Exploring Task Atomicity and a Theory of Group Flow Using Collaborative Programming

John T. Nosek
Temple University, nosek@thunder.ocis.temple.edu

Follow this and additional works at: http://aisel.aisnet.org/amcis1997

Recommended Citation
http://aisel.aisnet.org/amcis1997/118

This material is brought to you by the Americas Conference on Information Systems (AMCIS) at AIS Electronic Library (AISeL). It has been accepted for inclusion in AMCIS 1997 Proceedings by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.


Exploring Task Atomicity and a Theory of Group Flow
Using Collaborative Programming

John T. Nosek
Computer & Information Sciences Dept.
Temple University
Philadelphia, PA 19122
215-204-7232, fax: -5082
nosek@thunder.ocis.temple.edu

Keywords: Theory, collaboration, laboratory and field experiments, programming

Introduction

When organizations face complex tasks, they usually divide them into definable atomic tasks which can not be easily subdivided. To increase speed, more people can be added to the point where there is one person per atomic task. Can we assign teams to work on atomic tasks to increase performance and reduce completion time?

Software development is an especially good exemplar to explore task atomicity. Team programming usually means coordinating efforts of individual programmers who divide up a large, complex system. Collaborative programming is used here to mean two programmers working jointly on the same algorithm/code, usually considered atomic, i.e., the purview of the individual programmer. Additionally, reducing software development time is critical: "crunch mode is not a matter of opportunity -- it's a matter of survival ... The ability to get working software quickly into the hands of users will be characteristic of successful data-processing organizations for the foreseeable future. Groups that can produce and install software systems within tight time frames will prosper. Those [that] can't will fail and, in some cases, they will bring the enterprises of which they are a part down with them. Fast response to changing information-processing requirements is a necessity in today's world. (Bodie in [1])."

Evolving Theory of Group Flow

Wilson et al [12,13] report on earlier laboratory experiments where team programmers outperformed individual programmers. These initial experiments were done mostly with undergraduates using a canned problem developed by Ratcliffe and Siddiqi in a study of decomposition strategies in novice computer science students with one half to three years programming experience [10]. It was chosen because it is simple enough for novice programmers to solve within 60 minutes and because it is sufficiently challenging to benefit from collaborative effort. In these early experiments, undergraduate teams reported significantly greater enjoyment (p < .1) and confidence (p < .01) as compared to individuals.

Succeeding laboratory experiments with professionals and graduate students using the same canned problem did not indicate significant differences in performance between
teams and individuals. Additionally, professionals teams and individuals did not show statistically significant differences in enjoyment and confidence, while graduate student teams only showed greater confidence (p < .1) as compared to individual members.

Field Experiment in Collaborative Programming

A field experiment was conducted using very experienced programmers, who worked on a challenging problem important to their organization, in their own environment, and with their own equipment. The subjects were fifteen (15) full time system programmers from a program trading firm working on system maintenance of 3 Unix networks and a large database running Sybase. All of them used the X-window system using the C language. The subjects were asked to write a script to perform a Database Consistency Check(DBCC) on a very large database, called TICK, to get information on program trading, with the output for errors to be written to a file. None of the subjects had worked on this kind of problem before. This DBCC check is considered so critical to the organization's success and generally beyond the skill of in-house programmers, that outside consultants are usually hired to perform them.

The significance level for all tests was set at alpha < .05. Statistically, groups outperformed individuals, enjoyed their problem solving process more, and were more confident in their solutions. The qualitative data also support the statistical results. Prior to the experiment, the majority of programmers were somewhat skeptical of the value of collaboration in working on the same algorithm/program module and thought that the process would not be enjoyable. However, as the results indicate, and supported by their comments, collaboration did improve their performance and they enjoyed their efforts. In fact, all groups outperformed individuals. While there was not a statistically significant difference in time to completion at the .05 level, the groups completed the task 40% more quickly, and perhaps, more importantly, more effectively. The collaborative programmers produced better algorithms and code in less time. As we all know, poorly written code completed quickly, may in fact cause great delays in overall development and implementation time.

The pattern that emerged from these experiments indicates that there is a relationship between the difficulty of the problem, enjoyment in the solution process, and confidence in the solution. As the problem becomes too difficult for individual programmers, collaboration increases both enjoyment and confidence in the solution. Bonus-effect [6], minority influence, or human information processing theories [9] do not adequately explain this synergistic, qualitative jump. Ghani et al advocate the theory of "optimal flow", based on a cognitive theory of human motivation to explain the experiences of individuals as they participate in group work [5]. "Flow" is the term used to describe the "holistic sensation that people feel when they act with total involvement" (Csikszentmihalyi in [5]). Figure 1 illustrate factors associated with the flow experience [5]. When perceived control and perceived challenge are optimally balanced, people enter the state of "flow" where their concentration and enjoyment are higher.
It appears that for tasks that may be too challenging for individuals, collaboration can move the point of confidence (perceived control) which positively affects enjoyment and concentration (as measured by performance) (See Figure 2).

Moving from Individual to Group Flow

In the field experiment with a much more challenging task, one programming pair experienced a "qualitative jump" as compared to the others in the experiment. To the amazement of the manager, it was better than previous scripts written for the company. It is computer-time costly to run these scripts and efficiently written scripts are considered so difficult to create that the company hires expert outside consultants to write them. However, the script written by this one programming team was twice as efficient as previously purchased scripts.

While Ghani et al offer some preliminary evidence that the theory of flow explains the experiences of individuals using group support systems, the theory fails to explain the synergistic effect which appears to have occurred in at least one group [5]. There are several speculations that may help explain the phenomenon. First, one member of the team may have made an intellective jump without much help of the other team member.
This would correspond to the body of literature that indicates that individuals outperform groups in creative, ideation tasks [2]. However, in this experiment, no individual outperformed any of the groups.

Another explanation is that collaboration and verbalization of goals and plans increase motivation and improve metacognitive skills in problem solving which include selection of problem representation, strategies for problem decomposition, and solution [3,4,7,11]. Peer interaction motivates the problem solver to modify claims that are not supportable, resulting in a solution that is often superior to what collaborating individuals could manage alone [3]. In this case, the intellective jump is due to the presence of the other team member with the concomitant improvement in problem solving. However, the increase in performance may still be due to a single member of the team who enjoys an increase in problem solving performance.

Another explanation would be an extension of the theory of flow, I'm calling "Group Flow", to explain a more equal, participative, synchronized, synergistic effect. In this case, "synchronization" of more than one mind is believed to be the added effect which explains synergistic, qualitative jumps in group performance. It is not just that each member of the team has improved problem solving because of the presence of another, but that interchanges between members stimulate and calibrate mental models. In effect, the thought and problem solving process more aptly belongs to the team than to the individual. These results are very recent and these explanations should be considered at a very early stage of development. The theory of group flow is offered to stimulate discussion, and guide additional research questions. One question, among many which remain, is whether Group Flow as described above can be applied to all the cases where teams outperformed individuals or only in those cases where a "qualitative jump" occurs. Another important question is how Group Flow informs the design of information technology to support this process [8]

Summary

From laboratory and field experiments, collaborative programming teams working on atomic tasks of sufficient challenge outperform, enjoy the process more, and have higher confidence than individual programmers. This indicates that organizations may be able to reduce task completion time by employing teams on atomic tasks. It may also mean that two average workers working together may perform better on atomic tasks which may be too challenging to each one alone. A theory of group flow is offered as a possible explanation to "qualitative jumps" in performance and to inform future research, including the development of information technology to support these kinds of collaborative processes.

Acknowledgments

Judith Wilson and Sunil Madhugiri deserve acknowledgment for their excellent work on these experiments.

References
References available upon request from the author.