Reengineering Organizational Structures from Within

Dipl.-Wirt. Inform. Marcus Ott — Dipl.-Inform. Carsten Huth — Prof. Dr. Ludwig Nastansky
{Marcus.Ott;Carsten.Huth;Ludwig.Nastansky}@notes.uni-paderborn.de
University of Paderborn, Department of Business Computing, Warburger Str. 100, D-33098 Paderborn
Phone: ++49 (0)5251-603368, http://gcc.uni-paderborn.de/winfo2/GroupOrga

Abstract
The GroupOrga project (Groupware Based Organization Design) examines how structural organization design takes place in its traditional form and also provides a vision of an organization design process. Moreover, its objective is to present an architecture of an organization design application environment that supports workflow management systems (WfMS).

After specifying the term collaboration in the context of GroupOrga, we discuss how collaboration technology can enable better organizational design and design procedures. As a contribution to the WfM discussion, we examine how traditional WfMS are constructed in respect to organizational design processes and point out their current weaknesses. The paper explains the linkage between WfMS and organization design systems.

As a major focus, the varying types of workflow and organization design users in an organization are examined and their involvement in the organizational design process is discussed. The result is a scale of user classes that ranges from employees who only want to get informed to those who actively and regularly participate in the design. In the following, it will be made clear how one enables participative organization design with diverse applications such as Web-based tools.

Concluding, some impediments to distributed organization design will be named and commented upon.

1 Introduction

Katzenbach and Smith ([8], p. 70) describe a team as a small group of people whose capabilities complement one another and who are all engaged for a common cause. In an earlier investigation, the trend towards teams in organizations has been diagnosed as stimulated by numerous developments in businesses, such as fast-moving markets, a trend towards flatter hierarchies, team-based performance ratings, or reports about role model organizations with massive team-orientation ([15]). The assignment given to a team from the organization will be coordinated, directed and completed in the form of independent processing as collaborative teamwork.

"Organization modeling is concerned with describing the structural aspects of an organization. It describes the different parts of an organization, how these parts are related to each other and their properties" ([10], p. 193). Organization design or organization modeling is the branch of management that addresses problems of inefficiency in organizations.

This paper combines both aspects and proposes how collaboration technology can enable better organizational design and design procedures. Thus, it is about collaboration between team members in a specific business context: structural organization design.

2 Collaboration in Organization Design

As a preliminary we want to clarify our understanding of collaboration in the following section. Afterwards, a scenario is set up which requires collaboration technology at different loci of collaboration: Structural organization design through teams. We show how collaboration technology can contribute to the success of modern organizations and how it relates to organizational structure.

2.1 Collaboration — A Working Definition for Organizational Design

Dhar and Olson ([6]) use the term collaboration "to refer to a goal-oriented process involving contract definition and execution among two or more individuals" (p. 34). Hence, collaborative work comes about, when tasks for completion of a product or service are carried out by several people simultaneously. The necessary relations for this collaboration are (pre-)planned and may be predetermined by the product's or service's own characteristics. Apart from collaborative work in physical teams, collaboration can come about in other forms, as well. In the case of the partners not interacting directly, the term distributed collaboration (cp. [2]) can be found. In this indirect model the participants do not (always) communicate personally, but use communication systems to interact and to adapt their personal behavior to the common task. In addition, collaboration is not bound to the physical borders of organizations. It is characterized through collaborative behavior as such, which can involve partners in various organizations and locations.

Difficulties in defining the term collaboration are elaborated by Dhar and Olson when they identify three
influencing factors to collaborative work: complexity, uncertainty, and ambiguity. By complexity they describe the problem of mapping the necessary activities in collective work with the resource requirements associated with these activities and the complexity from an individual's perspective resulting from involvement in several projects simultaneously. Uncertainty refers to the lack of knowledge about what environmental states will prevail and about the time estimates of projects' activities that the individual is involved in. Ambiguity refers to the fact that the collaborative activities may not be well defined. This may particularly be the case in the early stages of a collaboration.

Drawing from this discussion, GroupOrga presents a contribution towards a collaborative work system for the participative design of organizational structures.

2.2 Two Systems Interact — Workflow Management and Organizational Design

The actors in collaborative scenarios have to perform complex tasks. If these tasks are executed by multiple actors in a more or less predefined manner, this is often called a business process. Business processes describe what has to be collaboratively done and when and how this is accomplished. Various WfMS exists which support business processes, i.e. their definition, design, and enactment. However, regardless of what form of collaboration we consider (direct or distributed, within an organization or across its boundaries), in a collaboration scenario the question ‘Who has to carry out a task within the process?’ has to be dealt with, as well.

Thus, similar to other approaches of collaboration modeling (e.g. [4], [7] or [17]), GroupOrga clearly distinguishes the aspects what, when, how from the aspect who, or in other words, between process modeling and organization modeling. The latter aspect—on which chapter 3 focuses more deeply—specifies the actors, who are responsible and qualified to carry out tasks within the collaborative setting.

For a smooth interaction of workflow systems and the necessary organization modeling environments, a definition of arbitrary, problem-oriented actor assignment strategies is mandatory. Workflow modeling and organization modeling aim at different goals. The first is concerned with the flow of work. The aspect who is also of interest eventually but cannot be coped with by process modeling experts, since they do not have sufficient knowledge about the organization and how to depict it in enterprise directories. The necessary link (or better: integration) between the two sides is done through organizational references saying who has to perform which work (cp. [14], p. 563).

Investigations of collaborative software in general, and WfMS in particular (cp. [12], [13], [9], [1], and others), reveal that none (e.g. [18], [3], [5]), or only very few, approaches pursue the design and documentation of structural organization—nevertheless, its necessity as a mandatory task for WfMS is acknowledged ([11], [12]).

GroupOrga will bridge this gap between WfMSs and organization systems. It will provide a technological environment for various requirements for collaboration in order to obtain fast, coordinated structural design with the support of enterprise directories that transcend intraorganizational boundaries.

3 Varying Types of Users in an Organization

The GroupOrga concept involves potentially any employee in this organizational design process, so that participative organization design will be carried out through strongly diverging user classes in the organization. The varying requirements of the user classes result in a scale of possible user types which are illustrated in Table 3-1. In other words, the involvement ranges from employees who only want to be informed to those who actively and regularly participate in the design. The scale which will be presented explains the varying requirements of different user types in an organization.

More thoroughly than done in [14] (p. 568), this paper addresses the requirements of the different users and presents technological solutions for their requirements. It refines the classification of users into five types, each with six attributes.

Section 3.1.1 starts with the requirements-profile of full-time organizational designers. Afterwards, the requirements of those users, who are less strongly involved in the organizational design process will be described. Sections 3.1.2 to 3.1.5 focus on users who regularly modify the organizational structure, users who occasionally adapt the model, users who administer their personal data, and users who have only read access. A concluding section, 3.1.6, compares the various user classes and comments on their involvement under certain circumstances, such as organizational sizes and types.

3.1 User Classes of Applications for Organizational Design

3.1.1 The User Class "Strategic Changes"

Users of the class "strategic changes" (see Table 3-1) are those who are responsible for the organizational top-level design. Such design forms a framework for a later, detailed structuring of subordinated levels. Such a design is also referred to as top-level design (cp. [16], p. 5). Depending on the overall size of an organization, this user class is represented by a single person or by a group of
employees—in any case a very small number of employees in relation to their total number. The intensity of use (the depth or profoundness of modifications to the structural model) is considerably high, while, in relation to the other user classes to be discussed later, frequency of use of organizational design applications is rather low (cp. "Frequency of use" in Table 3-1).

The typical tasks of organizers of this user class are far-reaching changes in the global organizational structure, for instance initiated by a change in the core processes of the organization. If an organization is newly established, a completely new design from scratch may become necessary. Concrete examples of design tasks are, for instance, to move whole sub-units within the organization, or to flatten organizational structures. In case of the modeling of a virtual organization such an organizer may have to integrate the various individual organizational (sub-)models into one. Hence, the typical focus of modeling for members of this user class is the hierarchical organizational model—an organizer of this class will most likely not modify bottom-level workgroup membership, particular role assignments, or individual job descriptions.

From this scenario it can be concluded that an organizer of this class intensively uses the organizational design applications and tools, which results in very specific and detailed requirements for the tools.

### Table 3-1: GroupOrga scale of varying tool-centered requirements by different user type classes

<table>
<thead>
<tr>
<th>User class</th>
<th>Read access only</th>
<th>Administration of one's own personal or organizational data</th>
<th>Occasional adaptations</th>
<th>Regular changes</th>
<th>Strategic changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type and intensity of use</td>
<td>Push-button information needs</td>
<td>Administration of one's own personal or organizational data</td>
<td>Occasional changes or adaptations within a units</td>
<td>Regular departmental design and planning across units</td>
<td>Regular design from scratch, design, planning analysis, reporting</td>
</tr>
<tr>
<td>Managerial level</td>
<td>Independent from levels of management</td>
<td>Without management tasks</td>
<td>Lower-level management</td>
<td>Tactical management</td>
<td>Strategic management</td>
</tr>
<tr>
<td>Type of user</td>
<td>Enduser</td>
<td>Administrator</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Relative proportion of employees</td>
<td>High</td>
<td>Very low</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Intensity of use</td>
<td>Low</td>
<td>High</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Frequency of use</td>
<td>Low</td>
<td>High</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Typical object of organizational modeling</td>
<td>-</td>
<td>Person, Skill</td>
<td>Role, Position, Workgroup</td>
<td>Hierarchical submodel, units, authorizations</td>
<td>Hierarchical model</td>
</tr>
</tbody>
</table>

### 3.1.2 The User Class "Regular Changes"

When considering management levels, the members of the user class "regular changes" may be classified as tactical, mid-level management (see Table 3-1). Members of this class deal with the design of single units in the organizational structure and with their subordinated units. Hence, the sub-model of all subordinated units (i.e. the unit tree) is the focus of this design and responsibility. The task of employees in this class is regular departmental design and planning which may spread across a department's or unit's borders. Since modification and planning mainly concentrates on single units rather than on complete organizational structures, the intensity of use of organizational design applications will decrease, while the frequency of use will increase.

The tasks of units or managers in middle management include the creation of new (sub-)units when the scope of responsibility is increased, or the amalgamation of several units into one in the course of BPR projects. Other tasks may include a flattening of organizational sub-structures, the definition of new positions, or the integration of new employees into the structure. Moreover, managers at this level may have to assign authorizations and competencies to their employees, such as access rights to information and resources or rights in the organizational design process itself. These examples explain why Table 3-1 indicates an increasing frequency of use of organizational design tools for this user class.
The distinction between this user class and the following is rather fluid and differs according to the organizational culture, as well. Some tasks may be subsumed to this class, as well as to the next one of "occasional adaptations". However, other characteristics clearly distinguish the two classes as the next section shows.

3.1.3 The User Class "Occasional Adaptations"

Members of an organization, who belong to the user class "occasional adaptations" (see Table 3-1) most likely belong to lower-level management, often entitled operative management. The leadership of a manager of this class directly addresses the members of a single unit which in turn has generally no further subordinated units. The number of subordinated employees, that is, the span of control varies immensely. The literature mentions a span of control between three and ten employees in normal situations, while other tasks may require a span of control of ninety staff.

The average span of control influences the number of managerial employees who have responsibility over others. In case of larger control spans, their number decreases while with a smaller span of control the number increases, respectively. Accordingly, the number of hierarchical levels will decrease or increase, which results in flatter or steeper organizational structure. Thus, the number of employees in the operative management level is considerably higher than in those management levels already discussed. Table 3-1 shows this connection in terms of the relative proportion of employees of the organization.

According to this scenario, organizational design activities of this user class belong to the category of bottom-level design ([16], p.5). Changes made to the organizational model do not affect other organizational entities (units, workgroups), but are restricted to their own organizational unit and happen within their own organizational borders.

The intensity of use of organizational design application decreases, since only their own organizational entities are modified. Similarly, the frequency of use of design tools generally decreases for the same reasons. In specific cases, for instance in organizations which strongly focus on workgroup structures, frequency of use of organizational modeling tools may increase, since workgroups do cross hierarchical borders and frequent remodeling may be necessary.

A clear distinction between this category and the category "regular changes" can be made when considering internal organizational borders: while users of the class "regular changes" will design organizational structures across such borders (i.e. spanning more than one organizational unit), this is not the case in the class of "occasional adaptations". In contrast, the design of workgroups has to be assigned to both categories as Table 3-1 indicates. The workgroup concept as such explains why: a workgroup does not belong to a particular organizational level and it does not belong to a specific unit—it is thus spanning organizational borders, which classifies the workgroup design as a tactical management task. On the other hand, the way tasks in workgroups are accomplished (spontaneously, flexibly) would categorize the design of workgroups into operative management.

Due to their flexibility, the design of workgroups cannot separately be assigned to several management levels or user classes. The highest degree of design activities for workgroups will be found in operative and tactical management, which is why the workgroup icon in Table 3-1 has been positioned between the two categories.

Examples of concrete tasks of employees in the "occasional adaptations" class are to modify authorizations and rights of specific employees, to define new roles, to assign roles to employees, and to create, modify, or abolish positions. The assignment of actual employees to organizational positions is another task in this category, as is the creation and forming of workgroups.

3.1.4 User Class "Administration of One's Own Data"

This category comprises employees with no specific managerial responsibilities. Although, these employees have no specific organizational design responsibilities per se, the GroupOrga concepts also involves them in the organizational design process. It has been shown that the complexity of design tasks can be reduced if everybody is involved in the modeling process "self-responsibly". This requirement can be reached, if users of this class administer their personal organizational data independently. This may, for instance, include address data, telephone- and fax-number, email-address, etc. In the ideal case, the employee also updates this data in times of unavailability, such as before a business trip. Another example is the unavailability of an employee due to vacations, work at another location, or illness. In these cases, the employee would have defined a delegation or substitution regulation himself.

A further aspect of "administration of one's own data" is the skills, qualifications, and knowledge of employees. Such skills comprise seminars attended, education, certifications, language skills, knowledge about programming languages, etc. If these skills are stored and managed in an enterprise knowledge base, the employees can be assigned to tasks according to their skills.

The amount of data that has to be covered by this user class seems to be tremendous for the present, but in general this information will change rarely. In addition, the changes and modifications are carried out by a large
number of employees, which reduces the workload of the individual worker significantly. Thus, a low frequency of use of organizational design applications can be assumed, going along with low intensity of use, since only personal data will be affected. Another reason for low intensity of use is the fact that modifications do not affect entities beyond one's own personal data, such as unit, role, position, or workgroup entities.

While the former three classes of users implicitly describe a subset of organizational members according to their managerial level, this user class mainly addresses those at the base of an organization, but in general it comprises all organizational employees from top to bottom who administer their personal data. Table 3-1 indicates this with a high relative proportion of employees of the organization.

3.1.5 The User Class "Read Access Only"

So far, every involvement in the organizational design included modification and active design. This last user class describes those whose accesses to the organizational database which are read-only. Any type of organizational information may be necessary and requested in various scenarios: information about organizational units and hierarchies, workgroup composition, membership in units or workgroups, skills of employees, and so on. Such information may be requested from persons internal and external to the organization.

In case of reading information in the database, no valid assumption can be made about the frequency of use. To this point, the frequency of use was considered for active design tasks–this user class performs lookups only and performs no modifications. Such read-only access does not correspond with the management level and it does not directly depend on an employee's involvement in organizational design activities elsewhere. Hence, the frequency of use for read-only access can be considered constant across all organizational levels and may be added to those uses which are due to active design. The intensity of use can also hardly be compared to the other user classes, since no active modeling activities are performed. It would be zero in this case of "read access only" users.

3.1.6 General Considerations about the User Classes

All statements made in the previous sections may vary according to organizational size, culture, and type. Organizations to which the concepts apply have to have a minimum size in order to show the characteristics discussed, such as different management levels and clearly distinguished tasks. However, medium-sized organizations are also covered by GroupOrga concepts. For smaller organizations some classes of user types may be not applicable, specifically those of mid-level management ("regular changes"). A similar context exists for flat, decentralized, or virtual organizations, since in these forms the design activities may be more strongly polarized. Moreover, this type of organization requires less organizational design at all.

Other examples of differing organizational types are those that are strongly structured, conservative, and mostly hierarchical. Such organizations may have a large share of manufacturing workplaces, which implies that the category "administration of one's own data" may drop out due to their current state of IT infrastructure. On the contrary, organizations that focus mainly or only on workgroups have no traditional units and thus no hierarchical structure. However, the statements about the different types of user classes still hold true, since there will be users who intensively use the design tools and others who do not. In case of very large organizations or trusts, one will be able to identify even more managerial levels which implies that the categories of Table 3-1 may be refined. The continuum described here presents a basic form, covering the most likely and common user classes.

3.2 A Variety of Supportive Tools for User-specific Modeling

Although an organization design process can only be successful, if all affected parties can take part in the design, the following list—which can never be complete, but is significant–shows that there are technological barriers to participative organizational design which GroupOrga aims to resolve:

- High royalties for specific software of organization design prevent widespread use of these tools and thus hinders their organization-wide availability.
- Access to the enterprise models and their modification often necessitates highly specialized software and hence accordingly qualified and entitled employees. High costs for training prohibit qualifying a sufficient number of employees.
- An enterprise-wide provision with the organizational models fails because of improper data storage. Often these specifications cannot be exchanged due to a lack of standardized interfaces. Instead, the models are hidden in complex database management systems.
- Largely distributed organizations (have to) support a large number of different operating systems, desktop software, and end user applications. Moreover, a large variety of hardware platforms can be found, which may have built up over years and still have to be maintained.
- In very large organizations or in alliances, various organization design systems will already be in existence, consisting of disintegrated organizational databases or incompatible graphical design tools. This
situation is defeating any effort of setting up a distributed design process across the partner's borders.

The following section will explain in greater detail how these hard- and software-born restrictions have been eliminated in GroupOrga. While one would conjecture that the featuring of Web/Java-based tools is not unique for GroupOrga, the introduction of a combination of the underlying groupware technology and the technologies that can be summarized under the WWW certainly is. It will be pointed out that different users' needs have to be fulfilled with various types of applications on different sorts of platforms. Hence, a platform independent modeling is the long-term aim of GroupOrga.

3.2.1 Platform Requirements and Implementations

Important arguments for implementing the GroupOrga project on top of the chosen groupware platform have been the necessity for distribution technology (e.g., replication), high security standards, distributed database architectures and the fact that many WfMS are also groupware applications. While these reasons remain valid, the following arguments present an extension—not a replacement—to it. They provide a list of concepts and implementations in the project to ensure platform independence and thus a solution to the identified technological problems. In addition to highly specialized applications, several GroupOrga tool areas have also been covered with WWW and Java technology which brings about applications, such as a graphical organization modeler (see Figure 3-1), that run on virtually every hardware and operating system. Thus the extremely important need for platform independence is accomplished for all user types.

![Figure 3-1: The GroupOrga OrganizationModeler as Web/Java Application](image-url)
Organization design by all organizational members

The participation in the design process of every organization member implies that this task can be fulfilled from every computerized workplace. However, in most existing organizations, heterogeneous hardware technology and operating systems can be found, as section 3.2 has suggested. This aspect becomes even more valid in scenarios of virtual organizations with cooperating partners. While some partners may have internal standards for their organization design technology, this cannot be assumed for all parts of such a trust.

Java technology stands out from other technologies of this kind by its ability to produce programs for every platform (write once, run anywhere). With this technology as a source for GroupOrga applications:

- It is not necessary to recompile the software developed for all supported or thinkable system platforms. Hence, no knowledge about the respective hardware platforms is required to translate the source code.
- The software no longer needs to be distributed in dedicated versions to the users of the respective platforms. In addition, it is no longer required to explicitly establish distribution channels for the software over and over again.

For platform independent GroupOrga applications the source code is translated into bytecode, which is then executable on every platform that supports the so called Java Virtual Machine (JVM). The existence of JVM together with the GroupOrga applications reverses the present situation: The system environment is adapted to a uniform software, rather than adapting a software package to every platform.

In case a new platform is invented, a new version of an operating system is released or if a new partner with its own hardware joins the organization design environment, the GroupOrga tools can run on it without change. Thus they are future-oriented.

For those users who fall into the user classes of Read-access only to Occasional changes or adaptations (see Table 3-1) yet another advantage can be gained with this form of software implementation: Now the GroupOrga applications can be run in form of Java-applets in Web browsers. In the field of end user applications the Internet, in combination with Web browsers as front-end applications, serves as the largest conforming basis for software applications in order to reach a large and multi-layered group of users.

Short preparation-time

A substantial goal for today's organizations is to be capable of acting according to changed environmental circumstances without long delay. An application environment for the design and modification of organizational infrastructures should also meet this requirement, i.e. to be ready to use after a very short time of preparation. From an IT point of view, this implies that the effort for installing the software should be cut to a minimum and that training for the mass-user should ideally be unnecessary because of the simple design of the tools.

Being based on Java technology some of the GroupOrga tools can now be used in a Web browser. Hence, given the installation on a WWW server, the effort for the end user to deploy the organization design tools is comparably low. The only step which has to be done is to invoke the modeling tool via its address on the server, which may actually be already accessible via the organization's homepage. From then on, the modeling tool will be downloaded and can directly be used without any extra effort.

Low servicing expenditures

Users of the first two classes of the scale in Table 3-1 do not use their modeling software very often. Servicing, in terms of software packages, generally requires periodically installing new versions and distributing fixes and patches. This procedure of installing new software on many heterogeneous computers in an organization is time-consuming and expensive. Especially, for these first two user classes the additional benefit from installing new software stands in large contrast to the few advantages gained.

With GroupOrga's WWW- and Java-based tools for organization design, the costs for installation are drastically reduced. Their architecture allows to immediately distribute any changes undertaken to the end user, no matter how often this user actually employs the application, how many users are affected and what the cost/benefit situation would be in the different cases. This situation now generates a breakthrough in bottom-level design participation, since every user on every platform can be equipped with modeling tools.

Independence from a particular workplace

Another characteristic of dynamic organizations is, that design teams are immediately made up for a short period of time. Employees in such project-oriented organizational forms cannot assume to have a stable working environment over a long time. Instead, their workplace may change within the borders of their location and a project-driven change to a completely different location may also be unavoidable. In connection
4 Barriers to Distributed Organization Design

Despite the technological advantages of distributed organization design by users of all different types, organizational barriers to distributed and participative organization design can be found. Such participative design often involves global, distributed directories and, as our own experience from practice shows, barriers to implementing such distributed organizational repositories include organizational politics, privacy concerns, data cleansing, and multivendor interoperability. While some of these barriers appear to be technical (especially 4.3 and 4.4), they did intentionally not feature in section 3.2. Here, the technical problems have to be solved with organizational involvement and discussion, rather than with pure technology, such as Java, for instance.

4.1 Organizational Politics — Or: Hands Off my Data!

Some kinds of organizational data, such as user names and network access lists, already exist in application- and network operating system-specific directories. Mostly, these directories cover a fraction of the whole organizational structure’s information of an organization. The currently realized approach of distributed administration of structural information is a very sensible one. It allows the organizational members, who work at the core of the organization and who know the organizational details, to edit and change the structural information they need and to introduce always up-to-date data. Hence, this concept is highly advocated in the GroupOrga project.

However, while these advantages of distributed administration of structural data exist, distribution bears problems.

Typically, the aforementioned organizational data may be brought into a global distributed directory from existing specific directories. Alternatively, several directories may be integrated and mended together into an apparently seamless whole without moving data. Either way, as soon as a project for distributed modeling lays its hand on someone’s data, this is a matter of organizational politics.

Payroll and human resource departments are not accustomed to offering their data as a source for distributed design of organizational structure. As an example from our cooperation with a large, global, German-based business bank: If through some bureaucratic process the human resources’ database would be chosen as the center for a global enterprise directory and if then this database should be modified to suddenly store IP and email addresses or availability information, as well, the human resources responsible would not stand for this change. This single example shows, that organizational politics may derail any participative organizational design project unless a powerful champion in the organization can be found. In case of our cooperation project such a change agent or change promoter from the bank’s higher management had indeed been found to foster the trial implementation. Otherwise, it seems to be nearly impossible to do it from the bottom up.

A solution may be to set up a system where the central repository only reads information from other directories and combines them to one large directory. Any changes concerning information in the departments are to be undertaken in the departments only, rather than at a central unit unless there is a serious necessity for such a central line of action. This is usually ideal, because a central repository does not have to write to other directories due to the corresponding organizational regulation. Replication in real time between directories is possible and allows for fast and seamless distribution of changed information. Nevertheless, political issues and bureaucracy seem to remain major impediments to the deployment of distributed directories.

4.2 The Privacy Concern

When information is promoted from limited-access systems such as payroll or human resources to a much more widely accessible, participative directory (e.g. through replication technology to other locations), privacy will become an issue. Hence, certain information may need to be protected by password and finely tuned access rights and certificates. Integrating such strong security may complicate the directories usability but also makes it suitable for a wider range of information retrieval, as well as workflow and office management applications. In addition, searches should be reasonably limited which for two reasons: The first is the fact that unwanted phone calls and emails have to be dealt with, the other is that people do not want to download the whole directory into their mailing lists. On the other side of the coin, this search capability is the advantage a global directory offers in terms of workflow definition. Now the workflow designer can easily identify employees with...
certain skills and knowledge or with some extra time available to take part in a workflow procedure. While formerly design information was limited to the designer’s own (locally limited) know-how he/she can now fall back on the vast knowledge of the global enterprise directory and search for and select the appropriate actors within a much wider scope of people, roles, locations and skills.

4.3 Cleaning up the Mess

Each of those aforementioned distributed repositories is likely to contain entries that are improperly formatted, not detailed enough, missing, duplicated, or outdated. Such bad structural information may cause no harm where it is currently stored, however, when replicated into a global enterprise directory a thorough checking and clean-up becomes unavoidable. The problem is multiplied by the fact that data from each source will be unsuitable in its unique way.

A related problem includes “good” structural information that is formatted differently in different databases. Apparently, the obstacle of different data schemata is not only a problem of multivendor systems, as the following section will propose, but also one of different departments in an organization.

Our cooperation with a hardware and software producer showed, that an uncountable number of directories existed in the company's intranet in 1996. An attempt to install a company wide directory by simply linking the different departmental directories via intranet technology was soon abandoned and not resumed until the varying data structures were cleaned up.

4.4 Multivendor Systems at the Start

To integrate or synchronize directories from multiple vendors, three issues have to be addressed: standards, architecture, and schemata.

Great steps forward have been made concerning standards since April 1996, when 40 vendors made the first major announcement about LDAP, the Lightweight Directory Access Protocol. Since then, LDAP has been almost universally endorsed. However, despite LDAP, interoperability of directories is still in the early stages: Infrastructure vendors have to implement LDAP, operating systems have to support these infrastructures, developers have to use the infrastructures, and users then have to deploy the resulting applications. Replication between infrastructures of multiple vendors is still far away.

In terms of architecture the most basic question is whether or not to have a physical central repository, as opposed to a virtual directory. In the latter case multiple directories synchronize with each other, but there is no central repository as such. Such a directory refers a query to other directories until the correct result can be given. The first case includes a central repository which contains all that data from all distributed directories which is considered important to other directories, as well. Any other structural information is only stored in the distributed directories and is not replicated.

To read from and write to distributed directories, the synchronization operation must understand the different directory schemata which define organizational directory entities and their respective attributes. To give a very simple example, such common schemata are designed to provide a minimum definition of the attributes of a “person” entity, such as email address, name, knowledge and job position, etc. In other words, these schemata perform an invaluable function in telling client software (such as WfM software) what to look for in the directory. In this context it must be noted that, however many attributes are defined in such a schemata, they may be more than enough for one typical application and certainly not enough for every other conceivable application. To determine what schemata are sufficient in terms of generalness is a very complex task.

5 Summary — What is coming?

In this paper we have presented a refinement of the GroupOrga concept and framework for distributed, dynamic organizational design processes. It provides a user-friendly, collaborative framework for the specification and execution of organization structure entities for WfMS. User-friendly graphic interfaces and the comprehensive enterprise model have been taken into account. Implementation on the WWW has also been outlined and discussed. Finally, we outlined some social and political impediments to collaboration technology in organizations.

This paper intended to show how we can integrate collaborative and organizational requirements with today’s technological possibilities to successfully construct collaborative applications. With a focus on groupware technology and the World Wide Web we have introduced two technology fields that appear to fit perfectly with collaboration requirements.

Even though few companies can implement such distributed organizational design processes today, forward-looking users have grasped the idea of directory and application integration and are heading this way. GroupOrga has been implemented in the form of evaluation installations in various practical settings and conditions: Among others are cooperations with KPMG, Siemens Nixdorf Informationssysteme AG, Deutsche Bank AG, Deutsche Babcock Dienstleistungs GmbH, and agens Consulting GmbH. Moreover, parts of the GroupOrga toolset are integrated in an office and
workflow management application offered by Pavone Informationssysteme GmbH.

In the future, companies may want directories to deal with (technical) network issues and manage a broader range of services. For example, global directories could be used to manage e-mail systems and Web services. Although, using a global directory to lookup e-mail addresses, phone numbers and room locations is advantageous, the big payoff of global enterprise directories will be more of a single point of administration for all types of distributed applications and not so much in terms of a user and access control database for network administration.

The global enterprise directory must become more of a tool and information pool for everybody involved in business processes. It must allow for easy, scaleable administration from every desktop and simple integration with existing business information systems and workflow applications.

Acknowledgments We wish to thank the appointed referees for their very valuable and helpful comments

References