Klocwork®

Automated solutions for understanding and perfecting software

Minimizing the Risks of Architecture Driven Modernization Projects

March 24th, 2004

Agenda

- Business drivers of modernization (aka “the benefits”)
- The technical costs and risks
- The business costs and risks
- A method for de-risking and locking-in benefits
- Summary of benefits

Over time, the software base is turning into a key strategic business asset…

- Market share is expanding
- New markets are being entered
- Next generation products are being launched
- Competition is being beat up

It's a goose that lays golden eggs!

In the continued efforts to address markets and improve the bottom line, things begin to change …

- Software development is out-sourced
- Maintenance teams are formed
- Development teams become distributed
- Platform variants are customized
- Development teams are grown
- Security issues become important

Modernizing the existing software structure becomes both a need and an opportunity…

The need:

- ***productivity improvements***
- ***escaped defects to customer are persistent***
- ***new standards need to be supported***
- ***continual platform upgrades***
- ***increase software agility***

The opportunity:

- ***variants for new markets***
- ***new features for existing markets***
- ***capability extensions to out perform competitors***

But this is a fine time to find out that…

- The documentation is not up to date
- New, inexperienced developers are on board with in-sufficient training while your SMEs are career developing on other projects
- The code is growing at an alarming rate

The modernization that you need to undertake can suddenly become very risky and costly!

So you feel a bit daunted, but this is a career opportunity to become a hero. But what you don't know yet is…

- The boundaries between components are blurred and even totally obscured
- The interactions between components are complex and clustered
- Features have been added that were not planned as part of the original architecture, and have been bolted on
- Functions have been cloned and renamed and are virtually untraceable

And to think that you chose not to sky dive because you though it was too risky!

On top of all this, there is a budget to fight for and executives to convince…

- There are direct costs to consider:
  - Deployment of tools
  - Training to understanding your software
  - Addition of resources
  - On going maintenance of knowledge and architecture
- And opportunity costs:
  - Impacts to scheduled delivery of new functionality
- There is convincing to do:
  - Skepticism of need
  - Fear of touching code
  - $$$ hard to free up
  - Lots of churn in lots of areas

But why bet the farm by performing your architecture driven modernization in one massive high risk, high cost venture?

Architecture Driven Modernization should be undertaken using a controlled, incremental method that is completely integrated into the product development process.

Incremental and Integrated ADM: The steps

1. Assess the current situation and establish a baseline for all critical to quality indicators (CTQs).
2. Begin quick wins by establishing a defect reduction program with measured build by build results. Continue to monitor and trend all CTQs for effective project management.
3. Perform architectural excavation by aggregating to surface anomalies and then by re-factoring to establish required changes and improvements.
4. Establish model controls to immediately prevent any further erosion.
5. Plan out which modernization improvements will be done within this development cycle. ROI can be measured as a combination of benefits to customer and future development, and risk mitigation to current cycle by targeting changes to areas that are changing already due to feature development.
6. Continually track and manage (build over build) CTQs to ensure rate of improvements are tracking to plan. This is an excellent method of risk mitigation and cost control
7. Deliver and manage incremental architecture driven modernizations completely within the feature development cycle.

Incremental and Integrated ADM:

Summary of Benefits

Incremental and Integrated ADM: Reducing the Risks and Costs

- Quick and early wins builds team (and executive) confidence
- Erosion is prevented through integrated defect detection and model control rules
- Original investments in tools and understanding are fully leveraged through-out life of product
- Impact on new feature development is minimized, lessening need for pulling and merging separate streams
- Incremental gains can be reported and based on objectively measured data
- Can be project managed within the current software development program.

Incremental and Integrated ADM: All development can realize the benefits

- New features do not erode modernized architecture
- Architecture knowledge is always available to all developers
- Up to date architecture means that architect extensions required for new features are readily understood and can be executed in a controlled manner. Effort, ROI and risks are quantifiable
- Project management is greatly improved with the ability to measure and manage software system parameters

Klocwork®

Automated solutions for understanding and perfecting software

Thank you for your time