

Open CSCW Systems: Will ODP help?

Leandro Navarro ^a, Wolfgang Prinz ^b, Tom Rodden ^c

^a Universitat Politècnica de Catalunya, Barcelona, Spain.

^b GMD, Schloß Birlinghoven, 5202 St Augustin, Germany

^c Computing Department, Lancaster University, Lancaster LA1 4YR, U.K.

Abstract

This paper is concerned with the role of Open Distributed Processing systems within the development of cooperative systems.

In particular, we focus on the need to provide Open CSCW systems and their impact on distributed systems. The work currently being undertaken in Open Distributed Systems (ODP) is used to highlight significant trends for future open CSCW systems. It will be shown that the CSCW and ODP community share mutual interests and have complementary aims and goals developed from different perspectives.

The paper introduces CSCW highlighting the requirements CSCW places on distributed systems. The development of an environment to support open CSCW systems is introduced and briefly described. Finally, the relationships between requirements and models for Open CSCW systems and the Basic Reference Model of ODP are discussed.

1. Introduction

The term Computer Support for Cooperative Work (CSCW) has rapidly emerged as an identifiable research area over the last six years and has undergone dramatic growth in this time. CSCW applications are designed to support the work of groups. The research undertaken by the CSCW community combines a wide variety of disciplines each contributing a different set of skills. Similarly, the various systems which have emerged to date have adopted a multiplicity of approaches and techniques to support multiple group activities.

Distributed systems play a central role in many of these systems and it is likely that CSCW will provide a significant application area for future distributed systems [1]. The needs of CSCW systems will most likely play a significant role in shaping many features of future distributed systems. It is therefore important that future distributed systems developers are aware of not only the requirements of computing and communications resources,

but also the requirements of people and their organisations (CSCW).

Many existing CSCW systems have tended to ignore the existence of other systems and styles of support. In the same way that user isolation limited the usefulness of personal computers, it is important that CSCW developers avoid a blinkered outlook in the development of their systems and attempt to develop "open" CSCW systems. Because people is usually embedded in organisational structures and social norms, CSCW applications should be embedded in an Environment providing common support for dealing with people, organisations and norms. The result are Open CSCW Systems.

The distributed computing community have been examining the provision of open systems for some time and much of this work is now reflected in the Basic Reference Model of ODP [18].

We believe that both the CSCW and ODP community share mutual interests and have similar aims and goals which have emerged from different perspectives. ODP standardisation considers the portability, interworking and distribution problems of distributed computer systems. In contrast, CSCW has a focal point on the use of computers to support groupwork which need to be open to both users and other systems.

2. CSCW Systems

The term CSCW was originally coined by Greif and Cashman in 1984 [2] as a shorthand way of referring to the interests of a number of researchers involved in the use of computers to support user groups. The area has evolved over the last five years to combine the understanding of the nature of group working with the enabling technologies of computer networking, systems support and applications. It is now recognised that CSCW is inherently a multi-disciplinary research topic and requires the application of a number of disciplines including sociology, organisational science, psychology and computer science. Readers are referred to [3, 4, 16] for a more complete review of CSCW research.

The wide variety of CSCW systems developed to date reflect the many different views of cooperation which currently exist within CSCW. Two principal characteristics are used to describe CSCW systems:-

i) The form of interaction supported

CSCW systems are primarily concerned with supporting a number of users cooperating to address a particular problem, or range of problems. The nature of this cooperation can be distinguished by the way in which the group members interact. People either interact and cooperate *synchronously* or *asynchronously*. Synchronous interaction requires the presence of all cooperating users while asynchronous cooperation occurs over a longer time period and does not require the simultaneous interaction of all users. Synchronous systems are characterised by desktop conferencing systems [5] such as Shared X [6]. The majority of Asynchronous system are based around either message systems [7,8] or computer conferencing systems [9].

ii) The geographical nature of the system

CSCW has traditionally considered the case of geographically distributed groups. More recent research has complemented this emphasis by considering the support of face to face (or co-located) meetings. As a result cooperative systems are often considered as being either *remote* or *co-located* giving our second classification mechanism. In this classification the division between remote and co-located is as much a *logical* concept as a physical measure and is concerned with the accessibility of users to each other rather than their physical proximity. Co-located systems often exploit purpose built meeting rooms such as the COLAB[10] at Xerox Parc. Co-located systems include more sophisticated distributed applications such as multimedia conferencing systems [11].

	Same Time	Different Times
Same Place	Face to Face Interaction	Asynchronous Interaction
Different Places	Synchronous Distributed Interaction	Asynchronous Distributed Interaction

Figure 1 The groupware time space matrix

The two characteristics of interaction and geographical location of users are often used as the basis of a simple classification space for CSCW systems (figure 1). This space allows the technical characteristics of the various classes of system (or groupware) within CSCW to be

categorised and is often referred to as the groupware time space matrix [12].

3. The need for open CSCW systems

The current generation of CSCW applications provide diverse models and mechanisms aimed at supporting either a particular "artificial" cooperative activity or class of activities. These applications are often unaware of the existence of other applications [13] and provide few mechanisms for working in conjunction with other applications (figure 2). Thus, users of a particular CSCW application are presented with a particular interpretation of cooperative work and can only work within the confines of that closed world.

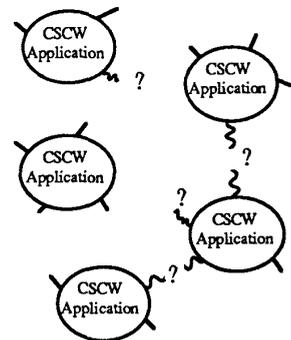


Figure 2. Independent CSCW applications

The reality of supporting cooperative work is that a wide range of CSCW applications, each exhibiting a distinctive model of cooperation, need to work in unison. Consequentially, the role of the environment in which CSCW systems exist becomes a crucial factor for the future success of CSCW applications.

A central aim of such environment is to provide *interoperability between a variety of applications* ensuring that CSCW applications can work in harmony rather than in isolation of each other (figure 3).

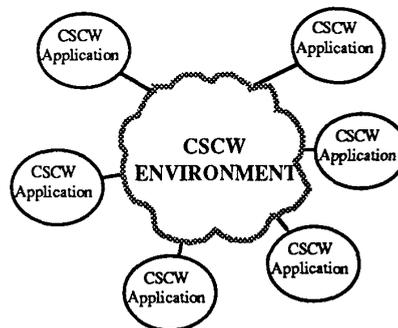


Figure 3. The role of a CSCW environment

We see the provision of a CSCW environment as a means of realising more "open" cooperative systems. As CSCW applications become more widely available, it will become increasingly necessary to focus on an open approach. Open CSCW systems should aim to *allow a multiplicity of approaches and paradigms* to co-exist. A central aim of these systems, and as a result of a CSCW environment, is to enable the co-existence of:

- remote/local cooperation.
- personal /group working.
- synchronous/asynchronous working.

In addition, the problems of developing open CSCW systems are compounded by the strong *inter-relationship between CSCW systems and the activities of the organisation* in which they exist. CSCW applications do not exist in isolation but are an integral part of the activities which take place within the organisation they are embedded.

For example, the management of a large scale engineering project (e.g. building the Channel Tunnel) can be undertaken as a cooperative activity. The overall task may involve an on-going programme of sub-activities such as team progress meetings, the joint production of reports, monitoring and interviews as well as more ad-hoc, informal communication between project members.

These sub-activities can be inter-related in a variety of ways, for example:

- Each person may be involved in many activities.
- Activities may use common resources
- Activities may share common information.
- Activities can have well-defined temporal relationships.

There are also many differences between activities. For example, some have well defined goals and fixed deadlines while others are on-going. They may also utilise different underlying communication technologies. Activities do not occur in isolation, rather, the general picture is of many inter-related activities taking place within a world of shared resources, people and information.

As a result, cooperative working needs to be considered in terms of *numerous related activities occurring within an organisational environment*. It is important that a CSCW environment allows these activities to be inter-related. This may be achieved by the provision of common functions by the environment, while applications concentrate on task-specific functions. However it should be stressed that we do not foresee the development of open CSCW systems allowing a standard model of either cooperation or organisations. In contrast, open CSCW systems will allow a greater diversity of cooperation techniques to work together.

4. Open CSCW systems requirements

The identification of a range of requirements for a CSCW application environment provides the starting point for our work on open CSCW systems. The requirements¹ presented here are by no means complete but are intended to illustrate requirements which are characteristics and distinctive of CSCW systems. We believe that they will directly impact future distributed systems.

Support for Information Sharing

The sharing of information is an essential precursor to cooperative working. It is important that patterns of sharing are adopted within the environment which enable effective cooperation to take place. The environment needs to provide a set of services which encourage the cooperative sharing of information. These services should include:

- maintaining a knowledge base of people, resources and on-going activities
- mechanisms for modelling organisations.
- appropriate access control mechanisms. (Traditionally, roles have been used to signify different access rights of users.)
- services for the access and exchange of information between CSCW and non-CSCW applications
- Support for the distribution of information across an number of machines over different media; including where applicable portable/personal computers.
- Smooth integration and utilization of standard information repositories, for example, the X.500 directory service [14].

Support for Communication

Communication plays a vital role in cooperation and it is important that this role is reflected within a CSCW environment. Subsequently, a CSCW environment will need to provide a range of communication services which should include:

- support for a wide range of media, including telefax and where applicable paper communication.

¹ MOCCA is a working group partly funded by the European Community within the framework of the Cost-14 CoTech programme. Many of these requirements were either informed or confirmed by an international developers workshop [17] held as part of ECSCW'91, the second European conference of CSCW, and attended by over 40 developers.

- The provision of many different forms of communication, including both real-time and asynchronous communication.
- Support for interchange across communication media.

Traditionally, communication support for CSCW systems has been provided by asynchronous OSI communication standards such as X.400. However, even though these standards provide a useful basis they don't allow a sufficiently diverse range of communication styles for many CSCW systems. Accordingly most CSCW applications adopt and augment these basic services for their own purposes [7,8].

Support for Activities

CSCW systems have a strong relationship with the various organisational activities which they support. It is essential that a CSCW environment provides mechanisms for representing the various *relationships* between these activities. Additionally the environment needs to provide a set of services to allow the *management* of these different activities within the environment. These services might include:-

- managing the membership of activities
- sharing resources between activities
- scheduling activities and monitoring the progress of activities
- mechanisms for negotiating the responsibility for activities
- mechanisms for negotiating the division of competence within activities
- coordination of activities

These services need to be provided in a neutral manner in order to allow as wide a range of CSCW systems as possible to be supported by the environment. No single model of cooperation is assumed but rather services are provided to allow a range of different models to exist in unison.

Support for Tailorability

Cooperative working is essentially a dynamic activity and consequentially CSCW systems need be malleable and tailorable. As a result, systems and the environment need to be tailorable both by developers and users. This has two important consequences, firstly, the environment need to provide a set of services akin to a developers toolkit to enable this tailorability. A more important consequence, perhaps, is that the traditional *divide between users and developers* becomes less clear with users having similar powers and status as system developers. An investigation of the limits and bounds of this tailorability and possible

notations, languages, or services to support this tailorability will be an important area of research for future CSCW developers.

Support for Transparency

Cooperative activities are carried out by a "Distributed Group": people who are located at possibly different places, employed at different organisations, working at different times, using different user-group interfaces, having slightly different goals, having different understandings of the activities, language, competence, culture, etc.

The CSCW environment should provide some *degree of transparency* to facilitate people cooperating from different coordinates, to hide some dimensions that are unnecessary for a cooperative activity and makes the system look more complex. Dimensions are: organisational (*organisational awareness*), temporal (*integration of synchronous and asynchronous*), linguistic, cultural, physical units (*distributed working*), etc [15].

A number of different forms of transparency are important to CSCW Systems.

Transparency of organisation means that activities need not dealing with the complexity of the possibly different organisations involved. Inter-organisational connections should/could hide the complexity of different organisational (particular of each enterprise) and inter-organisational (free market or other) policies. Sometimes, interaction is not possible due to incompatible policies (or cost too high).

Transparency of time deals with the mode of work, synchronous or asynchronous. The result of applying this transparency is that interaction will be independent of the mode we are using.

Transparency of view means that applications can be interested or not in the way users view data. WYSIWIS applications will not use this mechanism.

Transparency of activity means that a set of objects cooperating in one activity needs neither be aware of the mechanisms for starting and coordinating activities, nor be aware of other unrelated objects or activities present in the distributed environment. These unrelated objects can be located in a different location or in the same location but participating in other activities. This helps activities not to be disturbed by other unrelated activities.

This section has highlighted a number of requirements which are central to the development of a CSCW environment. In order to further investigate these requirements and to develop mechanisms by which these requirements can be met the MOCCA project has adopted

a model based approach to their work. The aim of the Mocca work is to identify and describe a set of models which will act as a reference for future developers of a CSCW environment.

5 Models to support open CSCW

We are currently in the process of identifying and populating a number of models which are intended to inform the development of future CSCW systems. The express aim of these is to develop a set of recommendations for characterising cooperative systems. *This set of models that compose the environment model will be used to determine the common functionality required (i.e. the set of functions/services provided) to support CSCW systems and allow interaction across a number of different CSCW systems.* We have identified five models which we believe are important to the development of future open CSCW systems.

The Organisational Model

A central motivation for the development of open CSCW systems and the Mocca project is the realisation that organisational context is crucial to the success of CSCW systems. The aim of the organisational model is to make explicit the sharing of organisational resources, policies and regulations. The model is constructed from a set of organisational objects (e.g. resources, projects, people, roles), organisational relations and rules.

The Inter-activity Model

The notion of activities and roles support a variety of CSCW systems. A number of different techniques are used to represent activities and roles within different CSCW systems. Consequentially, each systems has a different perspective of both. Rather than finding a common mechanisms for representing activities and roles the aim of the inter-activity model is to allow the dependencies between different activities and roles to be represented within the environment.

The Information Model

Effective sharing of information is central to cooperation and the role of information within cooperation is crucial. However, each CSCW systems has a different perception of information and sharing. The Mocca information model aims to allow information used within different CSCW systems to be represented externally and to be shared between systems. The model is expressed in terms of information objects, the relationships between these objects (e.g. composition, dependencies) and the access to these objects.

The Communication Model

Communication is an integral part of cooperation and the support and representation of different patterns of communication is central to a number of different CSCW systems. In particular communication plays a vital role in message based systems. The communication model aims to represents communication in terms of the communicators, the information objects they exchange, and the context within which communication takes place.

The User Expertise Model

The representation of users and user expertise are a central feature of cooperative systems. The aim of the user expertise model is to represent the users expertise for use by the environment and other systems. This models is expressed in terms of *user's responsibility*, which is imposed by the organisation and *user's capabilities*, which describes the users individual skills.

It is hoped that our investigation into these models will allow a greater realisation of what the development of open CSCW systems will entail. As part of this process it is important that previous experience on standardisation and open systems is reviewed. The following section briefly relates how the models proposed within our work relates to the current work of the ODP community.

6. ODP and CSCW

In the previous chapters we discussed the need for open CSCW systems and the requirements which we believe need to be met by both applications and a distributed systems support. The aim of open systems has been examined by many researchers within the distributed computing community particularly in the context of ODP. This chapter briefly relates the work described here and open CSCW systems to the on-going work of ODP.

The following sections investigate how the current views of ODP meet the requirements of CSCW and how the work being undertaken in the development of open CSCW systems relates to ODP. It is important to emphasize that ODP should not simply be regarded as a framework for the development of open CSCW systems. It is our belief that the concepts and models being investigated within CSCW are relevant to ODP and have a significant input to make in its future development.

More general remarks about the relation of CSCW and ODP are presented in section 6.2.

6.1 The impact of CSCW on ODP

Rather than dealing with the full complexity of distributed systems, ODP considers systems from five different viewpoints (Enterprise, Information, Computation, Engineering, and Technology Viewpoint). Each viewpoint represents a different set of abstractions of the original system, that is, a simplified view. The ODP design trajectory [19] prescribes that the design outline should start with the selection of a viewpoint that is most appropriate for the design and architecture of the considered application. For CSCW applications this is either the enterprise or information viewpoint. But also the ODP-aspects visibility and transparency of the computation viewpoint are of interest.

Enterprise and Information Viewpoint: Activities, Roles and Sharing.

The support for activities required by CSCW systems highlights aspects which are strongly related to the enterprise viewpoint within ODP. The current view in ODP is that enterprise and information modelling results in a set of requirements and restrictions for the computational model. This model is a structuring of the functions identified in the higher models, in terms of computational objects.

It seems sensible given the more active nature of organisational components in CSCW systems that the results of enterprise modelling should be used for more than simple input to the development of other viewpoints, because this is the central matter for CSCW, as the computational viewpoint is for ODP.

It is important for the management of open CSCW systems that knowledge central to enterprise modelling is supplied by an appropriate service. Perhaps, within future ODP systems aimed at supporting CSCW applications the organisational knowledge base considered in the Mocca environment will be associated to the trader, containing or dictating among other the trading policy.

The ODP community should be conscious to avoid the problems the office procedure/information system developers had to deal with. These were mainly the problems of being too rigid and procedural in the description of organisational activities and information flow. Systems developers often forgot the human factor and the fact that employees do often not behave as it is prescribed in the organisational handbook. (Some people are convinced that this is the only reason why large companies survive.)

In fact, the modelling of organisational structures and activities plays a significant role in the CSCW context. Therefore, the various models for the description of offices

procedures as well as (group) activities [16] developed in that context are a valuable source for the development of information and enterprise models.

The Computation Viewpoint: the aspects: visibility and transparency

The concept of user tailorability is central to CSCW systems and as a result a need exists for system visibility and transparency to the user. Of course, this is mainly directed towards CSCW application developers, but it also has a relevance for the underlying distributed system. Within ODP these concepts are reflected by the so called *aspects: visibility and transparency*. They play a central role in the computational viewpoint and become central in the discussion of distribution transparency.

The decisions involved in providing selective distribution transparency is important and ODP has considered the need to fully discuss the advantages and disadvantages of selective transparency. However, with CSCW systems selection mechanisms shouldn't be provided only for application designers and developers. The user centred view of CSCW systems means that the user should be allowed to select their required transparency.

It is important that ODP investigate mechanisms of providing user tailorability of transparency and provides appropriate presentation techniques to allow transparency and its tailoring to be described to the user.

6.2 The relation of CSCW and ODP

ODP and CSCW have different aims and requirements, they look at "real world" from different perspectives, and try to solve different problems. As a result, they build different models of the "real world" (related, but with different taxonomies). One describes *all possible distributed systems*, the other describes *all possible CSCW systems*. In order to satisfy the aims, in each case choices are made. These choices will prescribe which of all possible systems will qualify as open. Both, ODP and CSCW choices are compatible, this implies that both "Reference Models" will be inter-related [15].

The next stage will be to define standards for the functions, mechanisms, information, etc. identified in the model. Our view is that all possible Open CSCW Systems are a subset of Open Distributed Systems, that means that specific environment standards will be a subset (or a specialisation) of specific ODP standards.

Open systems do not need necessarily conform to all ODP standards (some of them may be domain specific), that means that we will find CSCW and non-CSCW systems, whether they conform or not with environment standards. As a result environment standards provide some additional CSCW capabilities, but this does not compromise the goals of ODP standardisation.

Our approach to open CSCW systems has been based on the development of an environment to support a number of different CSCW systems. This approach is similar to that adopted by the ODP community who have also exploited the development of an ODP environment to describe much of their work. A suitable ODP context for the development of a CSCW environment is shown in figure 4.

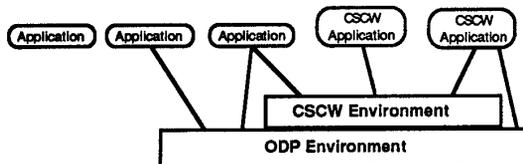


Figure 4 ODP and CSCW Environments

The CSCW environment is located between the basic ODP environment and CSCW applications. However, even applications which are not typically regarded as CSCW applications, like document processing systems, might use the CSCW environment when they are used in a cooperative context. In that way a CSCW environment augments ODP with CSCW specific functions and requirements. This means that the ODP environment will support all open CSCW Systems, but open CSCW systems will be a subset of ODP systems.

Even though a set of similarities between ODP and the approach towards open CSCW systems have been identified, a difference regarding the intention of ODP and the approach of open CSCW should be mentioned. It is the aim of ODP to *mask distribution and to obtain portability and interworking*, i.e. to overcome the heterogeneity between hardware, operating systems, networks, programming languages, storage services, administration, and management. In contrast the central aim of an open CSCW environment is to *overcome heterogeneity between different classes of cooperative applications, tasks and organisations*. However, both approaches are strongly interdependent and it is likely that future work on both CSCW and ODP will mutually inform each other.

7. Conclusions

This paper has presented a brief introduction into CSCW and has examined the need for open CSCW systems as well as their requirements. The Mocca project was introduced with its aim to develop an environment which will support open CSCW systems. It was shown that the experience and approach of ODP can be a valuable tool in realizing open CSCW systems in the future. Similarly, the ODP standardization effort can benefit both from the experience of CSCW application developers and

the requirements which CSCW systems place upon their distributed platform.

The Mocca group will continue its work on a CSCW environment. The future work will focus on the details and interrelation of the models outlined in this paper. The ODP work will be observed and included into our work where applicable. We are looking forward to a very interesting and exciting decade of new distributed CSCW applications and we hope that our efforts towards open CSCW applications will be anticipated. Looking back to title of this paper we would like to answer this question with yes!

8. References

- [1] Rodden T., Blair G. " Distributed Systems and CSCW: the problem of control", Proceedings of ECSCW'91, 25-27th September 1991, Amsterdam, Netherlands.
- [2] Bannon L., Schmidt K. " CSCW: Four Characters in Search for a context" In J.M. Bowers, S.D. Benford (Eds.) Studies in Computer Supported Cooperative Work, North-Holland, 1991.
- [3] Ellis C.A., Gibbs S.J., Rein G.L. " Groupware some issues and experiences", Comm ACM Vol 34, No 1, January 1991.
- [4] Kreifelts T., Santo H. (Eds.) "Group Support Systems", Springer 92.
- [5] Lauwers J.C., Lantz K.A. "Collaboration awareness in support of collaboration transparency: Requirements for the next generation of shared window systems", Proceedings of CHI '90 Seattle, Washington April-5, 1990, ACM press
- [6] Gust P " Shared X: X in a distributed group work environment", presented at the 2nd Annual X conference, M.I.T, Boston, January 1988.
- [7] Malone T W, Lai K, " Object Lens: A Spreadsheet for Cooperative Work", in proceedings of CSCW88, Portland, Oregon, September 1988.
- [8] Pankoke-Babatz U. (Ed.) "Computer Based Group Communication, the AMIGO Activity Model", Ellis Horwood, 1989.
- [9] Palme, J., "COM/PortaCOM conference system Design goals and principles", INTERACT '84, B. Shackel (Ed.), IFIP 1985.
- [10] Stefik M., Foster G., *et al* "Beyond the chalkboard: computer support for collaboration and problem solving in Meetings", Comm ACM Vol 30, No 1, January 1987.
- [11] Ahuja S.R., Ensor J.R., Horn D.N. " The Rapport Multimedia Conferencing system", in Allen R.B.(ed) COIS88 Proceeding conference on Office Information Systems, March 23-25, 1988, Palo Alto, California.
- [12] Johansen, R. "Groupware: Computer Support for Business Teams ", The Free Press, N.Y., 1988.
- [13] Kreifelts, et. al., " Experiences with the DOMINO office procedure system", Proceedings of ECSCW'91, 25-27th September 1991, Amsterdam, Netherlands, Kluwer Academic Publishers

- [14] Prinz W, Pennelli P. " Relevance of the X.500 directory to CSCW applications" In J.M. Bowers, S.D. Benford (Eds.) Studies in Computer Supported Cooperative Work, North-Holland, 1991.
- [15] Navarro, L., "An Environment for Open Distributed Group Communication", Position paper for the International Workshop on ODP, October 8-11, 1991, Berlin, Germany.
- [16] Prinz W., "Survey of Group Communication Models and Systems" In U. Pankoke Babatz (Ed.) Computer Based Group Communication, the AMIGO Activity Model, pp 127-180, Ellis Horwood, 1989.
- [17] Rodden T. , " ECSCW'91 Developers Workhop Report", SIGCHI Bulletin , In Press.
- [18] ISO/IEC JTC1/SC21 WD7 (N309-N315), Drafts of standard documents on Open Distributed Processing
- [19] Popien C., Spaniol O., de Meer J., "Systementwurf mit Open Distributed Processing",Praxis der Informationsverarbeitung und Kommunikation, No. 4, 1991,