MCIF – a systematic approach to software delivery excellence

By Steven Trevellion, IBM Rational

Measured Capability Improvement Framework (MCIF) was introduced as a systematic approach to incrementally improving software and systems delivery capability. The iterative approach, illustrated with a real-life scenario, helps organisations to stay always focussed on the next step of improvement which is clearly defined and measured.

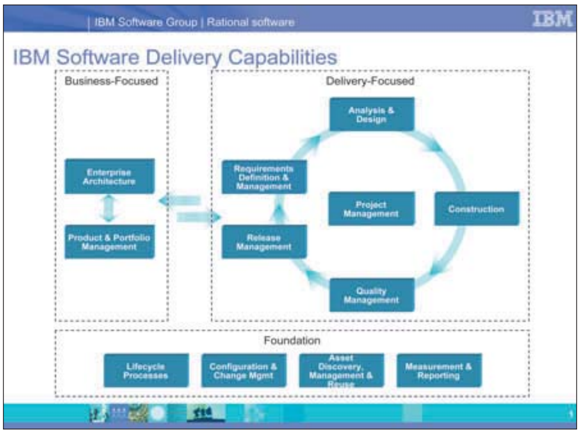

Figure 1. Software delivery capabilities

■ Nowadays business environment requires development organisations to continuously improve their capability to deliver high quality systems and software. This is particularly true for companies developing embedded systems because factors such as innovation and agility are key to building a competitive advantage, in addition to the more familiar concerns of cost, schedule and quality. In addressing all these factors, both economic and engineering discipline is required. Strategic risk-taking to explore new opportunities and good decision-making must rely on objective data to identify real business value. The IBM Measured Capability Improvement Framework (MCIF) provides a systematic approach to incrementally improving software and systems delivery capability. A capability in this context means the combination of process, tools, assets and skills that enable an organisation to efficiently and effectively execute an aspect of software development that results in delivered software. Figure 1 provides an overview of the capabilities addressed within MCIF, grouped into three categories; business focused, delivery focused and foundation.

Business-focused capabilities apply across one or more projects and, as a result, are more closely aligned with the business. Enterprise Architecture is the ability to define all aspects of the enterprise (such as e.g. business strategy, business processes, information, infrastructure and applications) as well as any transition planning and governance. Product and Portfolio Management is the ability to manage products and/or product lines as a portfolio and thereby increase the predictability of product development and alignment to business strategy. Delivery-focused capabilities apply to the execution of a single software development project that is focused on delivery: Project Management is the ability to manage projects to agreed parameters e.g. stakeholder management, planning, staffing, executing, monitoring and risk management. Requirements Definition and Management is the ability to define, validate and manage requirements resulting in improved collaboration and communication among business stakeholders and project teams. Analysis and Design contains the ability to transform all requirements into a solution that has a robust architecture and sufficient design to govern and guide construction.

Construction is the ability to perform detailed design tasks, source code organisation, to implement, unit test and integrate the code into an executable system ready for system verification and quality testing. Quality Management is the ability to ensure the required level of quality exists in the products or systems. Release Management is the ability to repeatedly and reliably build an auditable release of products or systems.

Foundation capabilities apply to all other capabilities and are in general a prerequisite for other capabilities. For example, quality management is of limited use unless the associated work products are managed within a configuration and change management capability i.e. the system under test is of a known, reproducible state. Lifecycle Processes contain the ability to follow a defined process (or method). Configuration and Change Management is the ability to uniquely identify and control all elements of a system that are required to be under configuration control and control changes to work products including version tracking and audit histories. Asset Discovery, Management and Re-use is the ability to identify, create, maintain and retire re-useable assets in such a fashion as to facilitate easy discovery and re-use on subsequent products / systems.

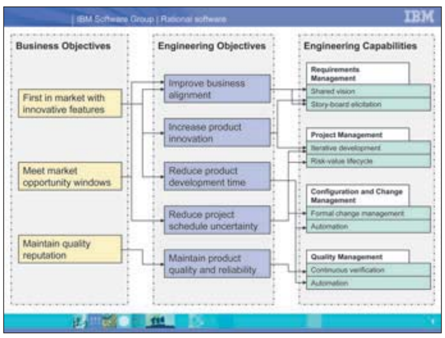

Figure 2. Value traceability tree

Measurement and Reporting includes the ability to provide measurements and reports (based on meaningful metrics) aligned with business value to inform business and development related decision making. There are inter-relationships between capabilities which can be as important as the capabilities themselves. As an example, the traceability from requirements to design and test elements provides the ability to perform an impact analysis of any proposed changes. The MCIF approach to instantiating or improving these capabilities is to follow a four phase approach:

- In the Target phase, business stakeholders define the set of business objectives (such as rapidly responding to market opportunities) before they work with development specialists to determine the associated engineering objectives (such as reducing product development times). The objectives are prioritised and over-arching strategies to achieve these objectives are defined. Each business and engineering objective has success criteria agreed to create a common view of how we will know when we have succeeded.

- In the Map phase, engineering objectives are mapped to delivery capabilities within the development organisation. Dependencies between capabilities and other interdependencies are identified and the deployment of capabilities time ordered resulting in a roadmap that provides a clear picture of when the engineering objectives (and, therefore, the business objectives that they are derived from) will be achieved.

- In the Adopt phase, the roadmap is executed. Capabilities are deployed via pilot and roll-out projects. Self-checking by the development teams enables monitoring and steerage of the adoption within an adoption cycle.

- In the Review phase, progress towards the business objectives is reviewed and any corrections in the approach applied. Focus then typically returns to the next adoption phase to address the next increment of capability deployment. On occasion, earlier phases may be revisited if significant refactoring is required due to market or business changes.

The following example illustrates how the MCIF approach helped a mobile telecommunication company. The actual case is fictitious but is based on real-world scenarios and experience gained by IBM Rational Software assisting customers to improve their software delivery capability over the last 20 years. The company, although successful to date, knows that it needs to undertake significant change to continue as a market leader. Software and systems engineering disciplines are well established and mature, but are struggling to respond to imprecise or fuzzy demands in a highly-charged and dynamic marketplace. The company

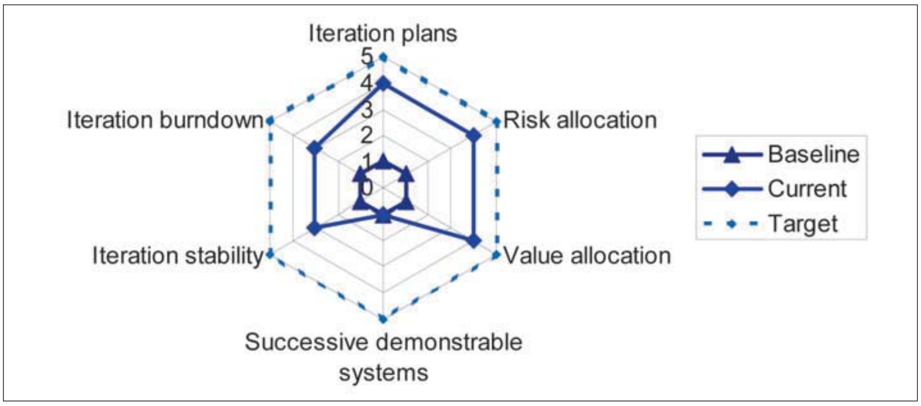

Figure 3. Self check results

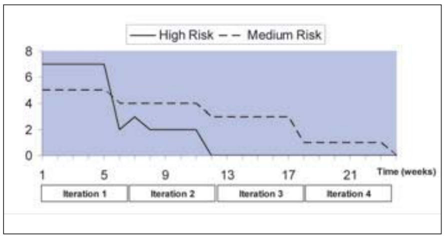

Figure 4. Trend of risk retirement

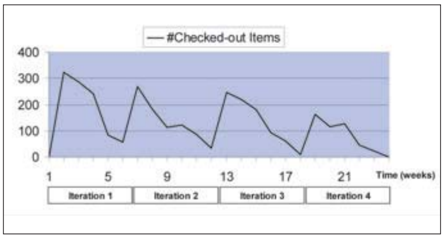

Figure 5. Trend of the number of checked-out work items

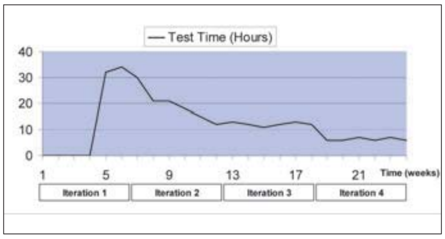

Figure 6. Trend of test time (hours per system test)

decides to undertake an improvement programme based on a systematic, measurable approach (MCIF). The lines of business, product marketing and the development organisation conduct a business value workshop (part of MCIF phase 1). This determines the key business objectives and the measures that will be used to inform steering. These are identified as: increase ability to be first in the market with innovative features (percentage of market leading products), meet market opportunity windows (products must meet production deadlines to hit seasonal market release campaigns, percentage of product campaigns launched on time), and maintain reputation for quality products (number of customer complaints, number of returned products). Note that these are in addition to the existing revenue, profit and market share measures the company uses to govern and manage the business.

The engineering objectives required to support the business objectives are defined: improve business alignment; engineering must be "in-tune" in terms of collaborating with product marketing on new product features and requirements (average percentage of product requirements agreed between stakeholders); increase product innovation, engineering must use its expertise to invent ways of harnessing new technology resulting in new features and market opportunities (number of new innovations per quarter year); reduce product development time, engineering must make efficiency gains in order to deliver products in shorter timescales (average project schedule), reduce project schedule uncertainty; engineering must increase its ability to deliver product commitments on schedule (project schedule variance), and maintain product quality and reliability; engineering must continue to uphold development rigour in terms of product quality and reliability (escaping defect rate, mean time between failures, MTBF).

The over-arching strategy selected is to adopt an iterative, risk-driven development process in which engineering rigour is right-sized to provide the optimum balance between governance and change stability needed to enable flexibility to respond rapidly to changing market conditions and to get innovations into production. In phase 2 the strategy is used to steer the selection of capabilities required to meet the agreed objectives. A high-level view of the resultant mapping (value traceability tree) of business objectives to engineering objectives to capabilities is provided in figure 2. In parallel the measurement system is developed to define the metrics to be used to monitor the improvement programme. A summary of the engineering capabilities and associated metrics is given below:

Requirements Management: The shared vision replaces the existing manual feature and requirement registers with a shared environment where all stakeholders have easy access to definitions and status. Associated metrics: No measure – this is effectively monitored as a binary function (Stakeholders do/do not have access to shared view) – the impact being evident in the requirement trend metrics given below. The story-board elicitation augments the existing textual requirements definitions with story-boarding and other elicitation techniques to aid collaboration and common understanding of imprecise needs. Associated metrics: Trend of requirements outlined, specified, agreed. In the Project Management Iterative development the development project is broken down into a series of successive iterations. Each iteration has a fixed plan (budget and schedule) including allocated requirements, development activities and resources, results in a verified, demonstrable system of incremental functionality, and has a change gate permitting requirement changes in the early, requirements-focussed part of the iteration (circa 20% of the schedule), then closing to provide iteration stability. Associated metrics are: Iteration burndown (percentage complete, remaining estimate), Schedule Performance Index (SPI), Cost Performance Index (CPI), percentage iteration stable (percentage change gate open, closed). In the Risk-value lifecycle the content of iterations is prioritised by risk (to drive risk resolution as early as possible in the lifecycle) and business value (to deliver more valuable content first). Prioritisation is reviewed at the end of each iteration based on results and planning for subsequent iterations adjusted as required (the overall programme is steered to the optimum outcome for all stakeholders). Associated metrics are trend of risks retired (H, M, L magnitude), trend of value implemented.

The Configuration and Change Management includes formal management and automation. Formal change management is the formal process to uniquely identify and control all development elements of the system (requirements, design, implementation source code) including version tracking and audit histories. Associated metrics are the trend of the number of work items configured (check in/out), trend of the number of change requests/status. Automation contains tooling support for configuration and change management workflows. Provides a fully configured developer workspace and functionality to draw down (or be dynamically updated) a complete set of configured (or latest) work items from across the development team (local and remote). Associated metrics are the percentage effort spent on CCM activities. The Quality Management includes continuous verification which means complete system testing (functional and performance), informal during iterations and formal for the end of iteration product. Associates metrics are the trend of number of defects (open, closed), mean time between failures (MTBF). Its Automation contains tooling support for automated functional and performance testing including reporting and generation of change requests based on identified defects. Associated metrics includes the trend of test time per build.

With the capabilities and their associated metrics defined the next step is to define the roadmap, the high-level plan for adoption. Dependencies between capabilities are identified, the capabilities time ordered and a roadmap for a pilot project created. Five capabilities are instantiated in part I, the reasoning is provided by shared vision: provides the shared view of requirements across the stakeholder community that supports the new requirement elicitation techniques of the story-board elicitation capability; formal change management and associated automation provide the mechanism for control of configuration and change required by iterative development and the allocation and management of iteration content (risk-value lifecycle), and continuous verification and associated automation provide the mechanism for informal and formal system functional and performance testing required for the verification of risk removal and value earned during each iteration.

A live project was selected to pilot adoption of the new capabilities (MCIF Phase 3). The selection of a live project ensured that the pilot attracted a high level of stakeholder visibility and that the outcome (positive or negative) reflected results in a true business context. At the end of the pilot, after six months, a formal business review was conducted. As the results were judged a success and no major refactoring was required the adoption moved on to propagate the improvements made across the organisation via a number of waves. Each wave addressed one or more projects within the organisation: Wave 1, three further live projects of approximate duration 6 months, Wave 2, all new projects, and Wave 3, selected legacy systems. As each capability comprised a package of process elements, tooling support, assets and staff enablement, a central team was setup to manage the provision of each of these elements to project teams starting adoption. Enablement was provided via quick-start workshops and on project mentoring. Process, assets and supporting tooling were provided as an integrated, collaborative development environment configured to meet the needs of each project.

A self-check tool was used by the teams undertaking the projects to ascertain their adoption progress. The example in figure 3 shows the results of one of the project teams in wave 1 self-checking their adoption of the new project management capabilities. The team scored the adoption as high for iteration planning, risk allocation and value allocation to iterations but low for successive demonstrable systems. In practice this meant the iterations were not resulting in executable, verifiable systems (of incrementally increasing functionality). This was a serious error in the adoption as demonstrable systems are required for the valid retirement of risks, value based progress reporting and steering. A short period of intensive mentoring was undertaken to re-plan subsequent iterations and train the project teams. Subsequent self checks showed that the capability to deliver successive demonstrable systems was in place.

Having reached a major adoption milestone (end of pilot project) a steering board was convened (MCIF Phase 4 Review). Figure 4 provide three examples of the capability level results reviewed. This chart shows the retirement trends for risks classified as high and medium magnitude. In the risk-value lifecycle retirement of a risk means that there is demonstrable evidence that the risk has been mitigated. For example, in iteration 1 testing proved that the system under development had met critical technical performance requirements identified as high magnitude risks and therefore those risks were retired. By the end of iteration 2 the pilot project had retired all high magnitude risks.

Figure 5 shows the trend of the configuration state of work items (checked in/out) across the pilot project iterations. The trend shows the efforts made by the team to put work items under configuration control (checked-in) ready for formal system testing at the end of each iteration (the dips towards zero items checked-out at the end of iterations). By the end of the final iteration the project had achieved 100% of work items under full configuration and change control.

Figure 6 tracks the time the pilot project spent per system test. At the start of the project the initial testing took approximately four days of effort. This was consistent with historical data for manually intensive testing on previous projects. As the testing was automated and experience of the new testing environment improved this time was reduced to approximately six hours. The review board judged the results as significant improvements. By referring to the value traceability tree they were able to see the linkage from these results to the engineering objectives and therefore the business objectives.

The board approved continuation of the programme to wave 1, adoption on three further six month duration projects (returned to MCIF phase 3 for next adoption cycle). Wave 1 delivered the expected results across two of the three projects selected. The project that did not deliver the same level of results was examined. The issue was identified (as described in the example self-check results) and corrective action taken but the time lost was not fully recovered. The cause of the issue was identified as insufficient skills enablement of the project team and the resources within the central support team were increased to prevent re-occurrence. Today, the improved capabilities are adopted as business as usual for all new projects launched. The board is now considering the application of capabilities to selected legacy systems and the options for further capability improvement in the areas of asset re-use and product and portfolio management. ■