PAS 754:2014

Software trustworthiness – Governance and management – Specification

MAKING SOFTWARE BETTER

HM Government

bsi.

Publishing and copyright information

ISBN 978 0 580 83242 0

ICS 35.040

Publication history

First published May 2014

Contents

Foreword

This PAS was sponsored by the Trustworthy Software Initiative (TSI), a public good activity supported by the UK Government National Cyber Security Programme (NCSP) on behalf of stakeholders from the public and private sectors and academia. Its development was facilitated by BSI Standards Limited and it was published under licence from The British Standards Institution. This PAS came in to effect on 30 May 2014.

Acknowledgement is given to the technical author Ian Bryant, from TSI, and to the following organizations that were involved in the development of this PAS as members of the steering group:

- Association of British Certification Bodies (ABCB)
- Centre for the Protection of National Infrastructure (CPNI)
- Department for Business, Innovation & Skills (BIS)
- Group 5 Training Limited
- The Institution of Engineering and Technology (IET)
- Microsoft
- The Motor Industry Software Reliability Association (MISRA)
- Nexor Limited
- Oxford Brookes University
- QinetiQ Group
- Trustworthy Software Initiative (TSI)

Acknowledgement is also given to the members of the wider review panel consulted in the development of this PAS.

This PAS is not to be regarded as a British Standard. It will be withdrawn upon publication of its content in, or as, a British Standard.

The PAS process enables a specification to be rapidly developed in order to fulfil an immediate need in industry. A PAS can be considered for further development as a British Standard, or constitute part of the UK input into the development of a European or International Standard.

Use of this document

It has been assumed in the preparation of this PAS that the execution of its provisions will be entrusted to appropriately qualified and experienced people, for whose use it has been produced.

Presentational conventions

The provisions of this standard are presented in roman (i.e. upright) type. Its requirements are expressed in sentences in which the principal auxiliary verb is "shall".

Commentary, explanation and general informative material is presented in smaller italic type, and does not constitute a normative element. The word "should" is used to express recommendations, the word "may" is used to express permissibility and the word "can" is used to express possibility, e.g. a consequence of an action or an event.

Spelling conforms to The Shorter Oxford English Dictionary. If a word has more than one spelling, the first spelling in the dictionary is used.

Contractual and legal considerations

This publication does not purport to include all the necessary provisions of a contract. Users are responsible for its correct application.

Compliance with this PAS cannot confer immunity from legal obligations.

Executive summary

From smart phones to power stations, airliners to e-commerce, our economy and society is increasingly dependent on software in many different guises. This makes software trustworthiness an underlying concern for all those who commission, write and use it.

This PAS, sponsored by the UK Trustworthy Software Initiative, is to provide a consensus specification for software trustworthiness, either as a stand-alone document or as a companion and complement to other relevant standards.

This specification identifies five aspects of software trustworthiness: safety, reliability, availability, resilience and security. The set of principles and techniques for any software implementation needs to be suited to the context and intended use.

This document describes a widely applicable approach to achieving software trustworthiness, which is based on the following concepts:

- Governance. Before producing or using any software which has a trustworthiness requirement, an appropriate set of governance and management measures shall be set up.
- Risk assessment. The risk assessment process involves considering the set of assets to be protected, the nature of the adversities that may be faced, and the way in which the software may be susceptible to such adversities.
- Control application. Risk shall be managed through the treatment of risk by the application of appropriate personnel, physical, procedural and technical controls.
- Compliance. A compliance regime shall be set up to ensure that creators and users of software ensure that governance, risk and control decisions have been implemented.

It also recommends the use of a trustworthy software management system, either as a standalone entity or by relevant extension to existing management system(s), including:

- creating a trustworthy software defect and deviation list;
- implementing control measures;
- creating a trustworthy software release authority;
- building a trustworthy software constraint and dependency model;
- using of trustworthy software release notices.

0 Introduction

0.1 Aim

The aim of this PAS, sponsored by the UK Trustworthy Software Initiative (TSI), is to provide a specification for software trustworthiness.

0.2 Objectives

This specification is intended to be widely applicable to software in its many guises from embedded equipment through consumer devices to industrial control systems. It aims to provide a consensus specification for software trustworthiness, either as a stand-alone document, or as a companion and complement to other relevant standards, by collating good practice from the five main facets of trustworthiness that currently typically operate in isolation (safety, reliability, availability, resilience and security).

In conjunction with methodologies such as *TickITplus*, a UK scheme that embraces quality management across IT in the form of a capability maturity method, and other similar frameworks PAS 754 could provide a foundation for software trustworthiness within organizations.

It supports the TSI's objectives as a public good initiative to improve software performance across organizations in all areas.

By helping to improve software quality, this specification could result in significant savings for the economy and reduce the risk of major disruptions to a range of industries across both the private and public sectors.

NOTE *See* Risk and Responsibility in a Hyperconnected World *[1].*

The requirements of PAS 754 can enable an organization to, for example:

- improve controls;
- improve operational effectiveness and efficiency;
- improve organizational learning.

These in turn can result in:

- improved stakeholder confidence and trust;
- increased likelihood of achieving objectives;
- reduced risk;
- enhanced business reputation.

0.3 Claims of conformance

0.3.1 General

An organization may claim conformance with PAS 754.

0.3.2 Form of claim

All claims are required to include a reference to PAS 754.

0.3.3 Basis of claim

A claim of conformance can be made on the basis of:

a) a first-party conformity assessment performed by the organization (self-assessment);

b) a second-party conformity assessment performed by, for example, a trade association; or

c) a third-party conformity assessment performed by an organization, such as a certification body, that is independent of both the organization and any linked trade association.

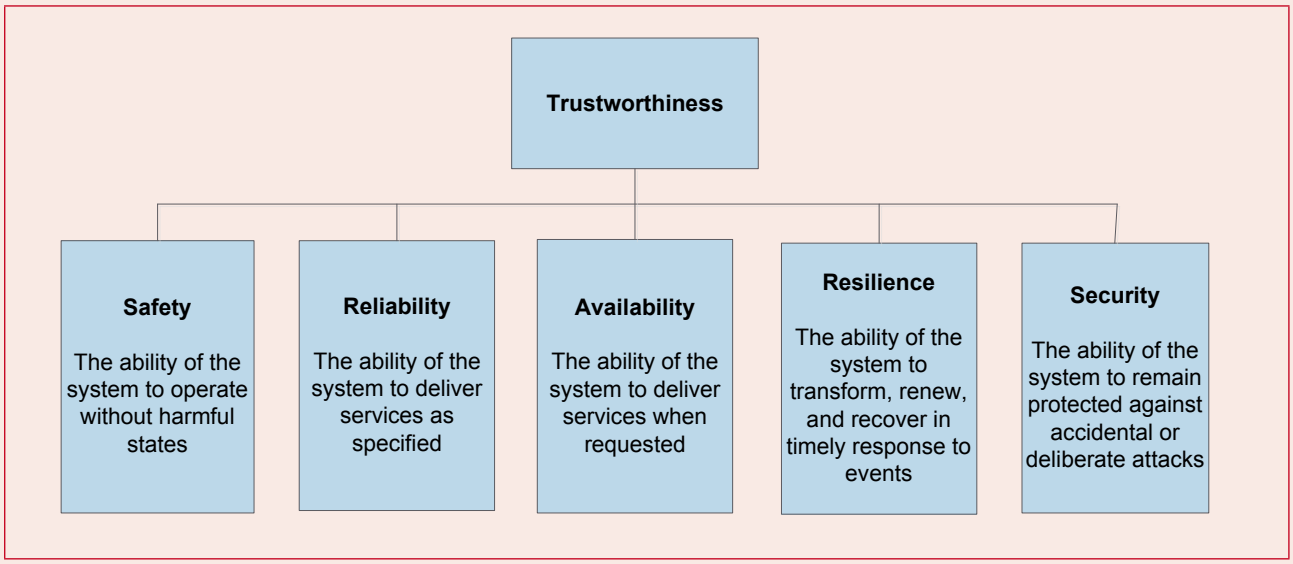
0.4 Context

0.4.1 Approach

The PAS is intended for any organization that seeks to establish or improve confidence in its software trustworthiness. It is applicable to all organizations regardless of their size, type and the nature of their business.

For this specification, software trustworthiness is identified as consisting of five facets, as described in Figure 1.

Figure 1 – Facets of trustworthiness



It is important that organizations review every software implementation to see which aspects apply and derive a set of principles and techniques to suit the context and intended use.

For each of these facets of trustworthiness there will be objectives, of varying complexity.

A common set of implied objectives apply to most software implementations, not least because of legal and regulatory requirements.

Safety aims to provide assurance that:

- the stated requirements are accurate and appropriate;
- safety issues are considered via safety requirements;
- the architecture and design are consistent with and reflect the stated requirements;
- technical defects are absent;
- application defects are absent;
- the stated requirements are identifiable in the low-level code and that all low-level code implements at least one stated requirement;
- the test data sets reflect the stated requirements;
- the test data sets cover the low-level code to a specified degree.

Reliability aims to provide assurance that:

- all the patterns of use are reflected in the stated requirements;
- there are no technical defects;
- the test data sets reflect patterns of use;
- there are no application defects.

Availability aims to provide assurance that:

- the stated requirements are accurate and appropriate;
- the architecture and design are consistent with and reflect the stated requirements;
- technical defects are absent;
- application defects are absent;
- the test data sets reflect the stated requirements.

Resilience aims to provide assurance that:

- the stated requirements are accurate and appropriate;
- the architecture and design are consistent with and reflect the stated requirements;
- technical defects are absent;
- application defects are absent;
- the test data sets reflect the stated requirements.

Security aims to provide assurance that:

- the security requirements consider all security issues;
- the architecture and design satisfy the security requirements;
- there are no security defects in the code;
- the test data reflects the security requirements.

0.4.2 Organizational controls

In order to deliver trustworthy software, an organization requires a set of underpinning controls that apply to all activities.

The software management system aims to provide assurance that:

- all personnel are appropriately qualified;
- adequate resources are allocated;
- all necessary communication takes place;
- activity proceeds in a series of measured steps;
- specific steps are performed independently;
- activity proceeds in a timely manner;
- all verification processes are completed within the specified criteria.

The software technical infrastructure aims to provide assurance that:

- all information, designs, algorithms and other such artefacts are retained for future use and analysis;
- the design and coding artefacts are adequately documented;
- all past and present versions of the software are available at any time and that future versions will similarly be available;
- all appropriate test data sets can be applied to the corresponding version and any future versions of the software;
- regression testing can be applied in order to ensure that the software changes only in the required manner.

0.4.3 Challenges

Software problems are generally characterized as one of three types:

- Weaknesses, which are generic classes of potential deficiency in software, such as buffer overflows.
- Vulnerabilities, which can be:
  - the existence of a generic weakness in a particular platform, such as a buffer overflow occurring in a specific operating system or application;
  - interactions between multiple software elements that bypass intended controls;
  - accidental actions of software developers that result in defects and errors;
  - deliberate actions of software developers that bypass intended controls, such as trap doors that permit unauthorized access to the system.
- Susceptibilities, which are the confirmed presence of one or more vulnerability within an implemented system, such as the presence of an operating system with a buffer overflow defect. Susceptibilities in systems stem from:
  - initial implementation;
  - changes to software, such as from adding new facilities or the correction of detected errors ('patching');
  - use of utility programs, which may be capable of circumventing security measures in the controlling or application software.

For the application of these terms specifically to software, see Clause **3**.

0.4.4 Tailoring

This PAS is scoped to include all aspects that contribute to trustworthiness of software, as illustrated in Figure 2.

This is achieved by using the appropriate elements of the consensus framework of measures – the trustworthy software framework (TSF) – decomposed as shown in Figure 3 and detailed in Clauses **5** and **6** and the Annexes.

This comprehensive trustworthy software framework (TSF) provides a domain- and implementation-agnostic way to reference the large existing body of knowledge, including functional safety, information security, and systems and software engineering and therefore acts as a collation of good practice for software trustworthiness.

When used as a stand-alone document for organizations with no current approach to software trustworthiness, this specification will facilitate the deployment of the TSF for software in its many guises from embedded equipment through consumer devices to industrial control systems.

For organizations that already address software trustworthiness though the lens of one or more of the five main facets of trustworthiness that typically operate in isolation (safety, reliability, availability, resilience and security), this specification provides a companion and complement to other relevant standards, and reviewing the concepts, principles and techniques in this specification alongside practices and management systems derived from individual facets allows the identification of gaps and enhancements.

Figure 2 – Aspects of trustworthiness

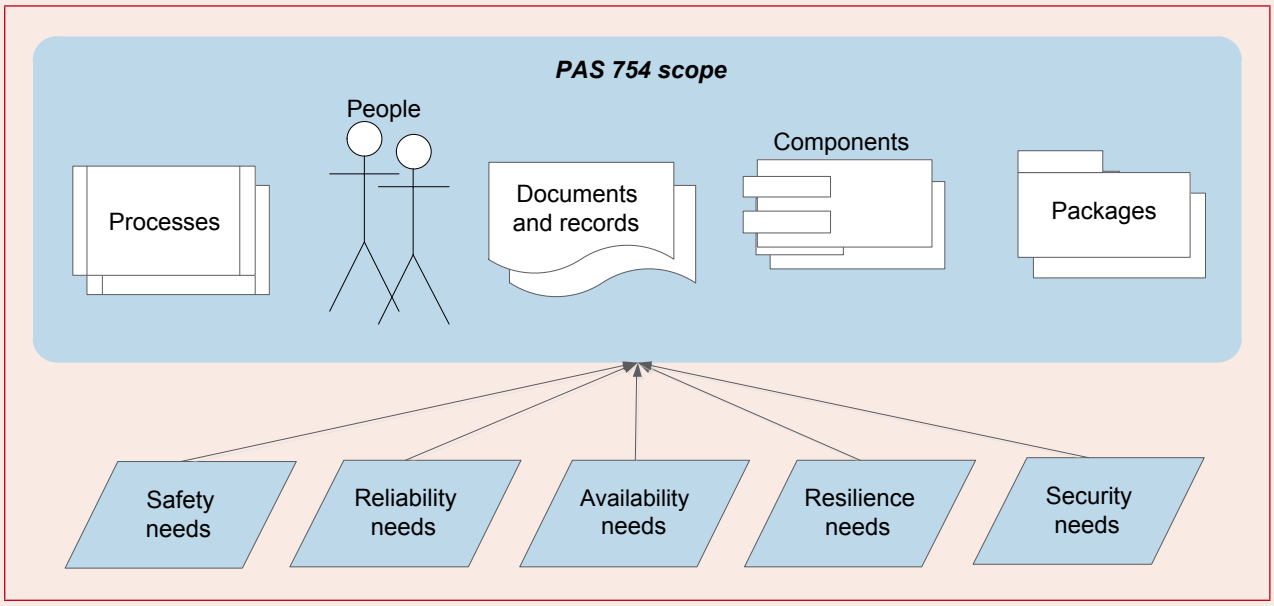
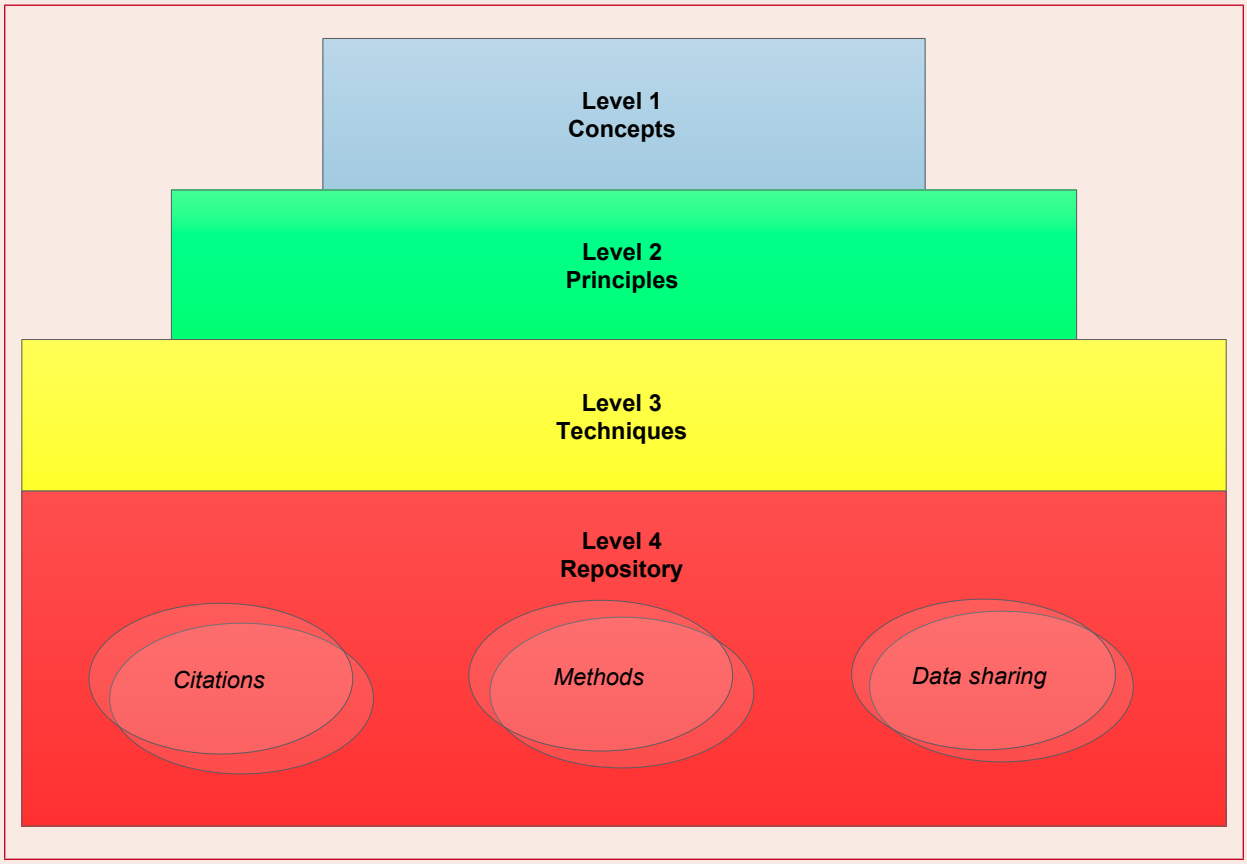


Figure 3 – Trustworthy software framework



The Trustworthy Software Initiative (TSI) is an independent UK organization supported by the public and private sectors and academia, which is charged with maintaining a repository of citations, methods and data sharing techniques about creating trustworthy software. More information is maintained on the website at: www.uk-tsi.org.

This PAS does not specify how any technique should be applied to a specific domain of application. This information is available in other standards, such as BS ISO/IEC 15408 and BS ISO/IEC 27001 for information security, and BS EN 61508 for functional safety.

0.4.5 Segmentation

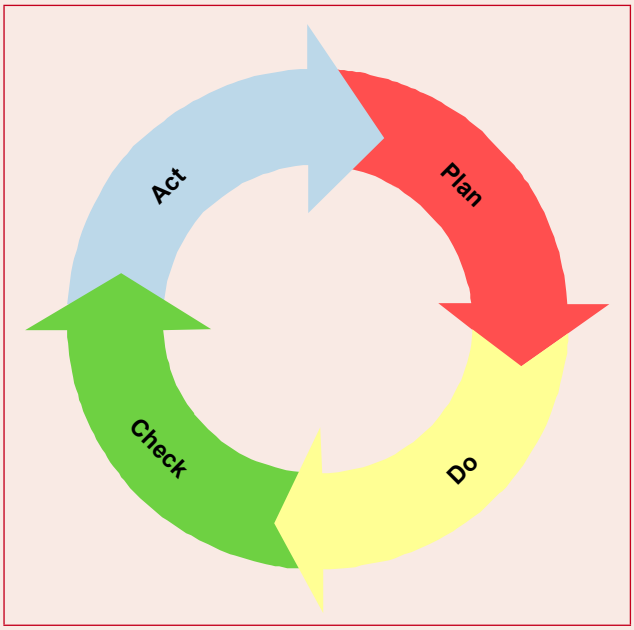
For the purposes of this PAS, the software audience can be divided into three groups:

- Mass Market with an Implicit Need[1)] (M/I) for software trustworthiness;
- Mass Market with an Explicit Need[2)] (M/E) for software trustworthiness;
- Niche Market with an Explicit Need[3)] (N/E) for software trustworthiness.

0.4.6 Continuous improvement

Continuous improvement of trustworthy software practices within an organization can be achieved using the PDCA (plan–do–check–act) cycle (see Figure 4).

Figure 4 – PDCA cycle



[1)] For the Mass Market with Implicit Needs, a majority of software trustworthiness requirements are perceived as non-functional e.g. a "non interference" property.
[2)] One or more functional requirements are for software trustworthiness.
[3)] For instance government and critical national infrastructure (CNI).

1 Scope

This PAS specifies requirements for software trustworthiness. It is intended to set out a widely applicable approach that can be customized for any organization and applied to software in its many guises from embedded equipment through consumer devices to industrial control systems.

This PAS defines the overall principles for effective software trustworthiness, and includes technical, physical, cultural and behavioural measures alongside effective leadership and governance. This PAS identifies the necessary tools, techniques and processes and addresses safety, reliability, availability, resilience and security issues.

This PAS does not specify the detailed processes or actions that an organization follows in order to achieve these outcomes.

NOTE 1 *These are defined in other standards, or can be defined by the organization.*

NOTE 2 *For organizations that already address software trustworthiness through the lens of one or more of the five main facets of trustworthiness that typically operate in isolation (safety, reliability, availability, resilience and security), this specification provides a companion and complement to other relevant standards, and reviewing the concepts, principles and techniques in this specification alongside practices and management systems derived from individual facets allows the identification of gaps and enhancements.*

This PAS is applicable to any organization aiming to adopt software trustworthiness practices.

2 Normative references

The following documents, in whole or part, are normatively referenced in this document and are indispensable for its application. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including and amendments) applies.

BS ISO/IEC 27001:2013, *Information technology – Security techniques, Information security management systems – Requirements*

BS ISO/IEC/IEEE 42010, *Systems and software engineering – Architecture description*

ISO/IEC 15288, *Systems and software engineering – System life cycle processes*