

# **Software Standards**

Przemek (Pshemek) Lach  
SEng 371

# What are we doing?

- Motivation for standards.
- Standards in the wild.
- Tetra Tech.
- What's in a standard?

# **Motivation for Standards**

# Motivation for Standards

- Sometimes you just have to:
  - Contractual (winning bid)
  - Physical (electrical outlet)
- Sometimes it is just a good idea:
  - They help guide you in the right direction. In other words, they help you ask the right questions.
    - You're not the first to write software.
    - Others have been through this pain and it is, in part, why some standards exist.
  - In general, it's a good means for transferring 'best practice' in software engineering.

# Motivation for Standards (cont'd)

Standard adherence may be a contractual obligation when software companies bid on contracts.

- Aerospace

- *AS9100 - "Companies that design, develop, and/or produce aviation, space and defense products; and by organizations providing post-delivery support, including the provision of maintenance, spare parts, or materials, for their own products."*

# Motivation for Standards (cont'd)

- **Military**
  - MIL-STD-498 - consistent requirements for the way software was developed and documented.
- **Medicine**
  - IEC 62304 - life cycle requirements for the development of medical device software.
- and many more...

# Motivation for Standards (cont'd)

- Your clients may require compliance:
  - Federal/Provincial Law - government stipulates that any software used by the government must adhere to certain standards for reasons of public safety.
  - Risk Mitigation - a company or organization may want to switch development teams. If teams follow standard practices then it is easier to bring new teams on board.
  - Worldwide projects - large projects sometimes employ several companies. Standards ensure that the process is smooth and that all the pieces 'fit'.

# Motivation for Standards (cont'd)

- You may want to follow standards:
  - As mentioned earlier, standards will help to guide you down the right path
  - You may ask yourself questions you may have otherwise missed.
  - If you apply at a company that is required to follow standards, having some standards knowledge on your resume is a good thing.



# Motivation for Standards (cont'd)

- Common excuses:
  - My project is too small.
  - There are only two developers.
  - We don't have time.
- These are common excuses that, in the end, end up costing you more time and money.
- e.g., start-ups are notorious for this.
  - Launch deadlines are very short (~6 months)
  - Thorough testing is ignored.
  - Software maintenance is not even on the horizon,
  - But then the product takes off on you're in trouble.

# **Standards in the Wild**

# Standards in the Wild

*Source: Anatol W. Kark*

- ISO/IEC/IEEE 15288, System Life Cycle Processes adopted by:
  - INCOSE
    - Adoption for processes and organization of SE Handbook
    - Referred to for SE Certification
  - US DoD
    - Adoption of ISO/IEC 15288 as acceptable standard for system life cycle processes
    - Defense Acquisition Guide (chap 4) aligns to and leverages.

# Standards in the Wild

*Source: Anatol W. Kark*

- NATO
  - Adopted as an acceptable standard
- US Dept of Transportation / CA Dept of Transportation.
  - Referenced and leveraged in Systems Engineering Guidebook for Intelligent Transportation Systems.
- Swedish Defence Material Administration and Swedish Armed Forces.
  - Process standard for their value added production.

# Standards in the Wild

*Source: Anatol W. Kark*

- Dutch Dept of Transportation.
- Defense companies
  - Adopted by many defense companies.
  - Used to form process requirements.
- Many others
  - Many presentations on usage from various type and size companies.

# Standards in the Wild (related std)

Source: Anatol W. Kark

Document ID	Document Title	Comment
ISO/IEC/IEEE 15288:2008	System Life Cycle Processes	Revision 2. Wide adoption including NATO, DoD, INCOSE ... 2008 Editions of 12207 and 15288 were the FIRST step in harmonization
ISO/IEC/IEEE 12207:2008	Software Life Cycle Processes	<u>Revision 2</u> . SW viewed as System Element
ISO/IEC 24728- 1:2010	Guide for life cycle management	Framework Document – available free at <a href="http://www.jtc1.org">www.jtc1.org</a>
ISO/IEC/IEEE 42010:2012	Architecture Description	42030 in development – Architecture Evaluation
ISO/IEC/IEEE 24765:2010	Systems and Software Engineering Vocabulary	Includes terms and definitions from ISO/IEC, IEEE, PMI
ISO/IEC/IEEE 15026	System Assurance	Four part standard. Published 2010-2012.
ISO/IEC 2700x	SW Security Standards	E.g., 27001 – Security Management System
SEH	INCOSE SE Handbook	Serves as guide to 15288 & basis for SE Cert.
<u>SEBoK</u>	SE Body of Knowledge	Release in Sep 2012
ISO/IEC/IEEE 26702 (will be 24748-4)	Management of the Systems Engineering Process	Aka IEEE Std 1220 – under revision – SE Planning and SEMP focus
EIA 632	Engineering of a System	Under revision
ISO/IEC TR 2655x	Tools & Methods for Product Lines	Technical Report series under development

# **Tetra Tech**

Standards Don't Just Sit On The Shelf

# Tetra Tech

Source: [http://www.iso.org/iso/home/news\\_index/news\\_archive/news.htm?refid=Ref1708](http://www.iso.org/iso/home/news_index/news_archive/news.htm?refid=Ref1708)

Tetra Tech is a leading provider of consulting, engineering, programme management, construction management, and technical services. The company supports government and commercial clients by providing innovative solutions to complex problems focused on water, environment, energy, infrastructure, and natural resources.



# Tetra Tech

Source: [http://www.iso.org/iso/home/news\\_index/news\\_archive/news.htm?refid=Ref1708](http://www.iso.org/iso/home/news_index/news_archive/news.htm?refid=Ref1708)

*"While it's true that standardization is more visible in the manufacturing sector, particularly in terms of achieving product compatibility to gain access to new markets, it is also fundamental to the engineering sector."*

*Denis Tremblay  
General Manager, Energy Unit*

# Tetra Tech

Source: [http://www.iso.org/iso/home/news\\_index/news\\_archive/news.htm?refid=Ref1708](http://www.iso.org/iso/home/news_index/news_archive/news.htm?refid=Ref1708)

*"International Standards serve as a bridging mechanism. They constitute a tool for a company, enabling it to overcome barriers and gain greater market access. The more standards are used, the greater their level of competitiveness. This, in the end, gives the ultimate client a big payoff in terms of cost savings."*

Denis Tremblay  
General Manager, Energy Unit

# Tetra Tech

Source: [http://www.iso.org/iso/home/news\\_index/news\\_archive/news.htm?refid=Ref1708](http://www.iso.org/iso/home/news_index/news_archive/news.htm?refid=Ref1708)

*"Just as barriers to market entry in the manufacturing sector can be costly, our customers in the engineering field also have stringent quality requirements, which they link to specific standards, and to which they want us to conform. And when a company makes the effort to meet an International Standard, the payoff is great."*

*Denis Tremblay  
General Manager, Energy Unit*

# **What's in a Standard?**

IEEE 1219-1993

Standard on Software Maintenance

# IEEE 1219-1993 (why?)

- Almost all software spends its 'working years' in the software maintenance phase.
- Software maintenance plays a large part in software evolution.
- Some of standards may seem too verbose or perhaps overkill but you'd be surprised how useful it is to have a common 'language' when it comes to software development.

# IEEE 1219-1993 (scope)

- Describes an iterative process for managing and executing software maintenance activities.
- Applies to new software under development and for existing software products.

# IEEE 1219-1993 (scope)

- Prescribes requirements for process, control, and management of the planning, executing, and documentation of software maintenance activities.
- Does not assume the use of any particular development model.

# IEEE 1219-1993 (definitions)

- **Adaptive maintenance** - modification of a software product performed after delivery to keep a computer program usable in a change or changing environment.
- **Corrective maintenance** - reactive modification of a software product performed after delivery to the correct discovered fault.



# IEEE 1219-1993 (definitions)

- **Customer** - The person, or persons, for whom the product is intended, and usually who decides the requirements.
- **Emergency Maintenance** - unscheduled corrective maintenance performed to keep a system operational.

# IEEE 1219-1993 (definitions cont'd)

- **Interoperability Testing** - Tested conducted to ensure that a modified system retains the capability of exchanging information with systems of different types, and how it uses that information.
- **Modification Request** - A generic term that includes the forms associated with the various problem reporting documents and configuration change control documents.

# IEEE 1219-1993 (definitions cont'd)

- **Perfective Maintenance** - modification of a software product after delivery to improve performance or maintainability.
- **Project** - A subsystem that is subject to maintenance activity.

# IEEE 1219-1993 (definitions cont'd)

- **Regression Test** - Retesting to detect faults introduced by modification.
- **Repository** - A collection of all software related artifacts belonging to a system and the location/format in which such a collection is stored.

# **IEEE 1219-1993 (acronyms)**

**SCM** - Software configuration management.

**CSA** - Configuration status accounting.

**FCA** - Functional configuration audit.

**PCA** - Physical configuration audit.

**PDL** - Program design language.

**SQA** - Software quality assurance.

**SCR** - System/software change request.

**V&V** - Verification and validation.

**VDD** - Version description document.

# IEEE 1219-1993 (phases)

- The standard defines the software change process for each of these phases:
  - a. Problem identification and classification.*
  - b. Analysis.*
  - c. Design.*
  - d. Implementation.*
  - e. Regression/System testing.*
  - f. Acceptance testing.*
  - g. Delivery.*

# IEEE 1219-1993 (phases cont'd)

- For each phase we have inputs, process, control, and output.
  - a. *Problem identification and classification.*
  - b. *Analysis.*
  - c. *Design.*
  - d. *Implementation.*
  - e. *Regression/System testing.*
  - f. *Acceptance testing.*
  - g. *Delivery.*

# IEEE 1219-1993 (phases cont'd)

*(a) Problem identification and classification.*

- Software modification requests (MRs) are identified, classified, and assigned an initial priority ranking.
- Classification can be from the following maintenance cycles
  - corrective
  - adaptive
  - perfective
  - emergency



# IEEE 1219-1993 (phases cont'd)

*(a) Problem identification and classification.*

- **Input Modification request (MR).**
- **Process - activities that must occur:**
  - Assign an identification number.
  - Classify the type of maintenance.
  - Analyze the modification to determine whether to accept, reject, or further evaluate.
  - Make a preliminary estimate of the modification size/ magnitude.
  - Prioritize the modification.
  - Assign MR to a block of modifications scheduled for implementation.

# IEEE 1219-1993 (phases cont'd)

*(a) Problem identification and classification.*

- **Control** - MR and process determinations shall be uniquely identified and entered into a repository.

# IEEE 1219-1993 (phases cont'd)

*(a) Problem identification and classification.*

- **Output** - The output of this process shall be stored in a repository that contain the following items:
  - Statement of the problem or new requirement.
  - Problem or requirement evaluation.
  - Classification of the type of maintenance required.
  - Initial priority.
  - Verification data (for corrective modifications).
  - Initial estimate of resources required to modify the existing system.

# IEEE 1219-1993 (phases cont'd)

- a. *Problem identification and classification*
- b. ***Analysis.***
- c. *Design.*
- d. *Implementation.*
- e. *Regression/System testing.*
- f. *Acceptance testing.*
- g. *Delivery.*

# IEEE 1219-1993 (phases cont'd)

## *(b) Analysis*

- Analysis phase shall use the repository information and the MR validated in phase 1, along with system and project documentation, to study the feasibility and scope of the modification
- To devise a preliminary plan for design, implementation, test, and delivery.
- Classification can be from the following maintenance types: corrective, adaptive, perfective, and emergency.

# IEEE 1219-1993 (phases cont'd)

*(b) Analysis*

- **Input**

- Validated MR
- Initial resource estimate and other repository information
- Project and system documentation, if available

# IEEE 1219-1993 (phases cont'd)

## *(b) Analysis*

- **Process - Analysis** is an iterative process having at least two components
  - A feasibility analysis
  - A detailed analysis
    - If the documentation is not available or is insufficient and the source code is the only reliable representation of the software system, reverse engineering is recommended.

# IEEE 1219-1993 (phases cont'd)

## *(b) Analysis*

- **Feasibility Analysis** - shall be performed for MR and a feasibility report (FR) shall be prepared. This FR should contain the following
  - Impact of the modification
  - Alternate solutions, including prototyping
  - Analysis of conversion requirements
  - Safety and security implications
  - Human factors
  - Short-term and long-term costs
  - Value of the benefit of making the modification



# IEEE 1219-1993 (phases cont'd)

## *(b) Analysis*

- **Detailed Analysis** - provide the following:
  - Define firm requirements for the modification.
  - Identify the elements of modification.
  - Identify safety and security issues.
  - Devise a test strategy.
  - Develop an implementation plan.

# IEEE 1219-1993 (phases cont'd)

## *(b) Analysis*

- **Control** - Control of analysis shall include the following:
  - Retrieval of the relevant version of project and system documentation from the configuration control function of the organization.
  - Review of the proposed changes and engineering analysis to assess technical and economic feasibility, and assess correctness.
  - Identification of safety and security issues
  - Consideration of the integration of the proposed change within the existing software.

# IEEE 1219-1993 (phases cont'd)

## *(b) Analysis*

- Verification that all appropriate analysis and project documentation is updated and properly controlled
- Verification that the test function of the organization is providing a strategy for testing the change(s), and that the change schedule can support the proposed test strategy
- Review of the resource estimates and schedules and verification of their accuracy

# IEEE 1219-1993 (phases cont'd)

## *(b) Analysis*

- Technical review to select the problem reports and proposed enhancements to be implemented in the new release. The list of changes shall be documented.

# IEEE 1219-1993 (phases cont'd)

## *(b) Analysis*

- **Output** - Shall include the following
  - Rate Feasibility report for MRs.
  - Detailed analysis report.
  - Updated requirements (including traceability list).
  - Preliminary modification list.
  - Test strategy.
  - Implementation plan.

# IEEE 1219-1993 (phases cont'd)

- a. *Problem identification and classification*
- b. *Analysis.*
- c. ***Design.***
- d. *Implementation.*
- e. *Regression/System testing.*
- f. *Acceptance testing.*
- g. *Delivery.*

# IEEE 1219-1993 (phases cont'd)

*(c) Design*

In the design phase, all current system and project documentation, existing software and databases, and the output of the analysis phase (including detailed analysis, statements of requirements, identification of elements affected, test strategy, and implementation plan) shall be used to design the modification to the system.

# IEEE 1219-1993 (phases cont'd)

*(c) Design*

- **Input** - shall include the following:
  - Analysis phase output, including
    - Detailed analysis
    - Updated statement of requirements
    - Preliminary modification list
    - Test strategy
    - Implementation plan
  - System and project documentation.
  - Existing source code, comments, and databases.



# IEEE 1219-1993 (phases cont'd)

*(c) Design*

- **Process** - shall include the following
  - Identifying affected software modules.
  - Modifying software module documentation (e.g., data and control flow diagrams, schematics, etc.)
  - Creating test cases for the new design, including
  - Safety and security issues.
  - Identifying/creating regression tests.
  - Identifying documentation (system/user) update requirements.
  - Updating modification list.

# IEEE 1219-1993 (phases cont'd)

*(c) Design*

- **Control** - shall include the following
  - Conduct software inspection of the design in compliance with IEEE Std 1028-1988.
  - Verify that the new design/requirement is documented as a software change authorization (SCA), as per IEEE Std 1042-1987.
  - Verify the inclusion of new design material, including safety and security issues.
  - Verify that the appropriate test documentation has been updated.
  - Complete the traceability of the requirements to the design.

# IEEE 1219-1993 (phases cont'd)

*(c) Design*

- **Output** - shall include the following
  - Revised modification list.
  - Updated design baseline.
  - Updated test plans.
  - Revised detailed analysis.
  - Verified requirements.
  - Revised implementation plan.
  - A list of documented constraints and risks.

# IEEE 1219-1993 (phases cont'd)

- a. *Problem identification and classification*
- b. *Analysis.*
- c. *Design.*
- d. ***Implementation.***
- e. *Regression/System testing.*
- f. *Acceptance testing.*
- g. *Delivery.*

# IEEE 1219-1993 (phases cont'd)

## *(d) Implementation*

In the implementation phase, the results of the design phase, the current source code, and system documentation (i.e., the entire system as updated by the analysis and design phases) shall be used to drive the implementation effort.

# IEEE 1219-1993 (phases cont'd)

*(d) Implementation*

- **Input** - shall include the following
  - Results of the design phase.
  - Current source code, comments, and databases.
  - Project and system documentation.

# IEEE 1219-1993 (phases cont'd)

## *(d) Implementation*

- **Process** - The implementation phase shall include the following four sub-processes, which may be repeated in an incremental, iterative approach
  - Coding and unit testing
    - Implement the change into the code and perform unit testing and other appropriate SQA and V&V processes

# IEEE 1219-1993 (phases cont'd)

## *(d) Implementation*

- Integration
  - the modified software shall be integrated with the system and integration and regression tests shall be refined and performed.
- Risk analysis
  - risk analysis and review shall be performed periodically during the phase rather than at its end, as in the design and analysis phases.
- Test readiness review (follow IEEE Std 1028-1988 Std).



# IEEE 1219-1993 (phases cont'd)

## *(d) Implementation*

### ● **Control**

- Conduct software inspections of the code in compliance with IEEE Std 1028-1988.
- Ensure that unit and integration testing are performed and documented in a software development folder.
- Ensure that test documentation (e.g., test plan, test cases, and test procedures) are either updated or created.
- Identify, document, and resolve any risks exposed during software and test-readiness reviews.

# IEEE 1219-1993 (phases cont'd)

## *(d) Implementation*

- Verify that the new software is placed under software configuration management control.
- Verify that the training and technical documentation have been updated.
- Verify the traceability of the design to the code.

# IEEE 1219-1993 (phases cont'd)

*(d) Implementation*

## ● **Output**

- Updated software.
- Updated design documentation.
- Updated test documentation.
- Updated user documentation.
- Updated training material.
- A statement of risk and impact to users.
- Test readiness review report (see IEEE Std 1028-1988).

# IEEE 1219-1993 (phases cont'd)

- a. *Problem identification and classification*
- b. *Analysis.*
- c. *Design.*
- d. *Implementation.*
- e. ***Regression/System testing.***
- f. *Acceptance testing.*
- g. *Delivery.*

# IEEE 1219-1993 (phases cont'd)

## *(d) System Test*

System testing, as defined in IEEE Std 610.12-1990, shall be performed on the modified system. Regression testing is a part of system testing and shall be performed to validate that the modified code does not introduce faults that did not exist prior to the maintenance activity.

# IEEE 1219-1993 (phases cont'd)

## *(d) System Test*

- **Input** - shall include the following
  - Test-readiness review report.
  - Documentation, which includes:
    - System test plans (IEEE 829-1983).
    - System test cases (IEEE 829-1983).
    - System test procedures (IEEE 829-1983).
    - User manuals.
    - Design.
  - Updated system.

# IEEE 1219-1993 (phases cont'd)

## *(d) System Test*

- **Process** - System tests shall be conducted on a fully integrated system. Testing shall include the performance of
  - System functional test.
  - Interface testing.
  - Regression testing.
  - Test readiness review to assess preparedness for acceptance testing.

# IEEE 1219-1993 (phases cont'd)

## *(d) System Test*

- **Control:**

- System tests shall be conducted by an independent test function
- the test function shall be responsible for reporting the status of the criteria that had been established in the test plan for satisfactory completion of system testing.
- The status shall be reported to the appropriate review committee prior to proceeding to acceptance testing



# IEEE 1219-1993 (phases cont'd)

## *(d) System Test*

- The customer shall participate in the review to ascertain that the maintenance release is ready to begin acceptance testing.

# IEEE 1219-1993 (phases cont'd)

*(d) System Test*

- **Output** - shall include the following
  - Tested and fully integrated system.
  - Test report.
  - Test readiness review report.

# IEEE 1219-1993 (phases cont'd)

- a. *Problem identification and classification*
- b. *Analysis.*
- c. *Design.*
- d. *Implementation.*
- e. *Regression/System testing.*
- f. ***Acceptance testing.***
- g. *Delivery.*

# IEEE 1219-1993 (phases cont'd)

## *(d) Acceptance Test*

Acceptance tests shall be conducted on a fully integrated system. Acceptance tests shall be performed by either the customer, the user of the modification package, or a third party designated by the customer.

# IEEE 1219-1993 (phases cont'd)

*(d) Acceptance Test*

- **Input**

- Test readiness review report.
- Fully integrated system.
- Acceptance test plans.
- Acceptance test cases.
- Acceptance test procedures.

- **Process**

- Perform acceptance tests at the functional level.
- Perform interoperability testing.
- Perform regression testing.

# IEEE 1219-1993 (phases cont'd)

*(d) Acceptance Test*

- **Control**

- Execute acceptance tests.
- Report test results for the functional configuration audit (FCA).
- Conduct functional audit.
- Establish the new system baseline.
- Place the acceptance test documentation under SCM control.

# IEEE 1219-1993 (phases cont'd)

*(d) Acceptance Test*

- **Output**

- New system baseline.
- Functional configuration audit report (see IEEE Std 1028-1988).
- Acceptance test report (see IEEE Std 1042-1987).

# IEEE 1219-1993 (phases cont'd)

- a. *Problem identification and classification*
- b. *Analysis.*
- c. *Design.*
- d. *Implementation.*
- e. *Regression/System testing.*
- f. *Acceptance testing.*
- g. ***Delivery.***



# IEEE 1219-1993 (phases cont'd)

*(d) Delivery*

- **Input** - shall be the fully tested version of the system as represented in the new baseline.
- **—Process**
  - Conduct a physical configuration audit.
  - Notify the user community.
  - Develop an archival version of the system for backup.
  - Perform installation and training at the customer facility.

# IEEE 1219-1993 (phases cont'd)

*(d) Delivery*

- **Control**

- Arrange and document a physical configuration audit.
- Provide system materials for access to users, including replication and distribution
- Complete the version description document (IEEE Std 1042-1987)
- Complete updates to status accounting database.
- Place under SCM control.

# IEEE 1219-1993 (phases cont'd)

*(d) Delivery*

- **Output**

- PCA report (IEEE Std 1028-1988).
- Version description document (VDD).

# Summary

# Summary

- There is a standard for almost anything you can think of.
- Some standards act as guides while others act more as a specification.
- Software related standards help guide you by asking the right questions.
- You've used standards whether you know it or not.
- Guaranteed to see standards in industry.