

Providing Verifiable Oversight for Scrutability, Assurance and Accountability in Data-driven Systems

Iain Barclay

November, 2022

**A thesis submitted in partial fulfilment
of the requirement for the degree of Doctor of Philosophy**

**Cardiff University
School of Computer Science & Informatics**

Abstract

The emergence of data-driven systems that inform decisions or offer recommendations impacts all sectors, including high-stakes settings where judgements affecting health, education and security are made. There is little visibility afforded into the qualities of the constituent components of these systems, or how they have been prepared and assembled. This makes it difficult for stakeholders to scrutinise systems and build confidence in system quality – which is important as problems resulting from poorly prepared or mismanaged data can have serious consequences. There is motivation to foster trustworthy systems, based on transparency and accountability, but there are currently shortcomings in tools that offer the desired scrutability onto data-driven systems, whilst protecting confidentiality requirements of providers.

This thesis adopts a design research approach to address these shortcomings by designing and demonstrating information systems artefacts that enable providers to take accountability for their contributions to data-driven systems and provide verifiable assertions of the properties and qualities of systems and components to authorised parties. The outcomes are a framework to help identify parties that contribute to the provision of data-driven systems, and a conceptual model that adopts a bill of materials document to record system supply chains. These artefacts are employed in software architectures that provide verifiable assurance of the qualities of digital assets to authorised parties and offer scrutability on data-driven systems. The software architectures adopt decentralised data models and protocols based on self-sovereign identity paradigms to place accountability on providers of assets. This enables domain users and other

stakeholders to seek assurance on the qualities of systems and assets, whilst protecting sensitive information from unauthorised access. This thesis contributes to the adoption of self-sovereign identity data models and protocols for parties to ratify qualities and take accountability for digital assets, extending their scope from the current dominant usage for personal identity information.

Acknowledgements

I am extremely fortunate to have met Professor Alun Preece and Professor Ian Taylor when I returned to Cardiff University in 2016. Together they guided me through my Master's dissertation and gave me the confidence to undertake doctoral research. I am thankful for the support, insight and opportunities that they have both provided so generously. I am grateful to Professor Jarek Nabrzyski from the University of Notre Dame, for welcoming me into his research group, and hosting me as a visiting scholar.

The COVID pandemic curtailed many adventures, but I have enjoyed working with some excellent researchers, and especially thank Swapna Krishnakumar Radha, Will Abramson, Michael Cooper, Jakob Hackel, Maria Freytsis, Chris Simpkin, Harrison Taylor for their enthusiastic collaborations. I thank the members of my Expert Review Panel, and my colleagues from Electric Pocket and SIMBA Chain for their support. My friends have been brilliant, with their interest and encouragement, especially Peter Luciani who has kept me buoyed up throughout.

The love and patience of my family has sustained me. I take inspiration from my children Cerys and Dan, and the successful starts they are making to their careers, and from the contribution my wife Kathy quietly makes to enrich the lives of so many. Geno, and Iolo before him, have helped me to relax. I am thankful to my Mum, and my brother Graeme and his family, for their support from afar. My dear old Dad is no longer here, but is never far from my thoughts. I thank him for the grit and determination that I didn't know I had inherited, but has been the most important contribution of all.

Contents

Abstract	i
Acknowledgements	iii
Contents	iv
List of Figures	x
List of Tables	xiii
Abbreviations	xv
1 Introduction	1
1.1 Data-driven Systems	1
1.2 Towards Verifiable Oversight	4
1.3 Hypothesis and Research Questions	5
1.4 Contributions	6
1.5 Organisation of the Thesis	7
1.6 Method	9

1.6.1	Design Science Research	9
1.6.2	Expert Review Panel	15
1.7	Published Papers	20
2	Background and Problem Definition	23
2.1	Introduction	23
2.2	Literature Review	24
2.2.1	Information Visibility, Transparency and Accountability . . .	25
2.2.2	Documentation of Shared Data Assets	26
2.2.3	Documentation of ML Models and Data-driven Systems . . .	30
2.2.4	Supply Chain Traceability	35
2.2.5	Stakeholder Roles in DDS	36
2.3	Emerging Policy Directions	37
2.4	Self-Sovereign Identity Protocols and Patterns	39
2.5	Gap Analysis	41
2.6	Summary	45
3	A Framework for Roles and Boundaries in Data-driven Systems	47
3.1	Introduction	47
3.2	Problem Identification	48
3.3	Design and Build	49
3.3.1	Design Cycle 1: The Machine Learning Pipeline	49
3.3.2	Design Cycle 2: UML Modelling	59

3.4	Research Outputs	65
3.4.1	The Roles and Boundaries Framework	65
3.4.2	Communication	67
3.5	Evaluation	68
3.5.1	The Rigour Cycle: Expert Review Panel	69
3.5.2	The Relevance Cycle	77
3.5.3	Limitations	78
3.6	Summary	79
4	A Verifiable Supply Chain Bill of Materials for Data-driven Systems	82
4.1	Introduction	82
4.2	Problem Identification	83
4.3	Design and Build	84
4.3.1	Design Cycle 1: A Bill of Materials	84
4.3.2	Design Cycle 2: A Schema for a DDS BOM	88
4.4	Research Outputs	93
4.4.1	A Bill of Materials for DDS and Supporting Data Model	93
4.4.2	Communication	95
4.5	Evaluation	95
4.5.1	The Rigour Cycle: Expert Review Panel	96
4.5.2	The Relevance Cycle	103
4.5.3	Limitations	105
4.6	Summary	106

5	Providing Accountability, Oversight and Information Security for Digital Assets	108
5.1	Introduction	108
5.2	Problem Identification	109
5.3	Design and Build	110
5.3.1	Design Cycle 1: Investigation of Verifiable Credentials	110
5.3.2	Design Cycle 2: Self-sovereign Data	114
5.3.3	Design Cycle 3: Policy Based Access Control	121
5.4	Research Outputs	130
5.4.1	SSI-based Software Architecture Providing Assurance and Accountability on Digital Assets	130
5.4.2	Communication	132
5.5	Evaluation	133
5.5.1	The Rigour Cycle: Software Quality Review	134
5.5.2	The Relevance Cycle	140
5.5.3	Limitations	142
5.6	Summary	143
6	Providing Verifiable Oversight on Data-driven Systems	145
6.1	Introduction	145
6.2	Problem Identification	146
6.3	Design and Build	147
6.3.1	Design Cycle 1: A Verifiable Bill of Materials	147

6.3.2	Design Cycle 2: Providing a Human-Machine Interface	154
6.4	Research Outputs	159
6.4.1	A Software Architecture Providing Oversight and Accountab- ility on DDS through a Verifiable BOM	159
6.4.2	Communication	161
6.5	Evaluation	162
6.5.1	The Rigour Cycle: Architecture Decision Review	163
6.5.2	The Relevance Cycle	168
6.5.3	Limitations	171
6.6	Summary	173
7	Conclusion	175
7.1	Introduction	175
7.2	Research Outcomes	176
7.2.1	Contributions to the Knowledge Base	178
7.2.2	Limitations	179
7.3	Future Work	180
7.3.1	Self-sovereign Identity	180
7.3.2	New Modes of Access Control	181
7.3.3	Human-Centred Research	182
7.3.4	Verifier Roles in Decentralised Architectures	183
7.3.5	Emerging Data-centric Design Patterns	184
7.3.6	Asynchronous Design Review Methods	184

Bibliography	186
A Definition of Terms	208
B Presentation to Expert Review Panel	213
C DDS BOM Schema Code Listing	218
D AI Scrutineer	224
E Decision-Centric Architecture Review (DCAR)	225
F Decision Descriptions from DCAR Review	228
G Presentation of Asynchronous DCAR Review	230

List of Figures

1.1	Interplay between problem and solution in Design Science Research .	11
1.2	Design Science Research Stages and Iterations	12
1.3	The role of Expert Interviews in Problem Identification	17
1.4	Introductory Slide presented to ERP interviewees	18
2.1	Technology Readiness Levels	33
2.2	The Triangle of Trust	41
3.1	DDS Production Pipeline	50
3.2	Hierarchy of Roles and Contributions	56
3.3	Relationships between DDS Components	60
3.4	Revised Framework for DDS Analysis	60
3.5	Framework Instantiated for the CCTV Monitor Scenario (S1)	61
3.6	Second Iteration of Framework Analysis on the CCTV Monitor Scenario (S1)	62
3.7	Revised Second Iteration on the CCTV Monitor Scenario (S1)	63
3.8	Analysis of the Aurora Scenario (S2)	64

3.9	The Roles and Boundaries Framework	65
3.10	DDS Representation in UML	66
3.11	Roles and Boundaries Slide presented to ERP interviewees	70
4.1	Data Model for a DDS Bill of Materials	94
4.2	Data Supply Chain Slide presented to ERP interviewees	96
4.3	Verifiable Supply Chain Slide presented to ERP interviewees	97
5.1	Credentials issued by Data Providers	115
5.2	Data Users verify Claims	116
5.3	Process Flows in Publication and Verification.	117
5.4	Architecture using Verifiable Credentials	123
5.5	Access to Data in MMA Architecture	125
5.6	Interaction Flows for SSI Policy-based Access Control	126
5.7	Architecture of Experimental Framework	127
5.8	SSI Roles in the Demonstration Scenario	129
5.9	Context View of Digital Asset Assurance and Accountability System .	132
5.10	Architecture in support of Digital Asset Sharing Requirements	133
6.1	Domain Authority Interactions with SSI Agents of Constituents	150
6.2	AI Scrutineer Architecture	156
6.3	Context View of DDS Oversight and Accountability System	160
6.4	Functional View of DDS Oversight and Accountability System	161
6.5	Decision Relationship viewpoint of Architecture	165

D.1 Screenshot of the AI Scrutineer Web Interface	224
---	-----

List of Tables

1.1	Members of the Expert Review Panel	16
2.1	Summary of Approaches to Dataset and Model Documentation	43
3.1	Solution Requirements	49
3.2	Roles in a DDS	51
3.3	Roles Identified from Scenario Descriptions	59
3.4	UML Nomenclature	60
3.5	Roles in a DDS	61
3.6	Entities in a DDS Ecosystem	66
4.1	Solution Requirements	84
4.2	Elements of an Asset in a BOM document	90
4.3	Entities in Scenario S1, CCTV Monitor	92
4.4	Entities in Scenario S2, Aurora	93
5.1	Solution Requirements	110
5.2	Use Case Scenario mapped to Architecture	124

5.3	Components of the Demonstration Scenario	128
5.4	Roles and Access Policies	129
5.5	SSI Principles as manifested in the System Architecture	135
6.1	Solution Requirements	146
6.2	Credentials issued to SSI Agents in CCTV Monitoring Scenario . . .	151
6.3	Credentials issued to SSI Agents in Aurora Scenario	153
6.4	Decision Description Template	165
6.5	Self-reported Knowledge of SSI and Time Taken	167
F.1	Decision Description: Standard SSI Interfaces	228
F.2	Decision Description: Use VCs for Metadata	229
F.3	Decision Description: Encode BOM as VC	229

Abbreviations

BOM Bill of Materials

DA Domain Authority

DDS Data-driven Systems

DID Decentralised Identifier

DP Data Provider

DSR Design Science Research

DSRM Design Science Research Methodology

ERP Expert Review Panel

ME Model Engineer

RB Roles and Boundaries

SI Systems Integrator

SSI Self-sovereign Identity

VC Verifiable Credential

VP Verifiable Presentation

Introduction

1.1 Data-driven Systems

Burgeoning cloud-based data storage, high bandwidth connectivity, and development of the internet of things have led to the emergence of data ecosystems [134]. Collaborators in these complex multi-actor environments adopt a variety of datasets, algorithms and machine learning models to produce *data-driven systems*¹, which manifest in solutions and services offering capabilities such as algorithmic decision making. This is in contrast to Knowledge-based Systems [27], which apply rules and logic to structured information.

Algorithmic data-driven systems (DDS) are increasingly prevalent, and the outputs they produce inform critical decisions made in high-stakes fields including agriculture, education, healthcare, and security. Data-driven systems include fraud detection systems [23], which utilise machine learning algorithms to analyse large volumes of data, including transaction records, user behaviour, and historical patterns, to detect anomalies or suspicious patterns indicative of fraud. In healthcare, datasets of medical images, patient records, genomic data, and clinical research, are analysed to assist clinicians in diagnosing diseases [2]. Law enforcement also makes use of DDS, with predictive policing systems [101] using historical crime data, demographics, weather conditions, and other relevant factors to identify patterns and predict areas where crimes

¹Definitions for terms written in italics in this section are provided in Appendix A

are more likely to occur.

Contributions to any individual data-driven system deployment can come from sources that include scientists and academic research groups, public and government agencies, as well as commercial entities sharing datasets, or providing access to machine learning (ML) models through AI-as-a-Service (AIaaS) subscriptions [24, 92, 155]. Parties with assets used as contributions in the development of DDS may be well known to each other, or, with access to datasets, ML models, and human work and expertise increasingly available through shared repositories and internet platforms [1] they may have no prior or direct relationship. Often, a vendor or systems integrator will assemble a DDS from third party datasets and other components, and provide it to operators in the deployment domain [74, 160]. The potential gulf between the original dataset providers, algorithm developers, and those relying on the deployed systems raises questions about how to provide *verifiable oversight* on a DDS, so that parties in the deployment domain can *scrutinise* systems, and be assured that the systems are *appropriate* for their use [74, 90]. The situation is further complicated, as providers of datasets or algorithms may have strong motivations to protect commercially secret or sensitive information about their datasets and other assets [4].

The ability for different stakeholders to scrutinise systems and demonstrate the *transparency* and *accountability* of data-driven systems is of increasing importance, as well-documented evidence of problems caused in systems due to bias in data collection and preparation, or poor engineering and management practices come to light [20, 30, 117]. In order to demonstrate that they are *trustworthy*, organisations adopting DDS need to be able to demonstrate that they know, and are confident in, the *provenance* of the underlying data and knowledge they use to make decisions [46, 50, 168]. This has resulted in motivation to improve documentation and governance practices, backed by policy directives from governments and influential global organisations. UNICEF’s Policy Guidance on AI for Children [32], for example, states that: “Data equity and representation of all relevant children for a particular AI system, including children

from different regions (including rural communities), ages, socioeconomic conditions and ethnicities, is essential to protect and benefit children. *For example, in the case of data-driven health care, children's treatment or medication should not be based on adults' data since this could cause unknown risks to children's health* [emphasis added].” – this raises important questions about how assurance that a DDS meets such requirements can be provided to stakeholders for any such system in deployment, or under consideration for deployment.

What is needed is a solution to the problem of a lack of *scrutability* on DDS. A viable solution will provide stakeholders with information on the system's production, and the qualities of components from multiple sources that are adopted in the system, in order to help build *trust* in the system. The solution must also protect and mediate access to commercially sensitive or private information belonging to vendors and component providers. Such a solution will enable stakeholders to perform scrutiny and gain confidence that systems they use are appropriate, providing them with evidence to justify system selection and usage. Current solutions do not adequately address this need, as they are unable to provide verifiable information and demonstrate accountability across the multiple participants in data ecosystems, and have no way to mediate between transparent provision of information and protection of the intellectual property of contributors.

The problem can be deconstructed into the following parts:

- Dependencies between stakeholders, and motivations for transparency, and confidentiality or privacy of different stakeholders and contributors can be unclear, leading to a lack of oversight, traceability and accountability in such systems.
- There is a lack of support for documenting contributions made to DDS, which need to be recorded in a structured, machine readable form, so that information can be exchanged between stakeholders and traceability through systems can be provided.

- Contributions towards a DDS may come from multiple providers, with different relationships between providers across the ecosystem presenting barriers to information disclosure, and confidence in the integrity of assets and their providers being developed.
- There are shortcomings in current provisions to offer scrutiny and provide verifiable oversight on DDS as a whole, and demonstrate accountability for their component assets and contributions.

1.2 Towards Verifiable Oversight

Researchers and practitioners concerned with the privacy of identity and personal data have developed a paradigm known as self-sovereign identity (SSI) [5]. Their work has led to the development of decentralised data models and protocols that can provide verifiable proofs of endorsement of claims made about individuals, whilst protecting information from unauthorised access. SSI allows any party to digitally sign and issue assertions about the attributes or qualities of another party, without reliance on a central agency or authority. Whilst the focus of the SSI community is largely concerned with personal identity and privacy, the approach is also suitable for use with non-human entities and assets. Indeed, much of what is proposed around the ability to issue and verify claims about an individual, whilst maintaining control of access to private information, aligns with the requirements to provide controlled, verifiable assurance of the integrity of claims made by, and about, contributors to a DDS.

The nature of a DDS is that contributions to the complete system can originate from multiple parties, often with very loose relationships. A decentralised approach, such as that offered by SSI technologies, has benefits in such circumstances as it allows individual parties to retain ownership of their own data, without having to rely on a central authority or agency to protect sensitive commercial or research information. After preliminary investigations, we determined that a viable approach to providing

verifiable oversight on DDS qualities could be delivered through the design of an architecture based upon SSI data models and protocols. With such an approach, stakeholders could be provided with oversight on DDS through an architecture that uses decentralised, self-sovereign technologies to maintain a record of contributions made to the system by different participants. The same technologies could be used to allow system integrators and component providers to ratify the qualities of assets used, whilst privacy-protecting properties afforded by SSI could be used to protect and mediate access to confidential information. This approach could provide stakeholders across the ecosystem of a DDS with verifiable assurance of claims made by contributors. Such an architecture could provide the scrutability, transparency and accountability increasingly required from DDS deployments, and help stakeholders to gain confidence in the suitability of assets used in a DDS, and in the DDS itself.

1.3 Hypothesis and Research Questions

The hypothesis of this thesis is that adoption of a decentralised approach using self-sovereign identity data models and protocols can provide stakeholders of data-driven systems with verifiable oversight onto systems and constituent parts of systems, offering scrutability on contributions to the systems and identifying parties who are accountable for contributions, whilst protecting commercial or private information from unauthorised disclosure.

Based on the identified problem of a lack of transparency and accountability in DDS, and our hypothesis that a decentralised SSI approach can be used to design a viable solution to the problem, the main research question for this thesis is formulated as:

How can self-sovereign identity models be used to provide verifiable oversight and accountability on DDS to stakeholders, whilst maintaining confidentiality and privacy requirements of contributing parties?

To address this question, the following sub-questions are considered:

- RQ1:** What are the roles involved in developing and using a DDS, and what are their responsibilities and requirements? This will help to clarify dependencies between stakeholders, and motivations for transparency, and confidentiality or privacy of different stakeholders.
- RQ2:** How can contributions to a DDS be recorded and documented, so that traceability can be provided to stakeholders? This will address the lack of support for documenting contributions made to DDS in a structured, machine readable form, and support information exchange between stakeholders.
- RQ3:** How can SSI models be used to provide accountability and assurance on the qualities of assets contributed by different participants to a DDS, whilst maintaining the information security requirements of the contributors? This will determine how asset providers can give confidence to authorised parties.
- RQ4:** How can SSI models be used to provide verifiable oversight and accountability on a DDS, so that systems can be scrutinised by authorised stakeholders? This will address shortcomings in current provisions to provide scrutability on DDS.

1.4 Contributions

The research described in this thesis provides contributions to the knowledge base which include:

C1. A Conceptual Framework mapping Stakeholder Roles

We present a conceptual framework that can be used to decompose the hierarchy of a DDS to identify different roles, responsibilities and requirements placed upon actors, and tensions that exist through conflicting requirements for transparency and confidentiality or privacy across role interfaces, and show its relevance through peer review. This contribution addresses RQ1.

C2. A Verifiable Supply Chain Bill of Materials for Data-driven Systems.

We define and demonstrate a method for documenting DDS based on an industrial supply chain bill of materials (BOM) model, such that contributing assets can be identified, and verification of claims made about those assets can be sought from accountable parties. We design a schema that supports data modelling for such a BOM structure for a DDS. The proposed supply chain BOM model and data model provides resolution to RQ2.

C3. A Software Architecture, providing Accountability and Assurance on Digital Assets.

We design a software architecture that uses decentralised, self-sovereign technologies to provide accountability on digital assets and artefacts. The architecture, demonstrated in a case-study, enables verifying parties to gain assurance by validating claims made about assets and asset providers, whilst protecting confidential commercial and personal information from unauthorised disclosure. This contribution provides a solution for RQ3.

C4. A Software Architecture, providing Oversight and Accountability across Digital Asset Supply Chains.

We present a software architecture based on SSI principles that uses signed digital credentials to provide verifiable oversight across the supply chain of contributions to a DDS, offering accountability on contributions to such systems. The architecture, evaluated by peer review, enables authorised parties to scrutinise systems and contributions, whilst offering protection for privacy and confidentiality of contributors. This architecture provides a solution for RQ4.

1.5 Organisation of the Thesis

Following this Introduction, which concludes with a description of the research method and an overview of the published papers that have contributed to the research presented

here, the remainder of this thesis is structured as follows:

Chapter 2 provides background to the research area, and identifies the research problem. We present a review of relevant literature, and consider research, along with policies and recommendations from governments, non-governmental organisations (NGOs) and industry bodies that identify the need for providing assurance on DDS, and requirements for providing oversight on the underlying data used to train and test such systems. The literature review identifies related work in the provision of information and documentation about data which is intended to be reused and shared by third parties, and documentation designed to accompany ML models, as well as related work from software and manufacturing industries. We also provide background on self-sovereign identity, the technology approach adopted in our research.

Chapter 3 develops a framework that identifies different participants in a DDS deployment - from domain stakeholders who require assurance on the suitability of a system, through systems integrators and engineers, to the parties curating and providing datasets. The responsibilities and individual goals of each role in the ecosystem are explored, and presented in a framework that identifies interactions and responsibilities between the roles.

Chapter 4 proposes the adoption of a supply chain model for tracing contributions from data and other sources in DDS, and argues that maintaining a bill of materials as record of contributions made to the development of digital assets can help to provide oversight on solutions in the field, so that deployed DDS can be used with confidence. A schema is designed to enable the BOM to be documented and maintained in an interoperable and machine-readable format.

Chapter 5 considers the role of shared data assets in a DDS, and the importance of such data being demonstrably trustworthy. The requirements of data providers and data users are investigated. A software architecture which supports the assertion of data qualities by trusted parties using SSI data models and protocols is designed, and demonstrated through application to a case study based on data sharing requirements

in the multi-messenger astronomy community.

Chapter 6 builds upon the research presented in Chapter 4 and Chapter 5, and designs a system architecture for a solution which uses a BOM to record the constituent parts of a DDS, and demonstrates the use of SSI technologies to ratify the system BOM, as well as qualities of the individual elements in the supply chain. The system is used in a demonstration of a web-based tool which provides end users of a DDS with a means of inspecting the contributions of digital assets, using SSI technologies to maintain up-to-date assertions of qualities and provide accountability.

The thesis concludes with Chapter 7, which considers the artefacts developed through the thesis against the solution objectives for a system to provide assurance to users of DDS. The research developed in this thesis is used to identify areas for future work towards the provision of verifiable oversight and accountability on multi-stakeholder digital systems.

1.6 Method

The research questions posed in this thesis centre upon DDS. These are complex, multi-actor socio-technical systems, and lead to our research questions exhibiting properties of “wicked problems” [28]. Problems classed as wicked have characteristics which include unstable requirements and constraints based on poorly defined contexts, complex interactions among sub-parts of the problem, inherent flexibility in processes and artefacts, a reliance on creativity to produce solutions, and a critical dependence upon human social abilities to produce effective solutions [67].

1.6.1 Design Science Research

The Design Science Research (DSR) approach is considered to be well suited to addressing wicked problems [69], as it is a problem-solving paradigm which seeks to cre-

ate innovations that define ideas, practices, technical capabilities, and products through which the analysis, design, implementation, and use of information systems can be effectively and efficiently accomplished [67]. The research conducted in this thesis was developed through the Design Science Research Methodology (DSRM) framework of Peffers, et al. [122, 121], for applying a DSR approach to Information Systems (IS). Hevner, et al. [69], state that the objective of IS research is to “acquire knowledge and understanding that enable the development and implementation of technology-based solutions to heretofore unsolved and important business problems”, using design “to change existing situations into preferred ones” [143]. Engström, et al. [52], provide the mapping shown in Figure 1.1, which shows relationships between the problem and the solution, and DSR activities which generate knowledge, as well as the forms that generated knowledge can take.

Hevner, et al. [69], provide guidelines for adopting DSR in IS, which Peffers, et al. [122], used in deriving their DSRM framework. In particular, Hevner and colleagues’s contention that research must produce an “artefact created to address a problem”, and that artefacts should be rigorously evaluated and the research communicated to appropriate audiences.

The DSRM framework identifies six activity groups to guide researchers: 1) Problem identification and motivation; 2) Definition of the objectives for a solution; 3) Design and development of artefacts that contribute to the solution; 4) Demonstration of the use of the artefacts to solve one or more instances of the problem; 5) Evaluation of the solution, by comparing the outcomes with the objectives; and 6) Communication of the problem and its importance, the artefacts, and findings from the research.

This thesis adopts the DSRM framework and takes a problem-centric approach, building on the identification of shortcomings in providing scrutability and oversight on DDS. This guides us to follow the sequence of activities in DSRM the framework in a linear manner, starting from Step 1. As Figure 1.2 shows, iteration and improvement based on learning is a core feature of the DSRM approach, and our research has

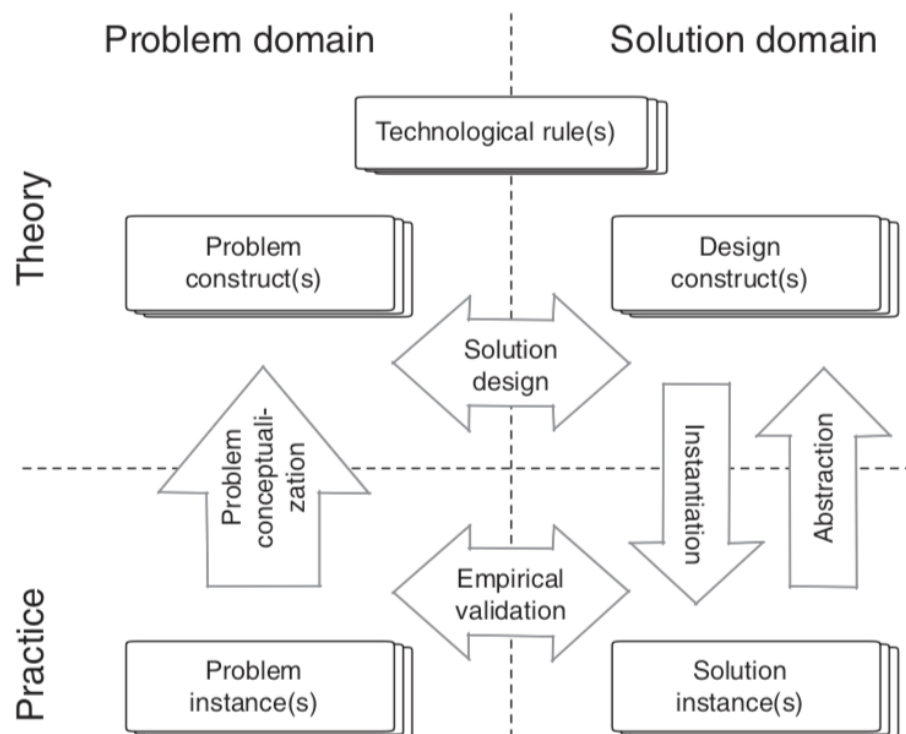


Figure 1.1: The interplay between problem and solution, and between theory and practice in design science research. The arrows illustrate the knowledge-creating activities, and the boxes represent the levels and types of knowledge that is created
From Engström, et al. [52] (CC-BY).

revisited many of the activities as our understanding both of the problem and of the emerging solution has developed. Iteration across a number of design cycles in solution development helps to refine the objectives of the solution based on lessons learned during the design, development and demonstration phases.

The DSRM is structured around the following activities:

1. **Problem Identification and Motivation:** This activity identifies the specific research problem, and justifies the value or significance of providing a solution to this problem. Section 1.1 of this chapter has presented the problem statement, and enumerated it into elements of the problem, and Section 1.6.2 provides in-

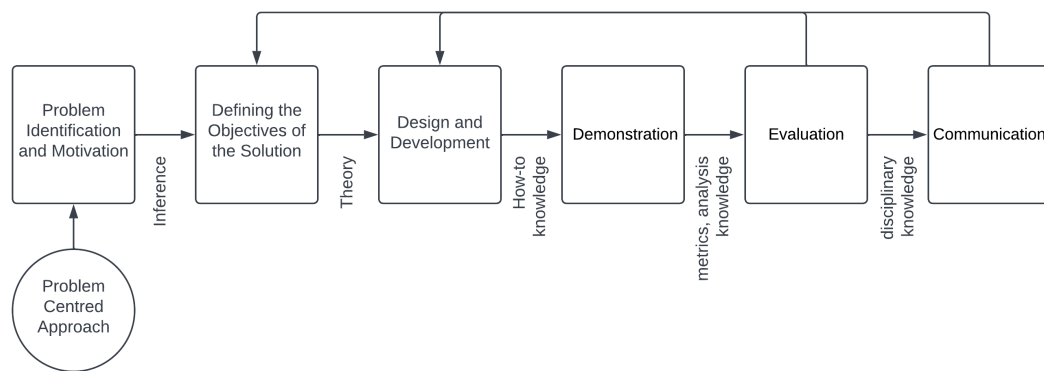


Figure 1.2: Design Science Research stages and iterations, adapted from Peffers, et al. [122].

sight on the the problem area from experts and practitioners. Chapter 2 provides further background, and examines the importance of finding a solution to the problem considering emerging policy in regards to providing transparency, assurance and accountability on classes of DDS, and through critical analysis of current solutions. Chapters 3, 4, 5, and 6 further consider specific aspects of the problem.

2. **Defining the Objectives of the Solution:** In this activity, the objectives of a solution are inferred from the problem definition, and knowledge of what is feasible or possible. Motivation of our solution is addressed through the need to resolve the main research question and the sub-questions described in Section 1.2. In particular, we seek to clarify and to address tensions in DDS between needs for transparency and accountability, and for confidentiality and privacy, whilst proving trustworthy information on asset qualities across different stakeholders and asset providers. As a result, artefacts resulting from our research are expected to contribute towards providing transparency and accountability on DDS deployments, whilst enabling asset contributors to maintain control of their private or confidential information. Building on our prior technical knowledge,

and to test the hypothesis presented in Section 1.3, a specific objective of our solution approach is that it is built upon decentralised technologies, and in particular self-sovereign identity models.

3. **Design and Development:** This activity is concerned with creation of IS artefacts that contribute towards the solution of the defined problem. Conceptually, an artefact can be “any designed object in which a research contribution is embedded in the design” [122], and can include architectures, designs, models, and methods. Artefact design and development forms the core of this thesis, and is presented across several chapters: Chapter 3 provides a conceptual framework to identify stakeholders and their roles, responsibilities and requirements in a DDS; Chapter 4 proposes a verifiable supply chain bill of materials model to enumerate and document DDS components; Chapter 5 designs an architecture using SSI technologies to enable publishers to ratify claims about their digital assets and Chapter 6 designs an architecture that enables a complete DDS to be documented to offer transparency, demonstrate accountability, and provide stakeholders with oversight and the ability to scrutinise systems.
4. **Demonstration:** This activity involves using the artefacts to solve one or more instances of the problem. Scenarios based upon case studies are used during the demonstration phase, and are described in context in Chapters 3, 4 and 5. Chapter 6 brings together research from the preceding chapters, describing a system which demonstrates the feasibility of the proposed solution approach by aggregating artefacts developed through the thesis in a case study that integrates with a popular ML development pipeline tool and offers scrutiny on a DDS to stakeholders.
5. **Evaluation:** This reflective activity enables us to consider how well the designed artefacts are likely to be able to deliver a solution to the problem. Evaluation in this thesis is guided by the Framework for Evaluation in Design Science Research (FEDS) provided by Venables, et al. [165], and adopts techniques to eval-

uate for rigour and for relevance. Evaluation approaches appropriate to each artefact are employed: Chapters 3 and 4 are informed by analysis of semi-structured interviews with expert witnesses on an Expert Review Panel (ERP), the format of which is described below in Section 1.6.2. Chapters 5 and 6 which contain technical artefacts in the form of software architectures, and are evaluated for their technical risk and efficacy, and against criteria that assess software quality attributes and the decisions made during the design process.

6. **Communication:** A significant activity in the DSRM framework is communication of the problem and its importance, and the utility and novelty of the artefacts to suitable audiences, including other researchers and professionals. A number of peer-reviewed academic papers have been published during performance of the research in this thesis, and are listed in Section 1.7. In particular, our research has been presented in workshops and published in journals of the science gateway community, who are often the providers and users of data-driven assets and systems. A peer-reviewed paper has also been presented to a global workshop concerned with accountability of DDS, and presented in several industry and academic seminars. The research problem and progress towards a solution has been regularly presented to colleagues and peers, in joint academic and industry meetings of the international DAIS-ITA program², and to the Blockchain Research Group³ in the Center for Research Computing at University of Notre Dame, USA.

Further inspiration for the method is drawn from the Agile Design Science Research approach of Conboy, et al. [40], and in particular their mapping of activities and processes from agile software development sprints to the DSRM stages has been helpful. Indeed, DSRM ties in well with modern approaches to software architecture adopted in industry, such as the Continuous Architecture approach developed by Erder and

²<https://dais-legacy.org/>

³<https://crc.nd.edu/research/blockchain-research-group/>

Pureur [53], which is motivated by the notion that “getting to an executable architecture quickly and then evolving it is essential for modern applications”. This links well with the iterative and reflective approach of DSRM across design cycles.

1.6.2 Expert Review Panel

Evaluation of the artefacts produced as a result of design work presented in Chapters 3 and 4 was performed during a series of semi-structured interviews with peers from government, industry, the third sector, and academia who formed an Expert Review Panel. A feature of the semi-structured interview is “an incomplete script” [109] in which some questions are prepared, but scope remains for improvisation. This approach was chosen so that discussion could be adapted based on interviewee’s answers to questions posed, and to enable areas of particular interest and expertise to be explored more fully.

ERP members were recruited from personal connections, and outreach messaging on social media channels (Twitter, LinkedIn). Subjects were invited to make contact in response to this appeal “I’m looking to hold short, informal, conversations with people with interests and experience in areas like #datagovernance, #opendata and ‘data driven systems’ (eg. ML/AI systems), etc.”. Interested parties were sent further information and a formal invitation. Of the nine subjects interviewed during the ERP sessions, five were previously known to the interviewer, and four were new connections reached through social media messaging. Six who had shown initial interest did not respond further. Interviews with members of the ERP were conducted over individual video calls using Microsoft Teams. Each interview began with the interviewee providing a short introduction to themselves, their current role and background, and their experiences with data or data-driven systems, which are summarised in Table 1.1. ERP members represented a range of experience and expertise with the data and ML field, helping to avoid “elite bias” [109]. Disappointingly, only one of the nine members of the ERP was female.

ID	Role	Sector	Size	Expertise
A	Research Leadership	Technology	Global	Data, ML, Privacy
B	Chief Data Officer	Publishing	Global	Data, ML, Privacy
C	Chief Executive	Maritime	Small	Data
D	Project Manager	Statistics	Government	Data, ML
E	Consultant	Data Provision	Small	Data, ML
F	Scientist	Identity	Medium	Data, ML
G	Researcher	Identity	University	Data, ML, Privacy
H	Researcher	Community	Small	Data, ML
J	Data Leadership	Non-profit	Large	Data, ML, Ethics

Table 1.1: Members of the Expert Review Panel

A set of slides was used to align discussion around particular themes (provided in Appendix B). The ordering of slides was not fixed, and the interviews used different slides and different sequences to maintain the flow of the discussion with each expert, as supported by the semi-structured interview format. Not all slides were shown to all participants, with content moderated depending on the progress of the interview and engagement shown by interviewees to particular subject matter, as well as the interviewee's background. In total, 368 minutes of interview video was recorded and automatically transcribed by Teams software. Following the interviews, the transcript files were exported from Teams and imported into Nvivo, where they were manually edited and corrected. Transcripts were manually coded in Nvivo, generating 42 different codes, which were subsequently grouped into 8 categories, which were largely aligned to the phase of the interview covered.

The ERP interviews were conducted following the completion of the design work presented through this thesis. In order to present results and knowledge arising from analysis of the ERP interviews in the appropriate context, findings from the interviews are presented as source data for evaluations of artefacts developed in Chapters 3 and 4,

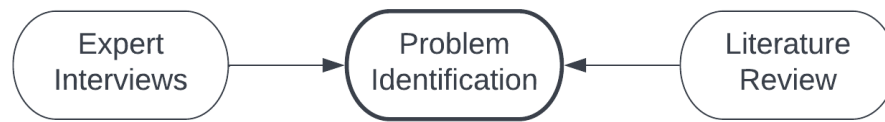


Figure 1.3: The role of expert interviews in problem identification, based on Offerman, et al. [116].

and as insight that contributes to the thesis conclusion, presented in Chapter 7.

An important objective of design science research is to develop solutions to “important and relevant business problems” [69]. To gain further insight on the importance and the relevance of the research questions, experiences in the problem area was sought from the Expert Review Panel members. Offerman, et al. [116], describe the role of interviews with practitioners and experts as an important contributor to problem identification and definition, as illustrated in Figure 1.3. During an early part of the semi-structured interviews, each member of the ERP was introduced to the research area of this thesis with a presentation of the slide shown in Figure 1.4, framing the problem in terms of being a “last mile challenge” [39] for DDS. Interviewees were subsequently asked if they had awareness or experience of the problem being presented. All nine of the experts said they were familiar with the problem with eight of interviewees providing examples, either directly from their own experiences or from situations that they had knowledge of. We provide results from the interviews below, as they add additional context to the problem area.

A data lead at a non-profit organisation, J, provided an example from the recruitment sector, which in J’s view had set up systems so that candidates similar to those already in place are selected, propagating privilege. J shared concerns about the ethical use of information in recruitment, and worried, based on their lived experience, that people were interested in such systems based on “hype”, yet did not have the cognitive capabilities to challenge the role of such systems, which were being delivered to them based

Introduction: Last Mile AI Challenge

- Data from many sources is used to build data-driven products (eg. AI/ML)
- Data is curated and aggregated, and experts build models and then products
- These AI/ML products go out in the field...
 - Used by “domain authorities” - users, not data scientists or AI engineers
 - How can practitioners, etc. check and monitor for ongoing suitability?
- Oversight is important in establishing confidence and trust in tools
- My research has been on providing oversight into multi-party data systems

“In the case of data-driven health care, children’s treatment or medication should not be based on adults’ data since this could cause unknown risks to children’s health”
UNICEF, Policy Guidance on AI for Children, 2020.

Figure 1.4: Introductory Slide presented to ERP interviewees

on what they had asked for, without “guardrails in place”. They worried that when problems arose from using these systems, there would be issues around who was seen as being accountable for making the decision to introduce the system, and whether there was proper support in place to help them. I also raised concerns about data sourcing, and people’s biases towards trusting or not trusting data. Their experience was that the further that people are removed from original data, the less perception they have on the accuracy and quality required from the data to do the job that is required of it, and the problem exacerbates as data is distributed or used by third-parties - as J phrased it, we lose the “ability to use our senses to smell that data” and “the further removed [the data] becomes, the less able we are to use some of those human senses that we would ordinarily use”.

Several interviewees provided examples relating to diversity issues, including race and gender. Participant A discussed media reporting of bias issues identified with datasets ‘Labeled Faces in the Wild’ [71] and ‘Penn Treebank’ [99]. B provided a personal example from the advertising industry, where they are developing a facial recognition system to help promote diversity, equality and inclusion in advertising media. They identified challenges they had found with datasets used to build the facial recognition ML model, and felt that when working with any particular dataset, it was very difficult

to know the risks inherent in that dataset. E described issues publicised in the media, where “facial recognition is still overwhelmingly going to put black people in the criminal field because the data sets at the beginning were drawn from a jail datasets.” and videos of “black hands coming under hand dryers, and the the hand dryer won’t turn on.”. G described similar cases. E also provided examples from their long history in the data industry, explaining that HIV medication was predominantly tested on white male patients, so there was very little data available on the impact medication would have on women. Similarly, the effects of medication on Asian men was unknown, as they tend to have smaller organs. E was concerned that “those sorts of problems haven’t changed with moves to AI”. F provided a hypothetical example of language models being trained on documents that were biased towards English language and written in California. Participant A commented on the UNICEF quote featured on the slide, pointing out that ethnicity and age should also be considered – “it’s not just children – you shouldn’t treat 80 year olds with data that was gathered from 14 year olds, for example”. A, F, and H identified a sparsity of data in medical fields. A stated “one of the biggest challenges is that there’s no good health data” and F pointed out that “the data that you do have will be much more influenced by the environment in which it’s been collected, compared with than the environment that might be deployed”. H felt that no one is comfortable turning over medical decisions to the output of an algorithm, yet reasoned that the use of DDS might have benefits in steering decision making away from human biases.

C uniquely categorised the challenge as a “black box versus gray box”, and related the problem to their experience with a system that provided navigation advice, and another that provided predictions based on vibrations detected by a sensor. They set the solution in the context of explainable AI, drawing on their experience of systems that try to provide an explanation for why a DDS provided a particular routing or prediction. C stated that “oversight in any kind of data driven system is of crucial importance”, but felt (based on their experience with rule-based maritime systems) that “many users don’t necessarily understand” what they are being told about a system.

Overall, a very strong feeling was conveyed by the ERP members that the problem area is seen as important in business and industry, and the majority of the interviewees provided descriptions of issues from their own personal experiences.

1.7 Published Papers

The research presented in this thesis has been developed through a number of published peer-reviewed research papers, which are listed below.

A paper presented at the ‘Workshop on Reviewable and Auditable Pervasive Systems’ describes the development of the framework outlined in Chapter 3, which results in Contribution C1.

1. Barclay, I., Abramson, W., 2021. Identifying Roles, Requirements and Responsibilities in Trustworthy AI Systems. In Adjunct Proceedings of the 2021 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2021 ACM International Symposium on Wearable Computers (UbiComp-ISWC '21 Adjunct), September 21-26, 2021, Virtual, USA. ACM, New York, NY, USA

A collection of papers describe the motivation to maintain a bill of materials record for the supply chain of contributions made to multi-stakeholder DDS, and the subsequent development of a bill of materials model for DDS, outlined in Chapter 4, in support of Contribution C2.

2. Barclay, I., Preece, A. and Taylor, I., 2018. Defining the collective intelligence supply chain. Presented at: AAAI FSS-18: Artificial Intelligence in Government and Public Sector, Arlington, VA, USA, 18-21 October 2018.

3. Barclay, I., Preece, A., Taylor, I. and Verma, D., 2019. A conceptual architecture for contractual data sharing in a decentralised environment. Presented at: SPIE Defense + Commercial Sensing, 2019, Baltimore, MD, United States, 15-17 April 2019. Published in: Pham, Tien ed. Proceedings Volume 11006, Artificial Intelligence and Machine Learning for Multi-Domain Operations Applications;. SPIE, 110060G. 10.1117/12.2518644
4. Barclay, I., Preece, A., Taylor, I. and Verma, D. 2019. Towards traceability in data ecosystems using a Bill of Materials model. Proceedings of the International Workshop on Science Gateways (IWSG), Ljubljana, Slovenia, 12-14 June 2019.
5. Barclay, I., Taylor, H., Preece, A., Taylor, I., Verma, D. and de Mel, G., 2020. A framework for fostering transparency in shared artificial intelligence models by increasing visibility of contributions. *Concurrency and Computation: Practice and Experience*, p.e6129.

Workshop papers and a published journal paper present research on the use of self-sovereign identity as a means of providing assurance on qualities and metadata of assets, outlined in Chapter 5, in support of Contribution C3.

6. Certifying Provenance of Scientific Datasets with Self-sovereign Identity and Verifiable Credentials, Iain Barclay, Swapna Radha, Alun Preece, Ian Taylor and Jarek Nabrzyski. In Proceedings of 12th International Workshop on Science Gateways (IWSG), 10th & 11th June 2020, Cardiff (Virtual).
7. Barclay, I., Simpkin, C., Bent, G., La Porta, T., Millar, D., Preece, A., Taylor, I. and Verma, D., 2020, November. Enabling discoverable trusted services for highly dynamic decentralized workflows. In *2020 IEEE/ACM Workflows in Support of Large-Scale Science (WORKS)* (pp. 41-48). IEEE.
8. Barclay, I., Simpkin, C., Bent, G., La Porta, T., Millar, D., Preece, A., Taylor, I. and Verma, D., 2022. Trustable service discovery for highly dynamic decentralized workflows. *Future Generation Computer Systems*, Volume 134, Pages

236-246. 10.1016/j.future.2022.03.035. (Note: Selected as an Editor's Choice paper, Fall 2022.)

A published journal paper describes the design and implementation of the AI Scrutineer system, outlined in Chapter 6, in support of the software architecture design presented as Contribution C4.

9. Barclay, I., Preece, A., Taylor, I., Radha, S.K. and Nabrzyski, J., 2021. Providing Assurance and Scrutability on Shared Data and Machine Learning Models with Verifiable Credentials. *Concurrency and Computation: Practice and Experience*; 10.1002/cpe.6997.

Further research which has informed the background work of this thesis through exploration of the core principles of SSI and decentralised, blockchain-based technologies has also been published.

10. Freytsis, M., Barclay, I., Radha, S.K., Czajka, A., Siwo, G.H., Taylor, I. and Bucher, S., 2021. Development of a mobile, self-sovereign identity approach for facility birth registration in Kenya. *Frontiers in Blockchain*, 4, p.2.
11. Barclay, I., Freytsis, M., Bucher, S., Radha, S., Preece, A. and Taylor, I., 2020. Towards a Modelling Framework for Self-Sovereign Identity Systems. *arXiv preprint arXiv:2009.04327*. (Note: Not peer-reviewed)
12. Barclay, I., Cooper, M., Preece, A., Rana, O. and Taylor, I., 2021. Tokenising behaviour change: optimising blockchain technology for sustainable transport interventions. *arXiv preprint arXiv:2104.01852*. (Note: Not peer-reviewed)
13. Barclay, I., Cooper, M., Hackel, J. and Perrin, P., 2021. Tokenizing behavior change: a pathway for the sustainable development goals. *Frontiers in Blockchain*, 4.

Chapter 2

Background and Problem Definition

2.1 Introduction

This chapter draws upon perspectives from published literature and policy to identify the research problem for this thesis. Foundational activities of the DSRM Framework [121] require identification of the problem and the motivations for solving the problem, which serve to define the objectives for a solution. Here we present a review and analysis of relevant literature, which identifies gaps in the provision of verifiable oversight, transparency and accountability in DDS. Subsequently we outline the requirements for a solution that can make a contribution towards addressing these gaps. These requirements motivate the design research that is presented in the remainder of this thesis.

The review of literature in Section 2.2 considers current and proposed approaches to presenting information on shared data assets and products derived from these data assets, such as ML models, and from systems which adopt these derived products as part of DDS. The importance of transparency and accountability in the adoption and reuse of digital assets from third parties is discussed, as DDS often make use of shared datasets and ML models, and parties adopting such assets need to have confidence in the integrity of those assets prior to their use, so that they can provide assurance to their own customers and users. To further establish the context for our research, Section 2.3 considers emerging policy directions from governments, NGOs and industry

bodies, who seek improvements in assurance provided to stakeholders across DDS. Having assessed the state-of-the-art in the literature review, and established the importance and relevance of the problem from policy, Section 2.5 discusses gaps found in current specifications and designs for providing verifiable oversight on DDS, and outlines requirements for a solution resulting from our design research. To meet these objectives, and to test the hypothesis of this thesis (Section 1.3), we propose to adopt the paradigm and design patterns, data models and protocols of self-sovereign identity, which are introduced in Section 2.4, establishing a technical constraint for the design work presented in the remainder of this thesis.

2.2 Literature Review

This literature review is presented in themes. First, we outline the role of information sharing in the adoption of digital assets from third parties. We then consider how information is currently shared by those responsible for assets, first as singular entities such as datasets, and then as aggregated assets and systems. Finally, we look outside of the data-driven systems domain, to identify how manufacturing and software industries provide documentation for systems with contributions from different parties, and maintain transparency and accountability. The body of literature in the review was identified by conducting searches of the Google Scholar database for keywords matching the broad themes of interest. Resultant papers were then assessed for relevance based initially on their titles and a subsequent review of their abstracts, with priority given to highly cited papers and recent work. Relevant and useful papers from the initial searches were taken as a starting set, and then a snowballing approach [75] was adopted, with citing papers and references reviewed for further promising papers, such that a body of related literature was assembled. Further literature has been identified in discussions on social media with relevance to the field, and has been added to the study set where appropriate.

2.2.1 Information Visibility, Transparency and Accountability

In order to gain confidence in the quality and suitability of a shared dataset, machine learning model, or data-driven system potential users need to be able to develop and maintain confidence in the originators of the asset under consideration, and any claims they make about the asset [34, 156]. In practice, even in environments where funding organisations insist on researchers adopting shared data, there can be a resistance to re-use – Pisani, et al., for example, found “lower-than-expected reuse of shared data may be because potential secondary users have few ways of checking the quality of those data” [124]. Historically, researchers have co-operated through informal groupings, identified as Communities of Practice (CoP) which provide and foster mutual credibility. Van House, et al. [164], determined that “one way that users judge whether work is to be trusted is to look at its source: is he or she a part of our community of practice? Can he or she be trusted to have used accepted methods to collect, analyse, and interpret the data? Do we speak the same language? Do they see the world the same way that we do?” As the scope and scale of asset sharing increases beyond established CoPs, effective data sharing environments need to be able to provide all participants with confidence that their fellow actors in the ecosystem are trustworthy, in order that data sharing can take place effectively [154].

McConaghy, et al. [100], argue that the uni-directional hyperlinked nature of the world wide web has resulted in a lack of opportunity for dialogue between the publisher and the consumer of digital assets, such that the consumer – here, a potential system or dataset user – is party only to the information made available at publication time. If the publisher chooses to share only a limited amount of information about an asset, then information asymmetry occurs almost by default, with the consumer unaware of any unreported information, such as any usage rights associated with the asset. McConaghy and colleagues assert that “information availability helps both initiate and inform action, thus impacting upon an individual’s decision making process” [100].

Information *availability* is one of the three core attributes of information visibility

provided by Stohl, et al. [148], the others being *approval* to disseminate information, and the *accessibility* of information to third parties. Transparency relates to the management of these attributes of visibility: information visibility and transparency are interdependent – without information visibility, organisations cannot be transparent, which depends upon visible information being used to make them open, and keep them accountable [152]. As such, making information on shared digital assets visible can help asset and system creators provide transparency, which “implies that third parties can clearly follow the chain of activity and decision making that led to a certain outcome” [148]. For DDS, Kroll [85] describes this traceability as an “enabling value” that exposes the “design choices made by system designers” making them available to stakeholders who are affected or interested in the system’s operation. Kroll observes that “traceability relates the objects of transparency (disclosures about a system or records created within that system) to the goals of accountability (holding the designers, developers, and operators of a computer system responsible for that system’s behaviours and ultimately assessing that the system reflects and upholds desired norms)” [85]. Novelli, et al. [115], describe accountability as “the expectation that designers, developers, and deployers will comply with standards and legislation to ensure the proper functioning of AIs during their life cycle”. This expectation can be tested through traceability, provided by transparency. Information visibility, derived from the constituent attributes of availability, approval and accessibility, is the foundation of transparency, traceability and accountability.

2.2.2 Documentation of Shared Data Assets

Wallis, et al. [166], describe the ‘long tail’ of research data, with small-scale projects, producing limited volumes of data that are typically only shared with trusted peers and colleagues, often as part of an informal gifting or bartering process. Now, niche datasets which may once have dwindled, have opportunities for increased adoption and new, wider audiences when they are used within data-driven products. However,

as these datasets are adopted and used in the creation of data-rich products, including ML models and DDS, sight of the original data source and knowledge of the data originators can be lost. As such, providing robust and trustworthy mechanisms to document and convey information about datasets, their origins and intended usage is becoming increasingly important.

Following the emergence of big data in scientific communities, schemes were developed to track the provenance, or history of changes, of data. Techniques employed included instrumentation of data workflows, through modified operating systems and large-scale software frameworks such as Hadoop. A W3C standard, PROV [105], was developed to support use cases ranging from documentation of information about who has responsibility for data, to complex descriptions of how data has been manipulated and combined. W3C PROV, as the standard became known, defined data models that enable the entities and actions within data systems to be modelled, so that a history of changes made to data items can be recorded. Subsequently, Groth [60] introduced the notion of a data supply chain, to provide better support for environments where data was being shared across organisations. Groth identified that a new approach was needed for emerging, multi-actor systems, with the recognition that “data supply chains are inherently distributed systems that extend across application and organizational boundaries” [60]. Groth envisaged that a traceable supply chain for data would lead to a situation where “data will have provenance as good as that of our coffee”. However, he expressed concerns about the complexity of procedures involved in providing such provenance, and the potential to overwhelm users with information conveyed by W3C PROV. Groth identified a need to develop abstractions that would provide insights into the production history of data, and proposed that improved mechanisms for communication were developed, suggesting that a “fair trade certificate for data” which gave a seal of approval to say that data was produced in a way “that we as data consumers think is correct” was needed.

More recent literature has outlined requirements and suggested approaches to docu-

menting data that is intended to be shared and adopted in other products, such as data being used in the training of ML models. Bender and Friedman’s “Data statements for NLP” [19] proposes a documentation structure which guides the formulation of a description of a dataset in order to provide context for researchers so that they can “understand how experimental results might generalize, how software might be appropriately deployed, and what biases might be reflected in systems built on the software.” In articulating contributions to dataset creation, Bender and Friedman recognise human roles in the data generation process, identifying the distinct roles of *annotator* – “Annotators may be crowdworkers or highly trained researchers, sometimes involved in the creation of the annotation guidelines” – and *curator* – “Curators are involved in the selection of which data to include”. Bender and Friedman also identify the *speaker*, which illustrates that different types of data sources will have different contributing entities that may need to be mapped and documented. Bender and Friedman propose that Data Statements should be included whenever new datasets are presented in publications, and with every system built from a dataset, in order to form a “chronology of system development” - a timeline which should include descriptions of the datasets used for model training, tuning and testing. An argument is made for providing two versions of a Data Statement, one which is detailed and published as a research paper or as part of system documentation, and a second more concise statement which is used when describing systems or experiments which make use of the data and which should be used alongside citation of the long-form statement. Data Statements provide a template for dataset creators to consider the context in which their data is used, such that they can expose motivations of the creation process and constraints in generation and use.

Along similar lines, Gebru, et al., have proposed Datasheets for Datasets [58] which take inspiration from industries such as electronics and manufacturing, where components are accompanied by a datasheet detailing the operating characteristics of the component, any test results, guidance on recommended usage, and other pertinent information for users. Gebru and colleagues eschew automation in the creation of the

datasheet, and identify the task of manually assembling the datasheet as providing an opportunity for researchers to reflect and perhaps alter how they create, distribute and plan to maintain their datasets. The proposed format provides an example series of open questions about the dataset, encouraging authors to avoid terse answers and to encourage the provision of rich information about the dataset. The datasheet format includes a section on Maintenance, which provides an opportunity for detailing how updates or obsolescence of the dataset will be communicated to secondary users.

The Dataset Nutrition Label designed by Holland, et al. [70], features a modular framework for presentation of information about datasets. Based on the well-established Nutrition Facts label scheme [140] from the food industry, and building on efforts to broaden this approach such as the Privacy Nutrition Label [80], the Dataset Nutrition Label shows a series of components or modules designed to present information on different aspects of the dataset to prospective users. Some modules will present non-technical information, whereas others can be highly technical, for example presenting machine-generated statistical information about the data in the dataset. The choice of modules presented in each case should be based on the availability of information, the level of willingness and effort volunteered to document the dataset, and awareness of privacy or confidentiality criteria around proprietary datasets. The intent of the system is to offer a flexible and adaptable framework which can be applied across different domains and data types, with an extensible collection of interactive qualitative and quantitative modules displaying their outputs in a standardised format. Holland and colleagues suggest that the Dataset Nutrition Label scheme will offer web-based tools for authoring and presentation to dataset users.

As our review on Information Visibility, Transparency and Accountability (Section 2.2.1) found, there is a well-recognised need to provide information about datasets to potential users, so that they can develop an understanding of the dataset from different perspectives, and develop necessary confidence in the qualities of the data and its providers. Information about datasets can include detailed provenance trails of the data

elements, following W3C PROV models, to well-considered user-facing documentation, that augments the dataset with information such as the context in which data has been generated and how it should be used. Each of the schemes reviewed here can make contributions to supplying valuable information about datasets, and helping potential adopting parties to build confidence. Whatever format dataset information takes, it must be offered within a framework that makes it available to parties that have a requirement for access, and it is within the bounds of confidentiality or privacy for which the dataset providers are able to grant their approval to share such information. The schemes reviewed here consider in great detail what should be shared, but do not consider how access to the information can be mediated.

2.2.3 Documentation of ML Models and Data-driven Systems

As data assets are adopted and used to create new products, which are themselves intended to be shared and used by others, researchers have identified requirements for documenting these new data-generated products. Primarily, the literature has focused on documentation approaches for ML models and AI Systems, yet the concepts can be applied to wider DDS systems.

Arnold, et al., propose FactSheets [6] which are based on a safety document called the Supplier's Declaration of Conformity (SDoC), used in industrial sectors including telecommunications and transportation. The SDoC is typically a voluntary document, which is developed and maintained by component or product suppliers, to provide written assurance of adherence to specified requirements. The FactSheet document is designed to consider systems as a whole, and is described as the type of document which might be delivered by a researcher with a shared ML model or DDS. Arnold and colleagues recognise that AI services are often offered to developers through programmable interfaces (API), and are integrations of many models trained on a variety of datasets. As such, developers are unlikely to directly use models or datasets, and their interface will be through the API offered by the AI service or DDS. The scope

of the FactSheet document is to “contain purpose, performance, safety, security, and provenance information to be completed by AI service providers for examination by consumers” and is positioned to provide information to fill a gap in expertise between the data scientists who created the AI service, and application developers or scientists who consume the service in order to provide applications for end-users. FactSheets are designed to cover aspects of an AI service relating to its development and use, detailing how the service was created and trained, which scenarios it was tested in, and how it should respond to untried scenarios, with the intent of grounding service use into particular usage domains. The document will also encourage coverage of security, in particular robustness of the service to adversarial attack, and safety. The holistic approach is proposed to enable a “functional perspective” to be taken on the overall service, and tests being conducted and documented for aspects of performance, safety and security which are not relevant for the individual components of the system, such as accuracy, explainability, and adversarial robustness. The FactSheets format encourages the attachment of datasheets or supporting information about contributing assets, to provide a route back to source information for developers trying to gain further insight and visibility into the makeup of AI systems.

Model Cards for Model Reporting [107], proposed by Mitchell, et al., are complementary to Datasheets for Datasets [58], discussed above (Section 2.2.2). Model Cards are intended to be short records providing details of the motivations for ML model development, recommendations for use and quantitative results of evaluation, in a document a page or two in length. Mitchell and colleagues suggest that a key role of the Model Card is to provide a documentation framework to promote a standardisation of “ethical practice and reporting” and to allow stakeholders to compare models “along the axes of ethical, inclusive, and fair considerations.” By providing information and metrics that capture and highlight bias, fairness and inclusion aspects of a shared ML model’s performance, and alerting model users to any potential pitfalls, Mitchell and colleagues hope to mitigate negative effects of model deployment. The structure for a Model Card is not rigidly defined, but it is recommended to include a section on the background to

the model’s development, detailing the model’s training and evaluation (and including reference to any Data Sheets or other documentation about the datasets used). The intended use of the model should be declared and described, so that readers can readily understand what the model has been designed to do, and in what contexts, as well as what it should not be used for. Mitchell, et al., reflect that Model Cards are just one approach to increasing transparency between developers, researchers, and stakeholders of ML models and systems, and that they are designed to be flexible, so that they can be used for a wide range of models across different use cases. As with all efforts to improve transparency through authored documentation, Mitchell, et al., identify that the usefulness and accuracy of a Model Card relies on the integrity of the creator of the card itself.

Building on the foundations of W3C PROV [105] in indexing systems to provide data provenance, Naja, et al. [110], introduce the concepts of accountability plans and accountability traces to DDS. Accountability plans are intended to represent the information that should be captured at different stages of a DDS life-cycle, whereas accountability traces are records representing the manifestation of those plans. Traces are intended to capture “structured information”, describing the outcomes of activities that can influence the accountability of the DDS – this might be the production of artefacts such as design specifications, or records of key decisions being made in a system’s development or deployment phases. Naja and colleagues describe the output of traces as potentially being similar in format to Model Cards [107].

In order to align DDS with approaches adopted in large-scale systems engineering projects and benefit from business protocols developed and lessons learned in these more established disciplines, Lavin, et al. [90], propose adopting Technology Readiness Levels (TRL). These metrics are widely used in organisations such as NASA, DARPA and Innovate UK to benchmark the maturity of technology. Lavin and colleagues suggest that “development and deployment of machine learning systems can be executed easily with modern tools, but the process is typically rushed and a means-

to-an-end”, showing a lack of diligence and process which “can lead to technical debt, scope creep and misaligned objectives, model misuse and failures, and expensive consequences.” Compared to systems engineering projects, which “follow well-defined processes and testing standards to streamline development for high-quality, reliable results”, Lavin and colleagues argue that DDS need to be documented systematically, so that stakeholders can be seen to be robust, responsible and reliable. TRL Record Cards are proposed, to collect information about a system as it progresses through the TRL stages (which the researchers have contextualised for ML systems, shown in Figure 2.1), including information on the models and their intended use, as well as results of a “gated review” process required for a system to graduate through the levels.

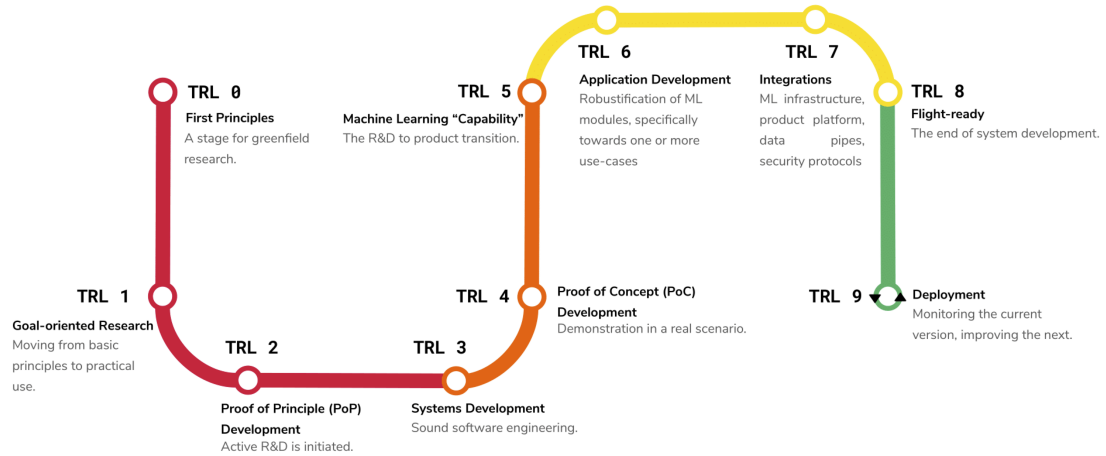


Figure 2.1: TRL Levels for DDS, from Lavin, et al. [90]

Historically, adoption of Commercial Off-the-shelf Software (COTS) in government and other organisations, led to concerns about a lack of oversight into systems developed by third parties [45, 93]. As with COTS, researchers, including Lavin and colleagues [90], are concerned that DDS practitioners will incur a technical debt [129] by adopting data and model components where there is limited transparency and understanding of the processes and dependencies involved in creation of the assets they come to rely upon [138]. In the UK financial services sector, for example, 24% of ML use cases surveyed in 2019 [78] were developed by third-party providers, with many of the systems developed internally also reporting use of off-the-shelf ML models or

services delivered by third party providers on an AIaaS [92] subscription basis. In clear recognition of potential problems that might occur as such systems are deployed, the United Kingdom's National Health Service explicitly points out the responsibilities of its organisations deploying DDS systems by asking stakeholders the direct question: "Can you develop a sufficiently robust understanding of relevant data feeds, flows and structures, such that if any changes occur to model data inputs, you can assess any potential impacts on model performance - or signpost questions to the vendor?" [77]. Recognition of the types of information required through the different stages of research, development and deployment as indicated by TRL stages, and guidance for the nature and depth of documentation of assets required at each stage may contribute to mitigating any problems.

The need to provide information about data-derived components and systems to provide confidence and assist in their proper usage is recognised by many researchers, with well-considered proposals for how such assets and systems might be documented. FactSheets and Model Cards observe the need to provide information on the underlying assets that contribute to DDS, as well as on the nuances of the derived systems. Mitchell, et al. [107], note that the documentation of systems depends on the integrity of its authors, and the work of Naja, et al. [110], in deriving traces of events that require accountability into a Model Card-style format, provides a foundation for automation of record-keeping. The TRL levels defined by Lavin and colleagues [90] call for structured record keeping and identification of accountable parties, which becomes more important as systems mature. Documentation formats reviewed here provide guidance and frameworks for the type of information that can be written and recorded and provided with systems, but do not offer mechanisms to tightly connect that information with parties that can take accountability for it. Being able to identify those parties, and develop a view on their integrity and expertise is key to being able to have confidence in any documentation they provide with their systems.

2.2.4 Supply Chain Traceability

In manufacturing industries it is standard practice to track products through their life-cycle from origin as raw materials, to component assembly, to finished goods in a store, with the relationships and information flows between suppliers and customers recorded and tracked using supply chain management processes [89]. In agri-food industries, traceability through the supply chain is necessary to give visibility from a product on a supermarket shelf, back to the farm and to the batch of foodstuff, as well as to other products in which the same batch has been used, and is critical in assuring food quality and safety [21]. In these industries, records of the components and sub-assemblies used in the production of the finished product are recorded in a document known as a Bill of Materials [76, 161]. The Bill of Materials model for recording contributions to asset development has recent adoption in the software engineering community, as software systems increasingly adopt libraries and components from multiple open source and commercial providers. A Software Bill of Materials (SBOM) is used to document the supply chain of dependencies, so that vulnerabilities can be readily and rapidly identified [48] and outdated sub-components fixed or replaced in updated versions of the module [96]. Tools which support SBOMs define schemas and formats for listing and tracking sub-components and enable SBOMs to be generated from the build processes of software tools, as part of the development and deployment pipeline. As well as documenting dependencies, SBOM formats also provide placeholders for supplementary information such as licenses for software components and libraries used. SBOMs are typically integrated into vulnerability tracking and component analysis systems, including Dependency Track¹, which provides notice of detected vulnerabilities in system sub-components from sources of publicly known vulnerabilities, such as those listed in the US Government's National Vulnerability Database². Carmody, et al. [35], describe the benefits this initiative has brought to a medical devices system, where critical infrastructure is now protected by providing warning on vulnerabilities

¹<https://dependencytrack.org>

²<https://nvd.nist.gov>

in previously hidden system components. The SBOM has been identified as a part of the future cybersecurity infrastructure in the United States, with President Biden giving an Executive Order³ that presents the motivation for its adoption as “a widely used, machine-readable SBOM format allows for greater benefits through automation and tool integration.”

2.2.5 Stakeholder Roles in DDS

In seeking to identify the parties who might require information about DDS, the field of Explainable AI [22] provides some insight into different stakeholder roles, here researchers and policymakers have sought to identify different roles with interests in receiving explanations of how an DDS has arrived at a decision. Tomsett, et al. [158], have identified six classes of explanation recipients: system creators, system operators, executors making a decision on the basis of system outputs, decision subjects affected by the executor’s decision, the data subjects whose personal data is used to train a system, and system examiners (e.g., auditors or ombudsmen). Building on this work, Preece, et al. [126], examined stakeholder needs for explanations and considered Users as one of four interested groups, alongside Theorists, Ethicists and Developers, and concluded that “the most influential of our four stakeholder communities is the users”.

Explainable AI covers a wide scope, and providing information and “audit trails” [26] on the parts and processes that have led to the generation of an ML or DDS model is recognised as a valuable component. Preece, et al. [126], identify “developers” as the “people concerned with building AI applications” – and include them as one of the roles having a need for receiving explanations about how a system operates. As such, it is apparent that both model developers and DDS creators occupy a position in which they have a need for explanations, but also have responsibilities to provide other parties with assurance and explanations about the qualities of their own work, and decisions

³<https://www.whitehouse.gov/briefing-room/presidential-actions/2021/05/12/executive-order-on-improving-the-nations-cybersecurity/>

they have made. Developers in turn are dependant on components that they use, with researchers showing concern for shortcomings in dataset documentation and dataset development practices - asking “how can AI systems be trusted when the processes that generate their development data are so poorly understood?” [72] – this can lead to problems propagating, or cascading up through the layers of a DDS [136]. There are published examples of DDS that have been subsequently discredited due to doubts being raised on the legitimacy of their data sources [131], or of societal bias being discovered in their training data after analysis [8]. Compounding this, MIT researchers [112], have found many labelling errors in published datasets⁴ that are widely used in ML model training and evaluation. Fortunately, development of ML models is becoming structured and operationalised, with the sequence of processes which lead to the development, testing and delivery of an ML model regarded as the “MLOps Pipeline” [130], based on the software engineering DevOps cycle. The MLOps pipeline involves a number of collaborating parties, from data engineers, data scientists and developers (model engineers, software engineers, backend engineers) [84], yet largely remains a model-centric view [97], stopping short of considering how models might be integrated into DDS for deployment – and subsequently managed and maintained when out in the field – the point at which Coiera finds “we face the reality that AI does not do anything on its own. It must be connected somehow to real-world processes, and its impact on those processes needs to be consequential. It is at this point that technology developed for its own sake quickly comes to grief.” [39]

2.3 Emerging Policy Directions

Globally, there is an increasing recognition from governments, NGOs, and intergovernmental organisations of a need to develop policy towards transparency and accountability on DDS, often identified in their context as AI Systems. The Organisation for

⁴<https://labelerrors.com>

Economic Co-operation and Development (OECD) have published a set of AI Principles⁵, which requires that “AI Actors should commit to transparency and responsible disclosure regarding AI systems.” Their use of the term “responsible disclosure” is prescient, as it originates in the field of cybersecurity [141], where it relates to sharing of information about known vulnerabilities in order to mitigate risk. Cybersecurity has been one motivation for providing oversight and accountability on DDS and the assets which contribute to their development has seen attention [150]. US President Biden issued Executive Order (EO) 14028 on ‘Improving the Nation’s Cybersecurity’⁶, which stated “In the end, the trust we place in our digital infrastructure should be proportional to how trustworthy and transparent that infrastructure is, and to the consequences we will incur if that trust is misplaced.”

Governments across the world are actively undertaking strategy reviews into DDS and the policies and infrastructure required to take full advantage of the benefits such systems can bring, whilst minimising negative impacts on citizens. The UK Government’s “National AI Strategy”⁷ declares an objective to “continue supporting the development of capabilities around trustworthiness, adoptability, and transparency of AI technologies”. The European Union (EU) AI policy⁸ identifies Trustworthy AI as a key focus area, and the Australian Government’s AI Action Plan⁹ “envision[s] Australia as a global leader in developing and adopting trusted, secure and responsible AI.” Concerns about consequences of poorly designed, implemented or managed DDS, and the role they play in critical prediction and decision making, are not limited to nation states as they can have a direct and indirect impact on all citizens – a report titled ‘The Case for Better Governance of Children’s Data: A Manifesto’ [32] from UNICEF, for example, states:

⁵<https://oecd.ai/ai-principles>

⁶<https://www.whitehouse.gov/briefing-room/presidential-actions/2021/05/12/executive-order-on-improving-the-nations-cybersecurity/>

⁷<https://www.gov.uk/government/publications/national-ai-strategy>

⁸<https://digital-strategy.ec.europa.eu/en/policies/european-approach-artificial-intelligence>

⁹<https://www.industry.gov.au/policies-and-initiatives/artificial-intelligence>

“The UN Committee on Digital Cooperation has warned against opaque algorithms where the underlying data and decision-making processes cannot be examined.”

As such, there is a strong motivation to undertake research into further understanding what is required for accountable and trustworthy DDS, and for designing Information Systems solutions that can support the provision of accountability, oversight and scrutiny in DDS.

2.4 Self-Sovereign Identity Protocols and Patterns

In Chapter 1 we identified that a decentralised approach based on emerging self-sovereign identity principles and design patterns offered a promising strategy that could be adopted in architectures designed to support accountability and provide assurance on qualities of digital assets. We do not propose to perform primary research on these technologies, but their usage underpins the solutions developed in this thesis and background information is provided here for completeness.

The term Self-Sovereign Identity [5] has been adopted to describe the ability of an individual to take ownership of their personal data. A goal of SSI is to couple ownership of personal data with methods to maintain control over access to that data, without the need for centralised infrastructure or authorisation being required by any third party. SSI has been the subject of research and ambition for several years, but has reached an inflection point in interest from industry and the research community as a result of the availability of distributed ledger and blockchain-based technologies [108], combined with an increased focus on individual’s data privacy as they interact with web-based and social networking services [157]. With the adoption of SSI, it is possible to secure a user’s personal data from unauthorised disclosure by allowing the individual to selectively provide elements of their data based on requests from a verifying party, and the value that the data owner places on that exchange. If the data holder considers it to be worthwhile sharing the requested data, they can provide consent to do so, but no

access is given to the data otherwise. As such, an individual gains the ability to decide how information about their identity and other personal data should be used and who has access to it.

SSI is decentralised, and is built upon well-established asymmetric cryptographic techniques whereby a user holds a private key and shares a public key [127]. The private key is used to sign documents, whilst the public key can be used by anybody with access to it to verify that the document has indeed been signed, and has not been tampered with. SSI uses a system which uses decentralised identifiers (DID) to identify parties involved, with the DIDs resolving to documents which explain, in machine-readable format, how to locate the public key needed to validate claims made about that DID, in the same way as web addresses resolve to provide web pages. The SSI research community has developed data models and protocols [147] that provide mechanisms for any party identified by a DID to issue cryptographically verifiable sets of credentials to any subject entity, also identified by a DID. In this way, a party which believes something to be true about another party can declare this in a standardised way, and sign this attestation using asymmetric cryptography techniques, based on the DIDs used being able to be resolved in order to validate the assertions made. This cryptographically signed document is known as a verifiable credential (VC), and will be held by the subject of the credential, or in the case of a child, or dataset or physical asset, by an authorised guardian or holder.

At a later date, when the holder seeks to enter into a transaction, a service provider may request proof of status or entitlements. The VC provides a means for this proof to be provided, as the holder of the credential can generate a Verifiable Presentation (VP) containing assertions from the VC document. By processing the VP, the Verifier can use the accessible public keys to check that (i) the presented proof pertains to the subject it is being presented on behalf of, (ii) the presented proof contains assertions signed by the original Issuer, and finally (iii) that the presented documents have not been tampered with. As such, triangles of trust [43] can be leveraged to enable parties

to issue, hold and verify credentials without reliance on any central authority, as shown in Figure 2.2. The use of asymmetric encryption and shared public keys in the VC protocols ensures that issued credentials are tamper-proof and that they have been signed by the identified issuer. If the verifier knows and trusts the identity and reputation of the issuer, then they can make a judgement on the value they place on the claims made by the issuer about the subject.

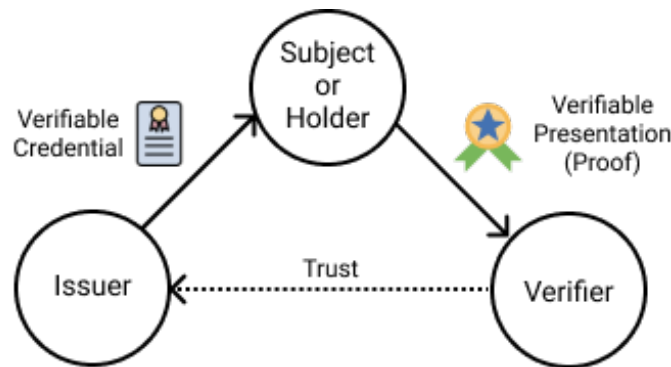


Figure 2.2: The Triangle of Trust

2.5 Gap Analysis

The review of literature and analysis of policy presented in this chapter identifies an ongoing recognition of a problem of lack of verifiable oversight in DDS, and demand from government, industry and academia alike for solutions that can provide transparency and accountability in DDS.

The literature offers a number of proposals for documentation styles and formats that seek to enable providers of DDS and digital assets of datasets and models to convey information about these assets to other parties. Table 2.1 provides a summary of the benefits and limitations of leading contributions. Some proposals, such as TRL Record Cards suggested by Lavin, et al. [90], stress the need for accountability being part of the documentation and require a signing off process through stages of deployment and development. Other, well regarded, proposals such as Datasheets [58] and Model

Cards [107] are more suggestive of the types of information that providers should or could provide, and offer themselves as a chance for creators to reflect on the asset they are offering. In the software industry, President Biden’s EO called for the SBOM structure to be adopted, and initiated a request for the details of a minimal SBOM to be researched and published. A motivation being that a shared schema would allow interoperable tools to be developed. Such a requirement is not yet in place for DDS, but policymakers around the world are aware of the issues caused by reliance on DDS, and a lack of transparency on the systems themselves, that has potential to cause harm if not used appropriately.

Describing a DDS in terms of the contributions to its supply chain could provide a way to identify and catalogue the models, data sources and other assets which contribute to the development of the components, or which are produced as the result of intermediate processes. Further, providing different stakeholders in an ecosystem with mechanisms both to certify and to validate or check the qualities of assets in the supply chain, and the credentials of the assets providers, offers a route to deliver trustworthy resources, based on reputations of scientists, engineers and other providers. Offering stakeholders oversight on these qualities, and the means to check and verify them serves to provide scrutability into systems across the “last mile” [33] of DDS deployment, with the potential to improve confidence in systems, and to ensure that deployed systems are appropriate for use. Explainable AI research provides insight into the different classes of stakeholders that might require information about how and why DDS behave as they do, yet we find a lack of research that delineates the roles of the system providers and developers, and that clearly identifies and expresses the dependencies that exist between these parties.

Following our literature review, and motivated by emerging policy directions, we identify a gap in a specification and design for a viable socio-technical approach to providing verifiable oversight, transparency and accountability across the ecosystems of contributors and users of a DDS. Succinctly, our review finds:

Approach	Overview
“Datasheets for Datasets” Gebru, et al. [58]	Framework provides focus on data assets, using answers to open questions. Not intended to be machine readable.
“The Dataset Nutrition Label” Holland, et al. [70]	Provides a modular framework for visual presentation of information about datasets. Not intended to be machine readable.
“Data statements for natural language processing: toward mitigating system bias and enabling better science.”, Bender and Friedman [19]	Framework guides the formulation of a description of a dataset to provide context. Strong focus towards metadata for natural language processing systems.
“FactSheets: Increasing trust in AI services through supplier’s declarations of conformity.”, Arnold, et al. [6]	Based on a safety document format used in industry. Intended to have datasheets, etc. available as attachments. Lacks mediated access to sensitive information.
“Model Cards for Model Reporting” Mitchell, et al. [107]	Framework can capture rich information about ML models, test conditions and intended uses. Not intended to be machine readable.
“A semantic framework to support AI system accountability and audit.” Naja, et al. [110]	Captures “structured information” describing the outcomes of activities that influence the accountability of a DDS. Lacks evidence of veracity of accountability claims.
“Technology Readiness Levels for machine learning systems” Lavin, et al. [90]	Promotes engineering discipline by defining Technology Readiness Levels for AI and ML, with a format for collecting metadata, but lacks systematic verification of claims.

Table 2.1: Summary of Approaches to Dataset and Model Documentation

- A need for transparency and accountability on DDS, identified by practitioners, policymakers and academics.
- Many proposals for documentation formats, which lack mechanisms to demonstrate accountability or provide verifiability, or an ability to be easily machine-read.
- A lack of proposals for viable technical solution designs for transparency, traceability, scrutability and accountability that takes into account complex multi-stakeholder nature of DDS.

As a result we identify a need to design Information Systems artefacts that can make a contribution towards meeting the problem requirements by addressing these shortcomings. Such artefacts can assist in understanding and communicating the structure of a DDS, and provide formats for developing machine readable documentation of the assets which comprise a DDS, which are designed to provide verifiable oversight, scrutability, transparency and accountability. Chiefly, we see an need to provide:

- Conceptual frameworks to help identify and record DDS composition, and roles, responsibilities and expectations of diverse stakeholders.
- Designs for IS architectures that can provide verifiable, accountable and appropriate levels of scrutability and transparency for diverse stakeholders in a DDS.

These requirements motivate the remainder of this thesis, which adopts the DSRM across Chapters 3 – 6 to design solutions which:

- *Guide identification of the roles and assets in a DDS*, through development of a conceptual framework.
- *Provide an interoperable, verifiable structure for describing DDS contents*, through proposal of a supply chain BOM record, and definition of a JSON Schema for a DDS BOM document.

- *Provide accountability on digital assets*, through a software architecture design, such that parties using assets can be assured of the qualities of the assets, and the party accountable for making such claims.
- *Provide verifiable oversight across a DDS*, through a software architecture design, facilitating scrutiny across a DDS and its constituents.

The technical designs that contribute to the research are based on SSI technology, which provides protocols and data models, along with nascent infrastructure, in support of the requirements for verifiability and accountability, and information security. A background to SSI is given above, in Section 2.4.

2.6 Summary

In this chapter we have presented a review of literature across a number of themes. We have considered the importance of information sharing in the adoption of shared assets, and presented work that provided information on singular assets such as datasets, as well as AI systems and DDS which might aggregate assets from different providers. We have looked to industry and software development to understand the importance of traceability and transparency in these domains, and understand how information in these domains is provided to stakeholders. We have also considered initiatives from governments and NGOs, that underpin concerns felt towards a need to provide transparency and accountability on DDS.

The literature and policy reviews identified a clear need for providing information transparency and accountability on DDS. We found that there are many proposals for documentation formats for data assets and DDS, but observed that these proposals lack mechanisms to provide verifiable oversight on systems, and to demonstrate ownership and take accountability. We also found a lack of proposals for technical solution architectures that can offer transparency, traceability, scrutability and accountability

in DDS. As such, we identified a need to design frameworks to help record contributions to DDS, and to identify the roles and responsibilities of participants in DDS ecosystems. We also identified a requirement to provide technical designs for software architectures that can provide oversight to diverse stakeholders in a DDS.

The self-sovereign identity approach for supporting trust in multi-actor ecosystems through the exchange of verifiable credential documentation, reviewed in this chapter, appears to offer mechanisms that can support provision of oversight and accountability on a DDS. Accordingly, to test the hypothesis presented in Section 1.3 that this is a viable approach, decentralised patterns using the protocols of SSI are adopted as a technical constraint for software architectures designed in this thesis.

A Framework for Roles and Boundaries in Data-driven Systems

3.1 Introduction

Data-driven systems are complex, multi-actor ecosystems, in which different parties are often required to contribute and collaborate in order to provide the required solution. By developing a framework that helps to identify the different stakeholders in a particular DDS ecosystem, we will be better equipped to understand the goals and motivations of different participants, and to determine the nature of interactions they need to have with each other. The research presented in this chapter enables us to address RQ1, “What are the roles involved in developing and using a DDS, and what are their responsibilities and requirements?”.

We design a conceptual framework for mapping the social structure of DDS. This Roles and Boundaries (RB) Framework provides a lens to help decompose DDS and identify the different roles, requirements and responsibilities of participants in ecosystems that contribute to the deployment and on-going maintenance of such systems.

The DSRM framework guiding our research, which was first introduced in Section 1.6, can lead to contributions of Information Systems artefacts of different types, including conceptual frameworks, through the steps of designing and demonstrating the artefact, and evaluating it through use within a context. The conceptual framework presented in

this chapter has been realised and subsequently refined over two design cycle iterations, which analyse the machine learning pipeline and develop an accompanying model expressed in the Unified Modelling Language [61](UML), as described in Section 3.3. The resulting research contributions are presented in Section 3.4, and evaluated in Section 3.5, prior to a summary which is offered in Section 3.6.

3.2 Problem Identification

The review of related work in Section 2.2 identified research concerned with the lineage of data in DDS, and the implications of errors in data cascading through systems, and the need to move towards adoption of engineering discipline through the DDS production lifecycle. A gap was found in identifying the participants in the DDS production lifecycle, and in enumerating requirements and responsibilities of each participant in ensuring a successful system is developed and maintained. This gap is problematic, as deployment and use of DDS and, in particular, systems categorised as AI systems, in real-world environments is beginning to highlight “last mile challenges” [39], amid concerns about the suitability of some systems and the digital assets they are derived from.

To contribute towards addressing this gap, we seek to develop an understanding of the general classes of stakeholder roles involved in developing and deploying a DDS, and to investigate interactions between the identified roles, the responsibilities of each role and the requirements and dependencies that each has on other roles. To provide a solution to the problem of a lack of clarity on the roles of stakeholders, and dependencies and motivations of different parties in a DDS, we seek to develop a conceptual framework which should meet the following requirements:

- *Assist in identification of different contributions and stakeholder roles in a DDS.*

- *Guide insight into responsibilities of each role, and requirements placed upon others.*
- *Facilitate exposure of tensions that might exist between roles, particularly where there are requirements for information sharing that might conflict with protection of commercial or private information.*

These requirements are enumerated in Table 3.1.

Requirement	Description	Design Cycle
R1	Assist in identification of roles and contributions in a DDS	1,2
R2	Guide insight into responsibilities and requirements	1,2
R3	Facilitate exposure of information sharing tensions	1,2

Table 3.1: Requirements of a Solution to the Problem

3.3 Design and Build

The research presented in this chapter is developed across two design cycles, which are described in turn. Design cycles are a feature of the DSRM and are iterative, with findings from the experience of building and demonstrating formative designs leading to new knowledge and insight that serves to improve both the understanding of the problem and the design of the solution in future cycles.

3.3.1 Design Cycle 1: The Machine Learning Pipeline

Definition of Objectives for a Solution

In seeking to gain further insight into last-mile challenges, and how they can be mitigated, we consider constructs that could support a party with responsibility in the

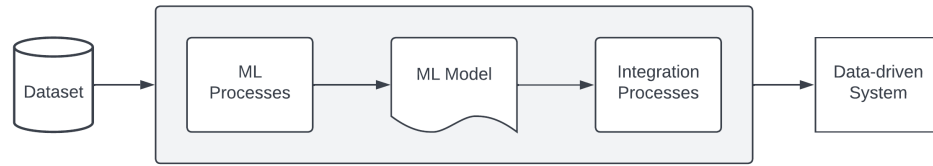


Figure 3.1: DDS production pipeline, adapted from Renggli, et al. [130]

deployment environment to seek and gain assurance that a DDS is appropriate for use within a given context. We identify this party as the Domain Authority (DA), which encompasses a role that might be adopted by a domain practitioner, a manager, inspector or other responsible party. What is considered to be appropriate will vary from case to case, and in some situations may involve regulatory authorities, best practice or simply good judgement from an experienced practitioner. In many scenarios a citizen, or their guardian, has no alternative than to put their faith in a DA to have performed some degree of due diligence and chosen their tools with care. This responsibility extends to the use of a DDS, which places an onus upon DAs to be able to provide citizens with assurance on the suitability of a system. To facilitate this, those in the domain must be able to develop a level of confidence in the supplier of the system, and in the suitability of the components and processes used to build the system.

The objective of the first design cycle is to inform an overview of the processes and participants involved in developing and deploying a DDS, and to begin to develop models and terminology that can help to explore the roles, responsibilities and requirements of the roles.

Design and Development

Design Cycle 1 considers a DDS as a deployable instance of a data-driven ML model, packaged within a user interface, to meet the requirements of a particular customer or end user. The production of the DDS includes processes which lead to the development of an ML model from one or more datasets, increasingly this chain of production

processes is referred to as the “MLOps Pipeline” [130]. ML models are then integrated with supporting interfaces and application logic through software development processes, with the aim of creating a usable, domain-specific DDS solution for deployment in the field. The production pipeline of a DDS yields the components and integration processes illustrated in Figure 3.1. We assign roles to the parties that undertake the processes in the pipeline and are responsible for the delivery of components, as outlined in Table 3.2.

Role	Task	Component
Data Provider	Generates and/or curates data	Dataset
Model Engineer	Uses data to develop abstractions	Model
Systems Integrator	Combines components to build a solution	DDS

Table 3.2: Roles in a DDS

We can ascribe characteristics, and identify responsibilities and requirements for each role:

Domain Authority

We identify the DA as the party with day-to-day responsibility for use of a DDS, deployed to provide analysis on a particular problem in a specialised domain. The DA, and related stakeholders¹, are unlikely to have AI or ML expertise, but have a duty of care to their clients, patients or customers - the people ultimately impacted by decisions or recommendations informed by the DDS.

Role Responsibilities

The DA has a responsibility to ensure that the DDS is appropriate for use in their domain. This may mean that it meets requirements set down in law, or rules set by regulators or professional bodies. The DA may be required to seek assurance

¹Domain Authority is used throughout, but this role is intended to include any party or stakeholder with responsibilities for the selection, procurement or application of the DDS in its domain environment.

that the underlying data and processes used in the development of the DDS are appropriate for use, to provide accountability for systems used.

Role Requirements

The DA is reliant on a Systems Integrator (SI) providing them with a reliable system, and the SI having acted with integrity and good judgement when selecting ML models or data abstractions for use in an DDS. In order to determine the suitability of the DDS for use in their environment the DA may have a need for some degree of visibility or transparency into the underlying components and processes that the SI used, so that they can form a judgement according to their principles. There is a further dependency on the Model Engineer (ME) providing sufficient, trustworthy information on the constituent components and design criteria of an abstraction, such as an ML model, to the SI, and, in turn on the ME having received reliable information from data providers, and being able to provide satisfactory evidence of this. This need for transparency is, however, potentially in conflict with other parties desires for confidentiality, due to protection of commercial information or in some cases legal requirements restricting provision of information on individuals, such as employees involved in system development or contributions to datasets.

Systems Integrator

The SI² is the role responsible for providing a DDS suitable for use by the DA. SI is a familiar role in other technical fields, and can be considered as “responsible for designing and integrating externally supplied product and service components into a system for an individual customer” [44]. The SI will typically provide an interface through which the DA can provide case data in the domain environment and where the results from the underlying modelling will be presented. This interface will often be an app or a web interface, but could use other technologies such as voice or physical

²Many aspects of this role include software engineering tasks, but the Systems Integrator moniker was chosen to highlight the need to integrate or combine many components to deliver the solution

sensors or actuators. The SI will select one or more ML models, either from within their own organisation or from an external supplier. The SI will have many other tasks to perform to deliver a reliable DDS. They may work directly with the DA to identify and serve agreed requirements, or deliver their DDS as a subscription service that can be found and used by any DA, without a direct relationship necessarily being in place between the SI and DA.

Role Responsibilities: The SI has a responsibility to ensure that the DDS they deliver is robust and reliable, and available when the DA needs it. There is a strong responsibility on the SI to choose suitable ML models, and to perform due diligence in this selection process. The SI also has a responsibility to use any adopted ML model within design parameters or in line with guidance set down by the ML model providers. In seeking assurance on the suitability of a system, the DA would in the first instance seek assurance from the SI on the nature of system's constituent components and their qualities. Furthermore, policy and legislative direction is towards requiring transparency and accountability on DDS – a task which is likely to become a responsibility of the SI, as the curators of the system.

Role Requirements: The degree to which an SI can determine the suitability of an ML model and the datasets used to develop the models they consider for adoption in DDS depends on information that is available about the model and the datasets. The SI will also have knowledge or a perception of the trustworthiness of the parties providing documentation and evidence in support of components. Such considerations can include a need to know the design parameters for the model, and the conditions under which the data was sourced. The required levels of trustworthiness can arise from knowledge of the party who created and shared the model, or may have to be delivered through documentation that is supplied with the model.

Whilst the SI may engage with the DA to provide assurances on the qualities of

their product, there is some information that they may be unwilling to readily share, or be unable to share. Such information might include details of specialised algorithms, or commercial relationships, where the SI may have motivation or contractual commitments to restrict information sharing.

Model Engineer The ME is the role responsible for developing, testing and publishing models or abstractions based on data. The ME selects data to use to develop and test the model. This data can come from a number of sources, within the ME's organisation, or sourced as secondary data from peers, commercially, or from open public sources, including Government or city authorities or global actors including UN or The World Bank [74].

Role Responsibilities: The ME has a responsibility to ensure that datasets used in the preparation of their models are appropriate, and will not cause unanticipated problems for other parties as they adopt and use ML models in development of DDS or in the use of such systems. This places an onus on the ME to take great care with the selection of datasets. Beyond selecting data, the ME has a responsibility to use the data correctly - that is, to use it in a manner that is consistent with the purposes for which the data was designed and shared, and in accordance with any terms and conditions set down in data licenses.

Role Requirements: The degree to which an ME can determine the suitability of datasets for use in models they create and publish depends on their perception of the qualities of the underlying data and the party providing it. Information of this nature about the contributing datasets is assessed by the ME from information provided by the data provider. As with the SI, there is a tension between calls for visibility or transparency into the contributing elements and processes of models coming from parties adopting or using the models, and a need for protection of information about commercially sensitive information belonging to the ME and their partners and stakeholders.

Data Provider The Data Provider (DP) role is considered to involve creation, preparation and publication of datasets for re-use, either by third parties³, or to internal customers within an organisation [95]. There is a dependency between the ME and the DP, in that the DP needs to prepare datasets such that the ME is willing and able to use the assets provided. Choices made by the DP in sourcing and preparing data have impacts that travel up the hierarchy, and can greatly impact models and DDS as ultimately perceived and used by DAs. Note that data governance [81] is an area that has received attention, and is where frameworks for addressing these responsibilities and requirements would lie.

Role Responsibilities: The DP has a significant responsibility to ensure that datasets are prepared with accuracy and integrity, and will not cause problems for other parties as they make use of the data, or products derived from the data. This places an onus on the DP to take care in data preparation, and in documenting their data. Sambasivan, et al. [136], provide examples of problems caused by “Data Cascades” in AI systems, as the effects of brittle or poor quality data ripples through to the domain. This is not to say that responsibility for avoiding such issues lies solely with the originating DP – there is a responsibility on other roles to make sure that adopted datasets are used sensitively, and within the bounds of which they are intended. The DP can facilitate this by preparing and documenting shared datasets accurately and purposefully.

Role Requirements: There is a tension between calls for transparency about data used in DDS and a desire for confidentiality or privacy, which can manifest in two ways. As we have seen above, there may be a research or business requirement from the DP to keep details of datasets and the processes of data generation and curation secret, in order to protect intellectual property. There are also significant potential implications surrounding privacy and traceability of individual data items within the dataset, especially when the data is about people.

³In some cases, the DP may offer data through an intermediary data broker [73]

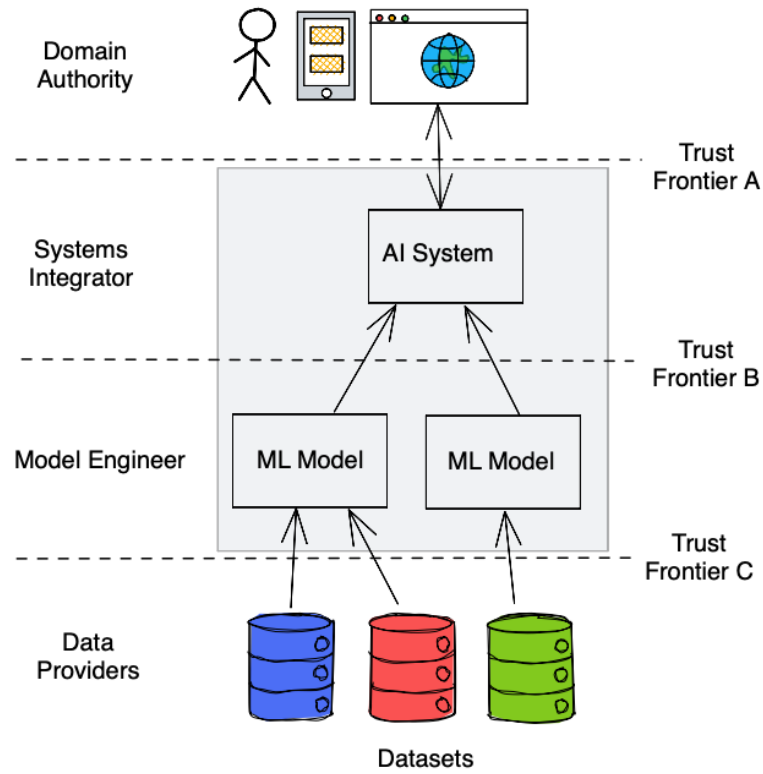


Figure 3.2: A formative sketch of the hierarchy of roles and contributions

Figure 3.2 re-orientates the production pipeline to a hierarchical view, serving to highlight dependencies between the parties involved and the components they provide to the overall system. Each of the illustrated components, such as datasets and ML models, can be used across multiple DDS in different combinations, and deployed in different domains. This hierarchy will be reproduced across every instance of a DDS, with some datasets or models being used in multiple systems. The boundaries between the roles discussed above, and shown in Figure 3.2, are identified as Trust Frontiers [66]. These boundaries denote the interfaces where information exchange is required between roles, with actors reliant upon the integrity of others in the hierarchy. The tension between the desire for transparency from some parties and other’s motivation to restrict information flow described for each role above, has been described by Trask, et al. [159], as “structured transparency”. Note that in some DDS deployments individual actors in the ecosystem will take on several of the identified roles, whilst in

other cases there could be no direct relationship between the parties or any knowledge of the other actors.

Demonstration and Evaluation

The demonstration phase of DSRM provides a motivation for the artefact to be instantiated during each design cycle. This supports formative evaluation [165] of the artefact, providing feedback to inform and improve performance [98] in later design iterations.

Case study scenarios are used to demonstrate and evaluate the artefacts developed through the design research of this thesis, and are described in Definition 1. Scenario S1 is an artificial concept for a DDS that has been envisioned by Preece, et al. [125] to be deployed to monitor NATO’s hypothetical Anglova urban setting [149]. This DDS is described as being owned and provided by partners in a military coalition, and procure and analyse multi-modal sensor data during times of situational uncertainty and rapidly-evolving circumstances. Military coalitions can be considered to behave as decentralised ecosystems, wherein different partners collaborate to deliver a solution to a problem in a fast-changing, high-stakes environment. Relationships between coalition members typically exhibit asymmetric power balances, and fragile trust. A further scenario, S2, is based upon a commercially-deployed conversational software agent (chatbot), named Aurora⁴, which is designed to provide families with advice on the care of newborn babies. The diversity of settings of the example scenarios is reflective of the differing natures of high-stakes settings in which DDS can be deployed.

Applying the framework for the hierarchy of roles developed in Design Cycle 1 (Figure 3.2) to each system, we can identify or infer the following roles (Table 3.3) from descriptions and further references in Preece, et al. [125], with ‘?’ used to indicate elements that remain unknown after initial analysis.

Whilst information is sparse, Table 3.3 immediately identifies the responsible SI for

⁴<http://auroratechai.com>

S1 CCTV Monitor: Conceived by Preece, et al. [125], as a demonstration of a setting in which data-driven systems are able to analyse multi-media data, and detect unexpected events that might require the attention of security forces. S1 is an artificial scenario in which a CCTV video stream processing service owned and operated by a local law enforcement agency (and described as being located outside a venue frequented by members of a minority community) detects what is believed to be an active shooter event via deep neural network (DNN) analysis of audio-visual data, running on an edge device. This CCTV Monitor scenario is representative of situations in which a domain authority is reliant on the output of a system created and operated by a third party systems integrator.

S2 Aurora: A commercially available web-based chatbot, that provides a conversational interface for parents and carers of newborns to ask for advice on any concerns they have about their babies, particularly related to feeding. Aurora offers conversation and advice in English and Portuguese. Aurora is intended to be sold as a business-to-business-to-consumer (B2B2C) service, to agencies such as healthcare authorities, for use by families when they have concerns about their children. Aurora has been developed by CM, a childcare expert, using a hosted AI service offered by Google. The Aurora scenario is representative of situations in which a commercial ML model has been used, and limited information on its characteristics is available. Aurora is designed to be deployed such that the systems integrator (the developer, CM) and the domain authority (the health board) are different parties, and the domain authority is reliant on the integrity and quality of work of the systems integrator.

Definition 1: Scenarios used in Demonstration and Evaluation

each DDS, and the providers of contributing assets where known. Relationships between actors in the system are also made apparent, with UK - UK perhaps providing situations where information might most freely be exchanged, and different relationships existing

Scenario System	DA	SI	ME	DP
S1 CCTV Monitor	UK Analyst	Anglova Law	?	?
S2 Aurora	Health Agency	CM	Google	CM

Table 3.3: Roles Identified from Scenario Descriptions

between UK - Anglova and UK - US. In Aurora, the use of a Google service to provide the model raises questions about how further information might be sought, and to what extent it would be forthcoming. There are also cases where a single party holds multiple roles, in Aurora, for example, CM is identified as the SI and the DP. Aurora is intended to be provided to families via health services, so we have identified the DA as a potential customer health agency – as stated previously, the DA could also be an inspector or an auditor.

3.3.2 Design Cycle 2: UML Modelling

Revision of Objectives

Design Cycle 1 has shown promise in guiding identification of the roles and relationships between parties in a DDS from available descriptions and information. We now seek to develop a formal model description and set of definitions, to provide a conceptual framework that can be used to analyse DDS and identify key parties and components.

Design and Development

Building on the work of Design Cycle 1, we can refine Figure 3.2, and develop a model in which relationships and dependencies between roles are described in terms of the Unified Modelling Language [61], a graphical language for visualizing and documenting the artefacts of software-intensive systems and associations between components

in such systems. Adopting UML allows us to document relationships between components in a DDS using a standardised graphical vocabulary.

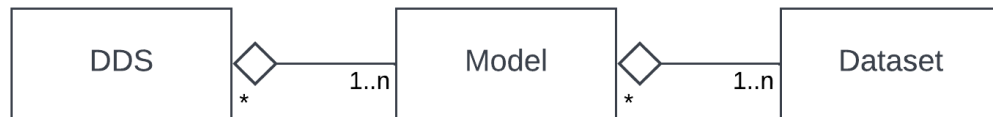


Figure 3.3: Relationships between components in a DDS. The white diamonds show aggregation.

Figure 3.3 shows a UML object diagram representing a system, with the label *1..n* indicating that the DDS needs one or more Models, and *** that a Model can be in 0, 1 or more DDS, and so on, as defined in Table 3.4. This provides a succinct representation that a dataset can be used in many models, and a model in many DDS.

Label	UML Meaning
1..n	Aggregation of one or more
*	Contained in 0, 1 or more

Table 3.4: UML Nomenclature

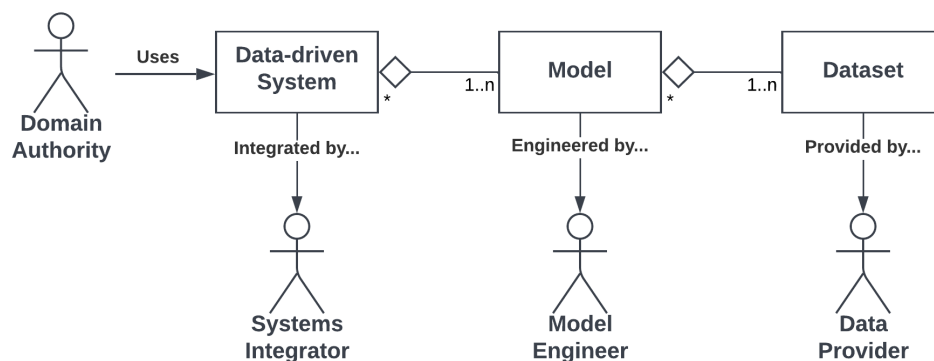


Figure 3.4: Revised Framework for DDS Analysis

The model can be extended by adding human actors and their relationships. A UML representation with key roles and connections in a DDS is illustrated in Figure 3.4.

Here, arrows are used to indicate dependencies – i.e. the DA is dependant on the DDS, whilst the DDS is dependant on the SI, and in turn on the parties responsible for the sub-components of Model and Dataset. Roles in the framework are based on the discussion above in Design Cycle 1, with definitions as in Table 3.5.

	Role	Definition
DP	Data Provider	Party that sources and provides data and datasets
ME	Model Engineer	Party that develops models and abstractions from data
SI	Systems Integrator	Party that provides usable systems from models
DA	Domain Authority	Party with responsibility for the deployed system in the field

Table 3.5: Roles in a DDS

The framework is intended to highlight the socio-technical nature of a DDS, and show the dependencies that the DA has on technical components and the human actors or organisations responsible for these components.

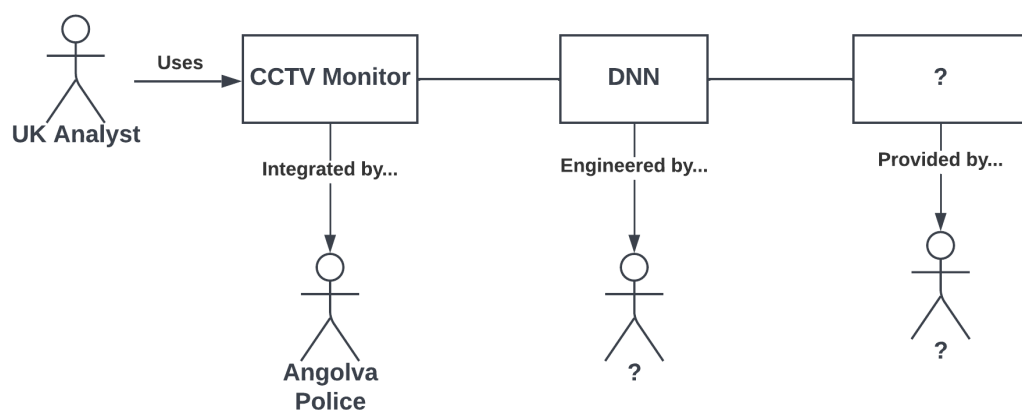


Figure 3.5: Framework Instantiated for the CCTV Monitor Scenario (S1)

Demonstration and Evaluation

To demonstrate the utility of the designed artefact, the framework is instantiated for the DDS scenarios described in Definition 1. Figure 3.5 shows a model of Scenario S1, the CCTV Monitor system. This identifies the UK Analyst as the DA, using the CCTV Monitor DDS provided by Angolva Law Enforcement, as documented. We also know from the system description that it uses a DNN to perform event detection on audio-visual data, whilst other details about contributions and data sources remain unknown on first inspection of provided documentation. We do not know anything of the nature of the DNN, how it has been trained, and by whom. This shows that there is limited oversight offered into the system. Operationally, the UK Analyst DA could make a decision as to whether this is an acceptable situation, which would depend on the nature of the DDS and the setting in which it is deployed, as well as the level of confidence that the DA has on the SI in adopting suitable components. Use of the modelling framework by the DA provides situational awareness, and facilitates further investigation to understand system composition.

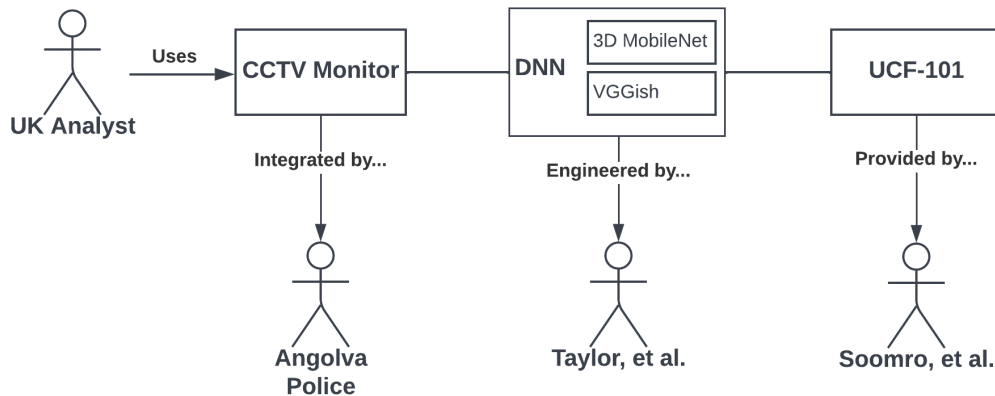


Figure 3.6: Second Iteration of Framework Analysis on the CCTV Monitor Scenario (S1).

Returning to system documentation, in this case the scenario descriptions of Preece and colleagues [125], more can be learned about the system by following references.

In this case, we learn that the model component is from Taylor, et al. [151], and known as Visual-Audio Discriminative Relevance (VADR). We further learn, that the “model architecture comprising a 3D MobileNet and VGGish as feature extractors for the video and audio input respectively.” and that “the bottleneck features of both sub-networks are concatenated and fed forward to a classification layer with 51 logits to be trained”. The training data is UCF-101 human activity recognition dataset [146] – but only samples that have an accompanying soundtrack – a subset of 51 classes. Using this further information, we can revisit the proposed framework, and produce a more complete mapping as shown in Figure 3.6.

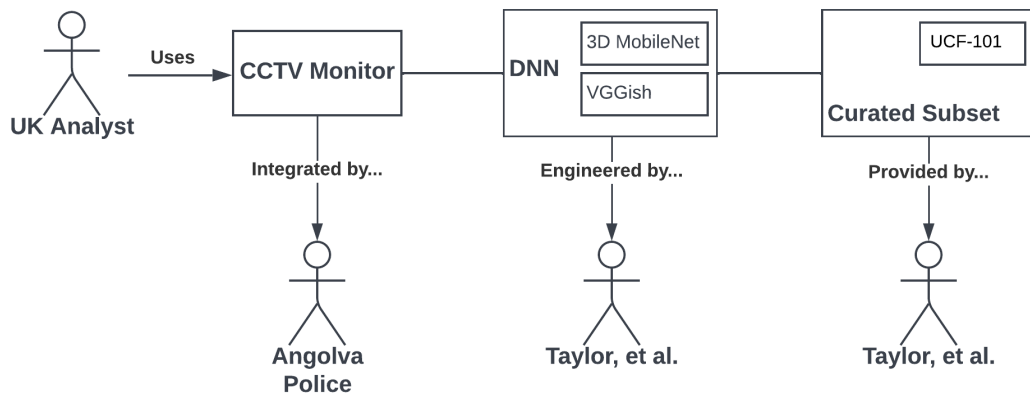


Figure 3.7: Revised Second Iteration on the CCTV Monitor Scenario (S1)

Further reflection leads us to re-work the analysis once more, in order to reflect that Taylor and colleagues documented that they used a curated subset of the UCF-101 data. This curation process puts Taylor, et al into the role of DP, in that they were taking responsibility for curating the data through publication of this information in their research. As such, they are a more plausible DP for this DDS than Soomro, et al [146], and so the framework mapping has been redrawn in Figure 3.7.

Considering a second scenario from Definition 1, S2, the Aurora assistant. The framework can be used to analyse the system with information provided in discussion with CM, the SI of the DDS. The analysis, shown in Figure 3.8, highlights a dependency on the Google ‘DialogFlow’ service, along with two models which are part of the Google

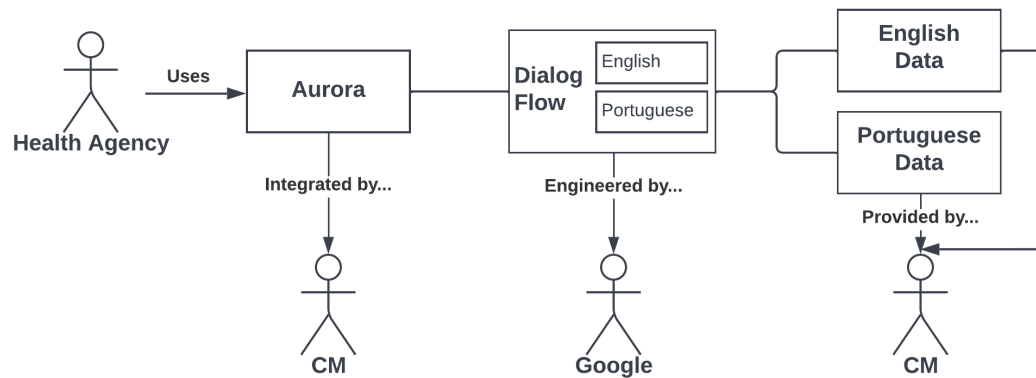


Figure 3.8: Analysis of the Aurora Scenario (S2)

service and provide support for interpretation of text in English and Portuguese. These models are trained on language specific data, which has been created and provided by CM.

Reflecting on the application of the framework to the scenarios, it was noted that the UML pattern defined in Figure 3.4 did not allow for the nested models and datasets shown in Figure 3.7. The UML model had provided for use of multiple models or datasets, but not provide a means to express a single model being an aggregation of other models, or a dataset having a curation process that resulted in the dataset being used being a subset (or superset) of another dataset. Analysis of the S1 CCTV scenario has produced evidence that these are plausible constructs, and there is utility in extending the framework to support them. This finding leads us to extend the framework to allow it to represent models which contain other models, and datasets which are reliant on other datasets. A revised model, which uses recursive UML composition symbols, is described in Section 3.4.

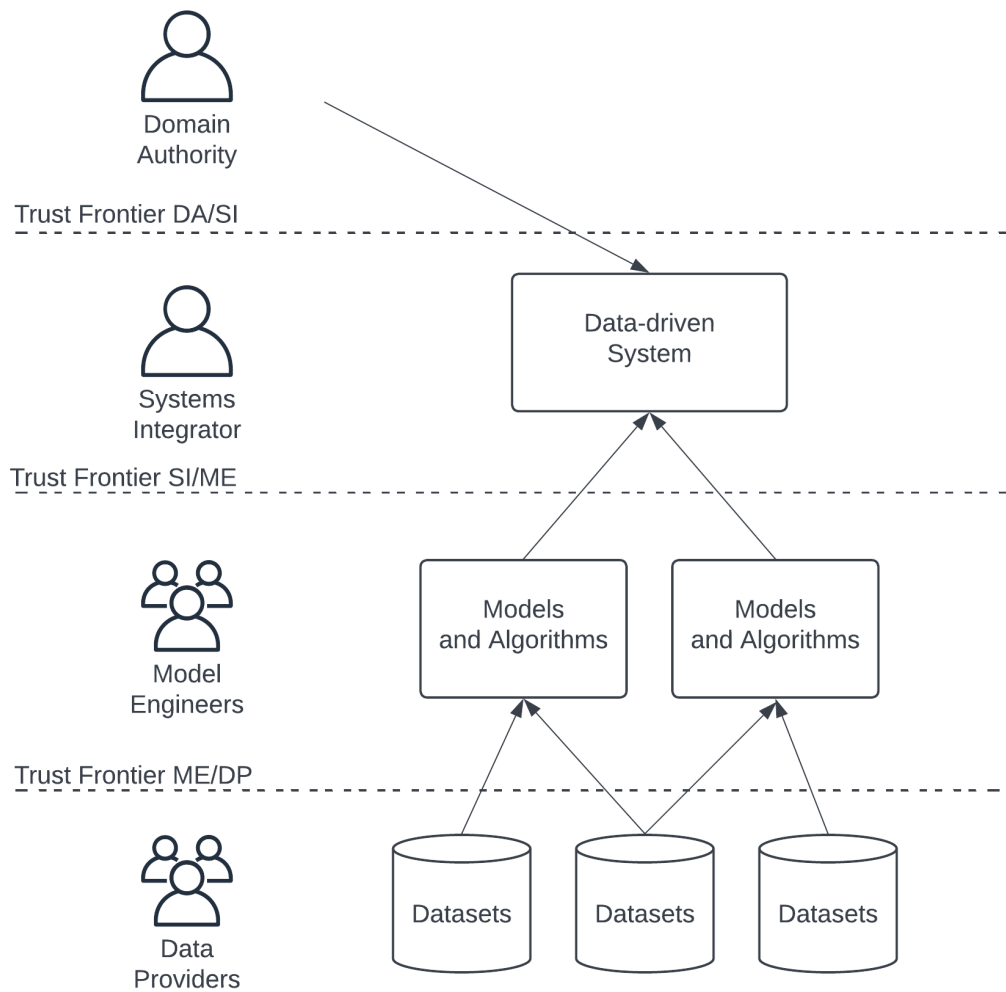


Figure 3.9: The Roles and Boundaries Framework

3.4 Research Outputs

3.4.1 The Roles and Boundaries Framework

As a result of iteration through the design cycles presented in this chapter a model that can be used to represent the roles and relationships in a DDS has been developed and refined. The resulting framework, which we name the Roles and Boundaries Framework, is shown in Figure 3.9 and an accompanying model in Figure 3.10.

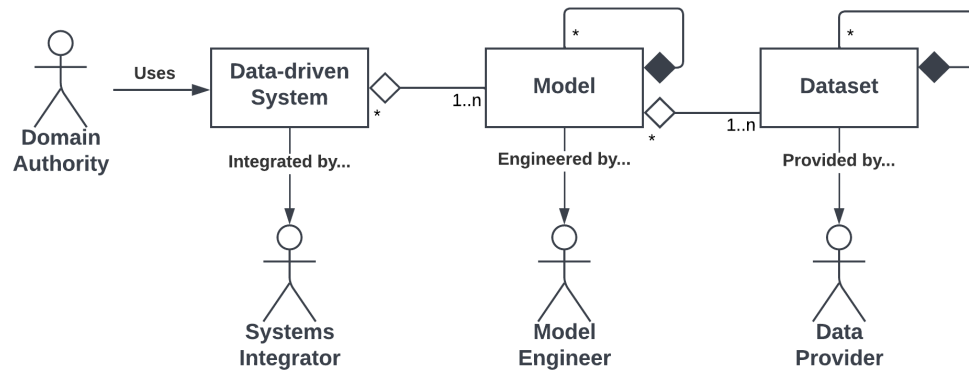


Figure 3.10: DDS Representation in UML. White diamonds show aggregation, black diamonds show composition.

The model uses UML symbols to describe the composition of a DDS, and to identify the roles responsible for each component. The model is to be interpreted as described in Table 3.6 which follows.

Entity	Description
Domain Authority	The DA is dependant on the DDS in their use case.
Systems Integrator	The DDS is dependant on an SI for its integration.
Model Engineer	The Model is dependant on an ME for its engineering.
Data Provider	The Dataset is dependant on a DP for its provisioning.
DDS	A DDS is an aggregation of one or more Models
Model	A Model is part of 0 or more DDS. A Model can be composed of other Models.
Dataset	A Dataset is part of 0 or more Models. A Dataset can be composed of other Datasets.

Table 3.6: Entities in a DDS Ecosystem

Technological Rules

Engström, et al, [52] reviewed software engineering literature through the lens of design science, and found utility in presenting a “technological rule”, building on Aken’s [3] definition as “a chunk of general knowledge, linking an intervention or artefact with a desired outcome or performance in a certain field of application.” The intent of the technological rule is to help to communicate the core of the research contribution to peer academics and industry.

Adopting this approach, technological rules can be developed in support of the research output of this chapter, which are provided as Definition 2

- A data-driven system has the property of *scrutability* if a stakeholder can identify the constituent components in the system, and inspect the qualities of those components, to the level that their authorisation determines.
- A data-driven system provides *verifiable oversight* if a stakeholder can verify the integrity of claims made about system composition and qualities of component assets.
- A data-driven system has the property of *accountability* if it can be demonstrated that claims being made about the system, or any assets in the system, were cryptographically signed by a private key controlled by an identifiable participant or role.

Definition 2: Technological Rules for a Data-driven System

3.4.2 Communication

Communication, which requires knowledge of the “disciplinary culture” [169] is considered an important aspect of the DSRM approach [122]. Communication of the research leading to the design of the RB Framework has taken place throughout the

design cycles described in this chapter. Following internal review, and discussion within the research group, a paper was written and submitted to Workshop on Reviewable and Auditable Pervasive Systems (WRAPS)⁵. The paper was published as grey literature, and the initial model (from Design Cycle 1) shared on social media with requests for comment from a professional audience providing useful feedback. Peer review of the WRAPS submission was used to publish a revised paper [10], and the updated work was presented to a virtual workshop audience as part of UbiComp 2021. The framework was presented to industry practitioners from the UK's Office of National Statistics during their weekly research seminar, and to a meeting of the Trust over IP Foundation (TOIP) AI and Metaverse Technology Task Force.

3.5 Evaluation

March and Smith [98] describe “Build and Evaluate” as design science research activities that are aimed at improving the performance of a design. The demonstration activity described here has seen the RB Framework built and revised through the design cycles described above, with each iteration given formative evaluation through its application to case study scenarios. Peffers, et al. [120], have developed a taxonomy of design science research artefacts, and a taxonomy of artefact evaluation method types. The artefact resulting from the design research presented in this chapter is a framework. In their research, Peffers and colleagues found that 4 of 9 published evaluations of frameworks as design artefacts had adopted illustrative scenarios, as used through the design cycle iterations leading to the development of the RB Framework. In these formative evaluations [165], scenarios were used in an artificial [165] paradigm to determine the efficacy of the model, and to inform design improvements toward the final outcome. We have used formative evaluation in this way to assess and improve the artefact through the design cycles described above, resulting in the research contribution, presented in Section 3.4.

⁵<https://wraps-workshop.github.io>

Further evaluation of the research contribution is framed in the context of Hevner's 3 cycle view of DSR [68], which was introduced in Section 1.6, and provides a summative evaluation of the outputs of the overall Design Cycle, presented in Section 3.3. Evaluation is conducted from the viewpoints of Hevner's Rigour Cycle (the grounding of the work in science) and Relevance Cycle (its positioning in the practice).

3.5.1 The Rigour Cycle: Expert Review Panel

In a summative, criteria-based evaluation [165], the final artefact resulting from the Design Cycle iterations is appraised. A series of semi-structured interviews were conducted with participants from the ERP (introduced in Section 1.6.2, and composed of peers from industry, governments and academia, with varying levels of expertise and experience across DDS, data and ML). Interviews were undertaken on conclusion of the design work, with the results contributing to refinement of the created artefacts presented in Section 3.4. A slide introducing the RB Framework (Figure 3.11) was shown to interviewees, and verbally introduced. The 'sketch' format of the framework was used to illustrate and describe the framework, The UML diagram was also on the slide, but was only discussed with interviewees who presented themselves as more technically minded. Following the introduction to the framework, a discussion with participants sought to ascertain if they understood its purpose and function, and to gauge their opinion on aspects of the framework. Each discussion was recorded and transcribed, and subsequent analysis of the transcripts used to group responses within evaluation criteria, based on the "5E's" [37] - Efficacy, Efficiency, Effectiveness, Ethicality and Elegance (Definition 3).

Efficacy

Do subjects understand the framework and its purpose?

Efficacy considers whether the proposed solution works. In evaluation of the RB

Roles and Boundaries

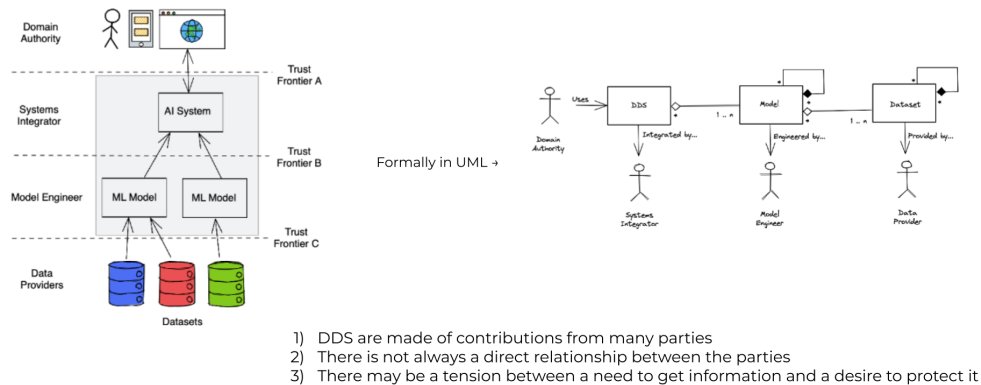


Figure 3.11: Roles and Boundaries Slide presented to ERP interviewees

Efficacy: Does the proposed evaluand work?

Do subjects understand the framework and its purpose?

Efficiency: Is resource use minimised?

Can subjects align their DDS structures with the framework?

Effectiveness: Does the proposal help to attain the long term goals?

Does the framework make elements of DDS more apparent and visible?

Ethicality: Is the proposal a moral thing to do?

Is it appropriate and acceptable to identify responsible parties?

Elegance: Is the proposal able to be performed aesthetically?

Can subjects express an appreciation of trust frontiers?

Definition 3: Evaluation Criteria formulated as the “5E’s”

Framework with the ERP, we consider the participants early impressions of the proposal, and whether they felt it would be of use.

Eight of the nine experts provided positive first impressions to the framework, describing it variously as “very pertinent”, “totally true” and “making sense”. Expert H, who uses ML and data in city communities was especially enthusiastic, “That’s so interest-

ing. I've never seen it labelled in quite that way.” Expert A, a senior researcher from industry, was the only subject who was initially sceptical. They later identified the role of the framework very succinctly, identifying a “kind of trust which is about the correctness of the system” and stating that “this diagram makes sense”.

The experts demonstrated understanding that the identified roles might be applied differently in different scenarios, A stating “even if it turns out that some of these layers are actually the same people, they’re sensible kinds of boundaries”. F, an experienced ML engineer described the framework as “a pretty good description of reality”, and identified that the “roles could be the same team even, depending on the size of the system.” B, an executive with a publishing and media firm, described the framework from a different perspective, pointing out that it could be used to describe systems which included contributions from external providers: “you may be purchasing a dataset and may not actually have any relationship with the dataset provider, so these things aren’t necessarily part of an organisation - you know there are third parties involved. So you could end up potentially having three different parties here.”

ERP members reacted favourably to the framework, and were able to express how it might be applied in different scenarios.

Efficiency

Can subjects align their DDS structures with the framework?

The Efficiency criterion is motivated by consideration of the ERP members reflections on the applicability of the framework to their environments, or environments that they have knowledge of.

Two of the experts expressed difficulty in visualising how the framework might be applied to their own work. C, a provider of data and business information systems to customers in the marine industry, was able to identify their customers as “domain authority people” and themselves as a data provider. However, they felt that most of

their customers lacked systems integrators and model engineers, because the DDS currently used in maritime are “quite simple rule-based operational tools.” D, an engineer in government, also struggled to fit their own work - developing ML for research - into the framework, however they said that they could see how it might apply in other scenarios, and was able to identify the roles for a hypothetical system they introduced.

Others were able to apply the framework to their own environments. J, responsible for data within a national, non-profit organisation works with people across the roles, and was able to speak of lived-experience of problems cascading through systems [136], as the behaviours of an individual in a data provider role become inherited by the system, “Things get bolted onto the top of it, added onto the side of it. Things get taken out and that isn’t always fully documented, and then that becomes a legacy” and resultant difficulties faced by a domain authority “who wants to trace something back”. G, an academic researcher with industrial software development experience, was able to map the framework to their view of software dependencies - “In any software development, you know when you pull in a dependency, you are trusting that that code is well written”. They posed the question “How do these system integrators get good information about the model to make the decision? We need to have information that we can trust about the things that we’re trying to make a decision about.” They further explained “The domain authority just has to trust what they’ve been given. They probably don’t even know that it’s made up of all these different components, they just say to the systems integrator, ‘Provide me something that that works’ and they wouldn’t realise that there are ML models that somebody else provided.”

The majority of the ERP members were able to understand and identify how the framework could be applied to their DDS, or systems that they could envisage.

Effectiveness

Does the framework make elements of DDS more apparent and visible?

Consideration of the effectiveness of the framework is presented in the context of topics

of conversation about components of DDS that were initiated by participants.

In the discussion, three experts noted that they felt the framework would have particular application where “foundational models” or models coming from third parties were used in a DDS. Community researcher H identified “where I think it has a particular application, is around the use of foundational models ... it essentially makes up a chunk of that middle slice that you’re using and has been trained on data which you don’t know anything about.”. H referenced the UML model on the slide, describing model constitution as “almost multidimensional in the sense that datasets will go into a model, but you might have another model with another originating set of datasets attached to it. It’s almost like turning that diagram you’ve got on the left hand side [the hierarchical framework] into that more sort of 3D one on the right hand side [the UML diagram]”. D also brought up models coming from third parties: “I think we’re going cloud-based now – there’s lots of excitement around having pre-trained ML models from Google or Amazon, and they’re seen as off the shelf service that anyone can come in and use.” They identified an issue with knowing what data those models are actually trained on, and asked “Are they relatable to your problem? You might be trying to classify something that’s completely different to what it was trained on in terms of number of features, type of data, etc.”. G understood that where there is a dependency on a model, it is “transferred to the engineer who’s built the model” and “if I don’t feel I can trust them, then to verify the model myself I need to look at all the surrounding context in which that model was produced”. They felt that there would be a tendency among Systems Integrators to rely upon the quality of third-party models based on their origin, “‘oh, it’s from Google and must be good’ - I imagine they do that all the time!”.

Discussions also covered data, and the framework prompted participants to share experiences and concerns around data coming from various sources. Participants tended to be sceptical about shared datasets – D mentioned “Wild West data” and shared an opinion that “smaller organizations would be using data that they could just get from

anywhere” whilst A expressed their view that “a lot of standard datasets are extremely weird”. B and F spoke particularly about health data (citing neuroscience and radiography), and how small datasets in this field were widely used – and very dependant on the environment in which they had been collected, which might affect utility in where models using them were deployed. They felt that being able to identify the use of particular datasets in a system “would be extremely useful and extremely important”. B explained that “what typically happens is you get a dataset and you don’t really know the risks inherent in that dataset.” and that when problems occur it is not clear whether the problem is with the model or the dataset. They felt that “if the dataset came with a with a set of oversight attached to it and accountabilities, I think it might help when you then develop in the model.”

The framework was found to be effective in generating a range of discussion topics with members of the ERP, highlighting different areas of interest and concern that might need to be taken into consideration for a DDS.

Ethicality

Is it appropriate and acceptable to identify responsible parties?

The RB Framework provides an opportunity to identify contributors to DDS. It is useful to consider whether this is ethical.

H, a community researcher with a strong interest in decentralised systems, was very passionate that individuals should be named, drawing an analogy with being named as an author on academic papers. B, D, and F wondered what might happen if a named individual left their employment, and how that might be resolved within an organisation. Other experts in the ERP felt that assets such as models and datasets were the result of work from several members of a team, or from multiple teams, and that they would not be comfortable putting their name to such work as an accountable party. Some roles within an organisation that might be expected to take responsibility were identified, and included product owner, chief technology officer, chief data officer, de-

partment head. Other experts identified that an organisation might be identified, but did not specify the role that might be responsible. J, a data lead, felt that insight might be gained from processes in place for responses to legal processes, such as Freedom of Information requests, where an individual would be accountable but with caveats defining what they were taking accountability for. This section of the interviews resulted in thoughtful discussion, with a diversity of opinion offered, depending on the experts' backgrounds and positions.

The experts on the ERP had mixed views on whether it was appropriate or viable for named individuals to be identified in the framework, or whether a role might be more appropriate. The RB Framework does not specify which is required, but provides a mechanism for discussion.

Elegance

Can subjects express an appreciation of trust frontiers?

The reaction of the ERP members to the novel depiction of trust frontiers between the roles in the RB Framework is used to evaluate the elegance of the framework.

Separation of the roles, and labelling of the boundaries between the roles as “trust frontiers” motivated discussion. H was very keen – “I like the different trust frontiers, I run into that a lot.” and suggested they might use the terminology themselves in future. They were able to express the challenge of the boundaries: “is the data good, or is your model good at synthesising this, or did curation reach a certain level of quality” and felt the trust frontiers were “a really clear way to delineate that”. F also adopted the terminology, and explained how it the framework might help when assessing a system, having to traverse frontiers because “they may have regulations that say that they can’t procure systems from companies that haven’t done their due diligence on the datasets”. Five experts on the ERP used the term “governance” as one aspect of the type of reliance that might exist across a trust frontier, with four of these participants also mentioning “impact” and four ‘ethics’. Expert E, in particular, was very passionate about these

considerations, and keen that a well documented “ethical review process” and “impact and risk assessments” were part of the information made available to parties in a DDS.

The ERP experts were able to identify characteristics of contributions that might need to be provided through trust frontiers, most typically from providers of assets such as data and models to SIs and DAs. Where experts mentioned specific documents or policy requirements, these could be generalised to fit into the framework.

3.5.2 The Relevance Cycle

An evaluation of the relevance of the artefact is focused on considering the utility that it contributes to its environment, and in particular how well it operates as a solution to the problems from practice identified earlier in Problem Identification, Section 3.2.

In its application to case study scenarios, the RB Framework and UML model has been shown to generalise, and to provide a mechanism that can be used to analyse what is known (and perhaps more importantly, not known) about the components of a DDS and the parties responsible for them, satisfying requirements identified in Table 3.1.

Demonstration activities based on artificial scenarios for each design cycle have provided a focus for technical risk and efficacy evaluation of the approach [165]. Discussion of how each requirement is met through the RB Framework follows.

R1 Assist in identification of roles in a DDS

The RB Framework defines contributions of a DDS and associates a role as an owner of each component. By applying the framework to a DDS implementation, parties and actors who assume each role can be identified from document or enquiry.

Design Cycle 1 built upon existing models of ML pipelines to extract key components and subsystems of a DDS. Roles have been identified as having responsibility for each. Instantiating the model for a DDS puts names of individuals or organisations into each roles.

R2 Guide insight into responsibilities and requirements

The RB Framework allows a DDS to be decomposed into models and datasets, which can in turn be derived from other models and datasets. Identifying these components for a DDS shows where responsibilities lie – which components have been developed locally for a project, and which have been used from public sources. This mapping shows where models have been combined and data has been curated, and assigns ownership and accountability to these actions.

Design Cycle 2 developed a UML model, which was applied to case study scenarios, and further refined for the final presentation in Figure 3.10 to support composite models and datasets.

R3 Facilitate exposure of information sharing tensions

Applying the RB Framework to a DDS helps to identify the parties that are responsible for each component of the system. Identifying the actors shines light on the relationships that each party has with the others, revealing the trust frontiers between actors. This serves to highlight issues that may arise due to poor or non-existent relationships between parties, such that they can be mitigated or addressed.

Design Cycle 1 illustrated trust frontiers between roles. These are shown in the RB Framework. They are not explicit in the UML representation, but can be traced through the dependencies on components and their providers.

3.5.3 Limitations

The RB Framework was generally well received by experts on the ERP during evaluation, and labelling has been revised somewhat in response to discussions with members of the group, becoming less centred around ML models and more open to other systems such as algorithms and rule-based systems. Labelling of the roles still has room for improvement - the DA role, for example, covers many possible viewpoints - from user or party impacted by the system, to other parties with an interest in systems, such as a regulator or an auditor. There is a balance to be struck between identifying all these possible roles, and making the framework more complicated, or in trying to find a suitable label for a generic role at the upper end of the system. Currently, we propose the latter, but are not convinced that Domain Authority is the correct term. Similarly, as we travel through the hierarchy, we feel that Systems Integrator is a technical term, familiar to those in Information Technology, but perhaps alien to those using systems - where a simpler Supplier or Vendor might be more appropriate. Model Engineer

and Data Provider too, are simplifications that may confuse those with little technical knowledge, whilst being troublingly imprecise for those with knowledge. We adopt these, until better terms are identified. This was evident during formative evaluation, when we consider the situation of Aurora as a direct to consumer service, we need to put Families, as the DA of the service (and the party making the decision to use the service). If it is deployed via a health agency, then the agency is the DA - and Families are a user. Clearly, we can have a range of stakeholders in the DA role, from auditors through to potential customers, and the framework might be extended or modified to reflect that.

The sketch of the RB framework drove the discussions with the ERP, and was used to identify and explain Trust Frontiers. The UML model was intended to formalize the framework, but does not identify communication paths and associated Trust Frontiers, or provide such a strong visual framework. As a result of the positive reaction from the ERP discussions, we have “promoted” the role of the sketch of the framework, and propose it as a contribution to knowledge. Our initial intent was to use only the UML model, but on reflection it is incomplete without identifying trust boundaries.

The framework as presented shows a static view of a DDS, in terms of showing its construction or composition, rather than its use. As such, it has no representation of the live data in the system, that might be used for inference or decision making. This would provide an equally valuable view on a DDS, and would identify the sub-systems and components that were being relied upon to drive the system, and perhaps to retrain its models.

3.6 Summary

In this chapter we have sought to answer RQ1, and identify the roles involved in developing and using a DDS, and their responsibilities and requirements. This has been approached through the development of a conceptual framework that can be used to

derive insight into the different roles of stakeholders involved in providing and using DDS, and understanding typical responsibilities and requirements of each role. In designing this model, the RB Framework, to represent roles and responsibilities in an DDS deployment environment, we have been able to consider different actors in DDS, which has helped to identify where and why each party might need information from other stakeholders, and what responsibilities each has to others.

The RB Framework provides a lens through which contributions and contributing parties to a DDS can be identified, and reveals primary, secondary and potentially tertiary dependencies between the parties in a DDS ecosystem. Communication links may need to be established between these parties, to exchange information in order to provide accountability and maintain oversight over the system. Analysis of a DDS through the RB Framework can lead to development of questions on who is responsible for the system overall, and for elements within the system and can be used to highlight shortcomings in identification of actors, which may prevent long-term problems if addressed sooner. The RB Framework brings focus on tensions that can exist between a desire for transparency, and competing needs for confidentiality and privacy which exist across the trust frontiers between roles.

The RB Framework defines a set of roles for actors involved in the development and application of a DDS within a domain. Each domain will come with its own unique responsibilities and requirements for actors fulfilling these roles. The ultimate responsibility, and hence risk, for acting upon these systems rests with the DA who will sanction its use and then be held accountable for that decision. In order to take on this risk, the DA needs assurance that the system is applicable, accurate and reliable for the use case it is applied to. They need to be able to trust it, and be able to demonstrate to others that it is trustworthy. This is especially important when a DDS is influencing decisions that can significantly impact human actors, such as the scenarios considered in the demonstrations set in military intelligence and healthcare domains.

Enumerating and clarifying the different roles and the hierarchy of interlinked depend-

encies between these roles helps to define expectations and establish a specification for an information sharing environment among actors, where often there is no direct relationship between parties, and distribution of power in the relationships is uneven. The RB Framework also helps to highlight tensions between desires for transparency about a DDS, and requirements for varying levels of confidentiality and privacy from contributors.

Chapter 4

A Verifiable Supply Chain Bill of Materials for Data-driven Systems

4.1 Introduction

This chapter considers how oversight can be provided on a data-driven system as a whole, as well as on the component parts, and other contributions made to a system. In particular, we seek to address RQ2 which asks “*How can contributions to a DDS be recorded and documented, so that traceability can be provided to stakeholders?*”.

Through the design research of this chapter, we develop an approach to providing oversight and traceability on DDS, such that systems can be scrutinised and assessed. We propose adoption of a bill of materials record for tracing contributions to DDS, as used in industry to keep records of the parts and assemblies used in physical products. We provide a data model and schema that describes how DDS can be represented in a bill of materials, in order that their contributions can be identified and scrutinised.

The research presented in this chapter has been realised over two design cycle iterations, described in section 4.3. The resulting contributions to the research knowledge base are presented in Section 4.4, with the results of evaluation by the Expert Review Panel provided in Section 4.5.

4.2 Problem Identification

As society increasingly relies upon inferences and predictions made by DDS, there is a pressing need to provide mechanisms for responsible parties, such as the Domain Authorities identified in the Roles and Boundaries Framework developed in Chapter 3, to be able to inspect DDS and the assets they are built from. Lack of oversight, or an inability to scrutinise contributions to components such as models and data assets, is exacerbated as the distance between specialist data providers, model engineers, and those using a DDS grows. The potential for problems to be observed in, or caused by, DDS compounds as datasets and models are themselves used to build new knowledge products, which move out of research laboratories and into deployment environments. This is a particular concern where shared datasets are used, or when models are sourced from third parties, through channels such as commercial AI-as-a-Service offerings, community marketplace platforms [173], science gateways and model zoos [87]. Analysis of experimental DDS developed rapidly in response to the COVID-19 pandemic to predict patients' conditions from CT scans [132], for example, uncovered systems built on public datasets which were found to have included data that was likely to lead to incorrect results. These so-called "Frankenstein datasets" [132], compiled from disparate, re-packaged sources, and assembled without due care or attention to detail can lead to unwitting introduction of bias into data – in this case using paediatric scans as data for healthy conditions, and duplicated use of certain images [137]. Such circumstances could present significant issues in a DDS deployed in high-stakes settings, and so it is vital that Domain Authorities and other stakeholders are able to identify models and datasets used in systems, and be able to assure themselves that appropriate standards are met.

As such, we contend that there is a need to develop mechanisms which can provide visibility on the components and contributions that lead to a DDS deployment. Further, any record of contributing assets should be machine readable, so that it can foster the development and adoption of tools that are able to support documentation and pro-

vision of scrutiny on DDS. A solution to the problem of lack of transparency and oversight on DDS should meet the following requirements:

- *Provide a framework that can give oversight on a DDS*, by documenting the components of, and contributions made, to the DDS.
- *Provide a data model for machine readable documentation*, such that interoperable tools can be offered in support of stakeholder needs.

The requirements are summarised in Table 4.1.

Requirement	Description	Design Cycle
R1	Provide a framework that can give oversight on a DDS	1
R2	Provide a data model for machine readable documentation	2

Table 4.1: Requirements of a Solution to the Problem

4.3 Design and Build

Following the DSRM, iterative design cycles were used to refine the research objectives and outputs, which are described below.

4.3.1 Design Cycle 1: A Bill of Materials

Definition of Objectives for a Solution

We propose to take a lead from industry, through the adoption of artefacts that record information about the supply chains of DDS. In the first design cycle, we develop an understanding of traceability and associated terminology relating to supply chains of other domains. We seek to use this knowledge to establish a framework which Systems Integrators can adopt to provide oversight on DDS.

Design and Development

The literature review of Chapter 2 introduced the notion of supply chain modelling, used across industries such as manufacturing and food production, which have had a long-standing need for traceability, for reasons of safety and quality assurance. A definition for traceability from industry is provided by Opara [118], as “the collection, documentation, maintenance, and application of information related to all processes in the supply chain in a manner that provides guarantee to the consumer and other stakeholders on the origin, location and life history of a product as well as assisting in crises management in the event of a safety and quality breach.” As these are needs increasingly evident in regards to DDS, describing the composition of DDS in terms of their supply chains has the potential to achieve similar results. As such, maintaining a record of the contributions that make up the supply chain of a DDS provides a mechanism to enumerate and identify models, datasets and other assets which contribute to the system. Furthermore, as new assets are created and used in other systems - perhaps by other parties - a record of the supply chain of these assets can provide traceability across the DDS ecosystem. Offering traceability will facilitate oversight, and lead to accountability, as Kroll observes, “traceability relates the objects of transparency (disclosures about a system or records created within that system) to the goals of accountability (holding the designers, developers, and operators of a computer system responsible for that system’s behaviors and ultimately assessing that the system reflects and upholds desired norms).” [85]

In industry, Jansen-Vullers, van Dorp, and Beulens [76] and van Dorp [161] discuss the composition of products in terms of a BOM, which is a list of the types of component needed to make a finished item. The BOM might specify a sub-assembly to be used, for example, and can be multi-level, wherein components can be used to create sub-assemblies which are subsequently used in several different product types. This maps well to the structure of a DDS, which can be assembled from sub-components in the form of ML models and datasets, for example. A DDS BOM could record a rich set of

information per contribution in the supply chain, with associated documents and payloads linked to or stored at each stage. By maintaining BOM documentation, SIs and system developers can create a record of the composition of each asset, as well as supporting artefacts, giving traceability onto models and datasets and the circumstances in which they were generated or obtained.

As noted previously, in Section 2.2.4, the concept of using a BOM to identify and record component parts of assets in a digital context is being recognised as good practice, with the US Department of Commerce working on the NTIA Software Component Transparency initiative to provide a standardised Software BOM¹ format to detail sub-components in software systems and applications. Here the intent is to provide visibility on underlying software modules, such that vulnerable or out-of-date code can be identified and replaced. This supports our motivation to develop a BOM for tracking the elements of a DDS, and in particular to document and provide a way to trace contributions from providers, so that any issues of data corruption, bias or vulnerabilities found in data sources or other components can be identified and flagged to DAs and other stakeholders. Research on the security and integrity of ML models, for example, identifies threat vectors which include Sybil attacks [167], data poisoning attacks [102], and model poisoning attacks [62]. Further, as models mature and are used in deployed DDS environments, it is conceivable that qualifications, best practice, and ethical or legal standards which were appropriate at development time are no longer adequate by the standards of the day or appropriate in the target domain.

Demonstration and Evaluation

In considering the development of a BOM record for a DDS, the contributing assets need to be identified and itemised. This can be achieved by analysing the production pipeline for a system, which will typically include ML models, data sources and datasets used for model training and validation, along with human expertise used in data

¹<https://www.ntia.doc.gov/SoftwareTransparency>

preparation and curation, and in development and testing. In addition to recording data and human effort, there may be other supporting assets which can be considered useful supplementary information when recording the characteristics of a data ecosystem, which Singh, Cobbe and Norval [144] have described as providing “decision provenance”. A BOM can provide a means to record such information so that it can be readily located and referenced. Describing DDS in terms of their supply chains provides a mechanism to identify data sources and the assets which contribute to the development of the data components, or which are produced as the results of intermediate processes. Being able to clearly identify and enumerate data sources contributing to a DDS provides a route to understand the “bibliometric data” [104] behind systems, as a way to assess contributions to AI research by factors including geography, gender, and other attributes.

Chapter 3 described the Roles and Boundaries Framework, a conceptual model designed to help identify and represent the components and contributions in a DDS. An instantiation of the RB Framework for a particular DDS, can provide the information needed to populate a BOM record of the supply chain for the system. As such, if the RB Framework is used to identify the components in a system and the stakeholders responsible for those components, then this can be used to derive a BOM record for the system. The process of analysis and inspection of the contributions to a DDS can guide stakeholders to consider where the data and the work that created the assets they are examining has originated, and how able they are to track changes and potential problems. This can particularly lead to new awareness on contributions that have low visibility, which evaluating parties can seek to address. Improving oversight on a DDS and its underlying contributions, gives stakeholders an opportunity to become aware of potential problems that might arise from components that might be considered to be of low quality or have inauspicious origins.

The drive for wider adoption of SBOM for software use shows that provision of machine-readable information in a digital asset’s supply chain record can foster improvements

in identifying and mitigating problems caused by underlying components. Providing a means to monitor the freshness of information provided about components and help to identify those which are abandoned [38], for example, is one aspect that could be automated once machine-readable supply chain information is made available, reducing potential attack vectors on systems due to neglected components [172]. Subsequent design cycles consider how to address this need for a machine-readable and verifiable BOM for DDS.

4.3.2 Design Cycle 2: A Schema for a DDS BOM

Revision of Objectives for a Solution

To advance the adoption of a machine-readable BOM record for a DDS supply chain, Design Cycle 2 considers how such a document could be structured, and what it would need to contain to convey information about a DDS such that the system can provide transparency and traceability. The intended outcome of this design cycle is to identify the elements required to describe the supply chain of a DDS, and to design a data model for a DDS BOM which defines how to represent these elements and the relationships between them. Developing a schema for the data model provides a structure that will support development of a machine-readable BOM.

Design and Development

The structure of a DDS can be described through the definition of a data model. The data model can be encoded in schema, providing an implementation of the rules that determine the validity of a data document. JSON Schema², a proposed IETF standard³, provides one such structure that can support this encoding. An advantage of using

²<https://json-schema.org>

³<https://datatracker.ietf.org/doc/html/draft-bhutton-json-schema-00>

JSON Schema is that both the schema and resultant BOM documents are machine-readable, and can be used to support the development of APIs and user interfaces, and to develop interoperable tool support around the documentation of a DDS [9]. Using a declarative schema formatted to the JSON Schema standard to describe the data structure requirements for the BOM document will help to ensure that underlying JSON documents are compliant with a consistent and constrained structure, supporting interoperability and integrity of exchanged data [123].

Building on the analysis of DDS which derived the RB Framework described in Chapter 3, a DDS can be considered to be composed of digital assets, which include models and datasets. These assets can have dependencies on other assets – both in aggregation and composition, as demonstrated in the UML model shown in Figure 3.10. We have identified a requirement for providing oversight on the component assets in a DDS. We have also identified a need to identify parties that have provided assets, and who may be accountable for their qualities. Furthermore, there may be confidentiality and privacy constraints that prevent open sharing of information among participants in a system. As such, we determine that the DDS BOM schema needs to provide a structure that can support:

- Identification and descriptions of assets
- Identification of dependencies on other assets
- Identification of parties accountable for assets
- Conditional access to confidential or private information

The proposed structure for a BOM is shown in Table 4.2. The *Asset Type* is defined as being one of a DDS, a Model or a Dataset, as used in Roles and Boundaries Framework, or an Artefact – a new type which has been introduced to provide support for recording other pertinent information – to build “decision provenance” [144]. *Provider*

is a name or identifier of the party providing the asset – this would be the Systems Integrator, Model Engineer or Data Provider in the RB Framework. The optional *Verification* attribute provides a route by which verifiable evidence of the claims made in the component description can be obtained. The Verification route may also provide additional – and perhaps conditional – information to requesting parties and offers support for structured transparency around sensitive commercial or private information. Verification is to be provided by a party that can be held accountable for the information provided in the BOM description. Verification could be supported through a technical approach (such as self-sovereign identity protocols, as future chapters will demonstrate), or by other means, such as a phone number or personal contact. The party identified as the Provider and the accountable party in Verification may be the same, but it is not required to be so. The proposed structure is iterative, such that an asset may contain other assets through the *Known Dependencies* field. This field is named in recognition of the fact that not all information may be publicly shared, and so there might be other dependencies that are not disclosed in the asset’s BOM document.

Field	Description
Id	An identifier so that entities can be uniquely referenced
Descriptive fields	Name, identifier, version, description, etc.
Metadata	Information about the asset and its generation, etc.
Asset Type	DDS, Model, Dataset or Artefact
Provider	Party providing asset
Verifier	Route to get verification from accountable party
Known Dependencies	List of other assets used to build this asset

Table 4.2: Elements of an Asset in a BOM document

This definition of entities can be used to develop a JSON Schema that describes an Asset, so that BOM documents can be instantiated using the schema. Appendix C provides Listing C.1, a JSON Schema definition that represents the Asset using the attributes from Table 4.2.

Demonstration and Evaluation

The proposed data model can be demonstrated by instantiating the JSON Schema as a BOM for a DDS. The schema defined in this design cycle provides structures that can be used to describe entities that contribute to a DDS as Models, Datasets and Accountable Parties (i.e., the SI, ME and DP previously identified). By application of the schema to different scenarios, and subsequent evaluation of these instances, it can be determined whether the data model defines suitable objects and relationships. Once again the artificial scenario S1 developed by Preece and colleagues [125] and S2, the Aurora chatbot, described in Definition 1 are used to motivate demonstration and evaluation of the design approach. The JSON Schema is instantiated for the DDS described in these scenarios, which supports an assessment as to whether the entities proposed in the schema are sufficiently expressive or require further revision.

Considering Figure 3.7, which illustrated the models and datasets for scenario S1 (the CCTV Monitor), and using Table 4.2 to identify entities of the scenario, results in Table 4.3. This can be encoded with the JSON Schema of Listing C.1 (Appendix C), to provide a JSON formatted BOM document that describes the DDS. The JSON representation of the BOM is shown in Listing C.2 (Appendix C), which shows definitions for the component assets identified previously. A format has been adopted which provides an *id* for assets which have further information available, and not for those which have been sourced from outside of the direct ecosystem. Where an *id* is specified, a JSON definition is provided for the asset. Note that in all cases, references to external documentation and other pertinent biographical information could be provided through the descriptive or metadata fields.

A similar analysis can be followed for Scenario S2, the Aurora chatbot. Figure 3.8 identified the model and dataset assets using the RB Framework, which are enumerated in Table 4.4. A JSON encoding of the data based on the JSON Schema of Listing C.1 (Appendix C), provides a BOM document for the Aurora DDS, which is shown in Listing C.3 (Appendix C).

Id	Asset	Attribute	Value
\$CCTVMonitor	DDS	descriptiveFields	"CCTV Monitor"
		Provider	AnglovaLaw (SI)
		knownDependencies	\$DNN
\$DNN	Model	Name	"DNN"
		Provider	Taylor, et al (ME)
		knownDependencies	3DMobileNet, VGGish, \$Curated_UCF-101
\$Curated_UCF-101	Dataset	descriptiveFields	"Curated UCF-101"
		Provider	Taylor, et al (DP)
		knownDependencies	UCF-101

Table 4.3: Entities in Scenario S1, CCTV Monitor

Open source code is available⁴ to perform verification of supplied JSON Schema and JSON data files instantiated against those schema. The JSON Schema and JSON files resulting from Design Cycle 2 were successfully tested using a web-based JSON Schema validation service⁵. This shows both that the JSON Schema is grammatically correct, and the data in the JSON files correctly follows the structure defined by the schema. Application of the data model, via its JSON Schema (Listing C.1), to the scenarios described above has demonstrated that it provides a vocabulary that allows the structure of a DDS to be documented, and the relationships between assets in the DDS to be recorded.

⁴<https://github.com/json-schema-org/JSON-Schema-Test-Suite>

⁵<https://www.jsonschemavalidator.net/>

Id	Asset	Attribute	Value
\$Aurora	DDS	descriptiveFields	"Aurora"
		Provider	CM (SI)
		knownDependencies	\$DialogFlow
\$DialogFlow	Model	Name	"Dialogflow"
		Provider	Google (ME)
		knownDependencies	\$EnglishModel, \$PortugueseModel
\$EnglishModel	Model	descriptiveFields	"Text Analysis (EN)"
		Provider	Google (ME)
		knownDependencies	\$EnglishData
\$PortugueseModel	Model	descriptiveFields	"Text Analysis (PT)"
		Provider	Google (ME)
		knownDependencies	\$PortugueseData
\$EnglishData	Dataset	descriptiveFields	"Conversation (EN)"
		Provider	CM (DP)
\$PortugueseData	Dataset	descriptiveFields	"Conversation (PT)"
		Provider	CM (DP)

Table 4.4: Entities in Scenario S2, Aurora

4.4 Research Outputs

4.4.1 A Bill of Materials for DDS and Supporting Data Model

Through the design cycles presented in this chapter we have motivated the adoption of a BOM document for DDS. The proposed DDS BOM provides a framework to record information about the assets and contributions that form the supply chain of a DDS. The

BOM enables parties to take responsibility for their contributions to a DDS, providing transparency and enabling stakeholders to seek verification of claims made about the DDS and assets that constitute it. The proposal is supported by the long-term use of BOM in industry and food production for tracking components and sub-components in an assembly in order to provide transparency and traceability. The approach is further endorsed by SBOM initiatives in the software sector, which maintain a record of contributions to software systems to help identify potential vulnerabilities in underlying components and modules.

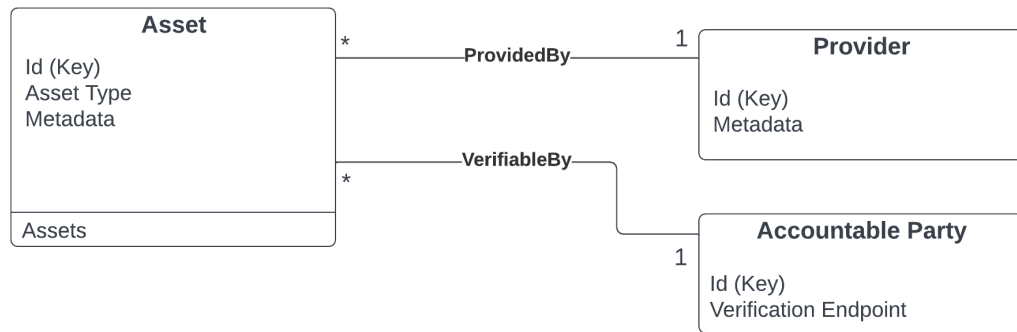


Figure 4.1: Data Model for a DDS Bill of Materials

In order to facilitate development of a machine readable DDS BOM document, a data model has been developed, and instantiated in a schema defined as a JSON Schema. The data model and schema define a format for recording contributions to a DDS, so that a BOM can be written for a DDS and used by other tools and systems that adopt the schema. The data model is shown in Figure 4.1. The model is presented in Entity Data Model (EDM) representation, which is intended to support development of rich, data-centric applications [25], by describing the structure of data in terms of its constituent parts. The data model diagram shows a conceptual model with three key entity types: Asset, Provider, and Accountable Party.

Following the recommendation of Engström, et al. [52], we provide a technological rule to summarise the research contribution, thus: *In order to provide transparency and*

traceability on contributions made to a DDS where machine readability is required, one should maintain a BOM document based on a published data model.

In comparison to proposals from published literature for recording the constituent parts of an AI System or an ML Model, including Model Cards [107] and Factsheets [6], the contribution presented here provides a record that is machine readable, through its observance of a data schema format. This will facilitate the design and development of software tools and other systems that can collect and convey information between stakeholders.

4.4.2 Communication

Research from this chapter has been formally communicated as it has developed by publication of peer-reviewed papers [11, 13, 17], and through informal venues. The proposal to analyse the supply chain for DDS and maintain a BOM record has been regularly communicated within research groups at Cardiff University and University of Notre Dame, and to academic and industry colleagues in the DAIS-ITA project. The work has also been presented in a research seminar to an audience of data industry experts at the UK's National Statistics Office, and to the scientific community through presentations at the International Workshop on Science Gateways.

4.5 Evaluation

The design cycle iterations described above have each included a formative evaluation, through the application of the BOM framework, data model and associated JSON Schema in artificial scenarios. To conclude this episode, an ex-post, summative evaluation on the adoption of a supply chain-based approach for recording of contributions to a DDS was undertaken.

A Data Supply Chain

- Machinery and food have well documented “supply chains”
 - eg. a Tractor has an engine, engine is made from various sub-assemblies
- Supply Chains are documented with a “Bill of Materials”
 - Provides traceability and transparency
- DDS are made from different components, produced by different parties
 - Data - collection, curation, labelling, etc.
 - Models - design, development, testing.
 - Different assets produced at different stages
- Is a “Bill of Materials” useful for a DDS? Is it possible?

Figure 4.2: Data Supply Chain Slide presented to ERP interviewees

4.5.1 The Rigour Cycle: Expert Review Panel

Evaluation of the proposed application of a supply chain model and BOM for DDS was conducted by expert peer-review, taking the form of semi-structured interviews with expert peers on the Expert Review Panel, as described in Section 1.6.2. Interviews with panel members took place following the conclusion of the design research described across this thesis. ERP participants were first introduced to the research problem, and then to the Roles and Boundaries Framework, which resulted in findings presented in Section 3.5.1. As discussion on the RB Framework concluded, subjects were shown slides introducing the notion of a supply chain for DDS, Figure 4.2, and an extension of the supply chain in which qualities of contributions might be able to be endorsed and subsequently verified, Figure 4.3. Both slides were presented verbally to panelists. The ensuing discussion with each interviewee was later transcribed and analysed, to develop a critique formulated against Checkland’s 5E’s criteria [37], summarised in Definition 4.

A Verifiable DDS Supply Chain

- Oversight is provided across the DDS supply chain
- Make “Shared Data” and ML systems more trustworthy
 - Anyone can certify and check qualities - creators, regulators, etc.
- Parties take accountability for their contributions
 - Make data and models “trustworthy”
 - Digital signatures used to “back up” quality claims
 - Responsibility is taken for parts of the system
 - Trust develops between participants
- The need to maintain confidentiality is recognised
 - Not all information is available to everyone
 - Information can be requested, but owner decides

Figure 4.3: Verifiable Supply Chain Slide presented to ERP interviewees

Efficacy: Does the proposed evaluand work?

Do subjects see value in the supply chain BOM model for DDS?

Efficiency: Is resource use minimised?

Can subjects align their DDS with the model?

Effectiveness: Does the proposal help to attain the long term goals?

Could a BOM provide traceability on contributions to a DDS?

Ethicality: Is the proposal a moral thing to do?

How might a DDS BOM be adopted?

Elegance: Is the proposal able to be performed aesthetically?

Could asset endorsement and verification work in practice?

Definition 4: Evaluation Criteria formulated as the ‘5E’s’

Efficacy: *Do subjects see value in the supply chain BOM model for DDS?*

The criterion of efficacy seeks to consider whether the proposed solution is likely to work. In this case, we report the early reactions of the subjects to the proposal of a supply chain model, and whether this is positive or otherwise.

Following analysis of the interview transcripts, eight ERP members introduced to this proposal⁶ expressed an opinion on the suggestion of considering the contributions to a DDS as a supply chain. Of these, six were positive – Machine Learning Engineer F saying “I definitely very much like the supply chain analogy” and B, a Chief Data Officer, thought along the same lines – “I think the analogy is a good one”. Software engineer G thought the proposal was “a good idea”, as did D, and H, who works with data in a community setting “loved the idea”. J, the data lead at a national scale non-profit organisation, expressed enthusiasm for the idea, and had actually suggested it during the part of the interview about the RB Framework (discussed in Section 3.5.1).

C was more reticent, based on their experience with rule-based maritime systems. They aligned the BOM to a “technical specification of a solution”, and stated that they prepared this type of documentation for their customers, but was doubtful that they “systematically use it for anything.”. A, a senior researcher at a global organisation, felt maintaining a BOM was a sensible approach, but said they had come across the idea previously, during work with the AI Now Institute⁷.

A significant proportion (six out of eight) of ERP members reacted favourably to the proposal, two members were more guarded, but not overly negative or dismissive of the notion. Two panel members (one very keen, and one less so) had previous exposure to the concept of a supply chain for DDS from other sources.

Efficiency: *Can subjects align their DDS structures with the model?*

Efficiency is considered in the context of the applicability of the model to the subject’s requirements, with the view that if it is a good fit, then it can be considered likely to efficient (or worthwhile to make an effort to apply), otherwise not. Here, we seek to understand whether interviewees could align the proposal to their understanding of

⁶The interview with E was cut short due to unexpected circumstances, and finished prior to this section

⁷<https://ainowinstitute.org>

DDS, as that would be likely to lead to an efficient adoption of the model.

In discussion, five of the experts from the ERP replayed their own version of the narrative about a supply chain for a DDS, and how it would work and the potential benefits it would bring. D referred back to the UML model shown to panel members on an earlier slide (3.11) and used that diagram to set the context of their description of a supply chain, relating the process to food production “the system that is providing the output, is the meal. And then within there’s the ingredients that go into it, being the data and the model”. J described the supply chain in terms of food processing, using a metaphor of tracing meat back to a farm, and saying that such a scheme “doesn’t seem to exist in this [DDS] world”. F and G both described a DDS supply chain. F imagined datasets being combined to create new data sets and trained models. G pointing out that “it is not just the materials it’s the way that they’ve been combined” and that “we start from the raw materials which are the datasets”. H also picked up on this point, “if you were able then to trace how that data was used, and how it was used to build models, in future studies you could know where the source data comes from.”

Participants were very clear that they understood the concept of the DDS BOM, and provided unsolicited descriptions of the approach in their own words and contexts, demonstrating that the model aligned well with their own mental models.

Effectiveness: Could a BOM provide traceability on contributions to a DDS?

To help evaluate the potential effectiveness of the supply chain BOM model, we consider ERP members expressed thoughts on it being able to bring transparency and traceability to a DDS, and the components of a DDS.

J, who had mentioned a supply chain before the slide was shown, used the food analogy, and suggested “you can have a look and trace it back, and there’s a record of where it goes back to”. F expressed that a supply chain BOM record would be “great from a visibility perspective”. H felt that it would be “extremely powerful” to be able

to “trace systems through to their roots, and then seeing how they move through networks would be awesome”. G similarly saw the opportunity to “track that materials go into something that is produced into a product and then pass on to the next”. D suggested that “the data, a lot of the time is probably the least known part of a system.” Drawing on a particular use case from their experience in neuroscience, H pointed out a potential benefit of such a system “if you were able then to trace how that data was used, and how it was used to build models, in future studies you could know where the source data comes from, and then you know any problems that arise.” C, one of the panelists showing the least enthusiasm towards the BOM approach, gave a stronger endorsement to the benefits of traceability on the origins of data in their systems: “to explain where each piece of IoT or other data comes from is a valuable thing for the the end users”.

A was less sanguine about the proposal, stating that “it doesn’t capture all the things that you need to capture.” They gave their view that a BOM in the manufacturing world is used to determine product cost and price, but doesn’t have any link to processes or information on how parts are to be used – they illustrated their point using an example of an Arduino: the BOM for which might state that it contained “a crystal in the CPU and some resistors” but wouldn’t cover how the parts were “glued together”. Similarly, they felt that knowing that an ML model was trained on MNIST “doesn’t particularly tell me what I’ve trained, and the shape of the model, and the function that determines what it’s learning, and the goal of the function.”

Several participants expressed a clear view of how a supply chain record for a DDS could provide a means to gain visibility and transparency on the components of a system. A provided further insight, and identified that a BOM record that is just a list of components in a system does not capture all the information that stakeholders might need to fully understand the system.

Ethicality: *How might a DDS BOM be adopted?*

The Ethicality criterion considers whether the proposal is a moral thing to do. Here, we report on the ERP participants observations on how such schemes might be adopted, and whether SIs or other participants might voluntarily share information, or whether it would require regulation from government or other bodies. As a result of the semi-structured nature of the interviews, this topic was not discussed with all of the panellists, but interesting perspectives were offered where it was covered.

C, the owner of a business that provides DDS to the maritime sector, revealed that they had “a very open policy to explaining how our system works, and we don’t hide much of what we do.” They reflected that the systems they provided were (in their view) simple, rule-based systems, based on common sense. They felt that the value their business offered to customers was “not the technical side or the fine algorithms, more the usability, the customer service levels.” In behaving in such a way, and focusing on developing relationships with customers as their SI, they were able to gain the trust of their customers. Based on C’s experience, they felt that “the end users neither care, or know about, where all data comes from” and that usually there were staff within the customer organisation who’s role was to understand how the system works, and where the data comes from. Other staff in turn trusted the person in that role – “every person in an organisation has a specific set of tasks, and if you have a data gatekeeper in your organisation, then the rest of the organisation should be able to trust those buyers” – by developing a good relationship with those gatekeepers, C was able to help them in their jobs as the “gatekeepers” in their organisations.

As the Chief Data Officer of a global publisher, H was able to share insight into the factors that might drive adoption of a DDS BOM. They related it to a recent surge in demand for evidence around the Environmental, Social and Governance (ESG) impact of operations. H identified that there is a lot of work being done around ESG supply chains, with no real underlying regulatory requirement. They felt that the moves seemed to be being driven by investment markets, and wondered whether that might

be an interesting first step – perhaps driven by “adverse consequences” from a DDS, which will motivate the financial investment community to make demands to have systems in place in order to make sure their investments were sound. H felt that a DDS BOM (or other approach) could be imposed either by future regulation, or be “driven by the financial investment community”.

Participants offered insight into the benefits of having an open approach to sharing DDS information via a BOM, for C this was as part of their customer service ethic. H felt that a move to such openness might be driven by demand from investors, as had been the case with ESG – this demand preceded regulatory requirements.

Elegance: Could asset endorsement and verification work in practice?

Finally we consider the elegance of the proposed approach, which considers how it could be applied, and what might be involved in that process. Panelists were introduced to the notion of parties having an ability to endorse qualities of contributions to the DDS BOM for subsequent verification. Discussion often turned to how aspects of the system might be endorsed and verified, and who might fulfil roles in that process.

A succinctly raised the concern, “One of the things that I think is tricky is verified by who?”. A was uneasy with the notion of verifying parties being assigned, feeling it is “deeply unsatisfactory for the world to tell me who the verifiers are” - conversely, they felt exposed to being “conned” when having to make their own choices. In summary they felt “It’s quite a tricky problem.”

F explained that “you can either self sign or presumably there might be some other external bodies that could sign something for you if you want like a that extra stamp”. B’s experience suggested that trusted third parties might have a role to play, in inspecting components of a system, and providing a certification of qualities. In particular, B felt that companies “will make all sorts of proclamations about their data sets which aren’t necessarily accurate or true” – and those dependant on these parties had to rely

on their honesty, but they felt it would be “really quite helpful” if a trusted third party could provide and attest to verification. B provided examples of bodies that might play such a role, often coming from existing regulatory groups in different domains, such as Financial Conduct Authority, Law Society, Solicitors Regulation Authority or the General Medical Council. They felt that a regulator would particularly have a role to play as AI regulations from the EU and other groups emerged. C suggested that an “NGO-backed organisation” might occupy a trusted role, and gave Wikipedia and Mozilla as other examples. They were sceptical of placing trust in claims made about “commercial technology managed by a specific company”. G suggested that endorsement of qualities of a DDS or components from a regulator or “somebody independent” might hold more credibility than assertions from an SI.

F explored the situation further, and raised an awareness of data and models being used across multiple DDS, asking “could a data set have multiple different certificates that would be contextual in a way, because the value or the potential harm from data set in the model or the value from a data model is context dependent”

Several panelists revealed concerns as to whether they would have confidence in parties making claims about assets, with a reluctance to place faith in claims from commercial companies. Enthusiasm was expressed for the role of a third party in endorsing assets, whether that was a party in an official, regulatory role, or an independent inspector or auditor of some kind. F raised a significant point about the value of endorsement being contextual, or domain dependant.

4.5.2 The Relevance Cycle

The relevance of a designed artefact can be evaluated by considering it in the context of its environment, and assessing how well it would operate as a solution to the problems identified in the environment, as described in Section 4.2.

Through application to artificial case study scenarios, the BOM model and supporting

data model, with its JSON Schema structure, has been shown to be adaptable, and able to be used to record contributions made towards different DDS. Table 4.1 identified requirements for a documentation structure that needed to be satisfied to address the problems of a lack of oversight on DDS. Discussion of how each requirement is met through the proposed BOM record and data schema follows.

R1 Provide a framework that can give oversight on a DDS

A BOM document provides a framework by which contributions to a DDS can be documented and recorded. By documenting a DDS as a BOM, details on each element of the system can be captured, such that contributions to the system can be identified. As such, a BOM document can provide oversight on a DDS.

Design Cycle 1 considered existing documentation of multi-contributor systems, from industry and food production, and identified similarities between the requirements from these sectors for transparency and traceability and the needs in a DDS deployment. The BOM approach used in the other sectors was determined to be appropriate for adoption in documenting a DDS to provide oversight on its components and contributors.

R2 Provide a data model for machine readable documentation

In order for BOM documentation to have maximum utility, it is desirable for it to be written in a machine-readable format. This will facilitate the development of APIs and tools for creating and later viewing and verifying the BOM of a DDS. JSON Schema was adopted as a format for defining a data model that described a BOM structure.

Design Cycle 2 developed a data model and schema that can be used to describe a DDS BOM. The JSON Schema structured format supports creation and use of interoperable and machine readable documentation of a DDS BOM, which can be used to describe the structure of a DDS and identify and document its components and contributors. Adoption of the JSON Schema will provide an integrity

layer to support data exchange between APIs and tools which use JSON formatted data records to document DDS BOM structures. As a technical IS artefact, the JSON Schema that encapsulates the data model can be verified for its grammatical correctness, and was analysed in Design Cycle 2 with a JSON Schema validation service⁸ which verified that the schema is grammatically correct.

4.5.3 Limitations

This chapter has presented design work from our research in identifying a supply chain model as a viable framework for providing documentation of contributions towards a DDS. Evaluation of the approach has been conducted during formative design cycles through application to artificial case studies, and through analysis of opinions on the approach offered during semi-structured interviews with expert peers. A set of experimental deployments using a wider range of DDS styles would provide further insight into the practical applicability of the approach, and its suitability for use in real systems. Practical application of the proposed approach to DDS would also help to support and validate the opinions offered by the members of the ERP, who were largely supportive of the proposed model, and its ability to record contributions to a DDS.

The data model developed through the design research of this chapter provides a simple, document-based structure for recording contributions to a DDS, backed by a schema that supports implementation of the data model. In practical use, additional data fields may be identified as required to provide the richness of documentation necessary to convey information about a DDS from its developers and integrators to practitioners. The data model and JSON Schema can be extended to add support for such new attributes as required. JSON's support for data representation as key-value pairs provides useful flexibility to integrate existing documentation or workflow pipeline data into an account of the supply chain for a system. This includes documentation for datasets and ML models which followed schemes such as Datasheets for Data [58], Model

⁸<https://www.jsonschemavalidator.net/>

Cards [107], or similar proposals. These artefacts would be able to be linked into the BOM by reference. Application of the approach to an extended set of use cases and deployments would serve to motivate further work into fully describing the data structure required to record a BOM, and the assets it contains.

4.6 Summary

In addressing the research question “How can contributions to a DDS be recorded and documented, so that traceability can be provided to stakeholders?” (RQ2), we have sought to identify an approach to providing oversight on a DDS so that systems and assets that contribute to the systems can be identified and scrutinised. Taking a lead from manufacturing and food industries, and inline with recent developments in US Government policy towards SBOM, we have adopted a supply chain model and applied it to DDS, through the definition of a data structure for a BOM document. There is a strong mapping between sub-assemblies used in manufacturing production, for example, and the composition of a DDS, which is reliant on integration of models, which in turn are reliant on generation and curation of data assets. Discussions with the ERP showed a very strong fit between the mental models of interviews about DDS, and the proposed BOM approach, to the extent that several panellists mirrored descriptions of the approach in their own words and contexts.

Proposals such as TRLs for DDS, as suggested by Lavin and colleagues [90] provides useful framing for the timeline of development and deployment of these systems. A DDS BOM becomes increasingly relevant from a TRL Level 4 onwards, where project teams become larger and more interdisciplinary, and the DDS is deployed further into its production environment. As a DDS progresses through the TRL stages and across organisations, the need for accountability and the ability to scrutinise and verify the qualities of assets becomes more and more important. We contend that adopting a BOM model based on an industry supply chain, and maintaining a record of the con-

tributing assets of a DDS through a BOM document is an effective way to maintain traceability of such systems, and provide accountability as systems mature, in support of a TRL-based approach advocated and used in large-scale systems engineering projects. The ERP discussions provided insight that a demand for such documentation may come from unexpected quarters, such as the financial investment community, as was the case with ESG evidence requirements for manufacturing supply chains.

The research presented in this chapter has also resulted in the design of a data model and a published schema definition, which provides a structure capable of describing a DDS and the digital assets that contribute to the DDS in a machine-readable form. Providing the data model definition in JSON Schema format facilitates the development of software tools that can support documentation of DDS and digital assets, as well as mechanisms for providing oversight and verifying accountability of contributions to a DDS.

Chapter 5

Providing Accountability, Oversight and Information Security for Digital Assets

5.1 Introduction

The hypothesis for this thesis, detailed in Section 1.3, is that principles and design patterns, data models and protocols from the emerging field of Self-Sovereign Identity can provide a technological means towards providing necessary assurance on the qualities of digital assets, and accountability on parties making such claims. The research in this chapter looks into the application of the SSI approach, and considers RQ3, “*How can SSI models be used to provide accountability and assurance on the qualities of assets contributed by different participants to a data-driven system, whilst maintaining the information security requirements of the contributors?*” In addressing this question, we seek to demonstrate that an SSI-based approach can be used to provide oversight and accountability on digital assets, whilst also protecting confidential and private material from unauthorised access. As a step towards the overall goal of this thesis in designing a solution for DDS oversight and accountability using SSI, the focus of this chapter is on how to provide oversight and accountability on a singular digital asset, which might subsequently be used as a component of a DDS.

Following the DSRM framework introduced in Section 1.6, the research of this chapter has been realised over three design cycle iterations, each containing the stages of defining objectives for a solution, design and development, and demonstration and evaluation, and informing the subsequent cycle. The design cycles are described in section 5.3, and result in a software architecture. This architecture is applied to a case study scenario based on scientific data sharing in the multi-messenger astrophysics community, which is described in Section 5.3.3. In addition to production of functional artefacts, DSRM design cycles are intended to provide contributions to the knowledge base through communication and formalised learning. Our contributions are presented in Section 5.4, and evaluated in Section 5.5. Concluding thoughts are given in Section 5.6.

5.2 Problem Identification

The review of literature and related work in Section 2.2 identified the lack of a viable technical solution for providing transparency, traceability and accountability in on assets in multi-stakeholder DDS, and found that maintaining control over access to confidential or proprietary information about assets is needed.

To address these shortcomings, a solution needs to meet the following requirements, which are summarised in Table 5.1:

- *Provide assurance on qualities of digital assets*, by enabling verifying parties to inspect and verify claims made about an asset's qualities.
- *Provide accountability for digital assets*, such that parties using assets can identify the party responsible for making claims about its qualities and suitability.
- *Protect confidential information* which might be commercially sensitive, or private personal information, relating to digital assets, from unauthorised access.

Requirement	Description	Design Cycle
R1	Provide assurance on qualities of digital assets	1,2
R2	Provide accountability for digital assets	1,2
R3	Protect confidential information	3

Table 5.1: Requirements of a Solution to the Problem

The concepts and design patterns of SSI were introduced in Section 2.4. Here, we consider how SSI concepts and nascent implementations of technology that supports these concepts can be used to provide a solution that meets the requirements outlined above.

5.3 Design and Build

The DSRM guides research through a series of design cycles which aid in refining the objectives and iterating towards a design solution. Our research, and the resulting IS artefact presented in the form of a software architecture is developed over 3 design cycles, described below.

5.3.1 Design Cycle 1: Investigation of Verifiable Credentials

Definition of Objectives for a Solution

The primary objective of the first design cycle is to apply SSI constructs of decentralised identifiers and verifiable credentials to digital assets, in order to develop a model that enables asset owners to use VCs to provide signed attestations about their datasets. The integrity of these attestations are then able to be verified by interested parties. For clarity, it is not the content of the claims about assets that can be verified here, rather it is the statements (the VCs, in SSI terminology) making the claims – SSI protocols can

be used to show that those statements were made by the party claiming to have made them, and to have not been tampered with subsequently.

Design and Development

A core tenet of the SSI approach is that entities claiming to be the controller of a DID can provide cryptographic proof that this is the case, facilitated by a protocol that provides a resolvable route to a verification mechanism. At its simplest, this proof can be provided in the form of a structured document file containing the public key of the DID, along with the methods by which a party can verify. By using the published verification mechanisms the holder of a document that purports to have been signed by the DID's controller can obtain cryptographic proof that it was indeed signed by the DID controller, and furthermore can verify that the document has not been tampered with since it was signed. This systematic mechanism for a party to prove that they have access to the private keys relating to a DID is used when issuing VCs. If the VC claim document is signed by a party with a DID that is known to the verifier, or can be found in a trusted registry, then the claims made in the VC document can be valued by the regard given to that party. Where the issuer is a representative of an entity, such as a university or other well-regarded organisation this trust may be inherent. In other circumstances the issuer may need to provide their own credentials from bodies with a better established reputation in order to assert their qualities as a trustworthy party. In other cases, where a peer-to-peer relationship exists, between scientists, for example, identification and knowledge of the issuer as a member of a Community of Practise [164] can be significant and powerful.

Demonstration and Evaluation

Guided by Venable, et al., FEDS framework for evaluation in design science research [165], we can create an artificial scenario in order to evaluate *technical risk and efficacy* of a

proposed approach in a formative manner, providing insight that can be adopted in future design cycles. To provide a demonstration, a scenario was constructed based upon scientific data, that is set to be shared with other parties, with VCs used to provide signed assertions of qualities of the metadata of the dataset.

This requires that the Principal Investigator (PI) signs a document using the private key of a DID they control, signalling that any claims in the document can be taken to be claims that they are willing to endorse [14]. As the credential issuer, the PI will be a “trust anchor” in the system, such that verifiers will need to have or develop trust in the PI (and “directly accept [them] as reliable”) to place value on credentials they issue [91].

```
{
  "@context": "https://w3id.org/did/v1",
  "id": "did:web:uniofscience.com",
  "authentication": [{
    "id": "did:web:uniofscience.com",
    "type": "Ed25519VerificationKey2018",
    "controller": "did:web:uniofscience.com",
    "publicKeyBase58": "71ANMccQC..."
  }]
  ...
}
```

Listing 5.1: A Fragment of the UniOfScience DID Document

To instantiate the demonstration, a VC document was produced for an example dataset, using domain names registered for *UniOfScience*, a fictitious university, and a web site for the dataset, at *DIDdoi.com*. An open source software package *vc-js* [47] was used to generate VC documents using the *did:web* [153] DID scheme for identifiers. This naming scheme takes advantage of the fact that most organisations operate web sites, with SSL certificates proving the legitimacy of the identity of the web site’s address.

The *did:web* scheme uses the web site of the organisation to host the DID document, resolving *did:web* to a JSON-LD file located on the web site at a well known path [114]. This provides assurance that is reliant on authorised users being able to upload files to an organisation's official web site. Listing 5.1 shows a fragment of the DID Document for *UniOfScience*.

A further requirement is a credential schema [147], which defines the semantic vocabulary used to describe the attributes of the dataset, and provides the format in which the claims about a particular subject will be made. To produce a VC document for demonstration, the *vc-js* library was integrated with the Node.js Express [63] framework to enable a simple web form to be served to allow a user — a PI preparing to publish a dataset — to enter metadata for the dataset. This was subsequently used to populate data fields in the credential schema, and *vc-js* invoked to encapsulate these values in a JSON-LD formatted VC document containing a proof issued by the DID belonging to *UniOfScience*. The inclusion of the DID of the issuer of the VC document enables third parties to check its integrity, achieved by resolving the DID to locate the DID Document holding the mechanisms for verifying signatures. Verifiers can use methods in *vc-js* to receive cryptographic proof of the integrity of the VC document, assuring them that it hasn't been tampered with since it was issued. As the payload of the VC document contains the DID of the dataset that it refers to, verifiers have cryptographic proof that the issuer has signed a document attesting to the properties of the DID of the subject. If the subject DID references a dataset, then the verifier can be assured that the VC document carries signed assertions about the properties of the dataset.

This demonstration provided insight into the use of DIDs and VCs for assets. However, providing a VC to another party is not sufficient to prove that a claim made about an asset is valid. An unauthorised party with possession of a VC could present it as valid, making verifying parties vulnerable to a replay attack [59]. To prevent such replay attacks, the verifier should ask the claim holder to present a Verifiable Proof (VP). This takes the form of a document signed by the VC holder, which contains a

response to challenge, typically a nonce, issued by the verifier, along with the credential and its proof. By inspecting the VP through resolution of the DID of its issuer, and comparing this DID with the DID of the VC's subject, the verifier can determine that the challenge response is acceptable, and has been generated in response to a request. Further verification of the credential contained in the VP can demonstrate that the credential has not been tampered with, and thereby provide assurance that a VC about the dataset has been issued by an authorised party, to a holder who is authorised to present it.

The credential and proof exchange mechanism demonstrated in this design cycle required that parties wishing to verify an asset's credentials make requests and that operators are on hand to manage incoming requests, and to generate and sign VP documents. This would likely become impractical where there was high demand, or a need for a timely response. One approach to mitigation might be to consider a digital asset such as dataset or an ML model as a self-sovereign entity in its own right, represented by a software agent, which maintains control over its own credentials. The following design cycle considers this approach, and investigates how to automate generation of proof requests for assets, such that they are suitable for adoption in machine-to-machine interactions.

5.3.2 Design Cycle 2: Self-sovereign Data

Revision of Objectives

To provide further understanding of the implications of using an SSI approach with digital assets, Design Cycle 2 investigates an approach in which the digital assets themselves are treated as self-sovereign entities, each with its own DID and the ability to hold and present credentials and proofs. The objective of this design cycle is to design and demonstrate a system that represents digital assets with SSI-capable software agents, so that parties in the ecosystem can issue, inspect and verify claims made

about the assets.

Design and Development

This design cycle extends the use of SSI for digital assets towards an agent-based paradigm. In this approach, software processes are used to represent different entities in the system, which are able to communicate with each other using SSI protocols. An agent-based approach to a data sharing scenario would use an SSI software agent to represent a published dataset. This dataset agent (DSA) would be issued with VCs, and would have the capabilities required to issue VPs to third parties that requested them.

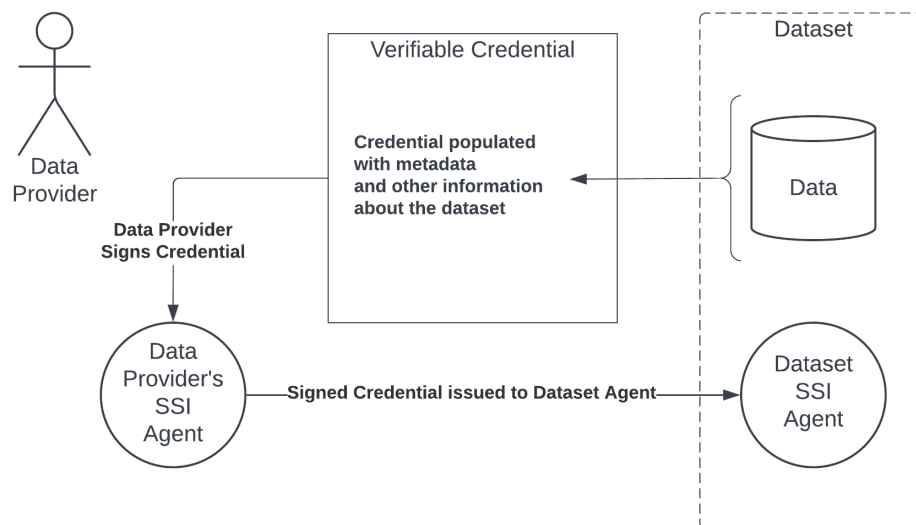


Figure 5.1: Data providers declare information about assets, and issue signed Verifiable Credentials.

To analyse this approach, we consider a setting in which a party responsible for a data asset wants to provide assurances on the asset's qualities to interested parties. In this case, the data asset is represented by DSA, a software agent capable of interacting through SSI protocols. The data provider is the credential issuer, and interacts with

their own software agent (via web page or mobile application) to issue credentials signed with their private key. As such, the data provider takes accountability for issuing claims about the dataset, as it is cryptographically provable that they were the issuing party. This provides a trust anchor in the system, in that anyone relying on VCs issued by the data provider to be assured of the qualities of the data asset, will need to have confidence in the data provider's reputation and expertise in order to place value on credentials signed and issued by the data provider.

To publish a new dataset, the data provider populates an instance of a VC schema with information about the dataset, and then issues a signed VC to the DSA, as shown in Figure 5.1. DSA runs as a software process, and its address and interaction protocols are made known to parties who might wish to access it. Sharing these endpoints would become part of the data publishing process, and be among the public information displayed on a web page about a dataset, for example.

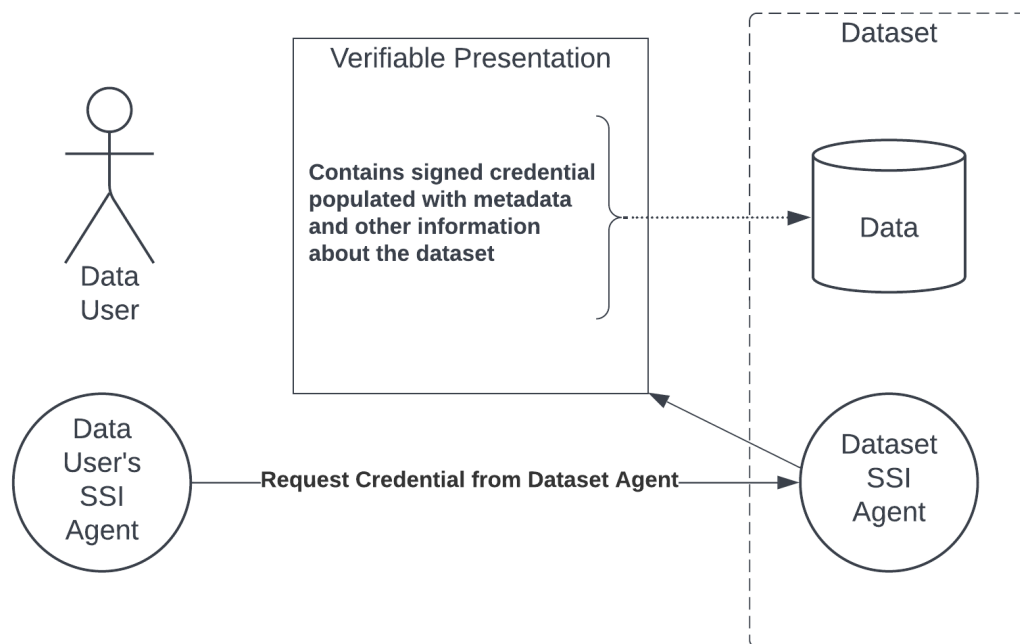


Figure 5.2: Data users request and verify claims about datasets.

Third parties interested in the dataset would be provided with instructions on how to

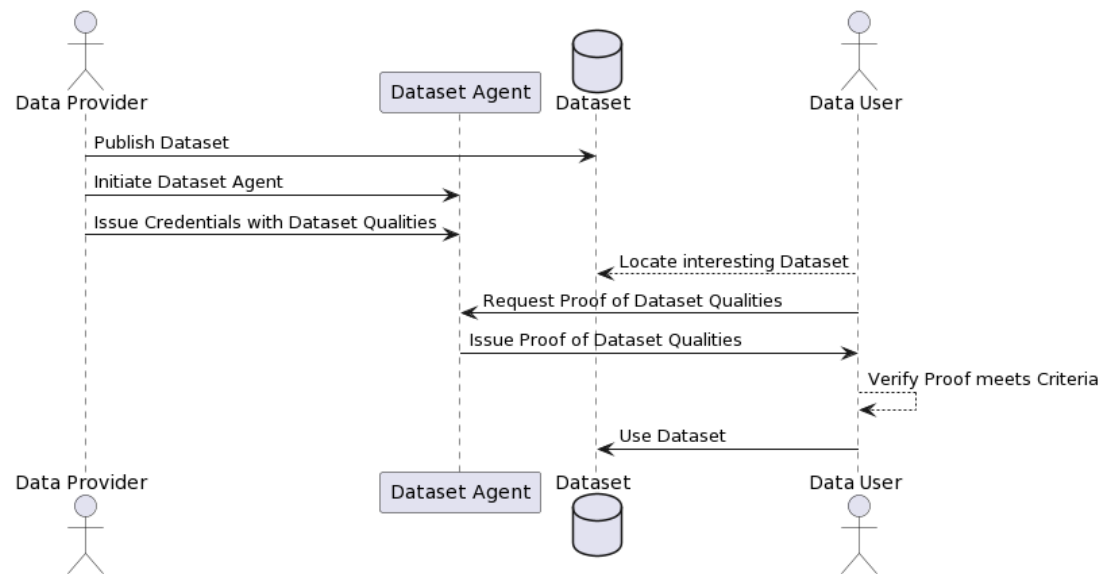


Figure 5.3: Process Flows in Publication and Verification.

request verifiable information about the dataset, in the form of a VP. They would be able to use the VP to inspect claims made about the dataset. To request a VP, interested parties would interact with DSA via its published endpoints, using their own SSI agents (either a cloud-based process, or a mobile application) to request proofs of credentials, as depicted in Figure 5.2.

The flow of interactions between entities in the ecosystem is shown in Figure 5.3, which illustrates the publication process undertaken by the data provider, and the process by which an interested party would request proof of claims made about the dataset from the DSA.

Demonstration and Evaluation

The DSRM approach favours implementation of design artefacts as part of the research process. This supports evaluation of technical risk and efficacy [165], which can be conducted in an artificial environment. In this design cycle, we instantiate the proposed solution for the scenario of sharing a scientific dataset, to provide technical validation of the approach.

As SSI sees adoption across different use cases, commercial and open source implementations of the protocols are becoming available. One such implementation is Hyperledger Aries [56], which provides an environment that supports development of SSI-based ecosystems using software agents. Each agent represents a party in the ecosystem, which could be a human or non-human entity. Agents are implemented as software processes, which provide endpoints on the network for communication with other entities. These endpoints can be used to issue VCs, request VPs, and to verify VPs and credentials they contain. The Aries framework provides secure storage of private keys and credentials through a digital wallet component embedded within each agent's process. SSI implementations developed on Aries can take advantage of infrastructure provided by the platform for core SSI functionality, and focus resources on developing interactions between agents necessary to provide the required application or business logic. An abstraction layer is provided by the Aries Cloud Agent Python (ACA-Py) toolkit [57], which provides Python language bindings to instantiate and manage Aries agents in the system. These agents are intended to run as software processes on servers, either locally within a network or on hosted cloud services. Building an implementation on the Aries platform provides an efficient means of demonstrating the design for providing assurance and accountability on shared digital assets.

In an example scenario, a city planner might seek to use a dataset shared by a mobile phone company to understand people's movements in the city. The planner should be concerned that data in the dataset had been collected ethically, with appropriate permissions from citizens, and that it didn't compromise the privacy of citizens by allowing individuals to be identified and tracked, for example. Here, the shared dataset is represented by DSA, which is implemented with the infrastructure provided by ACA-Py. As part of the on-boarding process for DSA, a configuration script generates a digital wallet, to hold its private keys and credentials, and creates a DID and an endpoint, through which DSA will be accessed on the network. The data publisher uses a software interface to cryptographically sign and issue credentials to DSA, using an Aries implementation of SSI protocols provided by the platform infrastructure. This provides

an implementation of the approach shown in Figure 5.1.

Credentials in the scenario are structured according to published JSON-formatted Credential Definition schema [94] and contain a set of key-value pairs which the issuing party asserts about the subject. Listing 5.2, *Hash of Data* shows a credential which contains the cryptographic hash of the dataset, which inextricably links the credential to the dataset it represents, and *Ethically Sourced*, which represents an ethical status of the dataset, as claimed by the data publisher. A practical scheme would hold other credentials, for metadata about the dataset, and could include conditions for dataset usage.

```
{  
  "Hash of Data": "0xFFEE...AA1122",  
  "Ethically Sourced": "YES"  
}
```

Listing 5.2: An Example of a Credential Set

The city planner, and any other researchers interested in the dataset, use an interface to DSA, facilitated by the Aries platform, to request verified proofs of qualities. These parties would use their own SSI agents to communicate with DSA, through its published endpoint. The SSI agents may be edge agents, such as a digital wallet application on a mobile device, or hosted cloud agents accessed through a web interface. Anyone knowing the published endpoint address for DSA can request proofs of credentials, and DSA will respond with a VP containing the credentials. In the implementation described, the credential includes the cryptographic hash of the dataset, which provides an inextricable link to the dataset it represented. The public DID of the data publisher is also included in the returned proof, providing accountability on assertions made about the dataset – in this case the claim that it was ethically sourced.

The demonstration implementation in this design cycle used VCs as a means to associate qualities with a dataset, as an example of a digital asset. These qualities could

include metadata, or any other information that a provider determines is useful in documenting their assets. The use of VCs and VPs to represent and convey this information provides a mechanism to link the qualities to the digital asset in a tamperproof, verifiable manner. As demonstrated, a scheme which enables parties responsible for assets to issue signed assertions about asset qualities, can be implemented using standards-compliant SSI software infrastructure, as provided by the Hyperledger Aries platform. All operations are performed using published protocols, with roles defined for parties to issue, hold and verify credentials. Using standardised SSI protocols to request and check the veracity of credentials provides a means for third parties to inspect and gain assurance on claims made about assets, in support of requirement R1. Furthermore, the party sharing assets is required to take responsibility for claims made and stored in VCs, underpinned by their cryptographically verifiable signature, with ownership provable by control of a DID. This can be used to provide accountability on digital assets, as recommended for DDS process improvement, such as the TRL proposed by Lavin, et al. [90], in support of requirement R2 from Table 5.1.

The use of software agents to store VCs and provide VPs has allowed the system to operate autonomously, without the need for human intervention to generate to and issue VPs. A limitation of this approach, however, is that DSA has to be configured to provide VPs in response to any request it could satisfy (i.e., if it held a matching credential, it would present it). This is in conflict with the identified requirement (R3) for providing protection to certain confidential or private information about assets. Design Cycle 3 seeks to address this limitation, and considers how to provide control over access to potentially sensitive information and digital assets, through the introduction of an access control subsystem.

5.3.3 Design Cycle 3: Policy Based Access Control

Revision of Objectives

Design Cycles 1 and 2 identified and demonstrated that SSI constructs and protocols can be used as the basis of a method to enable digital asset owners to issue signed documentation and claims about qualities of their assets, taking accountability for assets they provide. Third parties wishing to scrutinise asset qualities can request and inspect these claims, which are provided alongside proofs that they are as signed, and have not expired or been revoked. As identified in Table 5.1, a further requirement of a digital asset sharing architecture is that access to confidential or private information should be under the control of the asset owner. The objective of this design cycle is to extend the design approach of Design Cycle 2 to include an access control subsystem.

The multi-messenger astrophysics (MMA) research community provides an example of an ecosystem of participants from different organisations who need to share digital assets to meet individual and collective goals. In a review of requirements to support on-going progress in the MMA field, Chang, et al., [36] describe the environment as one which “requires diverse scientific teams to bring together their observational resources, data, analysis and modelling tools, and expertise to ensure the maximum scientific return” and identifies the need for collaborative groups to have “the ability to form flexible teams on short time scales, to share data, codes and other digital objects in real time, and to self-organise into spontaneous collaborations of varying scope”.

Data sharing in MMA is centred around observatories, who gather data from their telescopes. Most observatories list data policies on their web sites, and describe a range of data access conditions covering their research customers and the wider public. It is common for researchers to have a closed or proprietary period during which they have exclusive access to data they have requested from the observatory; beyond this proprietary period the data becomes openly available. Proprietary periods vary between observatories. At the East Asian Observatory, the James Clerk Maxwell Telescope

(JCMT), for example, operates on two semesters with data becoming openly available one year after the end of semester. Access to data from JCMT is tied to user accounts, with different accounts needing to be manually linked in order to gain access to new resources. The European Southern Observatory provides PIs (and their delegates) with exclusive access to their scientific data for a proprietary period of one year after the data is available. Liverpool John Moores University (LJMU) telescope data access policy provides insight into how data protection is implemented during a proprietary period and beyond: “science data are held in a password protected, online data archive. LJMU will provide the access password to the PI and Co-Investigators (CoI) named on the proposal application and to other individuals nominated by the PI. ... At the end of the one-year proprietary period the password protection will be removed and the science data will become publicly accessible.” In moving forward to support more flexible data sharing, the Rubin Observatory describes different categories of data products, including User Generated data products originating from the research community, which will “provide for users and groups to maintain access control over the data products they create, enabling them to have limited distribution or to be shared with the entire Rubin Observatory community”.

Implementing these differing requirements and data access policies currently requires intervention from observatory support staff, as well as placing administrative overhead on PIs in managing user accounts and access levels for their collaborators. As such, a situation exists in which researchers need to be able to share data resources of high value with flexibility, whilst maintaining data security. This provides a relevant case study to motivate multi-party data-sharing requirements, and discussion of the design and demonstration of an architecture that aims to meet these requirements follows.

Design and Development

A software architecture designed to offer controlled access to confidential or private information is shown in Figure 5.4. The proposed architecture supports two modes of

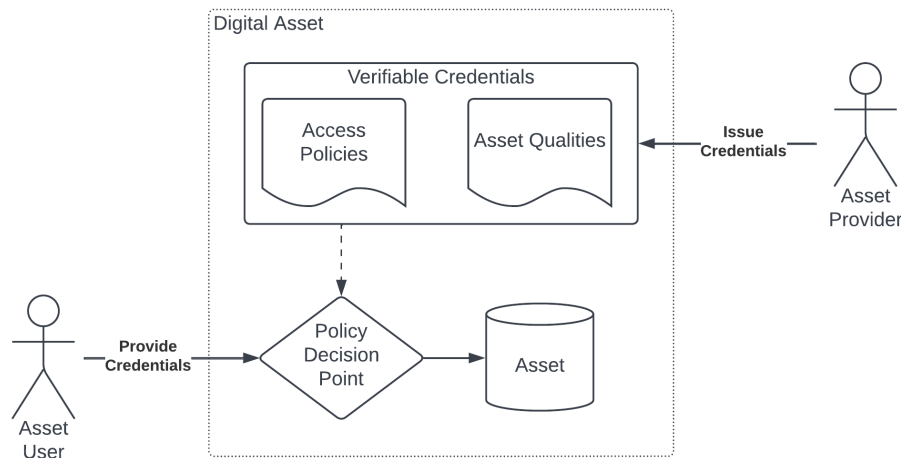


Figure 5.4: Architecture using Verifiable Credentials to provide quality assurance and access control for digital assets.

use, *asset publishing* and *mediated access to assets*. Asset Publishing is the approach developed in Design Cycle 2, and operates as described in subsection 5.3.2 to enable asset owners to make cryptographically signed claims about their assets, and in doing so, take accountability for the assets they seek to share.

Providing mediated access to digital assets requires the introduction of an access control subsystem. A possible strategy for access control is a model known as Policy Based Access Control (PBAC) [174]. This approach uses defined policies or rules to describe credentials that are required to be presented by an entity in order to access a protected resource. Such an approach to authorisation supports a zero trust security model [82], in that access is only provided to parties who are able to present the credentials requested by the access policies, and all other access is denied. The architecture proposed here uses SSI protocols to provide a PBAC scheme, via a component acting as a Policy Decision Point (PDP) [174]. The PDP mediates incoming requests from parties seeking access to assets, and determines whether to satisfy the request. The PDP provides digital assets with a representation of the decision-making process that a human actor would adopt when assessing whether to grant requests for informa-

tion. In an instantiation of the architecture, the PDP can be provisioned in a number of ways: it could offer a simple pass-through where all requests are granted, refer requests to a human supervisor, or enact a number of static or dynamic machine-driven policy checks.

Extending the architecture of Design Cycle 2, the proposed design follows the SSI paradigm, with each actor and entity represented by an SSI software agent. Table 5.2 shows a mapping between actors and entities of the case study scenario and elements of the architecture of shown in Figure 5.4.

Element	Use Case Mapping
Digital Asset	A shared scientific dataset
Asset Qualities	Metadata pertaining to a shared dataset
Access Policies	Rules mediating access to a shared dataset
Asset Provider	Principal Investigator (PI)
Asset User	Researcher with an interest in a shared dataset

Table 5.2: Mapping the Use Case Scenario to elements in the Architecture

Access to published datasets is controlled by a PDP, a system component which:

1. Intercepts incoming requests from researchers requesting access to datasets.
2. Identifies which dataset the researcher is requesting access to.
3. Determines whether the researcher has the rights to be granted access to the requested dataset.

An approach to the design of a PDP using SSI principles is that the asset publisher, here a PI, issues a VC containing access policies to an asset. The PI will also issue peer researchers with credentials which attest to their role on the project. The researcher's credentials are held in an SSI agent, or digital wallet. These access policies and status credentials are used in combination to determine whether a requesting party can be

provided with access to the requested asset. In the proposed design, the PDP employs an SSI agent process as an Access Control Agent (ACA). The ACA uses SSI protocols to request an Access Policy (AP) credential from the dataset’s agent (DSA), and then processes the AP to determine which credentials the requesting party needs to provide. The ACA uses SSI protocols to request a matching credential from the requester in order to fulfil their access request. If a suitable credential is provided, then access to the dataset is granted. If the request cannot be satisfied, then the access request is rejected and access to the dataset is denied.

The PDP can operate alongside a web server component, and mediate requests for access to the dataset, shown in Figure 5.5. Provisioning the PDP in this “sidecar” role [31] offers a practical approach to deploying PBAC into a data sharing environment, and can support existing working practices where data access is requested via a web browser or Jupyter Notebooks interface.

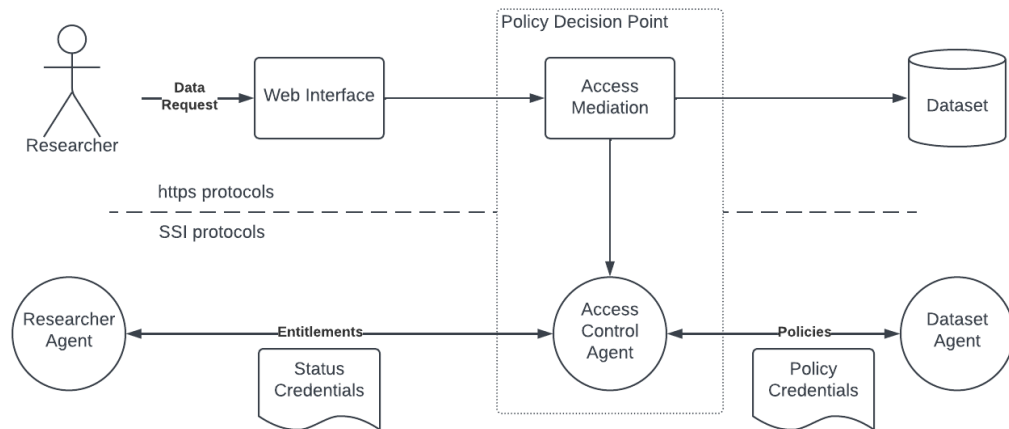


Figure 5.5: Access to Data in MMA Architecture

Interaction flows between the parties in the system are shown in Figure 5.6. An https server listens for incoming requests, and then determines which dataset is being requested. The Policy Decision Point component is activated, and makes an SSI protocol request via the ACA to the requested DSA to provide its AP credential. The PDP parses the returned AP credential and extracts the policy requirement. To enact the

policy, ACA makes an SSI protocol request to the researcher, and asks them to provide a VP that matches the policy requirements. If a credential is returned by the researcher, it is compared with the required value from the AP. On a successful match, the PDP allows the request for the dataset to proceed, otherwise the request is rejected. In this way, ACA performs the role of an SSI verifier, requesting and verifying VPs from other parties in the system in order to manage access to resources.

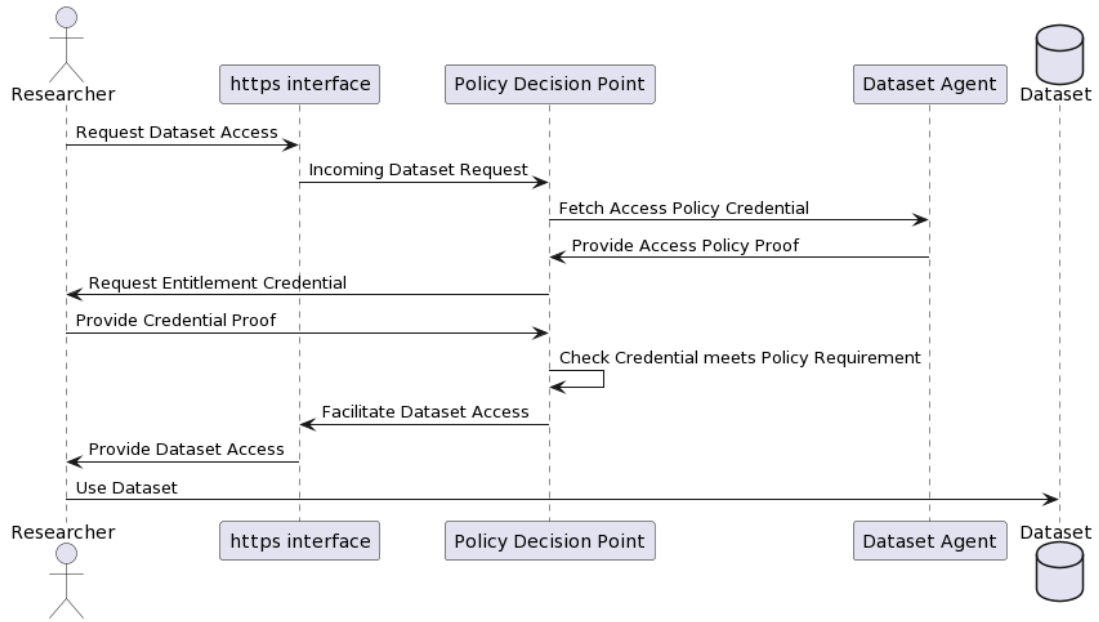


Figure 5.6: Interaction Flows for SSI Policy-based Access Control

Demonstration and Evaluation

Peffers, et al. [122] identifies demonstration of the use of IS artefacts developed through the DSRM framework in an example scenario as a suitable way to determine the extent that the artefact can provide a solution to the problem. The proposed artefact interacts with the context as a *treatment*, and building and demonstrating treatments based on the designed artefact provides an opportunity for evaluation of the artefact in the context of the scenario. In evaluation, we consider how effective the treatment would be – the treatment can be validated if it satisfies its requirements [169, p.59]. This approach

provides an instance of a “single-case mechanism experiment”, often performed in the laboratory to test an artefact prototype [169, p.64], and further contributes to a technical risk and efficacy evaluation [165]. Conboy, et al. [40], also advocate demonstration of the design approach in this way, as it shows not only that the design can be implemented, but also how it can be implemented.

The demonstration of Design Cycle 2, described in Section 5.3.2, used the Linux Foundation’s Hyperledger Aries SSI platform [56] and ACA-Py [57]. This provides a technical infrastructure, but does not readily map to application logic. To perform more effective and efficient modelling of systems, collaboration with colleagues at University of Notre Dame led to the specification and development of a higher level abstraction layer, operating above ACA-Py. This abstraction layer, named ‘Syndicate.id’, provides a framework for software engineers to efficiently implement SSI agents that represent entities in their environment. Syndicate.id uses the Go Programming language [49] to enable SSI credential exchange between agents to be performed with the ACA-Py framework, whilst the supporting application logic and interaction between agents is conducted through https-based REST interfaces. Figure 5.7 shows the layered architecture of the SSI infrastructure based upon Syndicate.id and Aries.

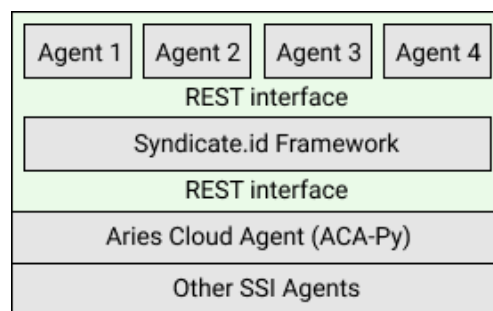


Figure 5.7: Architecture of Experimental Framework

For the demonstration, a simple web form interface was developed to support the tasks of the PI. An SSI agent was instantiated to represent the PI. This agent used the Syndicate.id framework’s https interface to populate the credentials in the system. An SSI agent was instantiated to represent a published dataset. This agent performed core SSI

tasks of accepting offered VCs and providing VPs on request, and no adaptation or custom code was required to support the scenario. A further SSI agent process was instantiated to represent a peer researcher. This agent had to accept VCs and provide VPs on request, and again no custom code or modification was required from a standard SSI agent implementation. Code was written to intercept incoming https requests, and interface with an ACA agent to request and parse access policies, and initiate and manage credential exchanges with researchers, implementing the design shown in Figure 5.7.

In SSI terminology, the PI had the role of the issuer, and the dataset and researcher were subjects and holders of credentials in the system, and ACA was a verifier, as detailed in Table 5.3 and illustrated in Figure 5.8.

Role	SSI Role	Implementation
PI	Issuer	Standard SSI Agent
Dataset Agent (DSA)	Subject/Holder	Standard SSI Agent
Researcher	Subject/Holder	Standard SSI Agent
Access Control Agent	Verifier	Extended SSI Agent

Table 5.3: Components of the Demonstration Scenario

To test the PBAC scheme, the demonstration used an AP that required that certain affiliations to be provided. This meant that any researcher requesting access to the dataset must be able to provide a VC that attests that they have a suitable affiliation. In the demonstration, the data value of the dataset was held as a VC by the dataset's agent. In order to retrieve the data value, ACA requested the value of the credential from DSA and then passed this value on to an authorised requestor, as a response to the original https request. In a practical system, access could be provided to a dataset through a one-time URL, or through integration with existing web-based authorisation schemes, such as oauth2 [65].

The access control subsystem was tested by using the PI's web interface to issue dif-

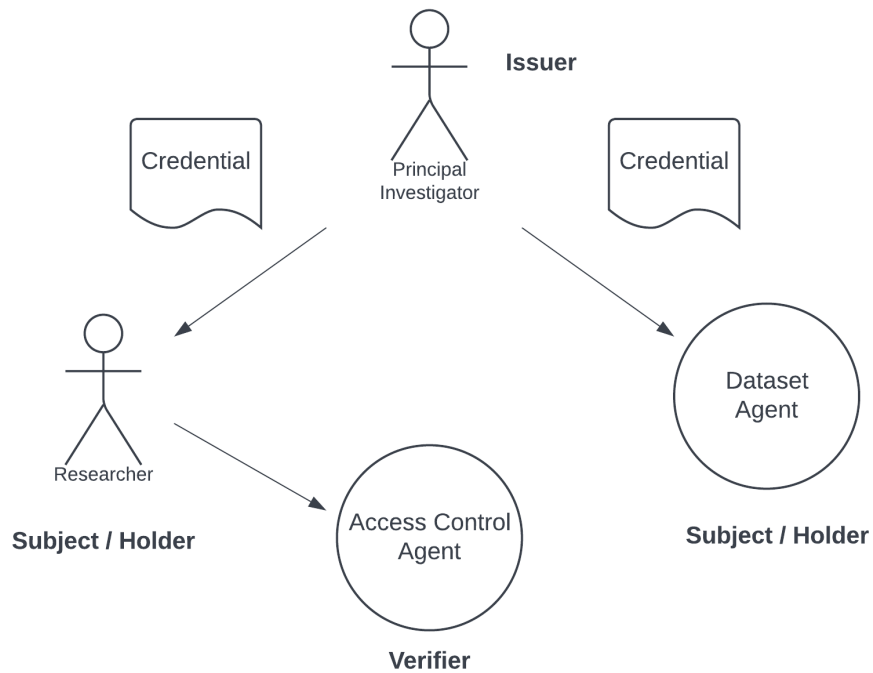


Figure 5.8: SSI Roles in the Demonstration Scenario

ferent values for the affiliation credential to the researcher, and by issuing different requirements in the access policy, and verifying that a match was required to provide the data value in response to the request, following Table 5.4. This demonstrated that multiple agents in the system could interact to change system behaviour dynamically, and that an AP issued as a credential to an agent could be accessed and enacted by another agent.

Requestor Role	Access Policy	Access Allowed
Student	affiliation=student	Yes
Professor	affiliation=student	No
Student	affiliation=professor	No
Professor	affiliation=professor	Yes

Table 5.4: Roles and Access Policies

To illustrate the self-sovereign nature of the solution, the agent representing the PI was shut down. This did not affect implementation of the AP, as the policy was held by the DSA, and so data could still be accessed according to the policy without ongoing involvement or availability of the PI in the system. A video demonstration of the system was made available¹ as part of the communication of the research.

The implementation of the MMA case study scenario focussed on the access control elements of Design Cycle 3, and on demonstrating that an SSI-based architecture could be used to provide a PBAC scheme to mediate access to shared resources. Using VCs for dataset access policies, and holding these in an SSI agent representing the dataset offers a novel approach to data security. Access policies are only available on request, in the form of VPs, and the demonstration showed that policies could be requested and enacted by an ACA agent, using SSI protocols. The ACA then interacted with the requesting party, without revealing the required value for the credential required by the policy, and only provided access when a suitable value was provided. Such a scheme provides an implementation of zero trust security architecture [82], as no party is able to access the dataset until they have provided the credentials specified in the access policy to access the resource.

5.4 Research Outputs

5.4.1 SSI-based Software Architecture Providing Assurance and Accountability on Digital Assets

A software architecture based upon SSI concepts and protocols has been designed through the 3 design cycles presented in this chapter. The architecture can provide verifiable assurance on asset qualities, placing accountability on parties making claims

¹<https://www.youtube.com/watch?v=4JBce6UM0wg>

about asset qualities. The architecture also provides information security support for digital assets across multiple stakeholders in an ecosystem.

Figure 5.9 shows a context view for a system architecture designed to provide assurance and accountability on a shared digital asset. A context view “defines the relationships, dependencies, and interactions between the system and its environment” [133] and is intended to show the people, systems, and external entities that interact with the architecture. The context view shows the role that the designed system has in providing parties interested in using digital assets with assurance on the qualities of the assets, engendered by information provided by parties responsible for the system and its components. Providing information about the shared assets places accountability on the parties providing the information. The system also provides a means for asset owners to define policies around provision of access to the shared assets, offering them control over access to confidential or private information.

Figure 5.10 shows the conceptual architecture for a system that provides assurance and accountability, and supports access control. The architecture adopts SSI principles, with software agents representing parties in the ecosystem, and exchanging VCs to attest to asset qualities, provision access control policies and provide evidence of access rights.

The contribution presented here provides a mechanism in which metadata and supplementary information about a dataset or other digital asset can be recorded in a structured, machine readable format. This information can be digitally signed by the responsible party. The use of standard’s compliant data models and protocols allows other stakeholders to request and verify the integrity of claims made, and the parties that are accountable for making such claims to be indentified. Previously published proposals, such as Datasheets for Datasets [58] and The Dataset Nutrition Label [70] do not provide mechanisms that demonstrate the integrity of information about assets, or place accountability on parties providing such information.

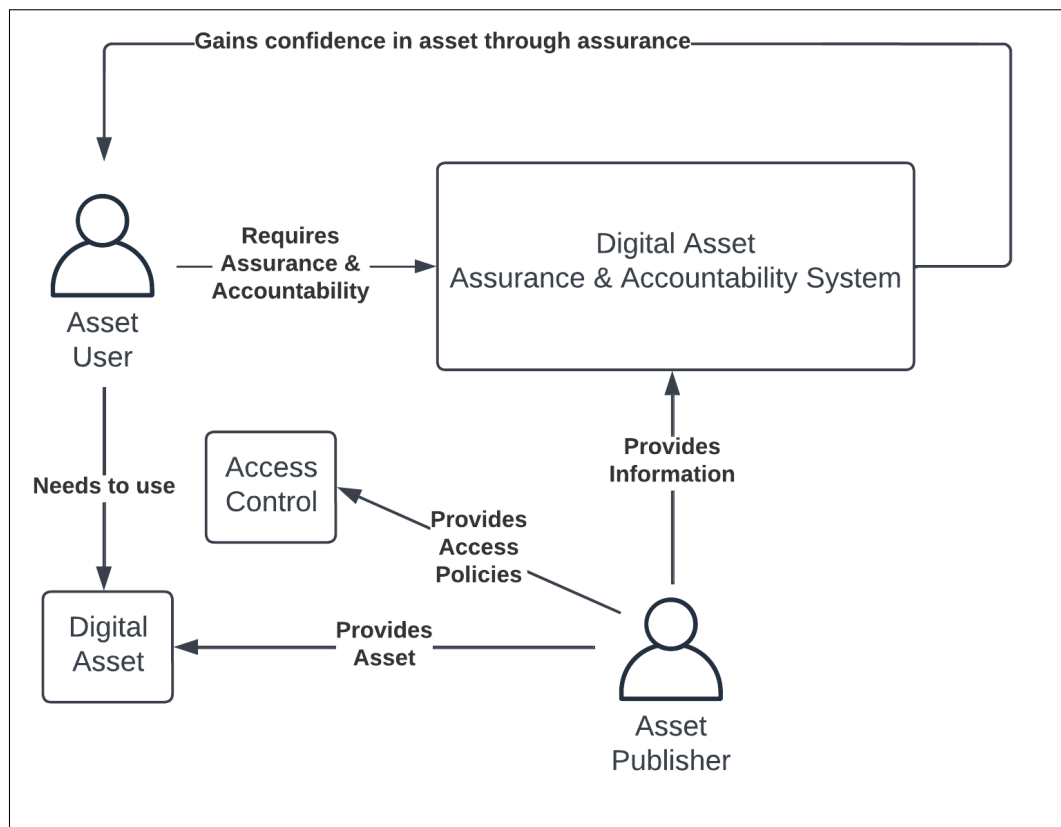


Figure 5.9: Context View of Digital Asset Assurance and Accountability System

5.4.2 Communication

Regular communication of the problem and the ongoing research work towards designing a solution has taken place in bi-weekly meetings with academic and industry peers as part of a research group within the DAIS-ITA project. Occasional formal presentations have also been made to a wider, international, group of peer researchers within DAIS-ITA. Dialogue with scholars and professionals through these forums propagated the work into the knowledge ecosystem, and resulting discussions fed into the design cycles, informing the objectives of the design and development stages in particular. The value of this form of continuous, “agile communication”, within internal and external teams is recognised as particularly valuable in DSRM by Conboy, et al. [40], in resisting “artificial equilibria”, and ensuring that ongoing relevance of the work is maintained.

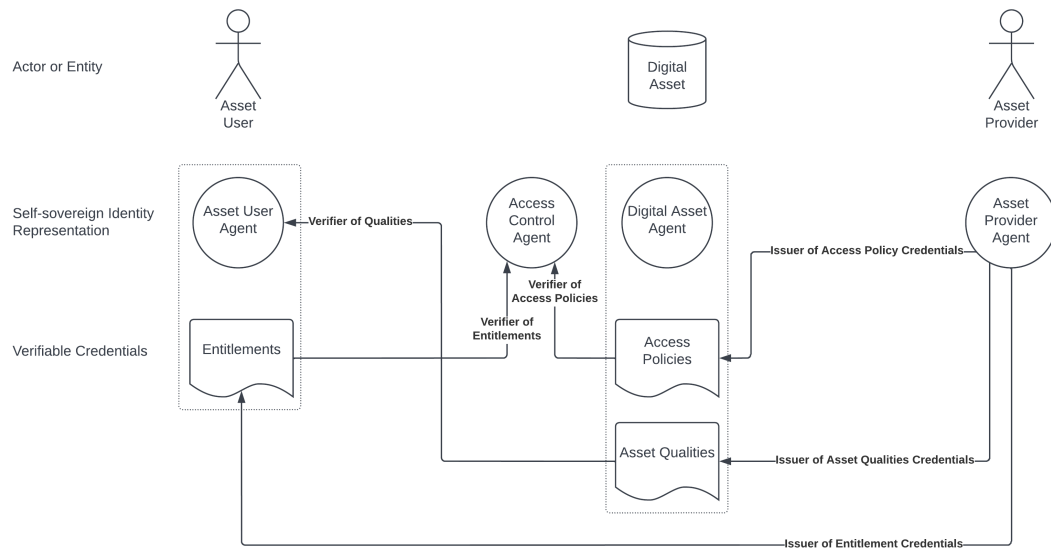


Figure 5.10: Architecture in support of Digital Asset Sharing Requirements

More formally, the research has been communicated through publication of peer-reviewed workshop papers [14, 15] and accompanying presentations, and journal papers [12, 16]. The publication *Certifying provenance of scientific datasets with self-sovereign identity and verifiable credentials* [14], has also informed development of published work by colleagues Radha, et al. [128], and Millar, et al. [103].

5.5 Evaluation

Section 5.4 described the contribution that the research developed through this chapter makes to the knowledge base, namely the design of a software architecture providing digital asset assurance, accountability and information security, and communication of the work. Demonstration activities based on artificial scenarios have provided a focus for a technical risk and efficacy [165] approach to formative evaluation of the artefact, through each design cycle. To conclude this design episode, a summative evaluation is performed on the artefacts resulting from the design process.

5.5.1 The Rigour Cycle: Software Quality Review

In evaluating software architectures as design artefacts, Venable, et al., FEDS framework [165] for evaluation in design science research and Wieringa [169, p.54] guide us to consider dimensions of software quality as evaluands. Mistrik, et al. [106], consider quality to be a “fundamental property of software systems” and describe it as “the degree to which a software system lives up to the expectation of satisfying its requirements”. The international standard, ISO 25010² (which superseded the better known ISO 9126³) provides a taxonomy of characteristics for software quality, and identifies *functionality*, *efficiency*, *compatibility*, *usability*, *reliability*, *security*, *maintainability*, and *portability* as the factors by which software quality can be determined. Within each of these dimensions, the ISO 25010 quality model further identifies properties that are suitable for use in an evaluation. We choose to consider *interoperability* and aspects of *portability* as worthy evaluands, as they form part of the motivation towards using SSI protocols in the solution design. We also consider *security* as being relevant for evaluation of the architecture, as information security forms part of the solution requirements defined in Table 5.1.

Interoperability and Portability

The proposed architecture adopts SSI data models and protocols, and builds upon the notion of digital assets being self-sovereign and represented by an SSI software agent. We seek to leverage characteristics of interoperability and portability afforded by the adoption of SSI standards. As such, it is appropriate to evaluate the proposed design approach against the principles of SSI, in order that the design does not diverge from those principles, and the protocols and standards that embody those principles. Following a structured literature review and interviews with members of the SSI community, Sedlmeir, et al. [139], identify ten design principles for SSI-based systems. These

²<https://www.iso.org/standard/35733.html>

³<https://www.iso.org/standard/22749.html>

principles can be used as a benchmark by which to evaluate our architecture. We have, however, replaced Sedlmeir and colleagues' Privacy guideline with Confidentiality, which we feel has more relevance in commercial settings, such as digital asset sharing. An evaluation of the architecture's performance against the design principles is summarised in Table 5.5, with commentary below.

SSI Principle	Approach
Representation	Human actors, and agents for data assets
Control	Humans control own identity, agents for data
Flexibility	Interoperable credentials employed
Security	Provided by underlying platform implementation
Confidentiality	By protocol design, and supported by PDP component
Verifiability	Supported by underlying SSI protocols
Credibility	Supported by underlying SSI protocols
Authenticity	Credentials need to be bonded via hash or id
Governance	Ecosystem needs governance policies
Usability	Web interfaces and digital wallets for interaction

Table 5.5: SSI Principles as manifested in the System Architecture

The SSI Design Principles [139], and their embodiment in the architecture, are as follows:

1. *Representation: SSI can represent any entity digitally*

The architecture uses SSI constructs for the human actors in the system, and represents data assets with SSI-capable software agents. Implementation of the PDP, for mediating access control, could be via a web interface to an SSI as demonstrated in the example, or on each dataset agent.

2. *Control: Only the holder has decision making power over their identity*

Human actors control their own data, such as credentials required to access datasets. The MMA use case scenario demonstrated that data assets can hold access

policies and mediate access through an Access Controller, as an example of a PDP.

3. *Flexibility: Adoption of interoperable standards, without vendor lock-in*

Interoperability is a key benefit of the proposed approach, and the implementation of the architecture in the MMA scenario demonstrated that standard digital wallets could be used for human actors and data assets.

4. *Security: State of the art cryptography, and end-to-end encryption*

The architecture is designed to be implemented using an SSI technology platform, either from a commercial vendor or open source. As result, any implementation will build upon security mechanisms in the platform provided for key management and end-to-end encryption.

5. *Confidentiality: Only essential data is revealed in each interaction*

This is an important aspect of the solution that is provided by the underlying SSI platform, which adopts principles of minimal information disclosure. The designed architecture augments this, by explicitly identifying a PDP for data assets, such that measures can be put in place to determine access rights according to policies held within the asset's agent.

6. *Verifiability: Issuers provide credentials that can be automatically and efficiently checked for their correctness*

Dataset quality certification is provided through VCs, which are issued by PIs and verified by research peers. The architecture takes advantage of SSI platform protocols and implementations to provide this functionality.

7. *Credibility: Validity and timeliness of claims can be checked*

This is provided by the SSI protocol and platform implementation, which should support revocation and expiry of VCs such that invalid information can be identified, and data freshness can be assured.

8. *Authenticity: Claims are bonded to their holders*

This is a design choice to be made when specifying schema layouts for VCs. It is possible to include a unique identifier into a VC, such that VCs are bound to the entity that presents them [64]. For a data asset this identifier might be a cryptographic hash, for a human it might be something that ties to an external identifier such as a photo id or a biometric marker.

9. *Governance: Guidance that helps verifiers to decide which issuers to trust*

This is a design choice that would be made for any individual ecosystem that implemented an SSI-based solution. Governance policies determine how participants in the ecosystem are expected to act, and identify the trust anchors in the system.

10. *Usability: User-friendly interfaces, as well as scalability. Minimum downtime, high performance, and streamlined processes should also be achieved.*

This strongly relates to the design and implementation of an individual system, and determines which components are used and how users interact with the ecosystem. Platform infrastructure choices will determine scalability capabilities.

Information Security

Security is a software quality attribute, identified by Ozkaya, et al. [119], as a “top-level quality attribute”, and is of particular importance in DDS, where new attack surfaces are found [62, 102, 167]. Information security can be considered through application of the STRIDE threat modelling technique developed at Microsoft [142]. STRIDE is an acronym, which motivates consideration of threats to a system across common attack vectors: Spoofing, Tampering, Repudiation, Information disclosure, Denial of Service, and Elevation of Privilege. Applying a STRIDE analysis to the proposed system architecture provides a description of the possible threats to the system, and any mitigation in the design.

- **Spoofing:** DIDs and VCs are based upon asymmetric cryptography, in which participants use private keys to provide proof that they are an authorised party. To provide a trustworthy data-driven system, it is important to be able to place accountability on parties that make contributions to the system, so that they are answerable for any claims that they make about the contributions they provide. The SSI-based approach uses asymmetric cryptography to include digital signatures into verifiable credentials as evidence that the signing party attests to the integrity of the claims being made. Stakeholders can trust the claims made if they trust the party that signed the claims. This trust system breaks down if an attacker is able to spoof the identity of a trusted party, and sign claims on their behalf. The integrity of the system is dependant on participants maintaining control of their private keys, which needs to be supported through education and policy.
- **Tampering:** Data, which can include metadata or Access Policies, can be held as VCs which will be encrypted and stored in a secure digital vault or digital wallet as part of the SSI infrastructure. Use of encryption and digital signatures provides proof that components and data within the system have not been tampered with, and is provided by SSI protocols and infrastructure. Verifiable Credentials contain claims and digital signatures which protect the integrity of the claims made. An attacker might wish to change the information held in the credential to make false claims about a data asset. However, any attempts to tamper with the data and provide misinformation would be identified by the protocols which check the content of the claim with the corresponding digital signature. If the claim data has been changed, the signature will be not be valid when it is checked. The protocols used in the architecture are designed to prevent such attacks.
- **Repudiation:** Access to data is consent-based, and parties must request access to information in the system. Access requests can be logged for future audit,

supporting non-repudiation. Where digital signatures are used to demonstrate accountability for claims made about assets in a DDS, a party may subsequently deny having had access to the keys that identify them as the signing party, or may claim that the keys have been stolen in order to try to deny having responsibility. A claim of this nature would need to be investigated by their employer, or in some cases by the police.

- **Information Disclosure:** Access policies and credentials with metadata and other information are held privately by the digital asset's agent, so that third parties are unable to determine which access policies are in place prior to making a request. They are required to provide a value for a named attribute, but they are not informed as to what the acceptable values for that attribute are, preventing leakage of information. Self-sovereign Identity protocols are designed to provide privacy, and access to any data is based upon consent being granted to the party requesting access by the party controlling the data.
- **Denial of Service:** Through adoption of a decentralised paradigm, SSI avoids a central point of failure. Each asset controls access to its own credentials through open protocols. As such, there is no central point in the system which could be compromised to create a denial of service attack. Individual entities could be targeted, but in large quantity this would be expensive for attackers and require coordination to cause widespread disruption.
- **Elevation of Privilege:** Access to information is not granted unless criteria defined in access policies are met. Following the Principle of Least Privilege [135], the PDP ensures that access to protected assets is prohibited unless the requesting party is able to provide the appropriate credentials, resulting in a zero trust environment.

Analysis of the proposed architecture against well-regarded software quality attributes of interoperability and portability, and information security has determined that the

approach based upon SSI principles and protocols, provides strong support for meeting requirements.

5.5.2 The Relevance Cycle

The relevance of a designed artefact can be evaluated by considering it in the context of the environment in which it is to operate, and considering how effectively it would address problems identified during the scoping phase, as discussed in Section 5.2.

Table 5.1 identified requirements for a solution to providing accountability on parties sharing digital assets, and giving assurance on qualities of those assets, whilst offering protection of confidential information. Discussion of how each requirement is met through the proposed architecture follows.

R1 Provide assurance on qualities of digital assets

SSI protocols allow responsible parties to specify, populate and sign digital credentials. These credentials can be used to make claims about qualities of a digital asset, such as metadata for a dataset. DIDs and asymmetric cryptography provides proof that the claims were made by the signing party, and have not been tampered with subsequently. This can provide a verifying party with trustable assurance on the stated qualities of an asset, based on the reputation of the signing party.

Design Cycles 1 and 2 demonstrated that a PI could issue signed claims about digital asset qualities as VCs. Third parties can request and inspect VPs, and verify that they were signed by the PI and not tampered with. The scenarios used in the demonstration phase of the design cycles enabled testing and evaluation of the technical risk and efficacy of the proposed design, and demonstrated that requirement R1 is met.

R2 Provide Accountability on Digital Assets

Adoption of SSI protocols in the architecture of Figure 5.10 provides mechan-

isms for responsible parties to specify, populate and cryptographically sign digital values as VCs. The use of DIDs and asymmetric cryptography creates a verifiable linkage between claims made in a VC and the signing party. Parties wishing to verify asset credentials, and who the claims in VCs are signed by, can request a VP from the asset's agent. This VP provides cryptographic proof that the signed claim was issued and has not been subsequently tampered with, and places accountability for making such claims onto the signing party.

Design Cycles 1 and 2 demonstrated that a responsible party could define and issue signed claims about digital asset qualities as VCs. Third parties are able to request and inspect claims made as VPs, and can verify that they were signed by a key under the control of the signing party, and had not been tampered with or revoked. The artificial scenarios used in the design cycles enabled us to test and evaluate the technical risk and efficacy of the proposed design in providing accountability on claims made about digital assets. Thus requirement R2 is met by the architecture.

R3 Protect confidential information

SSI is based on the principle of self-ownership of data, with a VC stored in a secure digital wallet under the control of the entity which owns it. In the proposed architecture, VCs for digital assets are held by a software agent, with secure storage provided by an SSI platform implementation. In the design, a PDP is used to mediate and control access to information. One way to implement the design is to use policies issued to the digital asset to control access to resources.

In Design Cycle 3, and the MMA case study scenario, we demonstrated that a PI could use VCs to issue and store access policies for digital assets. The PDP in the implementation used these access policies to control access to resources, based on credentials held by accessing parties. In this way, requirement R3 is met.

Section 3.4 defined technological rules for the properties of *scrutability*, *verifiable*

oversight and *accountability* in a data-driven system (Definition 2). Here, *scrutability* is provided for an individual asset, such as a dataset, by publishing a mechanism for parties to request information about the asset. *Verifiable oversight* is provided, as claims are stored in a Verifiable Credential data structure containing a digital signature. This signature can be used to verify the integrity of the information within the credential (i.e., that it has not been modified or tampered with). The Verifiable Credential data model and protocols also support the provision of *accountability*, as a public key is used to validate the signature provided with a claim, providing proof that the claim was signed by the associated private key. It should be noted that an external governance system or registry would be required to provide an irrefutable link between any individual and control of a particular public and private key pair.

5.5.3 Limitations

The architecture developed through the design cycles of this chapter and presented in Section 5.4 uses SSI concepts to provide assurance, demonstrate accountability and offer protection to shared digital assets. The subsequent evaluation has shown that there is value in adopting this approach in circumstances where assets need to be shared between parties. Many of the anticipated benefits come as a result of the underlying SSI platform's implementation of standards, and currently such platforms are at an early stage of development and deployment. There are still challenges to overcome in terms of operationalising frameworks such as Hyperledger Aries, for example, and in provisioning and deploying software agents to represent each entity of a system.

The evaluation of the architecture presented in this chapter has considered artificial scenarios, and has concentrated on technical risk and efficacy, following the FEDS approach [165]. FEDS determines that an artificial, technically focused approach is suitable for evaluation where testing with human users is infeasible, and deployment is far into the future. Deployment of the proposed architecture is dependant on provision of SSI infrastructure, and will be most likely to succeed when SSI is already adopted

for other use cases, such as personal credentials. This would be an appropriate time to perform a human-centered evaluation on the approach, as users would have familiarity with the core concepts and supporting technology. The COVID-19 pandemic stalled engagements with researchers in the MMA field, which might have otherwise allowed a more naturalistic evaluation.

5.6 Summary

This chapter has considered RQ3, which sought to determine how SSI could be used to provide assurance and accountability on claims made about qualities of digital assets from different participants in a DDS, whilst maintaining confidentiality and information security requirements of asset providers.

The problem is addressed through the design of a software architecture presented in Section 5.4. The proposed architecture employs decentralised, self-sovereign credential technologies to bring accountability to digital assets, by enabling parties responsible for publishing and sharing such assets to make signed assertions about asset qualities. By using asymmetric cryptographic protocols, these signed credentials provide immutable evidence that a particular party has made a claim about an asset, and gives oversight on accountability being taken for asset qualities. A further aspect of the architecture is the use of a PDP that mediates requests for access to assets. This provides the asset owners with a mechanism to control access to information that is shared about their assets, enabling them to create policies that can be used to protect sensitive commercial information.

In demonstrations of the proposed software architecture, a required stage of the DSRM framework followed through this thesis, the design approach was applied to a scenario based on data sharing requirements of the multi-messenger astronomy research community. The implementation used VCs to hold signed and verifiable metadata about data assets. Access policies were defined for a dataset, and issued as VCs, which

provided rules by which a PDP component could mediate access to the dataset. The implementation demonstrated that certain components (SSI agents for datasets and researcher) could be used without any customisation or modification, as requirements for storing and presenting the required credentials were supported by the core SSI platform, promising interoperability with other systems in the future. The MMA scenario offered a demonstration of the viability of the proposed architecture, and the role of SSI in providing verifiable oversight and accountability on digital assets, as well as a means of protecting data from unauthorised access through the adoption of policy based access control. A benefit of the Syndicate.id framework is that in abstracting the SSI layer away from the application logic, it is possible to change the underlying SSI technology platform if required in the future.

Further research will be required to understand how such an architecture can scale to support digital assets in volume. This will require definition of a supporting environment, to manage such shared assets, as well as experimentation on performance characteristics. This will determine the design of components in the system such as the PDP, which will increasingly benefit from being decentralised as the system scales, such that it does not become a bottleneck or a point of failure. Whilst this will bring benefit, it will also require customisation, and will potentially introduce complexity to the digital asset's agents - which are currently implemented as standardised SSI agents. Understanding this trade-off will be most beneficial as SSI platform maturity improves and deploying such systems becomes simpler.

Providing Verifiable Oversight on Data-driven Systems

6.1 Introduction

This chapter builds upon research introduced in Chapter 4, which presented a proposal to develop and maintain a bill of materials record for data-driven systems as a means to provide oversight across contributions to systems, and designed a schema to support representation of a DDS BOM in a machine readable form. We also adopt and extend the architecture designed in Chapter 5, which used signed digital credentials in an SSI infrastructure to provide verifiable accountability on digital assets. Here we consider how a BOM can be used as the basis for providing a verifiable and accountable record for a DDS as a whole. In doing so, we seek to address RQ4, which asks “*How can SSI models be used to provide verifiable oversight and accountability to DDS, such that systems can be scrutinised by authorised stakeholders?*”

The research contribution of this chapter is a method for providing verifiable oversight on a DDS, which can assign accountability onto providers of contributions to the DDS. The method is demonstrated by an implementation of a web-based tool, the AI Scrutiniser, which illustrates a scenario that allows Domain Authorities and other stakeholders to scrutinise DDS.

Once more the research is presented as a series of design cycle iterations, which are de-

scribed in Section 6.3. The research outputs of the chapter are described in Section 6.4, and evaluated in Section 6.5. A summary of the chapter is provided in Section 6.6.

6.2 Problem Identification

The review of related work in Section 2.2 outlined proposals from researchers to provide documentation of DDS contributions in various forms, in order to support DAs and other stakeholders in gaining oversight onto systems. Analysing this work, Section 2.5 identified a gap in current provisions for viable technical solution designs for managing information visibility and providing transparency and accountability in complex multi-stakeholder DDS. This gap is problematic, as deployment and use of DDS in high-stakes settings requires that Domain Authorities and other stakeholders are able to gain oversight and develop confidence in systems, such that they can form an opinion on the suitability of the system for use in their domain.

A solution to this problem should meet the following requirements, summarised in Table 6.1:

- *Provide verifiable oversight and assurance on DDS*, by enabling parties to inspect and verify claims made about a DDS contributions.
- *Provide accountability on DDS contributions*, such that parties using DDS can identify the party responsible for making claims about qualities of contributions.

Requirement	Description	Design Cycle
R1	Provide verifiable oversight and assurance on DDS	1,2
R2	Provide accountability on DDS contributions	1,2

Table 6.1: Requirements of a Solution to the Problem

In previous chapters of this thesis, a software architecture that can provide verifiable metadata about an individual digital asset, and identify the party that takes accountability for claims made about the asset has been designed. We have also developed a framework to help identify roles and contributions to a DDS, and designed a data schema that can be used to structure a BOM document to record the supply chain of contributions to a DDS, tracking models and datasets that constitute a system.

We now extend this work and seek to use the artefacts in combination, in the design of a method which can provide a verifiable record of contributions to a complete DDS, giving oversight and identification of accountable parties. The approach, which is once again developed through design cycle iterations of the DSRM framework, considers how SSI can be used in combination with the BOM JSON Schema designed in Chapter 4 to provide oversight and scrutability on DDS to stakeholders. A demonstration of the method is provided as a web-based tool, which enables a Domain Authority to seek oversight on a DDS.

6.3 Design and Build

6.3.1 Design Cycle 1: A Verifiable Bill of Materials

Definition of Objectives for a Solution

A robust and reliable architectural model needs to be designed in order to provide support for a verifiable record of the supply chain for a DDS to be maintained. This will help DAs and other stakeholders gain confidence in a DDS, by providing oversight, and the ability to perform necessary and ongoing scrutiny on the system. This architecture should provide those scrutinising a DDS and its contributing elements with evidence of accountability, confidence that claims are made by authorised and trusted parties, and that records are accurate and up-to-date. The philosophy and technological approach of a decentralised SSI-based system again offers a technical constraint on which the

solution is designed, in order to test the hypothesis of this thesis, presented in Section 1.3.

Chapter 5 provided the design for an architecture based on SSI principles to provide oversight and accountability on discrete digital assets. Here we extend that architecture, so that it can be used for multiple digital assets across a DDS.

Design and Development

In an extension of the architecture presented in Figure 5.4, we consider that a software agent can be used to represent a whole DDS as an entity. This agent will hold a BOM document describing the components used in the DDS. The BOM will reference models and datasets used in the DDS, employing the JSON Schema developed in Chapter 4 to define its structure.

To provide assurance of the integrity of the DDS BOM document, it can be issued as a VC by the party accountable for the DDS. As discussed in Chapter 3, this is the Systems Integrator providing the DDS. The BOM VC will be issued to, and held by, the software agent representing the DDS. Any party that needs to examine the composition of the DDS can request the BOM VC from the DDS software agent. The returned BOM can be read and deconstructed into representations of individual assets which comprise the DDS. The *verificationRoute* field of the asset descriptor in the BOM JSON data can be used to provide a route, or endpoint, to access an SSI agent representing other assets in the DDS. This provides a means to determine which party is taking accountability for each asset, and to verify claims made about individual assets.

Figure 6.1 shows three roles from the RB Framework developed in Chapter 3: an SI, a Model Engineer and a Data Provider. These roles represent the parties that are accountable for components in the system. SSI software agents represent the digital assets: one agent representing the DDS itself, with further agents representing each model and dataset used in the DDS. The BOM of the DDS and of each asset describes

the properties and relationships between these entities, and is digitally signed by the party issuing the BOM, and as a result, taking accountability for the BOM of the system or component. A further participant in the system is a verifying party - this represents an entity that makes enquiries about the DDS or its components. In the RB Framework, this is the DA role – the party that seeks oversight on the DDS. The DA will interact with the DDS agent through a published interface, and be able to request and process the DDS BOM credential. Figure 6.1 provides an overview of the process of the DA seeking verifiable oversight on a DDS, by requesting the BOM VC from an asset's agent (via a known endpoint), processing the returned VP, and then requesting the next VC in the chain. This enables the DA to develop an overview of the assets in the DDS as they proceed through the supply chain.

Demonstration and Evaluation

The method can be demonstrated by application to artificial scenarios. Initial consideration is given to Scenario S1 (Definition 1), a DDS providing CCTV monitoring services described by Preece, et al. [125]. This scenario has been presented in the RB Framework in Figure 3.7, and its components described in Table 4.3 and the code Listing C.2. Here, the DDS is modelled using SSI agents, with VCs used to hold a BOM for each asset.

The environment, which is based on Figure 6.1, is described first from the point of view of its initialisation, with parties providing verifiable documentation for their assets. Then the act of seeking oversight and verification on the DDS and its contributions is discussed. Table 6.2 provides a summary of the entities and the credentials issued.

The setup for the system requires the following steps:

1. SSI software agents are instantiated to represent the digital assets in the DDS – namely, the DDS itself *CCTVMon*, the dataset *CuratedUCF101*, and the model *VADER*.

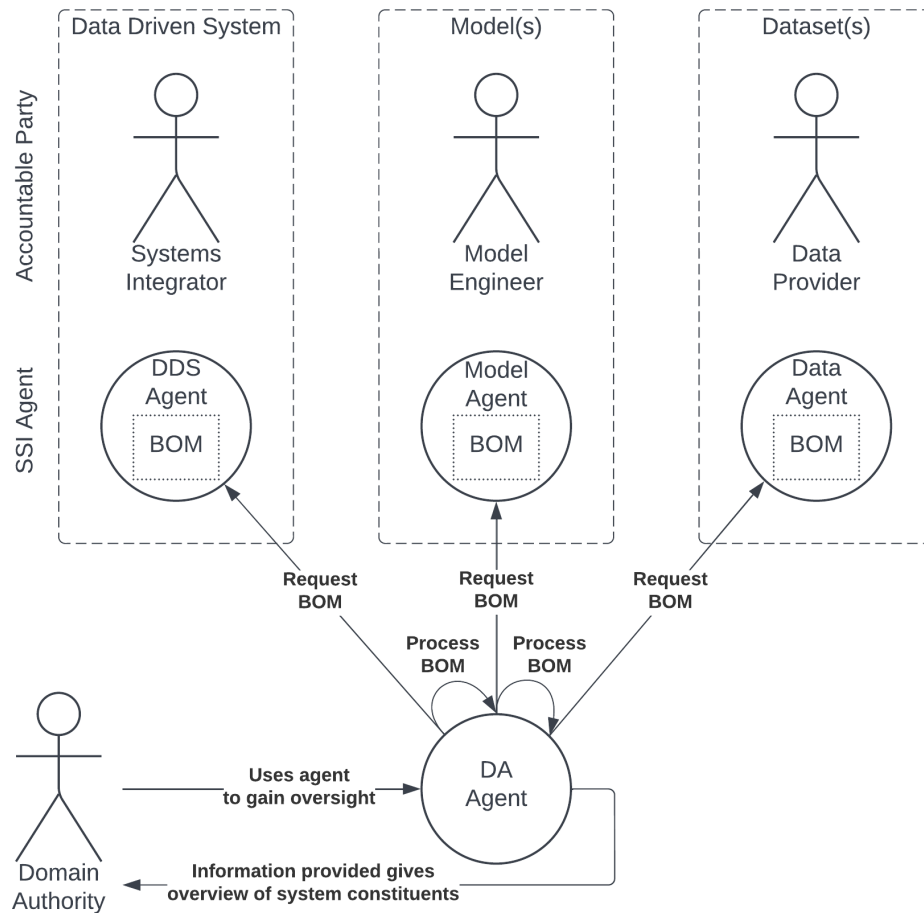


Figure 6.1: Domain Authority Interactions with SSI Agents of Constituents

2. Human actors are equipped with SSI wallet capabilities, the actors in this scenario are: *Hiley*, the DP; *Taylor*, the ME; *AnglovaChief*, an officer within the Anglova Law Enforcement, as the SI; and *UKAnalyst*, a DA who seeks oversight on the system. Hiley and Taylor are each accountable for a digital asset used in the DDS, and AnglovaChief, as SI, is accountable for the overall DDS. UKAnalyst is a verifying party – they will not issue credentials, but need to be able to access VCs for to have oversight on the DDS.
3. Following the process outlined in Section 5.3.2 for a single asset, Hiley and Taylor populate, sign and issue VCs about the digital assets that they are account-

Issuing Actor	Role	Credential	Issued to Agent
Hiley	DP	Dataset Qualities	CuratedUCF101
Hiley	DP	Dataset Bill of Materials	CuratedUCF101
Taylor	ME	Model Qualities	Model
Taylor	ME	Model Bill of Materials	Model
AnglovaChief	SI	DDS Qualities	CCTVMon
AnglovaChief	SI	DDS Bill of Materials	CCTVMon

Table 6.2: Credentials issued to SSI Agents in CCTV Monitoring Scenario

able for, and issue the VC to the asset’s SSI agent, as illustrated in Figure 5.1.

4. The SI, AnglovaChief, creates a BOM document for the DDS they have assembled, using the JSON Schema developed in Chapter 4. The BOM document is used to populate a VC, which is signed and issued by AnglovaChief to the CCTVMon agent, which represents the DDS itself.

To achieve verifiable oversight on the DDS, the process followed is:

1. UKAnalyst uses software that supports SSI protocols to make a request for the BOM Credential from the CCTVMon agent, via a published address or endpoint.
2. To satisfy this request, the CCTVMon agent issues a VP, which UKAnalyst uses to verify that the BOM Credential was issued and signed by AnglovaChief, and has not been tampered with subsequently – this puts accountability on AnglovaChief for its content. As such, if UKAnalyst trusts AnglovaChief, then UKAnalyst can put trust into the BOM credential that AnglovaChief has signed and taken accountability for.
3. The BOM credential adopts the JSON Schema previously defined, and can be read and interpreted by UKAnalyst. It will contain a list of the digital assets

that have been used to develop the DDS, with endpoints at which verification for each can be made.

4. UKAnalyst iterates through this list, and requests a BOM credential from each endpoint.
5. UKAnalyst uses the returned BOM credential to gain assurance on the veracity of each asset, and to identify the accountable party.

As a result, UKAnalyst can gain oversight over the supply chain of the entire DDS.

A similar method can be employed for Scenario S2 (Definition 1), the Aurora Chatbot, with entities shown in Table 4.4. This scenario needs further consideration, as the DialogFlow framework and the model sub-components of the DDS are provided by Google. The approach needs to consider how oversight on the system can be provided without the (likely unfulfillable) requirement for an organisation such as Google to contribute and run an SSI agent. A further difference in S2 is that CM adopts the roles of both SI and DP, having curated the conversation data, used it to train models provided by Google, and then produced the final system to offer to health agencies (HA). In this case, HA is in the role of DA, as the party seeking oversight on the DDS.

To support this scenario, an assumption is made that Google will not issue VCs to express qualities of services, and so CM is the only human actor. CM can take accountability for the datasets in English and in Portuguese, and for the DDS itself – which includes choices made during integration, including the use of DialogFlow. No identifiable party is able to take accountability for the Google components, but metadata about them – version numbers, internet locations of assets, etc. – can still be supplied as part of the DDS BOM. Indeed, the BOM Schema described in Chapter 4 provides structures for this type of information to be conveyed in the BOM for the DDS – a BOM can provide attributes of contributions through the *knownDependencies* structure. Furthermore, a JSON object can be created to represent the Google services, however its *verificationRoute* field will be empty, as there is no accountable party able to verify the

```

1   $BOM = DDSAgent.BOM$ 
2   $i = 0$ 
3  while  $asset = BOM.knownDependencies[i]$ 
4       $BOM' = asset.BOM$ 
5       $i = i + 1$ 
6       $j = 0$ 
7      while  $asset' = BOM'.knownDependencies[j]$ 
8           $BOM'' = asset'.BOM$ 
9           $j = j + 1$ 
10     (Repeat until all known dependencies reached...)

```

Algorithm 6.1: Iterating over a DDS and its assets' known dependencies

service itself. Table 6.3 identifies the entities in the system, and the credentials issued to attest to asset to qualities and the BOM structure for each asset.

Issuing Actor	Role	Credential	Issued to Agent
CM	DP	Dataset Qualities	HealthAdviceEN
CM	DP	Dataset Qualities	HealthAdvicePT
CM	SI	DDS Qualities	AuroraAgent
CM	SI	DDS Bill of Materials	AuroraAgent

Table 6.3: Credentials issued to SSI Agents in Aurora Scenario

The process of gaining oversight on a DDS is documented as pseudocode in Algorithm 6.1. The algorithm is implemented as a Tree Traversal Algorithm, performing a Depth-First Search, with Preorder Traversal.

Peffer, et al. [122] describe demonstration of the use artefacts developed through the DSRM framework in artificial scenarios as a way to determine the extent to which an artefact can provide a solution to the identified problem. Through the scenarios

described, we have considered two different types of DDS. In the first instance, based on S1, every element is verifiable and a known party is able to be held accountable for each asset. In the second, based on S2, an asset is provided on an AIaaS basis by a third party who has no significant relationship with the DDS ecosystem, and is unlikely to participate in credential exchange. The proposed method has been shown to be adaptable in such circumstances, and able to provide a verifiable entity and accountable party where such exists, whilst still functioning where no such provision is possible.

6.3.2 Design Cycle 2: Providing a Human-Machine Interface

Revision of Objectives

We have designed a method to provide oversight on a DDS, using the BOM schema designed in Chapter 4 and the SSI-based architecture from Chapter 5 and demonstrated its application in artificial scenarios. In a practical situation, a DA or other stakeholder will require tools to support requesting the BOM VC, processing it and iterating through the supply chain to scrutinise all contributions. To meet this need, we are motivated to investigate the feasibility of providing a web-based tool to support a DA in gaining verifiable oversight of a DDS. The tool would conduct the SSI interactions on the DA's behalf, and present information about the DDS in an accessible manner.

Design and Development

The system described is titled the *AI Scrutineer* (AIS), and provides a web-based user interface to demonstrate how verifiable oversight and scrutability on DDS could be provided to DAs and other stakeholders.

Entities in the AIS have the following roles:

- SI:** The SI is the party with accountability for the system as a whole. They take ownership of documenting the overall structure in the BOM to provide oversight.

The contents of the BOM is defined by the JSON Schema structure designed in Chapter 4 which provides a template that the SI can populate with information about the DDS, including models and datasets used. The JSON structure for the BOM is encoded as a VC, signed by the SI and issued to the software agent representing the DDS.

DDS Agent: The DDS is represented by a software agent, which holds the DDS BOM VC issued by the SI.

Model Engineer: The ME is responsible for the model, and populating the model's BOM schema, and signing and issuing it as a VC to the model agent. The ME can interact with the ecosystem through a digital wallet, or be represented by an autonomous software agent as part of the ML production pipeline.

Model Agent: The Model Agent (MA) is an SSI software agent which represents a model and holds the BOM VC issued by the ME, interacting with other parties requesting information about the model. The MA should remain available and accessible at a known endpoint, in order that it can satisfy credential requests from interested parties.

Data Provider: The DP is responsible for a dataset, and will populate the BOM record for the dataset and sign and issue it as a VC to the data agent.

Dataset Agent: The Dataset Agent (DSA) is an SSI software agent which represents the dataset, and holds the dataset's BOM VC issued by the DA. As with the MA, the DSA should remain accessible at a known endpoint location, to satisfy stakeholder requests.

In Design Cycle 1, the DA – as the party interested in the system – was described as interacting directly with the DDS agent, via the SSI infrastructure. AIS introduces a new layer into the interaction – the DA will interact with a human computer interface (HCI) accessed through a web browser. The HCI will interact with the SSI subsystem

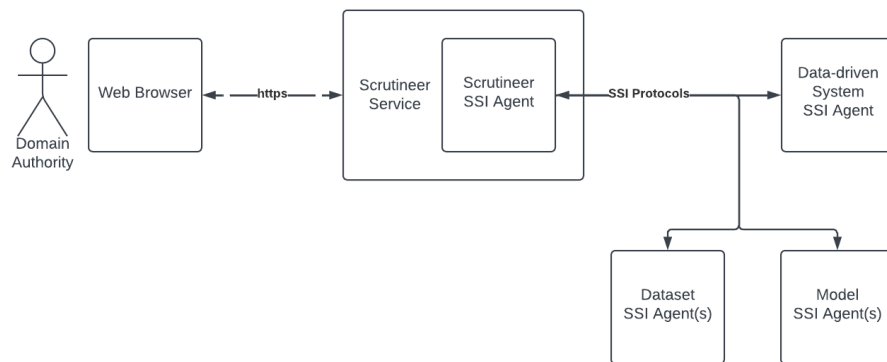


Figure 6.2: AI Scrutineer Architecture

on behalf of the DA. This interaction is delivered through an additional component, the Scrutineer Service.

Scrutineer Service: As the AIS design is intended to demonstrate how non-technical DAs can gain oversight on a DDS, the Scrutineer Service (SS) has been introduced to mediate between a web-based HCI and the SSI infrastructure. The SS represents the DA in SSI interactions.

When a DA wants to inspect a DDS, they visit a supplied web address for the AIS (perhaps by scanning a QR code with the address embedded, provided by the SI). On receiving the user's request via an HCI, an SSI agent component within the SS makes an SSI connection to the DDS Agent and requests a presentation of the BOM credential. On delivery of the presentation, the SS decodes the JSON payload data from the BOM. The SS iterates through the *knownDependencies* field of the BOM and shows information for all known models and datasets used in the DDS. This information can be presented to the DA in a human-readable form via a web interface. The architecture for such a system is shown in Figure 6.2.

Demonstration and Evaluation

The AIS demonstration integrates with existing ML workflows, and takes advantage of documentation that might already exist within a DDS production pipeline. The BOM document for the model in the demonstration uses metadata from Model Cards for Model Reporting [107] documentation by integrating with The Model Card Toolkit (MCT) [55], a software tool that supports Model Card authoring. MCT is an open source application, published by Google researchers, which integrates with the production workflow of TensorFlow¹ model development and deployment. The AIS demonstration has been designed such that metadata from MCT is used to populate the BOM for the model.

Implementation of the demonstration used the Hyperledger Aries platform, abstracted through the Syndicate.id layer, as described in Section 5.3.3. In the demonstration, a Jupyter Notebook interfaces with an SSI agent that represents the Model Engineer. The Notebook has integration with MCT and pulls model metadata from the TensorFlow production pipeline to populate the model's BOM VC. In particular, the MCT provides the name, overview and version number of the model. Other fields of the BOM are populated manually in the Jupyter Notebook. These provide information about datasets used to train the model and the team responsible for development of the model. A request to publish creates a connection between the ME's agent and the MA. The data for the BOM is encoded as the payload of a VC, which is cryptographically signed by the ME agent, and issued to the MA. The SI also populates a BOM, listing the models and any known datasets used in the DDS. This BOM is encoded as a VC by the SI, and issued to the DDS agent.

AIS demonstrates how a DA could be provided with oversight on a DDS through a web-based interface, presenting information that provides oversight, and identifies accountable parties. Figure D.1 in Appendix D shows a screenshot of the AIS web interface, as it might be presented to a DA. The interface employs a graph-based view

¹<https://www.tensorflow.org/>

of the DDS components, with the model and datasets enumerated. Further details on the model (automatically taken from the TensorFlow model, and encoded in the VC) are shown below the graph, along with brief biographical information about the team and the dataset used for the developing the model. The green check mark illustrates that the information displayed has been retrieved from a signed VC, and cryptographically proven to be as provided by the accountable party, and unchanged from the time of signature. The demonstration shows one way in which an SSI-based system for providing oversight and accountability could be used by a DA, without requiring the DA to directly interact with SSI software and processes.

The architecture that supports the AIS demonstration builds upon the foundations that can be used to attest to shared data qualities first proposed in Chapter 5, and extended in Design Cycle 1 of this chapter to integrate with a BOM record of the supply chain of assets and artefacts used in development of a DDS. The architecture has been demonstrated with an implementation of a system showing that the BOM can be requested and inspected by DAs who need to have oversight and assurance on the integrity of the constituents of the system. The demonstration uses a web-based interface layer through which DAs can request and inspect the DDS, and evaluate the suitability of its underlying models and data for their use case and domain. The web interface is dynamically populated from information retrieved from the BOM, held by a software agent representing the DDS and its components, and the information displayed is backed by a visual mark of assurance that it has been cryptographically verified. This verification is an important aspect of the approach, as it shows that the documentation and metadata has not been altered or revoked. As such, practitioners can gain confidence in the documentation provided with their DDS, through the accountability that VCs provide.

The implementation of the architecture in the AIS demonstration provides an instance of a “single-case mechanism experiment”, which is often performed in a laboratory environment to test artefact prototypes. This approach contributes to a technical risk and

efficacy evaluation [165], showing that the technical approach is effective. A demonstration of this nature shows not only that the design approach can be implemented, but also how it can be implemented [40]. Design of the method and architecture and the subsequent demonstration through an implementation of the AIS system has shown that it is technically feasible to use SSI protocols and VCs to attest to qualities of DDS and their components. A third-party ML model production tool has been integrated with the processes that populate the BOM, and cryptographically sign and issue it to a software agent representing a model in a DDS. Implementation of the design has demonstrated that VCs containing information about the DDS, models and datasets can be accessed and verified, and presented to a DA or other stakeholders.

6.4 Research Outputs

6.4.1 A Software Architecture Providing Oversight and Accountability on DDS through a Verifiable BOM

The research presented in this chapter has led to the design of a software architecture that supports provision of verifiable oversight on a DDS, and a means for asset providers to take accountability for their contributions. The architecture extends the design of Section 5.4, and uses a BOM record of the supply chain of contributions to a DDS to provide a means to deliver verifiable metadata and documentation about a DDS, and to demonstrate accountability of contributing parties. The BOM record is based on the JSON Schema structure presented in Section 4.4, and encapsulated as a VC, in order that it can be associated with an accountable party by using a digital signature through SSI protocols.

Figure 6.3 shows a context view for a system architecture designed to provide oversight and accountability on a DDS. The context view shows the role that the System Integrator has in providing the DA and stakeholders with verifiable claims provided by

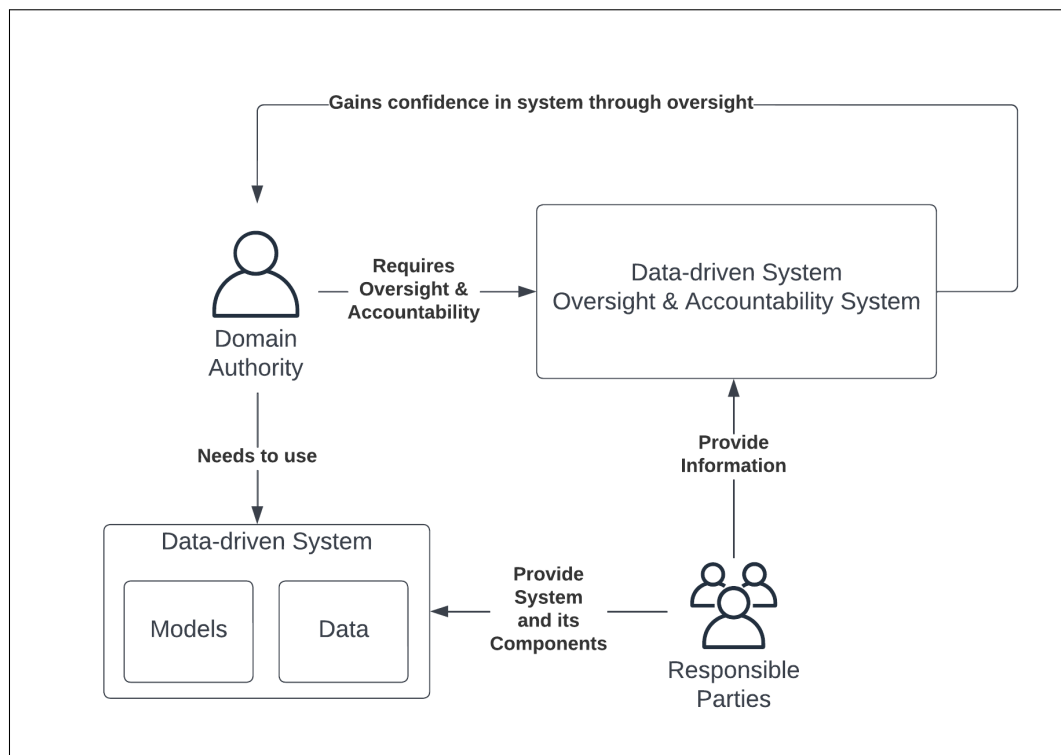


Figure 6.3: Context View of DDS Oversight and Accountability System

parties responsible for the system and its components. These claims should serve to increase the confidence that the DA has in the DDS, and the trust that they are able to place in the system.

Figure 6.4 shows a functional view for the architecture of a system to provide verifiable oversight and accountability on a DDS. A functional view “defines the architectural elements that deliver the functions of the system being described” [133]. Figure 6.4 uses UML symbols to represent the main components of the system, and the interfaces between them. The functional diagram shows a DDS BOM SSI Agent, which holds a credential storing the BOM of the overall system, built from knowledge of the individual contributions towards the system collected and curated by a DDS BOM Generator component. External DDS Contribution SSI Agents hold credentials relating to individual contributions to the DDS, from models and datasets. The BOM and the contribution credentials can be requested by an external Credential Inspector, through interfaces that follow SSI protocols. An Identity Registry can provide inform-

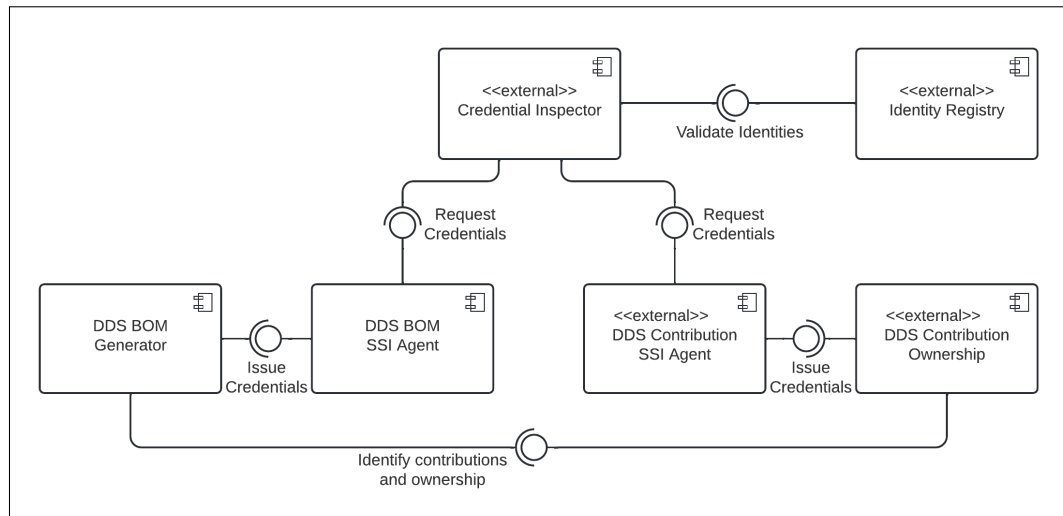


Figure 6.4: Functional View of DDS Oversight and Accountability System

ation about parties signing credentials, so that stakeholders relying on the the Credential Inspector can determine whether signing parties are known organisations, trusted peers, or members of a community of practice, for example, and use this knowledge to determine how much trust to place in the information provided.

The contribution presented here extends published approaches to sharing information about data-driven systems, notably Model Cards for Model Reporting [107]. The contribution provides a mechanism in which information about a system and the constituent parts of the system can be digitally signed by parties with responsibility for development or selection of those assets. This provides stakeholders with traceability on DDS, and the ability determine who is accountable for systems and their components. Published approaches do not provide verifiability and evidence of accountability on claims made for system qualities.

6.4.2 Communication

The problem and outcomes of the design cycles of this chapter have been regularly presented to academic colleagues and industry peers in research group meetings at Cardiff University, the University of Notre Dame, USA, and in the DAIS-ITA project.

Resultant discussion has served to inform direction of the research through the design cycles.

The AI Scrutineer demonstration of the method has been documented in an engineering white paper format, and was submitted as an entry to an SSI Use Case competition run by the Sovrin Foundation². The research problem, proposed solution and the AIS demonstration has been presented as part of a research seminar at the UK's Office of National Statistics. A presentation was also given to the AI and Metaverse Technology Taskforce of the Trust over IP Foundation³, an expert group which includes leading researchers and practitioners from the SSI field. One member of the group described the talk as a "very timely presentation on 'AI Data Stacks'". A rich and detailed discussion among the group ensued, which has informed the Future Work section of this thesis, in Section 7.3.

The research developed through this chapter has been formally described and published as a peer-reviewed journal article [12].

6.5 Evaluation

Evaluation of a software architecture is concerned with resolution of the question "will the computer system to be built from this architecture satisfy its business goals?" [79]. Following the approach of the DSRM, evaluation is considered from the viewpoints of rigour and relevance. In the Rigour Cycle, we adopt a formal approach to software architecture evaluation, aligning business goals to quality attribute requirements. In the Relevance Cycle, we consider the ability of the architecture to address the requirements introduced as the problem characteristics, presented in Section 6.2.

²<https://twitter.com/SovrinID/status/1366573747419770880?s=20>

³<https://trustoverip.org>

6.5.1 The Rigour Cycle: Architecture Decision Review

Erder and Pureur’s Continuous Architecture [53] is an architectural approach developed to support rapid delivery cycles, meeting demands driven by advances in software engineering. One of the principles of the Continuous Architecture approach is to focus on quality attributes, rather than on functional requirements. Quality attributes are classified into ‘Quality Attribute Requirements’ (defined as “qualifications of the functional requirements or of the overall product”) and ‘Constraints’, which are “design decisions with zero degrees of freedom”. Erder and Pureur’s view is that an architect should make design decisions in order to meet quality attribute requirements. In their work on software architecture evaluation, Eloranta, et al. [51], state that “the goal of architecture evaluation is to find out if made architecture decisions support the quality requirements set by the customer”. Erder and Pureur [53] describe a decision-centric evaluation approach as a “lightweight yet very effective architecture validation method”, and propose adoption of the Decision-Centric Architecture Review (DCAR) method [163], which is based on an evaluation of the decisions – “the key unit of work of architecture” [54] – that lie behind a software architecture design. This evaluation adopts DCAR to review the design decisions made, in the light of their impact on the quality attribute requirements of the proposed solution.

An overview of the DCAR method is provided in Appendix E. Eloranta, et al. [51], describe making adaptations of the DCAR approach to work in different settings, based on “observed real-life software architecture practices”. Being mindful of the availability of our colleagues and peers, and in alignment with emerging work practices following the COVID-19 pandemic [171], our evaluation adapted DCAR so that it was able to be conducted asynchronously. The sequence of the DCAR method was followed to produce a document that described the context and architectural decisions, and the reasoning behind those decisions. The document (provided in Appendix G) was shared with colleagues from different organisations, and with different levels of experience in software architecture design. Colleagues were invited to study the document in their

own time and provide their feedback via an online form.

DCAR evaluation begins with a Management Presentation, to identify business goals. To provide a business focus, we identified *cost effectiveness* as a quality attribute. Erder and Pureur [53] state that “Cost effectiveness is not commonly included in the list of quality attribute requirements for a system, yet it is almost always a factor.”. Indeed, cost effectiveness will likely be a significant factor in the adoption of a system to provide oversight and accountability on DDS, that is intended to be deployed across a number of stakeholders, often with very limited connections to each other.

Accordingly, the business goals are defined as:

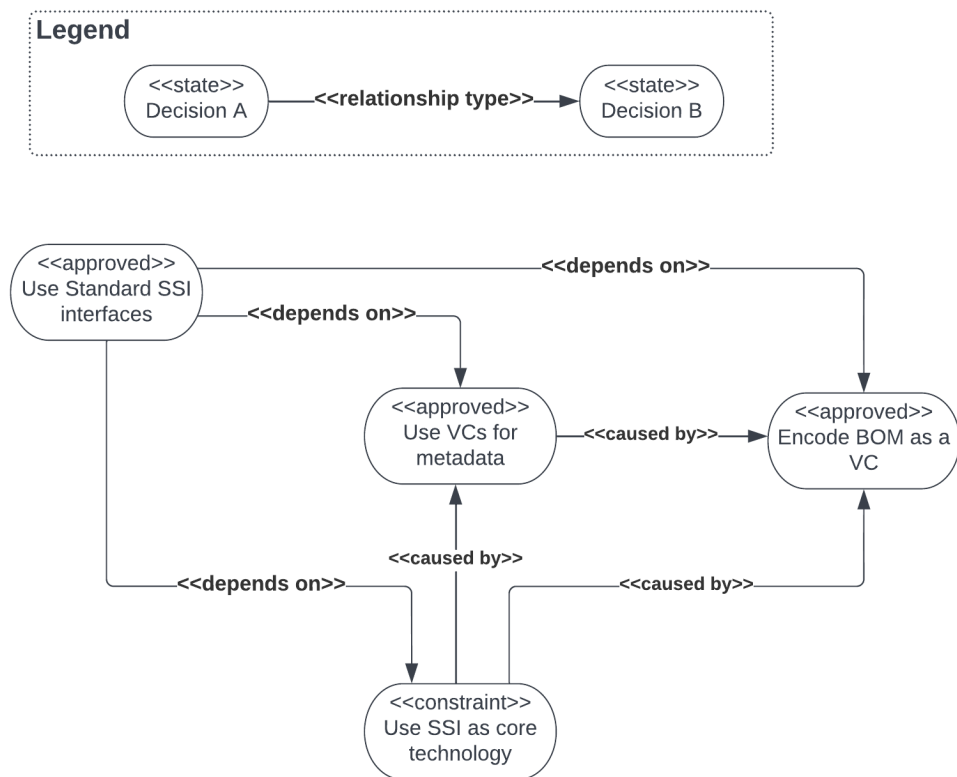
1. The solution adopts SSI Principles, as a technical constraint.
2. The solution is cost effective through its lifecycle.
3. The solution can be widely adopted.

These business goals are used to evaluate the quality attributes of the architecture design.

Aligning with Step 2 of the DCAR process – the Architecture Presentation – three existence decisions [86] made during the architecture design cycles are described. The DCAR method proposes a format for documenting decisions, which is shown in Table 6.4. Appendix F shows the template populated for decisions made in the design: using standard SSI interfaces (Table F.1), using VCs to hold metadata (Table F.2) and encoding the BOM as a VC (Table F.3).

Step 3 of the DCAR approach creates a decision relationship diagram, showing how architectural decisions influence each other, and where decision forces have had an impact. This thesis is focused on the use of SSI as a possible solution approach, which is considered to be a technical constraint. This constraint is a property decision [86], and has consequences for other architectural decisions in the design. As such, the adoption of SSI is considered to be a decision force [51].

Decision Name	
Problem or Issue	What problem or issue is solved by the solution
Solution	Description of the decision or solution.
Alternatives	Which alternative solutions were considered?
Arguments for	Benefits of the chosen approach
Against	Shortcomings of the chosen approach

Table 6.4: Decision Description Template**Figure 6.5: Decision Relationship viewpoint of Architecture**

The format of the diagram is based on the Decision Relationship viewpoint proposed by Van Heesch, Avgeriou, and Hilliard [162], which is designed to show architecture decisions, their relationships to other decisions, and their current states. The Decision Relationship viewpoint for the proposed architecture is shown in Figure 6.5. Decision

forces are presented as a bullet list, using the vocabulary of the domain – i.e., the domain in which the architecture is set, which in the studied case is taken to be that of DDS and SSI in general, rather than the domain of any particular deployment of the system.

The outcome of this step of the DCAR process is a list of business-related decision forces that should be taken into consideration during evaluation of the proposed architecture. These decision forces were identified as:

- SSI implementations are available as open source
- Components of SSI systems will be interoperable
- The system will be deployed when SSI infrastructure is more mature
- Running an SSI software agent will be no more complex than (e.g.) running a web server, so is not to be considered as a barrier to adoption

Step 4 of the DCAR approach involves narrowing the scope of the evaluation to focus on key decisions, which are voted upon during the review process. As this review considered the early stages of an architectural design, reducing the quantity of decisions was not a concern. As such, documentation produced in Step 3 was taken forward to the final evaluation, omitting the need for Step 5 – a similar adaptation was applied by Eloranta, et al., [51] when applying DCAR to projects in the software industry.

Step 6 is the evaluation itself, which requires presentation of the material produced and discourse with reviewers to determine whether decisions made are considered to be sound, or need to be re-visited. In this asynchronous adaptation of DCAR, this step was implemented by sharing the prepared material (Appendix G) with reviewers – colleagues with software architecture design experience or expert knowledge of SSI technology. The reviewers were invited to read the material, and to provide any thoughts and feedback electronically. In particular, reviewers were asked to consider whether they felt the designed architecture was capable of satisfying the business goals.

Results

The reviewers were asked to consider the design decisions, and decide whether they considered them to be sound. Three reviewers completed the process and submitted their thoughts on the designs. The reviewers remained anonymous, but each provided their level of familiarity with SSI, and the time they had taken to review the presentation and provide their feedback, which is summarised in Table 6.5.

Reviewer	Knowledge of SSI	Time Taken
A	A lot	30 minutes
B	None	30 minutes
C	A little	1 hour

Table 6.5: Self-reported Knowledge of SSI and Time Taken

- *Decision 1: Use Standard SSI Interfaces:* Reviewer A stated that “Using open standards allows innovation at the edges and prevents vendor lock-in by enabling multiple distinct implementations and technology stack to interface over common protocols.”. Reviewer B added that “Non-standard interfaces could cause a barrier to wider adoption”. Reviewer C did not provide a comment here.
- *Decision 2: Use VCs for Metadata:* Reviewer A stated that “I think it is an excellent idea. VCs provide a structured, integrity assured payload that can be used to assert claims about a subject.” Reviewer B, who had no knowledge of SSI, felt that when considering the decisions forces “It is a sound decision. It provides the required functionality with minimal extra complexity.”. Reviewer C, who claimed a little knowledge of SSI, speculated whether storing metadata on a blockchain might be useful to provide evidence of tampering. Reviewer A also outlined a challenge to consider – “who controls/holds the VC. The subject is the Model of Dataset, but the holder needs to be an ‘intelligent’ entity that is

capable of presentation.” We propose to use software agents as the holder, and to represent assets as the holder.

- *Decision 3: Encode the BOM as a VC*: Reviewer B repeated their response to Decision 2, “It is a sound decision. It provides the required functionality with minimal extra complexity.”. Reviewer A agreed with the decision, although expressed uncertainty about how the BOM contrasts with the Metadata. Pragmatically, the reviewer is correct, all VCs are JSON key/value pairs, and so the BOM is a type of metadata, just as any other credential issued to the asset would be. Reviewer C also agreed with the decision, stating that “Storing the BOM as VC is a good idea when compared to storing on database or file format. BOMs in the form of VCs can be controlled more securely and efficiently by the organization or author using an SSI controller agent.”.

Reviewers were asked to consider whether the design approach would satisfy the business goals. Reviewer B agreed. Reviewer A agreed, with the caveat that they felt that adopting SSI Principles was quite vague as a business goal. This reviewer also proposed that a decision force that focused on the adoption of open standards might be appropriate to drive wide adoption – they pointed out that the Hyperledger platform would not be a sound implementation choice if the adoption of open standards was an objective, as it does not currently follow emerging standards. Reviewer C also agreed, but wondered whether scalability also needed to be considered.

6.5.2 The Relevance Cycle

As previously discussed, the relevance of a design artefact can be determined by considering it in the context of the environment in which it is to operate, and evaluating how effectively it would address the problems identified in its domain. Section 6.2 identified the requirements for a solution to provide oversight and accountability in complex multi-stakeholder DDS, in order that DAs and other stakeholders are able to

form an opinion on the suitability of the system for use in their domain. These are summarised in Table 6.1.

R1 Provide verifiable oversight and assurance on DDS

Adopting SSI protocols in the solution architecture to support claims about contributed assets and their metadata, provides mechanisms for responsible parties to specify, populate and cryptographically sign digital credentials as VCs. The use of DIDs creates a cryptographically verifiable link between claims made in a VC and the signing party. Parties wishing to verify claims made about a DDS and its assets, and who those claims in VCs are signed by, can request a VP from the asset's SSI agent. This VP provides cryptographic proof that the signed claim was issued and has not been subsequently tampered with, and places accountability for claims made onto the signing party.

Design Cycle 1 demonstrated that parties responsible for assets in a DDS could issue signed claims about the asset qualities as VCs, and that a BOM could be used to collate the assets across the DDS, and itself be issued as a signed VC. Assets which did not have a party able to take responsibility for them, such as those provided by organisations outside of the ecosystem, could still be included in the BOM. Third parties are able to request and inspect the BOM as a signed VP, and are able to iterate through the BOM to identify individual assets. Each asset in turn can be inspected, and a request made for its own VP. Stakeholders can verify that claims were signed by a key under the control of the signing party, and had not been tampered with or revoked. Exploration of the artificial scenarios used in the design cycle enabled demonstration and evaluation of the efficacy of the proposed design in providing a BOM of a DDS, and accountability on claims made about the DDS, and the digital assets that contributed to the DDS. Thus we can assert that requirement R1 is met by the architecture.

R2 Provide accountability on DDS contributions

As previously shown, SSI protocols allow responsible parties to specify, popu-

late and sign digital credentials. These credentials can be used to make claims about qualities of a digital asset, such as the metadata of a dataset. DIDs and asymmetric cryptography provides proof that claims were made by the signing party, and have not been tampered with subsequently. This can provide a verifying party with trustable oversight on the stated qualities of an asset, based on the reputation of the signing party.

Design Cycles 1 and 2 showed that contributors to a DDS could issue signed claims about the digital asset qualities of their contributions. By accessing and iterating through a BOM for the overall DDS, itself a signed and verifiable credential, third parties can request and inspect VPs of each contributing asset, and verify that they were signed by an identified contributor and not tampered with. The scenarios used in the demonstration phase of the design cycles enabled testing and evaluation of the efficacy of the proposed design, and demonstrated that requirement R2 is met.

A set of technological rules for the properties of *scrutability*, *verifiable oversight* and *accountability* in a data-driven system were identified in Section 3.4 (Definition 2). Here, *scrutability* is provided for a data-driven system through the provision of access to a bill of materials record for the system. The AI Scrutineer demonstration in Design Cycle 2 provided an illustration of how scrutability could be offered to different stakeholders in a DDS ecosystem, with the adoption of a user-facing web interface. *Verifiable oversight* is provided through the use of a Verifiable Credential data structure, which contains a digital signature within its payload. This signature can be used to verify the integrity of the information within the credential, demonstrating that it has not been modified since it was signed. *Accountability* is also supported by the use of asymmetric cryptography, with a published public key being used to validate the signature provided with each claim, providing proof that the claim was signed by the associated private key. Protocols for decentralised identity adopted through this thesis provide a means to locate the public key that signed a verifiable credential. An external

registry would typically be used to maintain a registry of known DIDs and the associated parties, providing a mapping between a DID and the entity accountable for any claim being made.

6.5.3 Limitations

The evaluation has considered the technical viability of using SSI to provide oversight and accountability on a DDS. The DRSM approach adopted through this thesis provides a method that is in alignment with agile methods used in industry for the development of high quality software solutions. The choice of DCAR as an evaluation approach was motivated by its suitability for application in industry, alongside agile development methods [51].

It was not feasible to conduct a full, face-to-face DCAR evaluation, yet adaptation of the approach such that it could be conducted in an asynchronous manner enabled a wider set of reviewers with different backgrounds to contribute to the evaluation. Preparation of the material for the asynchronous peer-review followed the DCAR steps, with a focus on the decisions made, their relationships and the decision forces provided good insight into the evaluation methodology. This showed how the evaluation approach might be adapted to be suitable for use in a modern business setting, where key stakeholders have limited availability and different working patterns, often across timezones. The framework for the evaluation created a set of material which was able to be shared with experienced software architects and SSI experts, and supported the beginning of a constructive discourse on the merits of the proposed architecture and its ability to meet the defined business goals. The adaptation of the method provided access to a group of reviewers who would have been difficult to reach with a synchronous, face-to-face approach. As such, the asynchronous approach taken here to the DCAR evaluation may be applicable to a practical situation in industry, and we recommend that researchers further consider how their methods for software architecture evaluation can be adapted and applied in software businesses. Reviewers A and B

provided feedback on the evaluation approach itself. Reviewer A stating that “I think it worked well, because I already have a decent understanding of the architecture you are working with. Although, I can imagine those with less firm a grasp might struggle to get down to the details.” They added, “I think the approach works well in conveying how you made design decisions about the architecture in order to meet a set of goals/requirements.” Reviewer B felt that “it’s an interesting approach, and I like it. I do think it’s missing a feedback step - prior to asking if the decisions are sound, it might be pertinent to ask if the decisions appear well informed. A decision could be sound given the assumptions, but if those assumptions aren’t sound that’s not captured.”

Whilst the evaluation considered the architectural approach as sound, practically there are challenges to overcome in terms of operationalising the deployment of SSI-based systems, and in provisioning and deploying a set of software agents to represent each entity of the system. Reviewer C raised the question about scalability, for example. Further work is required to understand how a system can be deployed and how it can operate at scale, and development of a use case scenario in a practical environment will provide insight. The architecture, and the demonstration of the AIS system in particular, is developed on SSI software which follows standardised data models and protocols, and so the design principles and interactions can be ported to other SSI platforms as they become production ready. This should provide a robust foundation for further work, and is in line with the quality attributes and business goals identified during the DCAR evaluation.

There are other directions in which to extend the work. The demonstration AIS system employed an instance of the Scrutineer Service operating as a singular entity, yet the decentralised nature of SSI means that many different parties can operate agents in this role, and present the model’s BOM information in different ways, appropriate to their audience - the only information that needs to be known to the Scrutineer Service is the endpoint of the agent representing the DDS, and the credential which needs to be requested. That is not to say, however, that all data would be freely made available

to all Scrutineer Services. The agent receiving a credential request (in this case, the model agent) can determine whether it wishes to respond to the request, and further can decide what to return to the request, through a model of selective disclosure which can be used to protect confidential information belonging to the actors in the system. As such, different levels of transparency – or opacity – could be provided for different scrutineers, providing opportunities for customising information provision to partners and customers or the wider public, for example.

The AI Scrutineer demonstration was developed with a focus on providing insight into the technical feasibility of the proposed approach, rather than to understand the user requirements of a tool that could be deployed into an end user environment. Further research and usability studies should be undertaken to identify requirements for a practical tool with an HCI that would meet the needs of a range of stakeholders in a real-world deployment.

6.6 Summary

The architecture and demonstration of the AI Scrutineer system described in this chapter has shown that SSI data models and protocols can be used to assert and take accountability for properties and qualities of DDS, datasets and ML models. The solution has demonstrated that a data publisher can issue credentials that provide information on the metadata or other qualities of their datasets, which can be securely held by software agents, and provided on-demand to show the claims made by the publisher or owners of assets. If circumstances change, the claims can be revoked, and parties inspecting assets will be able to determine that the asset is no longer considered suitable. The ability to provide verifiable oversight on a supply chain BOM for a DDS can be used as part of a facility to provide assurance of the ongoing integrity of the DDS to practitioners and other stakeholders, who can access it directly or through a web-based user interface – as demonstrated in the AI scrutineer implementation. Such a system can

protect practitioners from using DDS where the training or test datasets have been discredited, as real-time integrity checks on the supply chain of the asset can be performed and presented.

The asynchronous adaptation of the DCAR approach used for review proved an effective mechanism of performing an architecture review with expert colleagues, without requiring scheduling or a significant time cost across timezones, with each reviewer stating that they spent 30 minutes to an hour on the evaluation, and found that the process went well. One reviewer commented that they liked the “clear justifications of decisions with arguments both for and against” that the DCAR framework facilitated, and felt that the approach worked well in conveying how design decisions were made in the architecture towards meeting goals and requirements.

While work remains to be done in regards to researching effective user interface design for presenting information to end users, as well as further integrating the system into data and ML production pipelines and operationalising for deployment, the technical approach has been shown to be effective in meeting its design goals. The architecture and implementation has shown that SSI protocols and data models can be used to add oversight and assurance to DDS, and provide mechanisms will lead to better visibility into DDS. This can help to build and maintain confidence between the different actors in the system and in the system itself.

Conclusion

7.1 Introduction

This chapter provides an overview of the results of the design research presented in this thesis, placing them in the context of the evolving socio-technological environment in which data-driven systems are deployed and operate.

In Chapter 2 we identified a need to design Information Systems artefacts that could contribute to the development of a viable approach to providing verifiable oversight, transparency and accountability across the ecosystems of contributors and users of a DDS. Section 7.2 coalesces results from our research towards this goal, described in detail in previous chapters, aligning it to the research questions posed. The primary contributions to the knowledge base resulting from our research are enumerated in Section 7.2.1, with the limitations of our approach in developing these contributions explained in Section 7.2.2. The climate in which DDS are developed and operate continues to change, as does the field of decentralised systems, including self-sovereign identity, which underpins our technical approach. We have identified areas in which future research can take advantage of these developments, and provide additional contributions to the knowledge base. This thesis ends in Section 7.3 with suggestions for future work.

7.2 Research Outcomes

The decision to adopt a DDS and apply it to a domain, is ultimately a decision to place trust in the system within the context that it will be applied. Where systems are provided by vendors, systems integrators or other third parties, the DA is unable to know with certainty how the system was developed and tested (and to what degree), including which actors were involved, and what data was used and for what purpose. As such the choice to adopt and use a DDS is risky. As risk increases, the act of placing trust becomes harder to justify [111] – as such, access to trustworthy information becomes increasingly important.

The research presented in this thesis sought to gain insight into this problem, and to develop Information Systems artefacts to contribute towards a solution. Chiefly, we identified a need for information visibility, transparency and accountability on DDS, coming from policymakers and academics, and recognised in analysis of semi-structured interviews with peers who formed the Expert Review Panel. There are many proposals in the literature for documentation formats for DDS and their constituents, but we found that existing work lacked mechanisms to provide verifiability and demonstrate accountability, and the proposed formats are not designed to be easily machine-read. We also identified a lack of viable technical solution designs for providing transparency, traceability and accountability across multi-stakeholder DDS. We adopted the DSRM framework to help us design IS artefacts that can assist in understanding and communicating the structure of a DDS, and to provide machine-readable mechanisms for delivering visible information on assets which comprise a DDS, such that transparency, oversight, and accountability can be offered.

The hypothesis that motivates this thesis is that adoption of a decentralised approach using self-sovereign identity data models and protocols can provide stakeholders of data-driven systems with verifiable oversight onto systems and constituent parts of systems, offering scrutability on contributions to the systems and identifying parties who are accountable for contributions, whilst protecting commercial or private inform-

ation from unauthorised disclosure. Research to test this hypothesis is guided by four research questions, which have formed the main body of the work presented in this thesis. We first considered how to identify the different roles involved in developing and using a DDS, and to understand their different responsibilities and requirements. This led to the development of the Roles and Boundaries Framework, described in Chapter 3, which was well received by many members of the ERP, who could see how they could apply and use it in their own environments. Interviews with the ERP members provided insight into who might be held accountable for contributions to DDS, and whether it was appropriate to place such responsibility onto a named individual, or if it should be a role-based responsibility. This was followed, in Chapter 4, by the development of a proposal for a verifiable bill of materials record of contributions to the supply chain of a DDS. The proposal was again largely well received by the ERP in their evaluation, and correlated with their own mental models on how system constitutions could be recorded. A data model and accompanying schema was developed in support of the BOM model, such that machine-readable records could be developed and maintained. Discussions with the ERP gave insight into how DDS and their contributions might be verified, and whether this was a role that a regulator or other party might perform, and whether legislation might ultimately require this.

Motivated by the hypothesis, the use of SSI patterns was adopted as a technical constraint in the design work of this thesis. We first considered, in Chapter 5, how SSI could be applied to a singular digital asset, and how the approach could be used to provide verifiable assertions about qualities of assets, backed by a digital signature from the publisher of the asset. We established this through a scenario based upon a researcher publishing a scientific dataset. We also considered how SSI could be used to mediate access to assets, and through a scenario based on data-sharing challenges in the multi-media astronomy community, we developed an architecture that used verifiable credentials to store access policies for digital assets, which were used by a policy decision point to control access to the asset. The approach used a sidecar design pattern to only allow researchers with appropriate credentials to access the requested digital

assets. The design was evaluated against software quality criteria, of interoperability and portability, and information security. Chapter 6 built upon the work developed in earlier chapters, bringing designs together to demonstrate how an SSI approach could be used to encode a BOM for a DDS, such that it could provide verifiable oversight on the contributions to the DDS. This was manifested in a demonstration of the AI Scrutineer system, which used an SSI agent as means to collect information about a DDS and present it to a domain authority in a web interface. The architecture of the system was peer-reviewed, using an asynchronous approach based on DCAR, which analyses the decisions made in the development of a software architecture. Here, the software quality of cost-effectiveness was adopted as an evaluand, in order that the solution might be widely adopted.

The results of the design work presented in this thesis, and subsequent evaluation of the designed artefacts, confirm our hypothesis that adoption of self-sovereign identity data models and protocols could provide stakeholders of data-driven systems with verifiable oversight onto systems and the constituent parts of systems, offering scrutability on contributions to the systems and identifying parties who are accountable for contributions, whilst protecting commercial or private information from unauthorised disclosure.

7.2.1 Contributions to the Knowledge Base

The DSRM approach is strongly motivated by the desire to communicate knowledge gained through the iterations of the design cycles, and resulting from summative evaluation of designs. Knowledge should be shared both through formal academic publication, and with professionals from industry. Communication of knowledge developed during the research of this thesis has been a strong feature of the work, resulting in publication and presentation of the formative stages of the research at workshops, and formal publication of peer-reviewed journal papers. Presentation to academic and industry colleagues has also been a core component: notably to the DAIS-ITA project

team, the Blockchain Research Group at University of Notre Dame, the UK National Statistics Office and the Trust over IP foundation.

Formally, we identify the core research contributions of this thesis as:

- The Roles and Boundaries Framework, which can be used to guide identification of the roles and assets in a DDS. This has been developed as a conceptual framework, and UML model.
- A conceptual model of a verifiable BOM for the supply chain of a DDS, with a supporting data model and schema design.
- A software architecture design, that provides verifiable assurance of claims made about digital assets, identifies accountable parties, and provides mechanisms to mediate access to such assets.
- A software architecture design, that provides verifiable oversight across a DDS, enabling stakeholders to perform scrutiny on the system and its constituent assets, and identify parties that take accountability for their contributions.

7.2.2 Limitations

The research presented in this thesis has been technical in nature, and development and evaluation has been framed in the context of artificial scenarios. Conversely, DDS are complex, socio-technological systems, with the potential for making a significant impact on people, both as users of the systems in the DA and as parties affected by the outcomes of the system. As such, our research has only been able to consider the viability of our technical designs in a laboratory context, and has focused strongly on how such solutions might work. We have been unable to consider how systems would be operationalised, and how they might be deployed to different stakeholders and how they would operate in practical environments. Our final technical architecture design has been evaluated using the DCAR approach, but it was not viable to perform

a full evaluation session as suggested by the method, as a peer-group with sufficient knowledge of the project was not available. The approach was adapted to be able to be performed asynchronously, which appeared to work well, but did not lead to discussion and consensus forming among a peer group as would happen in a face-to-face group setting. Conversely, feedback was obtained from colleagues from different organisations, who would not have met in a face-to-face setting. The conceptual frameworks were evaluated with a group of peers in the ERP, which led to very valuable insight. However, interviews were conducted by an inexperienced researcher, and better results might have been achieved if colleagues with more experience in qualitative research methods had been available to assist.

7.3 Future Work

The research described in this thesis has been developed alongside work by others in the wider academic and industry community that have an impact upon it, and its opportunity make a contribution. We have identified possible future work through this thesis, in the context of extending the design work of each chapter. We also see significant opportunities available to develop this work in line with research and industry developments in the field in which our work sits. Here, we describe some of these opportunities.

7.3.1 Self-sovereign Identity

When research toward this thesis began, SSI was in a formative stage. There were few implementations available for use, and what was available was quite fragmented. Now, SSI is well accepted as a paradigm, if not yet widely adopted. Several governments and industry coalitions, most notably in the EU are running research projects and pilot programs using SSI as the foundation for digital identity schemes, for driving

licenses, and proof of educational qualifications, for example. The DID scheme has been accepted as a W3C standard, and the VC scheme is progressing through a similar route. There are many companies offering commercial-grade SSI solutions, and industry bodies are increasingly working on pragmatic approaches to deploy the technology into and alongside established identity and security produces and frameworks. We have argued in the summary of our design chapters that the approach will be viable if SSI is adopted widely, such that users are familiar with the paradigm and have access to the technology. It is possible that this may come to pass. This presents an opportunity to continue to contribute research to the SSI community, in particular in fields such as improving trust in multi-actor systems, and in adopting SSI for machine-to-machine operations. Discussions which followed a presentation of the research of this thesis to The Trust Over IP Foundation's AI and Metaverse Technology Taskforce identified interest in developing schemas and attribute sets that would be appropriate for use in VCs or other mechanisms for providing verifiable declarations on data or ML qualities, as well as how to support trustworthy decentralised mechanisms for identifying parties taking responsibility for issuing and signing claims. Continued development of paradigms, and availability of underlying SSI technology also serves to provide better foundational components on which to develop architectures and to deliver demonstration systems. Ideally, these systems will be able to be deployed into case studies beyond the laboratory constraints of this thesis, and work will be undertaken to understand how to operationalise SSI-based approaches to information visibility, transparency and accountability.

7.3.2 New Modes of Access Control

The use of VCs for access control policies offers promise in providing a lightweight, decentralised, granular access control mechanism. The advantage of the approach is that assets can hold their own access policies, and parties wishing to access the resources can be provisioned through a range of attributes in credentials they hold. This

provides significantly more flexibility than role-based access control, for example. Further work can be conducted to investigate and demonstrate the viability of designing more advanced access policies and adoption of SSI and credential features such as expiration and revocation, and constraints based on the Issuer of the credentials being presented. Integration with policy-based access tools used in the wider community, such as the Open Policy Agent and its Rego policy language will extend the utility of the approach, as will investigation of integration with authentication approaches based on oAuth2. Integration with a blockchain platform could further improve security and accountability, by providing an immutable audit record of all data access attempts. The architecture designed in Chapter 5 adopted a sidecar design pattern as part of a web-based data sharing infrastructure. It is envisaged that the approach could be modified such that access policy implementation could be conducted directly by the dataset's own agent, providing a more decentralised architecture, and further work in this area is recommended.

7.3.3 Human-Centred Research

There are a number of areas in which research could be conducted into human-centric aspects of design. The ERP members were generally enthusiastic about the Roles and Boundaries Framework when it was presented to them. It would be very valuable to take the framework out into the field, and to apply it to deployed systems or systems being considered for deployment. Researchers could test the validity of the labels that the framework currently employs, and determine whether better alternatives can be found. Similarly, they could analyse whether there are effective ways to demonstrate and extract the trust relationships between parties. We see there is a huge scope for social experiments in this area, using the RB Framework as a starting point.

In considering the AI Scrutineer, a very primitive UI was developed in our demonstration. A useful area of future research would be to consider how information about DDS could be presented to stakeholders to best effect. This would involve close collab-

oration with human-computer interface experts, in order to understand how complex, live, hierarchical information about systems can be expressed most clearly and effectively. Some insight may be gained from the open source software domain, SBOMs are used to alert users of vulnerabilities in underlying software components – although this audience is typically other software developers, and not Domain Authorities or more generally skilled stakeholders. There is a rich area of future research into how to present complex, and important, verifiable information to an audience of DDS stakeholders.

7.3.4 Verifier Roles in Decentralised Architectures

The architecture of the AIS demonstration provided an instance of the Scrutineer agent operating as a singular entity, yet the decentralised nature of SSI means that many different parties can operate agents in the verifier role and present the BOM information in different ways, appropriate to their audience - the only information that needs to be known to this agent is the endpoint of the agent representing the DDS, and the credentials which need to be requested. Discussions with the ERP provided insight into different roles that might be entrusted as verifiers of information about DDS and their constituents, from regulators through to trusted commercial or community partners. Future research could consider this from a social perspective, as well as a technical perspective. Consideration could be given to how a regulator might operate in such a role, and how they might present information to stakeholders. Similarly, business-focused research might be able to determine whether there is a case for a commercial or not-for-profit organisation adopting the role of trusted verifier for a DDS, and interpreting and presenting information to their customers or communities in such a way that stakeholders can understand it and trust it.

7.3.5 Emerging Data-centric Design Patterns

Within the data community, there are emergent notions of “data as a product” and data mesh architectures [95] are being developing in support. The concepts that these developments bring, in terms of well-defined data products, with clear boundaries and ownership of the product, fit well with the approach developed through this thesis. Research could be conducted to explore how these new data paradigms can be mapped onto decentralised SSI design patterns, and whether the roles and responsibilities defined and required in data mesh architectures can be used to provide accountability on assets in DDS. As such, future work could determine how to align or modify the research presented in this thesis so that it can make a contribution to emerging work in the data mesh research community.

7.3.6 Asynchronous Design Review Methods

The evaluation of the software architecture presented in Chapter 6 adopted an approach which considered decisions made in the design, based on the solution requirements. The DCAR framework was said to be more time efficient than scenario-based evaluations, and suitable for integration with the scrum approach adopted in agile software methods. Nonetheless, a DCAR-based design review was still anticipated to take half a day, and involve several parties including business stakeholders and peer software architects. It was not viable to create such a group for the evaluation of this project, and so the DCAR approach was modified such that it could be conducted asynchronously. Material was developed and shared with peers, with participants able to take part and make contributions to the review at a time that suited their work schedules. In our experience, this process worked well, and delivered an effective evaluation without placing undue burden on colleagues or requiring synchronisation of schedules across a global team. We strongly advocate that further research is conducted into approaches for software architecture evaluation that can be conducted in this asynchronous man-

ner, as this aligns with working practices that are increasingly common in the software industry. If such research can lead to the development of viable methods for the evaluation of software architecture designs that work in remote, asynchronous team environments, this will bring significant benefit to those who seek to improve the quality of software architecture designs.

Bibliography

- [1] Zahra Shakeri Hossein Abad, Gregory P Butler, Wendy Thompson, Joon Lee, et al. Crowdsourcing for machine learning in public health surveillance: lessons learned from Amazon Mechanical Turk. *Journal of medical Internet research*, 24(1):e28749, 2022.
- [2] Md Manjurul Ahsan, Shahana Akter Luna, and Zahed Siddique. Machine-learning-based disease diagnosis: A comprehensive review. In *Healthcare*, volume 10, page 541. MDPI, 2022.
- [3] Joan E van Aken. Management research based on the paradigm of the design sciences: the quest for field-tested and grounded technological rules. *Journal of management studies*, 41(2):219–246, 2004.
- [4] Gabriele Spina Alì and Ronald Yu. Artificial intelligence between transparency and secrecy: From the EC whitepaper to the AIA and beyond. *European Journal of Law and Technology*, 12(3), 2021.
- [5] Christopher Allen. The path to self-sovereign identity.
<http://www.lifewithalacrity.com/2016/04/the-path-to-self-sovereign-identity.html>, 2016.
- [6] Matthew Arnold, Rachel KE Bellamy, Michael Hind, Stephanie Houde, Sameep Mehta, Aleksandra Mojsilović, Ravi Nair, K Natesan Ramamurthy, Alexandra Olteanu, David Piorkowski, et al. FactSheets: Increasing trust in AI

- services through supplier's declarations of conformity. *IBM Journal of Research and Development*, 63(4/5):6–1, 2019.
- [7] Mark Assad, David J Carmichael, Judy Kay, and Bob Kummerfeld. PersonisAD: Distributed, active, scrutable model framework for context-aware services. In *International Conference on Pervasive Computing*, pages 55–72. Springer, 2007.
- [8] Senthil Kumar B, Aravindan Chandrabose, and Bharathi Raja Chakravarthi. An overview of fairness in data – illuminating the bias in data pipeline. In *Proceedings of the First Workshop on Language Technology for Equality, Diversity and Inclusion*, pages 34–45, Kyiv, April 2021. Association for Computational Linguistics.
- [9] Guido Barbaglia, Simone Murzilli, and Stefano Cudini. Definition of REST web services with JSON schema. *Software: Practice and Experience*, 47(6):907–920, 2017.
- [10] Iain Barclay and Will Abramson. Identifying roles, requirements and responsibilities in trustworthy AI systems. In *Adjunct Proceedings of the 2021 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2021 ACM International Symposium on Wearable Computers*, UbiComp '21, pages 264–271, New York, NY, USA, 2021. Association for Computing Machinery.
- [11] Iain Barclay, Alun Preece, and Ian Taylor. Defining the collective intelligence supply chain. In *AAAI FSS-18: Artificial Intelligence in Government and Public Sector*, 2018.
- [12] Iain Barclay, Alun Preece, Ian Taylor, Swapna Krishnakumar Radha, and Jarek Nabrzyski. Providing assurance and scrutability on shared data and machine learning models with verifiable credentials. *Concurrency and Computation: Practice and Experience*, page e6997, 2022.

- [13] Iain Barclay, Alun Preece, Ian Taylor, and Dinesh Verma. Towards traceability in data ecosystems using a bill of materials model. In *11th International Workshop on Science Gateways*. IWSG, 2019.
- [14] Iain Barclay, Swapna Radha, Alun Preece, Ian Taylor, and Jarek Nabrzyski. Certifying provenance of scientific datasets with self-sovereign identity and verifiable credentials. In *Proceedings of 12th International Workshop on Science Gateways*. IWSG, 2020.
- [15] Iain Barclay, Chris Simpkin, Graham Bent, Tom La Porta, Declan Millar, Alun Preece, Ian Taylor, and Dinesh Verma. Enabling discoverable trusted services for highly dynamic decentralized workflows. In *2020 IEEE/ACM Workflows in Support of Large-Scale Science (WORKS)*, pages 41–48. IEEE, 2020.
- [16] Iain Barclay, Chris Simpkin, Graham Bent, Tom La Porta, Declan Millar, Alun Preece, Ian Taylor, and Dinesh Verma. Trustable service discovery for highly dynamic decentralized workflows. *Future Generation Computer Systems*, 134:236–246, 2022.
- [17] Iain Barclay, Harrison Taylor, Alun Preece, Ian Taylor, Dinesh Verma, and Geeth de Mel. A framework for fostering transparency in shared artificial intelligence models by increasing visibility of contributions. *Concurrency and Computation: Practice and Experience*, page e6129, 2020.
- [18] Feras A Batarseh, Laura Freeman, and Chih-Hao Huang. A survey on artificial intelligence assurance. *Journal of Big Data*, 8(1):1–30, 2021.
- [19] Emily M Bender and Batya Friedman. Data statements for natural language processing: Toward mitigating system bias and enabling better science. *Transactions of the Association for Computational Linguistics*, 6:587–604, 2018.
- [20] Ruha Benjamin. Race after technology: Abolitionist tools for the new jim code. *Social Forces*, 2019.

- [21] Adrie JM Beulens, Douwe-Frits Broens, Peter Folstar, and Gert Jan Hofstede. Food safety and transparency in food chains and networks relationships and challenges. *Food control*, 16(6):481–486, 2005.
- [22] Umang Bhatt, Alice Xiang, Shubham Sharma, Adrian Weller, Ankur Taly, Yunhan Jia, Joydeep Ghosh, Ruchir Puri, José MF Moura, and Peter Eckersley. Explainable machine learning in deployment. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, pages 648–657, 2020.
- [23] Rejwan Bin Sulaiman, Vitaly Schetin, and Paul Sant. Review of machine learning approach on credit card fraud detection. *Human-Centric Intelligent Systems*, 2(1-2):55–68, 2022.
- [24] Ben Blaiszik, Logan Ward, Marcus Schwarting, Jonathon Gaff, Ryan Chard, Daniel Pike, Kyle Chard, and Ian Foster. A data ecosystem to support machine learning in materials science. *MRS Communications*, 9(4):1125–1133, 2019.
- [25] José A Blakeley, S Muralidhar, and Anil Nori. The ADO.NET entity framework: Making the conceptual level real. In *International Conference on Conceptual Modeling*, pages 552–565. Springer, 2006.
- [26] Miles Brundage, Shahar Avin, Jasmine Wang, Haydn Belfield, Gretchen Krueger, Gillian Hadfield, Heidy Khlaaf, Jingying Yang, Helen Toner, Ruth Fong, et al. Toward trustworthy ai development: mechanisms for supporting verifiable claims. *arXiv preprint arXiv:2004.07213*, 2020.
- [27] Bruce G Buchanan, Daniel Bobrow, Randall Davis, John McDermott, and Edward H Shortliffe. Knowledge-based systems. *Annual Review of Computer Science*, 4(1):395–416, 1990.
- [28] Richard Buchanan. Wicked problems in design thinking. *Design issues*, 8(2):5–21, 1992.

- [29] Peter Buneman, Sanjeev Khanna, and Wang-Chiew Tan. Data provenance: Some basic issues. In *International Conference on Foundations of Software Technology and Theoretical Computer Science*, pages 87–93. Springer, 2000.
- [30] Joy Buolamwini and Timnit Gebru. Gender shades: Intersectional accuracy disparities in commercial gender classification. In *Conference on fairness, accountability and transparency*, pages 77–91. PMLR, 2018.
- [31] Brendan Burns and David Oppenheimer. Design patterns for container-based distributed systems. In *8th {USENIX} Workshop on Hot Topics in Cloud Computing (HotCloud 16)*, 2016.
- [32] Jasmina Byrne, Emma Day, and Linda Raftree. The case for better governance of children’s data: A manifesto. *UNICEF Office of Global Insight and Policy*, New York, May 2021.
- [33] Federico Cabitza, Andrea Campagner, and Clara Balsano. Bridging the “last mile” gap between AI implementation and operation: “data awareness” that matters. *Annals of translational medicine*, 8(7), 2020.
- [34] Carrie J Cai, Samantha Winter, David Steiner, Lauren Wilcox, and Michael Terry. "Hello AI": Uncovering the onboarding needs of medical practitioners for Human-AI collaborative decision-making. *Proceedings of the ACM on Human-computer Interaction*, 3(CSCW):1–24, 2019.
- [35] Seth Carmody, Andrea Coravos, Ginny Fahs, Audra Hatch, Janine Medina, Beau Woods, and Joshua Corman. Building resilient medical technology supply chains with a software bill of materials. *NPJ Digital Medicine*, 4(1):1–6, 2021.
- [36] Philip Chang, Gabrielle Allen, Warren Anderson, Federica B. Bianco, Joshua S. Bloom, Patrick R. Brady, Adam Brazier, S. Bradley Cenko, Sean M. Couch, Tyce DeYoung, Ewa Deelman, Zachariah B Etienne, Ryan J. Foley,

- Derek B Fox, V. Zach Golkhou, Darren R Grant, Chad Hanna, Kelly Holley-Bockelmann, D. Andrew Howell, E. A. Huerta, Margaret W. G. Johnson, Mario Juric, David L. Kaplan, Daniel S. Katz, Azadeh Keivani, Wolfgang Kerzendorf, Claudio Kopper, Michael T. Lam, Luis Lehner, Zsuzsa Marka, Szabolcs Marka, Jarek Nabrzyski, Gautham Narayan, Brian W. O'Shea, Donald Petravick, Rob Quick, Rachel A. Street, Ignacio Taboada, Frank Timmes, Matthew J. Turk, Amanda Weltman, and Zhao Zhang. Cyberinfrastructure requirements to enhance multi-messenger astrophysics. *arXiv preprint arXiv:1903.04590*, 2019.
- [37] Peter Checkland. Autobiographical retrospectives: Learning your way to 'action to improve' – the development of soft systems thinking and soft systems methodology. *International Journal of General Systems*, 40(05):487–512, 2011.
- [38] Jailton Coelho, Marco Tulio Valente, Luciana L Silva, and Emad Shihab. Identifying unmaintained projects in Github. In *Proceedings of the 12th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement*, page 15. ACM, 2018.
- [39] Enrico Coiera. The last mile: Where artificial intelligence meets reality. *J Med Internet Res*, 21(11):e16323, Nov 2019.
- [40] Kieran Conboy, Rob Gleasure, and Eoin Cullina. Agile design science research. In *International Conference on Design Science Research in Information Systems*, pages 168–180. Springer, 2015.
- [41] Pablo Cruz, Luis Salinas, and Hernán Astudillo. Quick evaluation of a software architecture using the Decision-Centric Architecture Review method: An experience report. In *European Conference on Software Architecture*, pages 281–295. Springer, 2020.

- [42] Anupam Datta, Matthew Fredrikson, Gihyuk Ko, Piotr Mardziel, and Shayak Sen. Use privacy in data-driven systems: Theory and experiments with machine learnt programs. In *Proceedings of the 2017 ACM SIGSAC conference on Computer and Communications Security*, pages 1193–1210, 2017.
- [43] Matthew Davie, Dan Gisolfi, Daniel Hardman, John Jordan, Darrell O’Donnell, and Drummond Reed. The Trust over IP stack. *IEEE Communications Standards Magazine*, 3(4):46–51, 2019.
- [44] Andrew Davies, Tim Brady, and Michael Hobday. Organizing for solutions: Systems seller vs. systems integrator. *Industrial marketing management*, 36(2):183–193, 2007.
- [45] Edward Demko. Commercial-off-the shelf (COTS): a challenge to military equipment reliability. In *Proceedings of 1996 Annual Reliability and Maintainability Symposium*, pages 7–12. IEEE, 1996.
- [46] Nicholas Diakopoulos. Accountability in algorithmic decision making. *Communications of the ACM*, 59(2):56–62, 2016.
- [47] Digital Bazaar. Verifiable credentials js library (vc-js).
<https://github.com/digitalbazaar/vc-js/>.
- [48] Donna Dodson, Murugiah Souppaya, Karen Scarfone, et al. Mitigating the risk of software vulnerabilities by adopting a secure software development framework (SSDF). *NIST: Gaithersburg, MD, USA*, 2020.
- [49] Alan A.A Donovan and Brian W. Kernighan. *The Go programming language*. Addison-Wesley, 2020.
- [50] Finale Doshi-Velez, Mason Kortz, Ryan Budish, Chris Bavitz, Sam Gershman, David O’Brien, Stuart Schieber, James Waldo, David Weinberger, and Alexandra Wood. Accountability of ai under the law: The role of explanation. *arXiv preprint arXiv:1711.01134*, 2017.

- [51] Veli-Pekka Eloranta, Uwe van Heesch, Paris Avgeriou, Neil Harrison, and Kai Koskimies. Lightweight evaluation of software architecture decisions. In *Relating System Quality and Software Architecture*, pages 157–179. Elsevier, 2014.
- [52] Emelie Engström, Margaret-Anne Storey, Per Runeson, Martin Höst, and Maria Teresa Baldassarre. How software engineering research aligns with design science: a review. *Empirical Software Engineering*, 25(4):2630–2660, 2020.
- [53] Murat Erder and Pierre Pureur. *Continuous Architecture: Sustainable architecture in an agile and cloud-centric world*. Morgan Kaufmann, 2015.
- [54] Murat Erder, Pierre Pureur, and Eoin Woods. *Continuous Architecture in Practice: Software Architecture in the Age of Agility and DevOps*. Addison-Wesley Professional, 2021.
- [55] Huanming Fang and Hui Miao. Introducing the Model Card Toolkit for easier model transparency reporting. *Google AI Blog*, July, 29, 2020.
- [56] Hyperledger Foundation. Hyperledger Aries.
<https://www.hyperledger.org/projects/aries>, 2019.
- [57] Hyperledger Foundation. Hyperledger Aries Cloud Agent – Python.
<https://github.com/hyperledger/aries-cloudagent-python>, 2019.
- [58] Timnit Gebru, Jamie Morgenstern, Briana Vecchione, Jennifer Wortman Vaughan, Hanna Wallach, Hal Daumé Iii, and Kate Crawford. Datasheets for datasets. *Communications of the ACM*, 64(12):86–92, 2021.
- [59] Paul Grassi, Michael Garcia, and James Fenton. NIST special publication 800-63-3 digital identity guidelines. Technical report, National Institute of Standards and Technology, 2020.

- [60] Paul Groth. Transparency and reliability in the data supply chain. *IEEE Internet Computing*, 17(2):69–71, 2013.
- [61] Object Management Group. The unified modeling language specification, 2001.
- [62] Tianyu Gu, Brendan Dolan-Gavitt, and Siddharth Garg. Badnets: Identifying vulnerabilities in the machine learning model supply chain. *arXiv preprint arXiv:1708.06733*, 2017.
- [63] Evan Hahn. *Express in Action: Writing, building, and testing Node.js applications*. Manning Publications,, 2016.
- [64] Daniel Hardman, Lovesh Harchandani, Asem Othman, and John Callahan. Using biometrics to fight credential fraud. *IEEE Communications Standards Magazine*, 3(4):39–45, 2019.
- [65] Dick Hardt. The OAuth 2.0 authorization framework. 2012.
- [66] Florian Hawlitschek, Benedikt Notheisen, and Timm Teubner. The limits of trust-free systems: A literature review on blockchain technology and trust in the sharing economy. *Electronic commerce research and applications*, 29:50–63, 2018.
- [67] Alan Hevner and Samir Chatterjee. Design science research in information systems. In *Design research in information systems*, pages 9–22. Springer, 2010.
- [68] Alan R Hevner. A three cycle view of design science research. *Scandinavian journal of information systems*, 19(2):4, 2007.
- [69] Alan R Hevner, Salvatore T March, Jinsoo Park, and Sudha Ram. Design science in information systems research. *MIS quarterly*, pages 75–105, 2004.

- [70] Sarah Holland, Ahmed Hosny, Sarah Newman, Joshua Joseph, and Kasia Chmielinski. The dataset nutrition label. *Data Protection and Privacy, Volume 12: Data Protection and Democracy*, 12:1, 2020.
- [71] Gary B Huang, Marwan Mattar, Tamara Berg, and Eric Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. In *Workshop on faces in 'Real-Life' Images: detection, alignment, and recognition*, 2008.
- [72] Ben Hutchinson, Andrew Smart, Alex Hanna, Emily Denton, Christina Greer, Oddur Kjartansson, Parker Barnes, and Margaret Mitchell. Towards accountability for machine learning datasets: Practices from software engineering and infrastructure. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, pages 560–575, 2021.
- [73] Anne Immonen, Marko Palviainen, and Eila Ovaska. Requirements of an open data based business ecosystem. *IEEE access*, 2:88–103, 2014.
- [74] Azra Ismail and Neha Kumar. AI in global health: the view from the front lines. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pages 1–21, 2021.
- [75] Samireh Jalali and Claes Wohlin. Systematic literature studies: database searches vs. backward snowballing. In *Proceedings of the 2012 ACM-IEEE International Symposium on Empirical Software Engineering and Measurement*, pages 29–38. IEEE, 2012.
- [76] Monique H Jansen-Vullers, Christian A van Dorp, and Adrie JM Beulens. Managing traceability information in manufacture. *International journal of information management*, 23(5):395–413, 2003.
- [77] Indra Joshi and Dominic Cushnan. A buyer’s guide to AI in health and care. Technical report, NHSX AI Lab, 2020.

- [78] Carsten Jung, Henrike Mueller, Simone Pedemonte, Simone Plances, and Oliver Thew. Machine learning in UK financial services. *Bank of England and Financial Conduct Authority*, 2019.
- [79] Rick Kazman and Len Bass. Making architecture reviews work in the real world. *IEEE software*, 19(1):67–73, 2002.
- [80] Patrick Gage Kelley, Joanna Bresee, Lorrie Faith Cranor, and Robert W Reeder. A nutrition label for privacy. In *Proceedings of the 5th Symposium on Usable Privacy and Security*, page 4. ACM, 2009.
- [81] Vijay Khatri and Carol V Brown. Designing data governance. *Communications of the ACM*, 53(1):148–152, 2010.
- [82] John Kindervag et al. Build security into your network’s DNA: The zero trust network architecture. *Forrester Research Inc*, pages 1–26, 2010.
- [83] Nitin Kohli, Renata Barreto, and Joshua A Kroll. Translation tutorial: a shared lexicon for research and practice in human-centered software systems. In *1st Conference on Fairness, Accountability, and Transparency. New York, NY, USA*, volume 7, 2018.
- [84] Dominik Kreuzberger, Niklas Kühl, and Sebastian Hirschl. Machine learning operations (mlops): Overview, definition, and architecture. *arXiv preprint arXiv:2205.02302*, 2022.
- [85] Joshua A Kroll. Outlining traceability: A principle for operationalizing accountability in computing systems. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, pages 758–771, 2021.
- [86] Philippe Kruchten. An ontology of architectural design decisions in software intensive systems. In *2nd Groningen workshop on software variability*, pages 54–61. Citeseer, 2004.

- [87] Abhishek Kumar, Benjamin Finley, Tristan Braud, Sasu Tarkoma, and Pan Hui. Sketching an ai marketplace: Tech, economic, and regulatory aspects. *IEEE Access*, 9:13761–13774, 2021.
- [88] Mary C Lacity and Steven C Lupien. *Blockchain Fundamentals for Web 3.0:-*. University of Arkansas Press, 2022.
- [89] Douglas M Lambert, Martha C Cooper, and Janus D Pagh. Supply chain management: implementation issues and research opportunities. *The international journal of logistics management*, 9(2):1–20, 1998.
- [90] Alexander Lavin, Ciarán M Gilligan-Lee, Alessya Visnjic, Siddha Ganju, Dava Newman, Sujoy Ganguly, Danny Lange, Atılım Güneş Baydin, Amit Sharma, Adam Gibson, et al. Technology readiness levels for machine learning systems. *Nature Communications*, 13(1):1–19, 2022.
- [91] John Linn. Trust models and management in public-key infrastructures. *RSA laboratories*, 12, 2000.
- [92] Sebastian Lins, Konstantin D Pandl, Heiner Teigeler, Scott Thiebes, Calvin Bayer, and Ali Sunyaev. Artificial intelligence as a service. *Business & Information Systems Engineering*, 63(4):441–456, 2021.
- [93] Howard F Lipson, Nancy R Mead, and Andrew P Moore. Can we ever build survivable systems from COTS components? In *International Conference on Advanced Information Systems Engineering*, pages 216–229. Springer, 2002.
- [94] Zoltán András Lux, Felix Beierle, Sebastian Zickau, and Sebastian Göndör. Full-text search for verifiable credential metadata on distributed ledgers. In *2019 Sixth International Conference on Internet of Things: Systems, Management and Security (IOTSMS)*, pages 519–528. IEEE, 2019.

- [95] Inês Araújo Machado, Carlos Costa, and Maribel Yasmina Santos. Data mesh: concepts and principles of a paradigm shift in data architectures. *Procedia Computer Science*, 196:263–271, 2022.
- [96] Tim Mackey. Building open source security into agile application builds. *Network Security*, 2018(4):5–8, 2018.
- [97] Sasu Mäkinen, Henrik Skogström, Eero Laaksonen, and Tommi Mikkonen. Who needs MLOps: What data scientists seek to accomplish and how can MLOps help? In *2021 IEEE/ACM 1st Workshop on AI Engineering-Software Engineering for AI (WAIN)*, pages 109–112. IEEE, 2021.
- [98] Salvatore T March and Gerald F Smith. Design and natural science research on information technology. *Decision support systems*, 15(4):251–266, 1995.
- [99] Mary Ann Marcinkiewicz. Building a large annotated corpus of English: The Penn Treebank. *Using Large Corpora*, 273, 1994.
- [100] Masha McConaghy, Greg McMullen, Glenn Parry, Trent McConaghy, and David Holtzman. Visibility and digital art: Blockchain as an ownership layer on the internet. *Strategic Change*, 26(5):461–470, 2017.
- [101] Albert Meijer and Martijn Wessels. Predictive policing: Review of benefits and drawbacks. *International Journal of Public Administration*, 42(12):1031–1039, 2019.
- [102] Chenglin Miao, Qi Li, Houping Xiao, Wenjun Jiang, Mengdi Huai, and Lu Su. Towards data poisoning attacks in crowd sensing systems. In *Proceedings of the Eighteenth ACM International Symposium on Mobile Ad Hoc Networking and Computing*, pages 111–120, 2018.
- [103] Declan Millar, Dave Braines, Laura D’Arcy, Iain Barclay, Doug Summers-Stay, and Paul Cripps. Embedding dynamic knowledge graphs based on observational ontologies in semantic vector spaces. In *Artificial Intelligence*

- and Machine Learning for Multi-Domain Operations Applications III*, volume 11746, pages 404–413. SPIE, 2021.
- [104] Saurabh Mishra, Jack Clark, and C Raymond Perrault. Measurement in AI policy: Opportunities and challenges. *arXiv preprint arXiv:2009.09071*, 2020.
- [105] Paolo Missier, Khalid Belhajjame, and James Cheney. The W3C PROV family of specifications for modelling provenance metadata. In *Proceedings of the 16th International Conference on Extending Database Technology*, pages 773–776. ACM, 2013.
- [106] Ivan Mistrik, Rami Bahsoon, Peter Eeles, Roshanak Roshandel, and Michael Stal. *Relating system quality and software architecture*. Morgan Kaufmann, 2014.
- [107] Margaret Mitchell, Simone Wu, Andrew Zaldivar, Parker Barnes, Lucy Vasserman, Ben Hutchinson, Elena Spitzer, Inioluwa Deborah Raji, and Timnit Gebru. Model cards for model reporting. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pages 220–229. ACM, 2019.
- [108] Alexander Mühle, Andreas Grüner, Tatiana Gayvoronskaya, and Christoph Meinel. A survey on essential components of a self-sovereign identity. *Computer Science Review*, 30:80–86, 2018.
- [109] Michael D Myers and Michael Newman. The qualitative interview in is research: Examining the craft. *Information and organization*, 17(1):2–26, 2007.
- [110] Iman Naja, Milan Markovic, Peter Edwards, and Caitlin Cottrill. A semantic framework to support AI system accountability and audit. In *European Semantic Web Conference*, pages 160–176. Springer, 2021.

- [111] PJ Nickel and K Vaesen. Risk and trust. *Handbook of risk theory: epistemology, decision theory, ethics and social implications of risk*, pages 857–876, 2012.
- [112] Curtis G Northcutt, Anish Athalye, and Jonas Mueller. Pervasive label errors in test sets destabilize machine learning benchmarks. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1)*, 2021.
- [113] Chris Norval, Jennifer Cobbe, and Jatinder Singh. Towards an accountable internet of things: A call for reviewability. *Privacy by Design for the Internet of Things*, 2021.
- [114] M. Nottingham and E. Hammer-Lahav. Defining well-known uniform resource identifiers (URIs) RFC 5785. April 2010.
- [115] Claudio Novelli, Mariarosaria Taddeo, and Luciano Floridi. Accountability in artificial intelligence: What it is and how it works. *Available at SSRN*, 2022.
- [116] Philipp Offermann, Olga Levina, Marten Schönherr, and Udo Bub. Outline of a design science research process. In *Proceedings of the 4th International Conference on Design Science Research in Information Systems and Technology*, pages 1–11, 2009.
- [117] Cathy O’Neil. *Weapons of math destruction: How big data increases inequality and threatens democracy*. Crown, 2016.
- [118] Linus U Opara. Traceability in agriculture and food supply chain: a review of basic concepts, technological implications, and future prospects. *Journal of Food Agriculture and Environment*, 1:101–106, 2003.
- [119] Ipek Ozkaya, Len Bass, Raghvinder S Sangwan, and Robert L Nord. Making practical use of quality attribute information. *IEEE software*, 25(2):25–33, 2008.

- [120] Ken Peffers, Marcus Rothenberger, Tuure Tuunanen, and Reza Vaezi. Design science research evaluation. In *International Conference on Design Science Research in Information Systems*, pages 398–410. Springer, 2012.
- [121] Ken Peffers, Tuure Tuunanen, and Björn Niehaves. Design science research genres: Introduction to the special issue on exemplars and criteria for applicable design science research. *European Journal of Information Systems*, 27(2):129–139, 2018.
- [122] Ken Peffers, Tuure Tuunanen, Marcus A Rothenberger, and Samir Chatterjee. A design science research methodology for information systems research. *Journal of management information systems*, 24(3):45–77, 2007.
- [123] Felipe Pezoa, Juan L Reutter, Fernando Suarez, Martín Ugarte, and Domagoj Vrgoč. Foundations of JSON schema. In *Proceedings of the 25th International Conference on World Wide Web*, pages 263–273, 2016.
- [124] Elizabeth Pisani, Laura Merson, Amrita Ghataure, Genera Castillo, Anne-Marie Castillo, and Yola Moride. Sharing health research data in low-resource settings: Supporting necessary infrastructure and building on good practices. 2018.
- [125] Alun Preece, D Braines, F Cerutti, G Pearson, and L Kaplan. Coalition situational understanding via adaptive, trusted and resilient distributed artificial intelligence analytics. *NATO STO MP-IST-190*, 2021.
- [126] Alun Preece, Dan Harborne, Dave Braines, Richard Tomsett, and Supriyo Chakraborty. Stakeholders in Explainable AI. *arXiv preprint arXiv:1810.00184*, 2018.
- [127] Bart Preneel. Cryptographic hash functions. *European Transactions on Telecommunications*, 5(4):431–448, 1994.

- [128] Swapna Krishnakumar Radha, Ian Taylor, Jarek Nabrzyski, and Iain Barclay. Verifiable badging system for scientific data reproducibility. *Blockchain: Research and Applications*, page 100015, 2021.
- [129] Narayan Ramasubbu, Chris F Kemerer, and C Jason Woodard. Managing technical debt: Insights from recent empirical evidence. *IEEE Software*, 32(2):22–25, 2015.
- [130] Cedric Renggli, Luka Rimanic, Nezihe Merve Gurel, Bojan Karlas, Wentao Wu, and Ce Zhang. A data quality-driven view of mlops. *IEEE Data Engineering Bulletin*, 44(1):11–23, March 2021.
- [131] Isadora Neroni Rezende. Facial recognition in police hands: Assessing the ‘clearview case’ from a European perspective. *New Journal of European Criminal Law*, 11(3):375–389, 2020.
- [132] Michael Roberts, Derek Driggs, Matthew Thorpe, Julian Gilbey, Michael Yeung, Stephan Ursprung, Angelica I Aviles-Rivero, Christian Etmann, Cathal McCague, Lucian Beer, et al. Common pitfalls and recommendations for using machine learning to detect and prognosticate for COVID-19 using chest radiographs and CT scans. *Nature Machine Intelligence*, 3(3):199–217, 2021.
- [133] Nick Rozanski and Eoin Woods. *Software systems architecture: working with stakeholders using viewpoints and perspectives*. Addison-Wesley, 2012.
- [134] Marcelo Iury S Oliveira, Glória de Fátima Barros Lima, and Bernadette Farias Lóscio. Investigations into data ecosystems: a systematic mapping study. *Knowledge and Information Systems*, 61(2):589–630, 2019.
- [135] Jerome H Saltzer and Michael D Schroeder. The protection of information in computer systems. *Proceedings of the IEEE*, 63(9):1278–1308, 1975.
- [136] Nithya Sambasivan, Shivani Kapania, Hannah Highfill, Diana Akrong, Praveen Paritosh, and Lora M Aroyo. “Everyone wants to do the model work, not the

- data work”: Data cascades in high-stakes AI. In *proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pages 1–15, 2021.
- [137] Beatriz Garcia Santa Cruz, Matías Nicolás Bossa, Jan Sölter, and Andreas Dominik Husch. Public covid-19 x-ray datasets and their impact on model bias – a systematic review of a significant problem. *Medical image analysis*, 74:102225, 2021.
- [138] David Sculley, Gary Holt, Daniel Golovin, Eugene Davydov, Todd Phillips, Dietmar Ebner, Vinay Chaudhary, Michael Young, Jean-Francois Crespo, and Dan Dennison. Hidden technical debt in machine learning systems. In *Advances in neural information processing systems*, pages 2503–2511, 2015.
- [139] Johannes Sedlmeir, Jasmin Huber, Tom Josua Barbereau, Linda Weigl, and Tamara Roth. Transition pathways towards design principles of self-sovereign identity. In *Forty-Third International Conference on Information Systems*. Copenhagen, 2022.
- [140] Fred R Shank. The nutrition labeling and education act of 1990. *Food & Drug LJ*, 47:247, 1992.
- [141] Stephen Shepherd. Vulnerability disclosure: How do we define responsible disclosure? *GIAC SEC Practical Repository, SANS Inst*, 9, 2003.
- [142] Adam Shostack. *Threat modeling: Designing for security*. John Wiley & Sons, 2014.
- [143] Herbert A Simon. *The sciences of the artificial*. MIT press, 2019.
- [144] Jatinder Singh, Jennifer Cobbe, and Chris Norval. Decision provenance: Harnessing data flow for accountable systems. *IEEE Access*, 7:6562–6574, 2019.
- [145] Ben Snait, Deborah Yates, and Ed Evans. Mapping data ecosystems. Technical report, Open Data Institute, June 2021.

- [146] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. A dataset of 101 human action classes from videos in the wild. *Center for Research in Computer Vision*, 2(11), 2012.
- [147] Manu Sporny, Dave Longley, and David Chadwick. Verifiable Credentials data model 1.1. March 2022.
- [148] Cynthia Stohl, Michael Stohl, and Paul M Leonardi. Digital age managing opacity: Information visibility and the paradox of transparency in the digital age. *International Journal of Communication*, 10:15, 2016.
- [149] Niranjan Suri, Kelvin M Marcus, Casper van den Broek, Harrie Bastiaansen, Piotr Lubkowski, and Mariann Hauge. Extending the Anglova scenario for urban operations. In *2019 International Conference on Military Communications and Information Systems (ICMCIS)*, pages 1–7. IEEE, 2019.
- [150] Mariarosaria Taddeo, Tom McCutcheon, and Luciano Floridi. Trusting artificial intelligence in cybersecurity is a double-edged sword. *Nature Machine Intelligence*, 1(12):557–560, 2019.
- [151] Harrison Taylor, Liam Hiley, Jack Furby, Alun Preece, and Dave Braines. VADR: Discriminative multimodal explanations for situational understanding. In *2020 IEEE 23rd International Conference on Information Fusion (FUSION)*, pages 1–8. IEEE, 2020.
- [152] Claartje L ter Hoeven, Cynthia Stohl, Paul Leonardi, and Michael Stohl. Assessing organizational information visibility: Development and validation of the information visibility scale. *Communication Research*, 48(6):895–927, 2021.
- [153] Oliver Terbu, Dmitri Zagidulin, and Amy Guy. did:web decentralized identifier method specification. 2020.

- [154] The Open Data Institute. Trustworthy data stewardship guidebook. Technical report, The Open Data Institute, 2021.
- [155] Lauren Thornton, Bran Knowles, and Gordon Blair. Fifty shades of grey: In praise of a nuanced approach towards trustworthy design. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, FAccT 21, pages 64–76, New York, NY, USA, 2021. Association for Computing Machinery.
- [156] Lauren Thornton, Bran Knowles, and Gordon Blair. Fifty shades of grey: In praise of a nuanced approach towards trustworthy design. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, pages 64–76, 2021.
- [157] Andrew Tobin and Drummond Reed. The inevitable rise of self-sovereign identity. *The Sovrin Foundation*, 2016.
- [158] Richard Tomsett, Dave Braines, Dan Harborne, Alun Preece, and Supriyo Chakraborty. Interpretable to whom? A role-based model for analyzing interpretable machine learning systems. *arXiv preprint arXiv:1806.07552*, 2018.
- [159] Andrew Trask, Emma Bluemke, Ben Garfinkel, Claudia Ghezzou Cuervas-Mons, and Allan Dafoe. Beyond privacy trade-offs with structured transparency. *arXiv preprint arXiv:2012.08347*, 2020.
- [160] Diane-Gabrielle Tremblay, Amina Yagoubi, and Valéry Psyché. Digital transformation: An analysis of the role of technology service providers in Montreal’s emerging AI business ecosystem. In *Digitalization and Firm Performance*, pages 17–44. Springer, 2022.
- [161] CA Van Dorp. A traceability application based on Gozinto graphs. In *Proceedings of EFITA 2003 Conference*, pages 280–285, 2003.

- [162] Uwe Van Heesch, Paris Avgeriou, and Rich Hilliard. A documentation framework for architecture decisions. *Journal of Systems and Software*, 85(4):795–820, 2012.
- [163] Uwe Van Heesch, Veli-Pekka Eloranta, Paris Avgeriou, Kai Koskimies, and Neil Harrison. Decision-centric Architecture Reviews. *IEEE software*, 31(1):69–76, 2013.
- [164] Nancy A Van House, Mark H Butler, and Lisa R Schiff. Cooperative knowledge work and practices of trust: Sharing environmental planning data sets. In *Proceedings of the 1998 ACM conference on Computer supported cooperative work*, pages 335–343, 1998.
- [165] John Venable, Jan Pries-Heje, and Richard Baskerville. FEDS: a framework for evaluation in design science research. *European journal of information systems*, 25(1):77–89, 2016.
- [166] Jillian C Wallis, Elizabeth Rolando, and Christine L Borgman. If we share data, will anyone use them? data sharing and reuse in the long tail of science and technology. *PloS one*, 8(7):e67332, 2013.
- [167] Gang Wang, Bolun Wang, Tianyi Wang, Ana Nika, Haitao Zheng, and Ben Y Zhao. Defending against sybil devices in crowdsourced mapping services. In *Proceedings of the 14th Annual International Conference on Mobile Systems, Applications, and Services*, pages 179–191. ACM, 2016.
- [168] Daniel J Weitzner, Harold Abelson, Tim Berners-Lee, Joan Feigenbaum, James Hendler, and Gerald Jay Sussman. Information accountability. *Communications of the ACM*, 51(6):82–87, 2008.
- [169] Roel J Wieringa. *Design science methodology for information systems and software engineering*. Springer, 2014.

-
- [170] Rebecca Williams, Richard Cloete, Jennifer Cobbe, Caitlin Cottrill, Peter Edwards, Milan Markovic, Iman Naja, Frances Ryan, Jatinder Singh, and Wei Pang. From transparency to accountability of intelligent systems: Moving beyond aspirations. *Data & Policy*, 4, 2022.
- [171] Longqi Yang, David Holtz, Sonia Jaffe, Siddharth Suri, Shilpi Sinha, Jeffrey Weston, Connor Joyce, Neha Shah, Kevin Sherman, Brent Hecht, and Jaime Teevan. The effects of remote work on collaboration among information workers. *Nature Human Behaviour*, 6(1):43–54, 2022.
- [172] Nusrat Zahan, Thomas Zimmermann, Patrice Godefroid, Brendan Murphy, Chandra Maddila, and Laurie Williams. What are weak links in the npm supply chain? In *Proceedings of the 44th International Conference on Software Engineering: Software Engineering in Practice*, pages 331–340, 2022.
- [173] Shuai Zhao, Manoop Talasila, Guy Jacobson, Cristian Borcea, Syed Anwar Aftab, and John F Murray. Packaging and sharing machine learning models via the Acumos AI open platform. In *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 841–846. IEEE, 2018.
- [174] Lin Zhi, Wang Jing, Chen Xiao-su, and Jia Lian-xing. Research on policy-based access control model. In *2009 International Conference on Networks Security, Wireless Communications and Trusted Computing*, volume 2, pages 164–167. IEEE, 2009.

Appendix A

Definition of Terms

For clarity, we provide definitions of terms used in this thesis. First, we provide a definition for the data-driven systems that are the focus of the thesis.

Data-driven Systems: We use DDS to refer to computational systems which use algorithms or processes derived from analysis of large quantities data to make decisions or predictions. Datta, et al., identify DDS as systems which “include machine learning and artificial intelligence systems that use large swaths of data about individuals in order to make decisions about them” [42]. Our definition is broader, and we do not restrict our DDS to including information about individuals. Systems we call DDS are described elsewhere [170] as algorithmic decision-making systems (ADM). We prefer the focus on data of the DDS moniker used here, as it serves to provide a reminder of the origins of such systems, which we build upon in our narrative.

This thesis discusses properties and characteristics of information, and relationships of different stakeholders to information. Definitions of the terms used are given here.

Accountability: Kohli, et al., provide a lexicon of terms for use in Human-Centered Software Systems [83] and define accountability as being “fundamentally about the answerability of actors for outcomes”. In the context of DDS, Kroll instantiates accountability as the ability to “hold the designers, developers, and operators of a computer system responsible for that system’s behaviours” [85].

Our motivation in this thesis is to be able to link actors to their contributions, such that they can become answerable, or be held responsible for the impact of their contributions on DDS.

Appropriate: When considering a DDS for adoption in a particular domain, we suggest that stakeholders consider that it is ‘appropriate’ for use. The intention is to convey that the system should be “ethical in the context of its deployment” [18].

Assurance: The Open Data Institute (ODI) are making ongoing contributions to research on assurance on data and DDS. The ODI provide an explanation of the concept of assurance: “Being assured is about having confidence in an action, result or process. One way to assure people they can have confidence in you is to show that you are reliable or trustworthy, which might require evidence” [145]. We seek to develop mechanisms that can support the processes of both providing and gaining assurance.

Provenance: In data, provenance refers to the process of tracing and recording the origins of data and its movement between databases [29]. Provenance of data was a particular concern during the emergence of the big data era, and a significant research area for scientific databases, where it plays a key role in the validation of data. Here, we see provenance on data and other contributions as a part of the information that can be made available about a DDS. Singh, et al. [144], for example, term the recording of history of actions which influence systems as “decision provenance”.

Oversight: In AI settings, oversight is generally associated with the supervision of a system during its operations, by providing an opportunity – or a requirement – for supervisors to inspect and potentially override the AI system. Here, we are concerned with the components from which DDS are developed, rather than their dynamic, operational state. As such, our use of oversight is more akin to an inspection than ongoing or active supervision.

Scrutable / Scrutability: In their work on ubiquitous, context-aware applications, Assad, et al. [7], described models in their environment as scrutable “if they are designed so that a person who chooses to investigate them can determine just what is modelled.” Here, we adopt a similar meaning – a DDS is scrutable if it designed such that a person investigating the system can understand how it is made. Conversely, it is inscrutable if it cannot be understood, which could arise as a result of incomplete or inaccurate information. Our desire is to provide mechanisms that can help make DDS more scrutable, or to assist those who need to scrutinise DDS – in other words, we seek to increase the scrutability of DDS. Norval, et al., use “reviewability” [113] to describe similar intentions, yet we prefer scrutability, as it implies that more of a critical review or critique can be performed.

Scrutinise / Scrutiny: We adopt the Oxford English Dictionary (OED) definition of scrutiny as “investigation, critical inquiry”. Our objective is to provide mechanisms that support the ability for stakeholders to perform investigations into the supply chains of DDS.

Traceability: The ability to trace component parts through a supply chain, is a core contributor to safety in manufacturing and food production. As such, we adopt the definition of traceability from the international quality standard, ISO 9000¹, as the “ability to trace the history, application or location of an object”, and further “When considering a product or a service , traceability can relate to: the origin of materials and parts; the processing history; the distribution and location of the product or service after delivery”. When applied to DDS, we require the ability to trace datasets, models and other contributing components. In DDS, Kroll [85] observes that “traceability relates the objects of transparency (disclosures about a system or records created within that system) to the goals of accountability”

¹<https://www.iso.org/obp/ui/#iso:std:iso:9000:ed-4:v1:en>

Trust: The definition of trust as “the degree to which subject A has confident positive expectations that object B will fulfil its obligations in context C to limit L” from Lacity and Lupien [88] is adopted here, as it is simple and elegant, and includes trust placed both in other humans, and in objects – such as communities, or data-driven systems. The definition accounts for the fact that trust is context dependent and has limits, depending on circumstances. High-stakes situations, for example, require very high degrees of trust to be placed by subject A on object B, whereas in other contexts the limits may be much lower.

Trustworthy: The OED defines trustworthy as reliable or dependable. Whilst simple, this definition is appropriate for documentation of DDS and components. We require that information provided is reliable, and can be depended upon.

Transparency: According to Kohli, et al. [83], transparency can be succinctly described as “the disclosure of system internals to look under the hood of a given technology”. Transparency results from information on shared digital assets being made visible [148]. We use transparency as a property of organisations or of systems.

Visibility: Stohl, et al. [148], characterise information visibility as the combination of three attributes: availability of information, approval to share information, and accessibility of information. Stakeholders who provide assets towards development of a DDS can provide different levels of visibility on information about their assets, which affects the transparency of the system.

Verifiable: Our use of verifiable relates to the ability to check the integrity of a claim that is being made. It does not necessarily mean that the claim itself is truthful or correct - we can verify that a claim was made by a particular party, and has not been tampered with, or revoked. In such a way, we can hold the party making the claim accountable for what they claim, and this can be used as evidence and contribute to our notion of the trustworthiness of the party. This definition is in line with that of the W3C Verifiable Credentials Data Model [147],

which describes a verifiable credential as being “authentic and timely”, but asserts that “verification of a credential does not imply evaluation of the truth of claims encoded in the credential”.

Verifiable Oversight: Our intention is that stakeholders can achieve verifiable oversight on DDS. By this, we mean that information can be requested which enables them to scrutinise systems, providing traceability on constituent components and identifying parties that are accountable for such components. The stakeholders can then determine if the parties are trustworthy, and if the system is likely to be appropriate for use in its intended deployment context.

Appendix B

Presentation to Expert Review Panel

The material provided in the following pages was shared with a group of experts from industry, academia and the non-profit sector as part of the semi-structured interviews which were conducted with the Expert Review Panel. This work is described in Section 1.6.2.

Oversight and Accountability in Data-driven Systems (DDS)

Iain Barclay - PhD Researcher, Cardiff University

Discussion forms part of final evaluation of research

- A few slides to set context
- Some questions to guide discussion
- Keen to hear your thoughts and opinions

1

Housekeeping

- Please complete consent form
- *Discussion will be recorded*
- Background...
 - Sector?
 - Role?
 - Experience in data and data-driven systems?
 - Familiarity with blockchain and decentralised technologies?

2

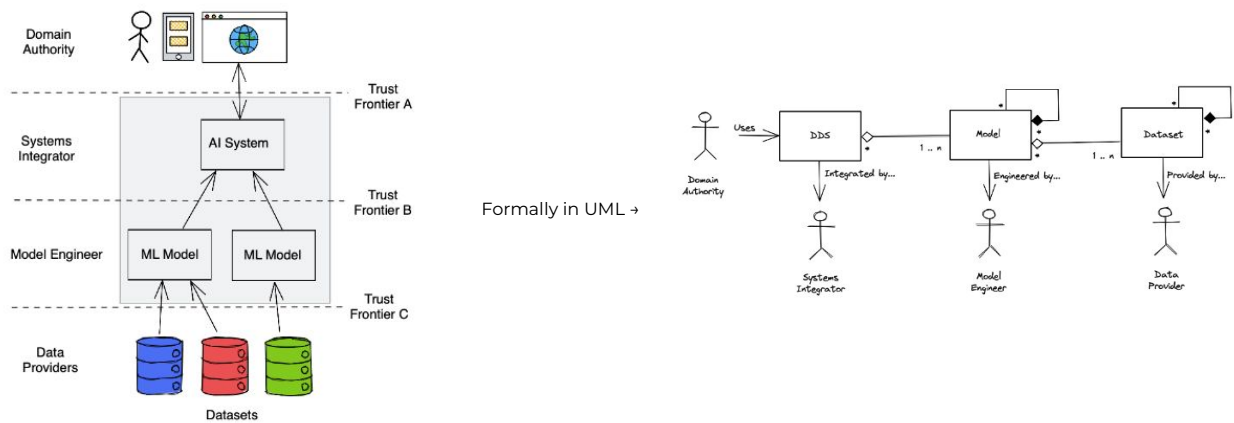
Introduction: Last Mile AI Challenge

- Data from many sources is used to build data-driven products (eg. AI/ML)
- Data is curated and aggregated, and experts build models and then products
- These AI/ML products go out in the field...
 - Used by "domain authorities" - users, not data scientists or AI engineers
 - How can practitioners, etc. check and monitor for ongoing suitability?
- Oversight is important in establishing confidence and trust in tools
- My research has been on providing oversight into multi-party data systems

"In the case of **data-driven health care**, children's treatment or medication **should not be based on adults' data** since this could cause **unknown risks** to children's health"
UNICEF, Policy Guidance on AI for Children, 2020.

3

Roles and Boundaries



- 1) DDS are made of contributions from many parties
- 2) There is not always a direct relationship between the parties
- 3) There may be a tension between a need to get information and a desire to protect it

4

A Data Supply Chain

- Machinery and food have well documented “supply chains”
 - eg. a Tractor has an engine, engine is made from various sub-assemblies
- Supply Chains are documented with a “Bill of Materials”
 - Provides traceability and transparency
- DDS are made from different components, produced by different parties
 - Data - collection, curation, labelling, etc.
 - Models - design, development, testing.
 - Different assets produced at different stages
- Is a “Bill of Materials” useful for a DDS? Is it possible?

5




A Verifiable DDS Supply Chain

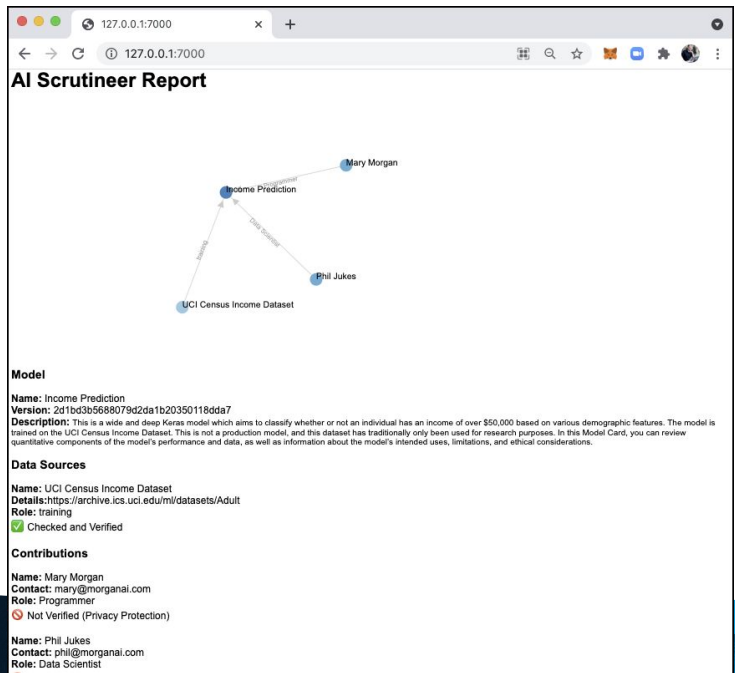
- Oversight is provided across the DDS supply chain
- Make “Shared Data” and ML systems more trustworthy
 - Anyone can certify and check qualities - creators, regulators, etc.
- Parties take accountability for their contributions
 - Make data and models “trustworthy”
 - Digital signatures used to “back up” quality claims
 - Responsibility is taken for parts of the system
 - Trust develops between participants
- The need to maintain confidentiality is recognised
 - Not all information is available to everyone
 - Information can be requested, but owner decides

“Creating a trustworthy data regime that ... enables responsible data use will ensure that the benefits of the data revolution are felt by all people, in all places.”
UK Government response to the consultation on the National Data Strategy, 18 May, 2021.

6

The AI Scrutineer

- Documentation for DDS includes a QR code
 - Scanning the code launches the AI Scrutineer
 - System is “unpacked” and checked in real time
- “AI Scrutineer” gives an overview of the system, and identifies and checks components:
 - Shows a  where claims can be verified, and accountable parties identified
 - Shows a  where claims can't be verified
 - Shows a  where issues are found, for further investigation

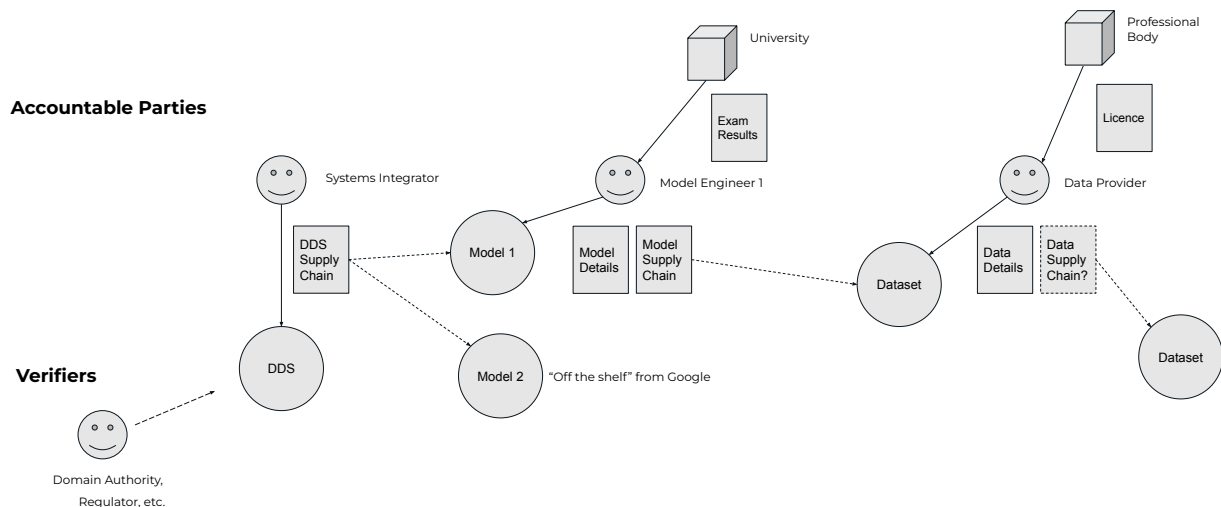


Make Systems Trustworthy

- Adds “trustworthiness” to data and systems : peer-to-peer, and official certifications
 - Information verifiable, but not “public” - privacy/confidentiality protections
 - Provides Users/Practitioners with “real-time” assurance
 - Verifiable evidence on underlying contributions - data, people
- Shows that data still considered “good” when checked
- Deeper, selective disclosure on consent basis

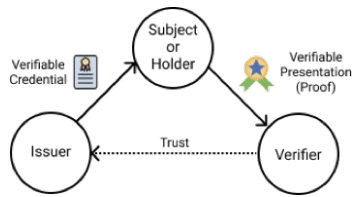
8

A Verifiable Data Supply Chain



9

Triangles of Trust



- An “Issuer” signs statements (claims) about a “Subject”
- A “Verifier” can check the signed proof
 - If Verifier trusts Issue, they can trust claims about Subject
- Subject can be a person, or a “thing”
 - - eg. a system, a model, or a dataset

Issuer of Signed Claims	Subject of Claim	Claims Made (and signed)
University	Model Engineer	Qualifications passed
Model Engineer	An ML Model	Facts about the model <ul style="list-style-type: none">• Features• Data Sets used• Governance Policies / Standards
Systems Integrator	The DDS	The “Supply Chain” of parts and contributors
Data Provider	A dataset	Facts about the dataset <ul style="list-style-type: none">• Metadata• Governance Policies / Standards

Appendix C

DDS BOM Schema Code Listing

Code shown below (Listing C.1) is a JSON Schema definition that represents DDS assets using attributes from Table 4.2.

```
{
  "$schema": "https://json-schema.org/draft-04/schema#",
  "title": "JSON schema for Data-driven Systems",
  "type": "object",
  "description": "A bill of materials for a DDS",
  "properties": {
    "billOfMaterials": {
      "type": "array",
      "items": {
        "$ref": "#/definitions/Asset"
      }
    }
  },
  "required": [
    "billOfMaterials"
  ],
  "definitions": {
    "Asset": {
      "type": "object",
```

```
"properties": {
  "id": {
    "type": "string"
  },
  "descriptiveFields": {
    "type": "string"
  },
  "metadata": {
    "type": "string"
  },
  "assetType": {
    "type": "string"
  },
  "knownDependencies": {
    "type": "array",
    "items": {
      "$ref": "#/definitions/Asset"
    }
  },
  "provider": {
    "type": "string"
  },
  "verificationRoute": {
    "type": "string"
  }
},
"required": [
  "id", "descriptiveFields"
]
}
```

Listing C.1: JSON Schema for an Asset

```

{
  "billOfMaterials": [
    {
      "id": "$CCTVMonitor",
      "descriptiveFields": "Name - CCTV Monitor",
      "assetType": "DDS",
      "knownDependencies": [
        {
          "id": "$DNN",
          "descriptiveFields": "Name - DNN",
          "assetType": "Model"
        }
      ],
      "provider": "Anglova Law Enforcement"
    },
    {
      "id": "$DNN",
      "descriptiveFields": "Name - DNN",
      "assetType": "Model",
      "knownDependencies": [
        {
          "descriptiveFields": "Name - 3DMobileNet",
          "assetType": "Model"
        },
        {
          "descriptiveFields": "Name - VGGish",
          "assetType": "Model"
        },
        {
          "id": "$Curated_UCF101",
          "assetType": "Dataset"
        }
      ],
      "provider": "Taylor et al"
    }
  ]
}

```



```

    },
    {
      "id": "$Curated_UCF101",
      "descriptiveFields": "Name - Curated_UCF101",
      "assetType": "Dataset",
      "knownDependencies": [
        {
          "descriptiveFields": "Name - UCF101",
          "assetType": "Dataset"
        }
      ],
      "provider": "Taylor et al"
    }
  ]
}

```

Listing C.2: BOM for CCTV Monitor Scenario in JSON

```

{
  "billOfMaterials": [
    {
      "id": "$Aurora",
      "descriptiveFields": "Name - Aurora",
      "assetType": "DDS",
      "knownDependencies": [
        {
          "id": "$DialogFlow",
          "assetType": "Model"
        }
      ],
      "provider": "CM"
    },
    {
      "id": "$DialogFlow",
      "descriptiveFields": "Name - DialogFlow",

```

```

    "assetType": "Model",
    "knownDependencies": [
      {
        "id": "$englishModel",
        "assetType": "Model"
      },
      {
        "id": "$portugueseModel",
        "assetType": "Model"
      }
    ],
    "provider": "Google"
  },
  {
    "id": "$englishModel",
    "descriptiveFields": "Name - EN text to text",
    "assetType": "Model",
    "knownDependencies": [
      {
        "id": "$englishData",
        "assetType": "Dataset"
      }
    ],
    "provider": "Google"
  },
  {
    "id": "$portugueseModel",
    "descriptiveFields": "Name - PT text to text",
    "assetType": "Model",
    "knownDependencies": [
      {
        "id": "$portugueseData",
        "assetType": "Dataset"
      }
    ],

```

```
    "provider": "Google"
  },
  {
    "id": "$englishData",
    "descriptiveFields": "Name - EN Advice",
    "assetType": "Dataset",
    "provider": "CM"
  },
  {
    "id": "$portugueseData",
    "descriptiveFields": "Name - PT Advice",
    "assetType": "Dataset",
    "provider": "CM"
  }
]
```

Listing C.3: BOM for Aurora Scenario in JSON

Appendix D

AI Scrutineer

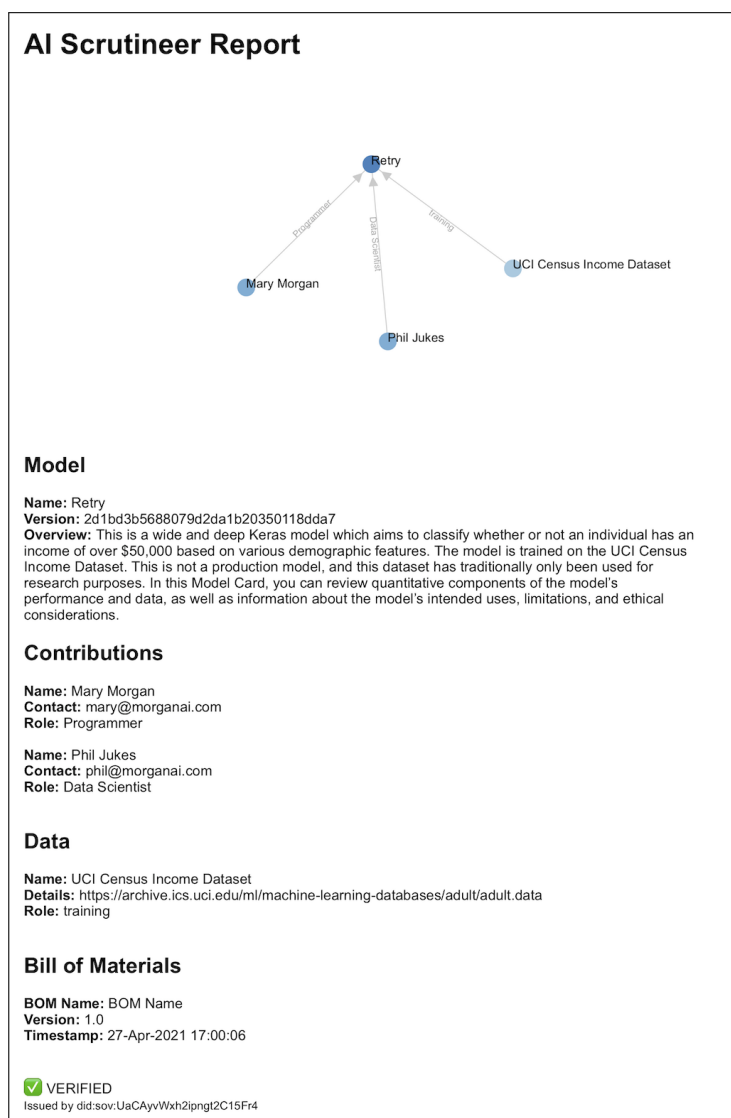


Figure D.1: Screenshot of the AI Scrutineer Web Interface

Decision-Centric Architecture Review (DCAR)

A software architecture is not typically designed in the context of a well-defined, coherent, and self-contained problem. Typically there will be a complex set of interrelated aspects of social and technical challenges, which Eloranta, et al., call “decision forces” [51] . Similarly, there is not just one way to solve a problem, but a variety of potential solutions. Software architects make decisions about different design options, balancing many forces. Different approaches have consequences, and trade-offs between different forces have to be found. The Decision-Centric Architecture Review [163] (DCAR) method defines a format in which architects and reviewers can consider the validity of architectural design decisions, in the context of the decision forces at play in the environment. Such “architecture decisions”, in DCAR terminology, determine the structure of a software system, and are influential in making sure that a system can satisfy its quality attribute requirements. Kruchten [86] groups architecture decisions into three types: existence decisions, property decisions, and executive decisions:

Existence decisions are concerned with the presence of architectural elements, their prominence in the architecture, and their relationships to other elements. Examples of existence decisions include the choice of a particular software framework, the decision to apply a software pattern, or to employ an architectural tactic.

Property decisions concern general guidelines, design rules, or constraints. In this thesis, the decision to use SSI as a technical constraint is a property decision. Property decisions implicitly influence other decisions, but are usually not visible in the architecture unless explicitly documented.

Executive decisions concern the process of creating the system, instead of affecting the system as a product itself. These tend to be driven by the organisation, and may have financial or methodological aspects to them, such as the number of developers that can be assigned to a project, or the use of agile processes.

Eloranta, et al. [51], find that existence decisions tend to have the highest impact on the ability of a system to meet its quality goals. Property decisions are also important, as they complement the requirements and help to explain the existence decisions. Executive decisions tend not to be considered in an architectural evaluation.

In a DCAR evaluation participants identify and clarify architectural decisions, and the relationships between those decisions. The objective is to try to understand different influences and consequences, and determine whether decisions made by the architect are sound, or need to be re-considered. A DCAR evaluation has seven steps:

1. **Management Presentation:** A management representative introduces the business viewpoint, outlining the application domain, the main financial drivers, and the business model - this will identify business goals, such as time to market or low price.
2. **Architecture Presentation:** The architects introduce the system, including the system objectives, architecturally significant requirements, and the main architecture decisions taken and the rationale behind those decisions. Reviewers may try to identify additional decisions by asking questions related to quality attribute requirements; Reviewers note down decisions and potential decisions.
3. **Decision Forces and Decision Completion:** Architecture decisions and their relationships are clarified, and a decision relationship diagram is created. Mutual agreement is reached on the correctness of the decision description. Forces relevant to each decision are identified, and presented as a simple bullet list, using domain-specific vocabulary.
4. **Decision Prioritisation:** The group identify the most important decisions for further analysis. These might include business critical decisions, decisions related to important quality attributes, any intensively discussed decisions or expensive decisions.
5. **Decision Documentation:** The architects document each of the selected decisions. Each architect selects a few decisions they are knowledgeable about. Decisions are documented by describing the architectural solution, the problem it solves, arguments in favour of the solution, arguments against the solution, and a list of alternatives that were considered.
6. **Evaluation:** The architects present the decisions they documented, and then the reviewers propose further arguments either in favour of, or against the applied solution. The decision forces and the decision relationship view are used to challenge each decision. The documentation of the decisions and the decision relationship view are updated during the process. All participants discuss whether the arguments in favour of the decision outweigh the arguments against it, and stakeholders decide whether the decision is sound or needs to be re-visited.

7. Retrospective and Report: The findings of the review, including decisions taken, alternatives considered and organised and documented, with arguments for and against the chosen solution, and any issues that need to be raised.

Following application of DCAR, Cruz, et al. [41], reflected that “while quality attributes are not directly addressed in the method, DCAR-using reviewers will find it necessary to consider appropriate discussion about them when challenging decisions. We believe this characteristic makes DCAR a particularly appropriate method for experienced software architects and architecture reviewers, as they can be expected to be knowledgeable about quality attributes.”

Decision Descriptions from DCAR Review

Decision Name	SSI Standard Interfaces
Problem or Issue	Provide verifiable accountability on claims
Solution	Exchange verifiable credentials with key-value pair payloads
Alternatives	Use custom interactions to provide specific enhancements
Arguments for	Interoperability with ecosystem software; Lower costs, through open source support
Against	Extending interfaces would offer richer M2M interactions

Table F.1: Decision Description: Standard SSI Interfaces

Decision Name	Use VCs for Metadata
Problem or Issue	Provide verifiable, owned proof of claims
Solution	Encode values, sign, and store a VC
Alternatives	Maintain records in a database or file
Arguments for	VC can encode key-value pairs to represent metadata; Signing provides tamperproof metadata; Signing provides accountability; Interoperable with SSI wallet software
Against	Requires SSI infrastructure deployment

Table F.2: Decision Description: Use VCs for Metadata

Decision Name	Encode BOM as a VC
Problem	Maintain record of contributions to DDS
Solution	Encode values, sign, and store BOM as a VC
Alternatives	Maintain records in a database or file
Arguments for	BOM VC can encode key-value pairs with schema; Signing provides tamper-proof metadata; Signing provides accountability; Standardises approach through system
Against	Practically, interacting with BOM requires additional HCI

Table F.3: Decision Description: Encode BOM as VC

Presentation of Asynchronous DCAR Review

The material provided in the following pages was shared with a group of colleagues and peers in order to solicit their opinions on the design decisions made in Chapter 6. The method and results of the evaluation are presented and discussed in Section 6.5.

Decision-Centric Architecture Review

This document describes a system designed in my PhD research. The purpose of this document is to collect colleagues' opinions on the design decisions made that have led to the proposed approach.

Background

In the research, we have identified a need to design a technical solution that can provide oversight on data-driven systems (aka AI systems) so that people using the systems in their work can have oversight on the constituent parts of the system (ie. any ML models, or dataset used) and be assured that these assets are suitable for use (ie. appear to be of good quality, or have a good reputation, etc.). We think this need is especially true in “high stakes settings” - such as healthcare, education, etc. We also want to use digital signatures to place accountability on the providers of any components used, so that any claims they make about datasets or ML models they provide can be traced back to those making them.

We have designed a solution that uses a supply chain “Bill of Materials” (BOM) to let a systems integrator (SI) record all the parts of the system that they provide - this BOM is encoded as a verifiable credential (VC), and is signed by the SI - as a proof that they are taking accountability for the system. Within the BOM, they can identify any models and datasets used, which in turn may have BOMs, signed by their own providers. The system or any of its parts might also have other VCs which contain signed claims about certain qualities, metadata, licence conditions, etc. - As a result, anyone interested in the system can request the BOM from the system or any of its parts, and then request any other credentials they are interested in - providing them with oversight on the system as a whole, and letting them see who is taking accountability for parts of the system.

Objective

Please review the decisions that are presented below. We want to make sure that the business goals can be satisfied by the architecture - because the decisions made (and described below) are good decisions.

Usually this is intended to be a collaborative discussion, held in a meeting - so please feel free to make use of comments in the document. I am also happy to have a discussion about any aspects - please message me, or email barclayis@cardiff.ac.uk if that would be helpful.

Assumptions

Self-sovereign Identity software and infrastructure becomes widely available and deployed, based on efforts from government and other agencies to use this approach for identity and other needs - as such, there is a large choice of commercial and open source software implementations for issuing, storing and using credentials, and the model is well understood.

Business Goals

The system is designed to meet the following business goals:

- The solution adopts SSI Principles (this is a technical constraint).
- The solution is cost effective through its lifecycle.
- The solution can be widely adopted.

Decision Forces

We have identified business-related **decision forces** that should be taken into consideration during evaluation of the proposed architecture.

These decision forces are:

- SSI implementations are available as open source
- The system will be deployed when SSI infrastructure is more mature
- Running an SSI software agent will be no more complex than (e.g.) running a web server, so is not to be considered as a barrier to adoption

Key Decisions Made in the Architecture Design

The goal of this review is to evaluate the decisions made, based on the business goals and the “decision forces”, as presented above (supporting diagrams and other material are also included in the Appendix, in case they are helpful).

Decision 1: Use Standard SSI Interfaces

Summarised in Table B.1 - In order to support the widest range of access to the system, via different levels of commercial and open source software, we propose to use open standards for interactions with the system, and not modify the protocols in any way.

Decision Name	SSI Standard Interfaces
Problem or Issue	Provide verifiable accountability on claims
Solution	Exchange verifiable credentials with key-value pair payloads
Alternatives	Use custom interactions to provide specific enhancements
Arguments for	Interoperability with ecosystem software; Lower costs, through open source support
Against	Extending interfaces would offer richer M2M interactions

Table B.1: Decision Description: Standard SSI Interfaces

Decision 2: Use Verifiable Credentials for Metadata

Summarised in Table B.2 - In order to put accountability onto claims made about assets contributed to systems, we propose to use signed digital credentials to store claims made.

Decision Name	Use VCs for Metadata
Problem or Issue	Provide verifiable, owned proof of claims
Solution	Encode values, sign, and store a VC
Alternatives	Maintain records in a database or file
Arguments for	VC can encode key-value pairs to represent metadata; Signing provides tamperproof metadata; Signing provides accountability; Interoperable with SSI wallet software
Against	Requires SSI infrastructure deployment

Table B.2: Decision Description: Use VCs for Metadata

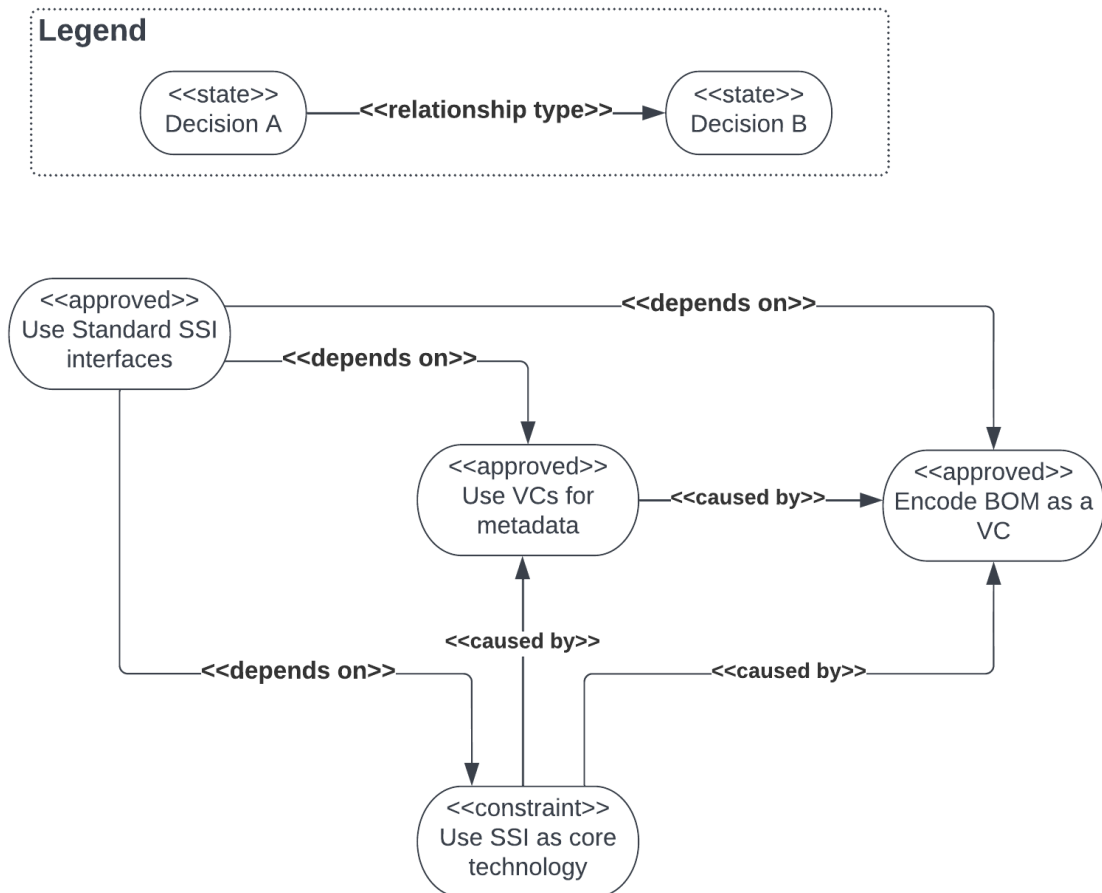
Decision 3: Encode BOM as a VC

Summarised in Table B.3 - In order to keep a record of the contributions to a system, or to any models or datasets used, we propose to create a digital “bill of materials” record, which will be signed and stored as a verifiable credential.

Decision Name	Encode BOM as a VC
Problem	Maintain record of contributions to DDS
Solution	Encode values, sign, and store BOM as a VC
Alternatives	Maintain records in a database or file
Arguments for	BOM VC can encode key-value pairs with schema; Signing provides tamper-proof metadata; Signing provides accountability; Standardises approach through system
Against	Practically, interacting with BOM requires additional HCI

Table B.3: Decision Description: Encode BOM as VC

Decision Relationships



Evaluation

Please consider whether you think the decisions made were correct? Recalling the business goals (below) - do you think these decisions will support these goals being met?

- The solution adopts SSI Principles (this is a technical constraint).
- The solution is cost effective through its lifecycle.
- The solution can be widely adopted.

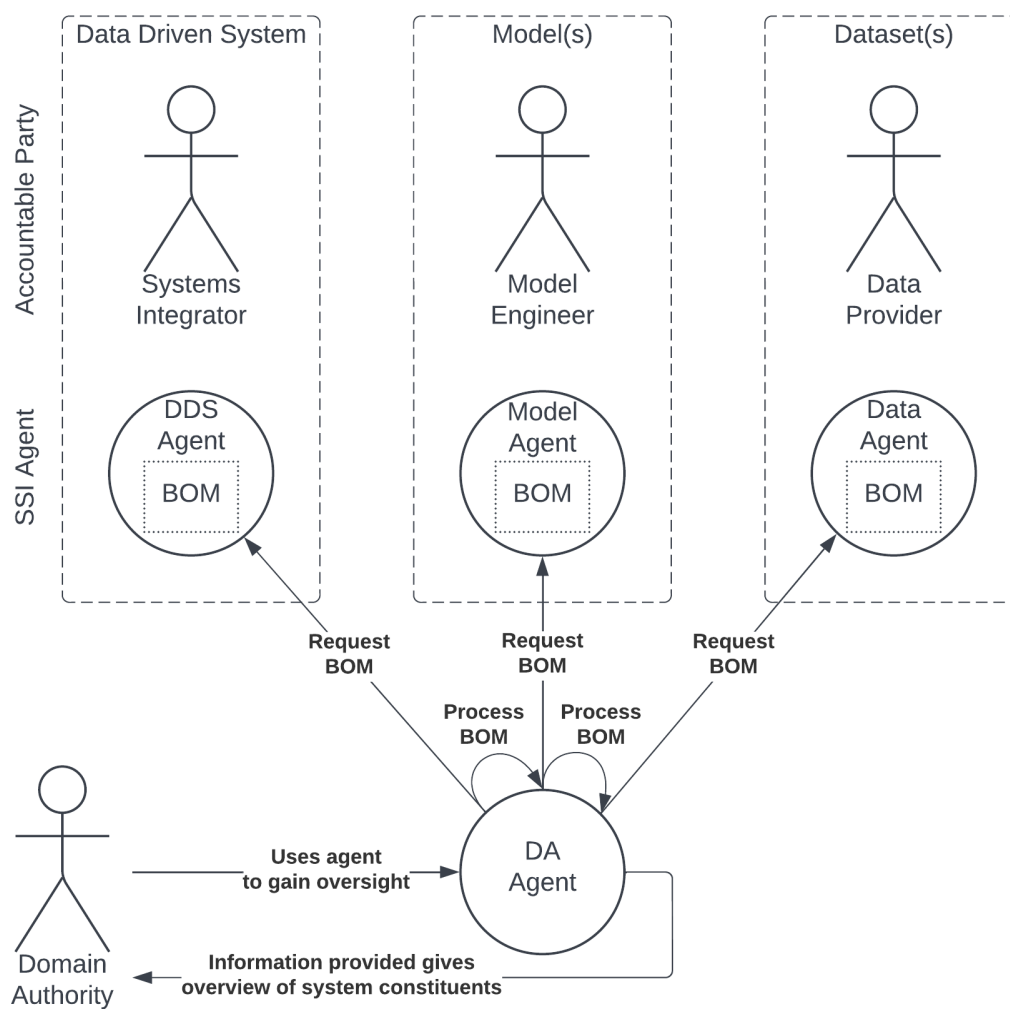
Are there any decisions that you feel should be re-visited, or anything else you wish to add?

Please submit your thoughts via this form - <https://forms.gle/wcneBQ6uqSfopcgT6>

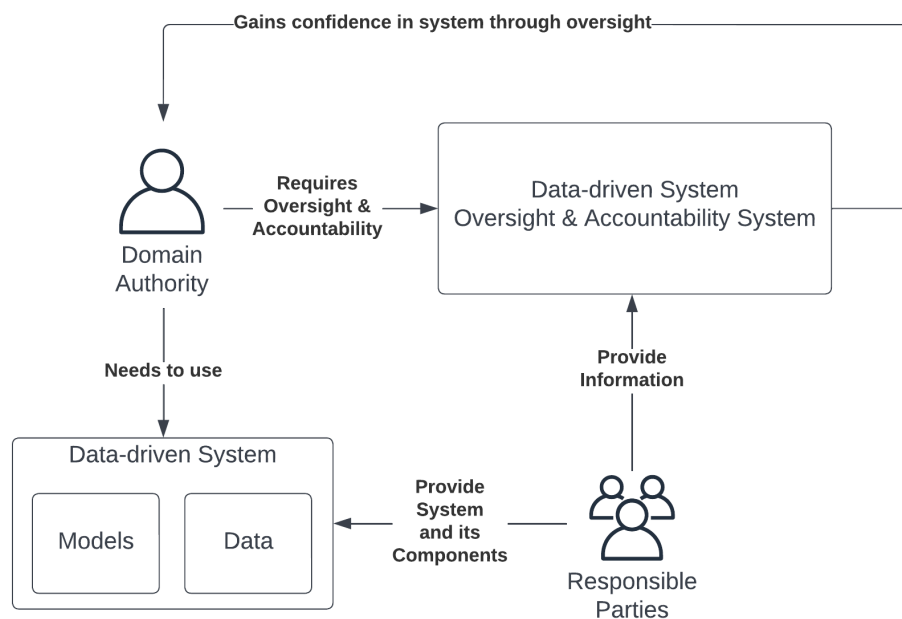
(Please also free to send email to barclayis@cardiff.ac.uk or message me)

Appendix - Supporting Diagrams

System Entities and Process Flow



Context View



Functional View

