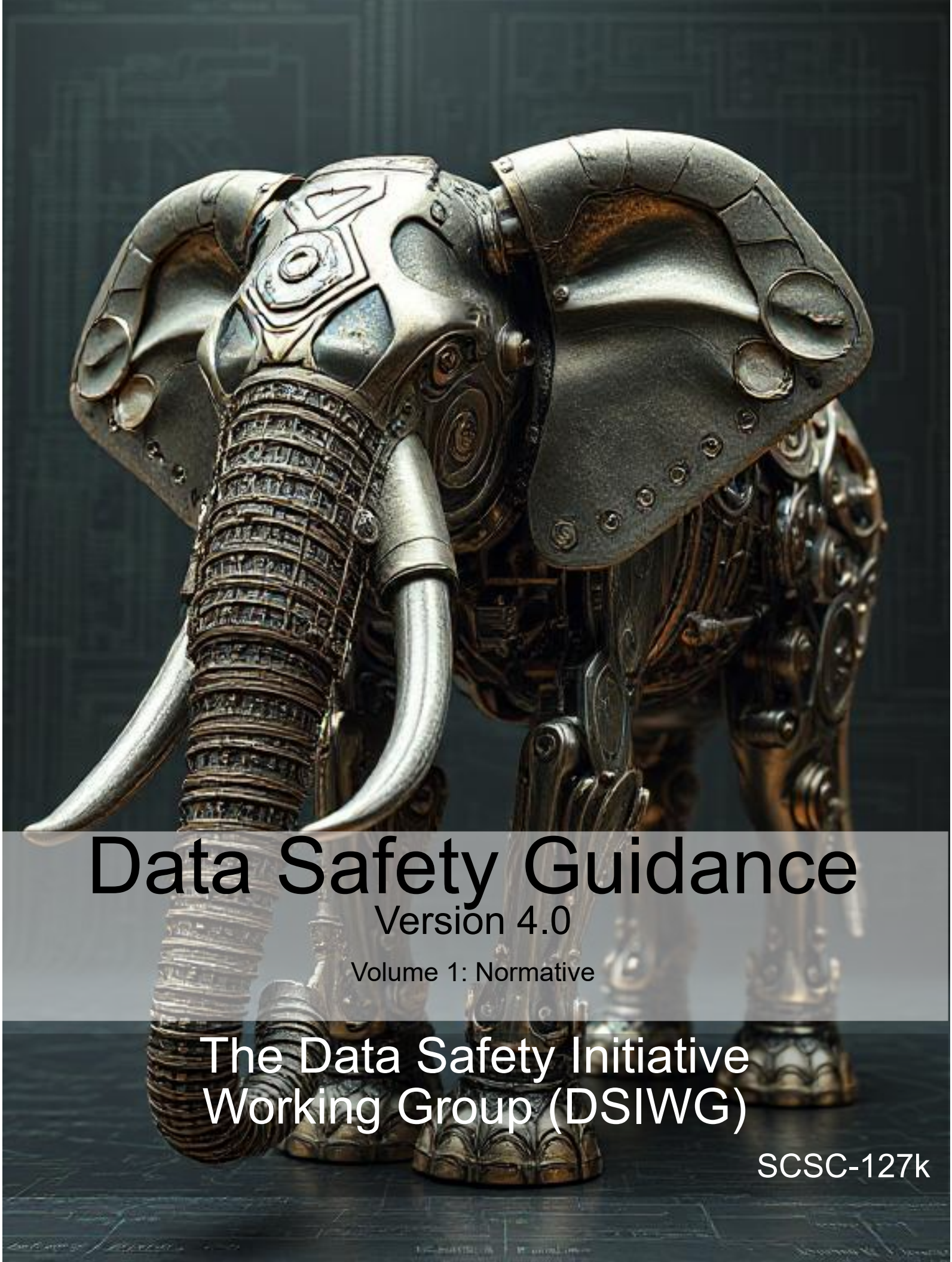

Data Safety Guidance

Version 4.0

Volume 1: Normative

The Data Safety Initiative Working Group (DSIWG)

SCSC-127k

SCSC Publication Number: SCSC-127J

This document was prepared using the LaTeX 2_ε typesetting system.

Editing and typesetting by Mark Templeton, supported by Tim Rowe.

Cover design by Paul Hampton.

The Safety-Critical Systems Club (SCSC) is the professional network for sharing knowledge regarding safety-critical systems. It brings together:

- engineers and specialists from a range of disciplines working on safety-critical systems in a wide variety of industries;
- academics researching the arena of safety-critical systems;
- providers of the tools and services that are needed to develop the systems; and
- the regulators who oversee safety.

Through publications, seminars, workshops, tutorials, a web site and, most importantly, at the annual Safety-critical Systems Symposium (SSS), it provides opportunities for these people to network and benefit from each other's experience in working hard at the accidents that don't happen. It focuses on current and emerging practices in safety engineering, software engineering, and product and process safety standards.

This document was written by the Data Safety Initiative Working Group (DSIWG), which is convened under the auspices of the SCSC. The document supports the DSIWG's vision, which is to have clear guidance that reflects emerging best practice on how data (as distinct from software and hardware) should be managed in a safety-related context. This update takes account of the consensus that a process-based guidance document will complement existing safety management processes, making it more usable. It was formally released at SSS'25, 4–6 February 2025, details of which may be found at https://scsc.uk/e1099

Comments on this document are actively encouraged. These can be emailed to:

comments@data-safety.scsc.uk.

Alternatively, a comments submission form is available at:

data-safety.scsc.uk/comments.

Data Safety Guidance

The Data Safety Initiative Working Group [DSIWG]

February 2026

This page is intentionally blank

Change History

Version	By	Status	Date
1.0	The DSIWG Team	First draft for external review	31-JAN-2014
1.1	The DSIWG Team	(Internal edition for DSIWG use only)	09-DEC-2014
1.2	The DSIWG Team	For publication at SSS'15	23-JAN-2015
1.3	The DSIWG Team	For publication at SSS'16	29-JAN-2016
2.0	The DSIWG Team	For publication at SSS'17	30-JAN-2017
3.0	The DSIWG Team	For publication at SSS'18	26-JAN-2018
3.1	The DSIWG Team	For publication at SSS'19	01-FEB-2019
3.2	The DSIWG Team	For publication at SSS'20	11-FEB-2020
3.3	The DSIWG Team	For publication at SSS'21	09-FEB-2021
3.4	The DSIWG Team	For publication at SSS'22	08-FEB-2022
3.5	The DSIWG Team	For publication at SSS'23	07-FEB-2023
3.6	The DSIWG Team	For publication at SSS'24	13-FEB-2024
3.7	The DSIWG Team	For publication at SSS'25	04-FEB-2025

Changes Since the Last Edition

The main changes in this edition are: This is a major rewrite of the guidance. The document has been split into three volumes:

- Volume 1 contains the normative guidance;
- Volume 2 contains the discursive guidance; and
- Volume 3 contains the informative guidance.

The process remains substantially the same, but some informative material has been made normative, so compliance with earlier versions of the guidance does not guarantee compliance with this version. The intention is that compliance with this version does imply compliance with earlier versions.

Future work

MCA Ltd has continued to work with the DSIWG to develop a prototype software tool to assist in the automation of the processes described in this guidance document. A working version of the tool has been developed and organizations that could benefit from the use and further development of the tool are urged to contact MCA at Mission Critical Applications Limited (radish@mca-ltd.com).

A number of improvements to the guidance are currently planned. These improvements are intended to clarify the application of the data safety process and include:

- further detail on the assurance of communications and data flows,
- data safety considerations associated with distributed datasets and Blockchain,

- addition of new treatments to the tables in **??**,
- review of the tables of treatments, with the aim of making them easier to use,
- further explanation of some treatments, where their use or benefit is not immediately apparent,
- reordering of parts of the document to improve readability, especially as regards likelihood,
- further detail on tool assurance,
- harmonisation of language and guidance on how organizations may expand the tables to incorporate their own internal processes.
- guidance on the application of the data safety culture questionnaire,

Several of these changes are likely to cause parts of the document to be re-ordered – they have therefore been deferred to the next major update, in version 4.0 of the guidance.

If you or your organization are interested in learning more about the work of the DSIWG or joining any of the sub-groups, please visit the SCSC website, where more information including contact details may be found on the "Working groups" section of the site.

Related working groups

The SCSC sponsors initiatives to develop methods and techniques through a number of working groups. These groups each address safety aspects peculiar to their domain, including data aspects when appropriate. The current list of working groups includes:

- Assurance Cases,
- Security Informed Safety,
- Safe AI Working Group,
- Safer Complex Systems.

This page is intentionally blank

Foreword

> *Data is here. Data is growing. Data is causing harm.*

Data is here: Data is becoming ever more important in our lives: influencing, managing and even controlling many critical aspects. The use of artificial intelligence (AI) systems is a new, exciting, but potentially hazardous use of data. Large language model (LLM) based systems are trained on vast amounts of data, and it is this data which enables them to be useful.

Some of this data is related to our personal safety and well-being. Consider, for example, the importance of data defining the layout of railway signals, data that indicates the position of underwater obstructions in nautical channels or data that is used to train a vision recognition system to detect tumours in medical images. Organizations now make significant decisions (including safety-related decisions) based solely on data held in systems. Hence, organizations need to safely manage, control and process their data. In particular, they must actively manage key data properties that preserve safety.

Data is growing: There are at least two reasons why the use of data has grown and, equally important, why it is expected to continue to grow. The first relates to the rapid expansion of the area loosely termed "Big Data", including the use of large datasets to support machine learning and AI applications. The second is the growing use of systems of systems, where data is the lifeblood that connects together disparate elements and allows a cohesive capability to be built. Put simply, the need to address data-related issues is a pressing problem and will continue to be so.

Data is causing harm: Strictly speaking, data can neither cause nor prevent harm. However, mistakes in data, or the inappropriate use of data, within safety-related systems have been factors in a number of documented accidents and incidents. Examples include aircraft attempting to take off from the wrong runway (and consequently crashing), ships running aground, and patients being exposed to higher than planned doses of radiation.

Against this background, the DSIWG was established under the auspices of the SCSC. The DSIWG's aim is to develop clear, cross-sector guidance that reflects emerging best practice on how data (as opposed to software or hardware) should be managed in a safety-related context. For the most part, this guidance is based on well-established techniques, and it has been designed to be compatible with current safety standards and to integrate with existing safety management systems. What is new, however, is the explicit and relentless focus on data, making it a "first-class citizen" within system safety analyses. Because of this focus, this guidance should help organizations identify, analyse, evaluate and treat data-related risks, thus reducing the likelihood of data-related issues causing harm in the future.

This page is intentionally blank

Quick Start Guide

> *Data really powers everything that we do.*
> ***Jeff Weiner***

This section provides a single-page introduction to data safety guidance (DSG). For first-time readers this should help place individual sections within an appropriate context. It should also help returning readers quickly navigate the document's contents.

- Systems are changing. The role of data is becoming more prominent. Hence, data needs to be considered as a "first-class citizen" in system safety analyses. This will help mitigate organizational and system-level risks associated with the use of data.
- A data safety management process has been developed. This is based on four phases:
  - establish context;
  - identify risks;
  - analyse risks; and
  - evaluate and treat risks.
- The underlying principles and an overview of the process are described in chapter 2.
- Normative definitions and abbreviations are described in chapter 3.
- The objectives associated with, and the outputs produced by, each phase are described in chapter 4.
- The activities of each phase (and associated tailoring information) are described in chapter 5.
- Additional guidance information for each phase is described in **??**.
- A worked example is provided in **??**.
- A collection of appendices provide more detail, including:
  - A discussion illustrating how the underlying principles link to the objectives (Appendix 6);
  - An organizational data risk (ODR) assessment questionnaire (Appendix **??**);
  - A data safety culture questionnaire (Appendix **??**);
  - A questionnaire to help assess the data maturity of a supplier (Appendix **??**);
  - A list of data categories (Appendix **??**);
  - A collection of hazard and operability study (HAZOP) guidewords (Appendix **??**);
  - The suggested contents of a data safety management plan (DSMP) (Appendix **??**);
  - A summary of accidents and incidents in which data was potentially a causal factor (Appendix **??**);
  - A discussion of topics loosely related to system lifecycles (Appendix 7);
  - Considerations regarding machine learning (ML) (Appendix **??**);
  - A discussion of the risks of AI and autonomy (Appendix **??**);
  - An introduction to the concepts of both dark and dazzle data (Appendix **??**);
  - The concepts of black swan, dragon king, perfect storm and Pudding Lane data (Appendix **??**).
  - Considerations for the assurance and qualification of data-handling tools (Appendix 8).

- An introduction to the RADISH tool, that has been developed to assist in the application of the guidance within this document (Appendix **??**).
- Issues that may arise when migrating, porting, importing or exporting data (Appendix **??**).
- Some of the data issues that made management of the Covid-19 virus difficult (Appendix **??**);
- Examples of ways that data safety assurance levels (DSALs) may be customised, with particular focus on likelihood (Appendix **??**);
- Lists of acronyms, definitions and glossary entries (Appendix 8.3); and
- A collection of references (Appendix 8.5).

This page is intentionally blank

This page is intentionally blank

Contents

This page is intentionally blank

List of Tables

This page is intentionally blank

List of Figures

This page is intentionally blank

Chapter 1

Introduction

extract principles

Chapter 2

Principles (Normative)

> *Errors using inadequate data are much less than those using no data at all.*
> ***Charles Babbage***

Hawkins *et. al.* established some generic software safety assurance principles, which are commonly referred to as "4 + 1" [**?**]. Given the close links between software and data it is helpful to consider these principles from a data safety assurance perspective. The results are detailed below, with each principle being considered in turn.

This page is intentionally blank

2.1 Principle 1: data safety requirements shall be defined to address the data contribution to system hazards

Data pervades active system operation, as well as the system's specification, realisation, verification, validation, certification, training, maintenance, and retirement. Moreover, data may be passed from one system to another, sometimes over a significant period of time. Data may be assimilated and converted from prior uses into new uses, or simply used as-is, by many systems. It is stored in media whose storage integrity decays. The system context for data safety requirements may be specific to a particular system's or process's use of the data, or it may be generalised to a class of related systems. Hence data safety requirements are needed for any safety-related system that interacts with data.

This page is intentionally blank

2.2 Principle 2: the intent of the data safety requirements shall be maintained throughout requirements decomposition

Data safety requirements establish the system's safety properties for data, for the system's use of data, for the management of data, and for the engineering lifecycle of both the system and its associated data. The system's requirements hierarchy must preserve the intent of the data safety requirements (and hence the system's safety-related data properties). Moreover, the applied engineering process for both the system's realisation and subsequent lifecycle stages shall demonstrate that the data safety properties are preserved.

This page is intentionally blank

2.3 Principle 3: data safety requirements shall be satisfied

Evidence is required that the system satisfies all of the data safety requirements imposed on it for all anticipated operating conditions. Moreover, the data safety requirements that pertain to the data's lifecycle outside of the system shall be evidentially demonstrated prior to the system acting on such data, or the system shall be able to adequately defend against unsatisfied data safety requirements. In other words, either the data can be shown to demonstrate the required data properties prior to being used or the system can implement adequate defences and mitigations against data that does not conform to the required safety properties.

This page is intentionally blank

2.4 Principle 4: Hazardous system behaviour arising from the system's use of data shall be identified and mitigated

This is an intentionally broad statement because data is conceptual and not physical; it is the contextualised use of data that could result in a system hazard. Data safety assurance principle 1 deals with system-level hazards arising from data, whereas Data safety assurance principle 4 is concerned with hazards that arise from the way the system uses its data, that is, whether the system's design and implementation introduce further hazards. An example is a ship navigation system's display of hydrographic chart data, where a wide field display results in small features disappearing (due to image scale) when it is critical that situational awareness of such hazards is maintained.

This page is intentionally blank

2.5 Principle 4+1: The confidence established in addressing the data safety assurance principles shall be commensurate to the contribution of the data to system risk

The confidence in the evidence that demonstrates establishment of the first four Data Safety Assurance Principles shall be proportionate to the contribution data (or a particular data artefact) makes to the system hazards.

This page is intentionally blank

Chapter 3

Definitions (Normative)

> *I love data. I think it's very important to get it right, and I think it's good to question it.*
> ***Mary Meeker***

This document is incomplete. The external file associated with the glossary 'normative' (which should be called `Vol1.normative-gls`) hasn't been created.

Check the contents of the file `Vol1.normative-glo`. If it's empty, that means you haven't indexed any of your entries in this glossary (using commands like `\gls` or `\glsadd`) so this list can't be generated. If the file isn't empty, the document build process hasn't been completed.

Try one of the following:

- Add `automake` to your package option list when you load `glossaries-extra.sty`. For example:
  ```
  \usepackage[automake]{glossaries-extra}
  ```
- Run the external (Lua) application:
  ```
  makeglossaries-lite.lua "Vol1"
  ```
- Run the external (Perl) application:
  ```
  makeglossaries "Vol1"
  ```

Then rerun LaTeX on this document.

This message will be removed once the problem has been fixed.

This page is intentionally blank

Chapter 4

Objectives and Outputs (Normative)

> *The goal is to turn data into information, and information into insight.*
> ***Carly Fiorina***

This page is intentionally blank

4.1 Establish Context

4.1.1 Objectives

1-1 System context and intended use SHALL be established.

1-2 Key stakeholders SHALL be identified.

1-3 Data artefacts SHALL be identified.

1-4 Interfaces SHALL be defined and managed.

1-5 A data safety assessment SHALL be planned.

4.1.2 Outputs

- A description of the system and its intended use. This SHOULD include an estimate of the level of data-related risks.
- A list of key stakeholders for data safety activities.
- A collection of data artefacts, described at an appropriate level of detail.
- An interface control plan or list of control measures. This MAY include a list of data owners, linked to data artefacts.
- A plan for the remaining parts of the data safety assessment.

This page is intentionally blank

4.2 Identify Risks

4.2.1 Objectives

2-1 Historical data-related accidents and incidents SHALL be reviewed.

2-2 Unintended behaviour resulting from data SHALL be identified and analysed.

2-3 Risks SHALL be identified and linked to data artefacts and data properties.

4.2.2 Outputs

- A description of the process used for risk identification.
- A list of risks, linked to data artefacts and data properties.
- An updated plan for the remaining parts of the data safety assessment, if required.

This page is intentionally blank

4.3 Analyse Risks

4.3.1 Objectives

3-1 DSALs SHALL be established.

3-2 DSALs SHALL be justified.

3-3 DSALs SHALL be incorporated into system safety activities.

4.3.2 Outputs

- A DSAL and supporting justification for each risk identified in the previous phase.
- An updated plan for the remaining parts of the data safety assessment, if required.

This page is intentionally blank

4.4 Evaluate and Treat Risks

4.4.1 Objectives

4-1 Data safety requirements SHALL be established and elaborated.

4-2 Methods used to provide data safety assurance SHALL be defined and implemented.

4-3 Compliance with data safety requirements SHALL be demonstrated.

4.4.2 Outputs

- A record of the agreed responses to each of the identified risks, along with supporting justification.
- A list of data safety requirements that follow from these responses.
- A record of the treatment adopted for each of the identified risks, including evidence that the treatment has been successfully implemented.
- An assessment as to whether the risk has been suitably mitigated (and, if not, plans for further mitigation activities).

This page is intentionally blank

Chapter 5

Activities and Tailoring (Informative)

> *It is a capital mistake to theorise before one has data.*
> ***Sherlock Holmes - "A Study in Scarlet" (Sir Arthur Conan Doyle)***

This page is intentionally blank

5.1 Phase 1: Establish Context

5.1.1 Overview

This phase involves developing:

- an understanding of the context within which the system development occurs;
- an understanding of the system requirements; and
- an understanding of the system design.

These factors help determine the risk appetite: essentially, how much effort will be devoted to making risks as low as practicable. In turn, this will inform the nature and scope of assessments that are conducted during system development, introduction to service, and operation. The factors also help identify stakeholders.

5.1.2 Activities

There are four activities associated with this phase.

5.1.2.1Describe the organizational context

Part of this activity involves understanding the system stakeholders. It is important to define how the stakeholders will interact, determine the derived requirements applicable to each stakeholder, and identify how these requirements will be managed through interface control (similarly to systems engineering interface control procedures already in place in many industries). External factors (e.g., economic, social, regulatory) and internal factors (e.g., cultural, processes, strategic) factors are key to defining appropriate interface control measures. Interface control may be iterative throughout the data safety assessment. In particular, interface control may need to be amended to take account of implemented data safety mitigations.

The ODR assessment (**??**) can be used to form a high-level understanding of each organization's risk. It can be used at programme level to cover all stakeholders or used by each individual stakeholder to determine the risk appropriate to their area. However, when adopting such a formulaic approach to risk assessment, care should be taken to ensure that the resulting system can meet its cumulative requirements and that that "fallen through the cracks". This is part of the requirements decomposition and verification / validation activities described later in the data safety management process.

Among other things, the ODR assessment considers:

- the severity of any potential accidents; organizational maturity;
- applicable legal and regulatory frameworks; and
- the size, complexity and novelty of the planned system.

It results in a rating from ODR0 (the lowest risk) to ODR4 (the highest risk). This rating provides an initial, top-level view of the magnitude of data-related risk. As such, it could form the basis for process tailoring; it

also gives an indication of the proportionate magnitude of effort that may be required in the management of data safety risks.

In cases where the ODR assessment has not been explicitly conducted, the qualitative scale presented in Table 5.1 is also used to allow tailoring to be applied.

Table 5.1: Qualitative Definition of ODR

ODR Rating	Qualitative Description
ODR4	High risk
ODR3	Medium risk
ODR2	
ODR1	Low risk
ODR0	Very low risk

In the case of an ODR0, no further work is required.

The ODR can be used to help identify key stakeholders and necessary approvers (i.e., those who need to formally accept the system). Some approvers might be within the organization, while others may represent external bodies. Approvers could be customers or regulatory authorities.

The process of establishing the internal context includes understanding organizational culture. A short data safety culture questionnaire is given in (**??**), which may help. It can be applied at an organization level or, more likely, within an individual project team. It could also be used to highlight the importance of data safety related issues within a project team, and before and after measurements could be taken to establish the effectiveness of data safety related training.

5.1.2.2Describe the system context

This activity is concerned with describing the system under analysis, as well as the key external influences on that system (for example, interfacing systems and human operators). There are many aspects to this activity; for brevity, only aspects directly relevant to data safety are discussed here.

When describing the system, it is often helpful to think in terms of producers and consumers of data. These may be external systems, sub-systems, or a combination of both. It may also be necessary to consider data supply chains, especially when there are a number of separate organizations involved. Note that this activity is to be addressed at a high level. The identification of specific pieces of data is a separate, but related, activity, and the identification of required properties is another activity.

As development progresses, the system description is likely to be refined. This may enable refinement of the data safety system context, supporting the data safety assessment planning.

5.1.2.3Plan the assessment

This activity involves scheduling the phases associated with the data safety management process and acquiring the necessary resources to complete them. It also involves tailoring the generic process to meet the specific needs of a particular system development.

Details of the planned assessment may be recorded in a DSMP. This could also be used to capture the scope of the analyses and the associated context. Together, this constitutes the first section of the DSMP. Note that if a DSMP is used, it would typically be updated with details from subsequent phases. Alternatively, if a safety management plan is used for the project, data safety aspects may be included in the project safety management plan.

Planning of the data safety assessment requires some knowledge of the quantity and complexity of data that requires assessment. Therefore, the DSMP (or project safety management plan) needs to be updated in subsequent phases once these details are known.

Planning of the assessment may be done through a procurement process. Procurers may wish to know their potential supplier's understanding of data safety and their plans to implement a data safety assessment. A data safety supplier questionnaire is given in (**??**) and may be used to determine the potential supplier's understanding and plans, and for auditing purposes.

5.1.2.4Identify data artefacts

Data artefacts are the key pieces of data that are generated, processed or consumed by the system, or used for training in its use or maintenance. They provide the foundation for the remaining phases of data risk management.

A wide variety of Data Categories have been enumerated to support the identification of data artefacts. Cross-referencing these categories against the system description can highlight the relevant artefacts.

Another way of identifying data artefacts is to consider the functions that the system performs and to establish the data that is required to support these.

A way to confirm that all relevant data artefacts have been identified is to consider the different phases of the system lifecycle. This can help prevent an inappropriate focus on operational use of the system at the expense of, for example, data artefacts associated with system test and evaluation.

5.1.3 Tailoring

The level of stakeholder interface control required will depend heavily on the number of stakeholders, complexity of their interactions, and the contractual controls already in place. Many programmes already require interface control documents (ICDs) to be developed for systems or equipment. The level of detail required in other programme interface control plans may be used as a guide for the requirements of data safety interface control.

The guidance in this document is general in nature, and so it is likely to need tailoring to align with the organization's attitude to risk, their existing processes and the relevant sector's regulatory environment. The provided ODR, or a version customised to suit the organization's needs, may therefore provide a structured approach to assessment. The ODR assessment can be conducted at the product line or the individual product level, as appropriate for the organization. It is generally not recommended to conduct the ODR without the context of a system type.

The ODR assessment is likely to be most useful for organizations that do not have significant safety engineering experience and that are operating in industrial sectors that are not strongly regulated. organizations with considerable experience in the development of safety-critical systems in heavily

regulated environments may also find it useful for defining data context and for augmenting any existing safety standards which do not explicitly consider data. If this assessment is not conducted, some high-level qualitative estimate of risk may still be required (e.g., to support process tailoring); likewise, there will also be a need to identify key stakeholders and necessary approvers.

The Data Safety Culture questionnaire is likely to be of most use for low-risk (i.e., ODR1) systems. Developers of higher risk systems will typically have existing processes to develop, maintain and monitor safety cultures, although the data-oriented questionnaire could help inform those existing processes.

Including data safety within a project safety management plan is recommended for complex or highly safety-critical systems. In this case the structure of the safety management plan may be maintained, with the data safety assessment process being tailored to align to the overall safety assessment process.

The data maturity questionnaire is likely to be of most use when new organizational relationships are being formed. It may offer little value in situations where both organizations are familiar with each other, they have worked on data-related projects together before and there are suitable audit / review arrangements in place.

Data artefacts may be defined at a number of levels. A data artefact associated with a medical system could be described as "patient data". Alternatively, this could be split into smaller parts (e.g., "blood group"). Generally, the highest possible level consistent with the system description should be used; this prevents an excessively long list of data artefacts being developed. If necessary, where further detail is needed those data artefacts can be refined as part of an iterative process focused on key issues.

Not every data category will be relevant to every system. Furthermore, for low-risk (ODR1) systems it may be sufficient to simply consider the groupings of categories (e.g., "context", "implementation", etc.). Conversely, high-risk (ODR4) systems might need to consider every category, even if this results in a conclusion that a specific category is not relevant for the system in question.

A function-based approach to identifying data artefacts is likely to be enabled by design processes that also adopt a function-based perspective. If information from a function-based perspective is readily available, then it should be used to support the identification of data artefacts. If this information is not readily available, it is recommended that it be generated for medium and high-risk systems (i.e., ODR2 to ODR4, inclusive).

Considering data across the system lifecycle is a relatively simple activity, which is applicable to all systems (i.e., ODR1 to ODR4, inclusive).

5.2 Phase 2: Identify Risks

5.2.1 Overview

This phase involves identifying sources of risk and understanding the potential consequences. It should result in a comprehensive list of risks. These activities are likely to be concurrent with the development of more detailed system designs.

5.2.2 Activities

There are three complementary activities that can be used to identify risks. There is also an activity associated with updating planning documents.

5.2.2.1Review the general, historical perspective

Some insight into potential risks may be gained by reviewing historical accidents and incidents, a collection of which is included in **??** of this document. Each domain might have its own catalogue of historical incidents, which can be consulted during a data-focused review.

5.2.2.2Conduct a top-down approach

If the system under consideration has clearly identified functions, data-related risks can be assessed by considering each function in turn and analysing what data artefacts and, more particularly, what data properties the function depends on.

If there are a limited number of safety-related functions, this is usually the simplest approach. This approach also has the advantage that it integrates well with other function-based, top-down approaches to assessing system safety.

5.2.2.3Conduct a bottom-up approach

This approach starts from the data artefacts and explores the effects of data errors. In this context an error is a situation where a required data property is not exhibited. This may be achieved by a variety of methods, including a HAZOP.

5.2.2.4Update planning documents

Once the data safety risks have been identified, the DSMP requires review to determine if it needs updating to take account of the quantity and complexity of the analysis and mitigation activities needed to address the risks. While the DSMP might be updated throughout the data safety assessment, updates may not be required for all projects.

5.2.3 Tailoring

The general, historical perspective review is a simple activity that does not require significant resources. It is recommended for all systems, regardless of risk level.

When conducting a bottom-up approach, it may not be appropriate to explicitly consider every possible property for every single artefact. For low-risk (ODR1) and medium-risk (ODR2 / ODR3) systems some form of tailoring may be expected. This might, for example, take the form of pre-selecting the properties that are most relevant, or limiting the layer of abstraction at which the system is considered.

Tailoring of the bottom-up approach may also be appropriate for some high-risk (ODR4) systems, but in this case an explicit argument that the tailoring has not adversely affected system safety would be expected. Furthermore, the risk identification process for high-risk (ODR4) systems is expected to be highly structured. To support this, a number of data-related HAZOP guidewords are presented in **??**.

The top-down and bottom-up approaches provide different perspectives on data-related risks. For low-risk (ODR1) systems it may be appropriate to consider just one of these perspectives. Both perspectives would be expected to be considered (to some degree) for medium-risk (ODR2 / ODR3) and high-risk (ODR4) systems.

5.3 Phase 3: Analyse Risks

5.3.1 Overview

This part of the risk management process involves developing an understanding of the consequences and likelihood of each risk. This understanding allows system (or safety) integrity levels or Development Assurance Levels to be determined. This understanding should be used to allocate DSALs.

5.3.2 Activities

There are two activities associated with this phase.

5.3.2.1Establish DSALs

The key activity in this phase is to establish the (untreated) likelihood and severity of each risk identified in the preceding phase.

To analyse risks and, more particularly, to align data safety with other risk management processes, problems stemming from the use of the term "likelihood" in situations where there may be no failure rates need to be overcome. For this reason the DSAL was developed. The DSAL metric is not a statistical measure of likelihood, or a literal numeric measure of integrity. Instead, the DSAL metric is an indicator for the level of rigour that an assurance argument requires. As such, DSALs share a common theoretical basis with concepts like item development assurance levels (IDALs) [**?**] and development process systematic capability [**?**].

DSALs are measured on a scale of DSAL0 (lowest-assurance) to DSAL4 (highest-assurance). They are typically allocated as indicated in Table 5.2.

Table 5.2: DSAL "risk" matrix

Severity	Likelihood		
	High	Medium	Low
Minor	DSAL1	DSAL0	DSAL0
Moderate	DSAL2	DSAL1	DSAL0
Significant	DSAL3	DSAL2	DSAL1
Major	DSAL4	DSAL3	DSAL2
Catastrophic	DSAL4	DSAL4	DSAL3

Definitions for severity and likelihood in Table 5.2 may be customised for a specific application. A default approach to the assessment of likelihood is presented in section **??** and **??**, while default definitions for severity are presented in **??**.

Although this allocation of DSALs is typically used, there are some situations where a different allocation matrix may be more appropriate. Hence, it may be appropriate for a tailored allocation matrix to be used. Regardless of whether tailoring is used, the matrix should be reviewed and confirmed as being suitable for the intended application.

As their name suggests, DSALs are focused on safety concerns. However, the framework of data artefacts, data properties, and so on, developed in this document could also be applied to other concerns. It could, for example, be used for to control data-related financial risks, or data-related reputational risks. In these types of approach, the severity terms would obviously relate to financial and reputational consequences, rather than safety ones.

The additional understanding developed during this part of the process may mean some previously identified data artefacts are no longer of consequence. Similarly, it is possible that this process may identify additional data artefacts or a need to refine the description of existing artefacts.

5.3.2.2Analyse DSALs as part of system safety activities

Allocating a DSAL is a significant part of controlling data safety risks, but it is not the only part. It is important that DSALs are considered as part of wider system safety activities, rather than being viewed as a separate item.

For medium-risk (ODR2 / ODR3) and high-risk (ODR4) systems it is likely that integrity, or assurance, levels will be calculated from perspectives other than data safety. Possible examples include item / function development assurance levels from aerospace recommended practices (ARPs) 4754A [**?**] and safety integrity levels from IEC 61508 [**?**]. Where such an approach is used, the mapping to DSALs should be included within the DSMP.

This activity involves comparing DSALs with integrity, or assurance, levels. Assuming a typical scenario of a system processing or manipulating data flowing through it, there are two cases to consider:

1. Can the data affect the software? In particular, can the data affect the software such that the safe operation of the system is jeopardised? An ideal system would be able to handle any data fed into it safely without problems, but this is often not the case. An example might be a legacy system which has limited error checking and so may fail in unsafe ways if fed with data which is outside of the expected range. Formally: *given a system containing software written to a particular software assurance level (which may be none), what should the DSAL of the processed data be to preserve correct operation of the system?*
2. Can the software affect the data? In particular, can the system's software affect the data being processed or manipulated in such a way that data properties that are important for safety might be lost? Some examples might be systems which transform messages, losing any associated checksum protections, thereby possibly affecting the integrity of the data within the message; as a minimum, this removes a means of checking data integrity. Another example might be a system that can delay data flowing through it, (e.g., due to buffering) when timely delivery of the data is critical. Formally: *given data at a particular DSAL, what should the software assurance level of the software in the system be to preserve the DSAL of the data?*

5.3.3 Tailoring

DSALs can be applied at different levels and to different constructs. For example, in the case of simple, low-risk systems it may be appropriate to apply a single DSAL to an entire system. Alternatively, it may be appropriate to apply DSALs to sub-systems, for example, to match the level at which other integrity, or assurance, levels have been determined.

Another option is to apply DSALs to data artefacts rather than directly to risks. This approach has the

advantage that treatments are often related to data artefacts; it can work well where there is a simple relationship between data artefacts and risks.

This page is intentionally blank

5.4 Phase 4: Evaluate and Treat Risks

5.4.1 Overview

This phase involves deciding, at a generic level, what action (if any) should be taken for each of the risks identified in preceding phases. This decision will be influenced by the organization's risk appetite and other factors determined as part of the establish context phase. From some perspectives it may seem strange that requirements are identified at such a late phase. This is a consequence of explicitly linking data safety requirements to risks associated with data properties of data artefacts, and the use of DSALs to describe levels of rigour.

This phase also involves identifying, implementing and verifying treatments for the risks emerging from the previous phase. Verifying the treatment includes checking technical details of the chosen approach and re-assessing the post-treatment risk to determine whether it is now acceptable.

5.4.2 Activities

In this phase each risk, including the associated DSAL is reviewed and appropriate response are determined. This phase aims to answer the question: can we accept this risk or does some action need to be taken? This is likely to require discussion amongst a number of stakeholders. From a system safety perspective, there is nothing intrinsically special about data-related risks. Hence, it is recommended that evaluation of data-related risks be conducted alongside the evaluation of other system risks, as part of an organization's standard risk evaluation process.

Risks may be managed in different ways, for example:

Avoid Risk avoidance can be employed where a risk can be eliminated by using different approaches to the design and / or operation of the system. It may also be the case that very significant risks cannot be adequately treated and the only option is to avoid the untenable risk by not proceeding with the project.

Accept For low likelihood and low severity risks (e.g., those ranked as DSAL0), where the cost of further risk reduction is judged to be unacceptable, the risk may be accepted as-is and managed as such. Appropriate justification is likely to be required for acceptance of risks ranked higher than DSAL0.

Transfer Ownership of the risk's consequence can be transferred to another organization, for example by taking out an insurance policy. It is important that any such risk transfers are formally documented, understood and agreed by both parties.

Treat The risk can be reduced. This can be achieved by reducing the severity, or the likelihood or both. Choosing this response often involves having an outline view of how the risk may be reduced.

In addition to deciding on and documenting the appropriate response to each risk, this phase also includes gaining approval for these decisions.

If a decision is made to treat a risk, suitable methods and approaches should be identified. A range of potential methods and approaches are included in this guidance. These are mapped against DSALs, data properties and a selection of lifecycle data categories.

Once a treatment strategy has been established and implemented, whether the expected risk reduction has been achieved needs to be determined. Equivalently, there is a need to consider whether the residual risk may now be accepted or whether another one of the responses identified above is necessary.

5.4.3 Tailoring

Records of the discussions that occur as part of risk evaluation should be kept. For low-risk (ODR1) systems, this may be in the form of a brief memo. For high-risk (ODR4) systems, a detailed, structured record, which is placed under formal control, may be required; in this case these discussions may be recorded as part of the system's hazard log or as part of a data safety management plan.

As outlined in section 5.3.3, DSALs can be applied at varying levels of abstraction. For small-scale or low-risk (ODR1) systems it may be appropriate to consider treatments at higher levels of system abstraction. For example, this could involve applying a single DSAL to a sub-system or even to the system in its entirety and implementing risk treatment techniques at that level. The latter approach could be appropriate where the data in the system interacts in complex ways and the associated safety risk does not warrant a detailed investigation of these interactions.

A significant amount of tailoring is implicit in the way the tables of methods and approaches are constructed. At best a method / approach may be highly recommended as a way of maintaining a required data property at a given DSAL. The tables are not exhaustive; additional or alternative methods and approaches can be used.

Chapter 6

Linking Principles and Objectives (Informative)

> *I'm a bit of a freak for evidence-based analysis. I strongly believe in data.*
> ***Gus O'Donnell***

This page is intentionally blank

6.1 General

The data safety assurance principles provide the underpinning philosophy for DSG. Conversely, an implementation of the guidance would be based around the objectives. It is thus appropriate to consider how meeting the objectives results in the principles being satisfied. To that end, each principle is considered in turn in the following paragraphs.

This page is intentionally blank

6.2 Principle 1

***Data safety requirements* shall be defined to address the data contribution to system hazards.**

This principle asks that requirements be defined. Hence, it is related to the objective that:

- 4-1 Data safety requirements SHALL be established and elaborated.

However, that relationship does not tell the full story. In particular, the principle is focused at a system level, whereas the above objective is most likely to apply at more detailed levels of design. The following two objectives provide a system-level perspective on risks (from which specific requirements are developed) and hence both objectives directly support this principle:

- 1-3 Data artefacts SHALL be identified.
- 2-3 Risks SHALL be identified and linked to data artefacts and data properties.

This page is intentionally blank

6.3 Principle 2

The intent of data safety requirements shall be maintained throughout requirements decomposition.

This principle is based on standard systems engineering practices whereby a system is gradually developed at increasing levels of design detail. In order to cater for a wide range of systems, across a wide range of economic sectors, DSG does not specifically require an explicit hierarchical decomposition of requirements. However, it does note that, if necessary, data artefacts may be defined at a number of levels of increasing detail. Hence, the following two objectives are related to principle 2:

- 1-3 Data artefacts SHALL be identified.
- 2-3 Risks SHALL be identified and linked to data artefacts and data properties.

At a more fundamental level, this principle is concerned with translating from high-level requirements to something that can be implemented. DSALs are a key element of this translation. As such, the following three objectives are also relevant:

- 3-1 DSALs SHALL be established.
- 3-2 DSALs SHALL be justified.
- 3-3 DSALs SHALL be incorporated into system safety activities.

Note that the third of these objectives could also provide a direct link to hierarchical decomposition, if that activity is part of the system safety activities conducted by the organization implementing DSG.

As noted earlier, this principle is about identifying low-level design descriptions that, firstly, satisfy the intent of the high-level requirements and, secondly, are described in sufficient detail to allow them to be implemented (and for this implementation to be verified). From the perspective of DSG these low-level items are data safety requirements. Consequently, the following objective also supports this principle:

- 4-1 Data safety requirements SHALL be established and elaborated.

This page is intentionally blank

6.4 Principle 3

***Data safety requirements* shall be satisfied.**

This principle is straightforward. It involves implementing the low-level design descriptions and verifying these implementations. As such, it is directly supported by the following objectives:

- 4-2 Methods used to provide Data Safety assurance SHALL be defined and implemented.
- 4-3 Compliance with data safety requirements SHALL be demonstrated.

This page is intentionally blank

6.5 Principle 4

Hazardous system behaviour arising from the system's use of data shall be identified and mitigated.

From one perspective this principle is about looking bottom-up to determine whether the detailed design decisions have introduced any new system-level risks. A HAZOP is one way this can be achieved. Similarly, a HAZOP is one of several techniques that DSG suggests can be used to achieve the following objectives, which consequently may support principle 4:

- 2-2 Unintended behaviour resulting from data SHALL be identified and analysed.
- 2-3 Risks SHALL be identified and linked to data artefacts and data properties.

More generally, identifying potential new system-level hazards introduced by detailed design decisions involves looking at the system from a variety of perspectives. One perspective that is useful is provided by historical accidents and incidents; another useful perspective is provided by top-level generic data-related issues, distilled from experience across a wide range of systems and activities. Hence, the following objective also supports principle 4:

- 2-1 Historical data-related accidents and incidents SHALL be reviewed.

In addition, understanding the system context and intended use, as well as perspectives provided by a suitably wide collection of stakeholders can inform risk considerations. Likewise, potential issues can also be identified by considering the boundaries of the system. It follows that the following three objectives are also relevant to principle 4:

- 1-1 System context and intended use SHALL be established.
- 1-2 Key stakeholders SHALL be identified.
- 1-4 Interfaces SHALL be defined and managed.

This page is intentionally blank

6.6 Principle 4 + 1

The confidence established in addressing the data safety assurance principles shall be commensurate to the contribution of data to system risk.

This principle provides a means of balancing available effort against risk. From the perspective of DSG, this is provided by DSALs. As such, this principle is directly supported by the following three objectives:

- 3-1 DSALs SHALL be established.
- 3-2 DSALs SHALL be justified.
- 3-3 DSALs SHALL be incorporated into system safety activities.

This page is intentionally blank

6.7 Summary Table

For ease of reference, Table 6.1 summarises links between the principles and objectives; these are shown by an "X" in the relevant cell. For completeness, this table shows the phase associated with each group of objectives.

Two things are apparent from this table. Firstly, each principle is supported by at least two objectives. Secondly, with one exception, each objective supports at least one principle. The exception is the objective that "A data safety assessment SHALL be planned", which acts as an overarching objective to ensure there is sufficient resource to meet the other objectives. For this reason it is loosely associated with each principle; this is shown by an "o" in relevant cells.

Having each principle supported by at least two objectives, along with the descriptive text above, provides confidence that meeting the objectives will satisfy the principles; equivalently, it provides confident that the collection of objectives is *sufficient* to satisfy the principles.

In addition, every objective supporting at least one principle (with the exception noted above) indicates that there is value in each objective being included in DSG. This observation is not quite enough to demonstrate that the collection of objectives is *necessary* to satisfy the principles. However, given the small number of objectives and the apparent lack of overlap between them, it is sufficient to suggest the necessity of the objectives.

Table 6.1: Principles and objectives: summary table

		Principle				
		P1	**P2**	**P3**	**P4**	**P4+1**
Establish Context						
1-1	System context and intended use SHALL be established.				X	
1-2	Key stakeholders SHALL be identified.				X	
1-3	Data artefacts SHALL be identified.	X	X			
1-4	Interfaces SHALL be defined and managed.				X	
1-5	A Data safety assessment SHALL be planned.	o	o	o	o	o
Identify Risks						
2-1	Historical data-related accidents and incidents SHALL be reviewed.				X	
2-2	Unintended behaviour resulting from data SHALL be identified and analysed.				X	
2-3	Risks SHALL be identified and linked to data artefacts and data properties.	X	X		X	
Analyse Risks						
3-1	DSALs SHALL be established.		X			X
3-2	DSALs SHALL be justified.		X			X
3-3	DSALs SHALL be incorporated into system safety activities.		X			X
Evaluate and Treat Risks						

Continued on next page

Table 6.1: Principles and objectives: summary table (continued)

		Principle				
		P1	**P2**	**P3**	**P4**	**P4+1**
4-1	Data safety requirements SHALL be established and elaborated.	X	X			
4-2	Methods used to provide Data Safety assurance SHALL be defined and implemented.			X		
4-3	Compliance with data safety requirements SHALL be demonstrated.			X		

Chapter 7

Lifecycle Considerations (Discursive)

> *Failure is an amazing data point that tells you which direction not to go.*
> ***Payal Kadakia***

This page is intentionally blank

7.1 Usage Scenarios

If safety-related data is incorrect it can become dangerous when used, either by making a computer or control system perform incorrect actions, or by misleading human users into making incorrect decisions. Since the danger can only be determined when the usage of the data is understood, risk assessment should involve both the consumer of the data and the producer.

Figure 7.1: Consumer-focused integrity requirements

The consumer assesses the use of the safety-related data. (In later phases of the data safety management process this information is used to define the required data properties: for example, how accurate a particular safety-related data artefact must be.)

The producer investigates how the safety-related data is collected and what errors might occur. (Building on activities in later phases of the data safety management process, the producer can provide some form of guarantee, or level of confidence, that the safety-related data meets the specific data-related requirements.)

In some cases a producer will be providing safety-related data without any knowledge of a specific user (e.g., mapping data or generic databases that are sold to many users). In these cases the producer will need to make some assumptions about possible users, and then clearly state what level of integrity the data has been produced to. It is then up to the users to check whether the declared integrity matches their need.

This page is intentionally blank

7.2 Data in System Lifecycles

Like other components of a safety-related system, the safety dependency of data is dictated by the context in which it is used and the causal links that become established where loss of one or more of the required properties can contribute to hazardous system states. For example, a given dataset (say configuration data) could be used in a number of separate contexts such as:

- prototyping a system to demonstrate solution feasibility of a safety-related system;
- development testing of a safety-related system; and
- live operational use of a safety-related system.

In these cases, the dataset is the same but the context of its use changes the safety significance and therefore the level of assurance that it may require. It follows that the DSAL of a dataset is also predicated on where and when in the lifecycle the dataset will be applied.

To illustrate this concept, a number of generic model lifecycles are discussed below. Note that these are not intended to be prescriptive or mandate the use of any particular model. Instead, they are being used to illustrate how the Data Safety Management Plan could articulate these lifecycle considerations.

Development: the diagram in Figure 7.2 represents a typical development lifecycle using an iterative development approach[1]. In this model there are key phases as the system transitions from concept through to testable executable code. The process is iterative in that several cycles of functional elaboration, design, development and test may be run and these typically will focus on the areas of the system that bear most technical risk or comprise the key functional use cases so the client gets early visibility of the system. This early awareness allows feedback to be provided into the next iteration to help steer the solution to the client's actual needs. Traditional waterfall implementation can map onto this model on the basis that there is only one iteration in each phase and all activities in one phase need to be completed before progressing to the next.

The model itself may vary depending on the specific needs of the project but the diagram illustrates that different data categories become significant at different points of the process.

Operational Once a system has been developed it will move into an operational lifecycle or indeed, if data safety has not previously been considered for an enterprise, then the system could already be in operational use. These operational lifecycles tend to be cyclical in nature; the diagram in Figure 7.3[2] illustrates a typical model.

Again, specific data will come into play at different periods in the process. Documenting the relationship between process steps and data categories will therefore give clarity as to when a particular assurance technique needs to be applied.

Data supply chains The previous models relate to typical system supply and operate perspectives but there are also other data supply chains where a number of organizations engage in the procurement and use of safety-related data. These processes may include the development and operational lifecycles but a different model is required to fully represent the wider processes that are being employed. The diagram in Figure 7.4 shows such a model representing a data acquisition lifecycle.

This model represents the interactions between three key organizations:

[1] The diagram is based on International Business Machines Corporation (IBM)'s Rational Unified Process, an iterative software development process framework. The original diagram is in the public domain.

[2] ITIL is a registered Trade Mark of AXELOS Limited. All rights reserved.

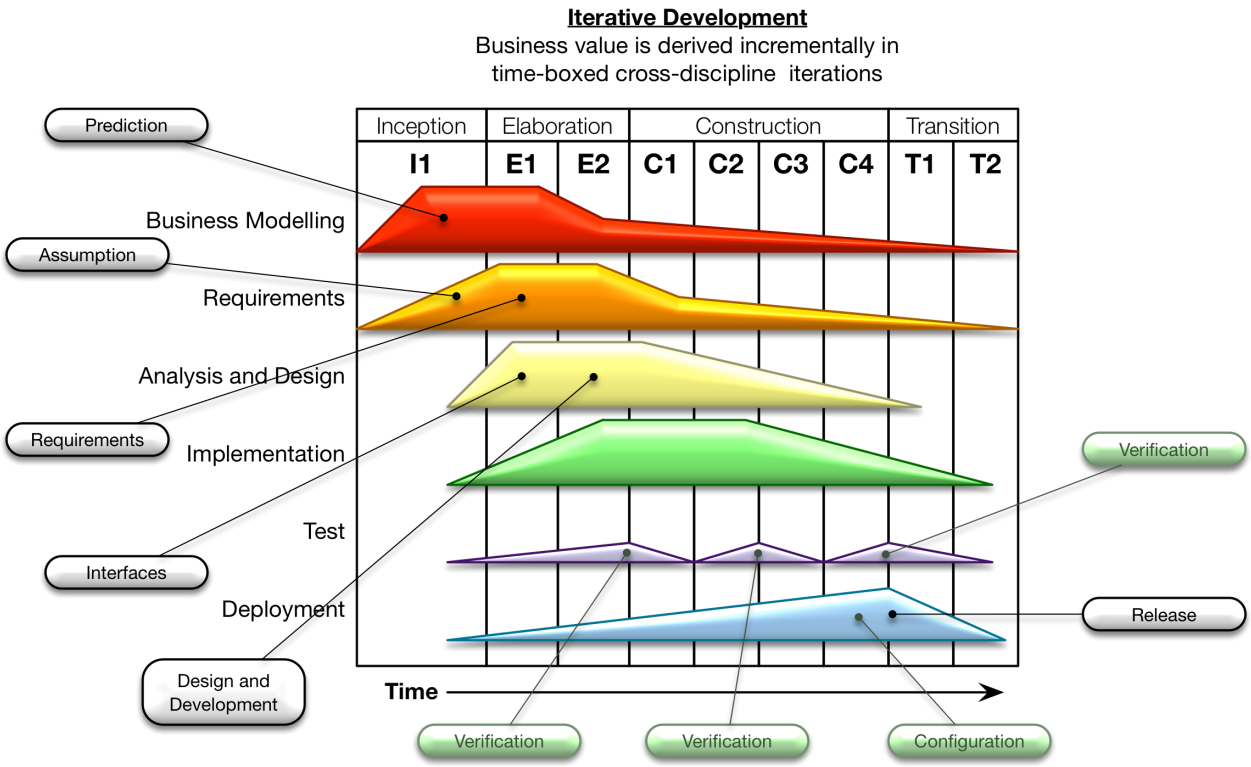

Figure 7.2: Development lifecycle

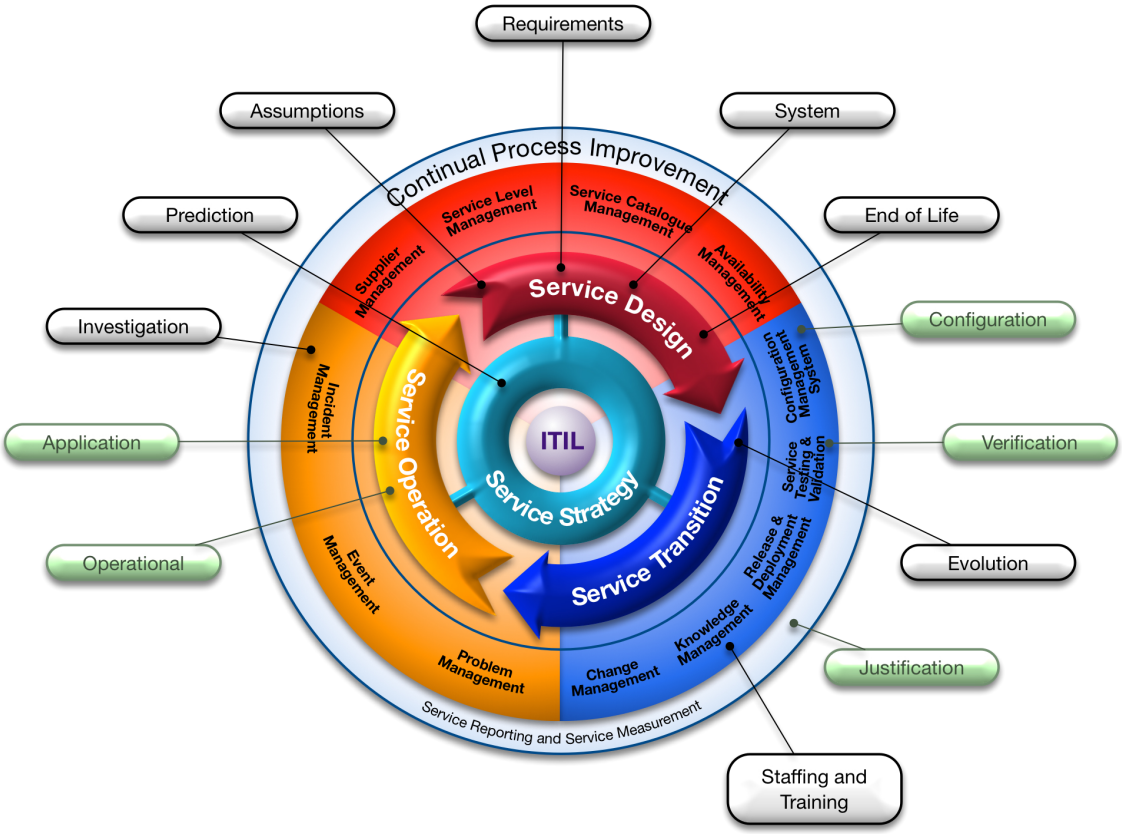

Figure 7.3: Operational lifecycle

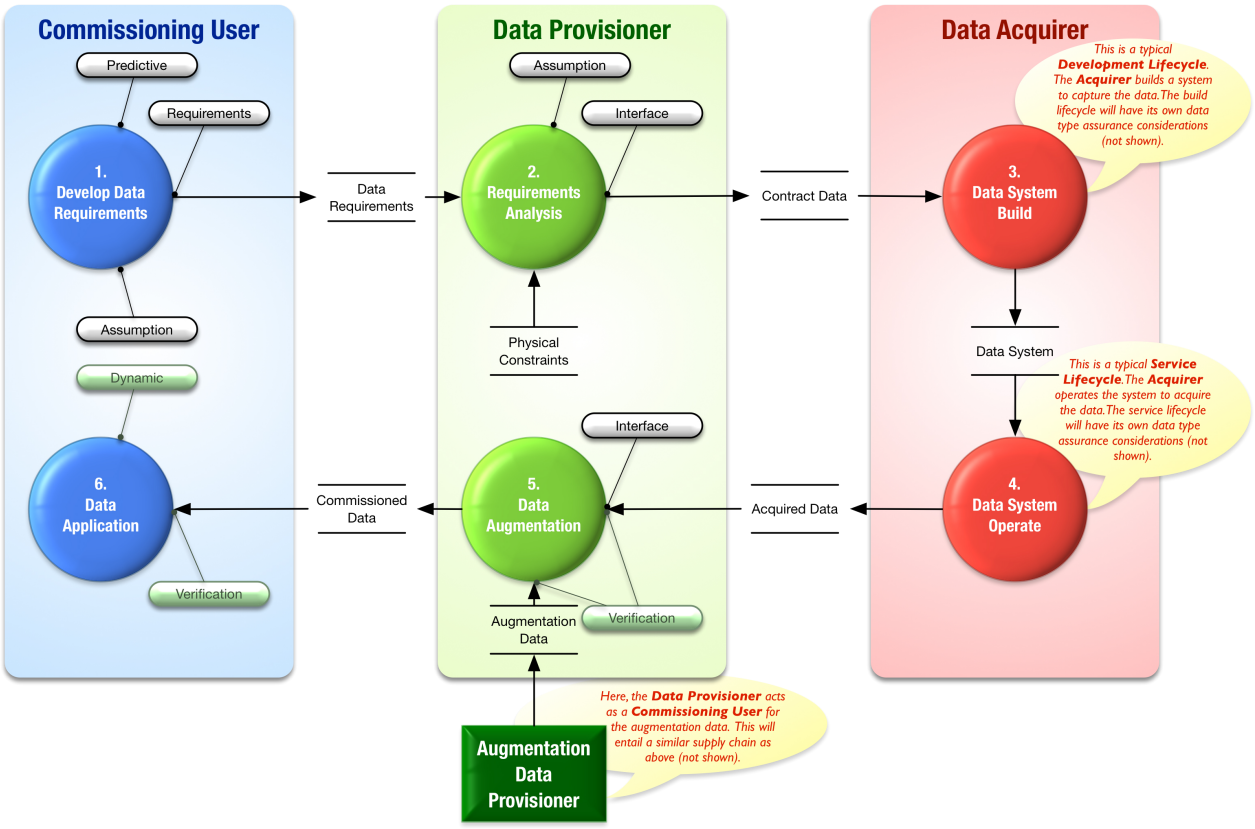

Figure 7.4: Data supply chain

- The commissioning user: the organization that has the need for the data;
- The data provisioner: the organization that will fulfil that need for data; and
- The data acquirer: the organization employed by the Data Provisioner to carry out physical collection of data.

Note that these may be three separate organizations, or they may be separate business units within the same, larger, organization.

In this supply chain, the commissioning user is a consumer of the data and the data acquirer is a producer of data. The data provisioner acts both as a consumer (from the data acquirer) and producer (to the commissioning user) of data. Similarly, an organization that augments datasets is both a consumer and producer of data in the supply chain.

The commissioning user requirement analysis is the key process step where the commissioning user's expectations for data are agreed with the data provisioner. The requirements may be adjusted because of physical constraints (e.g., loss of precision because of physical measuring device constraints) and may include additional requirements to augment the captured data with additional information (e.g., airport codes added to a measurement of a given runway length).

The data provisioner may employ a data acquirer to capture the data (e.g., to carry out a physical survey of a site). The acquisition phase may itself require a specialised system to be built to perform the capture and data refinement to meet the data provisioner's specifications. Such systems will then themselves be subject to the development lifecycle model considerations discussed above. Likewise, the data augmentation phase may require further system development processes or indeed, could trigger an instance of the model

again as the data provisioner acting as a Commissioning User.

Acquired and augmented data is then fed into the operational system that has been built for providing the service of generating the commissioned data. This system in its service provision role would then typically follow the operational lifecycle process model discussed earlier.

7.2.1 Tool Assurance

Tools in this context are considered anything that automates all or part of a process, for example, data creation or data transformation. Test tools are also included (i.e., the term is not limited to parts of an operational system).

Tools can impact data safety in different ways, depending on both their function and how they are to be used. For tools to be considered fit for purpose it is necessary to show that the tool meets its requirements in the context in which it is to be used. The activity to ensure a tool is fit for purpose is usually called "tool qualification".

The first step is to define the purpose for which the tool is required to be fit. Once that is done, and the tool's requirements are specified, there are three main strategies available for qualification:

- Use evidence of a previous certification of the tool by a trusted third party (unlikely to be available in most industry sectors);
- Base tool qualification on the practices used when designing and developing the tool (only practicable for tools developed within the organization); and
- Use one of the available industry-specific guidance documents that admit commercial off-the-shelf (COTS) solutions, e.g., EUROCAE Document ED-215 (RTCA/DO-330) [**?**].

Further details on tool qualification are presented in Appendix 8.

7.2.2 Test Data

The generation of suitable test data is critical to verification of a safety system. The test data must include both representative "normal" values based on intended usage and also values which push at, and beyond, normal use to provoke hazards that the system might produce. This latter type of test data is particularly hard to generate; generally it must be credible, yet it must stress the system to react in a way that the preservation of safety properties can be assessed.

In general, all the properties of the test data should be considered and an assessment made as to whether breaking a property (e.g., introducing corrupt or late data) would cause a problem to the system. If it does, then specific test data should be produced to facilitate testing of this potential problem.

Some suggestions for test data for safety-related systems are:

- Use of values on or around boundaries;
- Use of extreme values, way beyond what could be reasonably expected;
- Use of typical "everyday" values / sets;

- Some realistic but unexpected values;
- Try combinations of data values or data items that are problematic together (e.g., inconsistent);
- If possible, use some values known to have caused problems in the past;
- Where appropriate, use values related to timing, rollover or date boundaries;
- Where possible, use white box values (i.e., those derived from an understanding of the system);
- Use a set of values with drift or bias over time;
- Use datasets with particular statistical properties (e.g., distribution, patterns etc.);
- Use data which has incorrect formatting, ordering, or out of sequence, etc.; and
- Try data which has repeated sets of values or pseudo-random characteristics.

Typically very complex test data is derived from recorded live feeds of real data flows. While this data can be extremely useful for regression purposes, it should be recognised that it is unlikely to contain many outlying or boundary data items. Therefore it may need to be modified to test any hazardous situations; this modification can be difficult and may require sophisticated tools to both ensure correct properties and injection of the intended faults (for instance to introduce a statistical bias to the data).

Simulator / emulator derived values can be useful, but again the issue is how realistic the values are: often the accuracy, resolution or timing of simulated values may be different to real data.

Coverage with test data is something to consider. Sometimes the same dataset is used for multiple test scenarios, when in fact it is not stressing all of them to the same degree. Test data coverage can be collected over requirements, code or design, but it is important not to forget hazards: coverage of the hazards and mitigations identified in the hazard log is a key aim.

In general some measure of the quality and suitability of the test data can be useful. This could be based on statistical properties, coverage of hazards or coverage of requirements.

Test data must show continued relevance, through systems evolution and over time. It is good practice to build up extensive regression suites containing coverage of all detected problems to date.

7.2.3 Interfaces with Existing Assessments

7.2.3.1Data and Software

Although most people feel they have an intuitive understanding of the difference between software and data, upon closer examination the boundary is not always as clear as it may first appear.

Consider, for example, Java bytecode, which is operated on by a Java virtual machine. From one perspective, it could be argued that the Java bytecode is simply data. By extension, it could also be argued that the Java source code is also just data. This type of argument can be extended to suggest that any software can, at least from one viewpoint, be considered as data. Conversely, think about the data used in a 3D printer, perhaps to produce a part for an aircraft. This data could be viewed as a program for the printer; that is, it could potentially be viewed as software. This type of argument can easily be extended across a range of situations, especially those relating to configuration data.

While they are interesting, and potentially important, these philosophical considerations should not detract from the practical issue: there are some aspects of data (using the term in a generic sense) that are often not explicitly addressed in standards. These are a consequence of features that are more readily apparent in data than in software. Examples include:

- It is easy for data to be reused in a range of contexts and despite appearances it is not trivial to translate an assurance argument that the data is fit for purpose from one context to another.
- It is not always clear who owns or is responsible for data, especially when data is shared and processed amongst a collection of disparate systems.
- Data often has a lifetime, that is a time after which it is no longer valid. This may be a strict cut off, or a more gradual degradation in the utility (or applicability) of the data.
- There is often a default value for data. While this can make systems easier to use and hence more productive it can be difficult to identify a default value that is appropriate for all circumstances.
- It can be easy to change data. In some circumstances this can give rise to a temptation to make uncontrolled and potentially untested changes. It can also allow data to be fraudulently changed after an accident.

In summary, data and software are closely related and, as such, need to be considered together in system engineering activities, including system safety analyses. However, data and software emphasise different facets of risk and they are susceptible to different mitigation approaches; this means there is also a need to adopt a data-focused perspective. It also means that software assurance levels cannot be mapped directly to DSALs.

7.2.3.2Data Safety and Security

When generating high-level processes and techniques to manage the risks posed by data, it is worthwhile understanding the difference between the safety risks posed by accidental failure to preserve Data Properties and the security risks posed by actors maliciously undermining the properties of data.

The relationship between safety and security, as engineering concepts, can be summarised by their relationships to cultural, developmental and aspirational properties of systems development.

Culturally, embedding both safety and security into an organization is seen as a key strategic goal for creating systems that are both safe and secure. Developmentally, safety and security are quality factors, generating transverse requirements that impact the entire system. Most importantly, at the aspirational level, both safety and security have the common goal of preventing harm from accidental and malicious interventions respectively.

For an organization aiming to create systems that are both safe and secure, these connections can be both a benefit and a burden. The shared goal of preventing harm means that both quality factors seek to identify routes to harm through analysis of the system being developed. This can result in shared processes and tools, which in turn can save time and money during systems development. However, safety and security interact in a more volatile way at the functional level. Security failings can undermine the safety case for a system and, conversely, safety requirements can prevent the implementation of standard security solutions. For example, the German government published a report in 2014 into a fire at a steel works caused by a cyber attack that resulted in the control system being placed into an unsafe state and the

safety system being unable to intervene (Section 3.3.1 of [**?**] - in German). In addition, "fail-safe" states can often leave a system with exposed security vulnerabilities.

These links between safety and security infer that there are connections between the sub-categories of data safety and information security: both attempt to take a data-centric view of the system of interest in order to improve the associated quality factor; and both attempt to prevent harm through the preservation of the properties of data within that system.

In the security domain, the three key properties of data considered are confidentiality, integrity, and availability. Confidentiality, the failure of which is termed "information disclosure" in the Microsoft Security Model, [**?**] is typically not a safety concern as, without malicious intent, information sharing is not inherently unsafe. However, when considering systems where confidentiality is an important property, the interaction between data safety and security cannot be trivially resolved. For example, accidental disclosure of information can form part of a causal chain which leads to harm from a malicious actor.

Data integrity is a critical property for both domains. The Microsoft Security Model describes malicious removal of the property of integrity as "tampering". Whether by accident or through malicious intent, the potential harm from loss of data integrity can be disastrous to a safety-critical system, from the values of drug dosages to control system parameters.

Data availability is also important to both domains. Loss of availability, or "denial of service" in the Microsoft Security Model, is another property that can be lost accidentally or through malicious intervention. Loss of availability prevents systems from functioning properly and can result in undefined behaviour if not mitigated by design.

Further guidance on the integration of safety and security can be found in a code of practice published by the IET [**?**]. The Code of Practice is written for engineers and engineering management to support their understanding of the issues involved in ensuring that the safety responsibilities of an organization are addressed, in the presence of a threat of cyber attack.

This page is intentionally blank

Chapter 8

Tool Confidence and Tool Qualification of Data Processing Tools (Informative)

> *Don't fix the blame, fix the problem*
> ***Keith Pennington***

This page is intentionally blank

8.1 Introduction

A tool is a software that is required to build the product, but itself is not part of the product. For example, a compiler or a test tool.

This page is intentionally blank

8.2 What the Standards say

Main safety standards, like IEC 61508, ISO 26262, DO-178, EN-50128 require an assessment of "tool confidence" or the application of "tool qualification" to ensure that a lack of integrity in the tools does not impact the safety of the product.

These safety standards use an approach which may be broadly defined as a three step process:

1. Analyse the risks of the tool, so called "tool classification"
2. Make the tools safe, based on the tool classification, so called "tool qualification", usually by testing the tool. Note if a tool is classified as non-critical, for example, Tool Confidence Level 1 in ISO 26262, it does not need to be qualified.
3. Use the tools as classified and qualified, such as by following a so called "tool safety manual"

The approach to classification differs in different standards and also the proposed qualification methods differ slightly, but all share the same principle: use the tool carefully (safety manual) or qualify that the tool is doing what it should do. More details can be found in the safety standards.

This page is intentionally blank

8.3 Data tools

For safety of data the same should be applied to all used tools – they should be classified, eventually qualified and used according to the safety manuals.

A specific approach is not proposed here for data tools, but a recommendation to use the standards with which a product has to be compliant. For example, if data is to be used in the Automotive sector, then consideration should be given to chapters 8–11 "Confidence in the use of software tools" of ISO 26262.

The following generic recommendations are intended to manage tool confidence according to the standards applicable to the domain:

- The tool scope is wider: not only do tools for building the product have to be considered, but also the tools that create the data which impact the system, such as the maps of a self-driving car, or the training data which is used to train AI systems.
- Consider all tools that are working with the data:
  - Creation tools that create the data, for example, tools processing sensor data
  - Data storing tools, that store the data, perhaps into a database
  - Data manipulation tools, that transform data, such as by translation to a different geodetic system, merging of different data sources or the creation of summaries
  - Data visualization tools that show the data – viewers that display the results
- Consider all outputs of the tools. For example an AI training tool usually creates two outputs:
  1. The trained network and
  2. The achieved accuracy.

  Each can be impacted by tool errors and therefore both features of the tool have to be classified/qualified and used safely.
- When working with generic tools like "Excel", "SQL", or "Python", consider every new application as a new use of the tool – in other words, classification of a tool can never be truly generic, but must be reviewed for each specific use. An example of failing to carry out this reassessment is the Covid Summary Table (see **??**), which was written / created in Excel using some tables/schemes into which the data was imported and analysed.
- The classification and qualification do not depend on the DSAL, however the DSAL should be considered when testing the tool or when classifying the risks. For example if a tool is to be qualified by testing, it should be done with the same rigour that is required for the DSAL.
- When looking to the risks of tools, it is not sufficient to consider only the functional risks like "wrong data created" (such as the potential that 1+1 could be computed to be 0 instead of 2). In addition, the data-specific risks specified in this document (Section **??** Ways that Data Can Cause Problems) should be considered for impact from the tools. Data properties that can be impacted from tools should also be considered. Therefore the following aspects of tools to data should be analysed during tool classification:
  - Each issue listed in subsection **??**
  - Each Property listed in **??**

For all above aspects the impact of the tool has to be considered, for example “Interpretation”, the risk of “Wrong Interpretation” can be expressed in the question: “Can the tool change the data, such that it will be wrongly interpreted?”. If this is not possible / makes no sense, then the error “Wrong Interpretation”, can be classified as “no impact”. Otherwise the risk is valid and has either to be mitigated by a safety guideline in the safety manual, or it should be excluded by systematic qualification tests which show that the error “Wrong Interpretation” cannot occur.

Acronyms, Definitions and Glossary (Discursive)

> *The plural of anecdote is not data.*
> ***Mark Berkoff***

This page is intentionally blank

8.4 Acronyms

This document is incomplete. The external file associated with the glossary 'acronym' (which should be called `Vol1.acr`) hasn't been created.

Check the contents of the file `Vol1.acn`. If it's empty, that means you haven't indexed any of your entries in this glossary (using commands like `\gls` or `\glsadd`) so this list can't be generated. If the file isn't empty, the document build process hasn't been completed.

Try one of the following:

- Add `automake` to your package option list when you load `glossaries-extra.sty`. For example:

  ```
  \usepackage[automake]{glossaries-extra}
  ```

- Run the external (Lua) application:

  ```
  makeglossaries-lite.lua "Vol1"
  ```

- Run the external (Perl) application:

  ```
  makeglossaries "Vol1"
  ```

Then rerun LaTeX on this document.

This message will be removed once the problem has been fixed.

This page is intentionally blank

8.5 Definitions and Glossary

This document is incomplete. The external file associated with the glossary 'main' (which should be called `Vol1.gls`) hasn't been created.

Check the contents of the file `Vol1.glo`. If it's empty, that means you haven't indexed any of your entries in this glossary (using commands like `\gls` or `\glsadd`) so this list can't be generated. If the file isn't empty, the document build process hasn't been completed.

If you don't want this glossary, add `nomain` to your package option list when you load `glossaries-extra.sty`. For example:

```
\usepackage[nomain]{glossaries-extra}
```

Try one of the following:

- Add `automake` to your package option list when you load `glossaries-extra.sty`. For example:

  ```
  \usepackage[automake]{glossaries-extra}
  ```

- Run the external (Lua) application:

  ```
  makeglossaries-lite.lua "Vol1"
  ```

- Run the external (Perl) application:

  ```
  makeglossaries "Vol1"
  ```

Then rerun LaTeX on this document.

This message will be removed once the problem has been fixed.

The normative list of definitions is at chapter 3. Normative definitions have been repeated here for convenience.

This page is intentionally blank

References (Discursive)

> *Data opportunities multiply as the data is transformed.*
> ***Sun Tzu misquoted***

This page is intentionally blank

Acknowledgements (Discursive)

> *Our ability to do great things with data will make a real difference in every aspect of our lives.*
> ***Jennifer Pahlka***

The document contributors would like to thank:

- The SCSC.
- The SCSC Covid-19 Working Group for providing some of the data used in the Covid-19 Appendix.
- Brian Jepson of the SCSC for web hosting support and technical help with the SCSC web site.
- Tim Rowe for editing this edition.
- Paul Hampton and Mark Templeton for managing the publication processes.
- Nick Hales, Mike Parsons, Tim Rowe, Alan Simpson and Mark Templeton for developing the additional text for this edition.
- Martin Atkins and Divya Atkins for driving the development of tooling and promoting data safety.
- Mike Parsons for chairing the Working Group meetings.
- All those who have taken minutes at Working Group meetings.
- All the organisations that have hosted Working Group meetings.
- All the organisations that have provided support to the document's contributors.
- Those that have been unable to attend meetings but have made supporting contributions.

This page is intentionally blank

Contributors (Discursive)

> *Without data, you're just another person with an opinion.*
> ***W. Edwards Deming***

This document has had the benefit of contributions from a large number of people, who work for a variety of organisations, which collectively span a range of different sectors. Note that contributions have been made on an individual basis and, in particular, the inclusion of an organisation in the following list does **not** necessarily mean that organisation agrees with the entire contents of the document.

Updates to the most recent version of the document were written by:

- Divya Atkins, Mission Critical Applications
- Martin Atkins, Mission Critical Applications
- Paul Hampton, CGI IT UK Ltd
- Mike Parsons, Ebeni and SCSC
- Tim Rowe, TGR Safety Management Ltd

In addition to the above, contributors to earlier versions upon which this document is based include the following (the organisations listed were correct at the time of their contribution) :

- Mike Ainsworth, Ricardo
- Rob Ashmore, Dstl
- Michael Aspaturian, EDF Energy
- Janette Baldwin, Thales UK
- Dave Banham, Blackberry QNX
- Ian Bingham
- John Bragg, MBDA UK Ltd
- Jennifer Brain, Wood plc
- Eric Bridgstock
- Simon Brown, QinetiQ
- Dermot Martin Burke, BAE Systems

- Dale Callicott, DKCSC Ltd
- John Carter, General Dynamics
- Martyn Clarke, SCSS Ltd
- Steve Clugston, TSC
- Robin Cook, Thales
- Davin Crowley-Sweet, Highways England
- Dijesh Das, AMEC / BAE Systems
- Duncan Dowling, DARD
- Andrew Eaton
- Ashraf El-Shanawany, CRA Risk Analysis
- Paul Ensor, Boeing
- Alastair Faulkner, Abbeymeade
- Ken Frazer, KAF
- Richard Garrett, SQEP
- Paulo Giuliani
- Ian Glazebrook, Atkins
- Rob Green, NATS
- Nick Hales
- Louise Harney, Leonardo
- Ali Hessami, Vega Systems
- David Higgins
- Gordon Hurwitz, Thales
- Pete Hutchison, RPS
- Gavin Jones
- Amira Kawar, Kawar Engineering Consultancy Ltd
- Tim Kelly
- Andrew Kent
- Brent Kimberley, Durham, Canada
- Julian Lockett, Frazer-Nash Consultancy Ltd
- David Lund, David Lund Consultants
- Dave Lunn, Thales UK
- Nasser Al Malki, University of York

- Victor Malysz, Rolls-Royce
- Jim Mateer, SQEP
- John McDermid, University of York
- Paul McKernan, Dstl
- Thor Myklebust, Sintef
- Mark Nicholson, University of York
- Yvonne Oakshott
- Robert Oates
- David Perrin, Virtual PV
- Ashley Price, Raytheon UK
- Andrew Rankine
- Felix Redmill, SCSC
- Sam Robinson, EDF Energy
- Mark Simmonite, Highways England
- Alan Simpson, Ebeni
- Oscar Slotosch, Validas AG
- Dave Smith, Frazer-Nash Consultancy Ltd
- Peter Smith, Highways England
- John Spriggs, NATS
- Carolyn Stockton, BAE Systems
- Mark Templeton, Arcade Experts Ltd
- Andy Williams
- Lesley Winsborrow
- Fan Ye, ESC

DATA IS HERE. DATA IS GROWING. DATA IS CAUSING HARM.

This book has been developed by the Safety-Critical Systems Club Data Safety Initiative Working Group (DSIWG) to provide guidance on how data, as distinct from hardware and software can be managed in a safety-related context.

"If you torture the data long enough, they confess – even to crimes that were never committed."

Nihat Bülent Gültekin

~

This is the seventh minor update since version 3.0. Paragraph numbering within the body of the document remains aligned with that major release. Thus users of any previous 3.x release of the guidance document will find migration to this edition takes little effort.

SCSC

For Everyone Working in System Safety