SOFTWARE EXCELLENCE ACADEMY
THE WESTFALLTEAM

Software Configuration Management Audits
Part 3 – Physical Configuration Audits (PCA)

By Linda Westfall
lwestfall@westfallteam.com

In the first part of this article, we introduced the three different types of Software Configuration Management Audit:

- Functional Configuration Audit (FCA)
- Physical Configuration Audit (PCA)
- In-Process SCM Audits

We also talked about when these audits occur in the software development life cycle. The second part of this article focused on Functional Configuration Management Audits.

This third part of the article talks about Physical Configuration Audits (PCA) and their purpose. It will also provide examples of checklists that could be used during PCA evaluations and suggests evidence-gathering techniques for each item in those checklists.

Purpose of a Physical Configuration Audit (PCA)

According to the ISO/IEC/IEEE *Systems and Software Engineering— Vocabulary* (ISO/IEC/ IEEE 2010), a physical configuration audit (PCA) is "an audit conducted to verify that each configuration item, as built, conforms to the technical documentation that defines it." A PCA verifies that:

- All items identified as being part of the configuration are present in the product baseline
- The correct version and revision of each item is included in the product baseline
- Each item corresponds to the contained in the baseline's configuration status report

A PCA is performed to provide an independent evaluation that the software, as implemented, has been described adequately in the documentation that will be delivered with it and that the software and its documentation have been captured in the software configuration status accounting records and are ready for delivery. Finally, the PCA may also be used to evaluate adherence to legal obligations, including licensing, royalties, and export compliance requirements.

Like the Functional Configuration Audit (FCA), a PCA is conducted at least once during the life cycle, typically just before the final ready-to-beta-test or ready-to-ship review, and provides input information into those reviews. However, PCAs can also be conducted throughout the life cycle at checkpoints to verify the proper transition of the requirements into the subsequent successor work products. The PCA is typically held either in conjunction with the FCA or soon after the FCA (once any issues identified during the FCA are resolved). A PCA is essentially a review of the software configuration status accounting data to make certain that the software products and their deliverable documentation are appropriately baselined and properly built prior to release to beta testing or operations, depending on where in the life cycle the PCA is conducted.

Checklist Item Suggestions for Evidence-Gathering Techniques

Table 1 illustrates an example of a checklist and lists possible objective evidence-gathering techniques for each checklist item that would be used for a PCA conducted at any baseline or major milestone.

Table 2 illustrates an example of a checklist and lists possible objective evidence-gathering techniques for each checklist item that would be used for a PCA conducted at the product/release baseline.These checklist items would be used in addition to the checklist items in table a.

While several suggested evidence-gathering techniques are listed for each checklist item, the level of rigor chosen for the audit will dictate which of these techniques (or other techniques) will actually be used.

Table 1 – Example Checklist and Evidence-Gathering Techniques Used During Any PCA

Checklist Item	Suggestions for Evidence Gathering Techniques
1. Has each nonconformance or noncompliance from the FCA been appropriately resolved?	• Review findings from the FCA audit report, associated corrective actions, follow-up and verification records to evaluate adequacy of actions taken (or appropriate approved waivers/deviations exist).
2. Have all of the identified Configuration Items (e.g., source code, documentation, etc.) been baselined?	• Sample a set of Configuration Items and evaluate them against configuration status accounting records.
3. Do all of the Configuration Items meet workmanship standards?	• Sample a set of source code modules and evaluate them against the coding standards. • Sample a set of deliverable documents (or sections/pages of those documents) and evaluate them against documentation standards.
4. Has the software been built from the correct components and in accordance with the specification?	• Evaluate the build records against the configuration status accounting information to ensure that the correct version and revision of each module was included in the build. • Evaluate any patches/temporary fixes made to the software to ensure their completeness and correctness. • Sample a set of design elements from the architectural design and trace them to their associated detailed design elements and source code. Compare those elements with the build records to evaluate for completeness and consistency with the as built software.

Table 2 – Example of Additional Checklist Item and Evidence-Gathering Techniques Used for PCA at Product/Release Baseline

1. Is the deliverable documentation set complete?	• Evaluate the master copy of each document against the configuration status accounting information to ensure that the correct version and revision of each document sub-component (e.g., chapter, section, figure) is included in the document. • Sample the set of copied documents ready for shipment and review them for completeness and quality against the master copy. • Evaluate the version description document against the build records for completeness and consistency. • Compare the current build records to the build records from the last release to identify changed components. Evaluate this list of changed components against the version description document to evaluate the version description document's completeness and consistency.
2. Does the actual system delivery media conform to specification? Has the delivery media been appropriately marked/labeled?	• Evaluate the items on the master media against the required software deliverables (executables, help files, data) to ensure the correct versions and revisions were included. • Sample a set of copied media ready for shipment and review them for completeness and quality against the master media. • Sample a set of copied media ready for shipment and review their marking/labeling against specification.
3. Do the deliverables for shipment match the list of required deliverables?	• Evaluate the packing list against the list of documented deliverables to ensure completeness. • Sample a set of ready-to-ship packages and evaluate them against the packing list to ensure that media (i.e., CD, disks, tape), documentation and other deliverables are included in each package.
4. Have 3rd party licensing requirements been met?	• Evaluate the build records against configuration status accounting information to identify 3rd party components and license information to confirm adequate numbers of licenses exist.
5. Have export compliance requirements been met?	• Evaluate the build records against configuration status accounting information to identify components with export restrictions and confirmed export compliance.