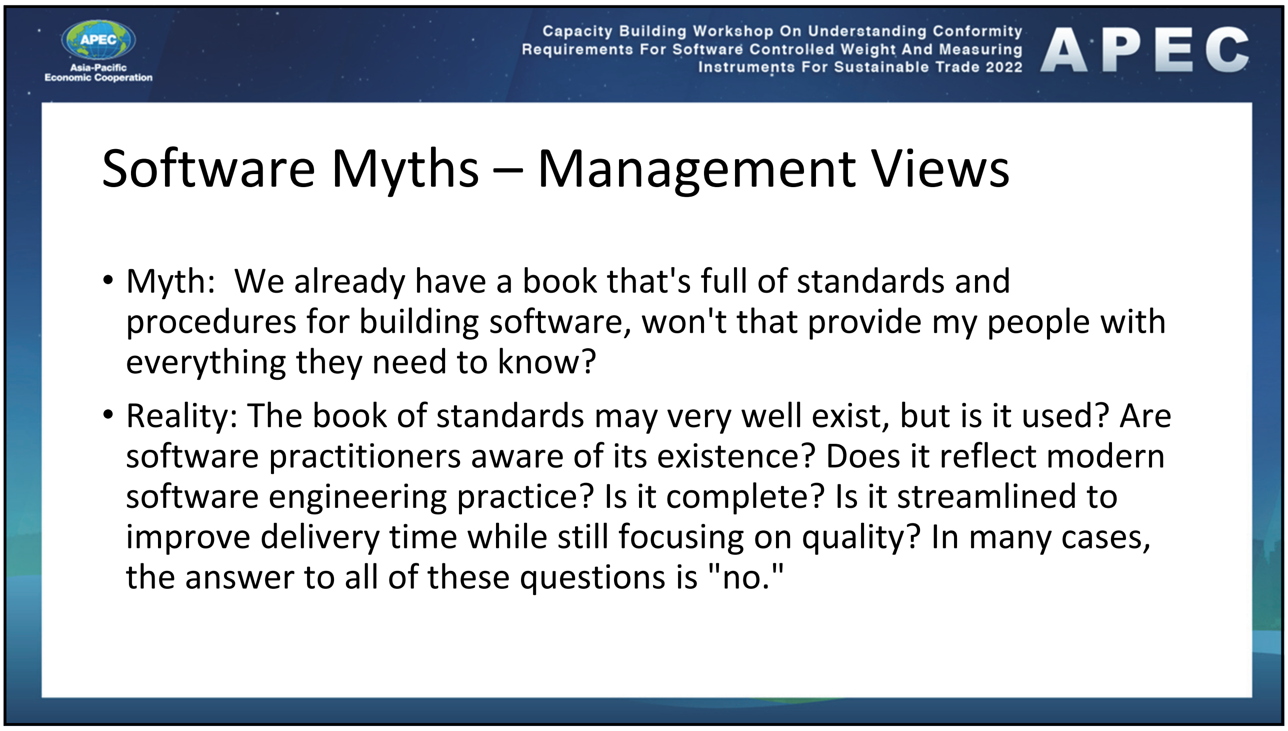
Prof. Dr. Hj. Okfalisa, St
Universitas Islam Negeri Sultan Syarif Kasim Riau

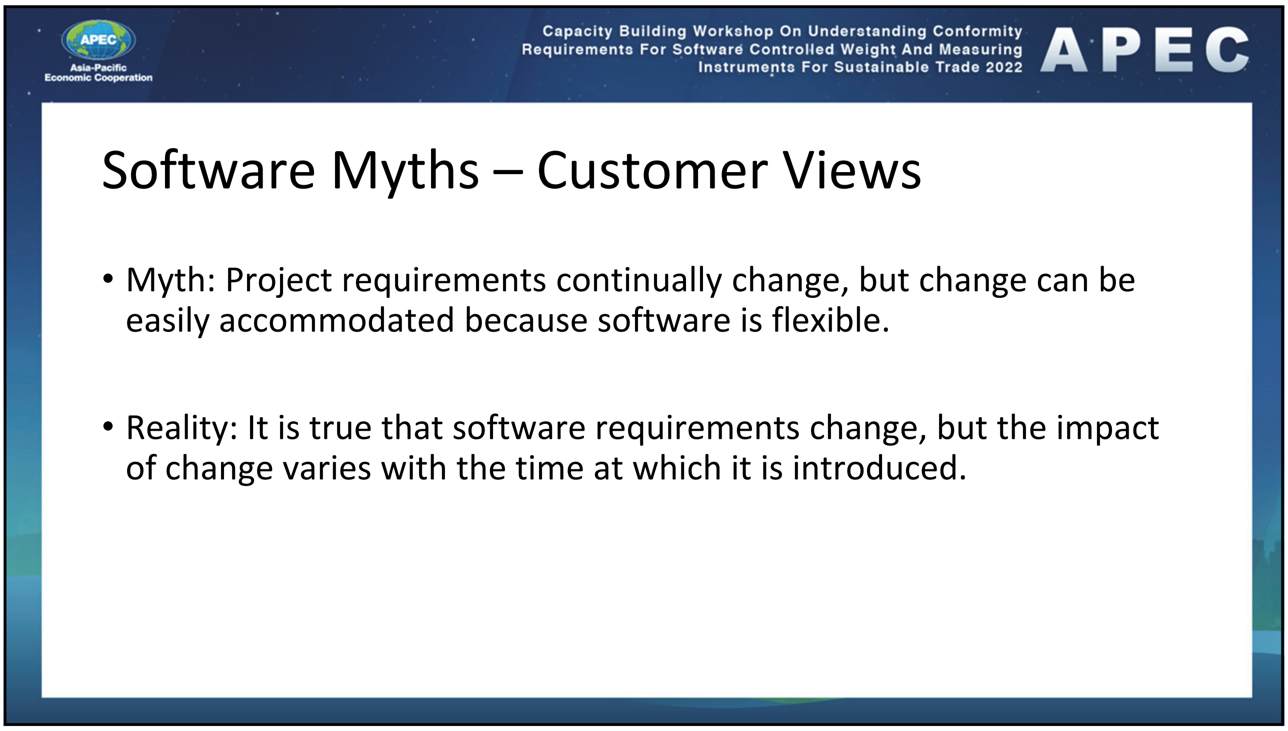
SOFTWARE DOCUMENTATION



Capacity Building Workshop On Understanding Conformity Requirements For Software Controlled Weight And Measuring Instruments For Sustainable Trade 2022

Software Myths – Management Views

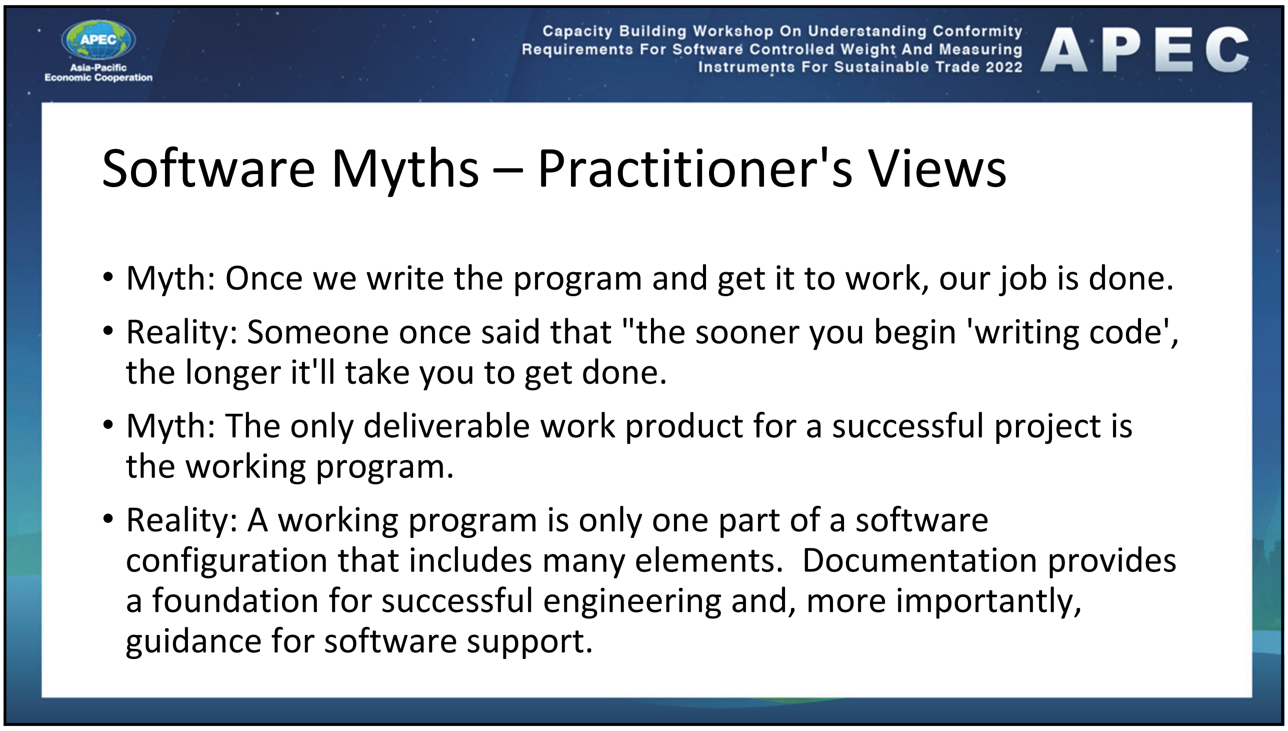
- Myth: We already have a book that's full of standards and procedures for building software, won't that provide my people with everything they need to know?
- Reality: The book of standards may very well exist, but is it used? Are software practitioners aware of its existence? Does it reflect modern software engineering practice? Is it complete? Is it streamlined to improve delivery time while still focusing on quality? In many cases, the answer to all of these questions is "no."



Capacity Building Workshop On Understanding Conformity Requirements For Software Controlled Weight And Measuring Instruments For Sustainable Trade 2022

Software Myths – Customer Views

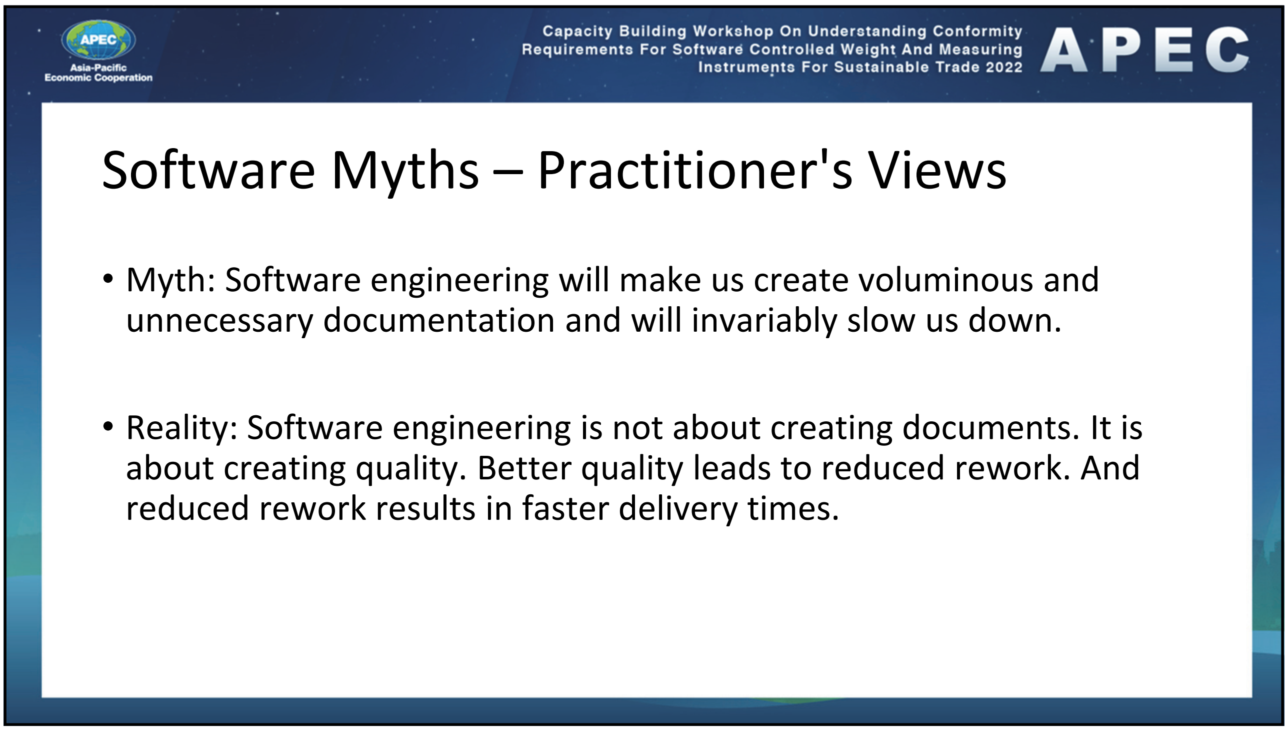
- Myth: Project requirements continually change, but change can be easily accommodated because software is flexible.
- Reality: It is true that software requirements change, but the impact of change varies with the time at which it is introduced.



Capacity Building Workshop On Understanding Conformity Requirements For Software Controlled Weight And Measuring Instruments For Sustainable Trade 2022

Software Myths – Practitioner's Views

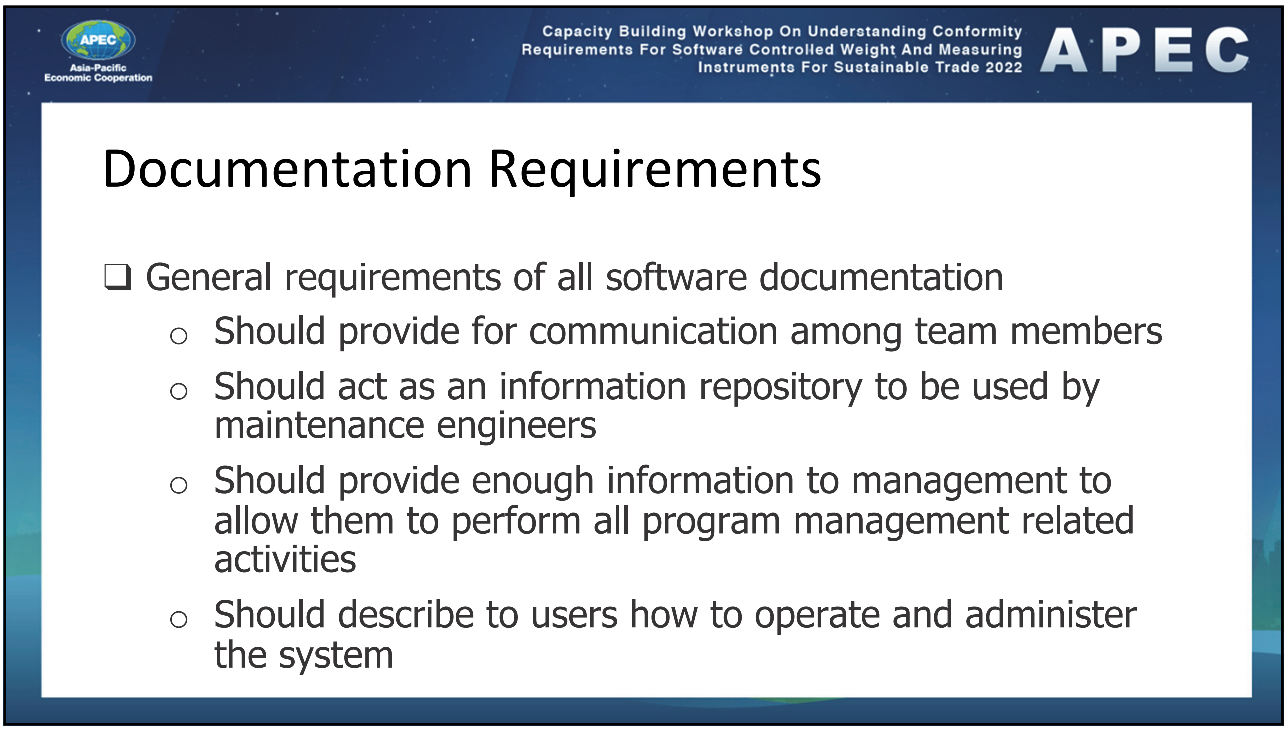
- Myth: Once we write the program and get it to work, our job is done.
- Reality: Someone once said that "the sooner you begin 'writing code', the longer it'll take you to get done.
- Myth: The only deliverable work product for a successful project is the working program.
- Reality: A working program is only one part of a software configuration that includes many elements. Documentation provides a foundation for successful engineering and, more importantly, guidance for software support.



Capacity Building Workshop On Understanding Conformity Requirements For Software Controlled Weight And Measuring Instruments For Sustainable Trade 2022

Software Myths – Practitioner's Views

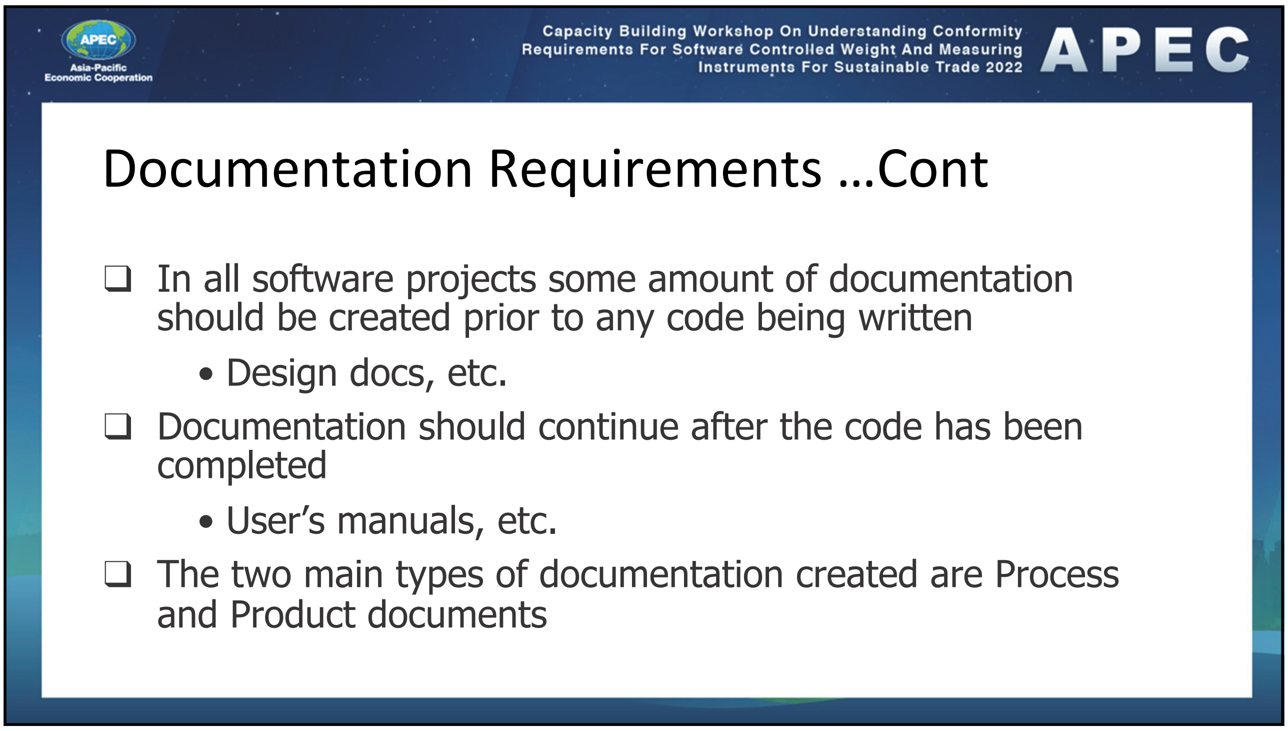
- Myth: Software engineering will make us create voluminous and unnecessary documentation and will invariably slow us down.
- Reality: Software engineering is not about creating documents. It is about creating quality. Better quality leads to reduced rework. And reduced rework results in faster delivery times.



Capacity Building Workshop On Understanding Conformity Requirements For Software Controlled Weight And Measuring Instruments For Sustainable Trade 2022

Documentation Requirements

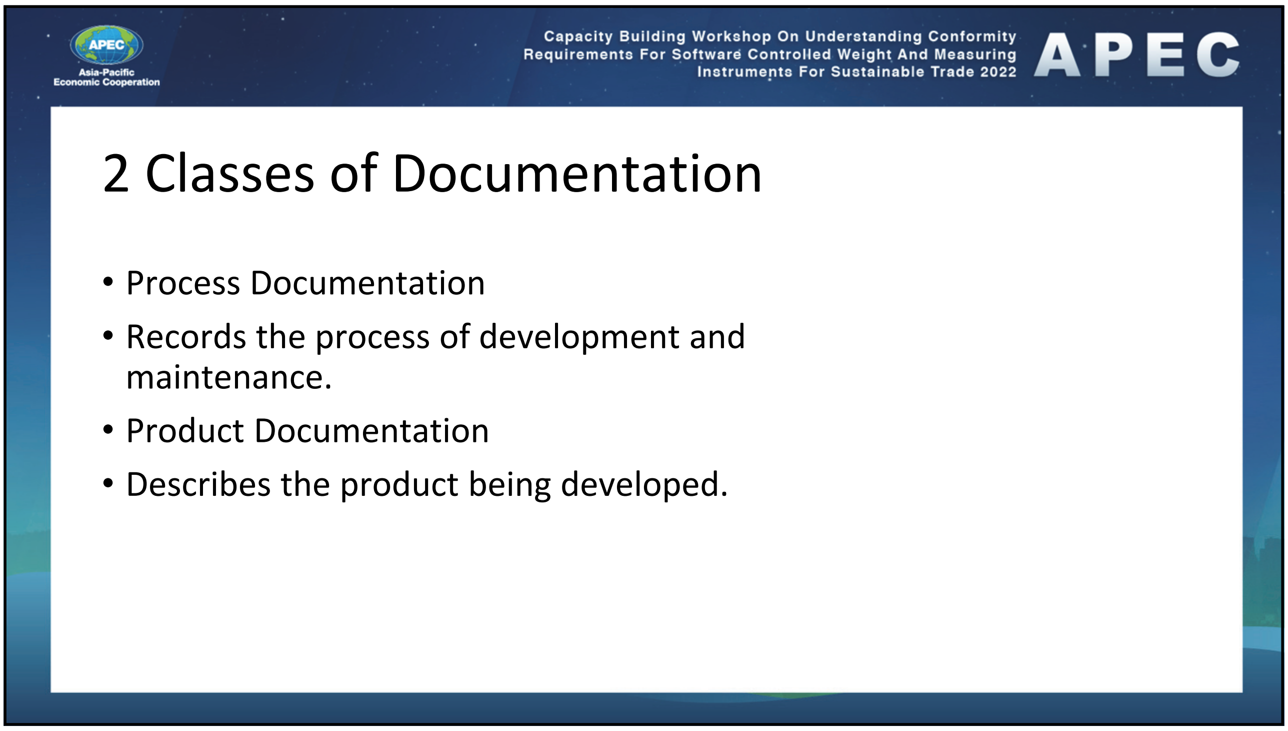
❑ General requirements of all software documentation
  - Should provide for communication among team members
  - Should act as an information repository to be used by maintenance engineers
  - Should provide enough information to management to allow them to perform all program management related activities
  - Should describe to users how to operate and administer the system



Capacity Building Workshop On Understanding Conformity Requirements For Software Controlled Weight And Measuring Instruments For Sustainable Trade 2022

Documentation Requirements …Cont

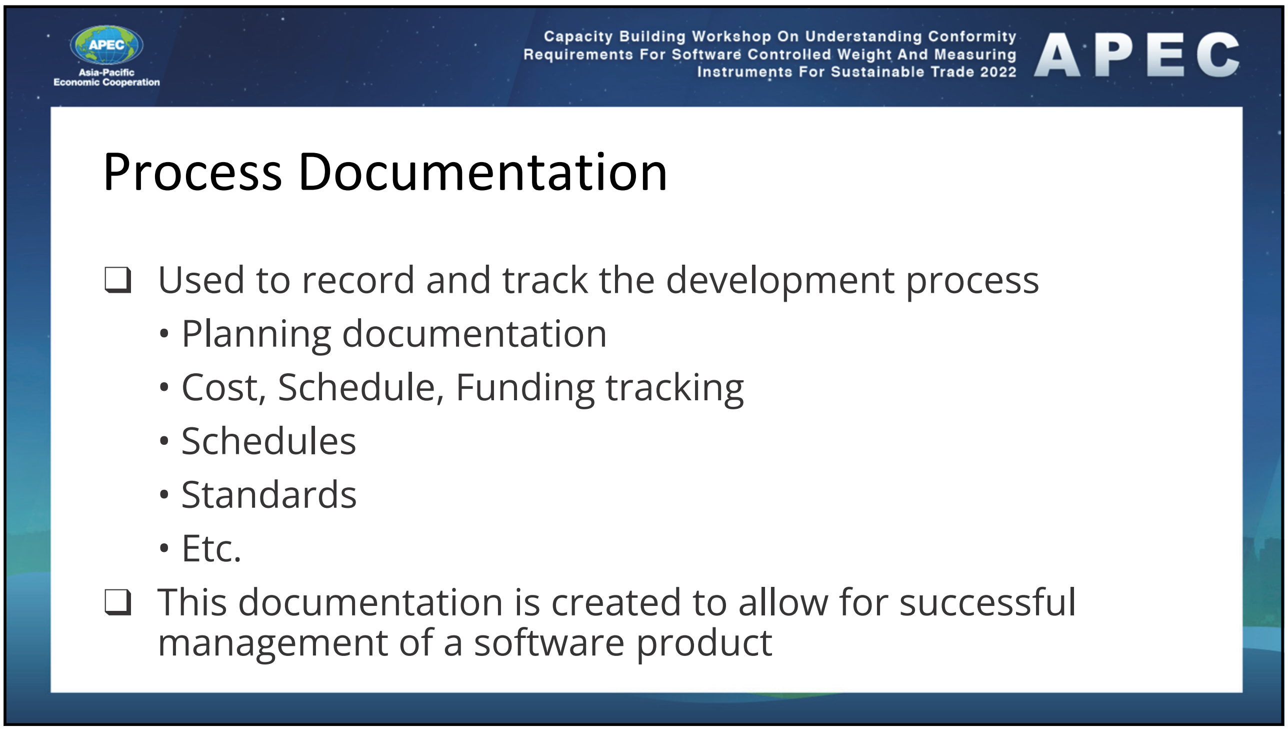
- ❑ In all software projects some amount of documentation should be created prior to any code being written
  - Design docs, etc.
- ❑ Documentation should continue after the code has been completed
  - User’s manuals, etc.
- ❑ The two main types of documentation created are Process and Product documents



Capacity Building Workshop On Understanding Conformity Requirements For Software Controlled Weight And Measuring Instruments For Sustainable Trade 2022

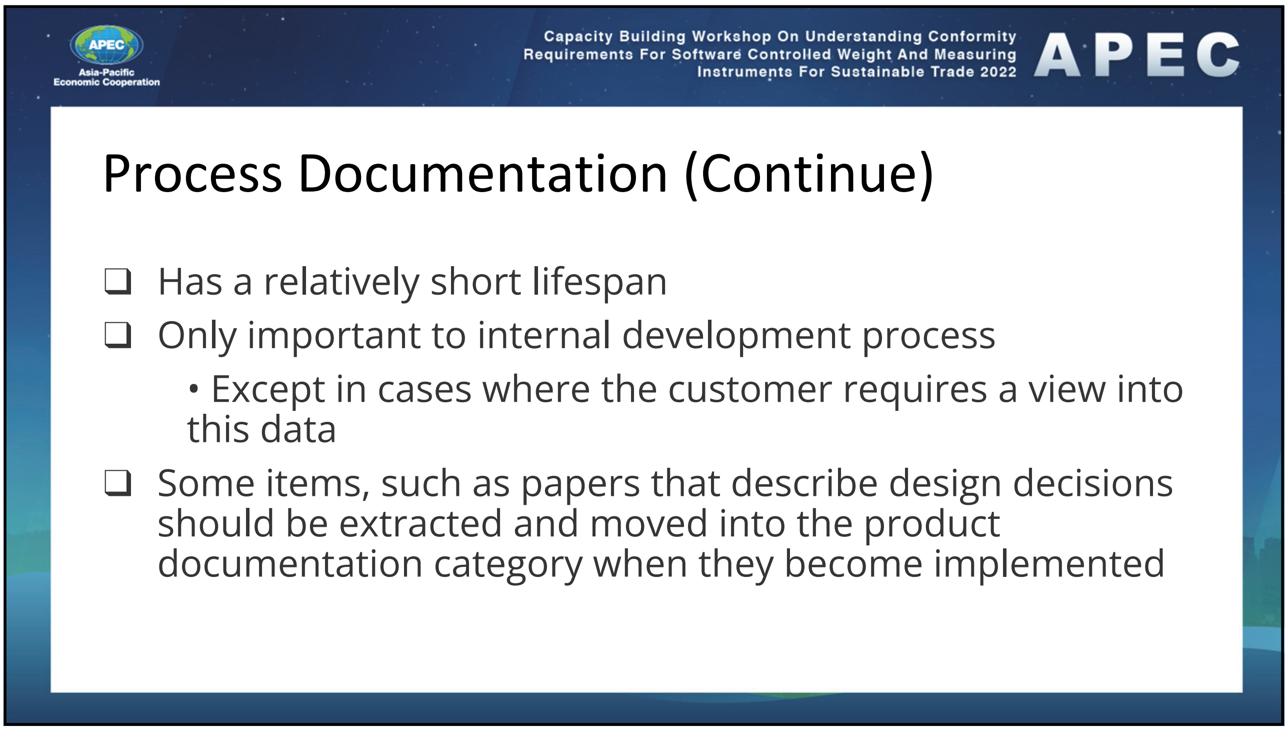
2 Classes of Documentation

- Process Documentation
- Records the process of development and maintenance.
- Product Documentation
- Describes the product being developed.



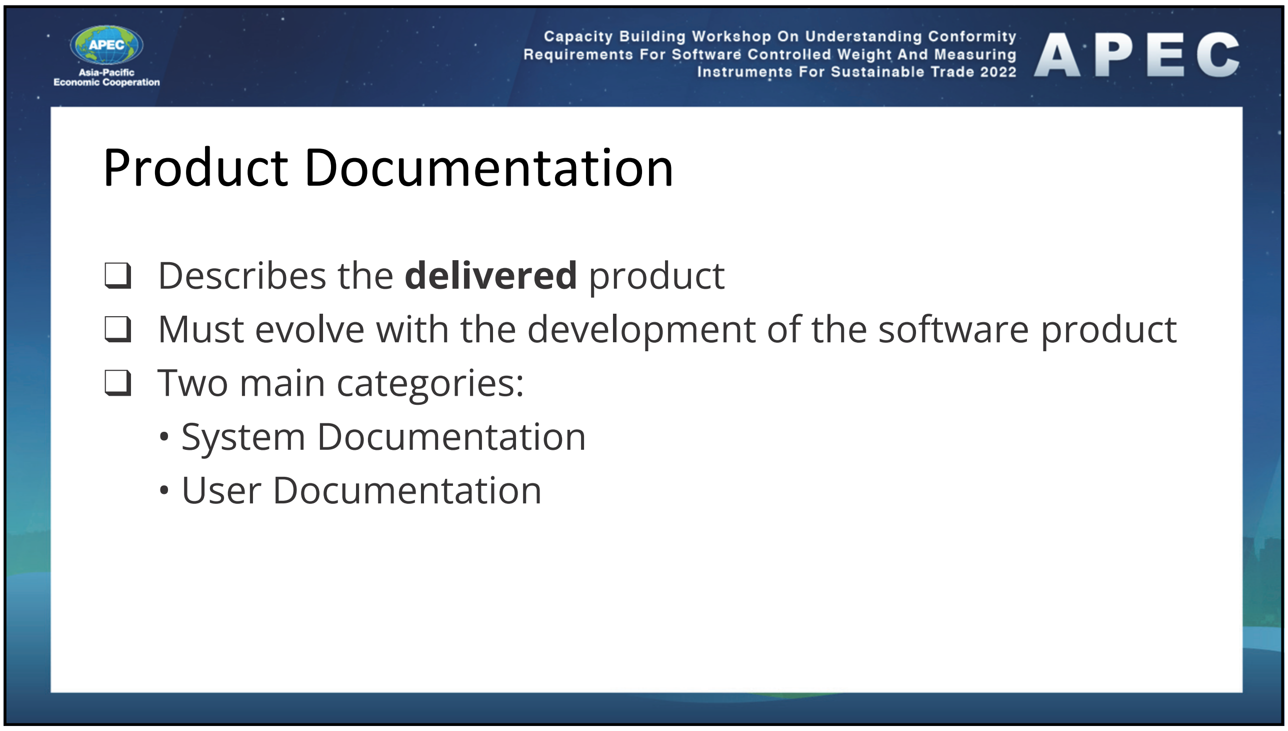
Process Documentation

- ❑ Used to record and track the development process
  - Planning documentation
  - Cost, Schedule, Funding tracking
  - Schedules
  - Standards
  - Etc.
- ❑ This documentation is created to allow for successful management of a software product



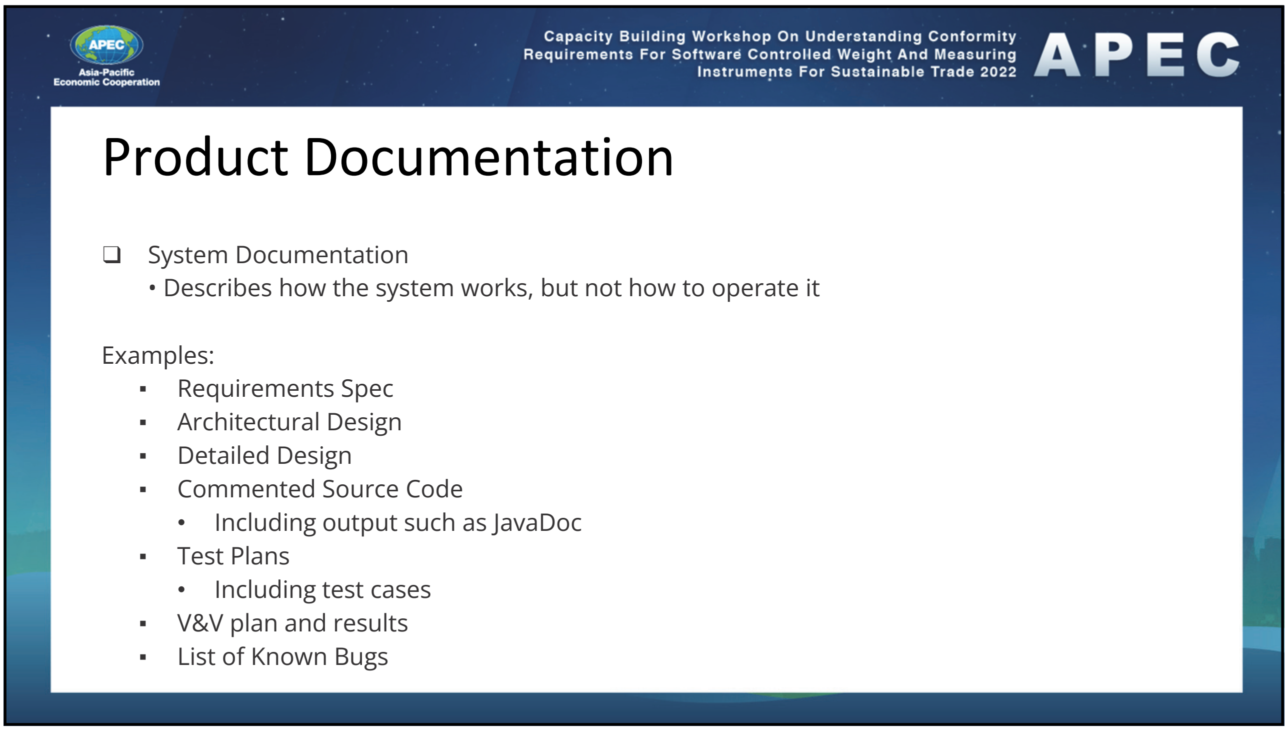
Process Documentation (Continue)

- ❑ Has a relatively short lifespan
- ❑ Only important to internal development process
  - Except in cases where the customer requires a view into this data
- ❑ Some items, such as papers that describe design decisions should be extracted and moved into the product documentation category when they become implemented



Product Documentation

- ❑ Describes the **delivered** product
- ❑ Must evolve with the development of the software product
- ❑ Two main categories:
  - System Documentation
  - User Documentation

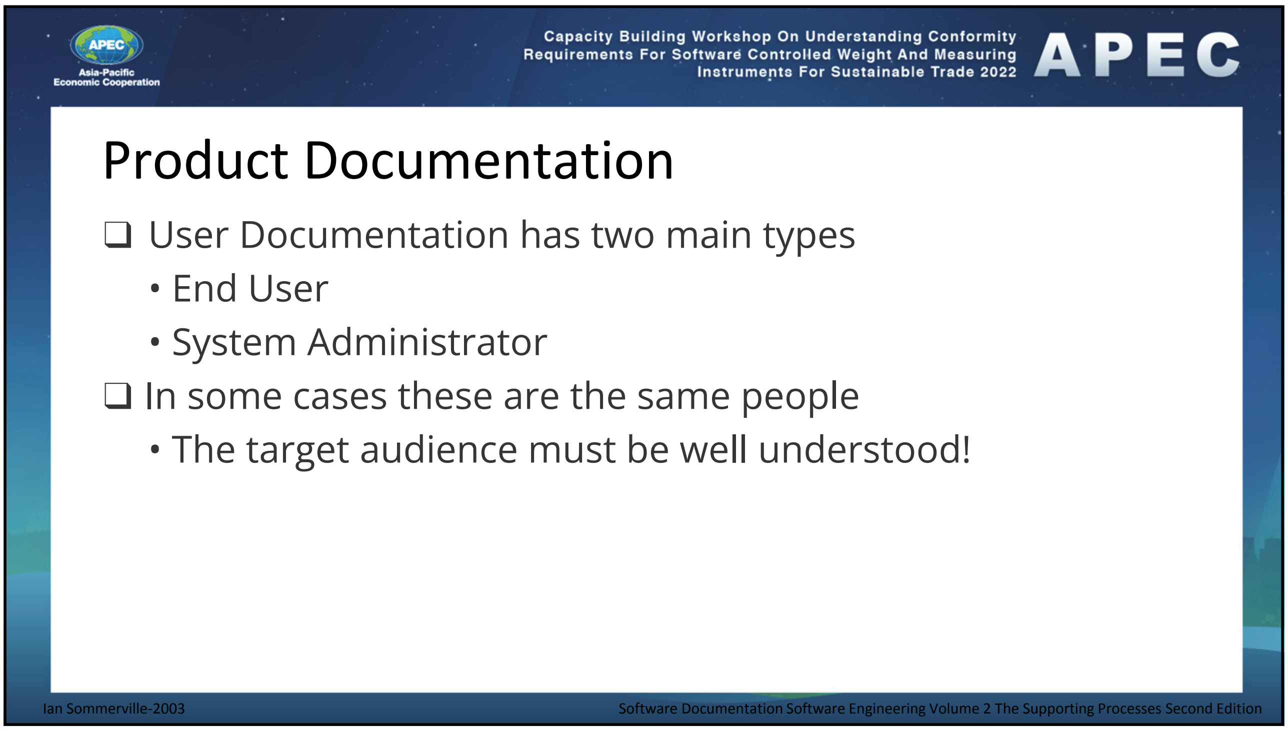


Product Documentation

- ❑ System Documentation
  - Describes how the system works, but not how to operate it

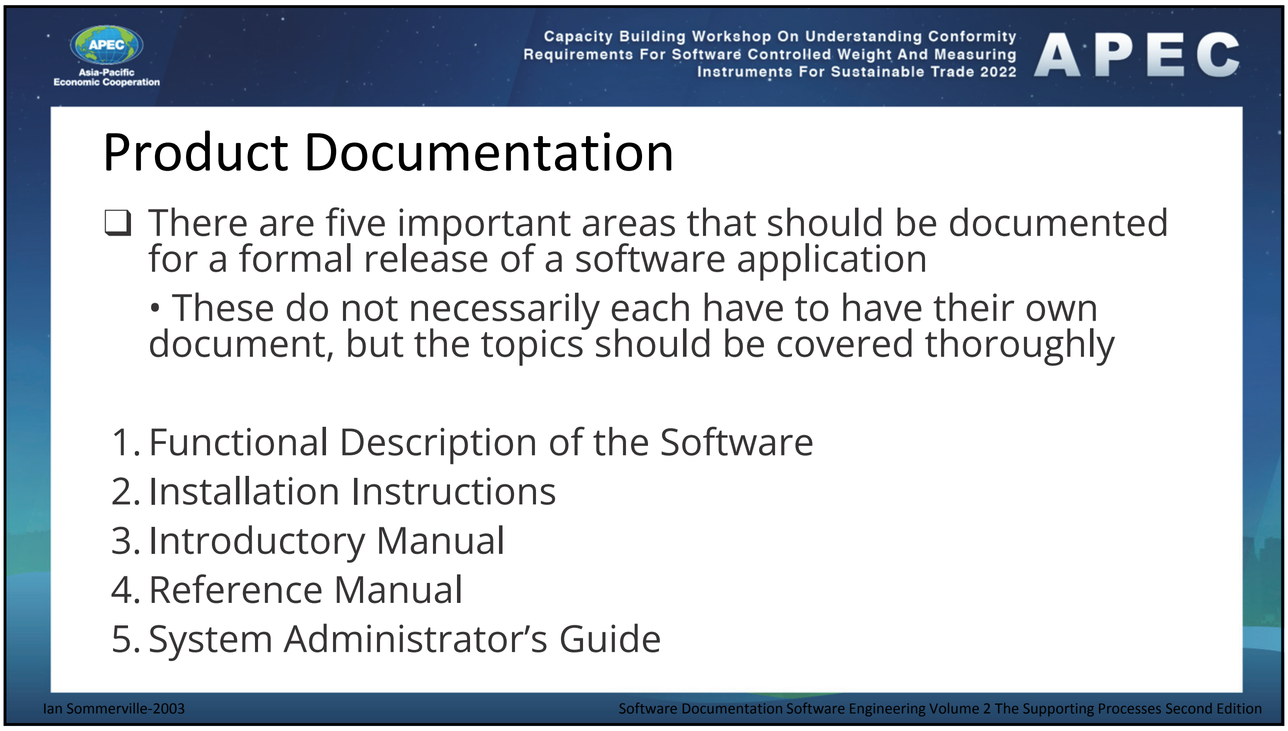
Examples:

- Requirements Spec
- Architectural Design
- Detailed Design
- Commented Source Code
  - Including output such as JavaDoc
- Test Plans
  - Including test cases
- V&V plan and results
- List of Known Bugs



Product Documentation

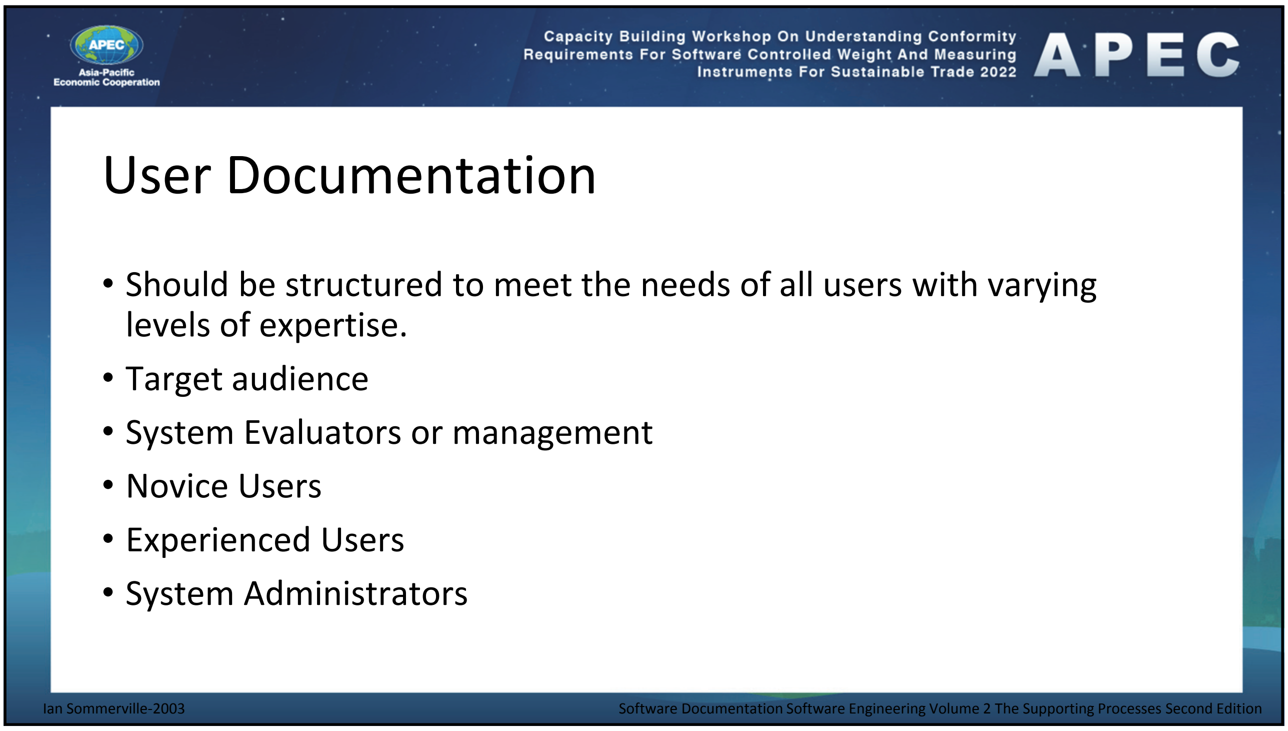
- ❑ User Documentation has two main types
  - End User
  - System Administrator
- ❑ In some cases these are the same people
  - The target audience must be well understood!



Product Documentation

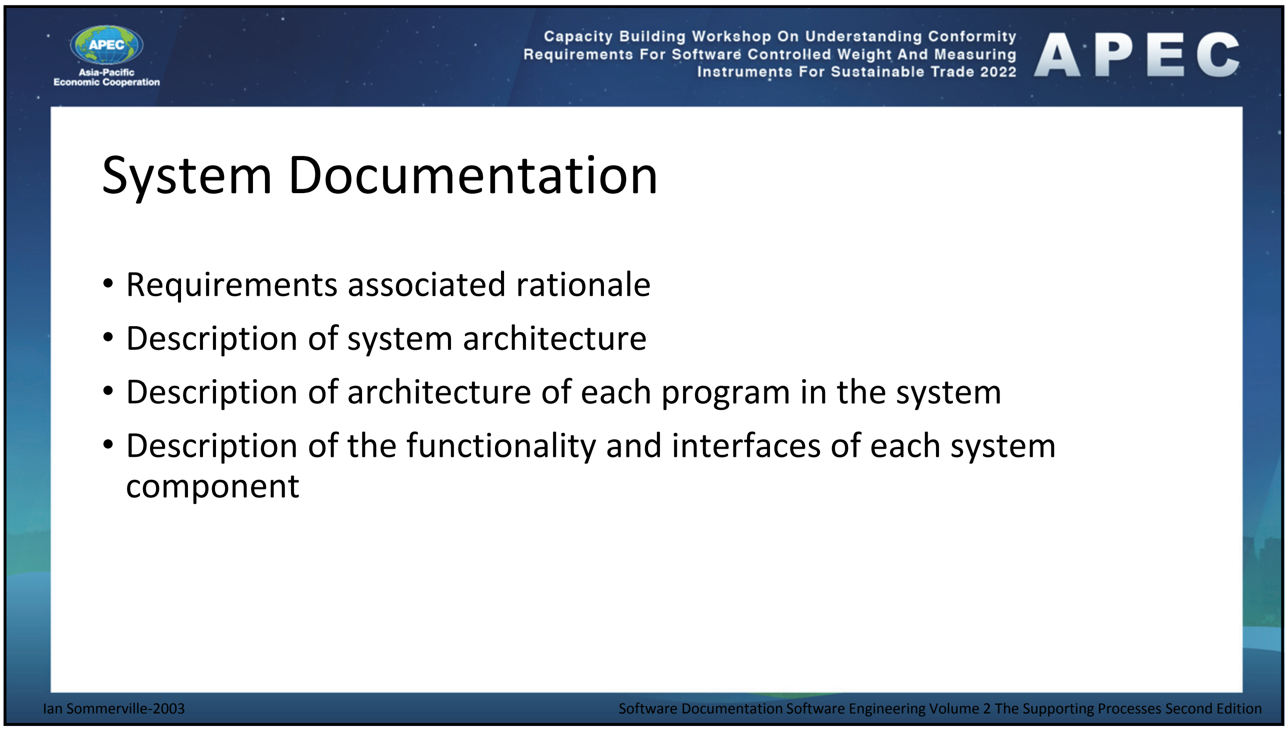
- ❑ There are five important areas that should be documented for a formal release of a software application
  - These do not necessarily each have to have their own document, but the topics should be covered thoroughly

1. Functional Description of the Software
2. Installation Instructions
3. Introductory Manual
4. Reference Manual
5. System Administrator’s Guide



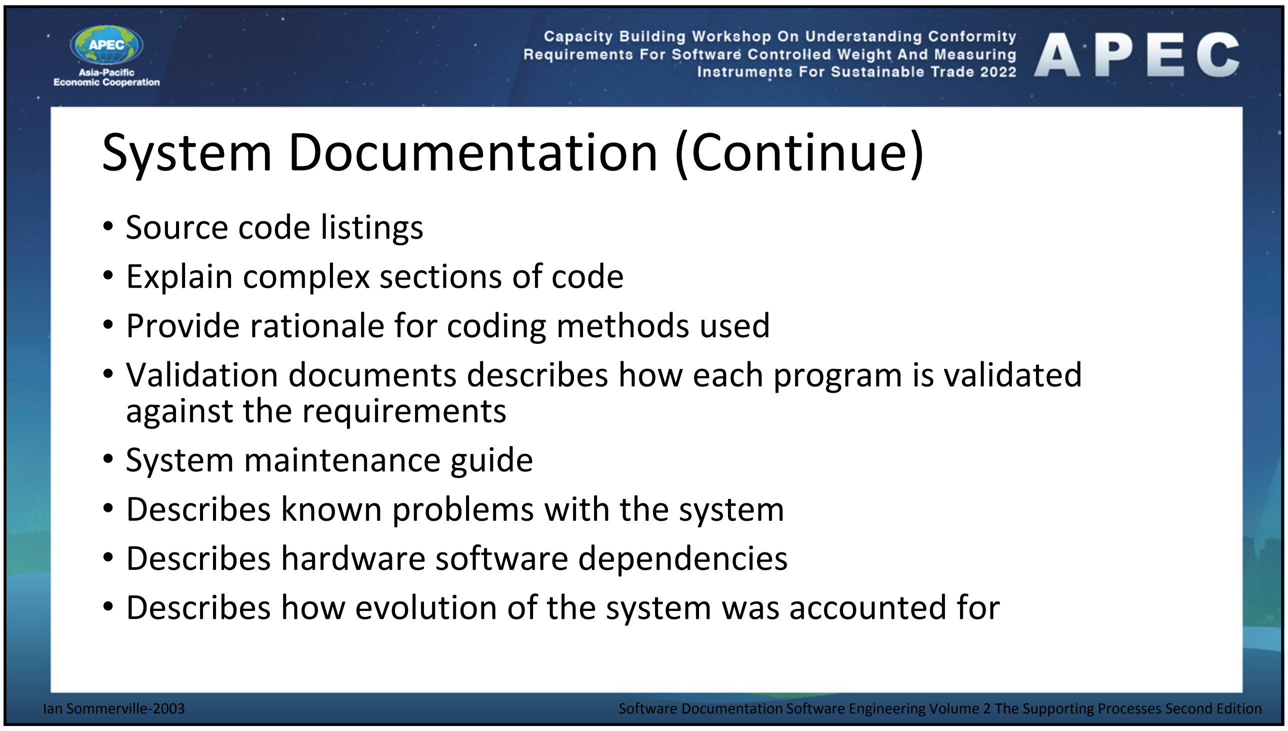
User Documentation

- Should be structured to meet the needs of all users with varying levels of expertise.
- Target audience
- System Evaluators or management
- Novice Users
- Experienced Users
- System Administrators



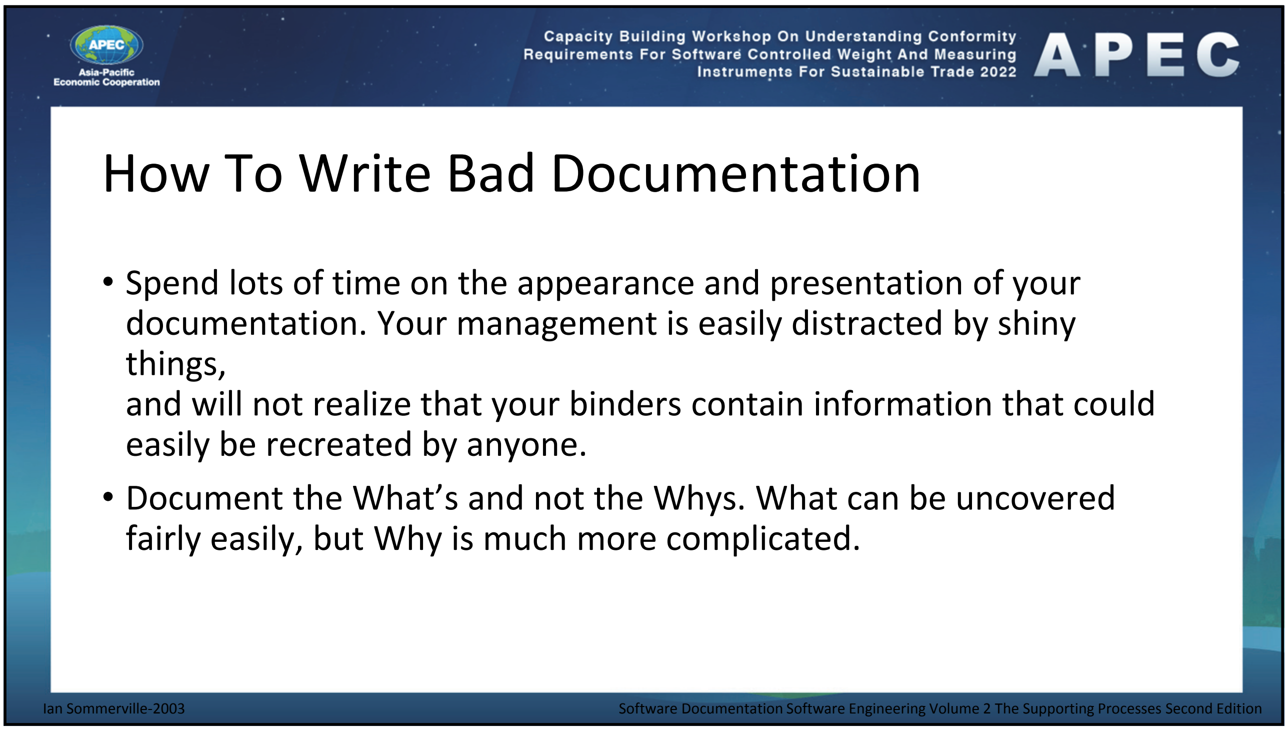
System Documentation

- Requirements associated rationale
- Description of system architecture
- Description of architecture of each program in the system
- Description of the functionality and interfaces of each system component



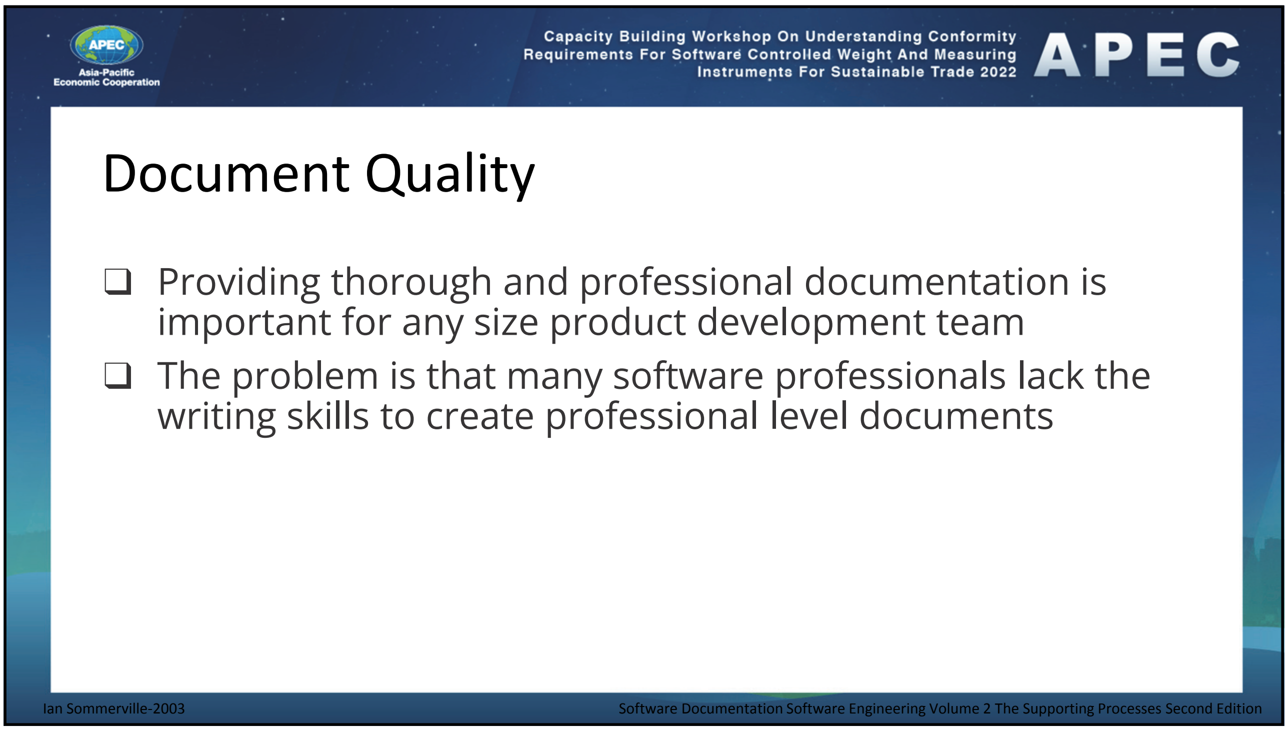
System Documentation (Continue)

- Source code listings
- Explain complex sections of code
- Provide rationale for coding methods used
- Validation documents describes how each program is validated against the requirements
- System maintenance guide
- Describes known problems with the system
- Describes hardware software dependencies
- Describes how evolution of the system was accounted for



How To Write Bad Documentation

- Spend lots of time on the appearance and presentation of your documentation. Your management is easily distracted by shiny things,
  and will not realize that your binders contain information that could easily be recreated by anyone.
- Document the What's and not the Whys. What can be uncovered fairly easily, but Why is much more complicated.



APEC Asia-Pacific Economic Cooperation

Capacity Building Workshop On Understanding Conformity Requirements For Software Controlled Weight And Measuring Instruments For Sustainable Trade 2022

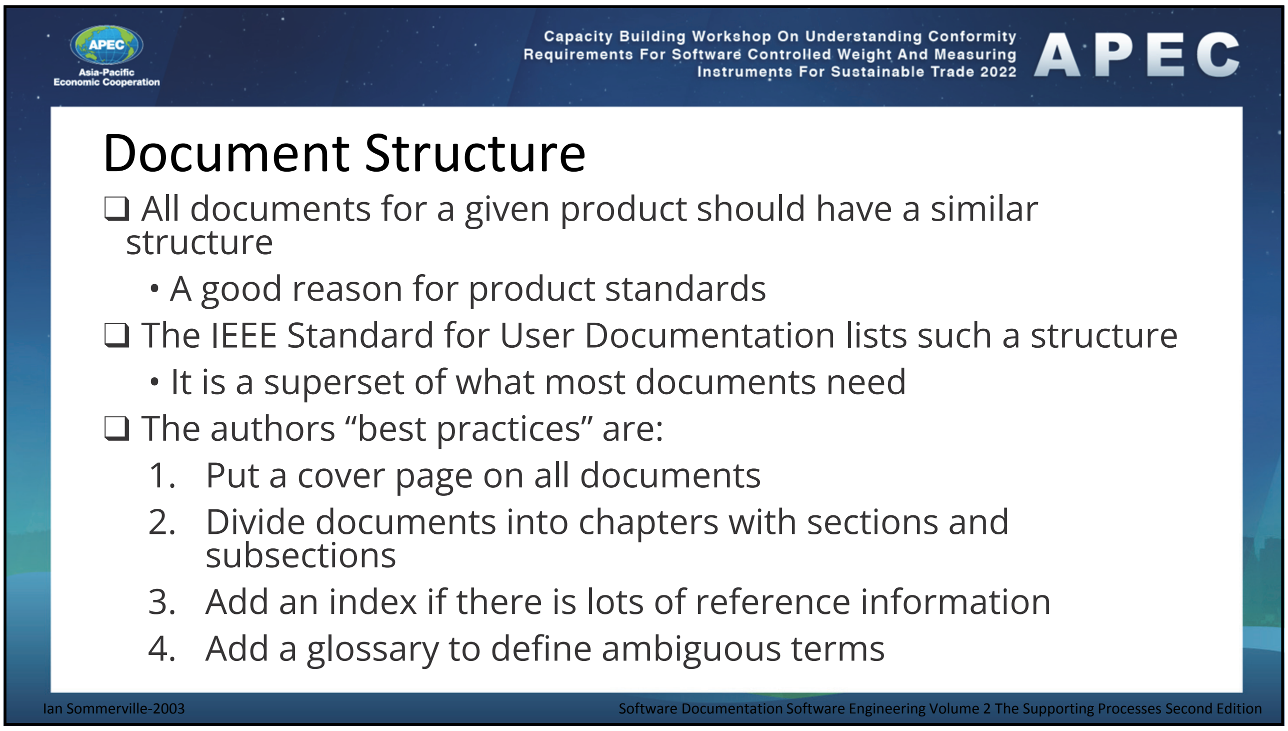
APEC

Document Quality

- ❑ Providing thorough and professional documentation is important for any size product development team
- ❑ The problem is that many software professionals lack the writing skills to create professional level documents

Ian Sommerville-2003

Software Documentation Software Engineering Volume 2 The Supporting Processes Second Edition



APEC Asia-Pacific Economic Cooperation

Capacity Building Workshop On Understanding Conformity Requirements For Software Controlled Weight And Measuring Instruments For Sustainable Trade 2022

APEC

Document Structure

- ❑ All documents for a given product should have a similar structure
  - A good reason for product standards
- ❑ The IEEE Standard for User Documentation lists such a structure
  - It is a superset of what most documents need
- ❑ The authors "best practices" are:
  1. Put a cover page on all documents
  2. Divide documents into chapters with sections and subsections
  3. Add an index if there is lots of reference information
  4. Add a glossary to define ambiguous terms

Ian Sommerville-2003

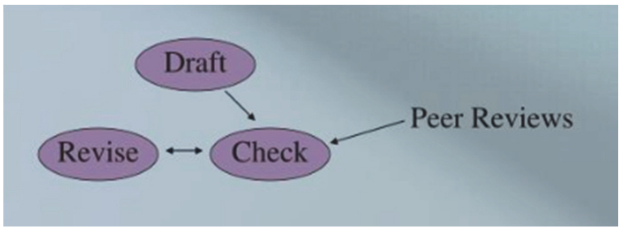
Software Documentation Software Engineering Volume 2 The Supporting Processes Second Edition

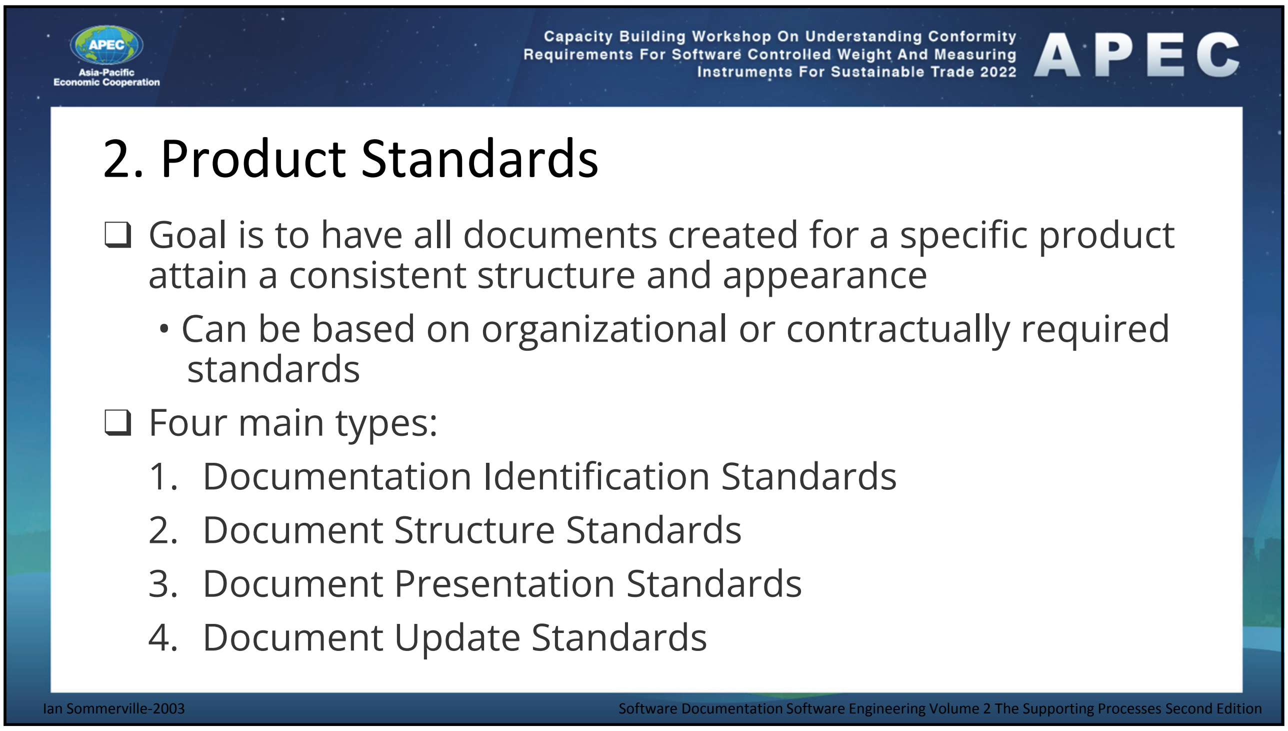
Standards

- ❑ Standards play an important role in the development, maintenance and usefulness of documentation
- ❑ Standards can act as a basis for quality documentation • But are not good enough on their own • Usually define high level content and organization
- ❑ There are three types of documentation standards

1. Process Standards

- ❑ Define the approach that is to be used when creating the documentation
- ❑ Don’t actually define any of the content of the documents Peer Reviews

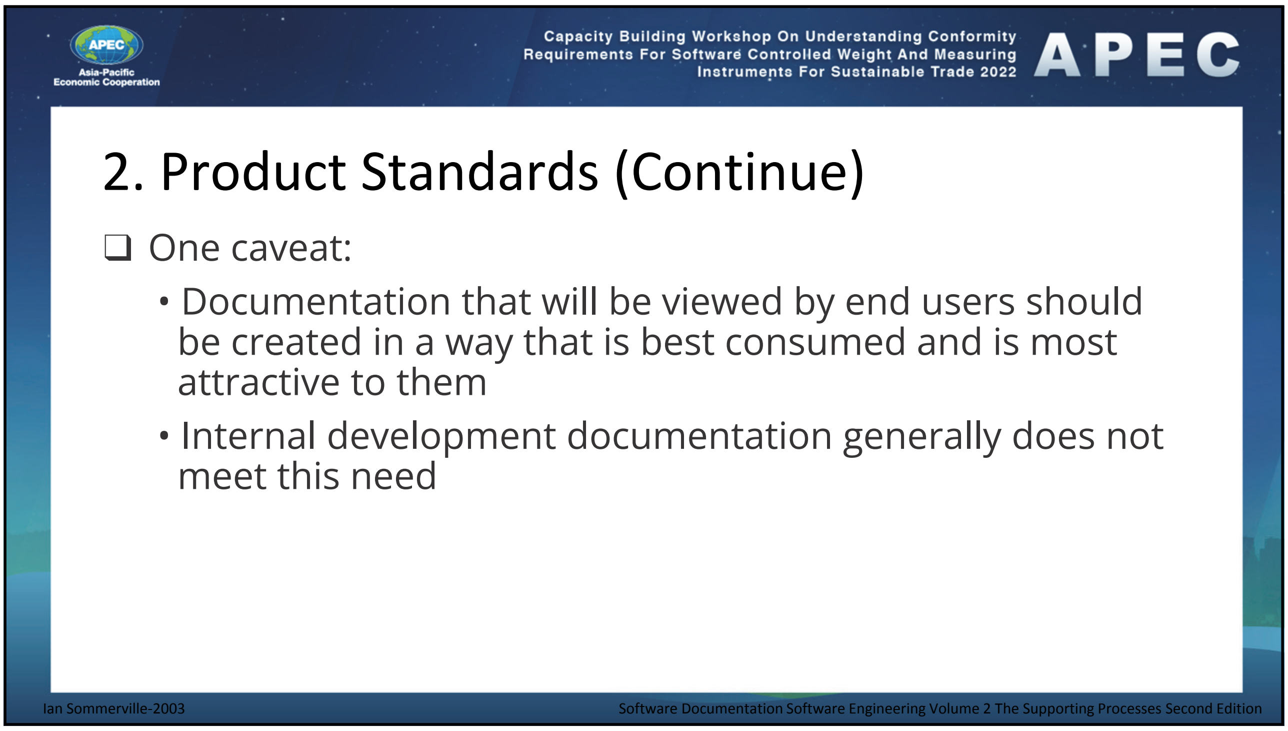




2. Product Standards

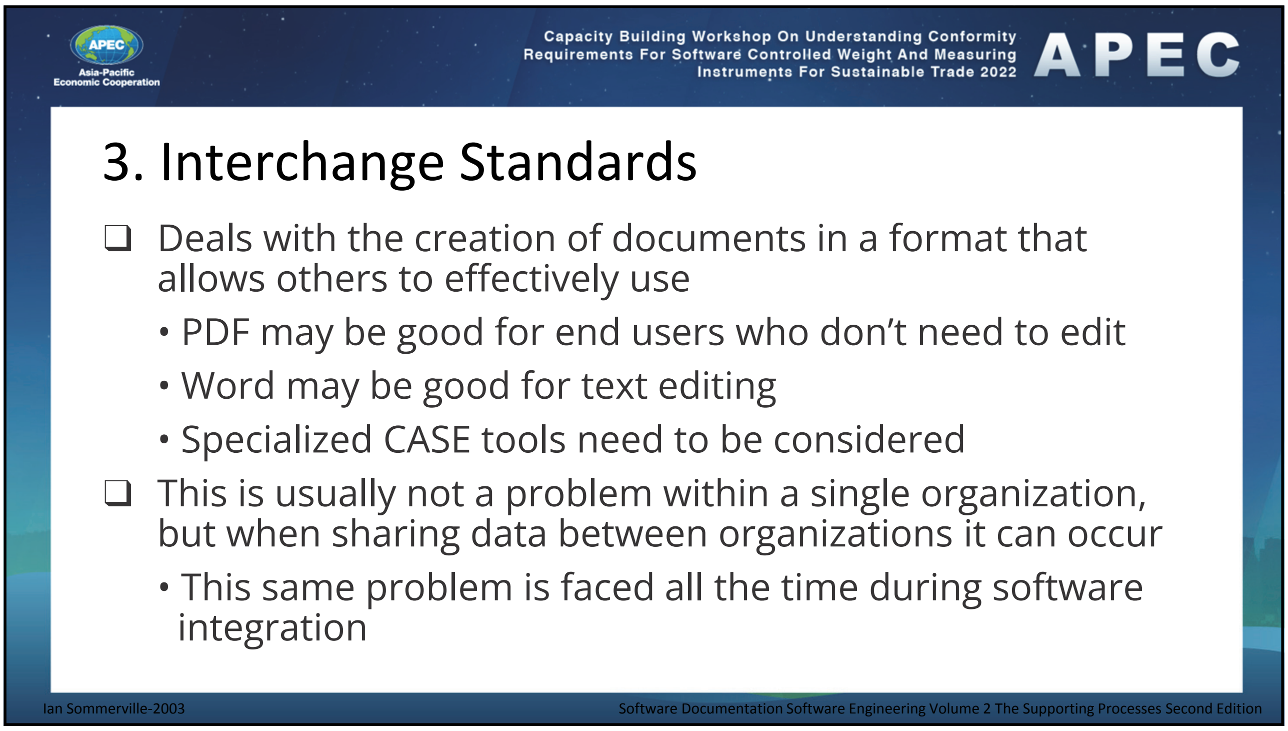
❑ Goal is to have all documents created for a specific product attain a consistent structure and appearance
- Can be based on organizational or contractually required standards

❑ Four main types:
1. Documentation Identification Standards
2. Document Structure Standards
3. Document Presentation Standards
4. Document Update Standards



2. Product Standards (Continue)

❑ One caveat:
- Documentation that will be viewed by end users should be created in a way that is best consumed and is most attractive to them
- Internal development documentation generally does not meet this need



Asia-Pacific Economic Cooperation

Capacity Building Workshop On Understanding Conformity Requirements For Software Controlled Weight And Measuring Instruments For Sustainable Trade 2022

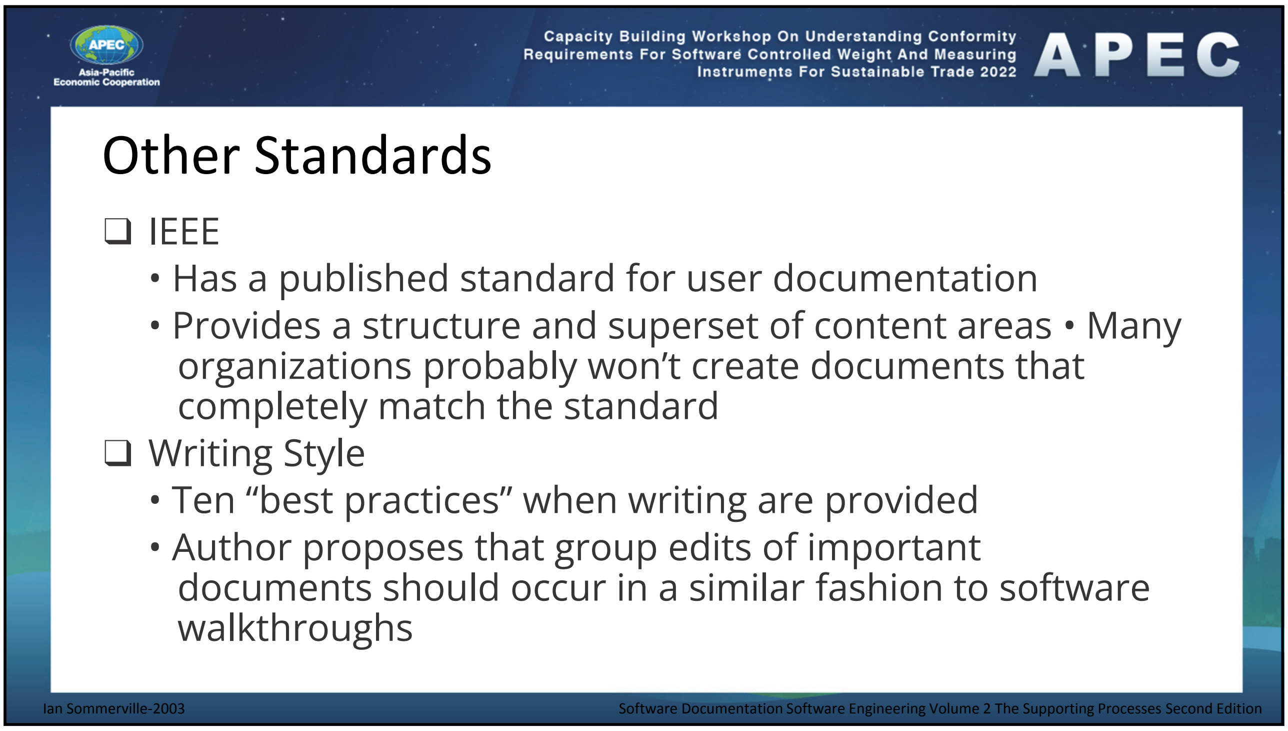
APEC

3. Interchange Standards

- ❑ Deals with the creation of documents in a format that allows others to effectively use
  - PDF may be good for end users who don't need to edit
  - Word may be good for text editing
  - Specialized CASE tools need to be considered
- ❑ This is usually not a problem within a single organization, but when sharing data between organizations it can occur
  - This same problem is faced all the time during software integration

Ian Sommerville-2003

Software Documentation Software Engineering Volume 2 The Supporting Processes Second Edition



Asia-Pacific Economic Cooperation

Capacity Building Workshop On Understanding Conformity Requirements For Software Controlled Weight And Measuring Instruments For Sustainable Trade 2022

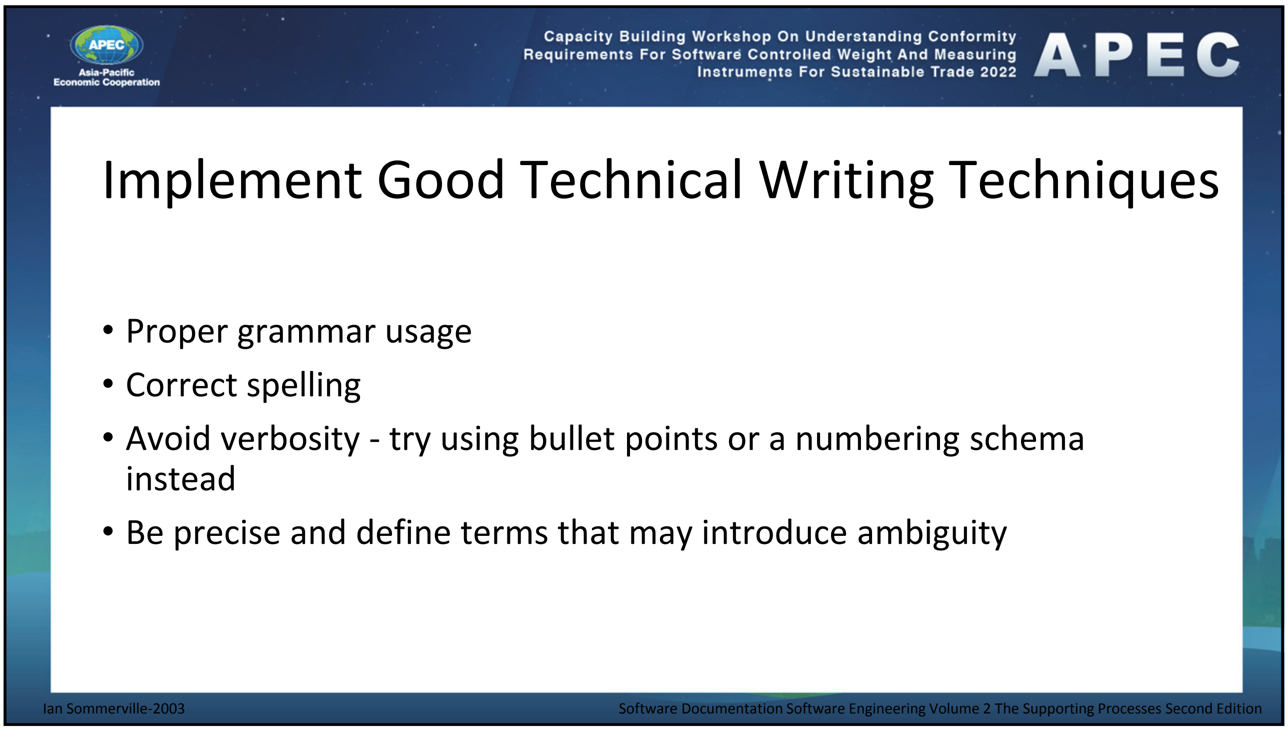
APEC

Other Standards

- ❑ IEEE
  - Has a published standard for user documentation
  - Provides a structure and superset of content areas • Many organizations probably won't create documents that completely match the standard
- ❑ Writing Style
  - Ten "best practices" when writing are provided
  - Author proposes that group edits of important documents should occur in a similar fashion to software walkthroughs

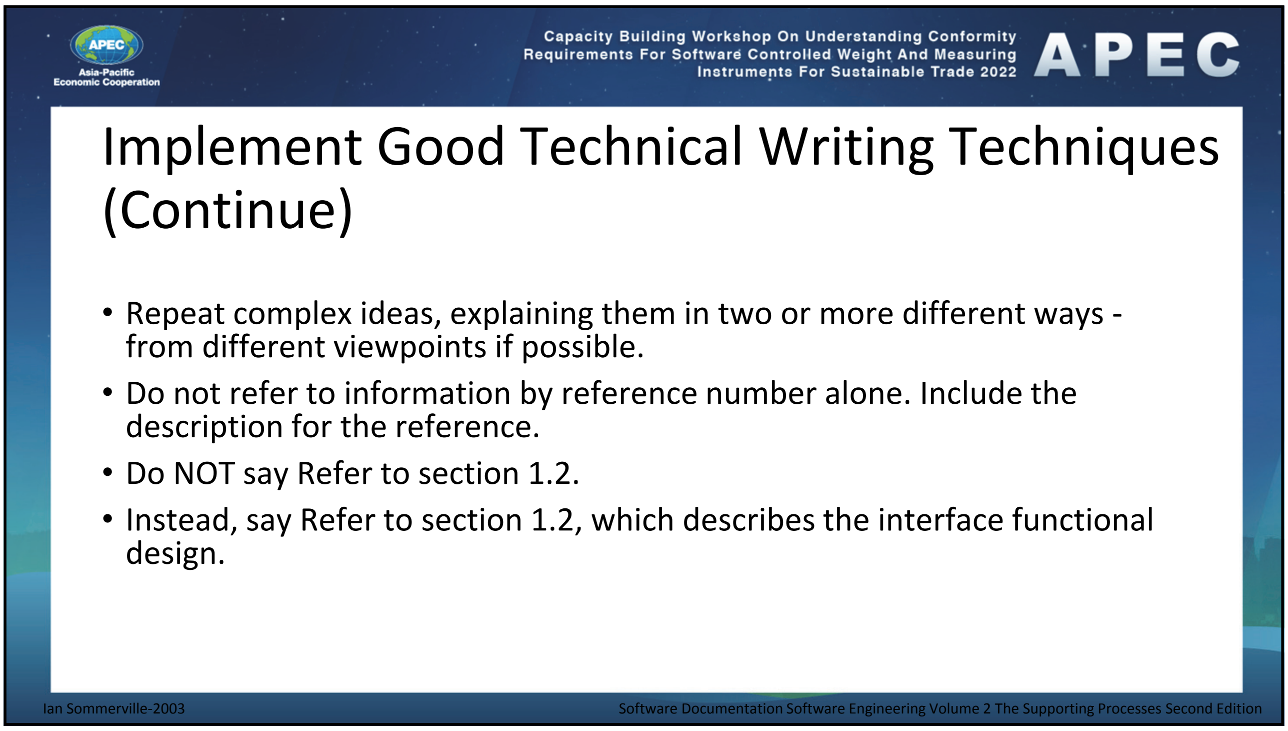
Ian Sommerville-2003

Software Documentation Software Engineering Volume 2 The Supporting Processes Second Edition



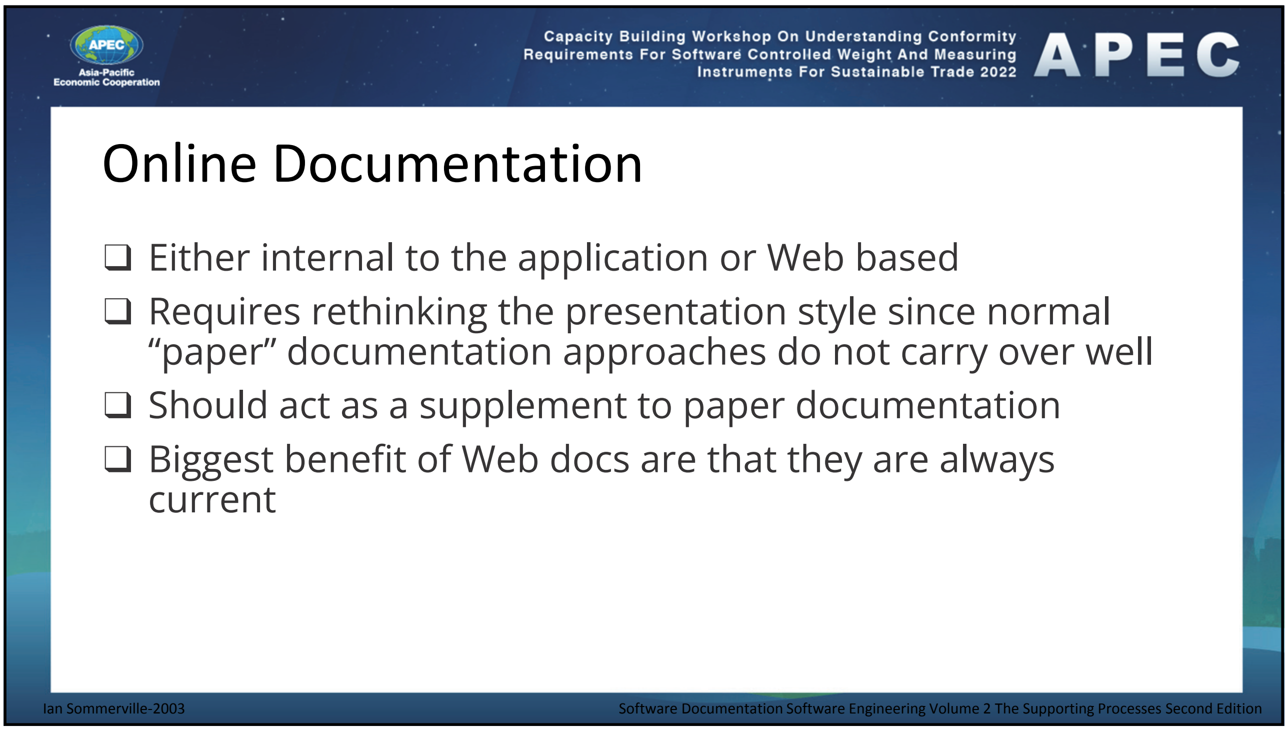
Implement Good Technical Writing Techniques

- Proper grammar usage
- Correct spelling
- Avoid verbosity - try using bullet points or a numbering schema instead
- Be precise and define terms that may introduce ambiguity



Implement Good Technical Writing Techniques (Continue)

- Repeat complex ideas, explaining them in two or more different ways - from different viewpoints if possible.
- Do not refer to information by reference number alone. Include the description for the reference.
- Do NOT say Refer to section 1.2.
- Instead, say Refer to section 1.2, which describes the interface functional design.



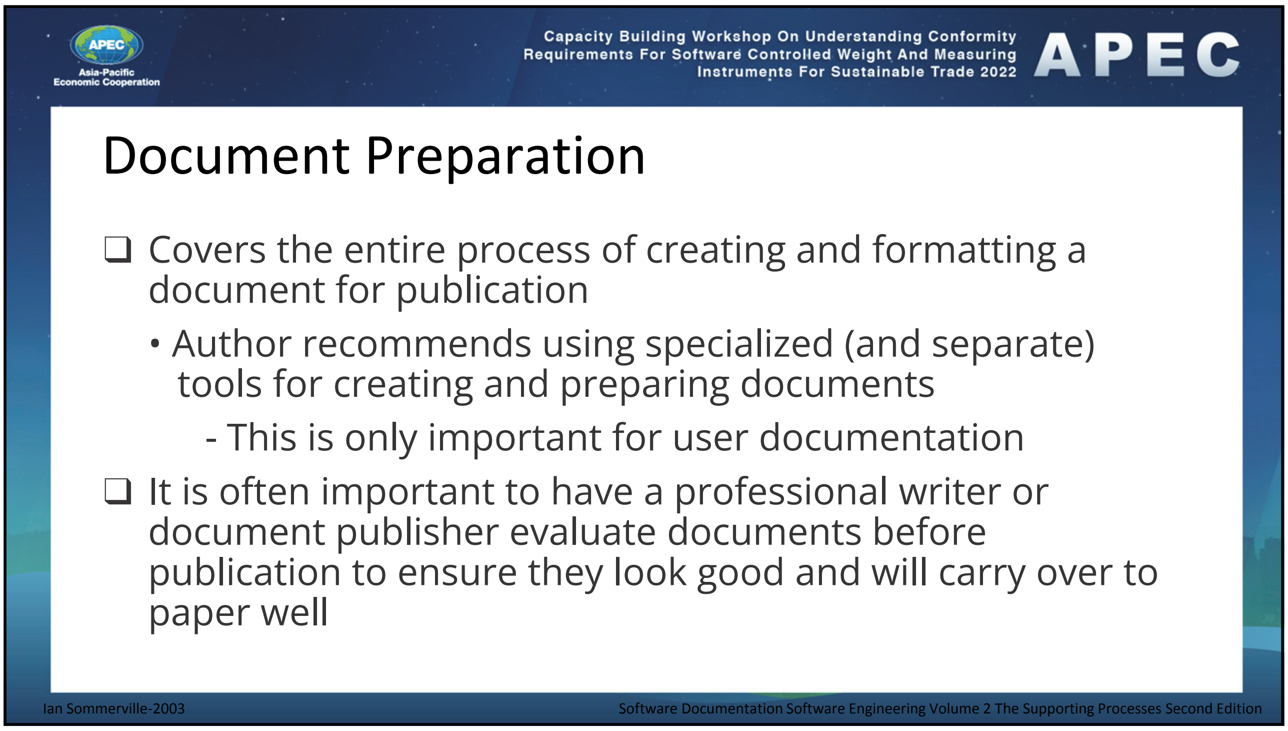
Capacity Building Workshop On Understanding Conformity Requirements For Software Controlled Weight And Measuring Instruments For Sustainable Trade 2022

Online Documentation

- ❑ Either internal to the application or Web based
- ❑ Requires rethinking the presentation style since normal "paper" documentation approaches do not carry over well
- ❑ Should act as a supplement to paper documentation
- ❑ Biggest benefit of Web docs are that they are always current

Ian Sommerville-2003

Software Documentation Software Engineering Volume 2 The Supporting Processes Second Edition



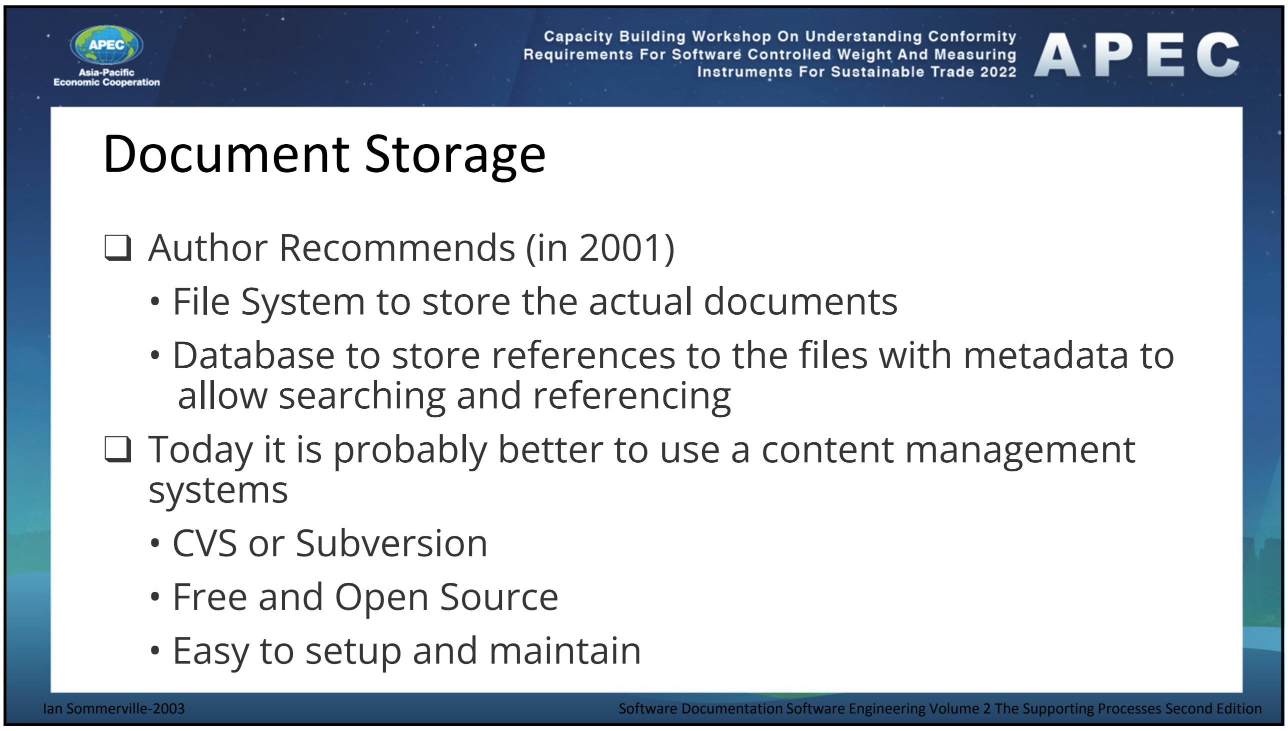
Capacity Building Workshop On Understanding Conformity Requirements For Software Controlled Weight And Measuring Instruments For Sustainable Trade 2022

Document Preparation

- ❑ Covers the entire process of creating and formatting a document for publication
  - • Author recommends using specialized (and separate) tools for creating and preparing documents
    - - This is only important for user documentation
- ❑ It is often important to have a professional writer or document publisher evaluate documents before publication to ensure they look good and will carry over to paper well

Ian Sommerville-2003

Software Documentation Software Engineering Volume 2 The Supporting Processes Second Edition

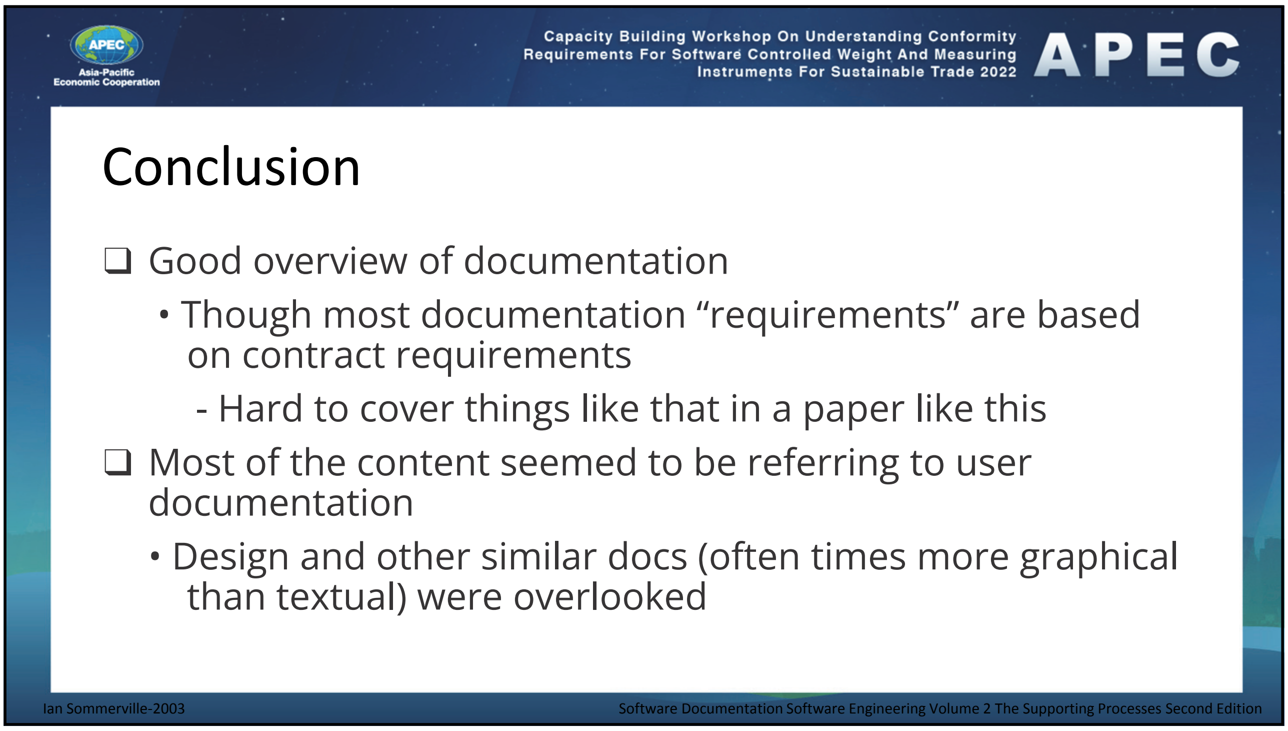


Capacity Building Workshop On Understanding Conformity Requirements For Software Controlled Weight And Measuring Instruments For Sustainable Trade 2022

Document Storage

- ❑ Author Recommends (in 2001)
  - File System to store the actual documents
  - Database to store references to the files with metadata to allow searching and referencing
- ❑ Today it is probably better to use a content management systems
  - CVS or Subversion
  - Free and Open Source
  - Easy to setup and maintain

Ian Sommerville-2003 Software Documentation Software Engineering Volume 2 The Supporting Processes Second Edition



Capacity Building Workshop On Understanding Conformity Requirements For Software Controlled Weight And Measuring Instruments For Sustainable Trade 2022

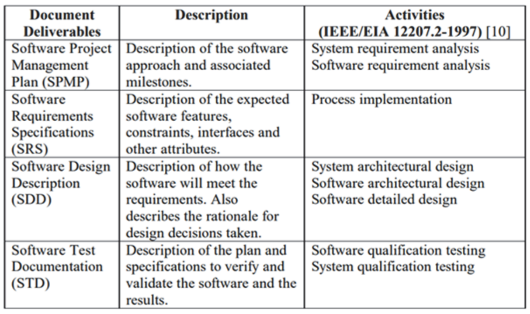
Conclusion

- ❑ Good overview of documentation
  - Though most documentation “requirements” are based on contract requirements
    - Hard to cover things like that in a paper like this
- ❑ Most of the content seemed to be referring to user documentation
  - Design and other similar docs (often times more graphical than textual) were overlooked

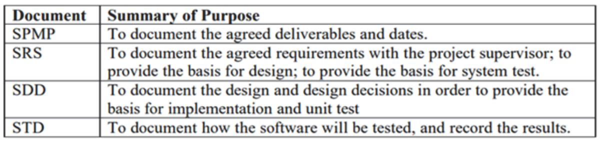
Ian Sommerville-2003 Software Documentation Software Engineering Volume 2 The Supporting Processes Second Edition

IEEE standards and US MIL-STD-498

Documentation Set

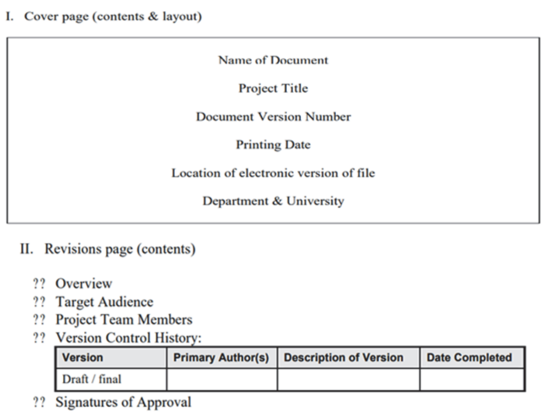


Document Deliverables	Description	Activities (IEEE/EIA 12207.2-1997) [10]
Software Project Management Plan (SPMP)	Description of the software approach and associated milestones.	System requirement analysis Software requirement analysis
Software Requirements Specifications (SRS)	Description of the expected software features, constraints, interfaces and other attributes.	Process implementation
Software Design Description (SDD)	Description of how the software will meet the requirements. Also describes the rationale for design decisions taken.	System architectural design Software architectural design Software detailed design
Software Test Documentation (STD)	Description of the plan and specifications to verify and validate the software and the results.	Software qualification testing System qualification testing



Document	Summary of Purpose
SPMP	To document the agreed deliverables and dates.
SRS	To document the agreed requirements with the project supervisor; to provide the basis for design; to provide the basis for system test.
SDD	To document the design and design decisions in order to provide the basis for implementation and unit test
STD	To document how the software will be tested, and record the results.

Common Sections for the Documentation Set



I. Cover page (contents & layout)

Name of Document
Project Title
Document Version Number
Printing Date
Location of electronic version of file
Department & University

II. Revisions page (contents)

?? Overview
?? Target Audience
?? Project Team Members
?? Version Control History:

Version	Primary Author(s)	Description of Version	Date Completed
Draft / final			

?? Signatures of Approval

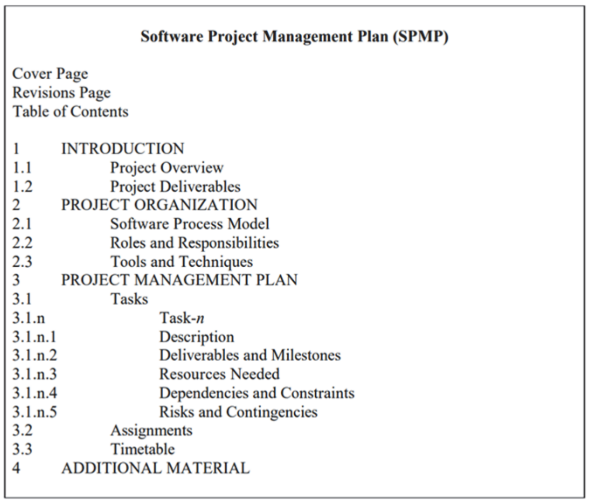
III. Additional Material (contents)

?? ADDITIONAL ISSUES
?? DFINITIONS, ACRONYMS, AND ABBREVIATIONS
?? REFERENCES
?? APPENDICES

Capacity Building Workshop On Understanding Conformity Requirements For Software Controlled Weight And Measuring Instruments For Sustainable Trade 2022

APEC

Contents of the Documentation Set

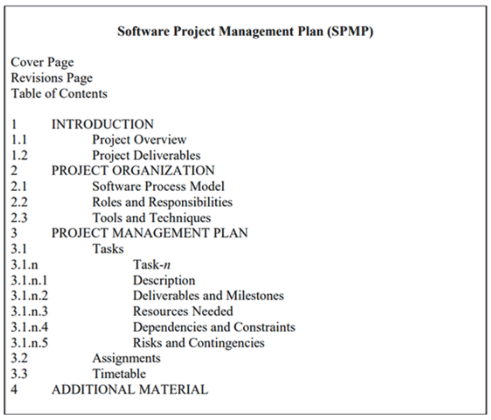


Software Project Management Plan (SPMP)

Capacity Building Workshop On Understanding Conformity Requirements For Software Controlled Weight And Measuring Instruments For Sustainable Trade 2022

APEC

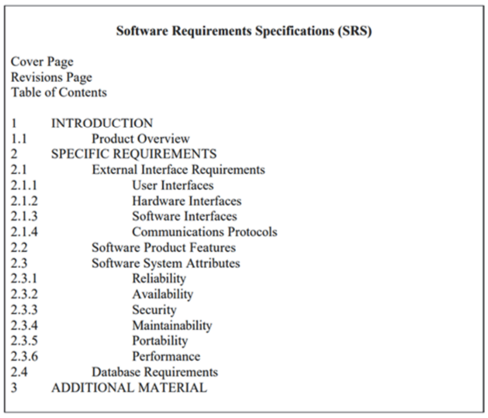
Contents of the Documentation Set …Con



Software Project Management Plan (SPMP)

Cover Page
Revisions Page
Table of Contents

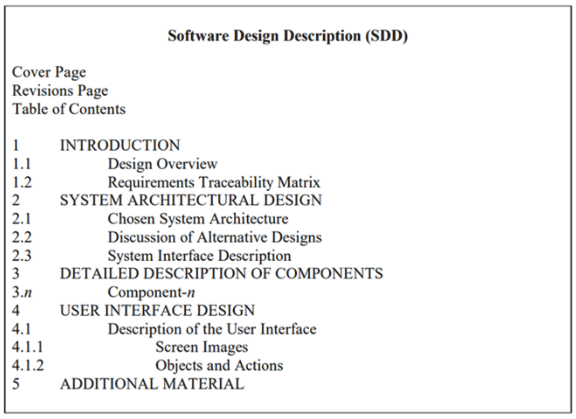
1 INTRODUCTION
1.1 Project Overview
1.2 Project Deliverables
2 PROJECT ORGANIZATION
2.1 Software Process Model
2.2 Roles and Responsibilities
2.3 Tools and Techniques
3 PROJECT MANAGEMENT PLAN
3.1 Tasks
3.1.n Task-*n*
3.1.n.1 Description
3.1.n.2 Deliverables and Milestones
3.1.n.3 Resources Needed
3.1.n.4 Dependencies and Constraints
3.1.n.5 Risks and Contingencies
3.2 Assignments
3.3 Timetable
4 ADDITIONAL MATERIAL



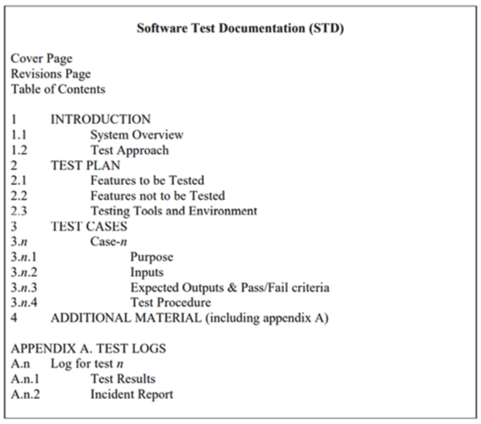
Software Requirements Specifications (SRS)

Capacity Building Workshop On Understanding Conformity Requirements For Software Controlled Weight And Measuring Instruments For Sustainable Trade 2022

Contents of the Documentation Set …Con



Software Design Description (SDD)



Software Test Documentation (STD)

Capacity Building Workshop On Understanding Conformity Requirements For Software Controlled Weight And Measuring Instruments For Sustainable Trade 2022

Software Documentation Standard

- IEEE Std. 829-1998 IEEE Standard for Software Test Documentation
- IEEE Std. 830-1998 IEEE Recommended Practice for Software Requirements Specifications
- IEEE Std. 1008-1997 IEEE Standard for Software Unit Testing
- IEEE Std. 1012-1998 IEEE Standard for Software Verification and Validation
- IEEE Std. 1016-1998 IEEE Recommended Practice for Software Design Descriptions
- IEEE Std 1058-1998 IEEE Standard for Software Project Management Plans
- IEEE Std 1540-2001 IEEE Standard for Software Life Cycle Processes – Risk Management
- IEEE 12207.2-1997 Industry Implementation of International Standard ISO/IEC 12207: 1995 (ISO/IEC 12207) Standard for Information Technology -Software Life Cycle Processes - Implementation Consideration

APEC
Asia-Pacific
Economic Cooperation

Universiti
Kebangsaan
Malaysia
National University
of Malaysia

SIRIM

NMIM
NATIONAL METROLOGY
INSTITUTE OF MALAYSIA

THANK YOU !