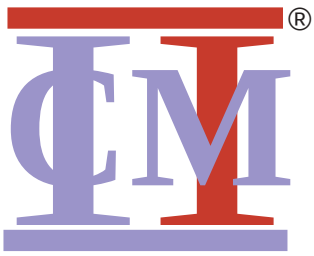
W H I T E P A P E R

Software Process Improvement and Capability dEtermination (SPICE) Relative to CMII

(Rev B)

SUMMARY —SPICE is built on a set of integrated processes that include CM as a supporting process. The CM component is derived from traditional CM which was not designed to accommodate change. Ability to keep a product's definition, its configuration and its records synchronized is thereby compromised.

The CM process, per CMII, must be able to accommodate change and keep associated information clear, concise and valid. Such improvements to the CM component would make SPICE much more robust.



Institute of Configuration Management

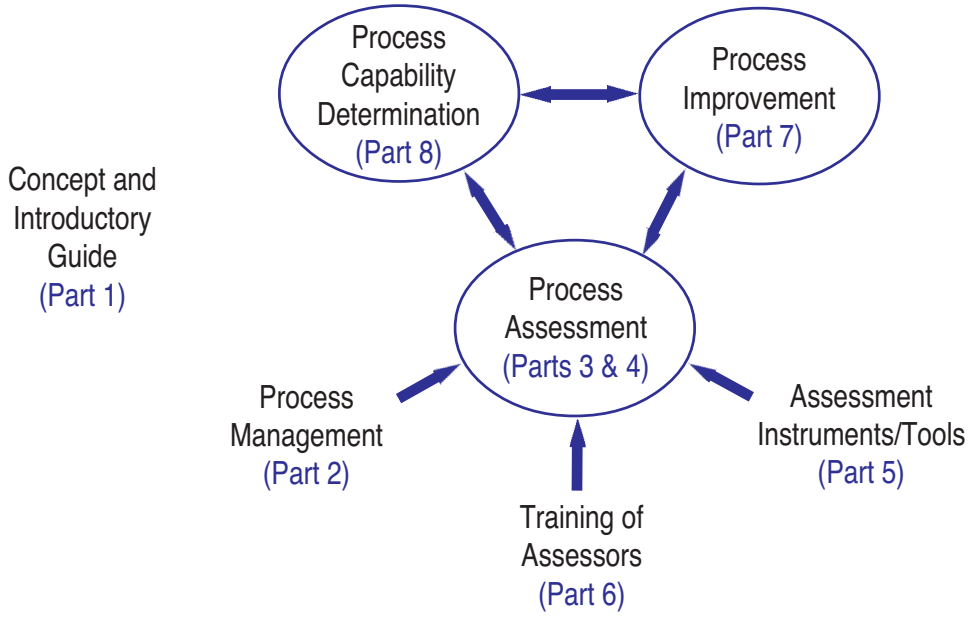
The Home of CMII

SPICE (ISO/IEC 15504) FOR SOFTWARE

Software Process Improvement and Capability dEtermination (SPICE) is described in ISO/IEC 15504.

SPICE is the European equivalent of SEI's capability maturity model integrated (CMMI) except SPICE is software-oriented and CMMI is more generic.

SPICE is comprised of eight parts as shown in the following diagram:



SPICE identifies five levels of process capability (which are very similar to the five levels of process maturity found in SEI's CMMI).

0 Not-Performed

No common features — a general failure to perform the base practices in the process — no easily identifiable work products or outputs of the process.

1 Performed-Informally

Base practices of the processes are generally performed. Performance depends on individual knowledge and effort. There are identifiable work products of the process.

2 Planned and Tracked

Performance of the base practices in the process is planned and tracked and moving toward a well-defined process. Work products conform to specified requirements.

3 Well Defined

Base practices are performed according to a well-defined process using approved, tailored versions of organization-wide standard and documented processes.

4 Quantitatively Controlled

Detailed measures of performance are collected and analyzed. This leads to a quantitative understanding of process capability and ability to predict performance.

5 Continuously Improving

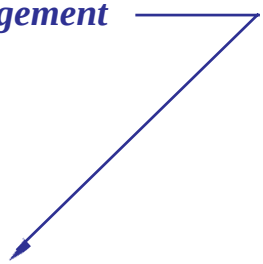
Quantitative process effectiveness and efficiency goals are established. Continuous process improvement against these goals is enabled by quantitative feedback.

Software Process Improvement and Capability dEtermination; http://www-sql.gu.edu.au/spice/, Nov, 2002

Instructions for configuration management reside in 6.4 of Part 2:

Part 2 — Model for process management
6 — Base practices
6.1 Customer-Supplier process (CUS)
6.2 Engineering process (ENG)
6.3 Project process (PRO)
6.4 Support process (SUP)
- ***SUP.1 Develop documentation***
- ***SUP.2 Perform configuration management***
- ***SUP.3 Perform quality assurance***
- ***SUP.4 Perform problem resolution***
- ***SUP.5 Perform peer reviews***

6.5 Organization process (ORG)



SUP.2.1 Establish configuration management library system that provides
- ***storage and retrieval of configuration items (and their versions);***
- ***sharing and transfer of configuration items between groups;***
- ***recovery of archive versions of configuration items;***
- ***correct creation of products from the library.***

SUP.2.2 Identify configuration items (products to be placed under CM) —
- ***requirements, designs, code, tests;***
- ***other product baselines (e.g., user documentation);***
- ***software project plans;***
- ***standards and procedures.***

SUP.2.3 Maintain configuration item descriptions for each CI —
- ***its decomposition into lower level configuration components;***
- ***the person responsible for each item;***
- ***when placed under configuration management.***

SUP.2.4 Manage change requests and problem reports for all CIs.

SUP.2.5 Control changes and maintain integrity of CM items in library.

SUP.2.6 Build product releases from CIs in the library.

SUP.2.7 Maintain configuration item history.

SUP.2.8 Report configuration status and results of above activities.

Software Process Improvement and Capability dEtermination; http://www-sql.gu.edu.au/spice/, Nov, 2002

In summary, SPICE describes how to assess and improve existing software-oriented processes and also determine their capability.

In addition, SPICE does a good job of identifying the major processes, supporting processes (which include CM) and key elements therein.

The role of CM is clearly summarized by SUPs 2.1 through 2.8.

SPICE uses the traditional configuration identification, change control and status accounting activities as the framework for CM.

SPICE also identifies products placed under CM as configuration items (CIs).

SPICE, however, identifies anything placed into the CM library as a configuration item. This includes supporting documents, project plans and records.

Methods for measuring CM process capability must be questioned since it is clear that the *ability to accommodate change* is not an underlying objective.

It is difficult, if not impossible, to maintain a high level of integrity in CIs (per SUP.2.5) when using a change process that is not fast and efficient.

Ability to maintain consistency between the product definition, the product configuration and the CM records is compromised accordingly.

Placement of all items, documents, plans and records in the same CM library does not, in itself, ensure their integrity.

Identifying *the person responsible for each CI* does not ensure integrity either.

The appropriate business process infrastructure (or CM process) is missing.

CM, AS REINVENTED, UNDER CMII

Configuration management is a multifaceted process with many elements. Those elements are often fragmented and may exist under various names.

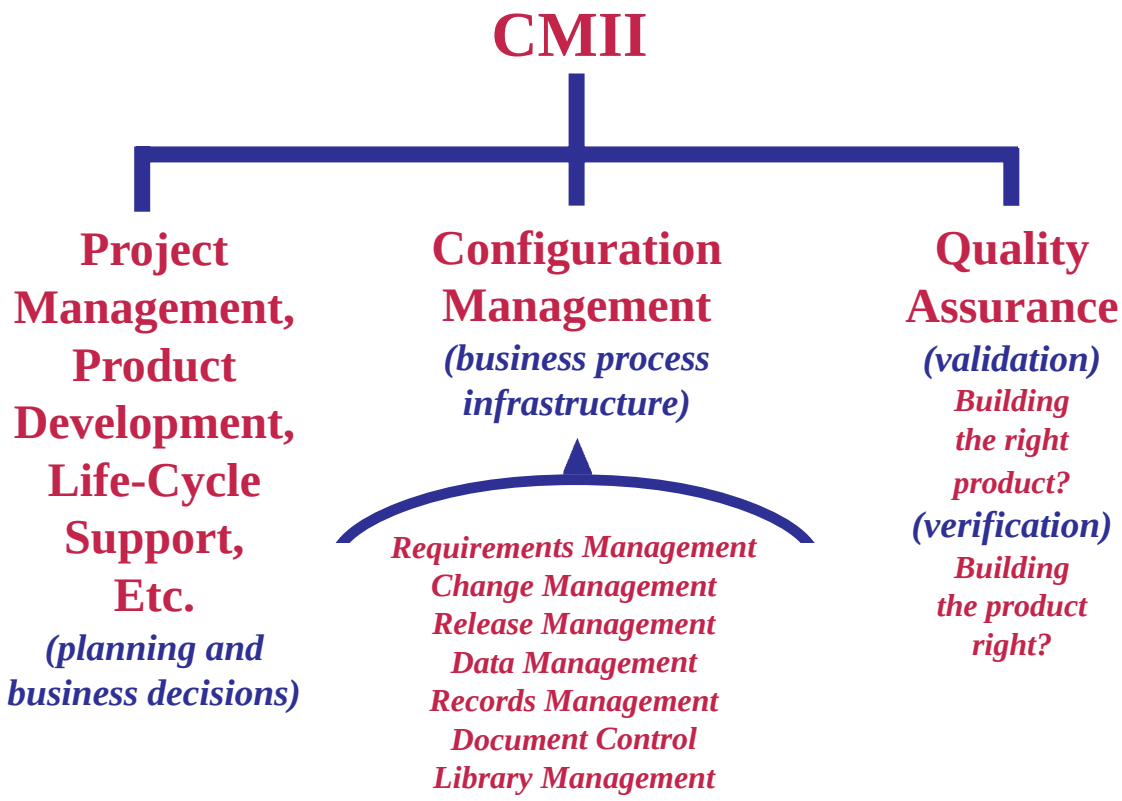
Each element is important in its own way. With CMII, they are brought together under one umbrella and integrated into one cohesive unit.

Overall effectiveness, before and after integration, is measured by the ability to accommodate change and keep released information clear, concise and valid.

As information integrity improves, the need for corrective action declines. As corrective action declines, real improvements become increasingly robust.

To "reinvent CM" is to provide a better *business process infrastructure* and thereby enable other core business processes to be more reliable and efficient.

The reinvented CM process provides an essential infrastructure that is missing in both SPICE and SEI's capability maturity model integrated (CMMI).



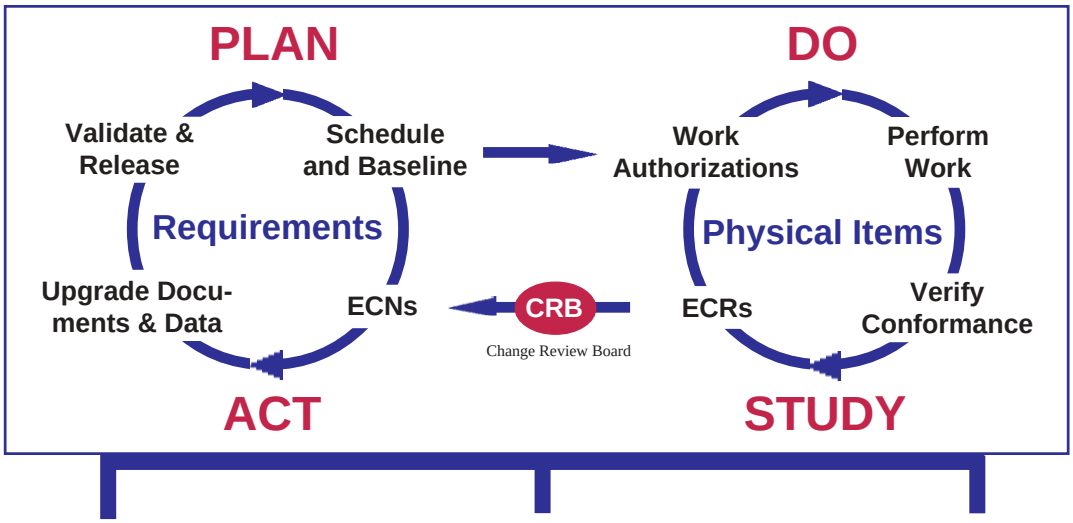
PROJECT MANAGEMENT: TWO CYCLES, NOT ONE

The project management cycle (plan, do, study and act) is ideal for demonstrating the proper role of CM and how key elements should be integrated.

First, it must be recognized that project management is two cycles, not one. A requirements cycle coexists with a physical item cycle.

The physical item cycle is driven by the requirements cycle. Quality assurance activities (validation and verification) have their proper place in each cycle.

Requirements must lead and physical items must conform. A fast and efficient change process is a prerequisite.



Project Management, Product Development, Life-Cycle Support, Etc.
(planning and business decisions)

Configuration Management
(business process infrastructure)

Requirements Management
Change Management
Release Management
Data Management
Records Management
Document Control
Library Management

Quality Assurance
(validation)
Building the right product?
(verification)
Building the product right?

KEYS TO CM AND INFORMATION MANAGEMENT

Again, a change process cannot be fast and efficient change if the information being changed is not properly identified, structured, linked and owned.

Baselines are the ideal place to maintain and display the structure for each model and the linkages from each item at each level to its supporting documents.

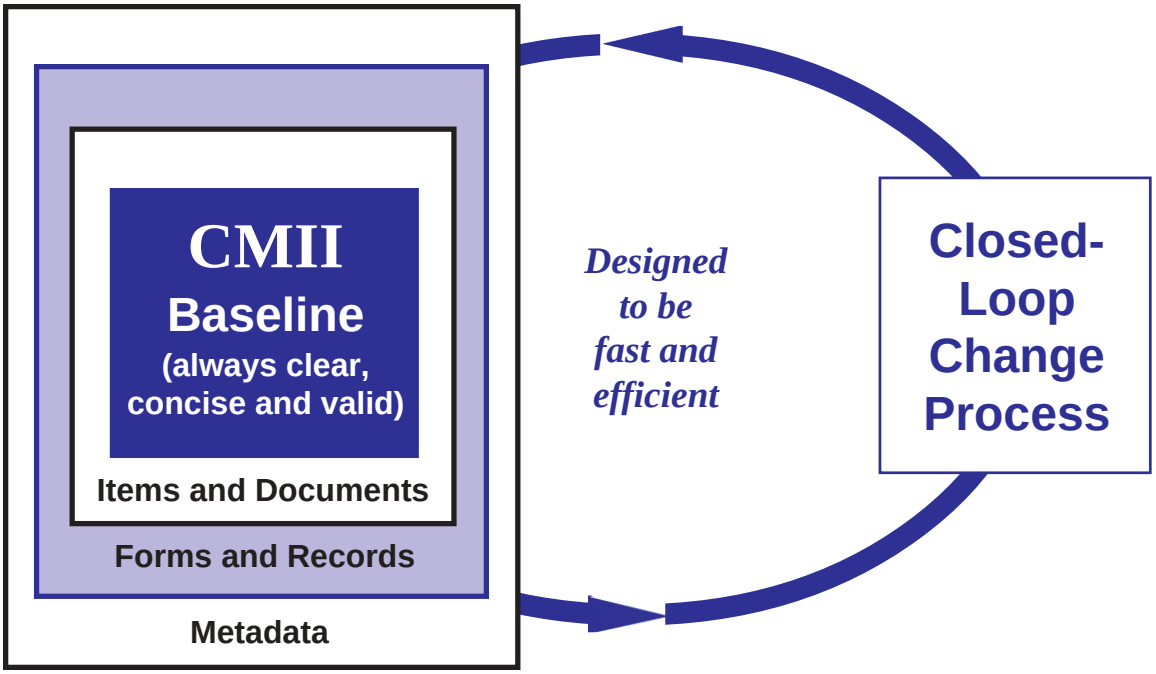
Physical item hierarchies provide the ideal framework for dividing huge amounts of design and process information into manageable increments.

Once baselines and their content are properly established, it is then possible to design the change process in a manner that optimizes speed and integrity.

The appropriate change process is closed-loop (to enhance integrity) and includes a fast-track feature (to enhance speed).

Speed and integrity are further optimized by ensuring that each information set is co-owned by its assigned creator and one or more designated users.

The potential benefits of SPICE will remain compromised until its CM component is upgraded accordingly.



Institute of
Configuration
Management

www.icmhq.com
(480) 998-8600