*, http://ccl.northwestern.edu/netlogo/, Center for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston, 1999.

[39] C.B. Zamfirescu. *Anthropocentric Group Decision Support Systems*. PhD Thesis, Politehnica University of Bucharest, 2006.

[40] S. Guerin, D. Kunkle. Emergence of constraint in self-organizing systems. *Nonlinear Dynamics, Psychology, and Life Sciences*, 8(2): 131-146, 2004.

[41] B.E. Barbat, S.C. Negulescu, C.B. Zamfirescu. Human-Driven Stigmergic Control. Moving the Threshold. *Proceedings of the 17^{th} IMACS World Congress - Scientific Computation, Applied Mathematics and Simulation*, Paris, 2005.

[42] V.H.D. Parunak, S.A. Brueckner, J.A. Sauter, R. Matthews. Global convergence of local agent Behaviors, *Proceedings of the 4^{th} International Joint Conferenece on Autonomouse Agents and Multi.Agent Systems (AAMAS)*, Utrecht, Netherlands, 2005.

[43] V.H.D. Parunak, S. Brueckner. Entropy and Self-Organization inMulti-Agent Systems. *Proceedings of the Fifth International Conference on Autonomous Agents*, Montreal, Canada, 124-130, 2001.

[44] B.E. Bărbat, C.B. Zamfirescu, S.C. Negulescu. The Best from Ants and Humans: Synergy in Agent-Based Systems. *Studies in Informatics and Control Journal*, 13(1): 47-59, 2004.

[45] F.G. Filip, K. Leiviskä. Large-Scale Complex Systems. *Handbook of Automation* (S. Nof, editor), 619-638, Springer, 2009.

Constantin-Bala Zamfirescu is currently associate professor (lecturer) at "Lucian Blaga" University of Sibiu, Romania. He took the PhD from "Politehnica" University Bucharest in Automation and Control field. His research interests include the employment of agent-oriented engineering paradigm in group decision support and manufacturing control systems. He is a member of IFAC Technical Committees 5.4, EU COST Action 298 and 2^{nd} EU NoE for the Advancement of Artificial Cognitive Systems, Interaction and Robotics.

Florin Gheorghe Filip member of the Romanian Academy (elected in 1991), received his M. Sc and Ph. D. in Control Engineering from the Technical University "Politehnica" Bucharest in 1970 and 1982, respectively. He has been with the National R & D Institute for Informatics since 1970 and was the managing director of ICI from 1991 to 1997. He was elected in 2000 (and re-elected in 2002 and 2006) as a vice-president of the Romanian Academy. He has been the chair of the IFAC Technical Committees 5.4 (2002-2008). His main scienti?c interests include: control and optimization of large-scale systems, decision support systems, technology management, and models for knowledge-information society. He is author/co-author of six books, editor of nineteen volumes and the author/co-author of over two hundred articles published in contributed volumes and scientific journals.

Author index

Allgayer R., 52
Antonelli D., 8

Chen W.L., 71
Chiabert P., 8
Cohen S., 20

De Haan U., 20
Dori D., 20

Engler J., 42

Ferreira A., 52
Filip F.G., 125
Freitas E., 52

Heimfarth T., 52
Huang C.Y., 71

Ko H.S., 6, 91
Kusiak A., 42

Larsson T., 52

Nof S.Y., 6, 71, 91

PereiraC. E., 52

Wachs J.P., 106
Wagner, 52

Yang T.T., 71

Zamfirescu C.B., 125