IT Project Guidance

On Data and Actionable Information

Version:

0.1

## Purpose

This document aims to help clarify the difference between *data* and *information*, and why that distinction matters. It introduces key concepts from information management that apply not just to large organisations, but to any system that collects, stores, or uses data. The goal is to support more thoughtful system design, better decision making, and greater alignment between how we manage data and what we actually need from it.

## Synopsis

Many systems treat data as something to be collected and stored indefinitely, assuming future value without clear purpose. But unmanaged data quickly becomes a liability—costly to store, hard to search, and risky to secure. This paper outlines a simple framework for understanding how data becomes information, and how that information can support insight, action, and audit. It also examines the real costs of keeping everything, and offers guidance for purposeful, principle-led data retention.

## Contents

[Purpose 1](#_Toc195791704)

[Synopsis 1](#_Toc195791705)

[Contents 2](#_Toc195791706)

[Introduction 3](#_Toc195791707)

[Background of this work 3](#_Toc195791708)

[Context 3](#_Toc195791709)

[Background of issue 3](#_Toc195791710)

[Problem Statement 3](#_Toc195791711)

[H3 4](#_Toc195791712)

[H4 4](#_Toc195791713)

[Appendices 5](#_Toc195791714)

[Appendix A - Document Information 5](#_Toc195791715)

[Versions 5](#_Toc195791716)

[Images 5](#_Toc195791717)

[Tables 5](#_Toc195791718)

[References 5](#_Toc195791719)

[Review Distribution 5](#_Toc195791720)

[Audience 5](#_Toc195791721)

[Structure 5](#_Toc195791722)

[Diagrams 6](#_Toc195791723)

[Acronyms 6](#_Toc195791724)

[Terms 6](#_Toc195791725)

# Purpose And Audience

Background  
The principles of information management are long established and well supported by international standards—such as ISO/IEC 11179 for metadata, ISO 8000 for data quality, ISO/IEC 38505 for governance, and ISO/IEC 25012 for the qualities of system data. These frameworks aren’t just for large enterprises. They offer practical guidance that applies to everyday system design, but are often overlooked or never studied at all.

As a result, we see the same patterns repeated. Data is handled using only the most basic operations—create, read, update, delete (CRUD)—with no attention to state transitions, reversibility, or accountability. Physical deletion is treated as the only way to remove something, even though it’s irreversible and often inappropriate. Instead of supporting proper lifecycle management, designers rely on stopgaps: modal prompts asking users “Are you sure?”, as if one last warning could replace a robust process. But mistakes still happen. Records are lost, actions can’t be undone, and systems offer no way to trace, restore, or reason about what was removed.

These problems don’t stem from malice or neglect—they come from missing concepts. Without recognising that deletion is a physical act, and that removal, retirement, or archiving are logical states that should be modelled and reversible, we fall back on weak workarounds. That’s not a technology problem. It’s a thinking problem.

# Two Continua: Understanding Value and Validity

To make sense of how data relates to information, and how both relate to decisions and action, it helps to view them not as fixed categories, but as points along two complementary continuums. Each reflects a different aspect of how we interpret, trust, and act on what is presented to us.

# The Meaning Continuum: From Signal to Action

The first continuum is about *meaning*. It begins with undifferentiated input—raw signal, which may include a mix of relevant and irrelevant noise. From that, systems or people identify patterns and extract data. Once data is contextualised, it becomes information.

Information, when analysed or reflected on, can produce insight.

Insight, when applied to a specific context, informs decision.

And decision, if executed, leads to action.

This continuum is especially useful when thinking about system capability. It shows that data is not an endpoint. It is a starting point—useful only if it can be transformed into something that improves outcomes, which only happens if Action is taken.

Signal → Data → Information → Insight → Decision → Action

The implication is clear: not all data deserves to be kept. If it cannot lead to insight that supports action—now or reasonably in the future—then retaining it adds burden, not value.

# The Trust Continuum: From Utterance to Law

The second continuum concerns *trust and truth*. It traces the path from casual or uncertain content to something that is verifiable, consistent, and authoritative. It starts with an utterance—a spoken or written claim. Some utterances become statements, formalised and recorded. Some statements can be verified as true. Those which are reliably true in repeated contexts are called facts. Over time, well-founded facts, consistently observed, may be codified as laws.

This continuum is important when reasoning about quality and governance. Most system records begin as utterances—entered by users, received from external sources, or derived from fallible sensors. They are not immediately facts, and treating them as such invites error. Information governance requires recognising what is known, what is believed, and what has been verified.

# How the Continua Work Together

These two axes—meaning and trust—overlap in practice. A piece of data may be highly meaningful but unverified (e.g. a strong insight drawn from a shaky source), or it may be factual but not currently useful (e.g. accurate weather readings from last month). Good systems support both: they trace provenance and improve confidence, while also enabling the transformation of data into value.

Together, these continuums help dispel the myth that more data is always better. They support a more mature view: that systems should focus not on hoarding, but on refining. What matters is not how much data is kept, but how well it supports understanding and decision-making—now, or at defined points in the future.

Appendices

Appendix A - Document Information

Authors & Collaborators

* Sky Sigal, Solution Architect

### Versions

* 1. Initial Draft

### Images

[Figure 1: TODO Image 2](#_Toc144995112)

### Tables

[Table 1: TODO Table 3](#_Toc145048484)

[Table 2: TODO Table 2 3](#_Toc145048485)

### References

**There are no sources in the current document.**

### Review Distribution

The document was distributed for review as below:

|  |  |
| --- | --- |
| Identity | Notes |
|  |  |
|  |  |
|  |  |

### Audience

The document is technical in nature, but parts are expected to be read and/or validated by a non-technical audience.

### Structure

Where possible, the document structure is guided by either ISO-\* standards or best practice.

### Diagrams

Diagrams are developed for a wide audience. Unless specifically for a technical audience, where the use of industry standard diagram types (ArchiMate, UML, C4), is appropriate, diagrams are developed as simple “box & line” monochrome diagrams.

### Acronyms

API

: [Application Programming Interface](#Term_ApplicationProgrammingInterface).

DDD

: Domain Driven Design

GUI

: [Graphical User Interface](#Term_ApplicationProgrammingInterface). A form of [UI](#Acronym_UI).

ICT

: acronym for Information & Communication Technology, the domain of defining Information elements and using technology to automate their communication between entities. [IT](#Acronym_IT) is a subset of ICT.

IT

: acronym for Information, using Technology to automate and facilitate its management.

UI

: User Interface. Contrast with [API](#Acronym_API).

### Terms

Refer to the project’s Glossary.

Application Programming Interface

: an Interface provided for other systems to invoke (as opposed to User Interfaces).

Capability

: a capability is what an organisation or system must be able to achieve to meet its goals. Each capability belongs to a domain and is realised through one or more functions that, together, deliver the intended outcome within that area of concern.

Domain

: a domain is a defined area of knowledge, responsibility, or activity within an organisation or system. It groups related capabilities, entities, and functions that collectively serve a common purpose. Each capability belongs to a domain, and each function operates within one.

Entity

: an entity is a core object of interest within a domain, usually representing a person, place, thing, or event that holds information and can change over time, such as a Student, School, or Enrolment.

Function

: a function is a specific task or operation performed by a system, process, or person. Functions work together to enable a capability to be carried out. Each function operates within a domain and supports the delivery of one or more capabilities.

Person

: a physical person, who has one or more Personas. Not necessarily a system User.

Persona

: a facet that a Person presents to a Group of some kind.

Quality

: a quality is a measurable or observable attribute of a system or outcome that indicates how well it meets expectations. Examples include reliability, usability, and performance. Refer to the ISO-25000 SQuaRE series of standards.

User

: a human user of a system via its UIs.

User Interface

: a system interface intended for use by system users. Most computer system UIs are Graphics User Interfaces ([GUI](#Acronym_GUI)) or Text/Console User Interfaces (TUI).