

V - 4: Records

“Those who do not remember the past are condemned to repeat it” – George Santayana

<i>What they are for:</i>	Provide an accessible and coherent record of past performance of software development activity
<i>When to introduce:</i>	When there is a need to establish a baseline of software development performance for operational needs and future reference.
<i>When to use:</i>	Operational use; and when data is required for comparisons of current or planned work or products with previous experience.
<i>When not to use:</i>	N/A

Good records of technical work are valuable for gaining visibility of an organization’s software development activities. Their ability to retain and deliver important data and information about working practices and product quality, either across an organization or over time, is considerable and can usually be enhanced with a little care. Never-the-less many organizations fail to value information recorded in the course of work, losing it soon after it is produced. Good record keeping, although mundane, is essential if an organization is to learn about its own performance and improve it.

The types of information that should be recorded may be indicated by the organization’s software development policies, procedures and standards. Typically they will be information that triggers, or records, expenditure of effort, or decisions. These types may include:

- ?? estimates used to produce plans;
- ?? plans for technical work, including decisions about the selection of methods, tools and standards;
- ?? ‘actuals’ data recorded against the plan - both resource consumption; effort, time, budget - and delivery; items delivered, requirements satisfied, earned value, etc.;
- ?? changes to requirements, plans and baselined items
- ?? outcomes of technical reviews – which may contain a record of defects found and fixed;
- ?? records of tests;

- ?? defects logged during reviews and tests, against baselined items and released products
- ?? risks - their identification, mitigation and outcome;
- ?? PIR reports.

Other commonly recorded information include logs of issues and actions, and observation reports. These to have value, day to day, but in the longer term do not have great value, except as a record of the effort consumed. This may be, in part, because they do not have a well defined process associated with them and because they tend to be 'one off'.

This 'typed' information retains its value when the types are distinguished from each other and structured in a manner that allows it to be navigated and accessed by those not directly involved in the work that generated it, perhaps at a later date. Well typed information that describes the planning and conduct of technical work, stored in a common well described structure has two benefits:

- ?? It allows comparison of working practices across different projects
- ?? It shows the absence of information - gaps in the information structure show where little is known

So a well defined project library structure, or repository, containing well typed data, whether supported by tools or not, is highly desirable.

Care should be taken to ensure that tools and technology that can militate against a robust typing and a consistent structure does not undermine a library. Email is pervasive. It's benefits are enormous, but it also causes problems. Ignoring the matter of large amounts of broadcast emails that flood organizations, the use of email to communicate legitimate technical concerns – changes to schedule, technical queries, defect reports can have the effect of 'untyping' important project information that may previously been formally captured and recorded; as change requests, defect report or some such, on a uniquely identified pro forma – paper or electronic. While the contents of an email may be self evident when received, the type of information it contains may not, and the information simply be lost amongst other emails in one or more 'in boxes'.

Similarly, carefully designed and managed web pages can provide the basis for an excellent project library together with all the benefits of ease of use, ease of access, and accountability.

However, the design and control of an electronic library with the attributes of accessibility, consistency and data integrity does take time and effort. Poorly structured and uncontrolled they can obscure and discredit information.

An effective information structure will tend to be simple and robust, enabling the capture of complex or incomplete information. Where the structure becomes over complex, with too much differentiation of information types some information that 'doesn't fit' will tend to be lost and other information difficult to assess or interpret.

As well typed and structured information accumulates during the course of development project managers, senior managers and other will want a high level view of the work's progress and prospects. Because it is typed and structured patterns and trends can be identified easily. This high level view can be provided by a number of means. The use of 'traffic lights' (see V – 10) tagged to project data, can signal priorities as well as identify patterns and trends enabling attention to be directed where needed. Graphics can also show trends, run charts and cumulative counts of, for example, change requests, items delivered, and defects. Cost of Quality analyses (V – 9) are also made easy with well typed and structured information; and ongoing, real time cost of quality analyses become feasible. Finally PIRs provide a view of the conduct of the work from the perspective of those doing it.

The value of this information is ensured if it is managed and controlled. In many organizations, typically engineering organizations this is well understood. In others, where records of technical work are not valued as an organizational asset the value of the information can be undermined by neglect, either conscious or unconscious change and distortion, or simply discarded at the conclusion of a project.

To put in place consistent, widely available record keeping:

1. It is advisable to begin by ensuring senior management support. Get senior managers to issue a policy statement requiring the use of new formats or procedures. Support tools to ease administrative tasks are also helpful but not necessary – however if tools are not used then clear, simple pro-formas are essential.
2. Defect logging is the first step. See V – 6, Defect Tracking.
3. With defect tracking in place use a similar process to establish an easy to use change index. Initially changes should be limited to changes to requirements or project plans. When these

are established change requests can be extended to completed work products used by developers – design documents and code.

4. A simple, widely accessible risk register is the next step;

5. ...followed by records of estimation calculations – showing how estimates were derived before they appeared in plans.

6. Extend record keeping scope according to need. When established and stable include a review of project data within the scope of Quality Assurance.