USING FACILITATED TEAM TECHNIQUES
TO DEFINE INFORMATION SYSTEM REQUIREMENTS

by Diane L. Lockwood, PhD
Albers School of Business
Seattle University
Seattle, WA 98122

ABSTRACT: Facilitated Team Techniques (FTT) consist of a series of highly structured group meetings designed to extract high-quality business information system specifications from end users within a compressed time frame. This paper discusses: the advantages of FTT over traditional data gathering methods such as interviews; some key features of FTT; reported benefits of FTT; and conditions under which FTT are appropriate.

KEY WORDS: systems analysis, requirements definition, participative systems

INTRODUCTION

How do systems analysts gather and analyze data needed to document requirements specifications for business systems? Typically, system analysts will proceed to gather data about requirements from such sources as forms and documents, procedures manuals, observation, work rate and volume sampling, and most importantly, by conducting personal interviews with people directly involved in the business activity.

After concluding the initial round of interviews, analysts recognize that the end users involved are frequently in disagreement with one another on some element of system scope, problem definition, process descriptions, input/output requirements, or priority rankings. Occasionally analysts even recognize that these disagreements are honest ones based on different perceptions of the way things are and the way they should be. Nevertheless, analysts must come to a timely resolution regarding these issues if they are ever going to get a system implemented within the mandated time frame and budget. Thus, the analysts have to go back again and again to interview people and try to resolve disagreements and inconsistencies. The interviewees

often get impatient with the analysts' iterative questions and they have that "Oh no, not you again," tone in their voices when analysts call to schedule follow- up interviews. The system analysts do try to schedule face-to-face meetings with all the interviewees at once, but management often seems reluctant to make the necessary scheduling commitments. By now, the requirements definition phase of the project begins to experience difficulties in clarification of

> *when inconsistencies arise in requirements' specification, it is typically the analysts, not the users themselves, who are ultimately forced to resolve differences.*

business system needs. Deadlines are going to be difficult to maintain. Management is impatient with the apparent lack of progress on the system.

After still more "back and forth" interviews, the analysts finally resolve the remaining loose ends in the best way they can given their collective knowledge

of the system and technical expertise. At this point the users sign off on the requirements specification deliverable usually without a thorough review of this thick document.

As is typically the case with this process, there are people from the user community who are dissatisfied with the new system once it is implemented. Several issues that the analysts thought were previously resolved among interviewees become the subject of future change board meetings, additional interviews, and continued downstream maintenance of the system.

In the scenario presented above, one should take particular notice of two points. First, each user has a different viewpoint and this is important information because changes to the system or a new system must attempt to satisfy as many users as possible. Second, when inconsistencies arise in requirements' specification, it is typically the analysts, not the users themselves, who are ultimately forced to resolve differences. Consequently, there will always be some users dissatisfied with the system.

Achieving good communications between users and developers has been a longstanding but elusive goal. How can

systems professionals deal with the nagging problem of getting top management and users to pay attention to system details during the crucial early phase of requirements specification? Furthermore, how can honest disagreements in requirements specifications be resolved in a timely manner and with a consensus of the people involved? Several techniques such as prototyping, the use of data flow diagrams during interviews, and presentation graphics in analysts automated "tool kits" have been offered to help bridge communication gaps. However, most of these are still oriented toward a single-user requirements specification environment, i.e., one user versus a group of users. This does not mean that these techniques are not applicable in a multi- user environment, but rather that their use in practice is usually confined to a single- user data collection method. Thus, the problem of resolving multiple user views in a requirements specification remains.

Facilitated team techniques (FTT) are an alternative to conventional data gathering and analysis methods. The major value of FTT lies in its ability to develop requirements specifications for large-scale application systems more quickly than a traditional one- on-one interview approach. Time saving occurs primarily via group processes which encourage consensus on requirements at the time they are documented. That is, FTT require users to author documents and agree collectively, in writing, to the content of specifications. This decreases the likelihood of later change requests since it would require considerable effort to reconvene the entire specifications group for another FTT meeting.

WHAT ARE FACILITATED TEAM TECHNIQUES?

Facilitated team techniques consist of a series of highly structured meetings designed to extract quality business system specifications from users and analysts within a compressed time frame, usually less than two weeks. A small number of consulting firms offer FTT methodologies

(1,2). Before discussing similarities across methodologies, a few differences will be identified.

1. <u>Analysis versus Design Phase</u>. Some consulting firms concentrate on the front-end requirements analysis phase while others focus more heavily on design considerations.

2. <u>Data versus Process Orientation</u>. Theoretical approaches may be relatively more "data centered," i.e., emphasizing logical data modeling, or more "process oriented" i.e., data flow diagrams.

3. <u>Level of Detail</u>. Some firms specialize in using FTT with top management teams in order to develop relatively abstract high level "enterprise-wide" models, while others address lower level details related to specific application systems.

Most FTT, however, have four key similarities: structure, workshop format, an impartial session leader, and consistent documentation (1,2,3).

STRUCTURE

Meetings with teams of users and analysts to define requirements is not a new idea, but the degree of structure associated with formal FTT is what differentiates it from an uncontrolled brainstorming session. In order to be considered a structured approach, FTT must have a) a clearly defined business objective and project scope, b) a rigorously documented deliverable, c) a structured agenda capable of being replicated, d) a formal system development methodology, e) a carefully chosen, representative team of participants, and f) a neutral, unbiased session facilitator (1).

The size of the team depends upon the scope of the system, i.e., the number of key business functions. However, the ideal group size based on the author's experiences is from 12 to 15 people. The composition of the team is typically 10% executives, 20% managers, and 70% operational level people. Systems analysts

and data administrators do not typically contribute directly to the requirements definition and analysis document during FTT, but rather function as silent observers. Their presence is needed in order to collect information for subsequent phases of design and implementation. The team meets eight hours a day from two-day to two-week blocks of time, depending on the organization's scheduling constraints and the size of the application system.

WORKSHOP FORMAT

The workshop format, a second key feature of FTT, helps attendees concentrate on exchanging ideas and minimizing politics (4). Team members are guided through a well-ordered sequence of workshop tasks such as data gathering or brainstorming, analysis of the brainstormed information according to a specified set of questions, organization (logical groupings of business functions and information needs), and documentation. Wall- sized charts are typically posted during FTT sessions to enhance the requirements specification process and to facilitate adherence to the workshop agenda. Automated tools such as word processing, data dictionary directories, and graphical data flow diagrams are also used to promote structure and facilitate ease of changes.

IMPARTIAL FACILITATOR

An impartial session facilitator, the third key feature of FTT, must be perceived by team members to be neutral, i.e., have no vested interest in the outcome of the requirements specification. For this reason, an outside systems consultant or "facilitator" will often be used to facilitate a FTT session. The primary role of the facilitator is to guide the team through all steps of the requirements specification process and documentation procedures in a structure, timely, and collaborative manner. Facilitators are challenged to perform in four distinct areas: a) technical skills in systems theory and development methods, b) project management and reporting, c) training groups in systems development concepts and d) managing

group processes (1). They should remain neutral on content issues in terms of the business ideas presented by users.

CONSISTENT DOCUMENTATION

Consistent documentation is also a key feature of FTT. Business functions and information needs are usually recorded by team members on pre-formatted documentation forms during the FTT workshop. An example of a completed documentation form used to describe an external data flow input to a company benefits system is provided in Figure 1 at

> *The primary benefits of FTT are increased productivity..., organizational learning, and user commitment .*

the end of this article. The company identity and employee names are disguised, but the contents are real. Capitalized items in the left hand margin are fairly standard system documentation requirements. The actual content descriptions, however, were both written and reviewed by members of the user team, not systems analysts. Thus, the requirements document is more likely to be understood and "owned" by the users who developed it. This of course requires some initial user training in proper documentation techniques, but as evidenced in Figure 1 they can learn to do this with practice. In many cases, the completed forms become the requirements specification to be used as input into the subsequent phase of system design. The resulting requirements specification should also become the basis for ongoing project monitoring.

WHEN SHOULD FTT BE USED?

The factors to be investigated when considering FTT are identified in Figure 2. In general, a conventional approach such as interviewing should be used when the project is small, i.e., fewer than nine key business functions or processes, and there is general agreement among key

users regarding the objectives and priorities of the system. The cost of hiring a skilled FTT facilitator and the inconvenience of scheduling team sessions would probably not justify the benefit of resolving minor disagreements among users. However, an exception to this general guideline would be the criticalness of the system to on-going business operations. Even if the scope of a system is relatively limited, disagreements regarding functional priorities among users could hinder the timely development and implementation of a key business system.

Conversely, when the system is large, the users are geographically or functionally isolated, and there is general disagreement among users concerning system scope and requirements, a FTT approach is appropriate. Team techniques are recommended in cases where integrated systems are being built to support users which are functionally isolated or geographically dispersed. Improved individual user understanding of the "big picture" helps broaden limited perspectives and reduce anxieties often felt by isolated users.

If there is much disagreement among users regarding the objectives and priorities of the system, a FTT approach should be considered over conventional one- on-one user interviewing techniques. When controlled by a skilled facilitator, direct confrontation is an effective way to resolve conflict and foster mutual understanding. In an interview approach, the task of resolving conflicting opinions is often left to the system analyst(s) rather than the users.

A key issue to be assessed is the extent of top management support. This support could include (5, p. 299):

- keeping in touch with the project team to monitor progress,

- expressing a willingness to consider recommendations resulting from the team sessions,

- providing funding on the project,

- agreeing to allocate sufficient human resources for the project, and

- agreeing to have external consultants (facilitators) to help get the project running.

BENEFITS OF FTT

The primary benefits of FTT are increased productivity due to the reduction in elapsed time to complete application development, organizational learning, and user commitment (1,2,3,4). The data and decisions needed to complete a requirements specification are collected and documented usually within a two-week time frame. This remarkably high level of productivity during the requirements stage of system development realizes substantial cost savings by reducing the total time needed to complete the application development cycle (1).

From the beginning of a FTT session, team members share their different perspectives and learning about different aspects of their company and its systems. It is probable that some of them will meet and talk for the first time. A consistent theme evidenced in user feedback evaluations at twelve large Fortune 500 sites where FTT were employed indicated that perhaps the greatest benefit gained was this opportunity for users to learn about their organization as a system. In particular, participants gain an awareness of how business functions are interrelated within their systems, and between their system and its external interfaces. While this organizational learning is unmeasurable, its potential impact on overall corporate operations is important.

It is axiomatic that a corporation's employees possess the best knowledge about what a system is supposed to do, but they do not always know how to communicate this in terms of accepted systems development practices. Thus, team sessions provide a useful vehicle to educate users about the format, rigor, and level of detail needed for requirements specification.

Additionally, team members develop a

strong sense of ownership of the requirements specification which in turn fosters commitment among users and the development staff. There is empirical and anecdotal support for the argument that user involvement increases the perceived usefulness of information systems (5,6,7,8,9,10).

Do FTT improve the quality of the requirements specification document delivered? To the extent that preformatted documentation forms and rigorous user reviews of the completed contents enforce consistency in documentation standards, the answer is yes. However, one hard measure of quality would be the time and cost involved with downstream maintenance activities resulting from incomplete or inaccurate requirements specification. An ideal approach for future research would be a quasi-experimental design using different system development methodologies in parallel. One project management group would use FTT, the other a more conventional interviewing approach. Two functionally similar, but geographically separated sites, would be used for parallel system development. Maintenance activities, identified as those pertaining to "enhancements beyond initial requirements definition" or "corrective modifications made in response to unmet initial requirements" would be tracked separately and the results compared. Total elapsed time from design through implementation could also be recorded for the two groups as another means to quantify results.

CAVEATS AND CONCLUSIONS

If FTT are incorrectly or poorly implemented, then the end results are often worse than if they had never been used. However, the same could be said of any system development technique. Examples of incorrect or poor implementation include: the use of an unskilled facilitator; lack of a sound theoretical systems development foundation; inappropriate team selection; inadequate planning and top management support; and nonrigorous documentation.

The most demanding feature of FTT is that they require a great deal of sustained energy and a heavy time commitment from all team members during consecutive, full-day sessions for a period of usually two weeks or longer. Indeed, team sessions have often been characterized by participants as "educational, but grueling" and rightfully so because systems are complex and demanding in detail.

FTT session also take participants away from other daily work responsibilities for the duration of the sessions. Top management commitment is needed to encourage and support full-time participation of a number of their key personnel. Practicing systems professionals, however, suggest that is far easier to convince budget-minded executives to invest in tangible hardware and packaged software, than it is to invest in the "people" cost associated with FTT. A company must seriously analyze the costs associated with conducting numerous one-on-one interviews that result in incomplete or inconsistent specifications. What are the costs associated with downstream propagation of specification errors because users are never forced to understand and resolve differences among themselves initially in a team environment? Stated positively, what is it worth to a company from a competitive viewpoint to deliver a rigorous requirements specification in 80% of the elapsed time that it takes using traditional interviewing and other data gathering techniques?

In conclusion, the characteristics of participants and the structure of the application project largely determine the likely success of a FTT approach. Visible top management support for FTT, skilled facilitators, and trained users who are motivated to be involved in requirements analysis are all determinants of successful FTT experiences.

A complex application project which cuts across interunit boundaries, i.e., common systems, will generate a diverse set of often conflicting user requirements. In such cases, consensus on requirements

will be facilitated by constructive FTT conflict and resolution. Further research needs to be conducted to quantitatively assess the impact of FTT on downstream maintenance activities and system implementation time.

REFERENCES

1. Leavitt, Don. "Team Techniques in Systems Development." Datamation, November 15 (1987): 78-86.
2. Rush, Gary. "A Fast Way to Define system Requirements." Computer World, October 7 (1985): 11-16.
3. Gill, Allen. "Setting Up Your Own Group Design Session." Datamation, November 15 (1987): 88-92.
4. Canning, Richard (Ed.). "Developing High Quality Systems Faster." EDP Analyzer, 24, no. 6, June (1986): 1-11.
5. Hirschheim, R.A. "User Experience with and Assessment of Participative Systems Design." MIS Quarterly, 9, no. 4, December (1985): 295-303.
6. Franz, Charles R. and Daniel Robey. "Organizational Context, User Involvement, and the Usefulness of Information Systems." Decisions Sciences, 17, no. 3 (1986): 329-356.
7. Hirschheim, R.A. "Assessing Participative Systems Design: Some Conclusions From an Exploratory Study." Information & Management, 6, no. 6, December (1983): 317-327.
8. Blokdijk, P. "A Participative Approach to Systems Design", in The Information Systems Environment, H. Lucas (Ed.), North-Holland: Amsterdam, Holland (1980): 155-170.
9. Olson, M. and Ives, B. "User Involvement in Systems Design: An Empirical Test of Alternative Approaches." Information & Mangement, 4, no. 4, August (1981): 183-195.
10. Mumford, E. and Henshall, D. A Participative Approach to Computer Systems Design. Associated Business Press: London, England, 1979.

FIGURE 1

Sample Documentation Form

EXTERNAL INPUT DATA FLOW DESCRIPTION

Page _1_ of _2_

SYSTEM XYZ Company Benefits Management System   IDENTIFIER BMS-EI-205

VERSION _2_   DATE _11/21/86_SOURCE Team Requirements Session

AUTHOR(S) John Doe and Jack Smith

REVIEWER(S) Jane Doe

TITLE Enrollment Elections

REQUIREMENT STATEMENT

SUMMARY
: The member's election of specific benefit plan options as well as dependents covered by the plans and beneficiaries of those plans as appropriate.

PURPOSE
: To allow enrollment of the member in specific, member- selected benefit options.

SCHEDULE
: On demand when participant returns enrollment form find an identified return date.

FREQUENCY
: Daily.

VOLUME
: 1 - 35,000 initially, 3-300 monthly

RETENTION
: Kept in file at least 18 months or longer if required.

CONTENT
: Group-number & SSN, participant-name, employment-status, effective-date, benefit-plan-option-choice, cost, calculated-cost-credit-difference, per-pay-period- deduction-addition, DC-amount, DC-percent, participant- signature, signature-date, eligible-dependent-names, dependent-sex-code, dependent relationship, dependent-birth-date, other-group-plan-coverage-indicator, other- group-plan-name, beneficiary-benefits-applicable-indicator.

FORMAT
: Sample document attached.

RELATIONSHIPS
: EI-204 Enrollment Authorization.

SECURITY
: Benefits Administration office and data entry clerk.

EDIT
: Group-number, participant-name and SSN must be on file. Benefit-plan-option-choice must be valid choices in the Benefit Plan.

FIGURE 2

WHEN TO USE FACILITATED TEAM TECHNIQUES (FTT)

VS

ONE-ON-ONE INTERVIEW METHODS

	TEAM	CONVENTIONAL
	_____	_____
	YES	NO

1. Is there much conflict or differences of opinion on requirements within the user community?

2. Does the size of the proposed system exceed nine key business functions or processes?

3. In cases where integrated systems are being developed to support several functions, are the users geographically dispersed or functionally isolated?

4. Is there a need to educate users about the rigor and level of detail necessary for requirements definition?

5. Is this a fast-track development project? For example, is there a need to document the requirements specification for the project in two months or less?

IF YOU ANSWERED "YES" TO MOST OF THESE QUESTIONS, THEN YOU MAY WANT TO CONSIDER USING A STRUCTURED FTT METHOD FOR YOUR SYSTEMS PROJECT.

AUTHOR'S BIOGRAPHY

Diane L. Lockwood is presently an Associate Professor of MIS at the Albers School of Business, Seattle University, Seattle, Washington. She was previously a Senior Associate with the consulting firm of Nielsen & Associates and has worked as a systems analyst. Her current research interests are in the areas of organizational spending trends in IS, and the management of systems projects in a CASE environment.

Information Systems & Computing
Academic Professionals
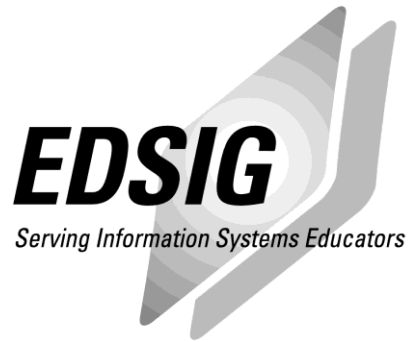
STATEMENT OF PEER REVIEW INTEGRITY

All papers published in the Journal of Information Systems Education have undergone rigorous peer review. This includes an initial editor screening and double-blind refereeing by three or more expert referees.