



**HAL**  
open science

## Improvement of CSCW software implementation in NPD: the CAM mechanism for a better adoption by users

Tomas Restrepo, Natalia Arbelaez, Dominique Millet, Thierry Gidel

### ► To cite this version:

Tomas Restrepo, Natalia Arbelaez, Dominique Millet, Thierry Gidel. Improvement of CSCW software implementation in NPD: the CAM mechanism for a better adoption by users. *International Journal of Learning and Change*, 2010, 4, pp.281 - 299. hal-01772748

**HAL Id: hal-01772748**

**<https://hal.science/hal-01772748v1>**

Submitted on 20 Apr 2018

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

---

## **Improvement of CSCW software implementation in NPD: the CAM mechanism for a better adoption by users**

---

### **Tomas Restrepo\***

Ecole Nationale Supérieure d'Arts et Métiers,  
151 Bd de l'Hôpital, 75013 Paris, France  
Fax: +33-1-44-24-63-59  
E-mail: tomas.restrepo@gmail.com  
\*Corresponding author

### **Natalia Arbelaez**

Université de Technologie de Compiègne,  
BP 60319, 60203 Compiègne, France  
Fax: +33-3-44-23-52-13  
E-mail: arbelaez.natalia@gmail.com

### **Dominique Millet**

'Ecodesign of Systems' Laboratory/  
E.R.T. 'Eco-conception de Systèmes',  
SUPMECA – Toulon, Quartier Mayol, Maison des Technologies,  
83000 Toulon, France  
Fax: +33-4-94-03-88-04  
E-mail: dominique.millet@supmeca.fr

### **Thierry Gidel**

Université de Technologie de Compiègne,  
BP 60319, 60203 Compiègne, France  
Fax: +33-3-44-23-52-13  
E-mail: thierry.gidel@utc.fr

**Abstract:** Cooperation between disseminated actors is a key factor in improving new product development (NPD) performance. In the last years, numerous CSCW software applications have been introduced in the industry to support NPD with a low success rate. This is partly due to the limited insight of the organisational and human factors influencing user motivation during the first stages of CSCW software implementation. This research suggests that users adopt more rapidly those features of CSCW software with whom they are more familiar and that the adoption of other features of the application is reinforced by the various links connecting them with the preferred feature. We propose a CSCW integration protocol called the 'catching module mechanism' (CAMM) which exploits user's inclination towards a particular feature of a CSCW application by training them on that same feature. The CAMM was tested in the NPD context of Saint-Gobain Glass.

**Keywords:** CSCW software implementation; CSCW adoption; project management; new product development; NPD; learning; change; CAM mechanism.

**Reference** to this paper should be made as follows: Restrepo, T., Arbelaez, N., Millet, D. and Gidel, T. (2010) 'Improvement of CSCW software implementation in NPD: the CAM mechanism for a better adoption by users', *Int. J. Learning and Change*, Vol. 4, No. 4, pp.281–299.

**Biographical notes:** Tomas Restrepo is the Vice President of New Business Development at Cementos Argos. He received his PhD in Industrial Engineering from ENSAM School in Paris, France from the 'Product Design and Innovation Laboratory'/'Laboratoire Conception de Produits et Innovation'. He worked for Saint-Gobain Glass as a Development Engineer where he conducted a three-year research project on the integration of new product development CSCW applications. He has published a book chapter and numerous articles in international congresses on product design and innovation.

Natalia Arbelaez is the Director of Innovation of Servicios Nacional de Chocolates. She received her Masters in Technology and Innovation Management, and her MS in Human Science and Cognition and Cooperation Technologies from Université de Technologie de Compiègne (UTC). She conducted a research study at Saint-Gobain Glass where she participated in the implementation of a CSCW application as part of the COSTECH laboratory of the UTC.

Dominique Millet teaches design and eco-design methods at the Engineers School SUPMECA in Toulon. He is the Director of the EOS lab (ecodesign and optimisation of systems) research team. He has undertaken numerous research programs on design and integration of design methods within organisations, particularly in the transport sector and has published numerous scientific papers on this subject. He is a member of the French Organisation for Standardisation (AFNOR) and has been a co-author of the ISO 14062 "Integrating environmental aspect into product design and development". He is the main author of the book *Integration of the Environment in the Design Process; Companies & Sustainable Development/Intégration de l'environnement en conception; entreprises et développement durable* published in 2003 by Hermes Science Publishing.

Thierry Gidel is an Assistant Professor in 'Université de Technologie de Compiègne. He received his PhD in Industrial Engineering from 'École Nationale Supérieure d'Arts et Métiers'. Before joining 'Université de Technologie de Compiègne', he had worked as a Project Manager in France, Asia, and UK. His research interests include robust design, multi-project management and innovation process. He is the author of several papers and books on project risk management and project management.

---

## 1 Introduction

New product development (NPD) in multinational enterprises strongly relies on cooperation between disseminated actors. NPD teams are usually composed of highly specialised professionals who share a common interest in a particular technology. Every NPD project is an intensive learning process where the different experiences

and competencies of the team members contribute to generate a new object. This context is what Wenger (1998) describes as a community of practice where interconnected actors of an organisation constantly learn in a cooperative environment and gradually build a body of knowledge that becomes a valuable asset for the company. Nonaka states that for an organisation to create knowledge it requires an intensive and laborious interaction among its members (Nonaka, 1995). Nonaka highlights the particular need for this intensive cooperation among NPD actors in order to produce meaningful innovations. These actors, often located in different time zones, frequently use CSCW software applications to communicate, coordinate their activities, and share their information and resources (Hubert et al., 1995; Boujut and Blanco, 2003). Many CSCW software solutions are commercially available supporting specific NPD activities such as cooperative drawing or project time management.

Although the technical advantages of most of these solutions have been demonstrated (Briggs, 2006), numerous authors argue that these applications often fail to meet the objectives fixed by the enterprises (Grudin, 1994; Orlikowski, 2000). Only 29% of software developments (covering the category of CSCW software) are successful (Standish Group, 2005). The effective use of CSCW software by the critical mass of users would seem to be much more difficult to obtain than just deploying an off-the-shelf application. The value for the enterprise of an information and communication technology (ICT) investment is determined in part by its integration with the organisational processes (Ozturk et al., 2006; Prasad et al., 1993; Hares and Royle, 1994). The majority of the studies on CSCW software applications in the field of concurrent engineering and NPD focus on the analysis of cooperative relations and the technological aspects of the proposed solutions (Orlikowski, 2000; D'souza and Greenstein, 2003). The process of CSCW software introduction in NPD is a subject that has been mistreated in the literature. This process needs to be detailed for further insight on how to stimulate the quality and frequency of use of these applications, which determine CSCW software integration within NPD. In the next paragraph, we will call the "CSCW software integrator" the person(s) in the enterprise whose objective is to successfully introduce the application in the organisation.

As stated by Grudin (1994), the usual practise in the industrial context is to implement a new CSCW application and provide a general training in all the features of the system to a comprehensive population of target users. This general training is expected to motivate users into adopting the application and then the group dynamics should help its intended cooperative use. However, with this general training approach the success rate of CSCW adoption is still limited.

This research explores a novel approach to CSCW adoption based on fulfilling the user's curiosity for a specific feature of the application. The article proposes a mechanism to catch the attention of users into a specific functional module of the CSCW software and obtain the user engagement to use the application. This mechanism is based on the motivation to the use of CSCW software through the targeted training on this specific module based on their individual pre-acquired skills. This mechanism, called 'catching module mechanism' (CMM) proposes a series of methodological tools to establish a relationship between the CSCW software integrator and the user through targeted training. As a result of this relationship the user will progressively adopt the CSCW

software. The proposed mechanism could help the CSCW software integrator to control the adoption process favouring the effective use of the application by the critical mass of users.

First, we will present a literature review of CSCW software design and adoption. Then the structure of the proposed mechanism will be described. An experiment carried out at Saint-Gobain Glass will illustrate the implementation of this mechanism. Finally, we will discuss the results of our experiment.

## **2 Literature review**

The term CSCW software has been used since the early 1990's to describe the software applications designed to support a group's work (Bate and Travell, 1994). The NPD activity is an interesting development field for these applications because of the complexity of communication relations between actors (Hatchuel, 1996), the high level of geographical dissemination of project teams (Gutierrez and Friedman, 2005) and the interdependencies between actors (Rabardel and Bourmaud, 2003). Many CSCW software applications are commercially available. Among the most common features are project time management, electronic conferencing, cooperative drawing, knowledge management, workflow, file sharing and cooperative scheduling. The technical interest of these applications to improve the performance of NPD projects has been demonstrated by numerous studies showing gains in project team productivity, reducing design reprocessing and yielding shorter time-to-market cycles at lower costs (D'souza and Greenstein, 2003; Chiu, 2002; Eynard et al., 2006). However, several authors argue that a large proportion of both custom-made and off-the-shelf solutions fail to achieve the objectives fixed by the enterprise in terms of frequency and quality of use (Grudin, 1994; Orlikowski, 2000; Sun and Zhang, 2006).

Orlikowski argues that only a reduced fraction of the features included in off-the-shelf CSCW software fit the needs of NPD teams, and for this reason, the potential benefits of these applications are rarely obtained (Orlikowski, 2000). On the other hand, the statistics of the Standish Group show that only 29% of the custom-made software developments (covering CSCW software solutions) succeed, 56% of the projects lead to unsatisfactory results and 15% of the projects are stopped before completion (Standish Group, 2005). Grudin claims that this high failure rate is due to specification errors and poor introduction processes (Grudin, 1994). Technical aspects are rarely cited as a cause of failure (Orlikowski, 1991; Griffith, 1996).

### *2.1 The errors of the traditional integration processes*

CSCW software applications were born from the combination of information systems (IS) and PC applications and, in fact, these systems experience some of the advantages and limitations of these two technologies (Grudin, 1994). CSCW software are far more delicate than IS and PC applications because they rely on both the organisational support needed for IS success and on the ergonomic implications that determine the use of PC applications. Often, CSCW software fails because they are seen as either of these two technologies and not as a solution supporting both everyday technical tasks and cooperation processes.

For instance, the use of group work support tools should not be coerced (as practised in IS) because instead of 'having the process done' it can lead to 'defensive routines' (Argyris and Schön, 1996). An example of defensive routine in CSCW software is the mediocre use of a project management application resulting in a loss of time for project teams more than a productive activity (Orlikowski, 2000).

Another common mistake is to implement generic training campaigns (as practised for PC applications). NPD teams use collective tools for their specific cooperation requirements and as a result only a limited number of CSCW software features are needed (Kensing et al., 1998). For example, a project team that seeks the resource management features of a full project management CSCW software should only be interested in this specific feature. In this case, global training campaigns are not an appropriate solution because of the superficial knowledge of the whole application given to the users and the difficulty to target training as a function of their needs (Woodhall, 1987; Moonen, 1997).

## *2.2 Motivation to use of CSCW software applications*

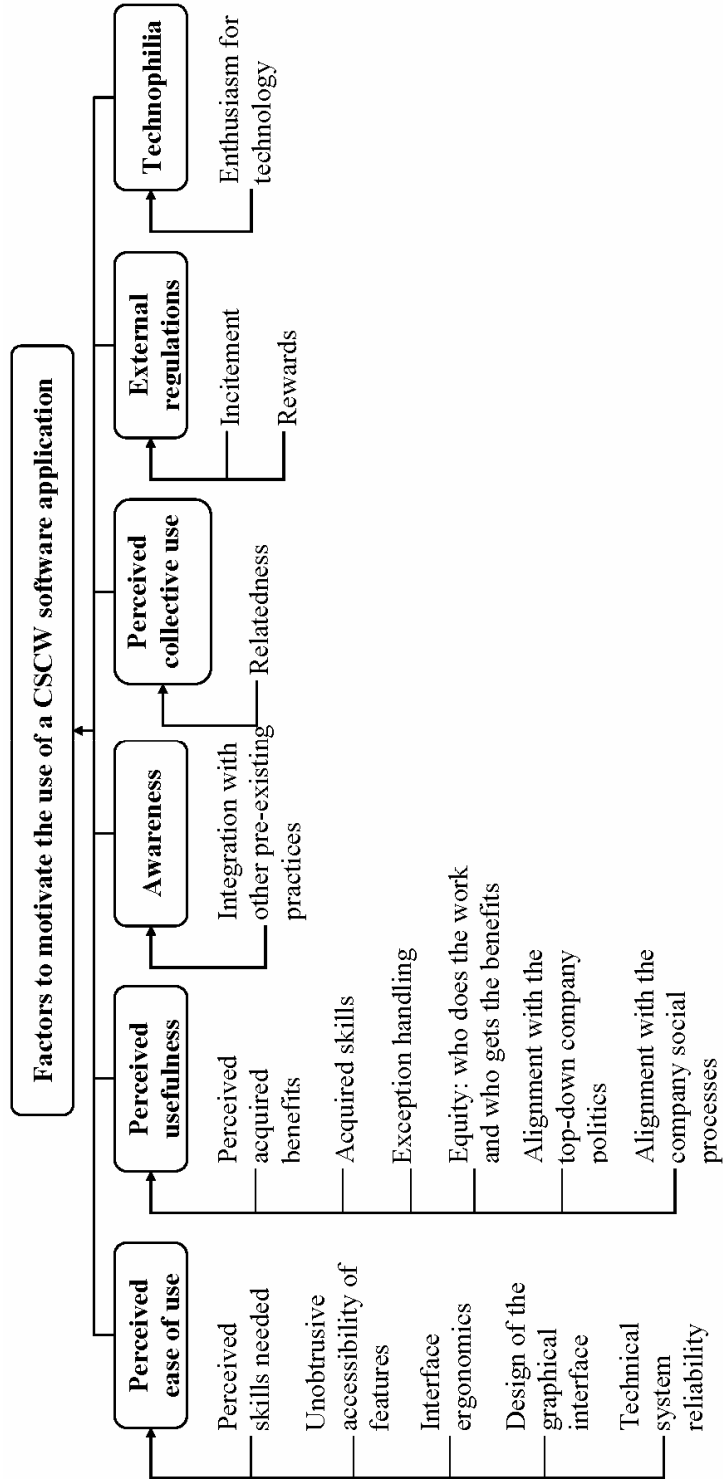
An important issue of ICT implementation is to understand the factors that determine the motivation of users to adopt the application. This is a key question to identify the levers of the NPD CSCW software integrator. Griffith established that not only technical aspects play a role in the adoption of a CSCW application. Other factors should be considered in the process of implementation such as the needs of potential users and their perception regarding the new application. It is important to bring together usefulness and other mechanisms to make attractive the use of the CSCW application (Griffith, 1996). This means that motivation is a variable influencing technology adoption: many factors outside the interaction with the system, such as intrinsic motivation, emotion and perceived enjoyment are important (Venkatesh, 2000). This motivational construct of technology adoption suggests that a thorough understanding of the functionalities of a new CSCW application would increase the chances of adoption by a potential user.

Based on the theories of motivation to work (Igalens and Roussel, 1999), on the technology adoption model (TAM) (Davis, 1989) and on the theory of user acceptance of IT (Venkatesh et al., 2003), a motivational model to the use of CSCW software was built. The model is composed of six main factors both based in technical aspects addressing the design of the application, and in individual and social aspects:

- 1 perceived ease of use (usability)
- 2 perceived usefulness
- 3 awareness
- 4 perceived collective use
- 5 external regulations
- 6 technophilia.

These factors are composed of several variables respectively. Our model is proposed for a non-coerced CSCW software implementation process. The model is presented in Figure 1.

**Figure 1** Model of the motivations to CSCW software adoption



This model shows that the decision of a potential user to experiment a CSCW software and to learn to use it is the result of a complex perceptive process where the user evaluates the personal cost and the benefits of using the application. A CSCW software integrator can influence the user's perception of an existing application through training. Training can influence the user's perception of: the required skills to use the application (Ryan and Deci, 2000), the obtained benefits (Igalens and Roussel, 1999), the acquired skills (Tannenbaum and Yukl, 1992), the integration with other tools and practices (Restrepo et al., 2006a, 2006b; Millet et al., 2005) and the relatedness to a group (Tannenbaum and Yukl, 1992). Some of the remaining motivational factors may be influenced during the specification and development of the application (Grudin, 1994; Griffith, 1996). The external regulations can be influenced by the management through rewards and feedback (eucity or noxity) (Argyris and Schön, 1996; Hill, 1985). Finally, the personal enthusiasm to the use of technologies, called technophilia, is an endogenous factor on which the CSCW software integrator may have no influence (Davis, 1989). The adopter of a new CSCW software application will go through the following phases before fully using the application: awareness, interest, evaluation, trial, and adoption (Rogers, 2003).

In this article we will focus on the influence that a targeted training has on the adoption of an existing CSCW software application.

### *2.3 Targeted training as mechanism to motivate the use of NPD CSCW applications*

Targeted training plays an important role when motivation to use is needed. Anticipation is required for motivation, and training is a way to give users the necessary concepts to anticipate the usefulness of the CSCW software. As stated by Stewart "a cognitive agent would be engaged in an activity only when he is able to anticipate the consequences of his actions" (Stewart, 2006).

Also, because targeted training is a method based in "the relationship between the trainer and the person being trained" (Van Wart et al., 1993), the trainer becomes a 'human mediator' (Okamura et al., 1994) and this 'mediator' is proposed as a key factor in CSCW implementation as he helps to keep interest and motivation (Christiansen and Kanstrup, 2004) and also because he influences in a direct way not only the interactions between users and CSCW software but also the perceptions of usefulness and usability (Okamura et al., 1994). By using a targeted training process, the CSCW integrator can accelerate the process of awareness generation, building interest, guiding evaluation and trial, bringing the user to the doors of adoption (Rogers, 2003).

### *2.4 The pre-acquired skills of users*

A CSCW software application could be divided into several functional modules. Each module assembles various features around a common skill (e.g., the chat module of an electronic conferencing of a CSCW application).

As Rosenberg argued, the pre-acquired skills and experiences of subjects affect the dynamics of adoption of a new technology (Rosenberg, 1972). Each potential user of a CSCW software will have different levels of pre-acquired skills on each module of the application as a function of their previous experiences and knowledge. If we compare the



real interface to the image that a person has of a particular module of a CSCW application before having any contact with the application, a skill gap can be established. The magnitude of this gap between the user's image and the real application will determine the level of learning and unlearning needed for the user to understand and use each module of the CSCW software.

The successful adoption and integration of CSCW software relies more than it is commonly accounted by IT developers on users motivational factors based on their particular perceptions of CSCW applications.

Our research question is formulated as follows: is there a method to boost users' adoption of a new CSCW application by leveraging his or her motivational drivers?

### **3 The catching module mechanism**

The learning process through which a potential user adopts the CSCW software is composed of an awareness phase followed by a decision of engagement and a period of activation where the user accepts or rejects the application (Jones et al., 1994). This is a complex process where the user analyses the interest of the CSCW software in comparison to the cost of adopting it. During this period the user is particularly attentive to the encouraging or discouraging aspects of using the application. These aspects vary from one user to another as a function of his expectations and experiences. The organisational integration of CSCW software is determined by the acceptance by the pilot users. If the application fails in the pilot phase, the success during the global deployment is highly unlikely to occur (Bowker et al., 1997).

We suppose that the targeted training of pilot users on a specific functional module will allow the CSCW software integrator to effectively communicate the benefits of using the tool and thus influence the users' perception of the application. This global process, called CAMM, consists in identifying the functional modules of a CSCW application, detecting the minimum skill gap of potential users and implementing a targeted training. The CAMM constitutes the instrument through which a CSCW software integrator will be able to engage the users into using the application.

#### *3.1 The links between modules*

CSCW software as any other multifunctional application is divided in functional modules. These modules are usually presented as drop-down menus or icons on the graphic user interface (e.g., the resource module of a project management CSCW application) (Norman, 1999). These modules are interconnected by interface or conceptual links. As stated by Carroll, these links produce a sense of awareness of the existence of the other modules of the application (i.e., the hyperlink) (Carroll et al., 2006). We identify four types of links between modules: edition, navigation, creation and methodology.

##### *3.1.1 The edition link*

This link allows a user to edit an object of one module while using a different module. For example, in cooperative drawing software, a user can modify the annotation of an

entity (annotations module) by clicking on the annotation icon on the 3D viewing module.

### *3.1.2 The navigation link*

To access a particular module of the CSCW application the user must sometimes browse through a different module. For example, to find an e-mail address (contacts module of a messaging CSCW software), a user may need to browse through the e-mail module of the CSCW software to reach the search feature.

### *3.1.3 The creation link*

In most CSCW software applications, users can create an object of one module from a different module. For instance, when a user with administrative rights fills a regular workflow form (workflow update module) on a change management feature of a web-based project management CSCW software, he or she might as well add new activities to that particular workflow (workflow setup module) such as requiring input from other areas of the company to support that particular change requirement.

### *3.1.4 Methodological link*

This link describes the conceptual relation between two functional modules of a CSCW application. For example, the tasks module of a project management CSCW application is conceptually linked to the milestones module of the application as tasks are usually defined to progressively construct a project milestone. As a result, learning how to use milestones requires learning how to use tasks. All these links are present in most CSCW software applications and facilitate the movement of the user from one module to another.

## *3.2 The skill gap*

A user may have different levels of pre-acquired skills in the different modules of a CSCW application. Two extremes of the pre-acquired skills (the minimum and the maximum skill gaps of a user) can be distinguished. We suppose that a potential user will better understand the module where he has a minimum skill gap. The detailed understanding of a module leads to the anticipation of the benefits that the user may expect from it.

## *3.3 Targeted training and the CAMM*

The implementation of a CSCW software is usually supported by an official communication (e-mail, memos, etc.), a limited number of global demonstrations and a self learning support (e-learning, manuals, etc.). However, these actions are often not enough to hook users into using CSCW software. Users may need to fully understand the costs and benefits of using the application before engaging into a relation of frequent use of a CSCW software application (Grudin, 1994). Based on this statement as well as the motivational theories of technology adoption presented above, we suppose that targeted training of one user in the module where a minimum skill gap exists will minimise the

amount of learning and unlearning needed to understand that particular module. This deep understanding of a particular module of a CSCW application will help the user to anticipate the required skills for using the application, the benefits he can obtain, the skills he can acquire, the way the application is integrated with other applications and practices and the sense of relatedness to a group through the use of the CSCW software.

By the implementation of a targeted training technique, the CSCW software integrator will facilitate CSCW software adoption as a function of their particular pre-acquired skills.

### *3.4 The diffusion of adoption*

Moreover, we argue that the links between modules (edition, creation, navigation and methodology) will facilitate the movement of the user through the application. The movement through the different modules of the CSCW software facilitated or induced by these links will in term encourage the self learning of the remaining modules to which the user has not been trained. We call this phenomenon the diffusion of adoption. From our point of view, the links between modules constitute an important design parameter that must be taken into account during the specification and development phases of the CSCW software.

### *3.5 Measuring the adoption of a CSCW software*

One user that has adopted a module of CSCW software should not only use the application frequently, but he should use it properly. The quality of use of a CSCW application reveals the level of understanding of the application by a user and thus his motivation to use it. To measure the quality of use of the CSCW software a detailed observation protocol must be established for each module of the application. This will allow the follow-up of the CAMM and the evaluation of the level of diffusion of adoption through time.

## **4 The experiment at Saint-Gobain Glass**

To test our hypotheses we conducted an experimental project in Saint-Gobain Glass, a world leader in flat glass manufacturing with a strong technological innovation activity. In this experiment we have placed ourselves as the CSCW software integrator. We used the CAMM to implement a custom-made project time management CSCW application programmed by an IT development firm exclusively for the R&D project teams of Saint-Gobain Glass. The aim of the experiment was to demonstrate the influence of the targeted training of users in the initial CSCW adoption and the role of the links between modules in enabling the diffusion of adoption of a NPD CSCW software. For this experiment we worked with five representative project leaders as pilot users. First, the users' skill gap is estimated in order to select the module with the minimum skill gap on which the targeted training is targeted. Then, after three targeted training sessions on a particular module, using feedback techniques, the frequency of use of each module is measured. By evaluating the frequency of use of the modules where the user did not receive any training, the diffusion of adoption is measured.

#### 4.1 The scales of pre-acquired skills

The first step of our experiment consisted in identifying the different functional modules of the CSCW application. By analysing the basic project management skills required to use the project management CSCW application, four modules were identified:

- 1 Project data
- 2 Tasks
- 3 Milestones
- 4 Gantt diagram.

For each module all the fields that must be defined by the user (text, dates, menus, etc.) were extracted. Each field corresponds to a specific project management concept that the user must understand in order to correctly use it. For instance, a correct use of the 'task leader' drop-down menu of the tasks module, as established in this CSCW software, requires the understanding that every task must be the responsibility of a particular member of the project team. Based on the list of required skills a scale of five skill levels for each of the four modules of the CSCW software was built.

#### 4.2 Skill gap estimation

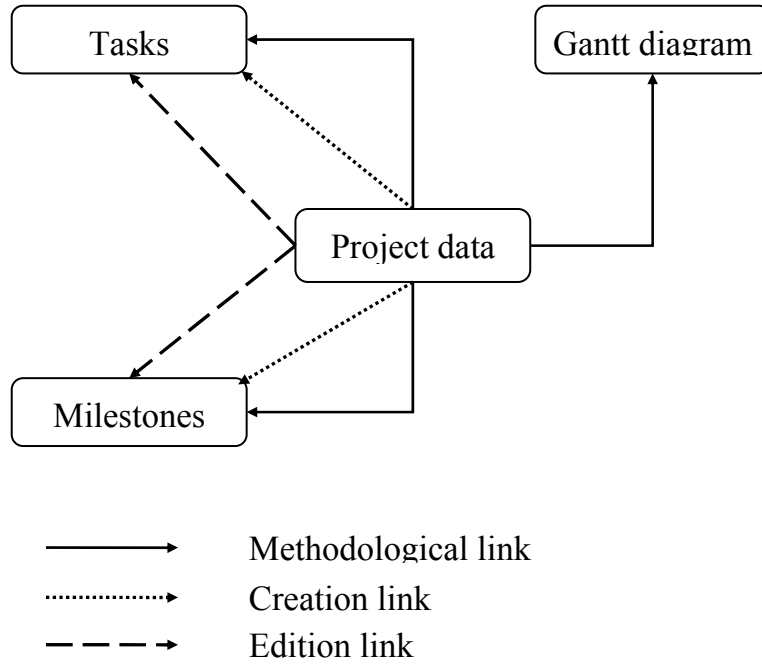
After defining the four modules, the second step consisted in the evaluation of the pre-acquired skills of each pilot user. For this, the five pilot users were interviewed. At that moment the pilot users did not know the specific CSCW software. In the interviews the users were asked to define the theoretical concepts behind each module of the CSCW software as detailed as possible. By evaluating the answers of the users, the level of understanding of each function of the CSCW software could be estimated, thus defining the skill gap. All the interviews were recorded. The responses were analysed by revealing the concepts cited by the users corresponding to the features of each module of the CSCW software. An evaluation grid was used to define the level of pre-acquired skills of each potential user for each module of the project management CSCW application. Table 1 shows the evaluation grid used to assess the pre-acquired skills of the potential users regarding the tasks functional module of the CSCW application.

**Table 1** Evaluation grid for the tasks functional module

<i>Theoretical objectives</i>	Understanding of the structure of a project. Familiarity with the concepts of tasks and sub tasks and its variables: priority, percentage of advance and ownership.
<i>Practical objectives</i>	The user should be able to create and structure tasks in his own project defining priorities and owners. The user should be able to update the percentage of advance and the work accomplished of each task and sub task.
<i>Skill level</i>	<i>Description</i>
1	The user understands that every task has a defined start date and duration
2	The user assign an owner to each task of a project
3	The user understands the concept of task subordination
4	The user understands the concept of percentage of advance and work accomplished of a task
5	The user distinguishes between tasks and sub tasks

From this evaluation we identified the CSCW software modules with minimum skill gap for each user. The results showed that one user should be trained on the tasks module, another user should be trained on project data module and the three remaining users should be trained on the Gantt module.

**Figure 2** The adoption diffusion links from the project data module



<i>Functionality</i>	<i>Link</i>	<i>Gantt diagram</i>	<i>Tasks</i>	<i>Milestones</i>
Project data	Creation		Once in the project functionality the user can create tasks	Once in the project functionality the user can create milestones
	Edition		In the project data functionality the user can modify the properties of a task.	In the project data functionality the user can modify the properties of a milestone
	Methodological	A project can be visualised and represented by a Gantt diagram	A project is formed by activities that need to be accomplished within a defined period of time	A project needs to have a marker that indicates the completion of a work package or phase

#### 4.3 *The links between modules*

Once the functional modules for the targeted training were selected for each user, the interface of the CSCW application was analysed in order to identify the existing edition, navigation, creation and methodology links between the four functional modules:

- In general, a minimum of two links existed between every two modules.
- Only one link existed between the Milestones module from the tasks module.

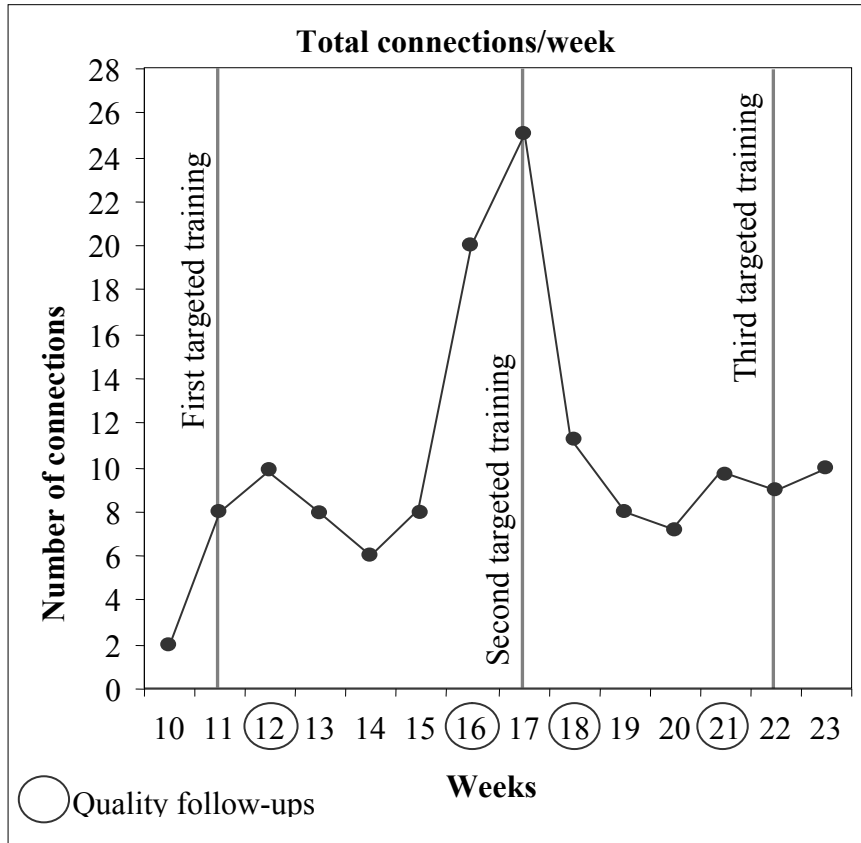
To correct this insufficiency of links between Milestones and tasks, the IT development firm was asked to add an edition link allowing the modification of Milestones from the tasks module. The adoption diffusion links from the project data module are presented in Figure 2.

#### 4.4 *The catching module protocol*

The experimental protocol implemented consisted of three individual targeted training sessions of two hours with one month of interval, in which the users were asked to autonomously use the CSCW software. The experiment lasted 14 weeks in total. In the first targeted training session a brief general presentation of the CSCW software was made to each user; then the control of the application was given to the user. At that time, the users were asked to create a new object of their module with the help of the integrator showing them how to fill the different fields on each page of the module. In the intermediate training the integrator gave a feedback to users on how they had used the application during the first period of autonomous use. For this a protocol to measure the quality of use was developed. The integrator gave a positive feedback (eucity) for the features that had been correctly used and a negative feedback (noxity) for the features that the user had not used properly in order to reinforce the high quality of use of the application (Huber, 1991). The integrator also answered to users' queries about the features that they had not used properly and he recorded their suggestions on how to improve the application. The suggestions were consolidated and submitted to the IT development firm who delivered a new version during the second period of autonomous use. In the third training period the integrator gave a feedback to users and emphasised on the features that had been less used.

#### 4.5 *The frequency of use*

Before starting the experimentation, the IT development firm was asked to include a connection log feature in the administration page of the CSCW application. The log file recorded the user login, the date and time of each connection to the application. Both the individual connections (logins per user per week) and the total connections (total logins of all users per week) were analysed. During the 14 weeks of the experiment with the pilot users a total of 142 logins were recorded. Every user connected to the application 2.02 times per week in average. This corresponds to the expected frequency of use of an asynchronous project management CSCW application. The evolution of the number of connections is presented in Figure 3.

**Figure 3** Total connections per week during the experiment

The most active period of the experiment was observed during the three weeks around the second targeted training sessions. During these weeks users were either updating their projects in the CSCW software to prepare for the training session (week 16) or actively using the CSCW software to update their projects with the new skills acquired in the training (weeks 17 and 18). From this analysis, the multiple targeted training sessions would appear to have a positive impact on the frequency of use of the CSCW software either because the users had the pressure of the upcoming revision or because of the curiosity generated by the recently acquired knowledge in the past training session.

#### 4.6 Evaluation of the adoption of the initial module and the diffusion of adoption

In order to measure the evolution of the levels of adoption of the initial module and the diffusion of adoption to the other modules during the two periods of autonomous use, four follow-up sessions of the quality of use were carried out. The quality follow-ups were positioned in the first week and the last week of the two periods of autonomous use.

A detailed protocol was developed to measure the quality of use for this specific CSCW software. For each module all the features on all the pages of the module were listed. One example of proper use was included for each feature. For instance, a correct

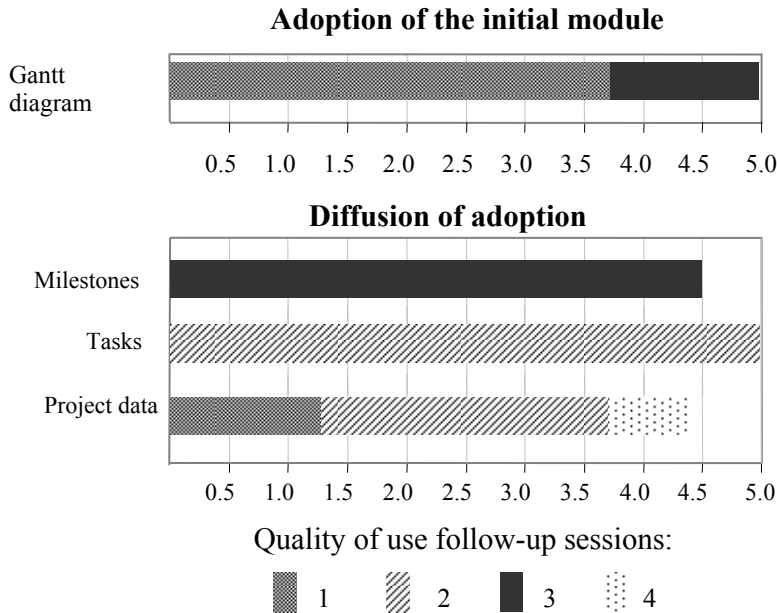
use of a feature modified via a drop-down menu consisted of a different value from the default value, i.e., a hyphen ‘-’.

To evaluate the quality of use of a module every object contained in the module (e.g., for the module tasks we examined all the tasks created by the user) has been analysed. For each feature (e.g., task name, task category, etc.) of each object (e.g., task 1, task 2, etc.) used correctly one point (*C*) was added. The variable *C* can be either one or zero depending if the feature was used correctly or incorrectly. Then, the average points per feature were calculated by adding all the points of a feature and dividing this value by the number of objects (*m*). The sum of the average points of all features was then normalised to five to obtain the value of Adoption to the module (*A*) by multiplying it by five and dividing it by the number of features of the module (*n*).

$$A_{Tasks} = \frac{\sum_{i=1}^n \left( \frac{\sum_{j=1}^m C_{j\_feature_i}}{m} \right) \times 5}{n} \tag{1}$$

By applying this protocol to each of the four modules of the CSCW application the different values of adoption of the initial module and the diffusion of adoption of other modules were obtained. An example of one pilot user is presented in Figure 4.

**Figure 4** Evaluation of adoption and diffusion of adoption



The user in Figure 4 showed a high level of adoption (3.75) of the Gantt diagram module (where he was trained) since the first follow-up in the first period of autonomous use. This user also presented a slight diffusion of adoption to the project data module during



this follow-up (1.25). At the end of the first period of use this user showed a high level of diffusion of adoption to the Tasks module and an additional diffusion of adoption to the project data module (2.5). At the end of the first week of the second period of use, this subject had a significant increase in the diffusion of adoption to the Gantt diagram module and a considerable diffusion of adoption to the Milestones module. In the last quality follow-up an increase of the diffusion of adoption to the project data module was noticed (0.5).

At the end of the experiment the five pilot users showed a high level of adoption of the initial module. The average adoption of the modules where the users were not trained was 3.65. Only one user did not use the Milestones module during the experiment. The most considerable increase (1.13) in the average adoption of all modules was after the second targeted training, during the third quality follow-up. This phenomenon confirms the positive impact of the multiple targeted training sessions on the level of adoption.

## 5 Conclusions and further research issues

CSCW software applications have been proposed as a useful solution to reduce the problems of cooperation within disseminated NPD project teams. A great number of applications exist today in the market. However, the statistics show that only 29% of these applications are used at the expected rate and quality of use. The common practice in CSCW software implementation is to provide a general training session to a large number of potential users. However, this training methodology does not guarantee an extensive adoption of the newly implemented application. A general training will only suggest the different features of the application thus it will hardly motivate users into adopting it. In contrast, our approach based on technology adoption theories shows that users will be more motivated to use those features of a CSCW application with which they are more familiar. By training users on their preferred feature of the new application, the CSCW integrator will be able to mobilise the perceived ease of use, the perceived usefulness, the awareness, the perceived collective use, the external regulations and the user's technophilia involved in CSCW adoption.

To assist practitioners confronted with CSCW implementation challenges we propose 'the CAMM', a methodology allowing CSCW integrators to control the adoption process by targeting the training efforts as a function of the pre-acquired skills of the pilot users. This study shows that

- 1 the users will adopt more easily the functional modules of the CSCW software where they have a higher level of pre-acquired skills because the necessary learning and unlearning will be minimised
- 2 it is more efficient to target the training of pilot users on the functional modules where they have the highest level of pre-acquired skills.

To test our hypothesis an experiment was conducted with five pilot R&D project leaders during the implementation of a project time management CSCW software in Saint-Gobain Glass. Our indicators show that the frequency and the quality of use were positively impacted by the CAMM resulting in the adoption of the CSCW software by the five pilot users. The proposed mechanism can also be used to help the implementation of other cooperative systems in concurrent engineering and NPD. The protocol may be

adapted to fit other organisational contexts (i.e., number of projects, number of actors, etc.). This novel methodology of CSCW implementation not only yields a higher level of adoption but it constitutes a more efficient way to train potential users.

Based on the experimental results of this study, we conclude that the multiple targeted training sessions had a positive impact on the frequency of use of the CSCW software either because the users had the pressure of the upcoming revision with the CSCW integrator or because of the curiosity generated by the recently acquired knowledge in the past training session.

The targeted training method used in the CAMM facilitates the adoption of one particular module of the CSCW software. The progressive adoption of the other modules of the CSCW application is achieved through the instinctive use of the links between modules created for this purpose.

Two new important CSCW software design parameters are revealed by this mechanism: the independence of the functional modules needed for the targeted training and the importance of the links between modules in the process of diffusion of adoption. The precise specification of these two parameters facilitates the implementation of the CAMM. First, this mechanism requires that the functional modules of a CSCW software are composed of an isolated group of features corresponding to a dissociable skill (e.g., the annotations module of a cooperative design software). Second, these functional modules should be interconnected by diffusion links (e.g., the hyperlinks or dropdown menus) that favour the self learning dynamics once the user has adopted a particular module through the targeted training proposed by the CAMM.

We believe that a quantitative analysis of the CAMM through an experiment with a greater number of users would help to elucidate the impact of the number and the type of links between modules on the dynamics of the adoption phenomenon. This quantitative experiment would allow the characterisation of different learning profiles and their probability of occurrence for further analysis.

The influence of user's behaviour in CSCW software and in general in IT acceptance must be deeply studied and supported with robust quantitative experimental data using ethnographic, anthropologic and psychological methods to properly determine the influence of motivational factors in the diffusion of technologies in the workplace.

Finally, this approach contributes to the emerging theory of the learning organisation by providing a learning mechanism that helps the enterprise to control CSCW software adoption and improve the rate of success of all the modules of the CSCW software.

## **Acknowledgements**

The authors would like to thank the staff of Saint-Gobain Glass for their support throughout the duration of this project. This work was supported by a CIFRE convention granted by the French Government.

## **References**

- Argyris, C. and Schön, D.A. (1996) *Organizational Learning II: Theory and Practice*, No. 0-201-62983-6, Addison-Wesley Publishing Company.
- Bate, J. and Travell, N. (1994) *Groupware*, Alfred Waller Limited.

- Boujut, J.F. and Blanco, E. (2003) 'Intermediary objects as a means to foster co-operation in engineering design', *Computer Supported Cooperative Work*, Vol. 12, No. 2, pp.205–219.
- Bowker, G.C., Star, S.L., Turner, W. and Gasser, L. (1997) *Social Science, Technical Systems, and Cooperative Work: Beyond the Great Divide*, Lawrence Erlbaum Associates, Mahwah, USA.
- Briggs, R.O. (2006) 'On theory-driven design and deployment of collaboration systems', *International Journal of Human-Computer Studies*, Vol. 64, pp.573–582.
- Carroll, J.M., Rosson, M.B., Convertino, G. and Ganoë, C.H. (2006) 'Awareness and teamwork in computer-supported collaborations', *Interacting with Computers*, Vol. 18, No. 1, pp.21–46.
- Chiu, M. (2002) 'An organizational view of design communication in design collaboration', *Design Studies*, Vol. 23, pp.187–210
- Christiansen, E. and Kanstrup, A.M. (2004) 'Support competence and the motivation of adopters to become self-regulated ICT-learners', *International Conference on Workplace Learning*, Copenhagen, Denmark.
- D'souza, M.E. and Greenstein, J.S. (2003) 'Listening to users in a manufacturing organization: a context-based approach to the development of a computer-supported collaborative work system', *International Journal of Industrial Ergonomics*, Vol. 32, pp.251–264.
- Davis, F.D. (1989) 'Perceived usefulness, perceived easy of use and user acceptance of information technology', *MIS Quarterly*, Vol. 13, pp.319–340.
- Eynard, B., Gallet, T., Roucoules, L. and Ducellier, G. (2006) 'PDM system implementation based on UML', *Mathematics and Computers in Simulation*, Vol. 70, No. 5, pp.330–342.
- Griffith, T.L. (1996) 'Negotiating successful technology implementation: a motivation perspective', *Journal of Engineering and Technology Management*, Vol. 13, pp.29–53.
- Grudin, J. (1994) 'Groupware and social dynamics: eight challenges for developers', *Communications of the ACM*, Vol. 37, No. 1, pp.92–105.
- Gutierrez, O. and Friedman, D.H. (2005) 'Managing project expectations in human services information systems implementations: the case of homeless management information systems', *International Journal of Project Management*, Vol. 23, pp.513–523.
- Hares, J. and Royle, D. (1994) *Measuring the Value of Information Technology*, John Wiley, Chichester, UK.
- Hatchuel, A. (1996) 'Coopération et conception collective: variété et crises des rapports de prescription', in De Terssac, G. and Friedberg, E. (Eds.): *Coopération et conception*, Octares éditions.
- Hill, W. (1985) *Learning: A Survey of Psychological Interpretations*, Harper and Row, New York.
- Huber, G.P. (1991) 'Organizational learning: the contributing processes and the literatures', *Organization Science*.
- Hubert, D., Thai, M. and Nogier Molly-Mitton, C. (1995) *Le groupware et ses applications: dossier prospectif*, CXP International, Paris.
- Igalens, J. and Roussel, P. (1999) 'A study of the relationships between compensation package, work motivation and job satisfaction', *Journal of Organizational Behavior*, Vol. 20, pp.1003–1025.
- Jones, B., Valdez, G., Nowakowski, J. and Rasmussen, C. (1994) *Designing Learning and Technology for Educational Reform*, North Central Regional Educational Laboratory, Oak Brook, USA.
- Kensing, F., Simonsen, J. and Bodker, K. (1998) 'MUST: a method for participatory design', *Human-Computer Interaction*, Vol. 13, pp.167–198
- Millet, D., Bistagnino, L., Lanzavecchia, C. and Camous, R. (2005) 'Does the potential of the use of LCA match the design team needs? Suitability of LCA for product design', *Journal of Cleaner Production*, Vol. 15, No. 4, pp.335–346
- Moonen, J. (1997) 'The efficiency of telelearning', *Journal of Asynchronous Learning Networks*, Vol. 1, No. 2, pp.68–77.

- Nonaka, I. (1995) *The Knowledge Creating Company*, ISBN 0195092694, Oxford University Press, Oxford.
- Norman, D.A. (1999) 'Affordances, conventions and design', *Interactions*, Vol. 6, No. 3, pp.38–42.
- Okamura, F. et al. (1994) 'Helping CSCW applications succeed: the role of mediators in the context of use', proceedings of CSCW'94', *Conference on Computer Supported Cooperative Work*, pp.55–66
- Orlikowski, W.J. (1991) 'Integrated information environment or matrix of control? The contradictory implications of information technology', *Accounting, Management and Information Technologies*, Vol. 1, No. 1, pp.9–42.
- Orlikowski, W. (2000) 'Using technology and constituting structures: a practice lens for studying technology in organizations', *Organization Science*, Vol. 11, No. 4, pp.404–428.
- Ozturk, N., Yıldız, A.R., Kaya, N. and Ozturk, F. (2006) 'Neuro-genetic design optimization framework to support the integrated robust design optimization process in CE', *Concurrent Engineering: Research and Applications*, Vol. 14, No. 1, pp.5–16.
- Prasad, B., Morenc, R.S. and Rangan, R.M. (1993) 'Information management for concurrent engineering: research issues', *Concurrent Engineering: Research and Applications*, Vol. 1, No. 1, pp.3–20.
- Rabardel, P. and Bourmaud, G. (2003) 'From computer to instrument system: a developmental perspective', *Interacting with Computers*, Vol. 15, No. 5, pp.665–691.
- Restrepo, T., Millet, D., Gidel, T., Armand, P. and Aoussat, A. (2006a) 'Evaluation de l'efficacité dans la spécification et l'intégration participatives des outils CSCW chez Saint-Gobain Glass', in Yannou, B. and Bonjour, E. (Eds.): *Évaluation et décision dans le processus de conception*, (Traité IC2, série Productique) Editions Hermes, Paris.
- Restrepo, T., Millet, D., Gidel, T., Armand, P. and Aoussat, A. (2006b) 'CSCW tools as an evolution vector in innovation projects', *IDMME Congress*, Grenoble.
- Rogers, E.M. (2003) *Diffusion of Innovations*, 5th ed., Free Press, New York.
- Rosenberg, N. (1972) 'Factors affecting the diffusion of technology', *Explorations in Economic History*, No. 1, pp.3–33.
- Ryan, R.M. and Deci, E.L. (2000) 'Intrinsic and extrinsic motivations: classic definitions and new directions', *Contemporary Educational Psychology*, Vol. 25, pp.54–67.
- Standish Group (2005) *2004 CHAOS Demographics and Project Resolution*, The Standish Group.
- Stewart, J. (2006) 'Cognition = vie: quelques implications élémentaires pour la question de la motivation et l'action', *Interdisciplinary Seminar of Cognitive Sciences*, January, Compiègne.
- Sun, H. and Zhang, P. (2006) 'The role of moderating factors in user technology acceptance', *International Journal of Human-Computer Studies*, Vol. 64, pp.53–78.
- Tannenbaum, S.I. and Yukl, G. (1992) 'Training and development in work organizations', *Annual Review of Psychology*, Vol. 43, pp.399–441.
- Van Wart, M. et al. (1993) *Handbook of Training and Development for the Public Sector*, p.334, Jossey-Bass, San Francisco, CA.
- Venkatesh, V. (2000) 'Determinants of perceived ease of use: integrating control, intrinsic motivation, and emotion into the technology acceptance model', *Information Systems Research*, Vol. 11, No. 4, pp.342–365.
- Venkatesh, V., Morris, M.G., Davis, G.B. and Davis, F.D. (2003) 'User acceptance of information technology: toward a unified view', *MIS Quarterly*, Vol. 27, No. 3, pp.425–478.
- Wenger, E. (1998) *Communities of Practice: Learning, Meaning and Identity*, ISBN 0521663636, Cambridge University Press, Cambridge.
- Woodhall, M. (1987) 'Cost-effectiveness analysis in education', in Psacharopoulos, G. (Ed.): *Economics of Education: Research and Studies*, Pergamon Press, Oxford.