Aristotle Data Model

Matthew Lawler

[Lawlermj1@gmail.com](mailto:Lawlermj1@gmail.com)

[Introduction 3](#_Toc95661687)

[Licence 3](#_Toc95661688)

[Warranty 3](#_Toc95661689)

[Purpose 3](#_Toc95661690)

[Audience 3](#_Toc95661691)

[Approach 3](#_Toc95661692)

[By 3](#_Toc95661693)

[Acronyms 3](#_Toc95661694)

[References 4](#_Toc95661695)

[Metadata Repository (MDR) 5](#_Toc95661696)

[How does an MDR provide economic benefits? 5](#_Toc95661697)

[Audit: Systems/database support or practicing what is preached 6](#_Toc95661698)

[Audit: Open Government 7](#_Toc95661699)

[Aristotle Metadata Repository (MDR) 8](#_Toc95661700)

[Grammatical Quality 8](#_Toc95661701)

[Additional Invariants and checks 9](#_Toc95661702)

[Building up the Aristotle MDR 10](#_Toc95661703)

[Other ideas 10](#_Toc95661704)

[Aristotle Metadata Views 12](#_Toc95661705)

[Aristotle Any Item Metadata 12](#_Toc95661706)

[API 13](#_Toc95661707)

[MDR Definitions 13](#_Toc95661708)

# Introduction

## Licence

This document is released under the Creative Commons Zero licence or CC0.

## Warranty

The author does not make any warranty, express or implied, that any statements in this document are free of error, or are consistent with a particular standard of merchantability, or they will meet the requirements for any particular application or environment. They should not be relied on for solving a problem whose incorrect solution could result in injury or loss of property. If you do use this material in such a manner, it is at your own risk. The author disclaims all liability for direct or consequential damage resulting from its use.

## Purpose

This document describes the data model underlying the Aristotle Metadata Registry (MDR).

## Audience

The primary audience for this document are metadata designers. This applies especially to designers who need to integrate Aristotle with other metadata tools. The reader needs to understand metadata concepts.

## Approach

This document presents an analysis of an instance of the Aristotle MDR. This analysis was based on profiling the extracted JSONs from the cloud. This is not a comprehensive document. As each Aristotle instance is different, so the population and usage of the metadata will also be different.

## By

This was written by Matthew Lawler.

## Acronyms

This is a list of acronyms used in the document.

|  |  |  |  |
| --- | --- | --- | --- |
| Acronym | Expansion | AKA | By |
| AKA | Also Known As |  | English |
| ANSI | American National Standards Institute |  | ANSI |
| DB | Database |  | ANSI |
| FK | Foreign Key |  | ANSI |
| GUID | Globally Unique Identifier | UUID | Microsoft |
| ISO | International Organization for Standardization |  | ISO |
| JSON | JavaScript Object Notation |  | ISO |
| MDR | Metadata Registry |  | ISO |
| OWL | Web Ontology Language |  | W3C |
| PK | Primary Key |  | ANSI |
| RDF | Resource Description Framework |  | W3C |
| RO | Read Only |  | OMG |
| RW | Read Write |  | OMG |
| UUID | Universally Unique Identifier | GUID | ISO |
| W3C | World Wide Web Consortium |  | W3C |

## References

| By | For | Path | Full |
| --- | --- | --- | --- |
| Aristotle | API | https://aristotle.cloud/api/v4/ |  |
| Aristotle | Cloud | https://www.aristotlemetadata.com/ |  |
| Aristotle | Source | https://github.com/aristotle-mdr/aristotle-metadata-registry |  |
| Aristotle | wiki | https://en.wikipedia.org/wiki/Aristotle\_Metadata\_Registry |  |
| IAASIST | Conference | https://iassistdata.org/ |  |
| ISO | 11179 | https://www.iso.org/obp/ui/#iso:std:iso-iec:11179:-1:ed-3:v1:en |  |
| W3 | Standard | https://www.w3.org/TR/owl-ref/#sameAs-def | OWL Matching standard |
| Wiki | Standard | https://www.wikiwand.com/en/Data\_element\_definition | Wikiwand |

## API

Haskell was used to access the API. All code is here: <https://github.com/lawlermj1/Aristotle-JSON>

# Metadata Repository (MDR)

The following are some thoughts on Metadata Repository, and on the Aristotle MDR is particular. These are not meant to be comprehensive but provide a starting for further investigation.

## How does an MDR provide economic benefits?

In the Economics of Information, the main benefit of an informational good is to reduce uncertainty. If the informational good here, the MDR, cannot reduce uncertainty due to incompleteness, unreliability, etc, then the beneficiary, such as a project manager, will make the economically rational choice of ignoring it, and using some alternative basis to reduce uncertainty, such as mandating choices, etc. It is easy to create a mess in an MDR, especially when there are few or no automated QA checks. Many of the following checks can be automated, which would lead to an increase in data quality, and enhance the economic value.

A Metadata Repository can help with a number of business goals.

These can be

1. Definition source

2. Initial Requirements

3. Requirements Traceability from Definitions to data sources

4. Auditing for systems/database support

5. Auditing for openness

The first 3 won't be examined as they are self-evident use cases. The Auditing use cases will be looked at below.

## Audit: Systems/database support or practicing what is preached

This involves collecting 2 corpus. One based on all public documents, such as any Acts, etc. The other would be based on available systems and database metadata. See diagram.

Diagram

Description automatically generated

1. Documents Corpus created from enabling Acts and Public Documents

2. Databases Corpus created from Databases and systems

3. Supported = Overlap between Documents and Databases Corpus

4. Unsupported = Words exists in Documents, but does not exist in Databases

5. Additional = Words exists in Databases, but not in Documents

If supported % is high, then there is a good fit between documents and databases. That is, the database would be justified as it supports the relevant Act.

If Unsupported % is high, then there is a database or systems capability gap, which could be used to justify additional projects.

If Additional % is high, then these words are either hidden or represent too much systems capability.

If hidden, then this can be made public, or reduce unneeded capability. It could also lead to the discovery of legislative or regularity gaps. Then the information could be added to the legislation, or the capability turned off on efficiency grounds.

Obviously, this is not a detailed capability assessment. It is just a check to see if the language used in the systems is consistent with the primary documents. Further, more detailed analysis is required.

## Audit: Open Government

A common Public Service anti-pattern is to impose unreasonable levels of security. All words used in publicly available documents would be tagged at UNCLASSIFIED. Any attempt to lift a PSPF tag to a stricter rating such as CLASSIFED should be preventable. Actual database names might have a higher PSPF, but these would be in distribution object. It would also enable the opening up of the API to the public, contributing open government.

# Aristotle Metadata Repository (MDR)

I have used the Aristotle API to extract and parse all available Aristotle JSON metadata objects into data structures. I have then used these data structures to define draft, exploratory checks to measure some aspects of the Aristotle metadata data quality. Altogether there are twenty-two parsed JSON files.

Aristotle has three core objects: Object Class, Property and Data Element Concept. However, these three have quite complicated definitions that are hard for anybody to understand. These are a case study of making definitions needlessly complicated. It is a case of accidental complexity, which leads to spaghetti metadata. A key insight is that these core objects are really Nouns, Adjectives and Phrases. That is, it is as simple as basic grammar. In effect, SA is building a Corpus of its words. To restate, an Object Class is a Noun or group of Nouns, a property is an Adjective, Adverb or Verb, and Data Element Concept is a Phrase formed by its parent Noun and Adjective. The word Adnominal means either Adjective, Adverb or Verb. That is, an Adnominal is a modifier to convert a Noun into a Noun Phrase.

As with other parts of the ISO 11179 standard, the grammatical specification is incomplete. However, even with a limited number of Parts of Speech (POS), it can still be useful.

## Grammatical Quality

See POS Tagging.

Diagram

Description automatically generated

In a typical corpus, the ratio of Nouns to Adnominals is about 3 to 1. In this example, the ratio of Nouns to Adnominals is 1 to 9, as the Object Class count is 899 and the Property count is 7,674. So, what is awry? The next step was to parse names into words, and attach a Part of Speech (Noun, Adjective, etc) tag to each word. This reveals that each Object Class on average uses 3 Nouns for each Adnominal, which is tolerable. However, each Property on average uses 2 Nouns to 1 Adnominal. Simply stated, there are too many Nouns in the Property set, and each Property's nouns should be in the Object Class. In addition, about 4% of Property words are misspelt, which will make it difficult to search to find them. Finally, this is quite a small corpus. It is reasonable to assume that a Corpus would have at least 10,000 nouns.

## Additional Invariants and checks

This is a preliminary list of additional checks.

% Separation of nouns between name spaces -> accidental complexity (unique/distinct vs common/shared nouns)

% Sharing of nouns across name spaces

% Nouns in OC vs non-nouns in OC

% Adjectives in P vs non-adjectives

% Of missing implied words, especially base words from phrases not yet included.

% Of spurious or invalid duplicates

% DEC constructed correctly

Check to determine if the ratio between corpus is credible.

Correct misallocation of current words, etc.

Type Consistency test?

## Building up the Aristotle MDR

These could be functions used to quickly populate an MDR.

Currently, this is done manually, without automated checks, which has produced very mixed quality.

1. Collect all words from all published documents on the web site or the governing Act. These should represent a full set Nouns, Adjectives, phrase of interest to the organisation.

2. Insert or Post these can be stored into Object Class, Property and Data Element Concept corpus.

The Distribution object can be used to populate the file or web or document sources.

3. Collect definitions of words from online dictionaries, such as the Governing Act, OED, Macquarie Dictionary etc. The purpose of Australian Government Agency is defined in the relevant enabling Parliamentary Acts. Included in these Acts are definitions that apply to the Act, and anybody administering these Acts. So, by definition, these definitions are superior to all others.

4. Insert or Post these into Object Class, Property and Data Element Concept corpus.

5. Add a spell check and grammar check to these and allow misspellings with a link back to the correct form.

Audit: Compare database metadata with the document metadata

6. Collect metadata from all available systems and populate into a separate corpus, along with the distribution.

7. Match the 2 corpuses to determine overlap. These can be used to determine the metadata completeness of systems supporting the organisation corpus.

## Other ideas

A. Acronyms can be up to 20% of the words used in a namespace. These can be treated as a Phrase type and placed into the Data Element Concept.

B. Phrases can be made up other Phrases. So, there is a need to provide a recursive link on this object, as an additional relation. This will allow Nouns and adnominals to remain a primitive words, and not compounds. These relations should conform to standard English Grammar. Where they did not conform, this could be a means to identify errors.

C. Additional relations will be needed between Value Domains and Distributions directly to Object Class and Property as a way of providing traceability. the html layer does not support this, but it could easily be implemented in the graph DB layer.

D. Historical, Superseded or archaic words; these are words used previously and have since been replaced. There needs to be come traceability of these terms.

E. Namespaces: Words that are unique to a particular area. These are often not understood outside a particular area of expertise. These need to be captured, and clearly placed into a domain. See diagram.

Diagram

Description automatically generated with medium confidence

F. In the implementation, there are too many workgroups and not enough visibility onto objects like relations.

G. Aristotle does not clearly support the standard security model of UNCLASSIFIED, OFFICIAL, UNOFFICIAL and RESTRICTED. Rules should be defined that apply to all Aristotle items.

H. Item State (candidate, recorded, etc) is not available to an API user, with standard permissions.

Z. It might be possible to use a Description Logic on top of the MDR.

## Aristotle Metadata Views

This shows the profiled or discovered objects used in the example.

Diagram

Description automatically generated

These are the primary populated JSON objects.

## Aristotle Any Item Metadata

This shows the subtypes of the Any Item object, which is the core object in the Aristotle’s data store.

Diagram

Description automatically generated

## MDR Definitions

As a general comment, this document describes the implied data structure of a tool which manages metadata data. This leads immediately into a language confusion trap, and the classic ‘name collision’ problem. The MDR implements the ISO 11179 which is incomplete. The gaps are filled with OOP terminology, so some confusion is inevitable. The major name collisions are Attribute, Class, Object, Property and the woefully named Object Class.

Profiling clarified definitions as examples are always useful in understanding abstract ideas such as metadata. Aristotle has not provided definitions for all JSON formats, so I derived these from profiling. These definitions have a Type of JSON. The Count column indicates the number of JSON records extracted when in late 2021. The Unused column indicates Aristotle concepts that are not used in the sample instance. Therefore, they will not be discussed further.

| 0 | Term | By | Type | Definition | Count | Unused |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | Any Item | Aristotle | JSON | This is the primary boilerplate or super type, which provides common JSON fields for ten formats. This covers the JSON formats for Data Element Concept, Data Element, Data Set Specification, Data Type, Distribution, Property, Relation, Value Domain, Object Class and Object Class Specialisation. All these JSON formats share critical JSON fields including the UUID. Each Any Item has 2 PKs (Primary keys) - a concept Id and a UUID. This is useful when mapping between metadata repositories. In graph theory, this represents the nodes of the underlying MDR graph DB. | 52,036 | 0 |
| 2 | Attribute | 11179 | Standard | This is a characteristic of an object or set of objects. |  | 0 |
| 3 | Attribute | DAMA | Definition | Any detail that serves to qualify, identify, classify, or express state of an entity. |  | 0 |
| 4 | Class | OOP | Definition | A class is an extensible program-code-template for creating objects. A class is a blueprint for creating objects (a particular data structure), providing initial values for state (member variables or attributes), and implementations of behaviour (member functions or methods). |  | 0 |
| 5 | Classification | Aristotle | Definition | A list of mutually exclusive categories representing values of the classification variable. | 0 | 1 |
| 6 | Classification Scheme | Aristotle | Definition | A Classification Scheme describes a set of ideas and standard values used to record codes when storing data. | 0 | 1 |
| 7 | Concept Delta | Aristotle | JSON | This shows the UUID, and date changed. This is useful when doing a change data capture approach for the metadata. | 52,036 | 0 |
| 8 | Conceptual Domain | Aristotle | Definition | A Conceptual Domain describes a set of ideas that can be recorded using codes when storing data. When linked to multiple Value Domains, a Conceptual Domain can be used to find similarities in different code sets. | 0 | 1 |
| 9 | Correspondence Table | Aristotle | Definition | A correspondence table is a collection of mappings that express the relationship between items in different classifications. | 0 | 1 |
| 10 | Custom Value | Aristotle | JSON | This JSON contains custom defined fields for each Any Item subtype. Examples could be Long Name, Tag, etc. | 69,157 | 0 |
| 11 | Data Catalog | Aristotle | Definition | A Data Catalog records a curated collection of data sets. | 2 | 1 |
| 12 | Data Element | Aristotle | Definition | A Data Element is a precise way of defining how a piece of data is recorded for a specific set of objects, using reusable metadata components. Data Elements are composed of a Data Element Concept, which describes the meaning of the data, and a Value Domain which describes how the data is recorded. | 21,654 | 0 |
| 13 | Data Element Concept | Aristotle | Definition | A Data Element Concept defines an idea that could be recorded by data, without specifying how it would be stored or measured. Data Element Concepts are composed of an Object Class, which describes the thing of interest, and a Property that defines which attribute of the thing would be recorded. Data Element Concepts can be referenced by multiple different Data Elements that each specify the Value Domain used to record the data. | 8,117 | 0 |
| 14 | Data Element Derivation | Aristotle | Definition | A Data Element Derivation describes a standardised rule or equation that transforms a set of input Data Elements to produce a set of output Data Elements. application of a derivation rule to one or more input | 0 | 1 |
| 15 | Data Element Path | Aristotle | JSON | This JSON provides the column name and column definition for a Distribution. It is well populated, but not complete. | 43,335 | 0 |
| 16 | Data Set | Aristotle | Definition | A Data Set describes a record of data, including any location or time boundaries for the data, which has been captured and is available for use under a specific licence. A Data Set may be included in a Data Catalog and can reference multiple Distributions. | 2 | 1 |
| 17 | Data Set Specification | Aristotle | Definition | A Data Set Specification describes an agreement to collect an ideal standard of data. A Data Set Specification may reference other Data Set Specifications or Data Elements to describe the data that should be collected under the agreement. | 250 | 0 |
| 18 | Data Type | Aristotle | Definition | A Data Type describes a way of storing a specific form of data within a system. | 13 | 0 |
| 19 | Distribution | Aristotle | Definition | A Distribution describes the structure and format of a specific downloadable collection of data. Multiple Distributions that capture various parts of data or provide different formats for data may be grouped into a single Data Set. | 7,724 | 0 |
| 20 | DSS Cluster Inclusion | Aristotle | JSON | This JSON provides more precise restrictions on a Data Set Specification. There are only a few examples. | 11 | 0 |
| 21 | DSS DE Inclusion | Aristotle | JSON | This JSON provides more details on Data Elements in a Data Set Specification. It is well populated. | 3,114 | 0 |
| 22 | Element | 11179 | Standard | An element or data element is a basic container for data. |  | 0 |
| 23 | Entity | DAMA | Definition | An entity may be defined as a thing capable of an independent existence of interest to the business that can be uniquely identified. |  | 0 |
| 24 | Enum | Computer | Definition | An enumeration of a sum type. This can also a code, such as Public or Private in Scope. |  | 0 |
| 25 | Framework | Aristotle | Definition | A Framework describes an organised collection of targets and strategic outcomes to assess a broad policy area. A Framework can collect multiple Indicator Sets and Outcome Areas to provide a complete understanding of the assessment of progress to a group of related goals. | 0 | 1 |
| 26 | Glossary Item | Aristotle | Definition | A Glossary Item records a business term that is commonly used within the metadata registry. A collection of Glossary Items is called a business glossary. | 2 | 1 |
| 27 | Graph DB | Computer | Definition | A graph database is a database that uses graph theory node and edge data type tables to represent data. Graph data type tables are sufficiently abstract to represent any architectural diagramming method, such as data models, Zachman diagrams, etc. |  | 0 |
| 28 | Identifier | Aristotle | JSON | This JSON contains the concept id and the UUID for each Any Item subtype. Both keys are used throughout Aristotle. This is important when mapping metadata tools. | 46,121 | 0 |
| 29 | Indicator | Aristotle | Definition | An indicator describes a measure that is regularly reported for tracking performance of a process or policy and provides relevant and actionable information about system performance. | 0 | 1 |
| 30 | Indicator Set | Aristotle | Definition | An Indicator Set describes a collection of targets and objectives. An Indicator Set collects multiple Indicators with common targets that are reported on together. | 0 | 1 |
| 31 | ISO 11179 | ISO | Standard | A standard for representing metadata. |  | 0 |
| 32 | Issue | Aristotle | JSON | This shows details about quality issues raised in the MDR. It is low volume. | 23 | 0 |
| 33 | Link | Aristotle | JSON | This provides more information regarding the relations. | 9,614 | 0 |
| 34 | Metadata | DAMA | Definition | Metadata is "data that provides information about other data". In other words, it is "data about data". |  | 0 |
| 35 | Narrower Class | Aristotle | JSON | This JSON provides more precise restrictions on an Object Class Specialisation. This helps make the types more specific. There are only a few examples. | 642 | 0 |
| 36 | Object | OOP | Definition | An object is an instance of a class that contains properties and methods. |  | 0 |
| 37 | Object Class | Aristotle | Definition | An Object Class defines a way of identifying or classifying a set of real objects, ideas, or events that all share common measurable attributes. | 880 | 0 |
| 38 | Object Class Specialisation | Aristotle | Definition | An Object Class Specialisation describes a relationship between Object Classes, where multiple specialised Object Classes are all contained by a common broader Object Class. | 17 | 0 |
| 39 | Org Record | Aristotle | JSON | This JSON defines Org Record data for an Any Item subtype. It is not used much. | 497 | 0 |
| 40 | Outcome Area | Aristotle | Definition | An Outcome Area describes a strategic target or standard for a process or policy that may not be able to be measured directly or efficiently. | 0 | 1 |
| 41 | Permissible Value | Aristotle | JSON | This JSON contains the enum or code values for a Value Domain. It is well populated. | 64,046 | 0 |
| 42 | Property | 11179 | Standard | A characteristic common to all members in an object class. |  | 0 |
| 43 | Property | Aristotle | Definition | A Property is an attribute common to all members of a set of things defined by an Object Class. | 7,336 | 0 |
| 44 | Property | English | Definition | Something that belongs to something. An adnominal, such as an adjective or adverb. |  | 0 |
| 45 | Quality Statement | Aristotle | Definition | A Data Quality Statement records any known issues that may be related to a data asset. A Data Quality Statement assesses data against seven key factors: Institutional Environment, Relevance, Timeliness, Accuracy, Coherence, Interpretability & Accessibility. | 0 | 1 |
| 46 | Relation | Aristotle | Definition | A Relation defines a relationship that can be used to link different metadata items within the registry. Each Relation can specify a number of roles that metadata can fill within the relationship. In graph theory, this represents the underlying edges of the underlying MDR graph DB. Due to permission issues, only a few were visible. | 4 | 0 |
| 47 | Relation Role | Aristotle | JSON | This JSON provides some definitions of the relations such as name and multiplicity. There are only a few examples, due to permissions restrictions. | 7 | 0 |
| 48 | Representation Class | Aristotle | JSON | A supplemental logical data type. Not populated as an object but called a 'managed item'. | 18 | 1 |
| 49 | Slot | Aristotle | JSON | This JSON provides additional information for Value Domains and Data Elements. It is not used much. | 182 | 0 |
| 50 | Stewardship Org | Aristotle | JSON | This defines the Organisation that controls the metadata and is part of authorisation. This appears on all JSON items. | 2 | 0 |
| 51 | Supplementary Value | Aristotle | JSON | This JSON contains some additional values for a Value Domain. There are very few values. | 34 | 0 |
| 52 | Units Of Measure | Aristotle | Definition | A Unit of Measure describes units (e.g., metres, litres, seconds) which can be used to record a measurement. | 0 | 1 |
| 53 | Value Domain | Aristotle | Definition | A Value Domain describes how to record the measurement of a particular type of data, either using a coded list of values or a description of the possible values. Value Domains can be linked to Data Elements that all share a common way of recording data, and its values can be linked to a Conceptual Domain to provide additional context. | 3,893 | 0 |