Experiences of Using Different Communication Styles in Business Process Support Systems with the Shared Spaces Architecture

Ilia Bider[1,2], Paul Johannesson[2], and Rainer Schmidt[3]

[1] IbisSoft ab/SU, Sweden
ilia@ibissoft.se
[2] SU, Sweden
pajo@dsv.su.se
[3] HTW-Aalen, Germany
Rainer.Schmidt@htw-aalen.de

Abstract. Though the concept of shared spaces had been known for quite a while, it did not become popular until the arrival of the internet and social software. Via this way, the concept has penetrated other IT-areas, including the area of Business Process Management, which brings about the needs of investigating the usage of shared spaces in connection to Business Process Support (BPS). The first part of this experience report describes the authors' experience of building, introducing in the operational practice, and using BPS systems based on the shared spaces architecture. It presents three examples of applications aimed at supporting collaboration/communication in the frame of business process instances. These systems use three different mechanisms for arranging communication/collaboration. The first system is based on collaborative planning; the second one is based on the specialized structure of the shared spaces, and the third one on changes in the status of the processes. The second part of the paper is devoted to the analysis of the examples from the first part in order to create a preliminary taxonomy of communication styles in systems with shared spaces architecture. For this end, the authors identified three binary parameters that characterize the way invitations to visit a shared space are issued. These parameters can be used for analyzing communication capabilities of BPS systems, as well as other types of computer systems, with the shared spaces architecture.

Keywords: business process, groupware, communication, shared space.

1 Introduction

The concept of shared spaces is well known [1][2] and has become widely used in the Internet era in connection with advances of social software. A blog, personal journal, and even a photo album are all examples of shared spaces, as they allow sharing information.

Now, shared spaces begin to penetrate the business world as a new generation of workers, brought up with computers, arrives to the labor market. Most of them are

H. Mouratidis and C. Rolland (Eds.): CAiSE 2011, LNCS 6741, pp. 299–313, 2011.

digital natives [3] fully integrated into a multitude of social networks and continuously using shared spaces. The trend of using shared spaces for communication concerns the area of business processes in the highest degree [4].

We believe that proper employment of shared spaces in BPS systems requires good understanding of how such spaces enable communication and what kind of advantages and limitations they have. Therefore, we undertake an attempt to identify important styles of communication via shared spaces.

The goal of this paper is twofold. The first sub-goal is to present our experience of building, introducing in the operational practice, and using BPS systems based on the shared spaces architecture. The second sub-goal is to analyze this experience in order to create a preliminary taxonomy of various communication styles in systems with shared spaces architecture.

In order to create a proper foundation for the later considerations, we start with reviewing the role of shared spaces in BPS systems (Section 2). We also explain how a system that employs shared spaces differs from a traditional business process support system based on workflow.

The first sub-goal of the paper is achieved by discussing three examples of applications aimed at supporting collaboration/communication in the frame of business process instances (Sections 3-5). All three systems have been built based on the state-oriented view on business processes [5], and all of them use shared spaces to facilitate communication between process participants. The systems employ three different ways of using shared spaces for communication. The first system uses collaborative planning. In the second one, communication is based on the specialized structure of shared spaces. In the third one, the communication is based on changes in the status of processes.

The way of using shared spaces in each of the systems is not arbitrary but reflects the types of business processes each system supports. In the first case, the system supports loosely-structured processes that require much ad-hoc communication between people engaged in them. In the second case, the system supports relatively structured processes. In the third case, the system supports simple real-time processes with high requirements on the speed of communication.

The second sub-goal of the paper is achieved by analyzing the differences between the systems in respect of how the invitations to visit shared spaces are issued (Section 6). Based on this analysis, three binary parameters are introduced to differentiate communication styles in systems with shared spaces architecture.

In the last part of the paper, we give a short overview of related works (Section 7), and discuss our experience and draw plans for the future research based on it (Section 8). As the paper is an experience report, in the “Related works” section, we pay special consideration to list our own works that explain the theoretical background of our experience as well as give the possibility for the reader to learn about some parts of our experience in more details.

2 A Role of Shared Spaces in BPS Systems

There exist numerous definitions of what a business process is, each of them focusing on a particular property of business processes. For the sake of this paper, we take the following view on business processes:

"A business process is a way of combining efforts of several people for reaching a (well or not so well defined) goal".

Accepting this view, we consider only those business processes in which people play a role of the driving force behind the processes, leaving totally automated processes, and processes where people are used just for well-defined operations aside. Such processes require extensive communication between the participants of each process instance in order to reach the operational goal of the instance.

By a BPS system, we understand a system that helps process participants to run their process instances according to a process (type) definition. From the communication point of view, a BPS system with shared spaces differs from a pure workflow system by the kind of "information logistics" the system employs for providing process participants with instructions and information needed to complete their tasks in the frame of a process instance [6]. The workflow-type BPS systems use a so-called "conveyor belt" logistics [6] in which instructions, and information that is needed are sent to the "next in-line".

A BPS system with shared spaces employs a so-called "construction site" information logistics [6]. Such a system has no explicit data/information flow. A shared information space is created for each process instance to hold all information that is relevant to the process instance, e.g., documents received and sent, information on tasks planned and completed, reports on results achieved when completing these tasks, etc. All this information is easily available each time a process participant is invited to visit this space and complete some task related to it. Thus, a shared space is similar to a construction site where different kinds of workers are invited to complete their own tasks and leave the rest to the others.

The functioning of a BPS system based on shared spaces can be described in the following way:

– When a new process instance starts, a new shared space is created. It gets a unique name, an owner (responsible for the instance), and possibly, an instance team.
– When the process instance reaches its operational goal, the shared space is closed (sealed), but remains accessible for reading (an instance goes to the archive).
– A person who is assigned a task in the frame of the process instance "goes" to this instance's shared space to get information he/she needs for completing the task and reports the results achieved in the same space.

The "construction site" information logistics via shared spaces has certain advantages compared to the traditional communication schemes for those business processes in which instances can vary considerably from one to another. In such a situation, it is difficult to decide what and how much information needs to be sent to a person completing a certain task. When a person is invited to visit a certain part of a shared space, he/she oversees not only this local part, but also everything adjacent to it, and can use this additional information when completing his/her task without being explicitly told to do so. In other words, if using a traditional communication scheme you send one document to a person, and this is all he/she gets. If you send a person to work on a certain document placed in some corner of a shared space, he/she can access not only this document, but also other documents in this corner, or anywhere else in the whole shared space. More detailed justification of using shared spaces in BPS systems from the business point of view can be found in [7].

For the shared spaces technique to work efficiently in a BPS system, two conditions should be fulfilled:

- Shared spaces are properly structured. In a normal business environment, a person participates in many process instances, and, often, in parallel. For the shared space technique to work efficiently, he/she needs to understand the situation in a shared space he/she is visiting at a glance, and quickly find all information related to the task at hand.
- Invitations should give process participants a clear understanding on why they have been invited and what they are expected to do in each particular shared space.

Note that invitations to visit shared spaces in BPS systems have a different meaning from that in social software. In the latter, invitations are not binding; a person invited may not visit the shared space at all. In a BPS system, however, following an invitation is mandatory or at least strongly recommended; otherwise the whole communication /collaboration scheme will break down.

Naturally, getting an invitation does not constitute the only reason why a person would like to visit a shared space. He/she can do it in an arbitrary manner, or because some event happened in the frame of a process instance that is of value to be registered in its shared space.

In the next three sections, we will introduce three examples of BPS systems that use the principles outlined in this section.

3 A System with Collaborative Planning

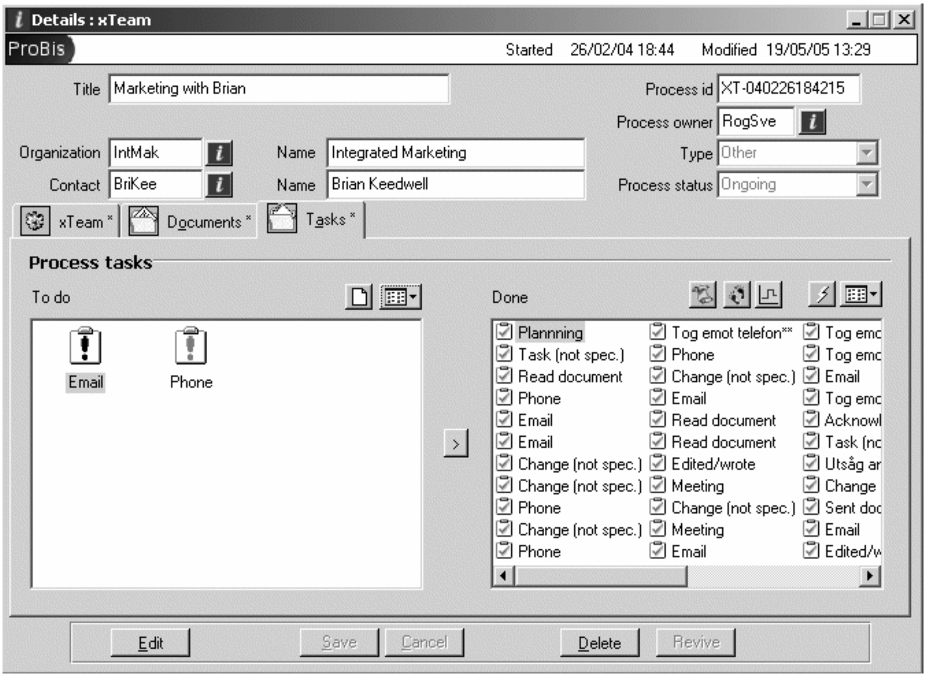
3.1 Description

A system called ProBis was developed based on ideas from [5] for a Swedish interest organization (The Swedish Union of Tenants) in 2003-2006 as described in [8][9]. A shared space in ProBis is presented to the end-user as a window divided in several areas by using the tab dialogues technique, see Fig. 1.

Some areas of the window are standard, i.e. independent from the type of the business process; others are specific for each process type supported by the system. Standard areas comprise such attributes and links as:

- Name and informal description of a process instance
- Links to the owner, and, possibly, the process team
- Links to the relevant documents, created inside the organization, and received from the outside

The standard part of ProBis shared space includes also the task area (tab) that contains two lists, as in Fig. 1. The *to-do* list (to the left in Fig. 1) includes tasks planned for the given process instance; the *done* list (to the right in Fig.1) includes tasks completed in the frame of it. A planned task defines what and when something should be done in the frame of the process instance, as well as who should do it. All tasks planned for a given person from all process instances are shown in the end-user's personal calendar. From the calendar, the user can go to any shared space for which a task is assigned to him/her in order to inspect, change, or execute this task.



Details : xTeam

ProBis Started 26/02/04 18:44 Modified 19/05/05 13:29

Title: Marketing with Brian

Process id: XT-040226184215

Process owner: RogSve

Organization: IntMak

Name: Integrated Marketing

Type: Other

Contact: BriKee

Name: Brian Keedwell

Process status: Ongoing

Process tasks

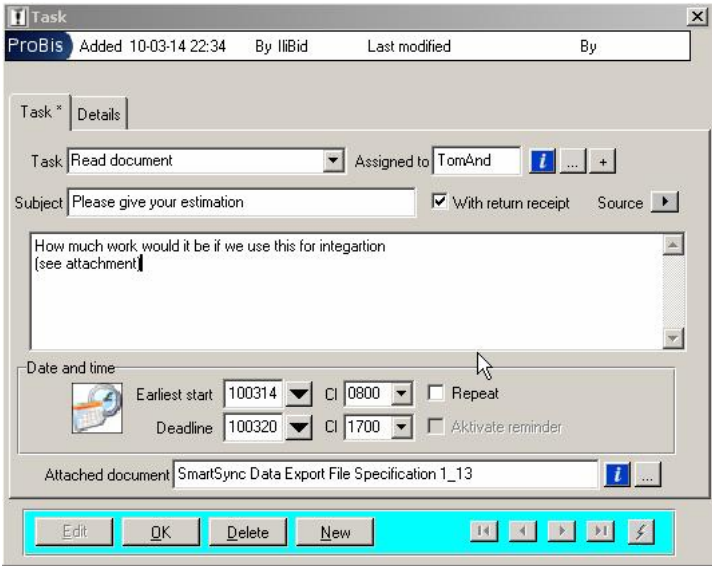
To do

Email
Phone

Done

☑ Plannning
☑ Task (not spec.)
☑ Read document
☑ Phone
☑ Email
☑ Email
☑ Change (not spec.)
☑ Change (not spec.)
☑ Phone
☑ Change (not spec.)
☑ Phone
☑ Tog emot telefon**
☑ Phone
☑ Change (not spec.)
☑ Email
☑ Read document
☑ Read document
☑ Edited/wrote
☑ Meeting
☑ Change (not spec.)
☑ Meeting
☑ Email
☑ Tog emc
☑ Tog emc
☑ Email
☑ Tog emc
☑ Acknowl
☑ Task (nc
☑ Utsåg ar
☑ Change
☑ Sent doc
☑ Email
☑ Edited/w

Fig. 1. View on the ProBis shared space



Task

ProBis Added 10-03-14 22:34 By IliBid Last modified By

Task: Read document

Assigned to: TomAnd

Subject: Please give your estimation

☑ With return receipt

Source

How much work would it be if we use this for integartion
(see attachment)

Date and time

Earliest start: 100314 Cl 0800 □ Repeat

Deadline: 100320 Cl 1700 □ Aktivate reminder

Attached document: SmartSync Data Export File Specification 1_13

Fig. 2. Assigning a task to another user in ProBis

The only way of communicating via ProBis is by assigning a task to the communication partner. This is done by filling a form as in Fig. 2. One chooses the task from the list, assigns it to another user of the system, adds a textual description and some parameters, for example a document that is already registered in the process instance space. The task list is configurable and can be adjusted for each installation and process type.

To further facilitate communication, several more advanced features were added to ProBis. For example, there is a possibility to plan the same task to many users. Additional users can be added from the list with the "+" button (See Fig. 2), or can be fetched from a predefined group. Each user gets its own task in the calendar and will need to go and complete it independently of other users. Multi-user planning gives a possibility to easily raise attention of several people to some event that has happened in the process instance. Another advanced feature is the "Returned receipt" check-box which ensures that the planner gets a special "Attention" task planned as soon as the task he/she has assigned to somebody else has been completed.

3.2 Experience of Use

Based on our experience with ProBis, collaborative planning provides a very efficient way of communication/collaboration in the frame of business process instances. It is especially useful for:

- loosely structured processes, i.e. processes for which there are no predefined ways for handling each instance.
- processes driven by a professional team that knows how to use the system quite well.

There are, however, two drawbacks with the approach when using it for more structured processes that involve occasional users:

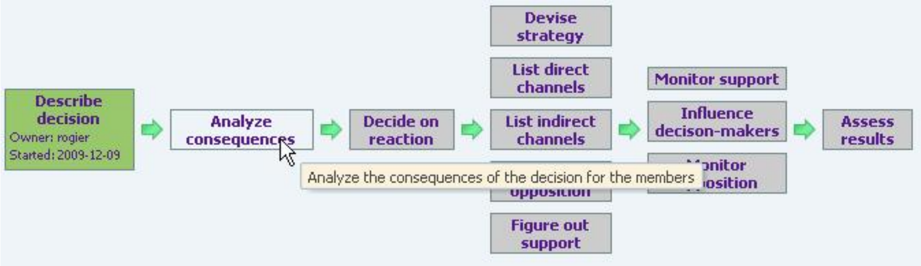
- The dynamic aspect of business processes is poorly visualized. One needs to go through the done-list and browse the history to get an understanding of how a given process instance is developing in time.
- Using the system puts some requirements on the user, as he/she needs to understand the general ideas built in the system and get some training. This means that the system is not very friendly for newcomers and casual users. Planning as a way of communication causes the major problem here, as it is considered to be counter-intuitive. Detailed planning is not as widespread in business life as one can imagine.

4 A System with Specialized Structure of Shared Spaces

4.1 Description

In a system with a specialized structure, shared spaces are structured according to the process map designed for a particular process type. In our case, such a map is designed with a tool called iPB [10]. Several systems have been built with the help of this tool. The biggest one is employed in the social office of one of the Swedish municipalities (municipality of Jönköping), where it helps to conduct investigations on suspected child abuse.

A process map in iPB is a drawing that consists of boxes placed in some order, see Fig. 3. Each box represents a step of the process, and the name of the step appears inside the box (no lines or connecters between the boxes). A textual description is attached to each step that explains the work to be done. Each process instance gets its own copy of the map that serves as a table of contents for its shared space, see Fig. 3. The map is used for multiple purposes: as an overview of the case, guidelines for handling the case, and a menu for navigating inside the shared space. The user navigates through the shared space by clicking on the boxes of the steps with which he/she wants to work. Not all boxes are clickable at the beginning; those that are grayed require that one or several previous steps are dealt with first, see Fig. 3. These constraints are defined with the help of so-called business rules.



Analyze the consequences of the decision for the members

Fig. 3. A map used for structuring the shared space of a process instance

A click on a step box redirects the end-user to a web-form that assists him in completing the step. The form contains text fields, option menus and radio-buttons to make choices, checkboxes, as well as more complex fields. The form may also include "static" texts that explain what should be done before one can fill some fields.

From the shared spaces architecture point of view, the iPB solution can be interpreted as follows. The total process instance shared space is divided into a number of subspaces called process steps. The steps are graphically represented to the end-users as boxes. Subspaces may or may not intersect. The structure of a step subspace is represented to the end-users as a form to fill. Intersecting subspaces means that forms attached to different steps may contain the same field(s). Usually, in this case, the intersecting fields can be changed only in one form; they are made read-only in the second one.

The progress in filling the step forms is reflected in the map attached to the shared space via steps coloring. A gray box means that the step form has not been filled and cannot be filled for the moment. A white box means that the step form is empty but can be filled. A step with a half-filled form gets the green color, and additional information about when the work on it has been started, and who started it. A step with a fully filled form gets the blue color, and additional information about the finish date.

The primary way of "forcing" a person to visit a particular shared space in iPB is by assigning him/her to become an owner/co-owner of some step. Such an assignment results in an email message being delivered to this person, and the process to appear in his/her list of “My processes”. When visiting a process shared space, a person can see directly on the map what step(s) are assigned to him, see a green box in Fig. 3.

4.2 Experience of Use

The communication possibilities in an iPB-based system may seem to be too limited. The only way of attracting a given person's attention to visit a particular shared space is by assigning him/her to be an owner/co-owner of some step in the given process instance. No clarification or explanation is given when making the assignment. This should be figured out by the person him/herself from the state of the process, i.e. from the partly filled step form.

However, in practice, this communication mechanism works quite well for relatively structured processes for which it is possible to identify steps. A system of this kind is quite easy to introduce in operational practice, which cannot be said about systems with a collaborative planning style. The communication works well even when participants do not know each other personally.

To extend communication possibilities, we added a rudimentary planning scheme similar to ProBis. In the systems currently introduced, however, this additional mechanism is not being widely used.

5 A System with Communication Based on Status Changes

5.1 Description

The system called eForm was developed for a large Swedish call center (Eniro 118118) to solve the daily staffing problems that can be defined as follows [11]. The scheduling software is run once per month. Staffing requirements may change from day to day (if not from hour to hour) due to changing volumes of inbound calls. In addition, unscheduled absences due to illness, traffic jams, snowfall, etc. make it impossible to totally rely on the pre-generated schedule even when the call volume follows the established pattern. Corrections in the schedule are constantly made to cope with fluctuations in volumes of calls and the number of agents not appearing for work. More agents need to be called in to deal with an increase in the volume of calls or increase in the number of absentees. Alternatively, fewer agents are required when the call volume decreases, or the sick rate due to a seasonable epidemics diminishes.

Effective dealing with fluctuations in the staffing level at a call center requires fast communication channels between agents and managers responsible for operative staffing of the call center. eForm is a web 2.0 system providing efficient channels of communication between three different categories of workers at a call center:

- Central staffing center, aka Control Tower, or just Tower
- Agents
- Coaches (managers)

The system works according to a simple scheme: a communication process starts when one of the participants fills an electronic form that serves as a shared space for this process. This form gets the status New and immediately appears in the list watched by another communication partner. The latter processes the form and sets its status to Finished or changes its status to something else that requires further processing, for example, Question. In the former case, the form disappears from the actual list; but it can still be found through search in the archive (if needed). In the second case, the form disappears from the actual list of this communication partner and appears in the list of some other communication partner, e.g. the one who originally filled the form.

Let us demonstrate how this scheme works on a particular example. Unscheduled absences are reported by agents to their coaches via direct phone calls. A coach communicates this information to the Tower via eForm with a couple of touches of the keyboard by selecting the agent's name, time, and reason for an unscheduled absence. While filling the form, the coach has access to the information about previous unscheduled absences of the same agent. Thus, he/she has a possibility to see a pattern of absences and discuss the matter with the agent. The coach has also a possibility to promptly fill an absence form that concerns more than one agent, for example, in case of an unscheduled training session.

As soon as the coach saves the form, it appears in the absentees list at the Tower. A tower worker makes corrections to the schedule accordingly, after which the form disappears from the actual list (but remains accessible via the archive search). In case of any uncertainty, the Tower quickly returns the form to the coach with a comment by changing its status to Questioned. The coach corrects the form, after which it again appears in the Tower's absentees list.

5.2 Experience of Use

In eForm, there are no explicit "calls" to visit a shared space. The form appears in the list of one of the participating partners dependent on the state of the process, more exactly on the value of one or more fields of the form. From our experience, this mechanism creates a very efficient communication channel, and the system is easy to learn, and introduce in operational practice. This is an important factor in the above business case, because the turnover of agents in a typical call center is, usually, quite high. The communication mechanism works very well for simple real-time processes with strong requirements on the speed of communication.

6 Identifying Communication Styles

Let us investigate differences between the three BPS systems types discussed in Sections 3-5. First of all, there is a difference in the structure of shared spaces. In ProBis – a shared space has a generalized, logical structure. Similar types of objects are gathered on the same tab. For example, there is a separate tab for documents, a separate tab for planned and completed tasks, etc. (see Fig. 1). This reflects the area of ProBis usage – loosely structured processes for which it is not possible to create a more exact structure. In iPB, a shared space is structured in steps according to the "dynamics" of a particular business process type (see Fig. 4). In eForm, shared spaces are quite simple and do not require complex structuring.

Secondly, these three systems implement, on the surface, completely different mechanisms of using shared spaces to facilitate communication/collaboration. ProBis uses collaborative planning. An iPB- based application uses assignment of owners/

co-owners to process steps. eForm uses a list management system to attract the attention of the end-users.

To compare the above mechanisms, we need to abstract from the technical details and consider each mechanism as a way of issuing invitations to visit a shared space. Then, the three mechanisms described in the previous sections can be interpreted as follows.

In ProBis, a task planned for a person represents a manually issued invitation for him/her to visit the process instance shared space with detailed instructions of what he/she is supposed to do there (whom to call, what document to read, etc.). In an iPB-based application, assignment to be owner/co-owner of a step represents a manually issued invitation for this person to visit a particular part of the process instance shared space (a form corresponding to the given step) without any instructions on what he/she is supposed to do there. In eForm, a form appearing in the given person's process list represents an automatically issued invitation to visit the process instance shared space without any instructions on what he/she is supposed to do there.

Generalizing the above, we suggest the following three parameters with binary values for identifying communication styles:

- Issuing technique (*Manual/Automatic*) – an invitation is issued manually by one of the process participants, or automatically by a system based on the state of the shared space.
- Invitation scope (*Global/Localized*) – a person is invited to visit the whole shared space or a particular part of it.
- Invitation instructiveness (*Non-instructive/Instructive*) – a person should him/herself figure out what to do in the shared space based on the state in which he/she finds it when he/she comes there, or an invitation may include instructions on what a person is supposed to do there.

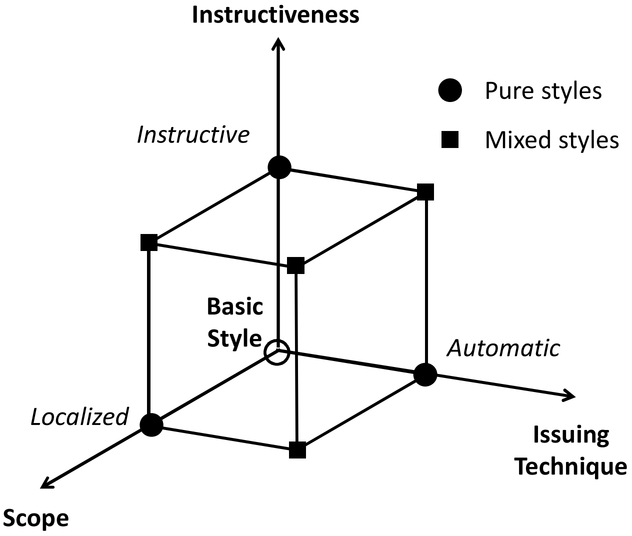


Fig. 4. Communication Styles

The above parameters formally give eight classes that we call communication styles. Let us consider *Automatic* as a more advanced feature than *Manual*, *Localized* as a more advanced feature than *Global*, and *Instructive* as a more advanced feature than *Non-instructive*. Then we can present the classification as a cube in a three-dimensional space where a "basic" style *Manual/Global/Non-instructive* constitutes a zero point, see Fig. 4.

Table 1 below presents styles employed in the systems discussed in this paper. As we can see, each system employs a style that includes only one advanced feature, which means that all styles employed in the systems lie on the axes of the cube in Fig. 4. We call such styles "pure" communication styles, as an opposite to other non-basic styles, which we call "mixed" styles.

Table 1. Communication styles employed in the systems discussed

	Issuing technique	Invitation scope	Invitation instructiveness
Collaborative planning (ProBis)	*Manual*	*Global*	***Instructive***
Specialized structure (iPB)	*Manual*	***Localized***	*Non-instructive*
Status change (eForm)	***Automatic***	*Global*	*Non-instructive*

In Section 2, we stressed that an invitation to visit a particular shared space should give the process participant a clear understanding on why he/she has been invited and what he/she is expected to do there. In the case of eForm, shared spaces are simply structured, thus neither localization nor instructiveness is needed for the invited person to understand what he/she is supposed to do.

Both ProBis and iPB allow quite complicated structures of shared spaces, thus some help is needed to find out what is required from the invited person. In ProBis, this help is provided by instructiveness, which compensates the absence of localization. In iPB, this help is provided by localization, which compensates the absence of instructiveness.

As we already mentioned, the systems discussed in this paper implement "pure" communication styles (only one advanced feature is present in each of them). It does not mean that a mixture is not possible or useful, but only that one advanced feature is sufficient for certain practical purposes as outlined in Sections 3.2, 4.2 and 5.2.

7 Related Works

Works on which the current paper is based are as follows. The earliest attempt to systematically build a BPS system based on the principles outlined in Section 2 (shared spaces along with collaborative and automated planning) was made in 1989-1990. The project, called "DealDriver", is described in [12], [13]. The ideas from the DealDriver project were first presented to the research audience in [14]. More detailed introduction into the state-oriented view on business processes that has been developed in connection to the DealDriver project can be found in [5]. For an overview of the practical experience in BPS systems development up to the time of building the first version of ProBis see [8]. The high-level theory underlying the state-oriented view on business processes is presented in [15][16]. Both DealDriver and ProBis belong to the category of case-handling systems later identified in [17].

Other research works that directly or indirectly belong to the topic of this paper are as follows. The concept of shared spaces is not a new one. It has been widely used in research literature for some time, see, for example, [1]. The simultaneous integration of face to face communication and the exchange of graphical information has been introduced by [2]. The integration of video and audio is proposed by [18]. Embodiments are used to enhance virtual shared spaces in [19]. In [20] media spaces are differentiated from collaborative virtual environments and spatial video conferences.

Using shared spaces for business collaboration was discussed in many research works. A task-oriented collaboration comparable to our collaborative planning can be found in [21]. The new media model introduced in [22] shows tight relationships to shared spaces. Media are spaces where agents can collect and represent information. Roles, describing rights and obligations determine the behavior of agents. In PRODNET [23], a federated database architecture has been used to support shared spaces. Gaia – a middleware infrastructure to enable spaces for cooperatively solving tasks - is presented in [24]. An introduction on how to manage multiple and collaborative tasks is give in [25]. An early system that incorporates some features of ProBis and iPB is OASIS [26]. It supports its users to cooperate for achieving a common goal.

8 Discussion and Future Plans

As was stated in the introduction, the goal of this paper was twofold. The first sub-goal was to present our experience in building domain-specific applications for supporting communication/collaboration in the frame of business process instances. This sub-goal was fulfilled by presenting examples of three systems. Each example contains the description of a system and the review of experience of its usage. In the latter, we outline types of business contexts for which each of the systems is best suited.

As any experience, ours is unique, though some features found in our systems can be found in others, as well. The main characteristic of our approach to building BPS system that differentiate us from others is that all our systems are systematically being built based on the principles of shared spaces architecture outlined in Section 2, and the state-oriented view on business processes from [7]. We are not aware of any other attempts of creating a line of on the surface dissimilar systems that implement these architectural and theoretical principles. In addition, our second example represents not a system, but a tool that can be (and already is) used by others for building BPS systems with the shared spaces architecture. The approach accepted for building BPS system implemented in iPB, as far as we know, is unique.

The second sub-goal was to introduce a taxonomy of communication styles for collaborative systems with shared spaces architecture. This was done based on the analysis of our experience and resulted in the introduction of three parameters with binary values: issuing technique, scope, and instructiveness.

The parameters proposed for the style identification can be applied for the analysis of publicly and commercially available systems that employ shared spaces for communication/collaboration. As an example, let us consider Google Sites [27], which is a shared spaces system created in the context of Google's E-Mail system Gmail [28]. A Google site is a tree-structured collection of pages that can be edited without an external tool using a browser-based interface. Other users may be invited to collaborate as a site owner, editor or reader. Furthermore, it is possible to share a site with everyone on the web. In the enterprise edition of Google Sites, it is possible to give read or write access to all users of the enterprise's domain. Google Calendar is used to delegate tasks. This can be done if the addressee of the delegation shares his/her calendar with the delegating person. The delegating person can create tasks in the calendar of the addressees and thus to delegate tasks to him/her.

According to the communication styles scheme suggested in section 6, Google Sites uses the manual issuing technique (through the calendars). An invitation always has a global scope, as it concerns the whole site. Invitations are instructive, as it is possible to specify what has to be done in the delegated task.

From the point of view of the parameters introduced, the three systems from our experience represent "pure" communication styles, which, of course, does not exclude creating a mixture of styles. In fact, we are in the process of adding the automatic issuing technique to both ProBis (based on [16]), and iPB.

The three parameters for style identification proposed in this paper give only basic characterization of communication capabilities of a system with the shared spaces architecture. More detailed classification is required for covering the nuances. For example, the automatic issuing technique can be divided into two subcategories: general rules, and instance rules. General rules ensure automatic issuing of invitation that covers all instances of the given process type. Instance rules mean a capability to ensure automatic invitations for a particular process instance/case. The latter are often expressed in the form of subscription to certain events in a shared space (see, for example, proposals for adding instance rules to ProBis in [16]).

As follows from the experience presented in the paper, different communication styles suit different kinds of business contexts. For example, high requirements on speed of communication, as in eForm, warrant automatic issuing technique. A complex structure of shared spaces, as in ProBis and iPB, requires either instructiveness or localization (or both).

To find a proper mixture of communication styles for a practical business case, the properties of each communication style should be understood, so that the styles are mixed based on the requirements of a particular business environment. Our current research in progress is devoted to this task. In this research, we analyze our experience from the point of view of business requirements that can be set on the communication/collaboration mechanisms. Here, we differentiate several groups of requirements: functional requirements (e.g., possibility of inviting an arbitrary person at any moment of time), security requirements (e.g., restricting a person's capability of viewing parts of the shared space), social requirements (e.g., support of week ties), and business process requirements (e.g., support for predefined tasks for the given process type).

Acknowledgments. This paper would have never been written without considerable efforts of the team of developers who have designed and implemented ProBis, iPB, and eForm. We are especially thankful to Tomas Andersson, Alexander Durnovo, Alexey Striy and Rogier Svensson.

References

[1] Takemura, H., Kishino, F.: Cooperative work environment using virtual workspace. In: Proceedings of the 1992 ACM Conference on Computer-Supported Cooperative Work, pp. 226–232 (1992)

[2] Ishii, H., Kobayashi, M., Grudin, J.: Integration of interpersonal space and shared workspace: ClearBoard design and experiments. ACM Transactions on Information Systems (TOIS) 11(4), 349–375 (1993)

[3] Prensky, M.: Digital natives, digital immigrants. On the Horizon 9(5), 1–6 (2001)

[4] Nurcan, S., Schmidt, R.: Introduction to the first international workshop on business process management and social software (BPMS2 2008). In: Ardagna, D., Mecella, M., Yang, J. (eds.) Business Process Management Workshops. Lecture Notes in Business Information Processing, vol. 17, pp. 647–648. Springer, Heidelberg (2009)

[5] Khomyakov, M., Bider, I.: Achieving Workflow Flexibility through Taming the Chaos. In: OOIS 2000-6th International Conference on Object Oriented Information Systems, pp. 85–92. Springer, Heidelberg (2000)

[6] Bider, I., Perjons, E., Johannesson, P.: In Search of the Holy Grail: Integrating social software with BPM. Experience Report. In: EB-PISM. LNBIP, vol. 50, pp. 1–13. Springer, Heidelberg (2010)

[7] Bider, I., Perjons, E., Johannesson, P.: A strategy for integrating social software with business process support. In: LNBIP, vol. 66, pp. 372–383. Springer, Heidelberg (2011)

[8] Andersson, T., Bider, I., Svensson, R.: Aligning people to business processes experience report. Software Process Improvement and Practice 10(4), 403–413 (2005)

[9] Bider, I., Striy, A.: Controlling business process instance flexibility via rules of planning. International Journal of Business Process Integration and Management 3(1), 15–25 (2008)

[10] iPB Reference Manual on-line documentation (Online), `http://docs.ibissoft.se/node/3` (accessed June 20, 2010)

[11] eForm – a Staffing Communication System for running day-to-day operations at a call-center (Online), `http://www.ibissoft.se/node/144` (accessed: June 20, 2010)

[12] Bider, I.: Developing tool support for process oriented management. In: Handbook of Systems Development, vol. 1999, pp. 205–222. CRC Press, Boca Raton (1998)

[13] Bider, I.: Object driver: a method for analysis, design, and implementation of interactive applications. In: Handbook of Systems Development, vol. 1999, pp. 81–96. CRC Press, Boca Raton (1998)

[14] Bider, I., Khomyakov, M.: Object-oriented model for representing software production processes. Object-Oriented Technologys, 319–322 (1998)

[15] Bider, I., Khomyakov, M., Pushchinsky, E.: Logic of change: Semantics of object systems with active relations. Automated Software Engineering 7(1), 9–37 (2000)

[16] Bider, I., Khomyakov: New technology - Great Opportunities. How to Exploit Them. In: Filipe, J. (ed.) Enterprise Information Systems, Kluver (2003)

[17] Van der Aalst, W.M.P., Weske, M., Grünbauer, D.: Case handling: a new paradigm for business process support. Data & Knowledge Engineering 53(2), 129–162 (2005)

[18] Bly, S.A., Harrison, S.R., Irwin, S.: Media spaces: bringing people together in a video, audio, and computing environment. Communications of the ACM 36(1), 46 (1993)

[19] Benford, S., Bowers, J., Fahlén, L.E., Greenhalgh, C., Snowdon, D.: User embodiment in collaborative virtual environments. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 242–249 (1995)

[20] Benford, S., Brown, C., Reynard, G., Greenhalgh, C.: Shared spaces: transportation, artificiality, and spatiality. In: Proceedings of the 1996 ACM Conference on Computer Supported Cooperative Work, pp. 77–86 (1996)

[21] Dillenbourg, P., Traum, D., Schneider, D.: Grounding in multi-modal task-oriented collaboration. In: Proceedings of the European Conference on AI in Education, pp. 401–407(1996)

[22] Schmid, B.F.: The concept of media. In: Workshop on Electronic Markets (1997)

[23] Garita, C., Afsarmanesh, H., Hertzberger, L.O.: The prodnet cooperative information management for industrial virtual enterprises. Journal of Intelligent Manufacturing 12(2), 151–170 (2001)

[24] Roman, M., Hess, C.K., Cerqueira, R., Ranganathan, A., Campbell, R.H., Nahrstedt, K.: Gaia: A middleware infrastructure to enable active spaces. IEEE Pervasive Computing 1(4), 74–83 (2002)

[25] Johnson, P., May, J., Johnson, H.: Introduction to multiple and collaborative tasks. ACM Transactions on Computer-Human Interaction (TOCHI) 10(4), 277–280 (2003)

[26] Martens, C., Woo, C.C.: OASIS: An Integrative Toolkit for Developing Autonomous Applications in Decentralized Environments. Journal of Organizational Computing and Electronic Commerce 7(2&3), 227–251 (1997)

[27] Google Sites, (Online), `http://www.google.com/sites`, `http://sites.google.com/` (accessed June 18, 2010)

[28] Google Mail, (Online), `http://www.google.com/mail` (accessed June 18, 2010)