

**A METHOD FOR EVALUATING
WORK GROUP PRODUCTIVITY PRODUCTS**

by

Barry D. Floyd

and

Jon A. Turner

Information Systems Area
New York University
90 Trinity Place
New York, NY 10006

April 1988

Center for Research on Information Systems
Information Systems Area
Graduate School of Business Administration
New York University

Working Paper Series

CRIS #179

GBA #88-27

A Method for Evaluating Work Group Productivity Products

Barry D. Floyd

Jon A. Turner

New York University

Abstract

Numerous software products claiming to improve work group productivity are making their way into the marketplace. How is a manager to decide whether or not to invest in these products? This paper proposes a suitable method of formulating the evaluation problem for work group application products. The issue of normal software evaluation is considered first. Then group work is described and contrasted with individual work activities. A two-level evaluation strategy consisting of broad coverage and detailed analysis is introduced. Detailed analysis consists of functional performance, administration, and fit applied across task, group and communications domains. Examples, drawn from some of the products demonstrated at this Symposium, are used to illustrate how this methodology may be applied.

1. Introduction

Most software products on the market today are tailored to support *individual* workers. One need only consider a typical text editor with commands for inserting, deleting and moving text to realize that no provisions have been made in these systems for working with others. For example, one cannot easily make comments on draft text, send them to the author and have them appear in his copy adjacent to the portions to which they refer, attributed to the commentator.

Yet, people rarely work in isolation. They interact, share information, apportion tasks among themselves, monitor each other's performance, and communicate. All of this activity occurs in a cultural and social environment whose quality affects the process and outcomes of the work itself. Furthermore, this work activity is influenced by the characteristics of technological tools themselves.

Improvements in computer and communications technologies have made possible a new class of application systems intended to support group work. The objective of these systems is to support meetings, coordination, project management, co-authoring, decision making, and collaboration on a variety of activities. However, the evaluation of these systems presents a unique set of problems.

The purpose of this paper is to provide guidance to management in the evaluation and selection of group work support systems. We first consider the evaluation problem in general. How does one go about evaluating any application product? The next section provides a definition of group work and describes the ways group work differs from individual work. We then extend the evaluation approach presented earlier to encompass systems supporting group work. Finally, we demonstrate the use of this evaluation methodology with one hypothetical group application, co-authoring. An

application of the methodology to four group support products demonstrated at the Symposium is contained in the Appendix.

2. The Evaluation Problem

Evaluating software for support of group work has two components. The first component involves the normal criteria and methods applied to the evaluation of **any** application software product. The second component addresses those issues that are unique to group work.

Normal criteria applied to application software include such factors as:

Functionality	What features does the product have and how well do they match the work task to be performed?
Integration	How well are the components of the product related to each other?
Interface	How well has the user interface been designed and how does it match the intended user population in the work setting?
Support	How much training and technical assistance will be needed by the intended user population?
Reliability	How well is the product coded, what type of developer support is likely to be needed, will the product be enhanced, and will the developer firm survive?
Efficiency	What resources are needed to run the product and how well does the product fit in with other products in the user's immediate operating environment?
Ease of Modification	What can be changed and how easy is it to do so?
Economics	What are the costs of using the product and how do they compare to the anticipated benefits?

The first three categories are considered **functional performance** because they identify a product's functionality and the methods by which this functionality is

invoked, i.e., attributes which are specific to the product. Other criteria, such as reliability and efficiency, are generic to all products.

Much has been written about these subjects and it is not our intent to repeat or summarize that material here. It is sufficient to observe that:

1. The intended use of a product and the specific operating environment determine which of these factors predominate in an evaluation.
2. The above list of criteria is not closed. Other categories could easily be added with proper arguments as to their importance.
3. While application software evaluation does have objective aspects, the process is essentially subjective. That is, the importance assigned to any factor is a function of **individual** assumptions and opinions that are often implicit and not divulged by the analyst.
4. The tendency is to perform an incremental evaluation; that is, compare one product to another in a product category, rather than using a top down approach where **needs** are determined and then compared with the available products. While incremental evaluation may result in a local optimum, it often provides a major error.

The evaluation of applications to support group work will be even more difficult than selecting systems for individuals. This is because group work extends the needed features and coverage of a system in new directions, creating a different, larger evaluation problem. Attention is now turned to the characteristics of group work.

3. Nature of Group Work

Group work differs from individual work in four respects:

1. It involves extensive and varied person-to-person communication. Consequently, any product supporting group work must have a variety of communications capabilities that can be laid on top of physical services. Furthermore, these capabilities need to be customized for individual workers. In addition, some entity has to **manage** the communications activity.
2. There are processes, **beyond** those necessary for individual workers, that need to be supported. Roles, protocols and procedures have to be established. Group process needs to be monitored and facilitated.

Interpersonal activities must be attended to.

3. Because the overall work task is partitioned among a number of workers, there is a task **management** function that must be performed. This involves assignment, status monitoring and integration of individual work products, and their assembly into a deliverable unit.
4. The **relationship** of the work group to the larger entity, of which it is part, needs to be considered. This includes such issues as organizational culture, structure, power, authority, norms and values.

There are many groups within organizations. Examples are:

- Committees
- Project teams
- Formal authority groups
- Peer groups
- Informal social groups or networks
- Information exchange networks

The project team will be used for the purposes of discussion in this paper because it is frequently found in business organizations and has both formal and informal components. A team can be defined as:

... a group of individuals (two or more) where the members assume specialized roles in both doing the work and maintaining the cohesiveness and morale of the team members. We assume the team has the resources needed to fulfill its role and functions, that the team has the ability to use these resources at the appropriate times and levels, and that they have the capacity to plan, organize, make decisions, communicate and negotiate the actions to reach the team's objectives. [1]

Teams are characterized by much communication for the purpose of information sharing, resource allocation, problem solving, and negotiation over courses of action [2]. Individuals working alone are spared this communication load because most of these activities are performed by themselves and whatever communication is needed takes place internally, presumably at much higher bandwidth and with much greater understanding (e.g., in some efficient internal representation).

4. Evaluation Methodology for Group Work Products

Because of the need to support the individual's enhanced role as a member of a group, normal evaluation methods must be extended to encompass group support application systems. Two levels of evaluation are considered: coarse and detailed.

4.1. Coarse Evaluation

Coarse evaluation refers to the coverage of a system, both in features provided by the application software, and in connectivity or topology of the work group supported. Application systems must exceed a critical mass of coverage for them to be interesting. If the software contains insufficient features, or does not cover a major part of the work group, and often key individuals outside of the immediate group, it is not worthwhile to do a detailed analysis.

Broad feature coverage can be determined by comparing the general features of the application system to a classification scheme, such as that provided by Johansen and described in chapter N of this book (refer to Figure 4-0. Connectivity coverage can be determined by comparing those who will use the system with the composition of the work group.

4.2. Detailed Evaluation

Although the focus of this evaluation method is on **group** work, at the most primitive level most work is performed by individuals. Consequently, a major portion of the evaluation process for a group support application product must consider how well that product supports an individual performing his/her work activity, what ever that may be¹. The domain over which this evaluation takes place is called **task**. But this is not sufficient for the evaluation of group support systems. Two new **domains** must be also considered: group processes and communications.

¹Subtle re-structuring of individual work activities may take place when a new, computer based **tool** is used in performing the task. For example, if the job were assembling a box and the tool was changed from a hammer to a screwdriver, the task sequence and activities would change.

Categories	System			
	A	B	C	D
Face to Face Meeting Support				
Facilitation services				
Group decision support	X			
Presentation support	X			
Computer supported meetings				
Support for Electronic Meetings				
Extensions of telephone		X		
Personal computer software				
Computer conferencing		X		X
Text filtering				
Audio or video teleconferencing				X
On-line resources				
Support Between Meetings				
Project management				
Calendar management				
Group writing software				
Conversational structuring				
Text filtering				
Spontaneous interaction				
Comprehensive support			X	

Figure 4-1: Group Support System Coverage Diagram

Group processes are those activities that support and facilitate individuals working together. For a team this might consist of agreeing on a leader; working out key group processes, such as the way of deciding important questions, or setting work agendas; and establishing a method of resolving conflicts when they arise. It is in providing an individual support for his/her role in a group that this technology has its greatest potential.

Communications are those exchanges between two or more people that result in transfers of information. These consist of the ability to send and receive messages,

methods of addressing individuals, network topologies, characteristics and protocols.

Not only must functional performance of the system be evaluated, but two additional **categories** need to be considered: the extent to which the system supports administrative activities and how well it fits the organization. Figure 4-1 shows the detailed evaluation matrix.

Categories	Domain		
	Task	Group	Communication
functional performance	Normal System		
admin			
org fit			

Figure 4-2: Detailed Evaluation Matrix for Group Support Systems

Functional Performance refers to how well the application system functions in each domain. For example, functional performance in the task domain refers to how well the system supports individual work tasks. This is similar to evaluating a **normal** application system for a single user where one is concerned with the features supported, the degree of integration of these features, and the quality of the interface. Performance in the group process domain refers to how well the system supports those activities that are **unique** to group work, for example, group decision making, or the sharing of a work product for mutual comment. Performance in the communications domain refers to the features provided by the application to support communications, for example, whether a return receipt can be requested and whether one can reply to a message.

Administration refers to how well the application system supports the management of each domain. In other words, how well does the system keep track of resources available, assigned and used; how easy is it to enter this information; and how

is status determined.

Organizational Fit refers to how well an application system compares with the explicit or implicit procedures, policies, norms and values of the task, communication and group process domains of the organization for which it is intended. Any designed object, including application products, have a central concept or theme. Philosophies of designers shape this central concept through decisions that produce features and procedures of the application product. The result is an implicit structuring of tasks through the specification of which can be performed by whom, when. Organization designers, as well as organization culture, influence policy and procedure decisions that establish norms of accepted behavior. These organizational norms interact with those of individuals to produce work group policies, procedures and norms. Fit refers to how well the system supports existing individual and organizational norms and procedures.

Organization fit consists of at least four components: philosophy, work process, structure, and control. **Philosophy** is the dominant style of an organization often expressed by the behavior of its leaders, for example, authoritarian, democratic, *laissez faire* or paternalistic. **Work process** is the sequence in which tasks are performed and their organization. **Structure** refers to relationships among workers. **Control** refers to strategies used to control worker behavior and performance.

An example may clarify the notion of organization fit. Suppose one were evaluating a presentation support system for a military command unit. The application system has a democratic design concept that lets a person message anyone else with their comments about the presentation. The functional performance and administrative support capabilities of the system are considered adequate. However, the democratic concept of the system, and specifically the ability to generate and transmit comments independent of role or level of authority of the commentator, conflicts with the military's notion of chain of command that routes information along specified channels based on role and where decision authority and responsibility rest with one individual. Thus, there is a lack of fit in philosophy (system is democratic while the organization is

authoritarian and the individual is not considered) and in structure (system permits lateral relationships while the organization does not) while in control there is more of a fit (because the system keeps a formal record of what were previously informal communications facilitating assignment of responsibility).

5. Evaluation Example

A hypothetical example will be used to illustrate the proposed evaluation framework.

5.1. Co-Author

Co-Author is a hypothetical system that supports two or more authors working on the same paper. It permits multiple reviewers to comment on portions of text and to have these comments be visible to readers of the paper attributed to reviewers. The system also allows both concurrent and non-concurrent conferencing to discuss the paper and a reviewer's comments with one or more of the authors. The system keeps track of various versions of the paper as well as proposed revisions. It contains a formatter and interfaces directly with a high quality output device as well as a PBX. In this system, a paper goes through several stages, among them "draft" and "final." Changes can only be made in the draft stage and only by the person who originally entered the data. In addition, an agenda must be created prior to establishing a computer conference.

Figure 5-0 provides the coverage diagram for Co-Author. The system provides facilities for supporting group management and project management in the sense of a paper being a "project." It also provides computer conferencing and some calendar management for conference scheduling.

Figure 5-1 shows the detailed coverage diagram for Co-Author. At the functional performance level, the system contains normal editing and formatting features to support individual authoring. In addition, it supports group activities such as agenda setting among the co-authors and the establishment of agendas when an electronic

Categories	System	Co-Author
Face to Face Meeting Support		
Facilitation services		
Group decision support		
Presentation support		
Computer supported meetings		
Support for Electronic Meetings		
Extensions of telephone		
Personal computer software		
Computer conferencing		X
Text filtering		
Audio or video teleconferencing		
On-line resources		
Support Between Meetings		
Project management		X
Calendar management		X
Group writing software		X
Conversational structuring		
Text filtering		
Spontaneous interaction		
Comprehensive support		

Figure 5-1: Co-Author Coverage Diagram

conference is planned. It also provides facilities for commenting back to reviewers and for exchanges among co-authors. In the communications domain, the system supports message and file transfer between authors and reviewers.

At the administrative level, the system supports the creation of individual work tasks, entering status data about them and reporting project status. It also provides facilities for assigning work among authors and to reviewers, determining the status of the paper, author or reviewers, and control over which revisions are entered into the draft and final versions of the paper. In the communications domain, the system maintains track of the locations of authors and reviewers, the status of outstanding

messages and the history of messages between partners.

At the organization fit level, there are no serious philosophy conflicts, but a rigid work process that does not permit changes at the final phase (task domain) and strict control that only allows the author of a section to enter revisions (group domain) makes the application unacceptable for the work group that is considering using the product.

Categories	Domain		
	Task	Group	Comm
functional	editing features	agenda setting	send/receive copy
perf	formatting features	feedback to reviewers exchanges among co-authors	co-authors and reviewers
admin	assing ind work tasks status of ind tasks	aportion work among co-auth status of co-auth relations control over revisions	conn to co-auth, rev status of msg loc of players conf
org fit	work proc "draft", "final" change only draft	only orig auth change draft agenda prior to conf	auth and rev interconn

Figure 5-2: Detailed Evaluation Matrix for Co-Author

Four examples applying this evaluation methodology to current group support applications are provided in Appendix I.

6. Conclusion

It has been argued in this paper that evaluating work group support products is a special case of evaluating any computer software. That is, a number of factors must be considered over a number of domains. Group work has been shown to differ from individual work in a variety of ways, particularly in the making of many internal processes explicit. The notion of functional performance across the task domain was

expanded to include two new domains of group and communications. Two new categories, administration and fit, were also introduced as levels or classes of evaluation. The methodology was applied to a hypothetical example of collaborative authoring.

While this methodology is by no means complete, it is an attempt to evaluate application products on those dimensions that differentiate group from individual work.

I. Product Evaluation

I.1. METAPHOR

METAPHOR is a comprehensive system that includes equipment, software and communications offering users a set of integrated tools designed to support data retrieval, analysis, and text processing. The system consists of high resolution graphics workstations interconnected via a local area network (LAN). This configuration allows for local storage at the workstation and shared data storage via a file server on the LAN. Users may communicate through a messaging facility over the network. Messages can be comprised of text, programs, models, and/or graphics.

Figure I-0 provides the coverage diagram for METAPHOR. The system's strength is in supporting an individual (typically a brand manager). This support is directed towards activities which take place between meetings; the system offers little direct support for real-time meetings, electronic or face-to-face.

Figure I-1 shows the detailed evaluation matrix for METAPHOR. At the functional performance level, METAPHOR mainly supports data retrieval and analysis tasks. It augments these capabilities by allowing a user to combine them into larger, executable modules. For example, a user can define a data retrieval task, have the results sent to a spreadsheet model, and have these results incorporated into a report. This sequence can be defined, saved, and then re-executed at any time. Further, it may be sent to another user over the network.

An important aspect of the system is its ease of use. The interface is icon-based. The design goal was to allow non-computer professionals to perform their own information processing tasks quickly and easily through direct manipulation of objects. It is highly successful at this and provides an almost ideal work environment.

At the administrative level, METAPHOR provides little support. There are no project or individual work management functions. For example, version control of

Categories	M	System		C
		IS	IB	
Support for face-to-face meetings				
Facilitation services				
Group decision support				
Presentation support software		X		
Computer-supported meetings				
Support for electronic meetings				
Extensions of the telephone				
Personal computer software				
Computer conferencing		X	X	
Text filtering				
Assistance for teleconferencing				
On-line resources				
Support between meetings				
Project management software				
Calendar management software				X
Group writing software				
Conversational structuring				X
Group memory management				
Spontaneous online interaction				
Comprehensive support systems	X			

Figure I-1: Group Support System Coverage Diagram

M - METAPHOR
 IS - INSYNCH
 IB - InBOX
 C - COORDINATOR

models, memos etc. must be maintained manually by the individual user, as must communication and distribution information.

At the organization fit level, METAPHOR does not provide any explicit structuring of tasks or communications processes. It is likely the

LAN will interconnect those workers who are in proximity to one another and therefore have relationships already established; it does not really support an extended community.

Domain			
Categories	Task	Group	Comm
functional perf	text processing graphics data retrieval data analysis	allows sharing	local area network messaging system
admin	-	-	-
org fit	individual performs his own tasks	no struct of work process	physically close no struct of com process

Figure I-2: METAPHOR Detailed Evaluation Matrix

An underlying philosophy in METAPHOR is that a user will be performing his own work on the system without the need of a data processing intermediary. The situation where the system would thrive is one where an individual user has extensive data analysis needs, an inclination to use a system personally, and a constraint in using Information Services. Although there is no explicit need to obtain a critical mass of users greater than one, more users are needed to take advantage of information sharing.

I.2. INSYNCH

INSYNCH is a real-time teleconferencing software package for microcomputer users connected via standard telephone lines. Users on each end of the line may view and interact with the same application, such as a sales forecast developed in the LOTUS 123 spreadsheet package. To provide this teleconferencing capability, the software captures keystrokes from the keyboards of each user and sends them to both processors. If both computers have the same software and data files, then the same operations occur on each system (provided synchronization is not lost).

Figure I-0 shows the coverage diagram for INSYNCH. The system supports electronic meetings and is typical of screen sharing programs.

Figure I-2 shows the detailed evaluation matrix for INSYNCH. At the functional performance level, INSYNCH offers a number of interesting features in teleconferencing. The first is a screen presentation subsystem where users may capture, annotate, and save any screen displayed on their computer. These screens may be organized into a structured presentation along with screens developed with a screen generation package. A second feature is file transfer between microcomputers. This allows users to assure that the data being used is the same on each computer. Furthermore, INSYNCH is easy to use, providing a menu-based interface which is simple to understand and easy to manipulate.

Categories	Task	Domain	
		Group	Comm
functional perf	presentation prep	presentations screen sharing	2 person comm link
admin	-	minutes	file transfers to share data/programs
org fit	-	little control mechanisms	2 persons only

Figure I-3: INSYNCH Detailed Evaluation Matrix

At an administrative level, INSYNCH provides a "minutes" or log function, recording detailed records of transactions occurring during a teleconference.

At the organization fit level, INSYNCH provides new communication channels for two users who are geographically separated. The meeting must take place in real time. INSYNCH does not explicitly structure tasks or communications. While the system does provide the infrastructure for display and interaction of two users working on the same task, there are few mechanisms to control communication. Much like the telephone, two users may speak at the same time. And though the impact of inappropriate timing may result in a perception of rudeness in telephone conversation, in real-time computing environments with keystrokes being entered, it results in

nonsense instructions being sent to the system. In viewing and revising a spreadsheet, this lack of control becomes potentially dangerous to unprotected cells. If problem solving and other interactive types of conversation is to occur with its concomitant interruptions, simultaneous beginnings, and other confusions, additional mechanisms for control and synchronization (such as easily going back to point x) need be included in the software.

I.3. InBOX

InBOX is a MacIntosh-based messaging system that allows users to send and receive messages over a local area network (LAN). InBOX is easy to learn, with an icon-based interface; the goal is to have an unskilled person able to use the system within 10 to 15 minutes. The expected set of users includes both professionals and secretaries. This differing level of computer expertise and skill led to a system interface designed to be comfortable for both. The primary concern was to have a critical mass of users, so that most people would find others they needed to communicate with on the system. Figure I-0 shows the coverage diagram for InBOX.

From a functional performance perspective, the package is similar to other messaging systems. Its market niche is MacIntosh users. The system provides relatively unstructured formats except for phone and memo messages. The detailed evaluation matrix for InBOX is shown in Figure I-3.

Categories	Task	Domain	
		Group	Comm
functional perf	easy to use	-	messaging
admin	-	-	mail lists
org fit	-	-	close proximity due to LAN

Figure I-4: InBOX Detailed Evaluation Matrix

InBOX is consistent with the Macintosh style of user interaction. Skills learned in this environment carry over to this package, reducing the amount of training required.

From an administrative perspective, InBOX offers mail lists to facilitate the sending of messages to groups of people and an RSVP function which alerts the sender when a message has been opened by the receiver.

No special organizational demands are made of the system; it allows communications among any members of the network. It also runs on a local area network (one of the processors acts as the message system controller) which limits its scope within the organization to those individuals connected to the network.

I.4. The COODINATOR

The COORDINATOR is a communications tool which structures conversations people have with each other. The designers of this product assert that conversations are structured (or should be) and that when this structure is explicit the process of communication is improved.

Conversations take place within domains. A domain is a central organizing concept or topic (e.g., hiring personnel) for a particular set of conversations. Two kinds of conversations within a domain are identified: those for action and those for possibilities. Conversations for action are those in which participants' intentions are to produce future actions. Conversations for possibilities are ones in which the result is not a commitment for action, but the possibility for one.

Within these conversations, four things can occur: requests, promises, assertions, and declarations. In conversations for action, the primary occurrences are requests and promises (e.g., You request an expansion plan and I promise to give it to you next Tuesday). In conversations for possibilities, the primary occurrence is a declaration (e.g., It is possible that John and George could work together on this project). The COORDINATOR uses this framework to structure conversations.

Figure I-0 provides the coverage diagram for The COORDINATOR. The system is a conversation structuring tool supporting activities between meetings. The COORDINATOR runs on personal computers connected via a communications network such as a telephone line or a local area network; one dedicated system serves as the "hub" or message server.

Figure I-4 shows the detailed evaluation matrix for The COORDINATOR. At the functional performance level, The COORDINATOR provides writing tools and calendar management for individual tasks. For communications it offers sending and receiving of messages to others with access to the network and running COORDINATOR software. At the administrative level, in the task domain, it provides support for storing and retrieving documents and calendar information. For communications, it has a directory containing the names and network addresses of people with whom a user has had or is likely to have conversations. Conversations are linked so that they can be recalled and reviewed together. At the organization fit level the most important and interesting aspect of THE COORDINATOR is the structure it imposes on conversations. The system requires a particular type of response to a request (e.g., a commitment) and specific information which is associated with that response (e.g., date when the commitment will be completed).

This structure removes ambiguity which occurs in many conversations and makes commitments explicit. It shows clearly when commitments are not being kept and makes it more difficult to let "things slide." If the intention is to commit without really performing, then this becomes clear. However, not all participants may be willing to make their commitments this explicit. And the system may promote process at the expense of content. The organization fit issue is then between this type of conversation structuring and the way group members prefer to interact. There is a certain formality, an unforgiving nature, to The COORDINATOR; it does not permit the kind of slack and flexibility which usually occurs and is realistically needed in day-to-day work.

Categories	Task	Domain	
		Group	Comm
functional perf	writing tools calendar mgmt	-	sending/receiving messages
admin	-	-	manages names on network organizes messages as conversations
org fit	-	-	structures messages and conversations

Figure I-5: The COORDINATOR Detailed Coverage Diagram

I.5. Product Evaluation Summary

The four packages reviewed in this paper vary with respect to their support for group work. Three are explicitly communications-orientated. Two of these have the establishment of new communications channels as their main objective (InBOX and INSYNCH) and these facilitate group interaction through enhanced functionality in data sharing and messaging. Neither offer explicit support of group process.

METAPHOR's focus is on the individual in providing an integrated and user-friendly analysis environment. It impacts the group only by shifting information processing power to area professionals (non-IS). Perhaps the most interesting of the systems from the perspective of this paper is The COORDINATOR. It attempts to redefine the components of communication and to make them explicit; thus it provides not only functionality, but a whole communications philosophy.

References

- [1] M. Cashman.
Improving Team Productivity and Effectiveness.
Technical Report, Kraft Associates and Cashman Consulting Corp., 1987.
- [2] P. Semprevivo.
Teams in Information Systems Development.
Yourdon Inc., New York, NY, 1980.