

Dig the SPACEDIRT

Peter Morgan, senior practitioner with e-testing Consultancy, on IEEE 829:

arguably still the most used testing standard

“Why standards? The use of standards simplifies communication, promotes consistency and uniformity, and eliminates the need to invent yet another (often different and even incompatible) solution to the same problem. Standards, whether ‘official’ or merely agreed upon, are especially important when we’re talking to customers and suppliers, but it’s easy to underestimate their importance when dealing with different departments and disciplines within our own organisation. They also provide vital continuity so that we are not forever reinventing the wheel. They are a way of preserving proven practices above and beyond the inevitable staff changes within organisations.”
– Ed Kit, *Software Testing in the Real World*

That paragraph neatly and (quite) succinctly describes why standards exist. But how does that affect testing practitioners who live, as in the title of Ed Kit’s book, in the real world? Anything that promotes better project communication has to be good for testers. So standards have to be effective, and produce recognisable (and measurable?) gains, while not adding disproportionate overheads. I once worked for a large organisation that had an internal (and mandatory) standard for almost all documents. This was such that a document of 200 real words was turned into 18 pages, when all the necessary parts (‘glossary’, ‘associated documents’) were added. Perhaps this was counterproductive and unnecessary.

IEEE 829 in overview

There have been diverse document types used in software testing, developed in many cases for the needs of a particular organisation. IEEE 829 (1983) is the *Standard for Software Test Documentation*, and this was an attempt to pull sources together and present some best practice ideas. The standard was revisited and revised in 1998. Note that the standard applies to any level of testing that may take place (including acceptance testing), although application in agile development methodologies may be less obvious. It is common to have ‘a full set’ of IEEE 829 documents for each testing stage that is being undertaken.

IEEE 829 is often thought of as being the standard for a High Level Test Plan or Master Test Plan (HLTP or MTP). It is more than this, as the standard describes eight documents that can be produced as part of the testing effort. These documents are sometimes distributed

between different categories, although there is no consensus on the subdivisions. I find the following partitioning helpful:

- Test planning
- Test plan
- Test specification
 - Test design specification
 - Test case specification
 - Test procedure specification
- Test reporting
 - Test item transmittal report
 - Test log
 - Test incident summary
 - Test summary

The eight parts

Most of these eight document types are well known; I will provide a very brief summary, before returning to the test plan.

Test plan: a high level view of *how* testing will proceed; *what* is to be tested, by *whom*, how, in what *time* frame and to what *quality* level.

Test design specification: details the test conditions to be exercised, with the expected outcome (in general terms).

Test case specification: Specific data requirements to run tests, based upon the test conditions identified.

Test procedure specification: Describes how the tester will physically run the test, including set up procedures. The standard defines ten procedure steps that may be applied when running a test.

Test item transmittal report: records when individual items to be tested have been passed from one stage of testing to another. This includes where to find such items, what is new about them, and is in effect a warranty of ‘fit for test’.

Test log: details of what tests were run, by whom, and whether individual tests passed or failed.

Test incident summary: details of instances where a test ‘failed’ for a specific reason.

Test summary: brings together all pertinent information about the testing, including the number of incidents raised and outstanding, and crucially an assessment about the quality of the system. Also recorded, for use in future project planning, are details of what was done, and how long it took.

This document is important in deciding whether the quality of the system is good enough to allow it to proceed to another stage. This assessment is based upon detailed information that was documented in the test plan.

Test planning revisited

Test planning is a key activity in any software testing project, and for that reason many people associate IEEE 829 *only* with test planning. The standard defines 16 items that should be considered for an MTP. This includes the key activities of estimation (as ‘schedule’ is one of the 16) and risk, both of which are large topics in their own right.

The 16 are given below with a well-known mnemonic for remembering the list; much more detail on each can be found in textbooks on the subject.

S Scope

test items, what to test, what not to test

P People

training, responsibilities, schedule

A Approach

the approach that will be taken to the testing

C Criteria

entry/exit criteria, suspension/resumption criteria

E Environment

test environment needs

D Deliverables

what is being delivered as part of the test process

I Incidentals

introduction, identification (of the document), approval authorities

R Risks

risks and contingencies

T Tasks

the test tasks that are involved in the testing process

It is worth noting at this point that the standard lists as ‘deliverables’ the seven other document types that perform part of the standard. Some organisations add to this basic list, including key items such as ‘glossary’, and ‘references to other documents’. I usually keep MTP documents from previous projects, and from projects I worked on for previous organisations, so that I can look back and see the specific details that were included.

MTP is a living document

The document specifies what is going to be done, and how it is going to be done. It needs to be published, to appropriate people, to make others aware of what is going to be tested, and what is not going to be tested. However, don't wait for everything to be completed before the document is circulated for comment and/or review. The MTP will change during the life of the project. This does not mean that it is not necessary to get individual and departmental sign-off; sign-off is achieved on the basis of what is known at a point in time. In one organisation I know of, sign-off is achieved by stating that unless this is received by a specified (and realistic) date, it will be assumed. It is remarkable how that concentrates the minds of those concerned.

Two areas that indicate the dynamic nature of the MTP concern schedules, and risks. During the testing phase, good news and bad news can occur, and this can change priorities. Does that mean that the original MTP was wrong? No; the MTP is what its name suggests, just a plan. At the time, it was based on the best available information, incomplete though this was. Information will be improved as testing progresses; for example what was at one time a critical risk may have now been addressed (eg by third-party security testing). The risk is now answered, and will possibly require no further action.

Review the document

The MTP needs to be reviewed, with review taking place face-to-face. If it is

contentious, points of conflict need to be talked through. The MTP is not just "owned" by the testing team(s), but development groups and users can contribute significantly to clarification and suggest the addition of new items. What is to be tested and is not to be tested are two key elements in the MTP. In October 2002 I worked on a project where testing (as always) was pushed for time. The MTP specified that significant testing would concentrate on the retail system, with respect to 53-week year processing (2002–2003 is a 53-week year). The development team had not realised the significance of 53-week years (that it was *this year*), and *merely the insertion of the testing intention* resulted in better code (development extended unit test coverage, found some problems and implemented fixes).

It is usual for the detail listed in the MTP to be used as a basis for deciding whether the software under test is suitable for the next stage of testing, deployment to production, etc. Therefore key individuals need to see this detail, and agree, before the crunch implementation meeting!

Face reality

The MTP is one place where testing faces reality. The MTP is not free-standing, but fits into the overall test strategy. In some ways, it is not a prescriptive approach, but more of a check list, to remind those responsible what should be *considered* to go into the MTP. The only prescriptive thing about it is the use of the 16 point "checklist". It is perfectly OK to exclude one of the 16 points – as long as the reasons why that has been excluded are listed (and agreed by the reviewers

of the MTP). Risks and assumptions are also included in the MTP; sometimes the explicit stating of a risk or assumption can promote lively discussion, and even resolution!

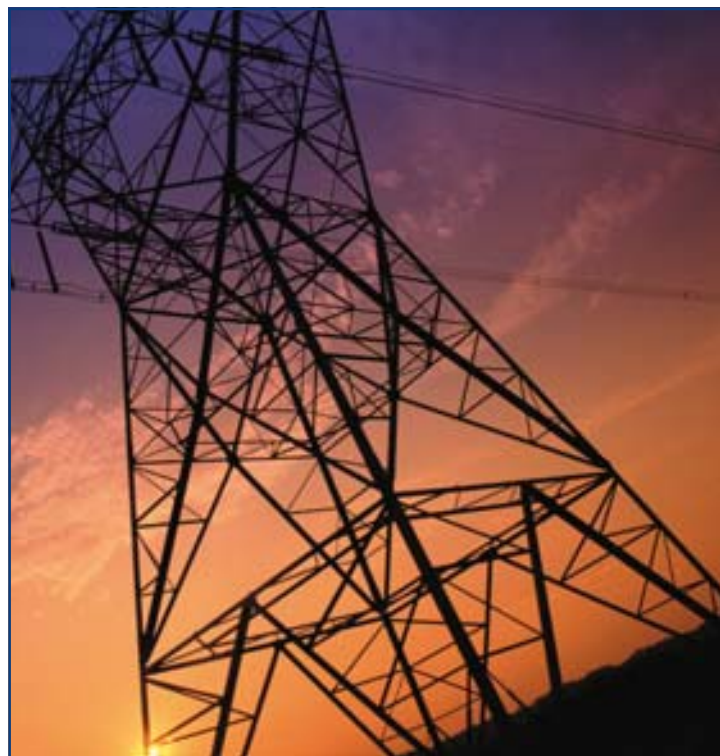
Relationship to other standards

IEEE 829 as a standard is not so much about how to test, but how to document that you have tested. These are some of the other standards that may be referred to when documenting according to IEEE 829:

- IEEE 1008 Standard for Unit testing,
- IEEE 1028 Standard for Software Reviews
- IEEE 1044 Standard Classification for Software Anomalies
- IEEE 1044-1 Guide to Classification for Software Anomalies
- BS 7925 Standard for Software Component Testing

Conclusion

IEEE 829 should be used as a standard appropriately, not blindly. In themselves, testers add nothing to the output of the project team; a tester does not make better software. Therefore, we need to slay the "documentation for documentation's sake" myth and ask ourselves "is this output enabling the test and/or development teams to do a better job, or helping the users understand what is being developed and whether it meets their needs?" IEEE 829 can help to make this the case by giving useful guidelines; it points the way to truly useful documentation. PT



e-testing[®]
consultancy



Special ISEB Training Offer!

3 days Foundation Course – only £725*
10 days Practitioner Course - £2000*
delivered in 3x3x2x2 sessions

*Course price includes
course notes, exam fees & lunches.

To book a place on any of the above
courses or for further information,

Please email training@etesting.com
or contact Caroline Barlow on 020-8387 1701

e-testing only operate in the Testing sector of IT and are dedicated to providing testing services to its clients. We are, therefore, always looking for testing resources to join our firm on permanent or contract basis to work on new projects. So to register your interest with us please complete our on-line form on www.etesting.com or send your current CV to jobs@etesting.com

e-testing Consultancy Limited
Kinetic Centre, Borehamwood,
Hertfordshire WD6 4PJ