Improving Software Team Productivity During System Construction

I. Robert Chiang
University of Washington

Follow this and additional works at: http://aisel.aisnet.org/amcis1998

Recommended Citation
http://aisel.aisnet.org/amcis1998/405

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 1998 Proceedings by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.
Improving Software Team Productivity During System Construction

I. Robert Chiang
Department of Management Science
University of Washington

System construction (detailed design, coding and testing) is crucial to software project because it requires extensive (estimated 1/3 of the overall) human and computing resources. This is because during this period a large number of specialists implement the system in parallel. While a large team constructs the system at a faster rate, it also results to conceptual disintegration on system design and thus periodical synchronization among developers and integration of software modules become necessary to ensure system quality.

We investigate the practice of incremental development where the development of the system consists of many construction cycles. Each cycle starts with a development phase and ends with an adaptation phase. During the development phase, developers add functionality and unit-test the software modules in parallel. Tested modules then undergo module integration where the latest development is merged with previous work in system baseline. Such uncoordinated development-and-merge continues until the deterioration of the baseline makes it economically unattractive. A coordination phase then ensues to eliminate system-level inconsistencies through baseline adaptation and structural improvement. A milestone is established at the end of each construction cycle.

The objective of the analytical model is to minimize the coordination effort (and consequently improve the productivity) by allocating project milestones optimally through adjusting the number and duration of construction cycles. Factors considered in the model include: 1) team size and structure, 2) module interconnectivity and coupling, 3) size effect due to growing baseline complexity, and 4) stabilizing effect acquired from prior coordination activities. Given that \( T \) amount of time is allocated for system construction, let \( D_j \) represents the development time during the \( j \)th cycles, and let \( TCT (TCE) \) be the total coordination time (effort). The coordination problem is shown as follows:

\[
\begin{align*}
\min & \quad E(TCE) \\
\text{Subject to:} & \quad E(TCT) + \sum_{j=1}^{n} E(\widetilde{D}_j) \leq T \quad \text{(Schedule Constraint)} \\
& \quad S \sum_{j=1}^{n} E(\widetilde{D}_j) = L \quad \text{(Effort Constraint)}
\end{align*}
\]

\( t_j \geq 0, j = 1,2,\ldots n \\
\( n, S \) positive integer.

The solution to the above problem [\( S^*, n^*, m_1^*, m_2^*, \ldots, m_n^* \)] can be interpreted as: \( S^* \) number of developers should be staffed, and there should be \( n^* \) construction cycles (milestones). To decide the interval between milestones, \( m_j^* \) number of modules should be integrated into baseline before initiating the first milestone; similarly, \( m_j^* \) number of modules should be released between the end of the first milestone and the beginning of the second milestone, and so on.

Preliminary numerical results indicate that when the stabilizing effect is more significant than size effect, project team should coordinate less often toward the end of the project. When size effect caused by higher baseline complexity dominates the stabilizing effect, project team should coordinate more often as the project progresses. When none of the above effects dominates, the length of construction cycle does not increase or decrease monotonically.

Current analytical work will be tested by numerical methods such as sensitivity analysis and simulation. When possible, more validation will be achieved through real-world project metrics and industry interviews. One possible extension of this research is to relax the focus on application software and take the architecture layering issue of embedded systems into consideration. Another extension will be to consider the explicit tradeoff between product quality (features, reliability, etc.) and development cost (effort, time, etc.) during system construction.

References

References are available upon request from author.