

## Typical Software Delivery Issues

In any software project, one of the key challenges is in putting systems and processes in place to effectively manage the software and the resources. Almost all software projects are plagued by some common issues.

### Resource Management

At various points of the project life cycle resources would be over utilised or idle, this leads to loss of productivity as well as loss of billable hours. This occurs primarily due to varying workloads and a lack of information related to workloads and pending work. In addition 'routine' tasks such as timesheet entry and task status maintenance are not done on time and are time consuming.

### Requirements Management

Managing requirements is an issue even in small projects as there is a 'Chinese Whisper' problem between requirement definition, program specification, build, test plans and testing. This is due to each output being manually derived from the previous without any cross check against the original requirement. As a result the final code may not always do what the client specified, leading to delivery defects, rework and higher costs.

### Change and Defect Management

Change and Defect Management is never fully under control especially in the delivery and UAT stage with the focus being on completing the delivery first. As a result a number of changes and defects are never recorded, billable changes may not be billed, defects may reappear in future releases and the delivered product does not match up with the requirements.

### Configuration Management

Configuration Management is an issue within and across releases; typical problems such as traceability of changes, overwriting of previous changes and reappearance of fixed defects are a result of errors in configuration management.

## Options

The options open to software delivery organisations are as follows:

- Use packages for individual functions (Requirements, timesheets, configuration)
- Implement processes (CMM, ISO)
- Both the above

While these measures work to some extent they suffer from the following drawbacks:

- Are expensive to implement and maintain
- Are time consuming and depend upon manual data entry
- Create data discrepancies between various systems
- Do not provide adequate MIS support

## Software Delivery Management System

Terastone's Software Delivery Management System (SDMS) is an integrated tool which helps you assign, monitor, manage and report work progress in real time. It simultaneously manages delivery configuration in a single integrated environment thereby eliminating the need for developers and team leads to waste time filling in timesheets and reports.

### Resource Management

The Resource Management function enables the user to define resources and allocate them to teams. The system then tracks the resource utilization based on the work assignments thereby enabling managers to track and optimally utilize their resources.

Task status and timesheet entries are generated automatically eliminating the need for manual entry of work done and time taken and enabling the reporting of metrics such as efficiency, on-time completion, defects and defect cycles by resource.

### Requirements Management

The Requirements Manager enables users to define requirements against releases, assign work, maintain defects and changes, organise requirements into hierarchies by releases and across releases.

Requirements are automatically linked to assignments, resources and work products (documents and code). Users can therefore track progress and trace all documentation (including changes and defects) against individual requirements.

### Change and Defect Management

Change and Defect Management enables the user to enter changes and defects (internal and client reported), track changes and defects against multiple releases, monitor status and report progress.

Changes and defects can be linked to a requirement thereby providing complete traceability and history by requirement.

### Release Management

Release Management enables the user to configure releases, freeze check-ins prior to delivery, manage multiple deliveries against a release, review release status and monitor resource costs

### Configuration Management

Configuration Management enables the user to load project configuration from database definitions and program files and documents from existing components or define new ones during development.

It controls project configuration, manages check-ins and check-outs and automatically updates timesheets and statuses based on work done by the developers.

Additionally the Configuration Manager creates and maintains an archive of changes made, thereby enabling the user to compare versions (within and across releases), review changes and trace source of defects.

### Security Management

Application access is granted through either a windows login or through database login. The system manages security at two levels.

Functional access is restricted by user role and is configurable for an organisation. Roles can be defined by authorised users and functional access specified for them

Data access is restricted by teams, each team can be assigned one or more projects and members allocated to the teams will be able to access these projects only.

## Summary

The following table summarises how SDMS addresses the issues in software delivery.

| Issue  | Addressed Through   |
|--|---|
| Resource Utilisation                               | Resource utilisation is tracked through resource allocation, work assignment and automatic calculation of resource loading by date which can be monitored through the resource calendar and other reports.  |
| Timesheet Entry                                    | Timesheet entries are generated automatically on work completion thereby eliminating the need for manual entry of time sheets and ensuring timesheet updates in real time   |
| Requirement Tracing                                | All deliverables, changes, defects and assignments are linked to requirements thereby providing a full history of any requirement within and across releases.<br>The system also provides a reverse tracing whereby a deliverable history can be traced against   |
| Requirement Status Tracking                        | Requirement status is automatically tracked against work assignment and work status giving a real time view and eliminating the need to maintain status separately  |
| Change Tracing and Status                          | Changes are managed in the same manner as requirements enabling automatic tracking and status reporting against assignments and resources   |
| Managing Changes Across Releases                   | Changes can be linked to a requirement and can be assigned to multiple releases (other than the one against which it has been raised). This ensures that changes are not lost in subsequent releases.   |
| Defect Tracing and Status                          | Defects (both internal and client reported) are managed in the same manner as requirements enabling automatic tracking and status reporting   |
| Managing Defects Across Releases                   | Defects are linked to a requirement and can be assigned to multiple releases (other than the one against which it has been raised). This ensures that defects do not reappear in subsequent releases.   |
| Deliverable Tracing                                | All deliverables can be traced by release, resource and requirement providing a comprehensive history of the deliverable, changes made and why they were made   |
| Preventing inadvertent overwriting of deliverables | The Configuration Management System prevents users from simultaneously checking out the same deliverable, in addition an archive is maintained of all previous versions that were checked in, this enables users to compare versions and rollback to previous versions where required   |
| Code Reviews                                       | The Configuration Management system provides a one-click option to compare changes made to the deliverable against the current version or any previous versions, this makes code reviews simple and easy to execute   |
| Integrated Data                                    | The single data source for all aspects of the project life cycle makes it possible to view any report or metrics from multiple views, by Release, Requirement, Resource or Deliverable without any issues of data consistency.  |
| Saving Time on routine activities                  | All reports and metrics are generated on the basis of user actions as part of the normal project life cycle process. For example the status of assignments is maintained based on the developers action in checking in and checking out deliverables. The same action also updates the timesheet thereby eliminating the need for multiple entries. |