CS1TS2 – Software Engineering
Autumn Week 9 – Seminar Topic

The Scenario

Large-Multinational Ltd. have a requirement for an order processing and invoicing system. This is a well understood system for which a detailed and precise specification exists. There are expected to be few problems with developing the system, however due to the extensive and diverse nature of Large-Multinational there is a great deal of development work to do, probably in excess of 60 man-years of effort over at least 24 months. This will be a client server system with all components written in Java, using the Enterprise Java Beans framework.

Large-Multinational are currently in negotiation with Bigg-Sistems-Hice Inc., a specialist software development company to build and deliver the order processing and invoicing system for them. These negotiations are currently foundering on the precise payment schedule for the work. In addition, (but unknown to their potential clients) Bigg-Sistems-Hice are in tense and delicate negotiations with their staff representatives (TURGID - The Union of Rapacious, Grasping, Idle, Developers) on the development of a new performance pay scheme.

The Players

Large-Multinational Ltd.

You wish to devise a payment schedule that only rewards genuine progress. You need some sort of productivity measure for the delivered products that means you are only paying for what has actually been written. Ideally you would like the bulk of the payment to be as late as possible as this will be best for your cash flow. In addition, it is only with the completion of the whole system that the existing legacy mainframe can be switched off and the major savings obtained. Some of the issues that you might need to consider are:

- Much software appears to spend a considerable length of time being “80% complete”
- Given the precise nature of the specifications available, and the existence of similar systems it is probably possible to estimate, with a reasonable degree of accuracy the total “size” of the system by any measure you care to name
- Much as you may to like to pay the whole amount “on-delivery” you realise that 2 years will be too long for your suppliers to wait for their money and that you will have to come some arrangements for payments during the development.
- To avoid any messy and expensive legal disputes the chosen productivity measure should be as objective (i.e. unarguably measurable) as possible

Bigg-Sistems-Hice Inc.

You would like a payment schedule that brings payments in as quickly as possible as this will be best for your cash flow. You need some sort of productivity measure that shows rapid progress, but also allows for some element of profit to be made, rather than just recovering your costs. Some of the issues that you might need to consider are:

- Given the precise nature of the specifications available, and the existence of similar systems it is probably possible to estimate, with a reasonable degree of accuracy the total "size" of the system by any measure you care to name
- Your software modules will each spend some considerable length of time undergoing testing, with the consequent danger that no apparent "progress" is being made (and therefore no payment may be being received).
- You also need to balance the money in from your customer with that flowing out to your workforce
- To avoid any messy and expensive legal disputes the chosen productivity measure should be as objective (i.e. unarguably measurable) as possible

TURGID

You would like a productivity measure that can be used in the performance related pay scheme that will bring the maximum benefit to your members with as little effort on their part as possible. Ideally the productivity measure would be:

- Subjective and open to (mis)interpretation
- Easily manipulated to (apparently) achieve the set goals
- Equally rewarding to all aspects of software development -
    - designing
    - coding
    - testing
    - playing "Quake" during the three hour lunch break

Your Tasks

Either split into groups, or consider each player in turn. Try to devise a productivity measure for the development of Java code modules that meets the aspirations of each player. Each group will be asked to propose a productivity measure as might be proposed by one of the players.

Karl R. Wilcox December 2002