Abstract

Nowadays the traditional idea about information systems has changed due to the fact that communication technologies are better. An information system is not a simple program which resolves a particular problem. Now it is a coordination, cooperation and/or communication oriented application. Applications which are sometimes used by a high number of space and time distributed users.

The newest applications are Web collaborative systems which use communication across the Internet. In this way, they improve their potential.

Developing collaborative systems is a difficult task. To do this in the best way, efficient methodologies must been used.

Lots of methodologies resolve the software life cycle. In this paper, we describe a new specialized, collaborative systems implantation oriented methodology called IGCWS.

1. Introduction

It is true to say that Enterprises or institutions have a lot of workers. It is also true to say that workers in organizations need to communicate with each other. Usually many people participate in the same job, in the same project and usually at the same time. Computer Science has invaded everywhere in the world, the whole of society and it has provided something new.

Moreover, everyday it is easier to find people who prefer working at home (tele-working). Some organizations have several offices in a building, in the town, in the country or even in the world.

Communication and collaboration have always been necessary and everyone can see this in the newest applications. Before the introduction of computers, daily phone calls and periodic meetings were very common. For this reason, companies had to spend a large sum of money in order to pay telephone bills, or journeys because of these meetings. Then, a new way of communication appeared: video-conferencing and other tools across the Internet or an Intranet. These tools allow communication between all the members of an organization, sometimes, even in real time, and always, with very low cost, at least, cheaper than the other way.

You can also communicate with other members in the company by electronic mail or similar. These are non-real-time tools but they are quicker than traditional tools e.g. ordinary mail. And again cheaper.

There are some systems which allow all the members in the company to be informed: forums, news etc.

There is no doubt that groupware, systems with communication, cooperation and coordination tools [3] [4], are strongly joined to the most current applications. Groupware is “computer-mediated collaboration that increases the productivity or functionality of person-to-person processes” [2].

We are going to show the need for communication tools using an example: Information systems allowing Internet and intranet communication.

Two analysts go to a client enterprise and they take all the requirements about a Web project.

At the time, there does not seem to be any problem. Both of them have to document the project definition. This document is very important because it is the base of the project, and it includes Web features, requirements, what the portal has and what the portal does not have, the budget etc.

This document is made between the two analysts, but they work in different places, and that is precisely the big problem.

Firstly, when the analysts write the document in their place of work, on their own computers, how do they make only one document? It would be logical for both of them to work on the same document, placed on a common server. But, if we suppose that they
work on the same document, on the same server, and perhaps at the same time, what are the correct changes? Which is the correct document? In fact, neither of them because the correct file would be the one which has all the changes and inserts in a coherent way.

Another example: two designers who make one single design. Besides the problem of the first example, if they are not near each other, how are they to agree on the design? In other words, if one cannot see what the other is making, it is very difficult to discuss whether one likes it or not. Even by telephone it is not easy. Why do they not use a whiteboard?

You can resolve all these kinds of problems with particular tools, with a collaborative system, with a Groupware application. Cooperation, coordination and communication tools, person-to-person and person-to-computer. Moreover, there are people designing new tools so that commercial toolkits not only provide basic communication facilities such as socket programming or remote procedure calls, but also something more sophisticated [10].

In this paper we describe the development of collaborative systems, of Groupware systems in general and their current importance. Then we see why a methodology should be used in all software development and in Web-based collaborative applications in particular. After that IGCWS methodology is described as a part of our work in the research group. Finally, we show, briefly, a real case in which we applied this methodology.

2. Collaborative Systems Development

In general, when we talk about Groupware, we are talking about special systems which make working in an organization easier. It provides coordination, communication and cooperation tools (Communication an Information Technologies) across computer networks.

So, a new philosophy arises in applications development. These applications resolve a particular problem and, besidees, they make communication between final users easier (just like coordination and cooperation).

Traditional methodologies can be used in this new generation application.

That is, Software Engineering must be applied from system planning. Before doing an implementation and/or an implantation some stages should be followed, and a methodology is perfect for doing this.

A new methodology will be used in these cases: IGCWS, Implantation Guide for Collaborative Web-based Systems. It is based in Métrica v3 which is applied by the Spanish Public Administration Ministry. It is born from the generalization of all the steps in all our implantation projects.

In the fifth point an implantation example is described. It is a real case where we deployed a Groupware system in an organization.

IGCWS is originated from some common points found in various collaborative projects. One of these was a communication and shared information system thought of by the group of pupils of Visual Programming, which is an optional subject of the degree in Computer Science at the Escuela Politécnica Superior de Albacete UCLM; or the implantation of a documented management system in our investigation and development group, LoUISE, also in UCLM.

These are just two examples of many cases nowadays of solving problems of groups of people with IGCWS.

There is always a good way of doing things and the use of a methodology can help in software life cycle, though it does not always have to be used for developing a product, but it is possible to deploy the solution based on a group of existing tools.

3. Methodologies

Groupware systems design and development have to be defined using suitable methodologies.

Recently several new software engineering methodologies have been defined in order to cover the software life cycle, from planning to deployment with application to many different systems.

A previous study was made into Metrica v3 [8] and RUP [9] because they are well known and proven methodologies. Metrica is used in Spanish public administration. These two methodologies are not specifically oriented to collaborative system design.

A research group from Twente University (Holland) has developed a specific methodology called Component-Based Groupware Development Methodology [5]. They define four abstraction levels related to groupware application development: Enterprise, System, Component and Object and use three views: Structure, Behavior and Interaction. Each view provides a different perspective and all are interrelated.

Another interesting methodology is AMENITIES [6], oriented to collaborative system modeling, defined by a research group from the University of Granada (Spain). AMENITIES defines four views
with the detection of the most relevant aspects. The four views are Group, Knowledge, Interaction and Information. The views are described using UML charts [7], in connection with other software techniques and methods.

The system is carried out in three phases. The first, the requirement model, is defined via case uses and so on. In the second phase the collaborative system is modeled with its special features and, finally, in the third phase the system reaches its first design.

4. IGCWS

IGCWS (Implantation Guide for Collaborative Web-based Systems) [1] has emerged as a product of several mature Groupware systems which are being deployed successfully by the Laboratory of User Interaction and Software Engineering (LoUISE).

In this way we should say that after every deployment there is a very solid human team made up of analysts, developers, system administrators, researchers and so on because it is an issue that is horizontal to many knowledge areas going through analysis of data for requirements and to decision making (by system analysts) and to the development of many tools or components, made by developers and managed by system administrators for the management of networks and servers.

With all these real cases a practical methodology has been developed and put into practice with some interesting features that result in an efficient way of working and a method that allows rapid deployment of the system.

IGCWS (Implantation Guide for Collaborative Web-based Systems) is based on the next main stages:

a. First contact: First of all some formal meetings should be had, involving Project manager and analysts (one or more) with the people in charge of the objective group which needs the integral collaborative solution.

In this meeting some initial needs concerning objectives and possible solutions are captured. Alternative possibilities are explained and also the place, date and time for the next meeting is established.

b. Project definition: a time to make a document reflecting main the points of the needs and system requirements including an abstract of the analysis made and a detailed explanation of the proposal offered, a time planning, the cost of the solution and a special section with some relevant information to be considered about the project.

This document must be presented in next meeting with the initial participants in order for them to express their opinions about whether the document represents the real solution and put the plan in action.

c. Requirements collection: the next meetings are between analyst and future system users, where an exhaustive analysis of user requirements is made using some software engineering techniques such as brain storming, interviews, etc. This is not so easy because of communication problems between analyst and users, lack of understanding etc. so audacity is very important on the part of the analyst.

It will probably be necessary to have further meetings to clear up some requirements and user doubts that are very often confusions or simply were not made clear in the first place.

d. Technological situation: At the same time as requirements are being cleared up other information about technological needs can be collected in order to know more about the hardware, software, data, and

![Figure 1. Planning stage diagram](image-url)
network infrastructures that the group has. This is a task for the analyst together with the system administrator. Usually there is at this point a need for some adjustments in technology to introduce all the information that has not yet been included. It may be necessary to introduce an Intranet if there is not one already present.

In many cases it only supposes the addition of a component like a server and its subsequent configuration to achieve a particular objective and therefore giving support to the main system.

e. System planning and viability study; these stages are based on METRICA 3 methodology processes. These tasks have been carried out by analysts and have to be explained very clearly to the person responsible for the group. It is now time to make a decision about whether to develop a new tool or use a commercial product already on the market.

In figure 1. a planning diagram is shown, as an example, corresponding to the planning stage.

f. If the decision is to develop a new system it is necessary to implement new stages: system analysis and design carried out by the analyst and after that the development and deployment by developers and the people in charge of the group respectively.

g. If on the contrary the decision has been to use an existing tool, neither the stage of system analysis and design nor system development would be necessary, going directly to the deployment of the solution to the final users and the acceptance by the people in charge of the group.

But it may be necessary to introduce a new component or module or another compliant tool that the development team should tackle in a small stage of development after a more or less simple analysis and design by the analyst.

We say “simple” referring to the task as a small portion of the whole project.

h. Finally, whatever choices have been made, a maintenance system is needed, by the same team or by other groups or companies. If this task is run by the same team it would be done by the developers.

In any case, the work carried out should be supervised by the project manager, from first contact to maintenance.

A very good practice is to have some formal meetings in order for the people in charge of the group to be informed about the current state of the project; these meetings give them confidence and make them feel looked after. At the end of every main meeting it is very important for all agreements to be recorded as minits, reflecting the date, the people present, the subject and the details of the meeting, with a section expressing the agreements and the date of the next meeting.

5. Case study

The IGCWS methodology has been applied in several real projects relating research groups, university teaching, and companies [1]. In this paper we show the case of our research group called LoUISE (Laboratory of User Interaction and Software Engineering at the University of Castilla-La Mancha).

There are 18 members working together in the research group with different roles from students to researchers and professors. The group carries out R+D projects in the ICT area.

The team needs to access shared resources, and also the system should be protected from unauthorised access. Additionally, collaborators out of the research group should have restricted access.

R+D projects are special in the sense that the participation of many different people from public and private institutions is usual. All these people have to work together in the projects.

Documents like snapshots, internal organization, draft documents, participation in congresses and conferences, project prototypes, technical and information reports, news, links, references, and many other resources have to be shared in an orderly way. This information changes frequently and, in many cases, the success of a project depends on how the information is managed.

Searching for information should be a quick well organized operation. Communication between members has to be available in different models: synchronous and asynchronous.

The document management system is the most important part of the collaborative Web systems. We need a central catalogue or directory of all documents organized in working groups or areas. The solution has to be customized.

This kind of collaboration is based on a set of tools that support the workgroup, via Web protocols. The information can be shared easily in an intranet and all members work on the same document. The document versioning is a key aspect of this system as was previously established.

Nowadays the protection of documents and personal information has increased due to new laws. The users will play different roles in the system. This feature must be supported by the system with different levels of access.
The deployment of the system does not consist purely of the installation of one or two separated applications, but in the study of the needs of the workgroup, planning, viability study which decides if a commercial tool is selected instead of developing the system from the ground. In addition deployment planning should be defined, installation and final user system validation. After that system management and administration should be planned.

### Table 1. IGCWS stages.

<table>
<thead>
<tr>
<th>Stage</th>
<th>Roles</th>
</tr>
</thead>
<tbody>
<tr>
<td>First contact</td>
<td>Project director, group responsible, advisor (and analyst)</td>
</tr>
<tr>
<td>Project definition</td>
<td>Project director, group responsible, advisor (and analyst)</td>
</tr>
<tr>
<td>Requirements collection</td>
<td>Analyst, client, users</td>
</tr>
<tr>
<td>Technological situation</td>
<td>Analyst, system administrator</td>
</tr>
<tr>
<td>System planning</td>
<td>Advisor</td>
</tr>
<tr>
<td>Viability study</td>
<td>Analyst</td>
</tr>
<tr>
<td>Analysis</td>
<td>Analyst</td>
</tr>
<tr>
<td>Design</td>
<td>Analyst</td>
</tr>
<tr>
<td>Building</td>
<td>Analyst, developer</td>
</tr>
<tr>
<td>Implantation</td>
<td>(depend on)</td>
</tr>
<tr>
<td>Acceptation</td>
<td>Group responsible, Project director</td>
</tr>
<tr>
<td>Management</td>
<td>Developer team or third party</td>
</tr>
</tbody>
</table>

Defining the tasks to be done is as important as deciding responsibilities.

In Table 1, the different stages of IGCWS are shown. After following the first five stages (first contact, project definition, requirements engineering, technology description, planning and viability study) we decide to use a commercial solution.

In the viability study, we first analyze commercial solutions (GNU included) or individual resources that could be applied. In this point a comparative cost/benefits rate is calculated for each solution together with other important factors (deployment, technology culture and other risks).

This study concludes by selecting a solution. In this case we decide to use a set of commercial collaborative tools to form a groupware system. The resources needed are an information management system, a synchronous communication system (chat style) and asynchronous communication system (email).

With the solution selected we need to plan the implantation and installation of the system.

### 6. Conclusions

In this article the IGCWS methodology is presented as a result of different projects, making a generalization of the steps followed in each of them. Thus, a line of investigation centered on the study of collaborative systems and how to solve them by some predefined steps is opened.

### 7. References