SOLID Pods for social media

Daniel Klausen

Master's Thesis
Department of Information Science and Media Studies
Supervisor: Csaba Veres
Co-Supervisor: Enrico Motta
University of Bergen

May 31, 2023

Contents

List of Figures

0.1 Acknowledgements

I would like to thank Evy-Ann for helping me through all the struggles. Anika, Victoria, and Carla for their eternal patience. Daniel and Emma for suffering through my writing. There are many other people who also deserve my thanks, I am lucky to be able to deliver this thesis, the time during Corona made me want to quit. It was a hard time for everyone, being a student at the time was terrible, sitting at home, with two Zoom lectures a week. Not socialising or having an academic environment to support oneself was terrible, I am happy that I stuck with it and did not give up, even if I wanted to. Finally, I need to thank my parents for their confidence and belief in me and my supervisors, Csaba Veres and Enrico Motta for their advice and help.

> Once men turned their thinking over to machines in the hope that this would set them free. But that only permitted other men with machines to enslave them. - Frank *Herbert* (1965)

Daniel Klausen
Bergen, 31.05.2023

"My dynamite will sooner lead to peace than a thousand world conventions. As soon as men will find that in one instant, whole armies can be utterly destroyed, they surely will abide by golden peace. -Alfred Nobel "

(Charlton, 2002, p.114)

0.2 Abstract

Much has been discussed about the future of the Internet, and many speculate on its future development or if it has a future at all. A central man to these speculations is the inventor of the internet himself, Sir Tim Berners-Lee. He, along with others has created SOLID, a specification and what Berners-Lee has described as: "Solid is a mid-course correction for the Web" *The_Solid_Team* (2021a). Like Alfred Nobel before him, his invention did not become exactly the tool he had envisioned. In this thesis, I will discover what possibilities there are for using SOLID pods for social media with decentralized storage. To be able to validate the data in the pods I will be testing shape technology. There are two main shape technologies SHACL and ShEx, and I will discuss them both and use SHACL to test bookmarks in the test site LexiTags. For this thesis, a hybrid approach to methodology will be used, as a combination of design science research and critical theory research paradigm will be used in conjunction. What SOLID proposes to achieve could lead to an interesting paradigm shift. This creates an interesting backdrop for understanding how it works and how it can be combined with other available technologies. SHACL as a concept has potential, but there are challenges that the system would benefit from having resolved to fulfil its full potential. Decentralized sites have challenges other sites that operate traditionally do not have, these challenges need to be addressed before solutions like these can be realised. If decentralized sites are to be realised, they will need systems such as shapes technology and SOLID.

Chapter 1

Introduction

The internet has been a big influence on how we communicate and work together, it brought knowledge and connected people from all over the world. Never before has the world been smaller and had more possibilities. When Sir Timothy John Berners-Lee made the World Wide Web in 1990 (*Klint*) he could not have foreseen what pitfalls would befall the travellers on his World Wide Web. The first era of the Internet was a beautiful expansion of culture and connection before the end all be all was monetization, innovating profit extraction, and venture capital leveraging. After the dot-com bubble there was a restructuring of how the Internet made money, no longer was there a plethora of investors ready to throw money at any Internet endeavour. The internet has changed since the days before the dotcom-bubble, now the biggest companies and corporations look at personal information and data collection for advertising, as Shoshana Zuboff calls *The Discovery of Behavioral Surplus* (*Zuboff*, 2019, Chapter 3. IV). From its inception in 1990 and up until the dotcom bubble the monetization of the internet was difficult, as proven indirectly by the dotcom bubble. *Goodnight and Green* (2010)

> Companies built out on anticipated future demand, and in turn the building signaled reflexively future profits, thereby calling up more investment. Together, all these activities were read as convergent signals that the new economy was here to stay. It was a house of cards. (*Goodnight and Green*, 2010, Crash)

In the last 20 or so years, the internet has changed considerably and rapidly, a tool built for connecting people, now largely used for surveillance capitalism, and the exploitation of personal information. Manishkumar points out in the section dedicated to "Exploitation of the present web" (*Tuba et al.*, 2021, pg.258), where he discusses the use of SOLID pods (*The_Solid_Team* (2021b)) to possibly remedy this exploitation. With the heightened focus we have seen on information gathering and information harvesting via a catch-22 [1], personal data protection seems to be one of the biggest challenges of the digital age, with no obvious solution on the horizon. When we discuss the value of personal data it is important to realize the gargantuan scope of the money involved.

> The data produced when individuals use the internet has become a critical resource that can create a significant imbalance of power in favor of today's

[1] opting out might not be a realistic option

corporations, who appear to have a limitless capacity to collect and use this information (Cadogan 2004; Greenstein 2015). *Agogo* (2020)

1.1 Motivation

I have not always had a personal interest in the privacy concerns we are faced with in the digital age, I could not envision the implications of the systems I used growing up. My household got the internet in 1995, and from then I surfed the internet, the older, wilder internet of the late 1990s and early 2000s was a different experience compared to the modern more moderated internet. I did not consider there to be any downside to the internet, not beyond vague warnings about 'dangerous men', seen in the commercials, sitting in a dark room with hoodies and doing 'bad stuff'.

The realization that it might be the institutions that were problematic did not appear for a long time yet. I still remember when Facebook came to my high school, and a few people did not want to accept Facebook's privacy policy and decided not to be on Facebook like the rest of us. At the time, I found this behaviour weird, and I remember thinking that we will be surveilled no matter what we do and so it is pointless to not use these sites and services.

Fast forward 11 years and I handed in my Bachelor's thesis in Digital Culture titled "Do we pay with information?" *Klausen* (2017), where I detailed how we pay for all the 'free' services (and some paid) with our information. My focus switched from: "some things are just unavoidable" to "we can change how the world works". I changed as well, I moved from digital culture to information sciences, I wanted to not just understand how the digital age shapes us, but how we can actively change it for the better, to not only observe the issues we face but maybe change a small part of the whole for the better. This is the key reason behind wanting to work on this thesis, and why I decided to go with the critical theory research paradigm that will be discussed in the methodology chapter.

1.1.1 Impersonal Motivation:

The need for increased control of personal data has been evident for many for some time, and it became evident to the public eye when the *Cambridge Analytica* scandal was presented to the public *Cadwalladr and Graham-Harrison* (2018). The public got a stark reminder of the power of personal information gathering and the powerlessness of individuals in keeping one's personal information private. One counterargument often encountered is the "opt-out" argument, where when faced with privacy concerns regarding a service or a system, advocates of said system say "You can opt-out, not use it; delete Facebook if you don't like the user agreement" however this is not a real option. This argument can be countered in two ways, one: The user agreement is a mess, and nobody but the consumer advocates read them; additionally, how can we know that the sites follow the laws and user agreements when we have a lack of transparency of their business? Two: opting out makes your life more difficult and poorer, and it is hard to compete with free, especially when the best services are free for example, Googles Gmail. Gmail offers the best spam filter in the world, using a different email client will open you up to possible scams and viruses for your system, how can you compete with free? As *Siva Vaidhyanathan* points out in his book *The Googlization of Everything* (*Vaidhyanathan*, 2012, 19).

> Opting out of any Google service puts the Web user at a disadvantage in relation to other users. The more Google integrates its services, and the more interesting and essential the services that Google offers, the more important Google use is for effective commerce, self-promotion, and cultural citizenship. So the broader Google's reach becomes - the more it Googilizes us- the more likely it is that even informed and critical internet users will stay in the Google universe and allow Google to use their personal information. (*Vaidhyanathan*, 2012, 90)

He ends his book with this:

> We must build systems that can serve us better, regardless of which companies and technologies thrive in the next decade. Most important, we should learn to beware of false idols and empty promises. The future of knowledge and thus the future of the species-depends on getting this right. (*Vaidhyanathan*, 2012, 210)

The book was published in 2012, over ten years ago, and not much has changed. This is part of the societal background for this thesis, and also why I think SOLID or other technology like it is needed in the future.

1.2 Problem Statement

"This project will investigate technical challenges in managing social media data in pods." (*Veres*)

This project will be looking at Shapes for social media and how they could interact with the SOLID framework (*The_Solid_Team* (2021b)) to create a standardization for social media shapes. A shape defines the fields and structure that clients and apps can expect to find in a view over a piece of data, how the data is structured and presented, and what information is given or asked for. Social Media applications can contain a complex tangle of data: posts, likes, comments, discussion threads, shares, photos, videos, etc. We would need this type of standardization to use SOLID for social media.

1.3 Research Questions

- What sort of data shapes do we need for social media?
- How to manage the enforcement of shapes in the pod?
- How can we manage evolving requirements for data in pods?
- Does SOLID PODs enable data reuse and innovation in the application space?

1.3.1 Objectives

The objectives for this thesis are focused on shapes and exploring what they are and the potential use case for shapes, with changing data requirements. More specifically how difficult are they to work with, what are the limitations, and what are the possibilities? Discovering the aptitudes and difficulties of working with shapes.

1.3.2 Goals

- investigate shapes technology
- Create shapes that match the bookmarks of our site
- Change the shapes and data
- Introduce semantic meaning to the bookmarks with DBpedia

1.4 Contribution

This thesis will contribute to understanding the development of the SOLID specification and shapes for social media. The goal is to understand the requirements for a model or a system that can be used as a first step to develop a way to use SOLID for 'real' social media sites. It brings us closer to having the shapes to use with social media sites that fit their data requirements. The use case for SOLID pods as a mainstream service for social media is racked with hurdles to overcome; I would speculate and guess that it would take massive consumer support or a big governing body (such as the European Union) to force social media sites even to allow users to use a system like this. That type of problem is far out of the reach of this thesis, however, and this thesis will contribute to developing the SOLID system space.

1.5 Thesis outline

The structure of the thesis will be as follows. Firstly we look at the background for what has been done, a look at SOLID and shapes. After that, we move on to methodology, where I will go over critical research paradigm and design science. Then follows the methods used to develop and evaluate the artifact before we end with a discussion and a concluding chapter with reflections on future work.

Chapter 2

Background

This chapter will present the thesis's background theory and technologies, focusing on the SOLID platform and shape technologies: SHACL [1] and ShEx [2] . There will also be sections devoted to discussing alternative technologies and the usage of SOLID, ethics, and the problem space. First, we shall look at the SOLID specification and what specific parts of it will be relevant to this thesis.

2.1 The World Wide Web

The World Wide Web was initially intended as a document management system but turned out to become so much more; it was created by English computer scientist Sir Tim Berners-Lee at CERN. A workable system with the World Wide Web browser and an HTTP server was implemented by the end of 1990. Since then, as previously discussed 1, the internet has expanded, and according to *Statista Research Department* (2022), there are now 5.03 Billion users on the internet; with more users come more websites.

Looking at the numbers, however, we see a trending down of sites per user, looking at *RealTimeStatisticsProject* (2022) and their site *internetlivestats* [3] we can see an apparent rise in both users and sites but not in the sites per user.

On the modern internet many of the common sites from the older internet, such as forums, blogs, and personal websites, all seem to have been replaced by a few big sites. The internet is bigger than ever, but has fewer sites per user; this centralization of users, linked with the common usage of OAuth [4], using one site to give information to another without using passwords. This leads us to SOLID, as SOLID would be an alternative to using OAuth from the big websites.

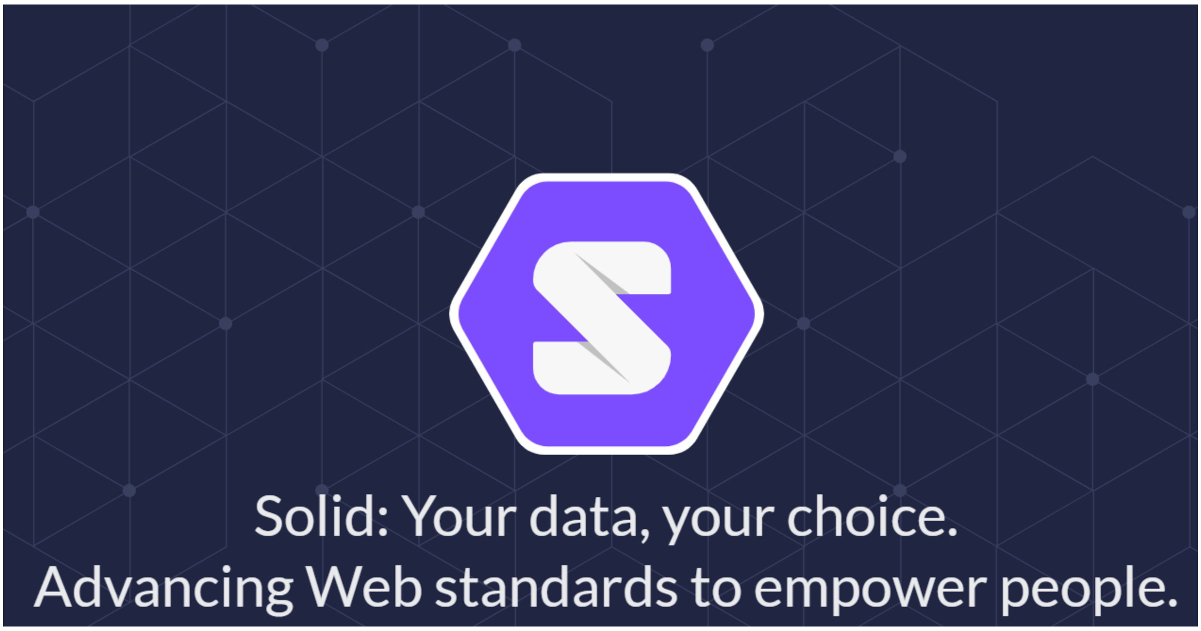


Figure 2.1: SOLID project homepage

2.2 SOLID

The SOLID specification [5] is a part of the project that intends to assist in the decentralizing of the internet. SOLID is interested in shifting control of the data from companies and external entities back to the users. On their *about page*, they describe SOLID as:

> SOLID is a specification that lets people store their data securely in decentralized data stores called Pods. Pods are like secure personal web servers for your data.
>
> Any kind of information can be stored in a SOLID Pod.
>
> You control access to the data in your Pod. You decide what data to share and with whom (be it individuals, organizations, and/or applications). Furthermore, you can revoke access at any time.
>
> To store and access data in your Pod, applications use standard, open, and interoperable data formats and protocols. *The_Solid_Team* (2021b)

From this description, we can gather what is at the core tenants of SOLID: control of data. Who controls the data now, and who they think should control the data? The way they propose to change this is to decentralize data into pods that users control themselves. The users can then access the Internet services they will want to use with said pod, to find out how this will work, first, then we must investigate what a pod is.

[1] Shapes Constraint Language
[2] Shape expressions
[3] https://www.internetlivestats.com/
[4] Open Authorization
[5] A specification is a technical document with the requirements for the design of a product or service

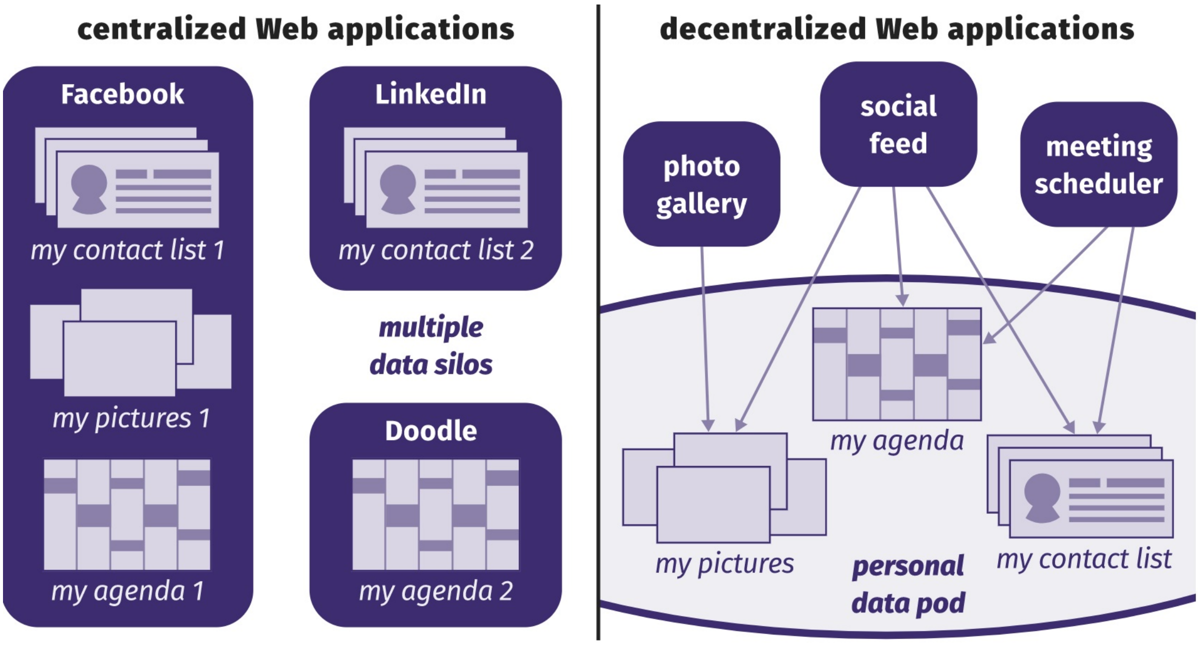


Figure 2.2: The SOLID pods diagram gkontos (2021)

2.2.1 Pods

A SOLID pod, or Personal Online Datastore, is where users would store their data, either in their own self-hosted pod or using a service for their pod needs. In Figure 2.2 we can see what data storage looks like in a centralized web application compared to what it could look like using a SOLID pod. You can then store any data in your SOLID pod, the structural presets that exist in other storage technologies, like limitations to file types and data structure, do not exist in SOLID pods, where the way data is stored has no restrictions. The option of storing all kinds of data in your pod is part of the interoperability goal of the SOLID project. Interoperability is the idea that multiple applications or services could use the same data from your pod, enabling you to only store data once and not multiple times in your pod with the same results.

What is the best way to store a specific type of data, is there a 'natural' way to structure data? How would we decide on what structure a collection of pictures will have, do we base the structure on groupings of dates, folders, or type? What other data is required and possible to add about each picture in the grouping, is there data about the grouping? How would a site you access with your pod know what form to expect your data from? And let's say one site is where you save the pictures to albums in your pod; now, how will other sites know the structure of the data they use? These are some of the problems that we will have to solve if we want the data we store in SOLID pods to be interoperable. From your POD you can decide which services and applications can access your data, and you can revoke that access or change what type of access (read, write) any of the applications or services have.

2.2.2 Pod servers

SOLID servers are where the SOLID pods are hosted, you can use one of several pod providers to host your pod or run a pod server yourself. The SOLID pod server handles the storage, retrieval, and access control of data stored in SOLID pods. The SOLID pod server acts as the intermediary between users and their data. It receives requests from users or applications to perform operations on the data stored in a pod, such as reading, writing, updating, or deleting information. The server is responsible for authenticating and authorizing users, enforcing access control policies, and ensuring the security and privacy of the stored data. Users would control the access to their data in their pod, even if they use a pod provider, The pod server handles data according to the user's settings and authenticating.

2.2.3 BBC Together

One project I want to remark on is currently underway: *BBC Together* is a project by the BBC to create a way to create watch parties using SOLID as the backend. The test project has been concluded, the pilot of the project ended on the 24th of April 2023, and the project seems to have been successful *Atherton* (2023). The future holds more for BBC and Inrupt as their partnership continues. But as *Grace Atherton* discusses in her blog post about the subject, there are some problems:

> However, there are some immediate challenges to using the SOLID pod as a zero-party browsing and identity data source. The most obvious is that advertisers and publishers may be able to keep a copy of a user's data even if the user revokes access. Another more systemic issue is the existing identity, demographic, and psychographic data that is legally collected and sold by third parties, such as credit bureaus. This type of data would not go away even if every internet user suddenly started using SOLID pods. And finally, decentralizing data storage into pods also means that the security of the pod rests in the hands of the individual user. *Atherton* (2023)

2.3 Problem space

Interoperability is a pragmatic goal for any data storage solution, the reality of data interoperability and re-usability is that it is a difficult and complicated business. We face several problems when discussing data reusability and interoperability, how do we know what data is stored where, and how do we know what format this data is stored in, in which structural hierarchy the data is stored? In traditional structures, a site's data is stored on the site or service and obfuscated from the user, if not hidden from them intentionally. However, the site knows how the data is stored, and what is stored there. When, if ever, sites and services offer interoperability, it is in a limited fashion and the data is often obfuscated from the users even if it is disclosed. An example of this would be how Facebook engages with third-party sites with their data using OAuth. As one is accessing the third-party site, one is offered a log-on option from Facebook, using Facebook's data and the OAuth API the site can collect the data they require, with your authentication and authorization. Facebook will then ask you if you want to share the requested data with the third party site, but it does not actually show what data is being shared, it might say "contact information" but what does that include? What information about your contacts is being shared with this third-party site, and what is the third party going to do with this information? From Facebook's settings, you can see what sites you have shared information with, but you only see a general idea of what has been shared such as "friends", or "pictures". In the next section, we will be introducing shapes as a concept as part of the solution to data interoperability and re-usability obstacles.

2.4 Alternatives and options

Traditional storage solutions have remained centralized due to reasons such as cost, speed, accessibility, and reliability. Furthermore, the lack of necessity for decentralization has contributed to the absence of proposed solutions. The lack of necessity is twofold, one part of it stems from corporations not desiring it due to loss of direct control and monetization. Secondly, before decentralization was the proposed solution to privacy concerns, the individual user had little interest in attaining decentralized storage. Blockchain technology has garnered significant attention since its introduction in 2008, with claims of resolving numerous issues and enabling various applications, many of these envisioned applications have yet to materialize. However, the vastness of the blockchain technology topic necessitates a brief exploration of the topic. Notably, storage space poses challenges when employing blockchain technology, as highlighted in a paper *Interoperability and Synchronization Management of Blockchain-Based Decentralized e-Health Systems* (*Biswas et al.*, 2020, 1364). The limited storage capacity for each block in the blockchain necessitates off-chain storage as a workaround. Although there may be additional concerns regarding blockchain as a decentralized storage solution, delving deeper into this topic would require a separate thesis. Consequently, the discussion will shift from blockchain and briefly touch upon interoperability.

> Interoperability is the ability of two or more software components to cooperate despite differences in language, interface, and execution platform.
> *Wegner* (1996)

When we talk of interoperability in computer science, usually we are talking about interoperability between systems, implied in that is that these systems have a centralized storage solution. Peter Wegner's definition confers the idea that interoperability is systems trying to interact with another system (system-to-system interoperability), not data built to be decentralized which offers interoperability to systems (data interoperability). The interoperability discussed in this thesis is the interoperability of data for different systems (data interoperability), not needing to store the same data many times.

Marcia Lei Zeng, a professor of Information Science at Kent State University, provides valuable insights into the landscape of interoperability, and in her article from 2019, titled *Interoperability Zeng* (2019), she details the landscape around interoperability with a focus on semantic interoperability. In this article, Zeng discusses the

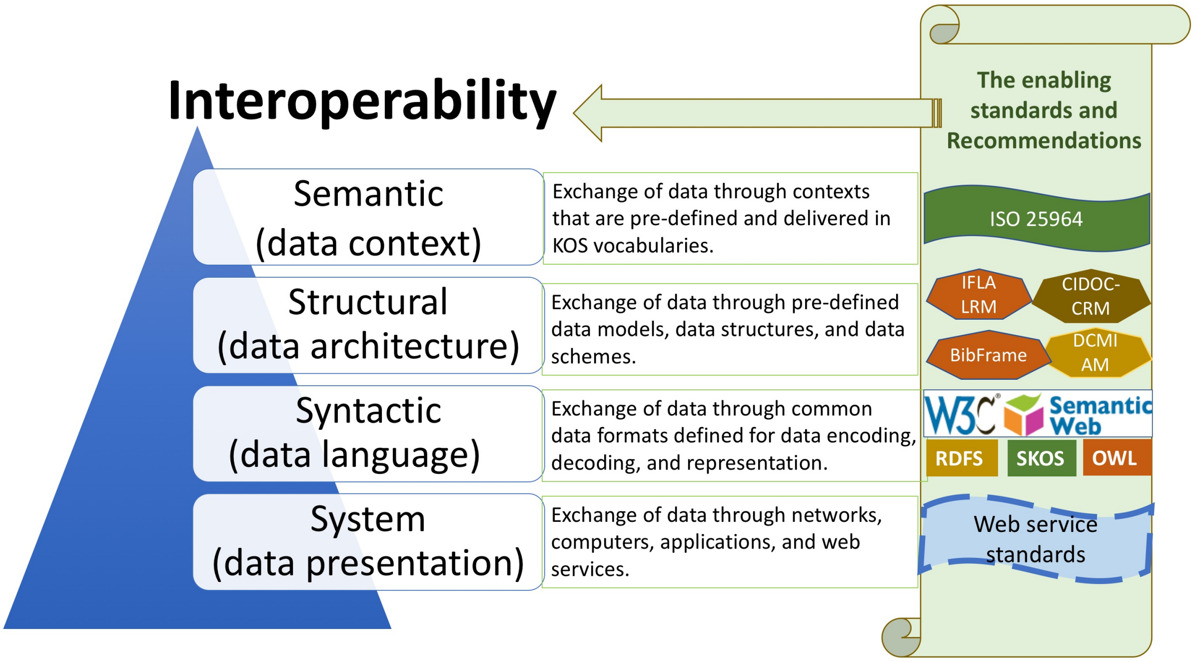


Figure 2.3: Standards and recommendations addressing interoperability issues Zeng (2019)

challenges and current systems in use to facilitate interoperability, though these systems are focused on system-to-system interoperability. In figure 2.3, Zeng goes through recommendations, standards, and obstacles related to interoperability. Data shapes as technology is meant to facilitate functionality across these levels, using existing recommendations and functioning across these. The existing technology is not built to handle this kind of interoperability, to be able to support this kind of interoperability we would need a different solution, shapes technology is intended to be such a solution. Now that we have briefly examined the technological landscape, we will continue with the ethics surrounding this thesis.

2.5 Ethics

This section will discuss some of the ethical hurdles we face with personal information; how can we discuss personal information without discussing the ethical implications that loom over the topic? There are two significant entities one must consider above others when discussing the active collection of personal information. First, there are big technology corporations worldwide, and social media corporations are leading the field among them. Secondly, there are nation-states, authoritarian or democratic, all have a vested interest in collecting and aggregating data from all users around the world. How much of our private information is actually private, and how much control do we have over who can access our information? From the revelations brought forth by Edward Snowden in the Guardian *Greenwald and MacAskill* (2022) we saw what many had speculated to be true; the massive collection of data by the NSA unbeknownst to all but them. The NSA's informational net extended long into the social media sphere, including a willingness to investigate all contacts and family members of the persons of interest they collect data from. The Snowden revelations blew up internationally and opened the eyes of many to the dangers of trusting social media with personal information; others took the position that there was no way to keep one's personal information private, so why try? This argument falls into the same argumentative fallacy as "I have nothing to hide, so why should I want my information to stay private?", there are many refutations of this argument; Snowden rebutted it with this in a Reddit AMA in 2016:

> Arguing that you don't care about the right to privacy because you have nothing to hide is no different than saying you don't care about free speech because you have nothing to say. *snowden* (2016)

Maintaining privacy and keeping our personal information safe is more important than one might want to admit. The average user lacks the understanding of the scope of the surveillance they are under; also, the difficulty is in having the digital know-how to find and utilize the tools necessary to protect their information, if that is genuinely achievable. One cannot use the internet to its full extent and keep one's information private. One could propose that; we own our personal information, but we don't control it in any meaningful way anymore; it has become part of a system that implicitly expects you to give up your personal information on entry. Why is it so important to discuss this when developing and researching shape technology? What happens in the world around us affects us, and we must include a thought for the ethical framing that our research is influenced by. We would not be talking about returning control over our personal information if that had not been lost to us in the first place.

For several reasons, the power relations between the user and the global social media corporations should be a factor in this type of academic text. Individuals cannot influence the development of technology alone; in this sense, we participate in systems with no real power to change them. The digital sphere has permeated the real world to the extent that you cannot live without interacting with digital systems in your daily life. Not participating means exclusion from society, and carries negative consequences if attempted. This is especially apparent in Norway; there have been several cases of older Norwegians being stuck in digital limbo *Tønset* (2023). We should not underestimate the extent of the mismatch in power and influence on the topic of personal information, individual vs corporations, and even individuals vs society. The effect of the power imbalance is an environment where corporations are free to create systems designed to extract information from the user while presenting themselves as tools.

The implications of this mismatch in power are far-reaching, including the exploitation of personal information, loss of control and autonomy, and the perpetuation of inequality and discrimination. Keeping personal information safe is essential to protect privacy, prevent surveillance, and maintain personal control. The significance of discussing ethics in developing and researching technologies like shape technology lies in acknowledging the loss of control over personal information and the need to reclaim it. The power dynamics between users and global social media corporations should not be ignored, as individuals alone cannot influence technological development. However, participation in digital systems is essential for societal inclusion, recognizing the extent of the power imbalance is crucial as it allows corporations to extract user information while providing helpful tools. By incorporating ethical considerations, we can create a more equitable and user-centred digital environment that respects privacy, empowers individuals, and promotes social well-being.

Protecting user rights, accountability, transparency, protection from discrimination, and equal opportunity, there are many reasons for discussing ethics in the information science sphere. Holding corporations accountable is something that has been an issue since the creation of the Internet, the creation of new laws and regulations takes a long time and the Internet develops fast. There has not been anything that has spread as fast as the Internet or some of the services on it before. This is something Shoshana Zuboff remarks on in her book *The Age of Surveillance Capitalism Zuboff* (2019), the lack of laws is one of the critical success factors for corporations utilizing the Internet. This is one of a plethora of reasons she cites for the success of these corporations, which work very hard to collect as much data as possible.

> This kind of lawlessness has been a critical success factor in the short history of surveillance capitalism. Schmidt, Brin, and Page have ardently defended their right to freedom from law even as Google grew to become what is arguably the world's most powerful corporation. Their efforts have been marked by a few consistent themes: that technology companies such as Google move faster than the state's ability to understand or follow, that any attempts to intervene or constrain are therefore fated to be ill-conceived and stupid, that regulation is always a negative force that impedes innovation and progress, and that lawlessness is the necessary context for "technological innovation." (*Zuboff*, 2019, Chapter 4, II)

Going by what is legal has always been a strenuous approach to what is ethical, and more so now that the development of new technologies comes faster and faster, and adoption speed increases, waiting for a legal declaration might be ill-advised. A critical look at why we need to solve problems is one of the ways we can develop things that are not just useful or effective, but also good, or maybe failing that; not evil. Preservation of choice is imperative, currently not implemented when giving up our information happens upon entry, which has become the baseline for interactions online.

What are the ramifications of this system, what do we risk as users, and what are the real dangers? To the individual, the perceived risk is small, which is part of the problem, as individuals we stand the risk of having advertisements targeting us with higher accuracy this is the one that most are aware of and many disregard as the cost of being online. We do risk the breach of our personal information, databases are hacked on a daily basis, and credit card information, names, birthdays, emails, and addresses, are all at risk in these data breaches. Unencrypted passwords in databases have led to many problems for users, especially those who use the same password on multiple sites. Groups who are discriminated against, and those in opposition of tyrannical or totalitarian governments have much to fear from data collection and must go far to protect themselves. Corporations and governments both stand to gain much from the increase of surveillance and the user stands to gain little. What can we do against such forces of imbalanced proportions? Zuboff advises friction:

> Friction, courage, and bearings are the resources we require to begin the shared work of synthetic declarations that claim the digital future as a human place, demand that digital capitalism operate as an inclusive force

bound to the people it must serve, and defend the division of learning in society as a source of genuine democratic renewal. (*Zuboff*, 2019, Chapter 18, VI.)

In this section, we have looked at the ethical perspective of this thesis, and the relevant ethical landscape. In the next chapter, we will discuss the critical theory paradigm because of the need for technological solutions to these issues.

2.6 Tools

This section will detail the tools and systems used in this project. This is nonetheless meant to be a quick overview for readers to familiarize themselves with the tools and systems.

2.6.1 RDF

RDF or *Resource Description Framework* is a framework developed by W3C to be used as a standard for metadata. RDF is a way to express information in semantic triplets, a triplet is a way to express information as a graph built with triplets composed of three elements: subject, predicate, and object. Looking at figure 2.4 we can see a graphical representation of the relation between the three components of a semantic triplet and an example of what such a triplet could consist of.

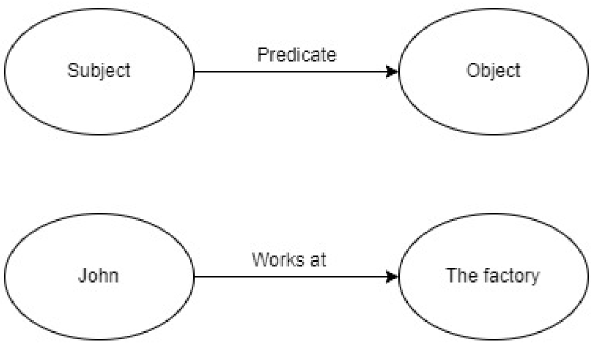


Figure 2.4: Diagram showing the relation between subject, predicate, and object Ltd (2011)

With these data triplets, we can construct graphs with meaning, not just for computers, but also readable for humans. Multiple RDF triplets are called a graph (*Hogan*, 2020, p.75). One can add nodes to any of the edges of a triplet to add to the graph, as long as these nodes follow the restrictions set by the existing triplets. Making RDF human-readable is one of the aims of languages like Turtle [6], RDF triplets can be hard to get the hang of reading for us humans, and Turtle aims to make the triplets easier to understand, here in figure 2.5 you can see an example from the Turtle primer, compare the RDF graph to the same information in the Turtle syntax:

```
@prefix RDF: <http://www.w3.org/1999/02/22-RDF-syntax-ns#>.
@prefix contact: <http://www.w3.org/2000/10/swap/pim/contact#>.
```

[6]Terse RDF Triple Language (Turtle).

Figure 2.5: This is an RDF Graph Describing Eric Miller Manola et al. (2022)

```
<http://www.w3.org/People/EM/contact#me>
  RDF:type contact:Person;
  contact:fullName "Eric Miller";
  contact:mailbox <mailto:em@w3.org>;
  contact:personalTitle "Dr.".
```

Manola et al. (2022)

The Turtle syntax does a fine job of structuring RDF data in a fashion that enables humans to read and comprehend it in an easy way. More Turtle syntax will be utilized in the SHACL examples in the development 2 section. In the discussion section of the concluding chapter on RDF Hogan reflects on the need for what he calls a web of data, with some of his examples of personas discussing RDF:

> Julie: So to create the Web of Data we need RDF?
> Aidan: Well we need something like RDF: a core data model that is standardised, agreed upon, and flexible for many use-cases, with the ability to use globally-unique Web identifiers. RDF has all the fundamental ingredients we need. (*Hogan*, 2020, p.122)

Hogan discusses various options for developing a flexible and versatile core data model, and one of the potential solutions he presents is SHACL. In the upcoming section, we will delve into both SHACL and ShEx as we concentrate on shape expression languages.

2.7 Shapes

Imagine you have many gingerbread cookies (data) with different shapes and a cookie-cutter (shape) with a star shape. The cookie-cutter star is our shape, and we can test if the gingerbread cookies have the desired shape. The gingerbread cookies are stored in a box, this box would be the SOLID pod.

We can trace shapes back to the early 2010s when many solutions to RDF validation issues were proposed, In a 2013 meeting of the RDF validation workshop *DavidBooth et al.* (2013) there was a broad discussion of what the future of RDF validation would be, and what kind of solutions they required.

In this workshop, they discussed several shapes technologies; "Resource Shapes" by IBM as part of the *Open Services for Lifecycle Collaboration* (OSLC) initiative *for Lifecycle Collaboration* (2022) among others. Presented in the *OSLC Resource Shape: A Linked Data Constraint Language Ryman et al.* (2013) they suggest that their shapes system could be used for specifying constraints on RDF data. They suggest that a shape could be used for checking if data had a certain structure and report back if there are any errors with the query. At the workshop, or rather next door there was a Semantic Web Gathering where Eric Prud'hommeaux was having a presentation on his ShEx project, so shapes were definitely a widely discussed topic at the time. Concluding the workshop was an agreement that they needed a standard for validation of RDF data, in addition, they stated that:

> In addition to being able to validate data, the workshop revealed the need for being able to communicate the constraints against which data is to be validated in a way which is both easy to understand by human beings and discoverable by programs. *DavidBooth et al.* (2013)

The workshop concluded that there is a need in the industry for a way to deal with validation, they also discussed that the word *validation* does not adequately describe the whole scope of the issue. There is a need for functionality beyond what validation of data provides, for example, other functionality I.E degrees of violation and form creation. The participants decided that the W3C should start an effort to create a description of the "shape" of the RDF graphs that a service produces or consumes; a "shape" that is supposed to be both machine- and human-readable. this is what prompted the creation of the Data Shapes Working Group. The group made efforts to explore the options for validation and shapes technology, and in 2017 they selected SHACL to be the W3C recommendation.

> "Shape Expressions associate RDF graphs with labeled patterns called "shapes". Shapes can be used for validation, documentation and transformation of RDF data." *W3C* (2022)

Why is data shape validation something we should discuss or care about at all? There are many systems for data validation and different ways to make sure that a system can use the data users have stored. The next section will introduce the two shape constraint languages central to this thesis.

2.8 SHACL and ShEx

Both SHACL (Shapes Constraint Language) and ShEx (Shape Expressions) are languages used for expressing shapes, but they have distinct characteristics and applications. In this section, we will explore SHACL and ShEx individually before highlighting their differences and discussing why SHACL was chosen over ShEx for this project.

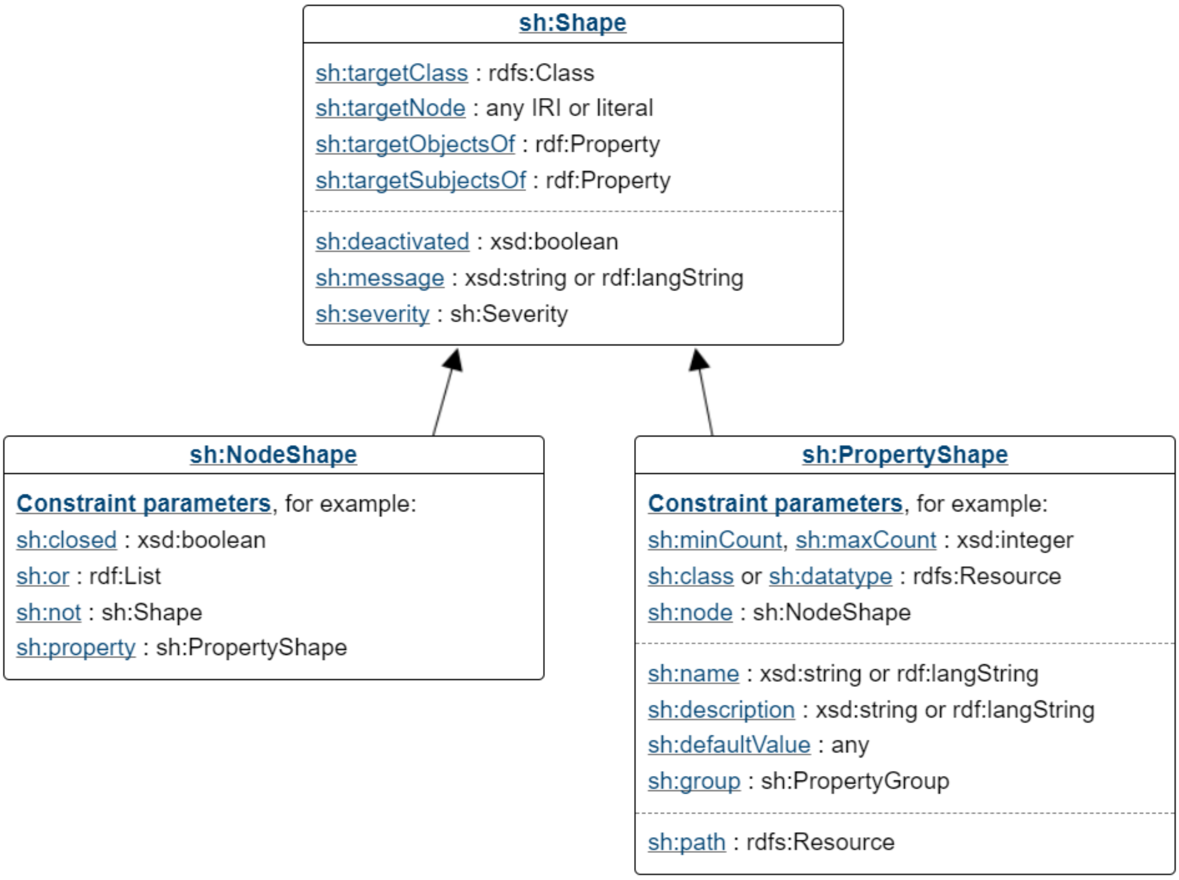


Figure 2.6: This is a diagram showing some of the key classes in the SHACL vocabulary taken from the SHACL specification document W3C (2022)

2.8.1 SHACL

SHACL is a W3C recommendation, and as of writing the latest update to the document was 20 July 2017, but the editorial document was updated in January 2021. SHACL is structured in RDF, and SHACL files are Turtle files (*ttl* files). Having SHACL files in Turtle, as described earlier, this type of formatting is much easier to read for humans and makes SHACL more accessible. SHACL is divided into SHACL CORE and SHACL-SPARQL [7], in this thesis we will discuss and use SHACL CORE; SHACL-SPARQL gives ways to represent constraints more flexibly and expand on the Core vocabulary but does not lie within the scope of this thesis, and as the project carried on, there was no need to venture into the space of SHACL-SPARQL.

In figure 2.6 we can see some of the key classes in the SHACL vocabulary [8] and in figure 2.7 we can see what different parts make up a shape. Here I will describe some of the restraints in SHACL.

[7] SPARQL Protocol and RDF Query Language (recursive acronym)
[8] the arrows denote rdfs:subClassOf triples.

2.9 Target Nodes

In a SHACL shape we declare the target nodes for the shape as seen in figure 2.6. The targets can be any of the following types:

> SHACL Core includes the following kinds of targets: node targets, class-based targets (including implicit class-based targets), subjects-of targets, and objects-of targets. (*W3C*, 2022, 2.1.3)

MinCount and MaxCount Constraints

The MinCount and MaxCount Constraints are used to define minimum and maximum occurrence constraints for properties. The MinCount constraint specifies the minimum number of occurrences expected for a property, while the MaxCount constraint defines the maximum number of occurrences allowed. These constraints enable the specification of cardinality requirements for properties, ensuring that data conforms to predefined expectations (*W3C*, 2022, 4.2).

Value Constraints

Value Constraints are used to define specific requirements or restrictions on the values of properties. SHACL provides various types of value constraints, such as DataType Constraint, NodeKind Constraint, and Enum Constraint, among others. These constraints allow developers to validate the data type, node kind, or specific values of properties, ensuring that the values adhere to the specified constraints (*W3C*, 2022, 4.1).

Property Constraints

Property Constraints enable the specification of constraints on individual properties within a shape. This includes defining constraints such as allowed value ranges, regular expression patterns, uniqueness requirements, or referencing other shapes for validation. Property Constraints provide flexibility in shaping and validating individual properties based on specific criteria (*W3C*, 2022, 4.5).

Property Paths

SHACL supports Property Paths, which are used to define constraints based on the relationships between properties in a graph. Property Paths enable the expression of complex constraints by traversing the graph structure and applying constraints based on the path taken. This allows for sophisticated validations that depend on the relationships and connectivity of properties in the data (*W3C*, 2022, 2.3.1).

Logical contains components

OR

> sh:or specifies the condition that each value node conforms to at least one of the provided shapes. This is comparable to disjunction and the logical "or" operator. (*W3C*, 2022, 4.6.3)

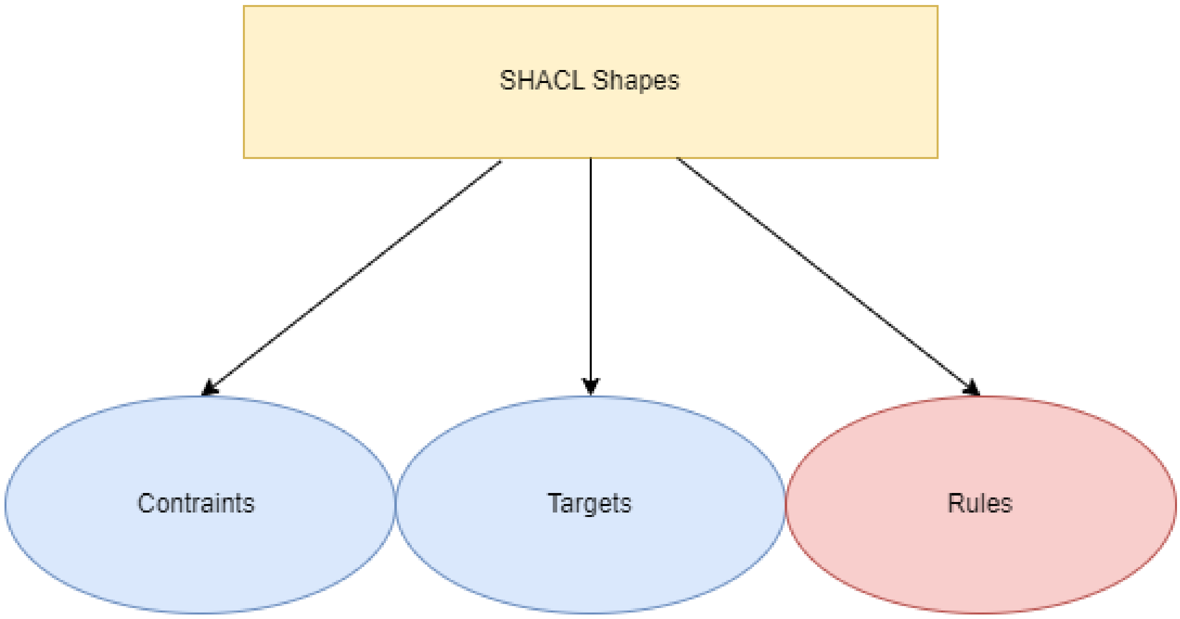


Figure 2.7: This is a diagram showing what parts a SHACL shape consists of

sh:or is essential if we are to be able to use different naming practices when designing our websites and services. Its function is rather simple, but its use might be complex, we can declare that one shape or another is required. It is important to note that this does not function if the shape is closed, hence we need an open shape environment for real-world usage of sh:or. The sh:or constraint only works between different shapes, not between class or properties, this is important to note, as one would need several similar shapes with target nodes and classes of their own. This does allow for the use of different naming conventions, and even structure, as long as the targeting of these is accounted for in the structure of the site.

Closed Constraint

The Closed Constraint is a vital component in SHACL that ensures that only properties defined in the shape are allowed in the data. It prevents the presence of additional, undefined properties. This constraint promotes data integrity by enforcing a strict schema that restricts the usage of properties to those explicitly specified in the shape (*W3C*, 2022, 4.8.1).

> The SHACL Core language includes a property called sh:closed that can be used to specify the condition that each value node has values only for those properties that have been explicitly enumerated via the property shapes specified for the shape via sh:property. (*W3C*, 2022, 4.8.1)

Validation

In Figure 2.8 we can see how a validation process with SHACL works, the process shown is what happens when you try to validate a graph or parts of it with a shape. One thing to remark is that a successful validation grants no rapport other than a Boolean *true*, no rapport on which nodes were validated or what shapes they have.

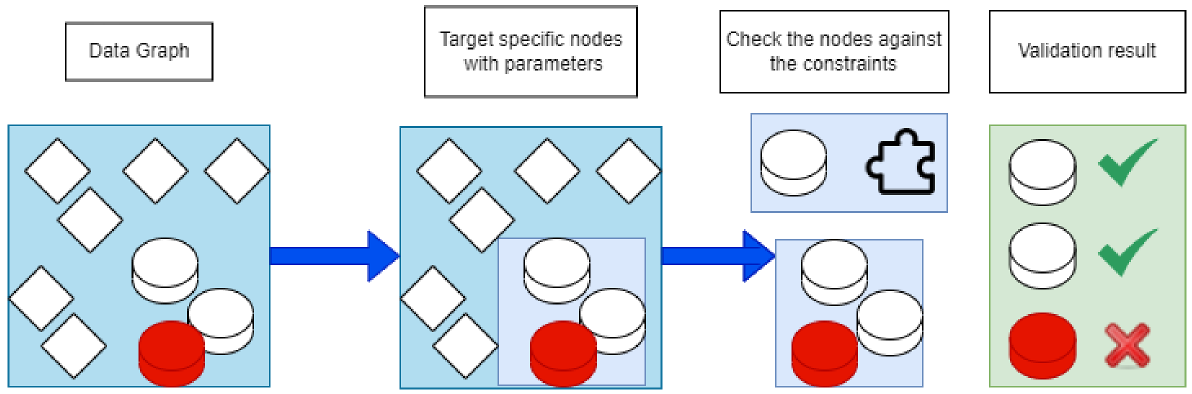


Figure 2.8: This is a diagram showing what a validation process would look like with SHACL (maybe mixing shapes and colors is a bad idea to visually present this?)

It could be useful to get a warning instead of a failed validation, here we can use the "sh:resultSeverity" to assign either an "Info" or a "warning", as a warning will not result in a failed validation, just as the name implies, a warning, and "Info" indicates additional information [9].

After this exploration of the SHACL specifications, the next section will be devoted to ShEx. We are going to discuss the differences and similarities between these languages.

2.9.1 ShEx

ShEx, or shape expressions [10], is another shape expression language, similar to SHACL, developed in the same time period. However, ShEx has some key differences from SHACL, ShEx is a W3C community group effort and was derived from the same need for data validation with additional functionality, as SHACL. ShEx is schema-based, unlike SHACL where there are target nodes, in ShEx there is instead a *Shape map* used to select the individuals, as shown in figure 2.9.

In figure 2.9 we can see the ShEx validation process with all its components, we have the RDF data, the shape map, and the schema. First, a ShEx schema specifies the requirements that RDF data graphs must satisfy in order to qualify as "conformant", including the combinations of subjects, predicates, and objects that may occur in a given graph as well as the cardinalities and datatypes that they may use. The ShEx model compares an RDF graph to the ShEx schema and produces a validation result that indicates any portions of the data that do not match. The shape map specifies the target nodes or classes that are to be checked against the ShEx schema. The result of the validation is also a shape map, but now with the result of the validation. In figure 2.10 we can see that the shape map is actually a query shape map and a fixed shape map, this distinction relieves some of the confusion about what part of the process we are discussing. The ShEx language is designed to be used for a variety of tasks, including verifying RDF data, transferring interface parameters and data structures, creating user interfaces, and converting RDF graphs into different data formats and structures.

[9] the standard is "violation" which results in a failed validation
[10] https://shex.io/

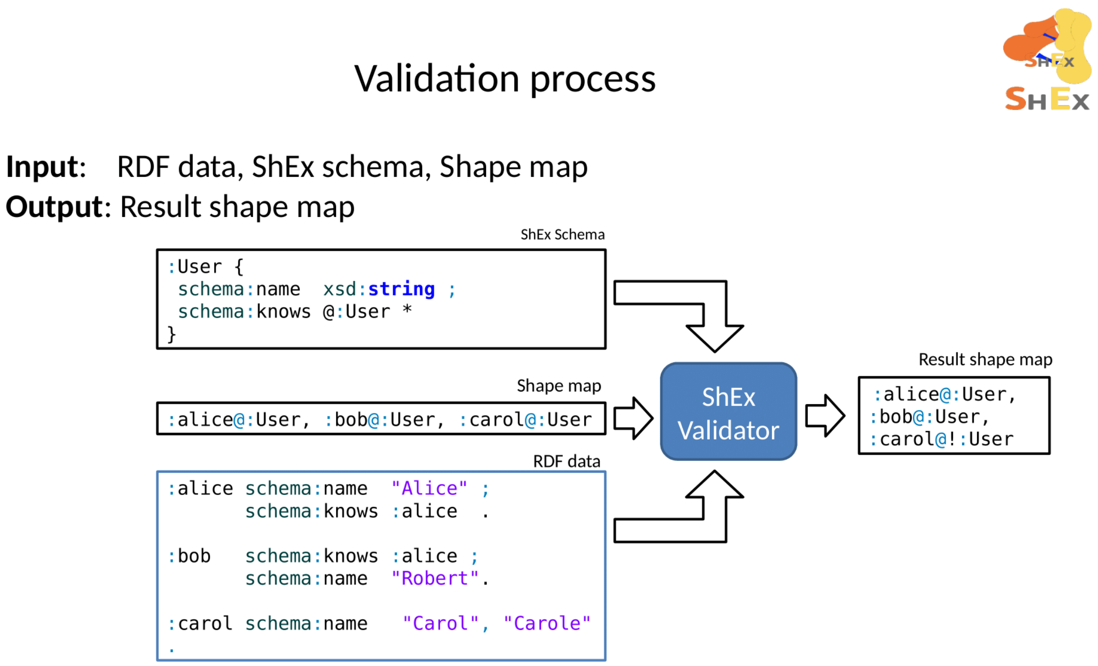


Figure 2.9: This is a diagram showing what a validation process would look like with ShEx taken from a presentation on ShEx in 2020 (Gayo, 2020, Slide 9)

The intention of a ShEx schema is to generate user interfaces, communicate interface parameters and data structures, validate instance data, and translate RDF graphs into different data formats and structures.

In ShEx (Shape Expressions), there are several important components and features that allow for the definition and validation of shape constraints on graph patterns. Let's explore some of the key aspects of ShEx:

Cardinality Constraints

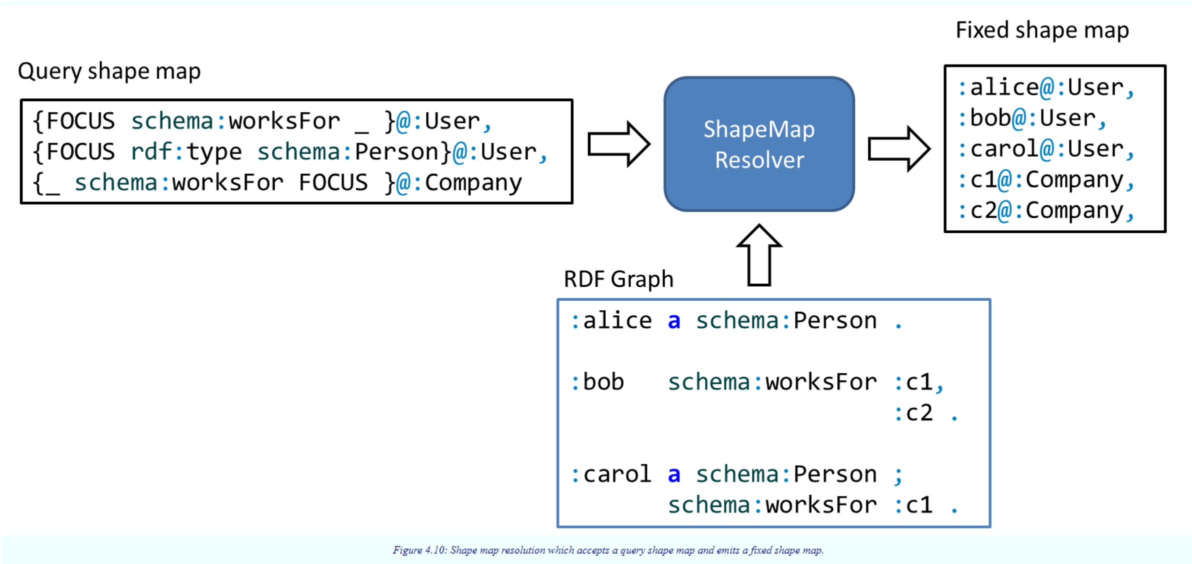
ShEx allows the definition of cardinality constraints to specify the expected number of occurrences of a property or a triple pattern within a shape. These constraints ensure that a particular property or pattern appears a minimum or maximum number of times in the data, enabling the enforcement of cardinality rules. (*GROUP*, 2019)

Logical Operators

ShEx provides logical operators such as conjunction (AND), disjunction (OR), and negation (NOT), allowing for the combination and negation of shape expressions. These operators facilitate the construction of complex constraints by combining multiple shape expressions and specifying the relationships between them (*GROUP*, 2019, 5.3).

Value Constraints

ShEx allows for the specification of value constraints on properties within shape expressions. These constraints enable validation based on specific values, data types,



S

Figure 2.10: Shape map and resolver figure (Labra Gayo et al., 2017, 4.9.2)

ranges, or regular expression patterns. Value constraints offer flexibility in defining constraints on property values in the graph (*GROUP*, 2019, 5.4.6).

Closed Shape Constraint

Similar to SHACL, ShEx also supports the concept of a Closed Shape Constraint. This constraint ensures that only properties explicitly defined within a shape expression are allowed in the data. It prevents the presence of additional, undefined properties, promoting strict adherence to the defined shape. In the same vein as for SHACL, this is useful for testing, or if we were to develop many small shapes to build a greater whole from, thus needing them to be very specific and closed (*GROUP*, 2019, 5.5.2).

These are some of the significant components and features in ShEx, By utilizing Shape Expressions, Triple Constraints, Cardinality Constraints, logical operators, and other features, developers can define rich and expressive constraints for validating RDF data. ShEx provides a powerful framework for schema validation, data transformation, and documenting graph patterns in diverse application domains. Now we will move on to discussing SHACL vs ShEx.

2.10 SHACL vs ShEx

In this section, we will discuss briefly the main differences between these two shape expression languages, and why the choice fell on SHACL rather than ShEx. There are many similarities and subtle differences between these two laguages. Choosing which one to test and work with proved somewhat challenging for a novice in this particular field. SHACL has SHACL-SPARQL in addition to the SHACL-CORE, which makes it a natural choice for projects involving SPARQL.

Outside of SPARQL-specific cases, the decision between SHACL and ShEx depends on whether form generation is required. Form generation refers to the ability to dynamically generate forms based on missing data or existing data that aligns with the defined ShEx shape. While form generation is an intriguing feature, it was not a crucial requirement for this particular project, and therefore, it did not strongly favour ShEx over SHACL.

SHACL and ShEx, despite addressing similar issues related to schema and constraints, differ in their fundamental approaches. This core difference in approach to shapes and validation can be seen in the ways the languages are constructed and what functionality they support and include. ShEx was created with the intention of serving as a kind of RDF graph grammar or schema. This architecture was influenced by Yacc, RelaxNG, and XML Schema (*Labra Gayo et al.*, 2017, chapter.7.3), among other languages. Its primary objective was to define and characterize RDF graph structures for subsequent validation. Conversely, SHACL's creators aimed to provide RDF with a constraint language. The primary objective of SHACL is to confirm that a given RDF graph complies with a set of constraints. In this way, SHACL shares similarities with the Schematron approach, establishing requirements for RDF graphs. Similar to how Schematron is heavily dependent on XPath, SHACL is heavily dependent on SPARQL (*Labra Gayo et al.*, 2017, chapter.7.3). These technical differences between SHACL and ShEx significantly contributed to the time-consuming process of selecting a framework to work with. In hindsight just picking one at random might have allowed for the time to test the use of both languages. We shall return to the differences between SHACL and ShEx in both the discussion and the development chapters. However, there were several reason for prioritizing SHACL over ShEx, Ultimately, the key reason for choosing SHACL over ShEx was its recognition as a W3C recommendation, and since the work would use SOLID, and as we have discussed the SOLID project is closely tied with members of W3C, and had a community group there, it seemed appropriate to elect to use this system.

2.10.1 Protégé

Protégé is an open-source and free software, used for creating systems and managing ontologies and terminologies. Protégé has a wide option for data types, it also includes support for many plugins to expand upon the selection. Protégé has a SHACL plugin called 'SHACL4Protege'. Using this allows us to test developed shapes against entities created in Protégé. *Musen* (June 2015)

2.10.2 Shape trees

Shape trees are central to shape technology, it is important to mention when discussing the future of shapes on the Internet. For shape technology to be accessible there is a great need for tools that allow for ease of use, shape trees are one of those technologies. Shape trees were released as an editors draft on the 3rd of December 2021 *Prud'hommeaux and Bingham* (2021a), and the editor's draft was published earlier that same year on the 16Th of April 2021 *Prud'hommeaux and Bingham* (2021b). This is new technology, not fully developed, but potentially a very useful addition to shapes to be able to expand their accessibility.

> Shape trees marry RDF vocabularies, shapes, and resources into "little trees" that provide machine to machine interoperability, combining them

> into concepts that humans can easily comprehend, such as medical records, notes, notebooks, calendars, and financial records. For applications that operate on more complex and interconnected resources, Shape Trees express the layout of those resources and associate them with their respective shapes.
>
> A shape tree is a machine-readable template describing the expected layout of a tree of resources in a container-based ecosystem. A shape tree expresses a tree hierarchy by containing other shape trees. The terms used to express a shape tree are described using an [RDF] vocabulary. (*Prud'hommeaux and Bingham*, 2021a, 2. Shape Tree)

Before, shapes were something that we had outside the data and used to validate, with shape trees we would marry the shape with the data *Prud'hommeaux and Bingham* (2021a). Going back to our gingerbread analogy 2.7, shape trees would be putting the cookie-cutter on the top of the box making it easy to know what is inside without having to look.

2.11 DBpedia

DBpedia (Data base) is a project started to create structured data from the Wikipedia project. The project was started at Leipzig University, they released their first dataset in 2007. DBpedia is maintained by individuals from the University of Mannheim, and Leipzig University. In 2011 DBpedia group created a tool for the automatic annotation of text. The project is intended to look through a text and analyze if any of the words likely are Wikipedia entries, then annotate the text with links to those pages. I will be using Spotlight to make semantic tags based on the text in a website later in this thesis *Team* (2022).

Chapter 3

Methodology

This chapter will introduce the methodology that will be utilized for the thesis, with explanations for why these choices were made, beginning with a look at the critical theory paradigm; following that, we will move into threats to validity and the design science section.

3.1 Methodology

Traditionally one methodology is chosen when doing research, a methodology that will fit the field, the type of research or because of limitations or any other of the myriad of reasons one can have when researching. This thesis will incorporate two methodologies, the reasoning for this is twofold; the two methodologies are used for their different focuses and to fulfil their respective shortcomings with each other's particular strengths.

It was logical to investigate *Design Science* since I would be creating some sort of artifact and design science is known for being particularly useful in information science. Some of the traditional methodologies can be found lacking in some areas when one is developing *"artifacts"* as design science names them. While design science does very well in remedying some of the blind spots of traditional methodology when it comes to artifacts and development of software and more tangible product of information sciences, there are other factors to consider. What came to mind when researching the background material was why do we need this technology. Why did Tim Berners-Lee and the World Wide Web Consortium decide we needed a "*fix*" for the internet? When devising solutions to a problem, diligence should drive one to investigate the cause of the need for an answer, the problem causing the requirement for the fix. These questions called for a critical look at these technologies and what powers influence the sphere in which we work.

The division of this thesis into producing an artifact for evaluation, but also seeing this thesis in the position in society and why we must look at these problems and solutions through the lens of a methodology/Research paradigm, and for that, the choice fell on "*Critical Theory*". Critical theory was the obvious choice for interpreting the implication of using shapes and SOLID pods, because of the implicit power imbalance between the actors involved. Design science theory is a suitable tool for information science and research involving the development of artifacts, though perhaps a different methodology is more suited for the discussion and analysis sections. *"Towards Critical Design Science Research"Iivari and Kuutti* (2017) is the title of Netta Iivari and Kari Kuutti's article published in the 2017 edition of *INTERNATIONAL CONFERENCE ON INFORMATION SYSTEMS (ICIS)*.

Netta Iivari is a professor of information systems at the University of Oulu and director of the INTERACT research unit. She has experience with information systems, human-computer interaction, and cultural anthropology. She earned her master's in cultural anthropology in 2001, built upon that with a master's degree in information processing science in 2003 and earned her doctorate in information systems in 2006. The traditions of interpretive and critical research have a significant impact on her work. She is particularly interested in the creation and application of lenses that are focused on culture and discourse as well as in the analysis and promotion of interdisciplinary research and design.

Kari Kuutti retired at the end of 2016 and is currently working as a professor emeritus. He does his own research, participates in group activities, and still chooses to oversee a few PhD candidates. His field of study, when viewed generally, is human interaction with information technology. The development of theoretical, conceptual, and practical tools for comprehending the connection and its dynamics as well as tools and methods for designing both products and services is the goal. His work is theoretically based on the cultural-historical activity theory (CHAT), which he has applied to issues related to information technology since the 1980s. The development of a perspective on critical design research, which they recently began with Netta Iivari, is one of the most active issues at this time.

These two researchers both have broad experience within information sciences and in 2017 they wrote the aforementioned paper *"Towards Critical Design Science Research"Iivari and Kuutti* (2017) where they present the idea of merging design science research with critical theory into a "Critical design science research" a combination of the two, applying some of the features and strengths of both onto the other.

> We argue for the usefulness of combining the forces of DSR and the critical research tradition. This way, the strengths of DSR – its present-day popularity as well as the concrete yet theory-inspired design outcomes – could be harnessed to serve the goal of the empowerment of the oppressed, which is the main goal of research from the perspective of the critical research tradition (e.g., Denzin & Lincoln 2000; Lincoln & Guba 2000; Myers & Klein 2011). *Iivari and Kuutti* (2017)

There is a power imbalance between the individual user and the corporations in favour of the corporations. If one were to only use DSR when researching a topic like SOLID pods, and data shapes, focusing only on the technology would not lessen the value of the research, although it would be leaving a blindside to why, and why can be as important as how. Perhaps stretching the famous quote from Jeff Goldblum in the 1993 movie classic, *Jurassic Park*:

> Your scientists were so preoccupied with whether they could, they didn't stop to think if they should. *Spielberg* (1993)

Why and if we should, are central topics in research and science, and particular critical theory, focuses on the forces in society and what is needed to practically change the current social reality. Put by Horkheimer's definition:

> ... a critical theory is adequate only if it meets three criteria: it must be explanatory, practical, and normative, all at the same time. That is, it must explain what is wrong with current social reality, identify the actors to change it, and provide both clear norms for criticism and achievable practical goals for social transformation. *Bohman et al.* (2021)

The paper goes on to state as a finishing remark on itself:

> However, this papers reports an academic exercise, and no actual DSR has been carried out. Naturally, this is a limitation of a paper arguing for critical DSR. In the future, critical DSR will definitely be carried out in practice by the authors, and other IS researchers are also warmly welcomed to join in the critical movement and practice. *Iivari and Kuutti* (2017)

3.2 Critical theory

Here we shall discuss why the use of the critical theory was chosen over the use of other valid choices such as positivism and constructivism. First an introduction into what defines the critical research paradigm.

The difference in power between the individual user and corporation has never been greater, at least not when we discuss big entities like Facebook and Google. When selecting what methodology would best fit this thesis, I had just read chapter 5 in *Research Methods Information, Systems, and Contexts* (*Johanson and Williamson*, 2018, chapter 5). One cannot/should not, consider the issues of privacy we face today and not reflect on the power relation between the parties involved. Assuming a positivist stance that:

> "Assumes both natural and social sciences are objective and value-free, operating separately from social and power structures; ideally positivist researchers are detached from the topic studies and collect value-free facts." (*Johanson and Williamson*, 2018, chapter 5 table 5.1)

Which does leave a want for the ethical concerns related to this thesis. The critical paradigm embraces that "Any research is a moral-political and value-based activity..." (*Johanson and Williamson*, 2018, chapter 5 table 5.1).

Effectively discarding the notion that we stand outside of the problem looking in. In fact, we are all part of the Digital world, and cannot discard the ethical issues of this thesis for a "detached from the topic study and collect value-free facts." We must embrace that we have a position on the subject, and argue that the reasoning still stands.

For this thesis ethics is central, our personal information and control of that information cannot be discussed without reflecting on what is, and what should be. The SOLID project is centred on the privacy concerns of the user, and the focus of this thesis is expanding on SOLID and its use for social media. Giving users an option that grants them more control of their own data is empowering them to have more control over their own lives, even if just in a very small way. It is common knowledge that internet companies and other corporations want our data, and many would argue that companies collecting our data is unavoidable. In his book *The Googlization of Everything Vaidhyanathan* (2012) Siva Vaidhyanathan has a critical look at the power Google holds, and how it wields it.

> Opting out of any Google service puts the Web user at a disadvantage in relation to other users.(*Vaidhyanathan*, 2012, pg.90)

One of the points Vaidhyanathan raises is that Google offers excellent services, and they are 'free', but at the cost of the users' data. He points out how the inherent power structure of established companies limits choice and competition. Reflecting on this there are most times options maybe of creating an account on a site or browsing with no account, but these options are limited. On most sites, this is a more difficult process than if you log in with Google or Facebook.

> Critical social science also aims to provide explanation, description and understanding but, unlike positivist and interpretive research, does not consider them sufficient; critical researchers are motivated by a liberating and emancipating purpose and aim at affecting practical affairs, life and working conditions of people. The key distinguishing feature of critical social research is its concern with moral and ethical questions related to (often hidden) forms of domination, control and exploitation through information systems and knowledge management systems (Brooke, 2002; Cecez-Kecmanovic, 2001; Stahl, 2003).(*Johanson and Williamson*, 2018, Chapter 5)

The SOLID project and the efforts of the members involved with increasing the user's control of their data, make it clear that SOLID is a clear candidate: *"critical researchers are motivated by a liberating and emancipating purpose and aim at affecting practical affairs"*.

3.2.1 Threats to validity

> The aim of critical educational research is not merely to explain or understand society but to change it (Patton, 2002). It is critical of both interpretive and positivist approaches to research because they are regarded to be "enmeshed in dominant ideology... neither has an interest in changing the world, and neither has an emancipatory goal" (Scott & Usher, 2000, p.35). (*Abdul Rehman and Alharthi*, 2016, pg.57)

Critical paradigm is vastly different from positivism and interpretivist approaches in that, the desire not just to define or interpret, but to change. To have an opinion is not inherently negative when doing research, we all have biases, it is undeclared biases that interfere with academic research. The biggest threat to the validity of this thesis is undeclared bias on my part, I intend to give enough background and declare my biases as to avoid any serious misstep. I will also dedicate a section to ethics, not just to explore the ethical implications of personal information exploitation, but also to disclose my own biases and how they influence my view of the world.

3.3 Design Science

The process of researching through design, arguably, has been done since we started developing science as a field, Richard Fuller an architect and systems theorist formalised design science in 1957 *Fuller* (1957) before this any science that was done with the same type of artifact making to design evaluation process could be said to have employed the same type of thinking. After Fuller started the formalization process there were several iterations of design science methods and with the explosion of information science alongside the development of computer science there was a lot of application. Design science is a research methodology that mostly is used in developmental sciences like engineering, informatics, computer science, and so forth. Design Science can be applied to any science with a proper use case for it, and is not limited to these mentioned.

No.	Guidelines	Compliance
1	Design science research must produce a workable, practical artifact in the form of a construct, model, method, or instantiation	It can be used according to the original purpose, by the intended users. Be careful not to over promise. Be careful to promise the right things.
2	Ensure that the artifact produced is relevant and important	Has anyone tried to solve it before? Why hasn't it been solved before? How important can it be? Is it too difficult?
3	Rigorously evaluate the artifact produced	How do you know you accomplished what you wanted? Don't just ask people if they like it. Analytically using a mathematical model. Empirically using field study or experiment
4	Produce an artifact that makes a research contribution.	Solve a previously unsolved problem. Show that an artifact can be produced when it was previously unclear that this is possible.
5	Follow rigorous construction methods.	The method must be rigorous and replicable
6	Show the artifact is the outcome of a search process	This is done after you're finished
7	Clearly communicate the research process and outcome	Say a little about your thesis, any conference papers planned

Table 3.1: The seven guidelines for rigorous design science and how the work reported in this thesis aims to fulfil them. (Hevner et al., 2004, pg.83)

From these guidelines, we can not only dictate a plan but also use these to evaluate how the design science process was used. Therefore we will use these guidelines to set ourselves goals for compliance with these guidelines where it applies. Later, in chapter 5, we will return to these and evaluate how the planned compliance compares with the work done.

One: The Artifact: This might be the most challenging one to predict; I will start with making a model or an instantiation of a model. This artifact will be a working system for a 'testing-purposes' social media site (Lexitags *Veres* (2021)) made for bookmarks. The system will utilize SOLID Pods and data shapes and enable the user to use SOLID to store and control all their data while interacting with the social media site. This will be the artifact, after that, it becomes hard to pinpoint the development direction, it all depends on what occurs during development.

Two: Artifact relevance: There are other systems being created with SOLID right now, most for specific sites or systems, I found none for social media. As discussed previously, users' control over their personal information has never been more important, and since the thesis has as an objective to increase users' control over their personal information, it is both relevant and important. This is covered rigorously in chapter 2.

Three: Evaluation: The evaluation will be twofold, one part will be the evaluation of the design science process. The second part will be evaluating the artifact produced and the learned experience of how it is to work with these systems.

For the evaluation of the design science process, Williamson and Johanson point out three main steps to this process:

- Evaluate how well the design-science research problem has been specified;
- Evaluate whether a solution to the design-science research problem constitutes a contribution to knowledge;
- Evaluate how a solution to the problem was obtained.

(*Johanson and Williamson*, 2018, Chapter 11)

In the second part of the evaluation, we will be using the three steps Williamson and Johanson suggest which are used in both conduct of high-quality experiments, and high-quality action research, as detailed here:

> First, we should assess the quality of any *theory* proposed by the design-science researcher to account for why their product will possess desirable characteristics or lead to desirable outcomes. Second, we should assess the *instantiation* to determine whether it possesses these characteristics or has led to these outcomes. Third, to the extent design-science researchers have themselves undertaken a formal evaluation of their product, we can assess the *design and execution* of their research method against published criteria. (Davison, Martinsons, & Kock (2004) as cited in *Johanson and Williamson* (2018))

Four: The artifacts contribution : Testing shapes in cohesion with SOLID and exploring these relatively new technologies will be the main research contribution. Developing some shapes and exploring how it is to work with these systems and then deliberating on this in this thesis is how that will be accomplished. If possible leaning about the technical possibilities and or limitations of these technologies.

Five: Methods: The methods will be detailed in chapter 4. The development process will be detailed with a presentation of the site I will be using to test the shapes. In addition there will be a section devoted to the expansion of the data with DBpedias Spotlight.

Six: Show the process to the artifact: Showing that the artifact is, in fact, the outcome of the search process is done in this thesis, mostly with the background and development chapters demonstrating what technologies exist, and thereafter why the choices made during the development was based on and what the thought process was behind the selection of the solutions.

Seven: Clearly communicate the research process: This is the thesis, possibilities exist for some of the results from this thesis to be mentioned in a paper being written by Csaba Veres on SOLID as a broader piece of literature.

3.4 Summary

To end this chapter, I would like to return to the use of two methodologies in concert. Design science research is tremendously important in the information science sphere of research. Using traditional research methodologies has had an ill fit for research that requires us to develop models and artifacts, there are however always improvements to be made and changes to be tried.

> We argue that both design-science and behavioral-science paradigms are needed to ensure the relevance and effectiveness of IS research. (*Hevner et al.*, 2004, p.98)

Hevner et al make an argument for not only asking *what is true?* But also asking *What is effective?*(*Hevner et al.*, 2004, p.98), in addition to these there is a case to ask *Why do we need to solve this problem* This would be the critical design science research approach, what is true, what is effective, and why was it needed.

Science is often divided into three branches, formal, natural, and social sciences, and in this division information sciences, is put in the formal branch. One could argue that information science is partly all of these, as it detains elements of each, making it a critically important field for the future. Utilizing critical design science theory, with the ideas of *Netta Iivari* and *Kari Kuutti* we can take design science and continue on the path of developing our understanding of research through design. Moving forward from methodology, the following chapter will delve into development and results.

Chapter 4

Method & Development

The main goal of this thesis is to discover if and how one can use shapes for social media, to do this we need some testing data and a site to use as a test case. The LexiTags website allows us some real data to work with. In this chapter, we will be going through the development process and exploring the use of SHACL shapes for social media. The social media used as a test case in this thesis is the *LexiTags* website made by Martin Bruland based on the previous rendition of the same site made by Csaba Veres *Veres* (2011). Then we will examine the possibility to use DBpedias Spotlight for creating semantic tags for the bookmarks. Lastly, we will evaluate the design science process and the artifacts that it produced.

4.1 Artifact

In the methodology chapter we discussed design science and what an artifact could be, in this thesis the artifact is rather abstract, it is the SHACL shape used to validate the bookmarks in the SOLID pods stored from the react website, also the updated shape and the data collected in the expansion to the spotlight data collected. All of these could be artifacts on their own, so there will be a focus on the SHACL shape. the whole system could also be a method, but mainly in this thesis we will consider the SHACL shape as the artifact, and discuss how well the shape works to fulfil the needs presented in the 2.3 problem space section.

4.2 The beginning

The development process started with familiarising myself with the SOLID specification and SOLID development environment, this proved to be time-consuming. I spent much time trying out the SOLID servers and pods to accumulate knowledge about the functions of the systems and the practical use of these applications. Having a knowledge of how SOLID pods and servers work is essential when we try to imagine future use for these systems, especially this is true when we discuss topics like shapes technology. Looking into and researching these systems and specifications, and testing them out was a lengthy process, the selection of shape expression language was the most comprehensive of them, as we discussed about the *SHACL vs ShEx* in section 2.10. I knew that the site that would be used for testing would pertain to bookmarks, when

I landed on SHACL, I started to create a shape with the information I thought that a bookmarking site would store. Developing shapes turned out to be more intricately than the first assessment concluded, it was difficult to develop in SHACL-Playground[1] *Knublauch* after the initial period of trying SHACL and SOLID the site that was going to be used for testing took form.

4.3 Outcome of SHACL or ShEx

SHACL and ShEx have, as we discussed in 2.10 section, many similarities between the two, and some more esoteric differences, some of these were not apparent until development was well underway.

According to (*Labra Gayo et al.*, 2017, Chapter 7) there was a hope for the unification of these technologies in 2014, no such convergence happened and both technologies still exist, the current SHACL primer being released in 2017 *GROUP* (2022), and ShEx primer in 2019 *SHAPE-EXPRESSIONS-COMMUNITY-GROUP* (October 2019).

Tim Berners-Lee has some thoughts on *W3C's Design Issues* page, it must be noted that these pages as for research and development purposes and are not published in a larger context, here is the disclaimer from the site. I include this with the reasoning that this thesis is for development purposes, but since this is also published I would like to include the disclaimer in quotes.

> These statements of architectural principle explain the thinking behind the specifications. These are personal notes by Tim Berners-Lee: they are not endorsed by W3C on anyone else. They are aimed at the technical community, to explain reasons, provide a framework to provide consistency for future developments, and avoid repetition of discussions once resolved. *Berners-Lee*

When discussing these specific systems, it can be good to keep in mind that for most cases it should be possible to "translate" between them, by this it is meant that we can take a shape written in SHACL and use a conversion program to create the same shape in ShEx [2]. (*Labra Gayo et al.*, 2017, chapter 7.1) This being said, there is a difference between what they will show us when we use them to validate a graph against a shape. Foremost of these is the validation result of SHACL when a graph is satisfactory we are not given a validation report, rather we are just given a "validated" output. This result might be enough for simple large-scale data validation on central data storage, but we are wanting to use data shapes for testing a validating decentralized data stored in SOLID pods. We will return to SHACL and ShEx in the discussion chapter, where we will reflect on the differences between the two and the limitations of the technology as it stands. Now with the first shape created in SHACL, we can use it to validate the data in the SOLID pods.

[1] https://shacl.org/playground/

[2] It is not a perfect translation, and manual inspection and oversight is required

4.4 The site and the data

The site used for testing shapes that were created by Martin Bruland is based on a site made by Csaba Veres to organise bookmarks. The site is built in ReactJS[3] using SOLID pods as the data storage, no other backend system is used. There is one pod used to store the other pod's address, as one needs a central location to know where to find the other data for functions such as most popular bookmarks, and for the Wordcloud. This entails that the site must access all the PODs to collect the most popular bookmarks and list them, this was included to test the speed of the system and how well it scaled with the number of users and bookmarks. The site is made to store the users' bookmarks and allow the user to make new, edit old, or delete their bookmarks, as well as collect and display the 10 most popular bookmarks. This process and details are available to read in Mr Bruland's thesis, This quick rendition is to give an idea of the site, and the data, used for testing, *Bruland* (2022).

4.4.1 SOLID PODs

The website LexiTags was made using NodeJS and ReactJS, these systems are meant to be run from a server and not in a browser. However, since SOLID allows us to operate a site in-browser with PODs as the backend. The issue with testing the shapes arose when I tried to validate the RDF data in the Pods against the SHACL shape. React does not allow for the collection of files from local storage, so we would need to store the files online, on a Pod or another platform. There were other problems with testing the data with the shape that we will get into in the next section. During the testing and trials, I consulted Mr Bruland about the site and how to get it to work, even with his assistance and knowledge, I was unable to make the validation pack [4] work. Part of the issue was that these systems are not intended to be run in-browser, and since very few systems have been designed to work in this fashion, the resources to discover facilitation are fewer. This is to say that there might be a solution to this problem, but I was unable to find it in the time span allotted for the research and testing before I had to move on with a different approach.

4.4.2 The Bookmarks

The original data structure was created based on the requirements for a bookmark with this data structure: the title, URL, date, description, and tags.
These data points were attributed to the Schema.org [5] *Group and Group* Schemas, SHACL will not check if the schema exists for the linked item, or if the data conforms to the specified data. It would be interesting to have the option of checking the schema for a *thing* control if it conforms to the data structure of the *thing*, but that would add considerable time to the validation process. With the site already having to check

[3] a javascript library ("`https://react.dev/`")

[4] "`https://www.npmjs.com/package/rdf-validate-shacl`"

[5] "`https://schema.org/`" is described by themselves as "Schema.org is a collaborative, community activity with a mission to create, maintain, and promote schemas for structured data on the Internet, on web pages, in email messages, and beyond." and SHACL when looking at

multiple PODs to find the most bookmarked bookmarks, lengthening the process does not seem desirable. However, this is something that could be implemented in future work, or just as a part of a different type of system. Hence the schema part of the data structure is to conform to best practices and not to add to the intricacy of this project. It can be noted that if one were to check many shapes the retrieval time for a schema might be insignificant in comparison, and this interaction might be desirable.

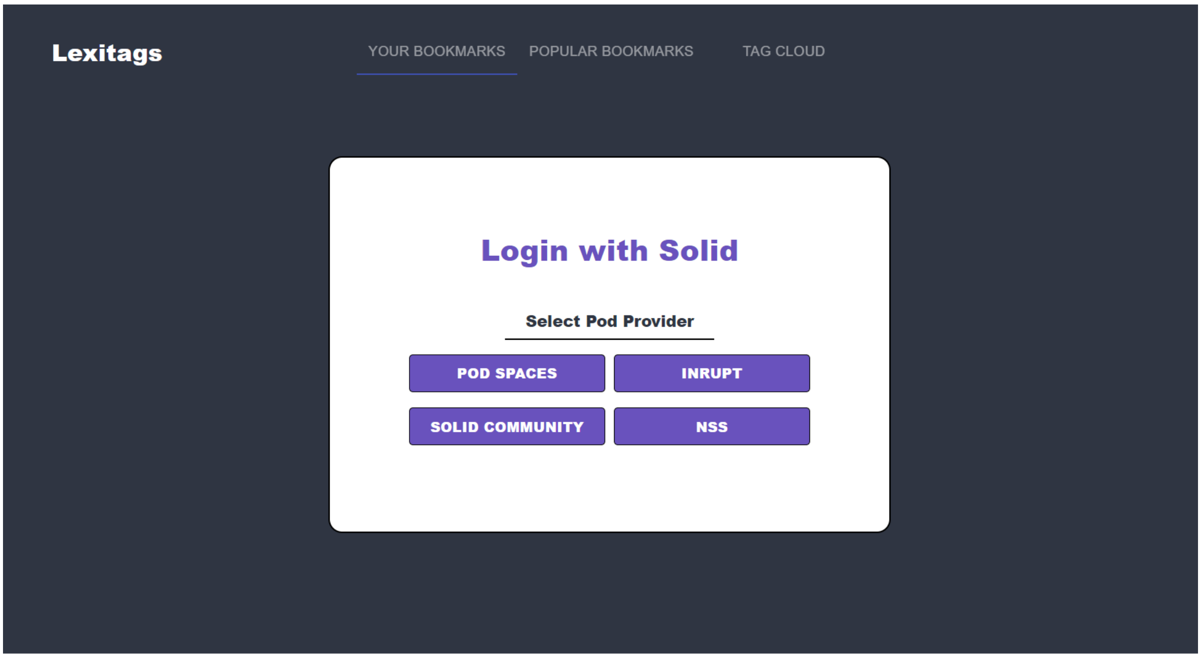


Figure 4.1: Picture showing the login page for LexiTags Veres (2021)

The LexiTags website is a good site for our testing purposes, perhaps a commercial site would include other functionalities, or even a community-driven resource, however for the testing data for this thesis it will work perfectly. Additionally, there is room for expansion and extension of this data, and limitation drives creativity. Based on how the bookmarks on the website were implemented by Bruland, I created a SHACL shape after the data to check if the data validates. For this I created some test data in Protégé, to make development easier I decided to create three versions of each test bookmark, one empty, one perfect, and a final one with two of each property. This was done to effectively test if the SHACL shape was validating properly. Since SHACL only gives warnings and errors when validation fails, it is important to have both cases to test if the requirements are being fulfilled, and not just show a false positive by not validating anything. The empty bookmark was valuable to the testing process to discover if the nodes in the data were being targeted correctly. Conversely, the bookmark with double the desired number of attributes controls that the maxnumber was functioning correctly. This kind of rudimentary testing environment was the easiest way to test when a validation report might be misleading; a graph that has none of the desired nodes will show nothing but a Boolean *validated*, for the reason that no parts are *wrong*.

Here is the SHACL shape with constraints:

Bookmark		
Name of requirement	type of data	requirement
Url	String	max 1 and min 1
dateCreated	date and time	max 1 and min 1
hasDescription	String	max 1 and min 1
TagString	String	no min or max
title	String	max 1 and min 1

This is how a bookmark looks like when store in a SOLID pod:

```
<#Bookmark/youtube.com>
    a schema:Bookmark;
    schema:dateCreated "2/2/2022, 2:18:26 PM";
    Thing:alternateName "youtube";
    Thing:description "video hosting and streaming site,
    where one can find anything, alsmost";
    Thing:disambiguationDescription
    <#Tag/Blogger>, <#Tag/Streaming>, <#Tag/Tutorials>,
    <#Tag/Video>, <#Tag/Kittens>;
    Thing:url "youtube.com".
```

And in figure 4.2 we can see what the bookmark looks like in-browser [6], we can see that the structure of the data corresponds well to the structure of the browser side, and is quite readable compared to the SHACL shape, as one could expect from a GUI [7]. The SHACL language is intended to be readable by humans, and for smaller snippets, it can be, conversely, for larger documents, it can be hard to get a grasp of the structure.

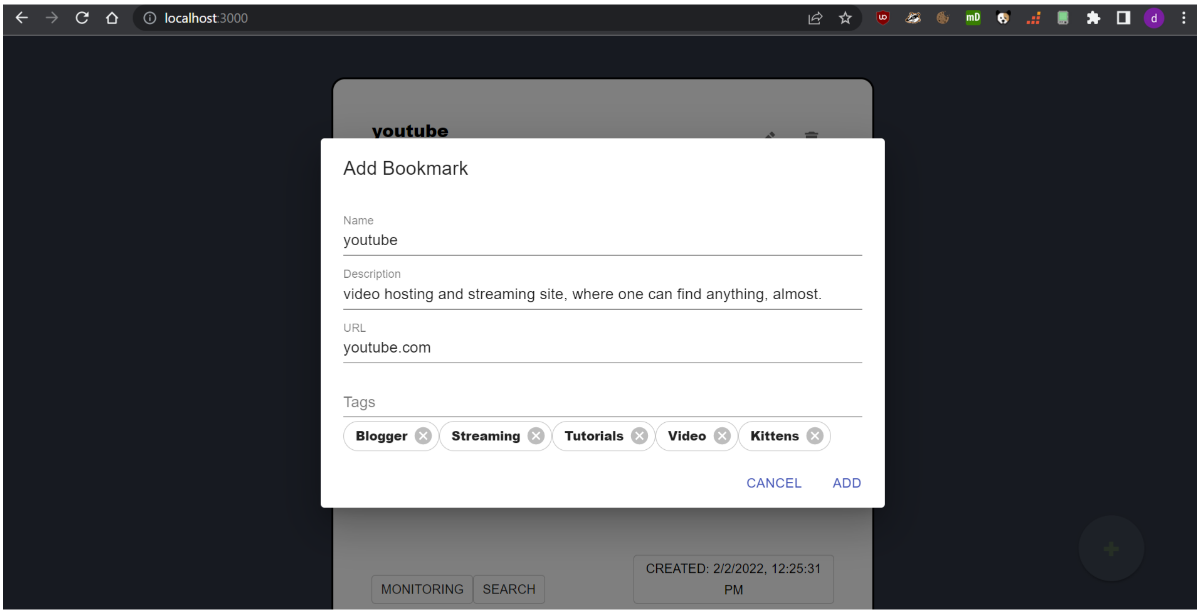


Figure 4.2: Picture showing Youtube bookmark on the LexiTags site Veres (2021)

For validating this data I used a simulation in Protégé, along with a data graph constructed for this purpose, this graph can be seen in the appendix A.1.8. This shape

[6] Google Chrome is being used
[7] Graphical User Interface

will validate the test data from Protégé, however, that data is not identical to the data in the SOLID pods created by LexiTags.

```
ex:BookmarkShape
    a sh:NodeShape ;
    sh:targetClass ex:Bookmark ;  # Applies to all Bookmarks
    sh:property [                 # _:b0
        sh:path ex:URL ;       # constrains the values of ex:hasURL
        sh:minCount 1 ;
        sh:maxCount 1 ;
    ] ;
    sh:property [              # _:b1
        sh:path ex:URL ;          # constrains the values of ex:hasURL
        sh:datatype xsd:string ;
        sh:severity sh:Warning ;
    ] ;
    sh:property [                 # _:b0
        sh:path ex:dateCreated ;    # constrains the values of ex:dateCreated
        sh:minCount 1 ;
        sh:maxCount 1 ;
    ] ;
    sh:property [                   # _:b1
        sh:path ex:dateCreated ;    # constrains the values of ex:dateCreated
        sh:datatype xsd:dateTime ;
        sh:severity sh:Warning ;
    ] ;
    sh:property [                # _:b0
        sh:path ex:Description ;    # constrains the values of ex:description
        sh:minCount 1 ;
        sh:maxCount 1 ;
    ] ;
    sh:property [                 # _:b1
        sh:path ex:Description ;     # constrains the values of ex:description
        sh:datatype xsd:string ;
        sh:severity sh:Warning ;
    ] ;
    sh:property [                  # _:b0
        sh:path ex:alternateName ;    # constrains the values of ex:alternateName
        sh:minCount 1 ;
        sh:maxCount 1 ;
    ] ;
    sh:property [                 # _:b1
        sh:path ex:alternateName ;   # constrains the values of ex:alternateName
        sh:datatype xsd:string ;
        sh:severity sh:Warning ;
    ] ;
```

```
    sh:property [                    # _:b0
        sh:path ex:disambiguationDescription ;  # constrains the values of ex:disambi
    ] ;
    sh:property [                            # _:b1
        sh:path ex:disambiguationDescription ;  # constrains the values of ex:disambi
        sh:objecttype xsd:Tag ;
        sh:severity sh:Warning ;
    ] ;

    sh:closed true ;
    sh:ignoredProperties ( rdf:type owl:topDataProperty owl:topObjectProperty ) ;
    .
 ###### add SHACL vocabulary ######
```

To validate the data from Lexitags, we require a shape where the Tag is its own class and the Bookmark shape includes a property of this Class. Here is how that would appear, mark the changes in case, it is important to be careful with case sensitivity.

```
ex:BookmarkShape
    a sh:NodeShape ;
    sh:targetClass ex:Bookmark ;  # Applies to all Bookmarks

    sh:property [
sh:path ex:disambiguationDescription ;
sh:class ex:Tag;
    ];

    sh:property [                 # _:b0
        sh:path ex:URL ;       # constrains the values of ex:URL
        sh:minCount 1 ;
        sh:maxCount 1 ;
    ] ;
    sh:property [              # _:b1
        sh:path ex:URL ;           # constrains the values of ex:URL
        sh:datatype xsd:string ;
        sh:severity sh:Warning ;
    ] ;
    sh:property [                 # _:b0
        sh:path ex:dateCreated ;       # constrains the values of ex:dateCreated
        sh:minCount 1 ;
        sh:maxCount 1 ;
    ] ;
    sh:property [               # _:b1
        sh:path ex:dateCreated ;        # constrains the values of ex:dateCreated
        sh:datatype xsd:dateTime ;
        sh:severity sh:Warning ;
    ] ;
    sh:property [                # _:b0
```

```
        sh:path ex:description ;          # constrains the values of ex:Description
        sh:minCount 1 ;
        sh:maxCount 1 ;
    ] ;
    sh:property [                         # _:b1
        sh:path ex:description ;          # constrains the values of ex:Description
        sh:datatype xsd:string ;
        sh:severity sh:Warning ;
    ] ;
    sh:property [               # _:b0
        sh:path ex:alternateName ;          # constrains the values of ex:title
        sh:minCount 1 ;
        sh:maxCount 1 ;
    ] ;
    sh:property [                         # _:b1
        sh:path ex:alternateName ;          # constrains the values of ex:title
        sh:datatype xsd:string ;
        sh:severity sh:Warning ;

];


.
ex:TagShape
    a sh:NodeShape ;
    sh:targetClass ex:Tag;
    sh:property [               # _:b0
        sh:path ex:NamedIndividual;          # constrains the values of ex:Tag
    ] ;
    sh:property [                         # _:b1
        sh:path ex:NamedIndividual  ;        # constrains the values of ex:Tag
        sh:TargetClass xsd:NamedIndividual;
        sh:severity sh:Warning ;
    ] ;

    sh:closed true;
    sh:ignoredProperties ( rdf:type owl:topDataProperty owl:topObjectProperty ) ;
    .
 ###### add SHACL vocabulary ######
```

For this shape to validate the data in the SOLID pods we need to alter it slightly, the unaltered data can be seen in the appendix A.1.7. However since I was unable to make an alternate branch of the Lexitags site where the semantic tags could be entered, I edited this Graph to fit the shape, the changes are mostly superficial. I had to alter the data, specifically, I changed the prefix ":Thing" to ":ex" for simplicity, more importantly, I had to delete aspects of the Tag elements, as the wordcloud elements were making problems. This was done to make the testing easier, the wordcloud elements are not essential for this thesis. If I was working with a branched site of LexiTags I could also comment out the wordcloud sections while testing, although the shape could also be expanded to include these elements. The Shape now validates the Tags as their own class when validating the Bookmark elements.

```
@prefix : <#>.
@prefix schema: <http://schema.org/>.
@prefix lexit: <./>.
@prefix www: <http://www.schema.org/>.
@prefix Thing: <http://schema.org/Thing/>.
@prefix : <http://www.example.org/#> .
@prefix owl: <http://www.w3.org/2002/07/owl#> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix xml: <http://www.w3.org/XML/1998/namespace> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@base <http://www.example.org/#> .
@prefix ex:    <http://www.example.org/#> .

###  http://www.example.org/#Bookmark
#### <Bookmark> rdf:type owl:Class .

###  http://www.example.org/#Tag
#### <Tag> rdf:type owl:Class .

lexit:bookmarks
    a schema:Dataset;
    www:Contains
        <#Bookmark/https://bt.no>,<#Bookmark/https://ba.no>,
        <#Bookmark/https://next-episode.net/the-boys>,
        <#Bookmark/https://next-episode.net/the-walking-dead>,
        <#Bookmark/https://tv2.no>.
<#Bookmark/https://ba.no>
    a schema:Bookmark, :Bookmark;
    ex:dateCreated "2001-10-26T21:32:52+02:00"^^xsd:dateTime; ###"1/31/2022, 9:15:54
    ex:alternateName "Bergens Avisen";
    ex:description "Local newspaper from Bergen";
    ex:disambiguationDescription
    <#Tag/Bergen>, <#Tag/Local-newspaper>, <#Tag/News>;
    ex:URL "https://www.ba.no"^^xsd:string .
<#Bookmark/https://bt.no>
    a schema:Bookmark, :Bookmark;
    ex:dateCreated "1/31/2022, 9:15:19 AM"^^xsd:dateTime;
    ex:alternateName "Bergens Tidende";
    ex:description "Another local newspaper from Bergen";
    ex:disambiguationDescription <#Tag/Bergen>, <#Tag/News>, <#Tag/Norway>;
    ex:URL "https://bt.no".
<#Bookmark/https://next-episode.net/the-boys>
```

```
    a schema:Bookmark, :Bookmark;
    ex:dateCreated "1/31/2022, 9:18:11 AM"^^xsd:dateTime;
    ex:dateModified "1/31/2022, 9:19:06 AM";
    ex:alternateName "next episode tracker - The boys";
    ex:description "online series episode tracker";
    Thing:disambiguationDescription
    <#Tag/Episodes>, <#Tag/Next-episode>, <#Tag/Series>, <#Tag/Tracker>;
    ex:URL "https://next-episode.net/the-boys".
<#Bookmark/https://next-episode.net/the-walking-dead>
    a schema:Bookmark, :Bookmark;
    ex:dateCreated "1/31/2022, 9:19:32 AM"^^xsd:dateTime;
    ex:dateModified "1/31/2022, 10:25:01 AM";
    ex:alternateName "next episode tracker - the walking deadasd";
    ex:description "";
    Thing:disambiguationDescription
    <#Tag/Next-episode>, <#Tag/Series>, <#Tag/Tracker>;
    ex:URL "https://next-episode.net/the-walking-dead".
<#Bookmark/https://tv2.no>
    a schema:Bookmark, :Bookmark;
    ex:dateCreated "1/31/2022, 9:16:38 AM"^^xsd:dateTime;
    ex:alternateName "TV2 News" ;
    ex:description "News from Norway";
    Thing:disambiguationDescription <#Tag/News>, <#Tag/Norway>;
    ex:URL "https://tv2.no" .

<#Tag/Bergen> a schema:Tag, :Tag.
<#Tag/Episodes> a schema:Tag, :Tag.
<#Tag/Local-newspaper> a schema:Tag, :Tag.
<#Tag/News> a schema:Tag, :Tag.
<#Tag/Next-episode> a schema:Tag, :Tag.
<#Tag/Norway> a schema:Tag, :Tag.
<#Tag/Series> a schema:Tag, :Tag.
<#Tag/Tracker> a schema:Tag, :Tag.
```

4.4.3 Validation and development issues

There were several instances of data not validating, but no issues were apparent with the data at first glance. This problem was multiplied by the inability to not only run the validation directly against the SOLID pods but also because developing shapes in SHACL gives very little feedback. When developing shapes, I discovered that the process was quite different from the programming experiences I previously had. Developing in Protégé is a temporary solution, the data created in Protégé is easier to test than real-world examples, and replication is not 1:1. Creating SHACL shapes has the unfortunate drawback that it does not grant a data map of the RDF like ShEx, rather, you gain only a validation report, that even if it validates nothing will return true.

I delved deep into possible issues with validation, and formatting, the possible issues with dateCreated; there was an issue where the only mistake that could be found was that *datetime* while seeming to be correct, did not conform to the *datetime* from Schema.org, and for a while, I had to investigate if SHACL did check conformity against the Schema.org specifics, it does not. Then I looked into the formation of the "Thing:Bookmark" believing that the structure of the bookmark itself was causing the validation errors. It turned out to be a wild goose chase, the real culprit was that the dateCreated was correct, however, they were not classified as having datetime as they were required to by the shape. They were in the right format, but the shape cannot tell that unless specified: "ex:dateCreated "1/31/2022, 9:15:19 AM"^^xsd:dateTime;". Many of the problems I found were caused by the lack of feedback on what was happening, it requires a deep understanding of the system, to figure out mistakes. Hence the need for a sturdier development space. This could cause issues in future development as much of the semantic conformance of any given page might be less than ideal.

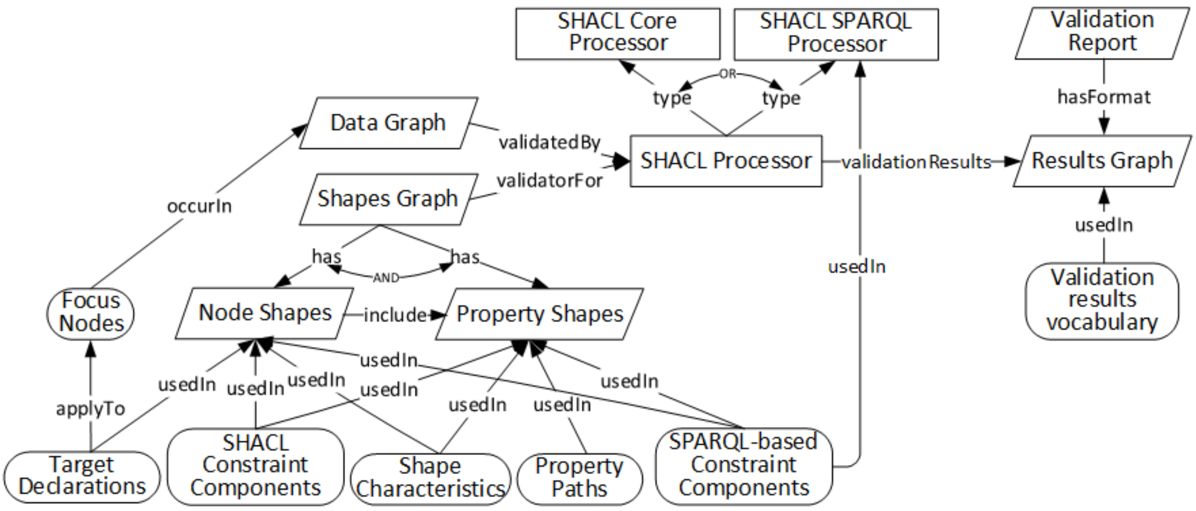


Figure 4.3: Graphical overview of the SHACL validation process, how it revolves around the DataGraph and the ShapeGraph Teijgeler (2016)

In figure 4.3 we can see how the SHACL processor pertains to the data graph and the shape graph, as of now we simply feed the processor both the graphs, giving us a validation result.

4.5 DBpedia spotlight

DBpedia Spotlight is as discussed previously an API used to annotate data with structured data from Wikipedia made available on DBpedia. On the suggestion of Enrico Motta, I started to implement the spotlight API, to give the tags some semantic depth, in addition, this will give us a chance to see how an expansion of the data is handled by the SHACL shape.

4.5.1 Expansion and semantic tags

The expansion of the original shape was a relatively simple addition, as the form of the semantic tags follows the same as the manually added tags, With the inclusion of a new name. The shape system is easily updated with this kind of expansion, the problem lies with the shapes being separate from the data and the system for storage and updating the data, and even separate from the data itself. One can include the shape that one has used for the data, in the data. Having the shape with the data only goes so far as to allow us to implement a checking system.

After some initial testing of Spotlight, a Python script was created [8] using BeautifulSoup[9], with this script we can scrape a website for its text and then we can strip it for the parts we are not interested in such as: the images, the scripts, the stylesheets, etc. After scraping and striping the site's text we are left with something we can run through Spotlight. After sorting out text we move on the Spotlight's annotation, the central element is the settings we have for Spotlight. *Mendes et al.* (2011) states that in configuring Spotlight we should first consider what we are annotating, herein lies some of the issue, we do not know what we are annotating. To be specific it could be a large internationally known company's website or a local community resource. To highlight these differences I tested some websites, with different configurations to Spotlight, to explore what generated tags we would be left with. These are shown in tables 4.2, 4.4, and 4.6.

Quickly we will go over the settings for Spotlight before we move on to the examples. There are two settings to Spotlight, confidence, and support. Support is the number of links to a DBpedia resource there has to be before it will appear as annotated. Confidence is a number between 0 and 1, it contains several parameters and they estimate that setting the confidence to 0.7 will eliminate around 70% of incorrect hits. Precision is all true positives selected divided by all the selected elements, while recall is the true positives divided by all the relevant elements. Finding all the relevant elements is difficult for us to do since it would require a manual search through all hits with Spotlight with no filter, we will be discussing the precision, as that is easier to determine and discuss.

They do have a disclaimer on the service that the more text you have the better the accuracy, it is hard for this type of automatic process to detect the nuances of language at the best of times, with a small sample size there will be a decrease in accuracy. They still count their model as competitive.

> We compared our system with other publicly available services and showed how we retained competitiveness with a more configurable approach. *Mendes et al.* (2011)

They also stated that the best F1-score [10] was reached with a confidence value of 0.6 *Mendes et al.* (2011), I choose to include this confidence value in one of the tests, but the paper was written in 2011 and it is probable that they have improved it since then, I found no logs of improvements made, but investigating the Github-repository

[8] Script in appendix

[9] BeautifulSoup is a PYTHON package for scraping HTML or XML and collecting what you need *Richardson* (2004)

[10] F1-score is an average(Harmonic mean) of precision and recall

Confidence 0.9 Support 400	Confidence 0.9 Support 10
Confidence 0.6 Support 100	Confidence 0.5 Support 400

Table 4.1: table showing the settings choosen for Spotlight

shows that the last update was made in 2021, so it is, at a minimum, being maintained. One thing to mention is that not all websites allow for scraping or access with bots, as a detection tool I added a printout for the HTML code A.2, this is a hack however, and in a larger setting, we would need to have real error messaging if this happened. Let us move on to the test cases used for trying Spotlight.

4.5.2 Test cases

In regards to the tables 4.2, 4.4, and 4.6 Please note that additions remove the results from the previous settings to make the difference apparent, some of the settings have a few results less, making the table bigger would make it hard to read if it were four columns.

I picked these three websites *equinor.com*, *uib.no*, and *www.wwf.no* [11] as examples to use for annotation, I also added them into LexiTags as bookmarks, and I will show how the spotlight data would have looked if I was able to integrate it into the site. [12]

The settings chosen for the confidence are of gradually decreasing strictness, first, at 0.9, then secondly, at 0.6, and thereafter 0.5 as the least strict setting. To demonstrate how the requirements for detection will create very different results depending on the extent of the site tested. For this purpose, the three different support settings chosen were: 400, 100, and 10. And these are combined with the confidence settings as shown in this table 4.1 [13].

For this thesis we will not have access to the true recall number, it would be hard to find in a reasonable time, we will look at precision.

University Of Bergen

I choose the University of Bergen's website to have a mid-range site for testing, in hindsight, I see that the front page might have been a bad choice since it is mostly changing content, nevertheless, I think it works well enough for our purposes here. Looking at 4.2 results from UIB's website we can see how most of the tags are relevant to the site, the front page for UIB has different articles and information all the time, so that is why the tags are about *Fiji* and *Artificial intelligence*. Although most of the tags are correct, correct in the sense that they are referring to what is being written about on the site, but not correct in the sense that they are about UIB. This is also what happens if you run a new site through Spotlight, you will get different results every day. For each run we see that Bergen is the first result, the rest are references to the articles at the time. let us look at the results for each of the settings, having the very

[11] "https://www.equinor.com", "https://www.uib.no", "https://www.wwf.no"

[12] I removed the prefix ""*http://DBpedia.org/resource/*" from all the tags, for examining of the DBpedia data about a subject add this prefix to the word, that will create a working link.

[13] Notes: the day I tested there was content about Fiji and Greenland, but I have no idea where it go middle earth from

Settings in spotlight	Site: https://www.uib.no	Additions
Confidence 0.9 Support 400	Bergen, Fiji, Pacific Ocean, Artificial intelligence, United Nations, New York City, Microscope	
Confidence 0.9 Support 10	Bergen, Fiji, Pacific Ocean, Trond Mohn, Artificial intelligence, United Nations, New York City, Mycobacterium leprae, Gerhard Armauer Hansen, Microscope	Trond Mohn, Mycobacterium leprae, Gerhard Armauer Hansen
Confidence 0.6 Support 100	Bergen, Greenland Sea, Fiji, Ice, Pacific Ocean, Artificial intelligence, Norwegian krone, United Nations, New York City, Mycobacterium leprae, Bacteria, Microscope	Greenland Sea, Ice, Norwegian krone, Bacteria
Confidence 0.5 Support 400	Bergen, English language, Fiji, Ice, Pacific Ocean, Discovery Channel, Evolution, Artificial intelligence, Norwegian krone, United Nations, New York City, Bacteria, Microscope	English language, Discovery Channel, Evolution

Table 4.3: table with the University Of Bergen data collected in Spotlight

strict confidence 0.9 and support 400 as a baseline for our purposes. When we lower the support to 10, but keep the confidence high we see that we have a number of true positives that did not have enough links to them to qualify for our first parameters. All of these results are relevant to the articles that were on the front page the day of testing, [14] As soon as we lower the confidence number to 0.6 we see more general terms like "Norwegian krone", "ice", and "Bacteria", but we also see a reference to the "Greenland Sea". Here we get some very general terms being annotated and one genuine annotation, with 11 correct annotations and 3 false positives we are well within the 70% estimation quoted in *Mendes et al.* (2011). Finally to see if we would get more false positives or if adjusting the support number up would weigh up for having a lover confidence number. Here we see some more false positives, very general terms, we do also lose "Greenland Sea" a true positive, giving good credence to a threshold setting of 0.6 confidence.

Equinor

Repeating the process used for the previous site, we start off by establishing a baseline with strict parameters; confidence 0.9 and support 400. Shown in the table 4.4 are many accurate annotations, more than seen for the other sites with these settings. Equinor being a large international company explains the larger set of annotations, their page has more information about them specifically. There are two false positives with the first parameters, "Test Cricket" and "Central European time", the rest have relevance to the company or partners. We can see from the first change of support from 400 to 10 that this does not really change much, there are some notable additions. Here we have some clear positives to lowering the support number and getting a number of

[14] Regrettably I did not take screenshots of the UIB page at the time I did the data collection, I only thought of doing so later.

Settings in spotlight	Site: https://www.equinor.com	Additions
Confidence 0.9 Support 400	Equinor, Norway, London, Greenhouse gas, Central European Time, Norwegian Sea, Test cricket, Glenn L. Martin Company, Stavanger, RWE	
Confidence 0.9 Support 10	Equinor, Norway, British Energy, London, Greenhouse gas, Allmennaksjeselskap, Central European Time, Norwegian Sea, Test cricket, Glenn L. Martin Company, Fornebu, Stavanger, RWE, Johan de Witt	British Energy, Allmennaksjeselskap, Fornebu, Johan de Witt
Confidence 0.6 Support 100	Equinor, Norway, British Energy, Coal, London, Greenhouse gas, Discover Card, Oslo Stock Exchange, New York Stock Exchange, Central European Time, Norwegian Sea, Njörðr, Tonne of oil equivalent, 1973 oil crisis, Global warming, Test cricket, Glenn L. Martin Company, Volcanic Explosivity Index, Fornebu, Stavanger, Liquefied natural gas, Germany, RWE, Energy security, Johan de Witt	Coal, Discover Card, Oslo Stock Exchange, New York Stock Exchange, Njörðr, Tonne of oil equivalent, 1973 oil crisis, Global warming, Volcanic Explosivity Index, Liquefied natural gas, Germany, Energy security
Confidence 0.5 Support 400	Digital media, Industry, Equinor, Energy industry, Norway, Coal, London, Greenhouse gas, United Kingdom, Annual general meeting, New York Stock Exchange, Central European Time, Norwegian Sea, Social media, 1973 oil crisis, Global warming, Test cricket, Glenn L. Martin Company, Stavanger, Liquefied natural gas, Information technology, Germany, RWE, Energy security, European Union	Digital media, Industry, Energy industry, United Kingdom, Annual general meeting, , Social media, Information technology, European Union

Table 4.5: table with Equinor data collected in Spotlight

annotations that did not show up on account of having to few links to them, the same as we saw with UIB. The only general item here is "Allmennaksjeselskap" but that would be a good fit for a semantic tag, even if it might seem generic. Lowering the confidence to 0.6 and increasing support to 100 grants a large increase in annotations. There are four tags that are false positives or to general to have meaning, two of which appeared with the first parameters. Lowering the confidence to 0.5 and increasing the support to 400 again creates many results too general for use as semantic tags. We do see the false positive percentage increase with lowering the confidence, even with increasing the support requirements. The last test settings give us eight new annotations of which (at least) three are too general to be counted as a true positive; "Digital media", "annual general meeting", "Social media", and "European Union" are all in some ways correct, but are not very useful to determine what the Equinor website is about. A higher false positive number is useful for semantic tagging. Next, we will look at the final site annotated WWF.

WWF

World Wide Fund For Nature or WWF is the last site I used Spotlight to annotate for the generation of semantic tags, I choose WWF because it is a Norwegian branch of a larger organisation and I wanted to see how Spotlight would deal with a smaller site comparatively with the others tested. In the 4.6 we can see what Spotlight only finds "Glossary of professional wrestling terms" showing that abbreviations, even when confidence is 0.9 and support is 400, are hard to correctly assess. Further, we see with the next step that we continue to get false positives, with the "Om Records", but also a

Settings in spotlight	Site: https://www.wwf.no	Additions
Confidence 0.9 Support 400	Glossary of professional wrestling terms	
Confidence 0.9 Support 10	Om Records, Glossary of professional wrestling terms, Living Planet Report	Om Records, Living Planet Report
Confidence 0.6 Support 100	Liv Tyler, Om Records, World Wide Fund for Nature, Glossary of professional wrestling terms, WNEW-FM, Red Party (Norway)	Liv Tyler, World Wide Fund for Nature, WNEW-FM, Red Party (Norway)
Confidence 0.5 Support 400	Tempo, World Wide Fund for Nature, Glossary of professional wrestling terms, Hectare	Tempo, Hectare

Table 4.7: table with World Wide Fund For Nature data collected in Spotlight

true positive with "Living Planet Report". Lowering the confidence to 0.6 and support to 100 gives us finally the WWF itself and some more false positives appear, a radio channel, and a political party. Finally with confidence at 0.5 and support at 400 we get the last results, "Tempo" and "Hectare" both actual things but not good items for out quest for semantic tags.

4.5.3 Expansion of shape with new data

Here I am going to detail how the semantic tags, generated by running a bookmark's web page through Spotlight, would be included in the new shape. First, we need to look at the requirements for what I am going to call the *SemanticTag*. Very much like the manually created tags for the bookmark, the SemanticTag also has to have the same parameters. There is no minimum number for SemanticTag either since we can not be sure to find any, conversely, we cannot have a maximum either, since we don't know the maximum number of tags possible. If we desired a maximum it could be achieved in the collection part, limiting the annotations passed to the site to the first ten or twenty tags, or even just having the first five. For our SemanticTag we will then duplicate the aspect from *disambiguationDescription*. Then as we can see in 4.1 adding the SemanticTag as a new property in the shape is very fast and easy, since we know that the parameters that we have from the tags can be used for the generated tags too. Note that we still use the *disambiguationDescription* as the "type".

We would also need to include the class in the Bookmark shape, and the RDF data:

Our shape poses few restrictions to the Tag shapes since this is modelled after the bookmarks used in LexiTags. However, the structure created here could implement other restrictions to the Tag or SemanticTag shape, and then have those restrictions be part of the validation of the Bookmark. This shows how the process of entering a new element into an existing shape can be handled. Below is the modified RDF data 4.4, changed to be validated by the shape. These changes would need to be implemented in the storage process in LexiTags. This is a roundabout way to illustrate how SHACL and SOLID would have to be very particularly handled to be used.

Listing 4.1: SHACL shape expansion

```
ex:TagShape
    a sh:NodeShape ;
    sh:targetClass ex:Tag;
    sh:property [                       # _:b0
        sh:path ex:NamedIndividual;     # constrains the values of ex:Tag
    ] ;
    sh:property [                       # _:b1
        sh:path ex:NamedIndividual  ;   # constrains the values of ex:Tag
        sh:TargetClass xsd:NamedIndividual;
        sh:severity sh:Warning ;
    ] ;

ex:SemanticTagShape
    a sh:NodeShape ;
    sh:targetClass ex:SemanticTag;
    sh:property [                       # _:b0
        sh:path ex:NamedIndividual;     # constrains the values of ex:SemanticTag
    ] ;
    sh:property [                       # _:b1
        sh:path ex:NamedIndividual;     # constrains the values of ex:SemanticTag
        sh:TargetClass xsd:NamedIndividual;
        sh:severity sh:Warning ;
    ] ;
```

Listing 4.2: SHACL shape semantic tag

```
    sh:property [
        sh:path ex:disambiguationDescription ;
        sh:class ex:SemanticTag;
    ];
```

Listing 4.3: SHACL shape class addition

```
###  http://www.example.org/#SemanticTag
#### <SemanticTag> rdf:type owl:Class .
```

4.6 Summary

In this chapter, I have described what occurred at each step of the process, with a focus on detailing the decision-making process and why the choices and decisions were made. There are several unfortunate occurrences where technological incompatibilities with different systems made development and testing difficult. The unforeseen technical incompatibilities are unfortunate, nevertheless, the abstraction of the proposed use of shapes in this fashion is possible. Using shapes as proposed is difficult to achieve with currently available tools. Nevertheless, we can observe a number of projects currently using these solutions and developing in this space, despite the challenges detailed in this chapter. Having described the development and experimental processes we will proceed with to discussion chapter.

Listing 4.4: SHACL shape version two

```
@prefix : <#>.
@prefix schema: <http://schema.org/>.
@prefix lexit: <./>.
@prefix www: <http://www.schema.org/>.
@prefix Thing: <http://schema.org/Thing/>.

lexit:bookmarks
    a schema:Dataset;
    www:Contains
        <#Bookmark/https://www.equinor.com>, <#Bookmark/https://www.uib.no>,
        <#Bookmark/https://www.wwf.no>.
<#Bookmark/https://www.equinor.com>
    a schema:Bookmark;
    ex:dateCreated "2/28/2023, 3:39:07 PM";
    ex:alternateName "Equinor";
    ex:description "Equinor is the biggest oil company in Norway";
    ex:disambiguationDescription
    <#Tag/Corporation>, <#Tag/Norwegian>, <#Tag/Oil>;
    ex:url "https://www.equinor.com".
    ex:disambiguationDescription
    <#SemanticTag/Equinor>, <#SemanticTag/Norway>, <#SemanticTag/London>,
    <#SemanticTag/Greenhouse_gas>, <#SemanticTag/Central_European_Time>,
    <#SemanticTag/Norwegian_Sea,>, <#SemanticTag/Test_cricket>,
    <#SemanticTag/Glenn_L._Martin>, <#SemanticTag/Company>,
    <#SemanticTag/Stavanger,>, <#SemanticTag/RWE,>;
<#Bookmark/https://www.uib.no>
    a schema:Bookmark;
    ex:dateCreated "11/12/2022, 1:16:54 PM";
    ex:dateModified "4/28/2023, 1:26:38 PM";
    ex:alternateName "University of Bergen";
    ex:description "This is the University of Bergens website";
    ex:disambiguationDescription
    <#Tag/Bergen>, <#Tag/Education>, <#Tag/University>;
    ex:URL "https://www.uib.no".
    ex:disambiguationDescription
    <#SemanticTag/Bergen>, <#SemanticTag/Fiji>, <#SemanticTag/Pacific Ocean>,
    <#SemanticTag/Artificial intelligence>, <#SemanticTag/United_Nations>,
    <#SemanticTag/New_York_City,>, <#SemanticTag/Microscope>;

Continued on next page:
```

Listing 4.5: SHACL shape vesion two continued

```
<#Bookmark/https://www.wwf.no>
    a schema:Bookmark;
    ex:dateCreated "1/26/2023, 3:15:36 PM";
    ex:dateModified "4/28/2023, 1:15:44 PM";
    ex:alternateName "World Wide Fund for Nature (WWF)";
    ex:description
        "WWF er en global milj\u00f8organisasjon som kjemper mot v\u00e5r
        tids st\u00f8rste utfordringer: natur- og klimakrisen.";
    ex:disambiguationDescription
    <#Tag/Animals>, <#Tag/Climate-change>, <#Tag/Fund-for-nature>;
    ex:URL "https://www.wwf.no".
    ex:disambiguationDescription
    <#SemanticTag/Glossary_of_professional_wrestling_terms>;

<#SemanticTag/Bergen>, SemanticTag;
<#SemanticTag/Fiji>, SemanticTag;
<#SemanticTag/Pacific Ocean>, SemanticTag;
<#SemanticTag/Artificial intelligence>, SemanticTag;
<#SemanticTag/United_Nations>, SemanticTag;
<#SemanticTag/New_York_City,>, SemanticTag;
<#SemanticTag/Microscope>, SemanticTag;
<#SemanticTag/Equinor>, SemanticTag;
<#SemanticTag/Norway>, SemanticTag;
<#SemanticTag/London>, SemanticTag;
<#SemanticTag/Greenhouse_gas>, SemanticTag;
<#SemanticTag/Central_European_Time>, SemanticTag;
<#SemanticTag/Norwegian_Sea,>, SemanticTag;
<#SemanticTag/Test_cricket>, SemanticTag;
<#SemanticTag/Glenn_L._Martin>, SemanticTag;
<#SemanticTag/Company>, SemanticTag;
<#SemanticTag/Stavanger,>, SemanticTag;
<#SemanticTag/RWE,>, SemanticTag;
<#SemanticTag/Glossary_of_professional_wrestling_terms>, SemanticTag;
<#Tag/Animals> a schema:Tag, :Tag.
<#Tag/Bergen> a schema:Tag, :Tag.
<#Tag/Charity> a schema:Tag, :Tag.
<#Tag/Climate-change> a schema:Tag, :Tag.
<#Tag/Corporation> a schema:Tag, :Tag.
<#Tag/Education> a schema:Tag, :Tag.
<#Tag/Fund-for-nature> a schema:Tag, :Tag.
<#Tag/Human-rights> a schema:Tag, :Tag.
<#Tag/Norwegian> a schema:Tag, :Tag.
<#Tag/Oil> a schema:Tag, :Tag.
<#Tag/University> a schema:Tag, :Tag.
<#Tag/Worldwide> a schema:Tag, :Tag.
```

Chapter 5

Discussion & Findings

In this section, we will review the outcomes of the design science process. This will be accomplished though: re-evaluating the RQs, an evaluation of the artifact and the design science process, and looking at the limitations and weaknesses. Additionally, I will explore potential solutions for this technology's challenges shape trees, and consider future work and expansions that could address these issues.

5.1 Addressing the RQs

In this section we will address the RQs one at a time we will utilize a combination of design science evaluation tools but also include some critical evaluation of the results and possibilities in accordance with what critical design science establishes. As stated in the introduction 1.3, these are the RQs that we are going to focus on for this thesis, first, we will go through them one by one in this section, and then we will do a summary discussion at the end of this chapter.

Research Questions:

- What sort of data shapes do we need for social media?
- How to manage the enforcement of shapes in the pod?
- How can we manage evolving requirements for data in pods?
- Does SOLID PODs enable data reuse and innovation in the application space?

5.1.1 What sort of data shapes do we need for social media?

Here I will discuss the shapes made, how they worked, and what types of requirements I can envision with the use in larger social media. How can we predict what different shapes we need in the future, and for all different social media: we can not! Luckily, we do not have to either, since we can quickly and easily expand shapes. Since most sites use much of the same data (email, name, age), there is an argument to be made for using a common shape that includes most of the data we would need in a profile, then if a specific site has more data they would like to have in a user profile they can either have two shapes or have a subclass shape with the additional requirements. If they were using a system that integrates forms and shapes (either DASH or ShEx), a user might be prompted to fill in the missing data or warned that they have missing data. Utilizing shapes in this way allows for different sites to have their own data requirement, and also allows for the reuse of data. This does require collaboration between creators with the use of, for example, a shape repository. It also requires the shapes not to be closed. In the background 2.9, we looked at some of the particular restrictions one can include in a shape, if one uses *sh:closed* the shape will not validate if it contains any other data then the data required to validate [1].

With the use of either nested shapes or multiple shapes, we can create all the shapes we need for social media, the requirements become apparent as they are required, and some discourse will be needed regarding the central shape repository.

5.1.2 How to manage the enforcement of shapes in the pod?

Validation of the data is built into both SHACL and ShEx, however, the SOLID pods might still be filled with data after much use. In direct contradiction to what I stated in the last segment, one solution is to have many smaller shapes and have them be closed. Build a profile with many smaller shapes, much like the puzzle pieces being used to illustrate shapes. Or to use the gingerbread example from the background 2.7, we could build our gingerbread cookies from smaller cookies. With open or closed shapes there can be issues with enforcement of restrictions placed on shapes, for example, if the shape of the profile has a maximum of two middle names, or one last name, and people with valid names for their culture try to use their name in the profile it might cause an issue [2]. Conversely having no upper limit for certain items might also be an issue for systems to handle. It would probably be prudent to set maximum numbers for some things and numbers, and rather increase the maximum if needed.

5.1.3 How can we manage evolving requirements for data in pods?

Here I discuss the expansion with Spotlight to the shape. There are two main options when it comes to data requirements evolving and shapes. One is to include the new requirements and update the shape repository with the new shape, and then all sites that want to use this expansion will need to add to their environment a form expansion to fill the data gap in conference with the users. Alternatively, a whole new shape could be created and we would be faced with the prospect of having many competing shapes for each use. Lastly, there could be created a subclass of the shape which can be included in the sites that require it. Which approach would be easier depends on the specific data inclusion, if it seems to be a permanent requirement witch many site use, it might be easier to go with the first solution. Regardless of the solution, the updating of shapes based on data requirements is easy, updating the data in pods might be more difficult, with data forms it could be streamlined.

[1] The shapes made in this thesis uses sh:closed for testing purposes, after testing it can easily be removed.

[2] a problem with a minimum required letters in names are people with only one or two letter last or first names, they might not be able to create profiles that require legal names to be used.

The data expansion of my shape in this thesis was very simple, since I was begrudgingly limited to running SHACL locally, I did not manage to implement my script for adding the Spotlight data to the shape directly in the test site LexiTags. Hence the data collection and updating of the data were done manually, as we saw in the development chapter. The results would be the same, it was however disappointing to not be able to integrate the process.

5.1.4 Does SOLID PODs enable data reuse and innovation in the application space?

This is the biggest and most difficult RQ to answer, to make it easier to establish what the answer to this is we are going to first look at the data reuse aspect of SOLID pods and SHACL, second, we will look at the innovation perspective.

Data reuse

Does SOLID pods, and subsequently SHACL enable data reuse, the short answer is "yes but", there are some caveats to this. The goal of interoperability is a hard one to accomplish, the promises from some of shape technology's promoters were exactly that. Shapes do not allow for data to be interoperable quickly, there are several hindrances for it to work. With a more extensive system, like shape trees, there could be sites where they look at the data in your pod, look for shapes in the shape trees, and if they are compatible with the data needed for the said site, no new data needs to be written, and the data can be used based on what is specified in the shape. The caveat here is that I was unable to test this, first, because I was not able to fully integrate shapes into the LexiTags website, and secondly, because the shape tree specification was released in December 2021, after the plan for this thesis was made, and a pivot to include shape trees was not in the cards at that time.

Innovation

From the development process and the expansion of the shape to include the semantic annotations from Spotlight, we can infer that both SHACL and SOLID pods and shapes do allow for innovation, even if the SOLID pods were not used directly. The difficulty here was incompatible technologies and using a site designed to test something different entirely for this purpose. This process has given me great insight into what would be interesting to try with SOLID and shapes, we will discuss this at the end of this chapter, now let us evaluate the artifact and thereafter the design science process.

5.2 Evaluation

The evaluation will be twofold, as discussed in the *Research Methods* by 3. There are many ways to evaluate different systems and projects in design science research. Hence I will do a twofold evaluation, as outlaid in (*Johanson and Williamson*, 2018, Chapter 11). First, we shall evaluate the products of the design science research with reference to the RQs, and then we will evaluate the design science process and how that was done.

5.2.1 Evaluation of the artifact

There are many criteria we could use to evaluate an artifact since we are evaluating an instantiation. I will use the three criteria I discussed in chapter 3. First, why the product will possess desirable characteristics. Second, the reason the artifact leads to desired outcomes. Thirdly, assess design and execution. What will be referred to as the artifact is the shape made in SHACL, the Python script created is only a tool to enable the semantic tags to be created.

Desirable characteristics

Does this shape have desirable characteristics? Definitely, we can use this shape to validate data and know if we have the data we require. If we have more data than required, we could still validate that, the data we have, is there. If this shape is added to the data we could use it to know what kind of shape the data has. These characteristics are desirable, but not be-all and end-all, as discussed the shapes need to do more than just validate, they need to enable more in the development space. In this respect the SHACL shape alone is not enough, perhaps with the addition of shape trees or DASH shapes we could have the desired functionality.

Reasoning for desired outcomes

Here we will discuss two things, one, the reason for the desired outcome, and two, the reason for the desire to be the desire in the first place. The shape was created in Protégé, but a considerable effort was spent to find a desirable and efficient environment to develop in. The development space for shape is evolving, there are new tools released since this thesis was started, I have no doubts that developing now would be easier than it was when I started. This is not to say that there is not room for improvement, for shapes to be utilised more in the public sphere there is a need for easier development tools, more support mechanics and automation.

Assess design and execution

For the purpose of validating the bookmarks in the SOLID pods the shape fulfils the expectations, it checks all the data and checks if the selected parameters validate. The big drawback with this validation is the lack of feedback: if you try to validate an empty object, it validates even if it checks no nodes. The lack of feedback poses issues to a system built to be decentralized. The execution could have been more streamlined, the technology incompatibility described in chapter 4 illustrated what issues there were, for many of them, there was little way to predict, and some were inevitable. This does not speak against the technology as much as it does the difficulty with testing and not building a system intended for this from the ground up. Limitations of this thesis will be discussed after the evaluation of the design science process which follows.

5.2.2 Design science process evaluation

Williamson and Johanson detail the different ways of evaluating the design science process in chapter 3 we looked at the three steps 3.3 to evaluating the process. After that, they say:

> Executing each step involves our making judgements. No hard-and-fast rules are available; no rote process exists on which we can rely. Moreover, we might conclude the successful completion of just one of these steps constitutes a significant contribution to design-science knowledge. (*Johanson and Williamson*, 2018, Chapter 11)

Based on these criteria, what can we evaluate here from the work presented in this thesis?

Evaluating the problem specification

Specificity of the problem, here Williamson and Johanson give us some simple questions to answer, they tell us to focus on these questions: Who? What? Why? When? Where? Stakeholders? (*Johanson and Williamson*, 2018, Chapter 11). In chapter 2 we went over the problem space, but we also discussed who this technology could be useful for, maybe not specifically what individuals, but users of the internet who care about their data privacy. What it could be used for, social media specifically, in addition, shapes technology has wider use cases, development in one area could bring innovation in others. I want to emphasize that the ethical implications of users not being in control of their data is a large concern that has been discussed widely in media, governments and courts. The last of these questions is the most important one for who possibly stands to gain from not having users in control of their data, the corporations currently in control of said data. In history, we have seen many technological revolutions sweep the world with increased speed of adoption, conversely, history is filled with survivor bias, and might not give us the direct guidance we require, something Denning and Lewis discuss in their article:

> Moreover, the models are unreliable when used as ways to organize projects—they explain what happened in the past but offer little guidance on what to do in the immediate future. *Denning and Lewis* (2020)

There are more difficulties in predicting the future; perhaps the biggest one is the opposed interest of the corporations and the users. If there is no change in how social media corporations make a profit, it isn't easy to see adoption as something they are likely to do. The future is not bleak, however, for there are alternatives, currently social media services need to make money to support the high cost of having servers with users' data. If a decentralized social media can have a different business model, with small storage requirements, the interests of companies and users might align on data protection again.

Evaluating the likely contribution to knowledge

From the list of likely situations one might find oneself in when doing design science research, the third one is the one that best approaches what applies to this thesis: "Exaptation: The researcher is adapting known solutions to new problems. (*Johanson and Williamson*, 2018, Chapter 11)" Shape technology already has multiple uses, and so does SOLID; testing the use of shapes in SOLID for social media is adapting a known solution to a new problem. The question now is if anyone finds this approach worth spending the development time for its practical benefits.

Evaluate how a solution to the problem was obtained

When evaluating how the solution to a problem was reached we are looking at the path taken versus the many possible paths to that goal. If we had a birds-eye perspective of the possible paths to the solution here, I predict that the biggest difference would lie in the choice between starting with SHACL or with ShEx. What is better, or how that changed the project is difficult to ascertain, we can however denote some key possibilities.

First, we should evaluate the extent to which the desired end state (goal state) was articulated formally (and thus clearly and precisely).

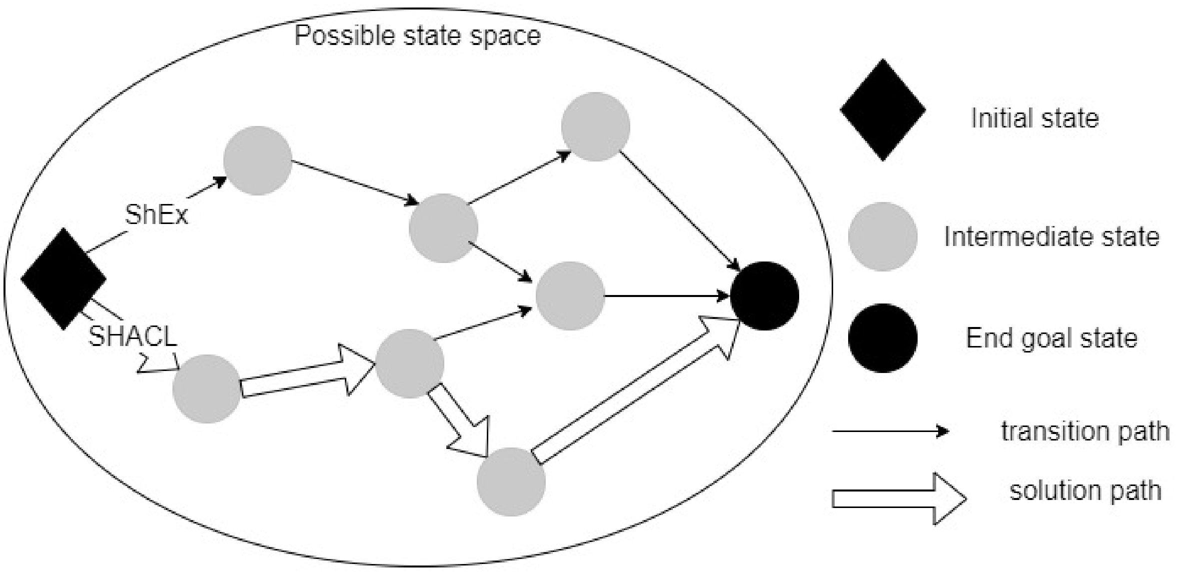


Figure 5.1: Picture showing the possible routes to a solution with emphasis on the first choice

To illustrate the choice I made this figure 5.1 based on a figure in *research methods information systems and contexts* (*Johanson and Williamson*, 2018, FIGURE 11.3).

> Recall that Hevner et al. (2004) argue a design-science research project can produce four types of outcomes: constructs, models, methods, and instantiations.

(*Johanson and Williamson*, 2018, Chapter 11)

5.3 Limitations & weaknesses

There are quite a few things to mention here, first, we will look at the limitations of the thesis, focusing on the problems encountered in development and limitations in testing. Then we will discuss some general weaknesses, especially in relation to the evaluation and results.

5.3.1 Technological limitations

The technological incompatibilities detailed in chapter 4 are a big limitation to the thesis, they caused severe time expenditure on what seemed to be easy implementations. Testing new technologies and new approaches might have unforeseen hurdles (See section 4.4.3 for details) by definition, but the implementation done with improvisation to the testing process can still be argued to have value for the continued development of shapes.

5.3.2 Scope

The scope of the thesis was difficult to accurately determine before the development process started. The scope was well scaled for the thesis, with fewer technical incompatibilities and a quicker start to the development process I surmise that all the goals of the thesis could have been fulfilled or at least explored conclusively.

5.4 Shape trees

Shape trees are an essential inclusion to the shape technology, be it SHACL or ShEx, having the shape or many shapes included in the data would be incredibly useful if one is to be able to use the same data for different sites like envisioned. Since the technology is not fully developed yet, and one would need to use at least two sites to test if shape trees would make the reuse of data easier, it was outside the scope of this thesis. However, one would benefit from exploiting this technology if there is to be a good precedent for reusable data in the future using SOLID. I wanted to include this about shape trees to illustrate the potential for this technology and the grey area that exists between the technology and its broad adoption of it. There are several obstacles in the way of broad use by the private and public spheres. In the next section, we will discuss the broader implication of this technology, what issues there are and what solutions are possible.

5.5 Future works

What are the possible uses for shapes and SOLID pods for social media? Do these technologies find purchase in the scramble for an option to the current system? Or will shape technology offer a new path to existing systems, another possibility is the appearance of another technology outpacing shapes at their own game. The future is impossible to predict, but we can speculate, and in this section, I am going to do some speculating, backed up with as much evidence as I am able to.

5.5.1 Problems

There are many problems facing shapes and SOLID, one of the biggest problems is the need for these systems to supplant an existing structure. When the Internet first bloomed and exploded there was nothing like it, there was radio and television, but nothing interactive and with the same possibilities of the Internet. SOLID faces an established infrastructure with tremendous support from the monetization systems involved. I suspect the likelihood of industry giants like Facebook and Google allowing users to use their services using SOLID without government insistence is close to absolute zero. With data being the central element in Internet advertisement, and also valuable on its own, there is no incentive for established corporations to accede to SOLID being used. There are other problems like the development space lacking tools and testing spaces. If shapes are to become widely used there is a need for better development tools and applications, this will come from broader adoption, but to expedite and allow for said adoption better tools might be essential. Developers like water will, mostly, travel the path of least resistance, therefore it is central to make shapes an easier path to travel. During the research for this thesis, I read untold issues and problems faced by developers, some I document in this thesis, others I do not. The development space is filled with good attempts and good intentions, without proper standards and tools however those will fade.

There are challenges for decentralized SOLID systems as discussed in Bruland's thesis *Bruland* (2022), perhaps large-scale decentralized systems will be out of reach for some years, perhaps faster processing, higher Internet speeds, or new technology will facilitate it.

5.5.2 Possibilities

There are definitely possibilities to create good systems with SOLID, as is proven by the systems that already exist 2.2.3. Shapes, or systems very much like it, are likely needed to enable decentralization. However, processing-speed and the validation process are issues that still need solving. These issues can be overcome, either through the development of tools or new technologies. In the last section, I deliberate on the disinclination of corporations to allow for the inclusion of SOLID solutions into their systems, all is not lost, however. Like the Internet of old and the resurgence of new sites after the dot.com bubble, now too there could be a rejuvenation. New applications and sites can be replacements for old, and like the phoenix rising anew, there can be a new dawn of privacy protection online. There are possibilities in any new development sphere, and it is not in the scope of this thesis to analyze the success chance for SOLID, however working with SOLID and reading the ideas of the developers I cannot dismiss the thought of how much we could benefit from this course correction.

Chapter 6

Conclusions and Future Work

6.1 Conclusions

In this chapter, we shall conclude with the results of the thesis and walk through the research questions and goals of the thesis again with what has been learned in the thesis.

6.1.1 Summary

We have discussed the technological landscape that SOLID, SHACL, and ShEx are central to and interact with in chapter 2. The methodology chapter 3 brought up design science research and critical theory, and introduced some ideas of how and why it is useful to have a hybrid methodology. We discussed the ethical considerations that prompt the use of a critical design theory over traditional approaches. In chapter 4 the project development, research and testing were presented, and the initial choice of SHACL began the development process. The first shapes created in Protégé with placeholder data made way for the more advanced shapes. Then the inclusion of the DBpedia data expanded the shape, and the data collection was mapped out and recorded. In chapter 5 we discussed the results of the development process and the implications of the results. There was also a section devoted to exploring the potential and pitfalls of both SOLID and shape technology. We will now go over the goals set for this thesis and discuss them individually.

6.1.2 Goals

The objectives for this thesis are focused on shapes and exploring what they are and the use case for shapes with changing data requirements.

- Investigate SOLID pods data storage
- Investigate shapes technology
- Create shapes that match the bookmarks of our site
- Change the shapes and data
- Introduce semantic meaning to the bookmarks with DBpedia

What was achieved in this thesis and what are the big points we can draw from the discussion, and did we discover the answer to any of these questions?

As far as the goals for the thesis are concerned much was accomplished, we can review them one by one. I did investigate SOLID pod data storage and shape technology. I did create a shape that matched the bookmark for the site, I did expand on the shape and included the semantic tags which also were the semantic meaning. There were however some hurdles and failings, I did not manage to test and validate the shapes directly against the SOLID pod data, and I did not manage to integrate the DBpedia data into the LexiTags website. There is definitely room for continued research and testing. Developing alternative systems to existing monolithic standards will take time. It would have been good to be able to test both SHACL and ShEx in this project and have more direct interaction with the SOLID site built for this project, as for now it just showed how far shapes have to go, and the issues pertaining to validation and re-usability of data.

6.1.3 Conclusion

Shapes for social media have a possible future though, many difficulties will have to be surmounted. The technology is promising and the activity in the field shows the need for such a system. Will shapes be the enabler for social media using SOLID, perhaps? They are strong contenders to solve the issues created by decentralizing interactive networks. They will have to solve problems traditional sites do not have and develop solutions capable of connecting all these purpose-built technologies into something they are not meant for. Issues arise because these technologies are intended to be used with servers and a central backend, SOLID requires the opposite, I.E. a system built to operate with no central storage and without a server, in the traditional sense. It might be necessary to develop new systems to handle SOLID-based systems without a central server for the code. Alternatively, we will have to restructure some of these systems or create new systems.

6.1.4 Moving forwards

The future for SOLID can deviate from the future of shapes, SOLID is being developed by Inrupt and has a promising future, even if it is filled with animosity, if not opposition, from those companies who stand to lose control. Shapes on the other hand do not have the same clear path as SOLID, data validation is useful, but shapes need to do more than validate data. There is a need for shapes to expand and include other functionality, and they have gotten some further development. We have some sense of the functionality needed such as form generation and shape trees, however, the list of the required functionality is not complete and the development of these technologies is still ongoing, as such the list might change in the future.

Furthermore, as the needs continue to grow other solutions might arise. We have at length discussed SHACL and ShEx as two languages that compete and in many ways try to do the same, but in other aspects differ in their approach. Here it is easy to fall into the classic "we need to develop a new standard to encompass all use cases" [1],

[1] see figure A.1 for an illustration of this

shapes have a possible future of playing a central role in making decentralized websites operate. Which standard will be the choice for future development, or if they will both play their separate and distinct roles, is impossible to predict, perhaps they will both influence a third solution.

6.2 Additional takeaways

One of the most significant issues with using SHACL and developing shapes was the poor feedback in the development stage, this was also commented on by *Garcia et al.* (2019) when they were developing ShEx shapes how difficult it actually is to write shapes, and test them. They propose a ShEx creator, but it seems like a prototype was created and never touched again.

Having a development space for shapes would make the process more available to newcomers, I used protege, which after trying to develop on SHACL playground [2] without much progress. Using Protégé worked, but it was cumbersome, Protégé has support for SHACL but the development process is not very dynamic. The process involves saving and loading different shape documents, and creating a test graph with different conditions to accurately ascertain if the shape is not allowing for edge cases and is testing for what you think it is. Having a development space created for shape development would be a great tool needed for the adoption of shapes in broader use. A tool with support for both SHACL and ShEx would be ideal and should be supported by both camps. Additionally, if SHACL had a testing mode where you could get a printout of what shapes validated what parameters, that would be supremely helpful in testing, a validation report stating validated even if nothing is checked makes for frustrating workflows.

For future development, I would suggest the following project detailed in this section.
I would propose to develop a system that used shapes to not only validate data but manipulates it. Furthermore, the design of prototype websites as a model of how this system would work. This model could check a pod to see if it has a shape tree for either the data the site requires or other shapes that contain some of the data required. Possibly two or three sites where the data is created for the first, then used and expanded in the second. If the data in the pod contains parts of the data we need, we only require some additional data. Lastly for the third, we use some from the first and some from the second.

Lastly, we store the data in the same place and then add our shape to the shape tree. There might be a need to build a system that reads the SOLID pod, looks for a shape tree and compares the shapes in it to the shapes allowed. This would entail extensive development. It would, however, be very indicative of the possibilities of shapes and by extension, re-imagine how social media sites operate.

The intention of this research has been to provide a stepping stone in the right direction. There is a clear need for alternatives to the current system, as detailed in this thesis 2.3. By researching solutions that enable users to control their own data I hope to have contributed to the advancements in this field.

[2] https://shacl.org/playground/

Appendix A

Appendix

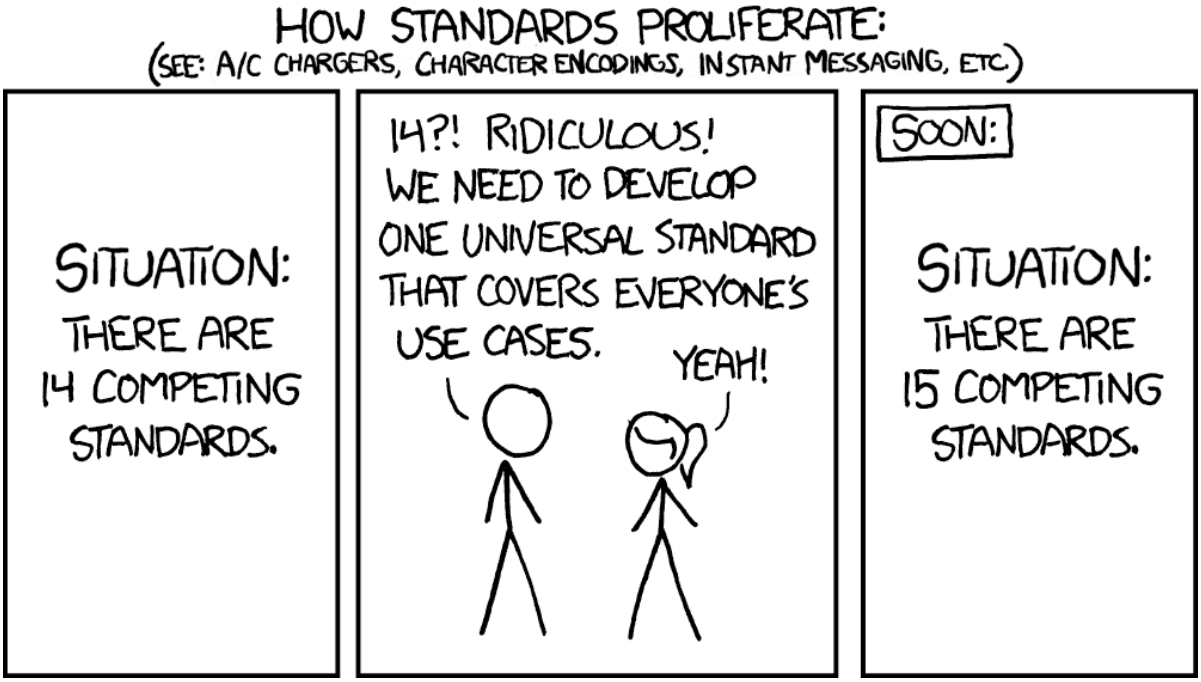


Figure A.1: xkcd comic about standards Munroe (2011)

A.1 Shapes and RDF

Should add the original and also the updated one?

A.1.1 First SHACL shape

```
@prefix rdf:   <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix sh:    <http://www.w3.org/ns/shacl#> .
@prefix xsd:   <http://www.w3.org/2001/XMLSchema#> .
@prefix rdfs:  <http://www.w3.org/2000/01/rdf-schema#> .
@prefix ex:    <http://www.example.org/#> .
@prefix owl:   <http://www.w3.org/2002/07/owl#> .

ex:BookmarkShape
    a sh:NodeShape ;
    sh:targetClass ex:Bookmark ;  # Applies to all persons
    sh:property [                 # _:b0
        sh:path ex:hasURL ;       # constrains the values of ex:hasURL
        sh:minCount 1 ;
        sh:maxCount 1 ;
    ] ;
    sh:property [              # _:b1
        sh:path ex:hasURL ;           # constrains the values of ex:hasURL
        sh:datatype xsd:string ;
        sh:severity sh:Warning ;
    ] ;
    sh:property [                 # _:b0
        sh:path ex:dateCreated ;       # constrains the values of ex:dateCreated
        sh:minCount 1 ;
        sh:maxCount 1 ;
    ] ;
    sh:property [               # _:b1
        sh:path ex:dateCreated ;        # constrains the values of ex:dateCreated
        sh:datatype xsd:dateTime ;
        sh:severity sh:Warning ;
    ] ;
    sh:property [                # _:b0
        sh:path ex:hasDescription ;   # constrains the values of ex:hasDescription
        sh:minCount 1 ;
        sh:maxCount 1 ;
    ] ;
    sh:property [                  # _:b1
        sh:path ex:hasDescription ;  # constrains the values of ex:hasDescription
        sh:datatype xsd:string ;
        sh:severity sh:Warning ;
    ] ;
```

```
    sh:property [                # _:b0
        sh:path ex:TagString ;         # constrains the values of ex:TagString
    ] ;
    sh:property [                 # _:b1
        sh:path ex:TagString ;       # constrains the values of ex:TagString
        sh:datatype xsd:string ;
        sh:severity sh:Warning ;
    ] ;
    sh:property [                # _:b0
        sh:path ex:title ;          # constrains the values of ex:title
        sh:minCount 1 ;
        sh:maxCount 1 ;
    ] ;
    sh:property [                         # _:b1
        sh:path ex:title ;          # constrains the values of ex:title
        sh:datatype xsd:string ;
        sh:severity sh:Warning ;
    ] ;

    sh:closed true ;
    sh:ignoredProperties ( rdf:type owl:topDataProperty owl:topObjectProperty ) ;
    .
 ###### add SHACL vocabulary ######
```

A.1.2 Updated SHACL shape

```
 this is the updated shape with the correct names from the bookmark in LexiTags:
@prefix rdf:   <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix sh:    <http://www.w3.org/ns/shacl#> .
@prefix xsd:   <http://www.w3.org/2001/XMLSchema#> .
@prefix rdfs:  <http://www.w3.org/2000/01/rdf-schema#> .
@prefix ex:    <http://www.example.org/#> .
@prefix owl:   <http://www.w3.org/2002/07/owl#> .

ex:BookmarkShape
    a sh:NodeShape ;
    sh:targetClass ex:Bookmark ;  # Applies to all Bookmarks
    sh:property [                 # _:b0
        sh:path ex:URL ;          # constrains the values of ex:hasURL
        sh:minCount 1 ;
        sh:maxCount 1 ;
    ] ;
    sh:property [                  # _:b1
        sh:path ex:URL ;            # constrains the values of ex:hasURL
        sh:datatype xsd:string ;
        sh:severity sh:Warning ;
    ] ;
```

```
    sh:property [                  # _:b0
        sh:path ex:dateCreated ;     # constrains the values of ex:dateCreated
        sh:minCount 1 ;
        sh:maxCount 1 ;
    ] ;
    sh:property [                # _:b1
        sh:path ex:dateCreated ;      # constrains the values of ex:dateCreated
        sh:datatype xsd:dateTime ;
        sh:severity sh:Warning ;
    ] ;
    sh:property [                 # _:b0
        sh:path ex:Description ;    # constrains the values of ex:hasDescription
        sh:minCount 1 ;
        sh:maxCount 1 ;
    ] ;
    sh:property [                   # _:b1
        sh:path ex:Description ;   # constrains the values of ex:hasDescription
        sh:datatype xsd:string ;
        sh:severity sh:Warning ;
    ] ;
    sh:property [                  # _:b0
        sh:path ex:alternateName ;   # constrains the values of ex:alternateName
        sh:minCount 1 ;
        sh:maxCount 1 ;
    ] ;
    sh:property [                  # _:b1
        sh:path ex:alternateName ;   # constrains the values of ex:alternateName
        sh:datatype xsd:string ;
        sh:severity sh:Warning ;
    ] ;
    sh:property [                 # _:b0
        sh:path ex:disambiguationDescription ;
        # constrains the values of ex:disambiguationDescription
    ] ;
    sh:property [                         # _:b1
        sh:path ex:disambiguationDescription ;
        # constrains the values of ex:disambiguationDescription
        sh:objecttype xsd:Tag ;
        sh:severity sh:Warning ;
    ] ;

    sh:closed true ;
    sh:ignoredProperties ( rdf:type owl:topDataProperty owl:topObjectProperty ) ;
    .
 ###### add SHACL vocabulary ######
```

A.1.3 SHACL shape with updated names

```
@prefix rdf:   <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix sh:    <http://www.w3.org/ns/shacl#> .
@prefix xsd:   <http://www.w3.org/2001/XMLSchema#> .
@prefix rdfs:  <http://www.w3.org/2000/01/rdf-schema#> .
@prefix ex:    <http://www.example.org/#> .
@prefix owl:   <http://www.w3.org/2002/07/owl#> .

ex:BookmarkShape
    a sh:NodeShape ;
    sh:targetClass ex:Bookmark ;  # Applies to all persons
    sh:property [                 # _:b0
        sh:path ex:URL ;       # constrains the values of ex:URL
        sh:minCount 1 ;
        sh:maxCount 1 ;
    ] ;
    sh:property [              # _:b1
        sh:path ex:URL ;           # constrains the values of ex:URL
        sh:datatype xsd:string ;
        sh:severity sh:Warning ;
    ] ;
    sh:property [                 # _:b0
        sh:path ex:dateCreated ;       # constrains the values of ex:dateCreated
        sh:minCount 1 ;
        sh:maxCount 1 ;
    ] ;
    sh:property [               # _:b1
        sh:path ex:dateCreated ;        # constrains the values of ex:dateCreated
        sh:datatype xsd:dateTime ;
        sh:severity sh:Warning ;
    ] ;
    sh:property [                   # _:b0
        sh:path ex:Description ;     # constrains the values of ex:Description
        sh:minCount 1 ;
        sh:maxCount 1 ;
    ] ;
    sh:property [                   # _:b1
        sh:path ex:Description ;     # constrains the values of ex:Description
        sh:datatype xsd:string ;
        sh:severity sh:Warning ;
    ] ;
    sh:property [                  # _:b0
        sh:path ex:TagString ;     # constrains the values of ex:TagString
    ] ;
    sh:property [                  # _:b1
        sh:path ex:TagString ;     # constrains the values of ex:TagString
```

```
        sh:datatype xsd:string ;
        sh:severity sh:Warning ;
    ] ;
    sh:property [                # _:b0
        sh:path ex:alternateName ;    # constrains the values of ex:alternateName
        sh:minCount 1 ;
        sh:maxCount 1 ;
    ] ;
    sh:property [                 # _:b1
        sh:path ex:alternateName ;   # constrains the values of ex:alternateName
        sh:datatype xsd:string ;
        sh:severity sh:Warning ;
    ] ;

    sh:closed true ;
    sh:ignoredProperties ( rdf:type owl:topDataProperty owl:topObjectProperty ) ;
    .
 ###### add SHACL vocabulary ######
```

A.1.4 SHACL shape with Tag Class

```
@prefix : <#>.
@prefix schema: <http://schema.org/>.
@prefix lexit: <./>.
@prefix www: <http://www.schema.org/>.
@prefix Thing: <http://schema.org/Thing/>.
@base <http://www.example.org/#> .
@prefix rdf:   <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix sh:    <http://www.w3.org/ns/shacl#> .
@prefix xsd:   <http://www.w3.org/2001/XMLSchema#> .
@prefix rdfs:  <http://www.w3.org/2000/01/rdf-schema#> .
@prefix ex:    <http://www.example.org/#> .
@prefix owl:   <http://www.w3.org/2002/07/owl#> .


ex:BookmarkShape
    a sh:NodeShape ;
    sh:targetClass ex:Bookmark ;  # Applies to all Bookmarks

    sh:property [
sh:path ex:disambiguationDescription ;
sh:class ex:Tag;
    ];

    sh:property [                 # _:b0
        sh:path ex:URL ;       # constrains the values of ex:URL
        sh:minCount 1 ;
```

```
        sh:maxCount 1 ;
    ] ;
    sh:property [              # _:b1
        sh:path ex:URL ;           # constrains the values of ex:URL
        sh:datatype xsd:string ;
        sh:severity sh:Warning ;
    ] ;
    sh:property [                 # _:b0
        sh:path ex:dateCreated ;       # constrains the values of ex:dateCreated
        sh:minCount 1 ;
        sh:maxCount 1 ;
    ] ;
    sh:property [               # _:b1
        sh:path ex:dateCreated ;        # constrains the values of ex:dateCreated
        sh:datatype xsd:dateTime ;
        sh:severity sh:Warning ;
    ] ;
    sh:property [                # _:b0
        sh:path ex:description ;    # constrains the values of ex:description
        sh:minCount 1 ;
        sh:maxCount 1 ;
    ] ;
    sh:property [                  # _:b1
        sh:path ex:description ;     # constrains the values of ex:description
        sh:datatype xsd:string ;
        sh:severity sh:Warning ;
    ] ;
    sh:property [                # _:b0
        sh:path ex:alternateName ;   # constrains the values of ex:title
        sh:minCount 1 ;
        sh:maxCount 1 ;
    ] ;
    sh:property [                   # _:b1
        sh:path ex:alternateName ;      # constrains the values of ex:title
        sh:datatype xsd:string ;
        sh:severity sh:Warning ;

];


.
ex:TagShape
    a sh:NodeShape ;
    sh:targetClass ex:Tag;
    sh:property [                # _:b0
        sh:path ex:NamedIndividual;          # constrains the values of ex:Tag
    ] ;
```

```
    sh:property [                             # _:b1
        sh:path ex:NamedIndividual  ;        # constrains the values of ex:Tag
        sh:TargetClass xsd:NamedIndividual;
        sh:severity sh:Warning ;
    ] ;



    sh:closed true;
    sh:ignoredProperties ( rdf:type owl:topDataProperty owl:topObjectProperty ) ;
    .
 ###### add SHACL vocabulary ######
```

A.1.5 Added SHACL for semantic tags

```
    sh:property [                   # _:b0
        sh:path ex:disambiguationDescription ;
        # constrains the values of ex:SemanticTag
    ] ;
    sh:property [                           # _:b1
        sh:path ex:disambiguationDescription ;
        # constrains the values of ex:SemanticTag
        sh:objecttype xsd:SemanticTag ;
        sh:severity sh:Warning ;
    ] ;
```

A.1.6 RDF data

```
@prefix : <http://www.example.org/#> .
@prefix owl: <http://www.w3.org/2002/07/owl#> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix xml: <http://www.w3.org/XML/1998/namespace> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@base <http://www.example.org/#> .
@prefix : <#>.
@prefix schema: <http://schema.org/>.
@prefix lexit: <./>.
@prefix www: <http://www.schema.org/>.
@prefix Thing: <http://schema.org/Thing/>.

<http://www.example.org/#> rdf:type owl:Ontology .

#################################################################
#    Object Properties
#################################################################
```

```
###  http://www.example.org/#disambiguationDescription
:disambiguationDescription rdf:type owl:ObjectProperty ;
                           rdfs:domain :Bookmark ;
                           rdfs:range :Tag .


#################################################################
#    Data properties
#################################################################

###  http://www.example.org/#Description
:Description rdf:type owl:DatatypeProperty ;
             rdfs:domain :Bookmark ;
             rdfs:range xsd:string .


###  http://www.example.org/#URL
:URL rdf:type owl:DatatypeProperty ;
     rdfs:domain :Bookmark ;
     rdfs:range xsd:string .


###  http://www.example.org/#alternateName
:alternateName rdf:type owl:DatatypeProperty ;
               rdfs:domain :Bookmark ;
               rdfs:range xsd:string .


###  http://www.example.org/#dateCreated
:dateCreated rdf:type owl:DatatypeProperty ;
             rdfs:domain :Bookmark ;
             rdfs:range xsd:dateTime .


#################################################################
#    Classes
#################################################################

###  http://www.example.org/#Bookmark
:Bookmark rdf:type owl:Class .


###  http://www.example.org/#Tag
:Tag rdf:type owl:Class .
```

```
#################################################################
#    Individuals
#################################################################


###  http://www.example.org/#google
:google rdf:type owl:NamedIndividual ,
                 :Bookmark ;
        :disambiguationDescription :streaming ,
                                   :video ;
        :Description "is the biggest searchengine"^^xsd:string ;
        :URL "https://www.google.com"^^xsd:string ;
        :alternateName "Google Search engine"^^xsd:string ;
        :dateCreated "2001-10-26T21:32:52+02:00"^^xsd:dateTime .


###  http://www.example.org/#searchengine
:searchengine rdf:type owl:NamedIndividual ,
                       :Tag .


<#Bookmark/https://ba.no>
    a schema:Bookmark, :Bookmark;
    :dateCreated "2001-10-26T21:32:52+02:00"^^xsd:dateTime;
    :alternateName "Bergens Avisen";
    :Description "Local newspaper from Bergen";
    :disambiguationDescription
    <#Tag/Bergen>, <#Tag/Local-newspaper>, <#Tag/News>;
    :URL "https://ba.no".

###  http://www.example.org/#streaming
:streaming rdf:type owl:NamedIndividual ,
                    :Tag .


###  http://www.example.org/#video
:video rdf:type owl:NamedIndividual ,
                :Tag .


###  http://www.example.org/#website
:website rdf:type owl:NamedIndividual ,
                  :Tag .
###  http://www.example.org/#youtube
:youtube rdf:type owl:NamedIndividual ,
                  :Bookmark ;
         :Description "a site for hosting video, and steaming"^^xsd:string ;
```

```
        :URL "https://www.youtube.com"^^xsd:string ;
        :alternateName "youtube.com"^^xsd:string ;
        :dateCreated "2001-10-26T19:32:52+00:00"^^xsd:dateTime .

###  Generated by the OWL API (version 4.5.9.2019-02-01T07:24:44Z) https://github.com/
```

A.1.7 RDF data unedited

```
@prefix : <#>.
@prefix schema: <http://schema.org/>.
@prefix lexit: <./>.
@prefix www: <http://www.schema.org/>.
@prefix Thing: <http://schema.org/Thing/>.
@prefix : <http://www.example.org/#> .
@prefix owl: <http://www.w3.org/2002/07/owl#> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix xml: <http://www.w3.org/XML/1998/namespace> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@base <http://www.example.org/#> .

###  http://www.example.org/#Bookmark
#### <Bookmark> rdf:type owl:Class .

lexit:bookmarks
    a schema:Dataset;
    www:Contains
        <#Bookmark/https://ba.no>, <#Bookmark/https://bt.no>,
        <#Bookmark/https://next-episode.net/the-boys>,
        <#Bookmark/https://next-episode.net/the-walking-dead>,
        <#Bookmark/https://tv2.no>.
<#Bookmark/https://ba.no>
    a schema:Bookmark, :Bookmark;
    schema:dateCreated "2001-10-26T21:32:52+02:00"^^xsd:dateTime; ###"1/31/2022, 9:15
    Thing:alternateName "Bergens Avisen";
    Thing:description "Local newspaper from Bergen";
    Thing:disambiguationDescription
    <#Tag/Bergen>, <#Tag/Local-newspaper>, <#Tag/News>;
    Thing:URL "https://ba.no".
<#Bookmark/https://bt.no>
    a schema:Bookmark;
    schema:dateCreated "1/31/2022, 9:15:19 AM";
    Thing:alternateName "Bergens Tidende";
    Thing:description "Another local newspaper from Bergen";
    Thing:disambiguationDescription <#Tag/Bergen>, <#Tag/News>, <#Tag/Norway>;
    Thing:url "https://bt.no".
```

```
<#Bookmark/https://next-episode.net/the-boys>
    a schema:Bookmark;
    schema:dateCreated "1/31/2022, 9:18:11 AM";
    schema:dateModified "1/31/2022, 9:19:06 AM";
    Thing:alternateName "next episode tracker - The boys";
    Thing:description "online series episode tracker";
    Thing:disambiguationDescription
    <#Tag/Episodes>, <#Tag/Next-episode>, <#Tag/Series>, <#Tag/Tracker>;
    Thing:url "https://next-episode.net/the-boys".
<#Bookmark/https://next-episode.net/the-walking-dead>
    a schema:Bookmark;
    schema:dateCreated "1/31/2022, 9:19:32 AM";
    schema:dateModified "1/31/2022, 10:25:01 AM";
    Thing:alternateName "next episode tracker - the walking deadasd";
    Thing:description "";
    Thing:disambiguationDescription
    <#Tag/Next-episode>, <#Tag/Series>, <#Tag/Tracker>;
    Thing:url "https://next-episode.net/the-walking-dead".
<#Bookmark/https://tv2.no>
    a schema:Bookmark;
    schema:dateCreated "1/31/2022, 9:16:38 AM";
    Thing:alternateName "TV2 News";
    Thing:description "News from Norway";
    Thing:disambiguationDescription <#Tag/News>, <#Tag/Norway>;
    Thing:url "https://tv2.no".
<#Tag/Bergen> a schema:Tag; Thing:Meaning "wordnet"; Thing:Word "Bergen".

<#Tag/Episodes> a schema:Tag; Thing:Meaning "wordnet"; Thing:Word "Episodes".

<#Tag/Local-newspaper>
a schema:Tag; Thing:Meaning "wordnet"; Thing:Word "Local-newspaper".
<#Tag/News> a schema:Tag; Thing:Meaning "wordnet"; Thing:Word "News".

<#Tag/Next-episode>
a schema:Tag; Thing:Meaning "wordnet"; Thing:Word "Next-episode".
<#Tag/Norway> a schema:Tag; Thing:Meaning "wordnet"; Thing:Word "Norway".

<#Tag/Series> a schema:Tag; Thing:Meaning "wordnet"; Thing:Word "Series".

<#Tag/Tracker> a schema:Tag; Thing:Meaning "wordnet"; Thing:Word "Tracker".
```

A.1.8 Protoge Data Graph

```
@prefix : <http://www.example.org/#> .
@prefix owl: <http://www.w3.org/2002/07/owl#> .
```

```
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix xml: <http://www.w3.org/XML/1998/namespace> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@base <http://www.example.org/#> .

<http://www.example.org/#> rdf:type owl:Ontology .

#################################################################
#    Data properties
#################################################################

###  http://www.example.org/#TagString
:TagString rdf:type owl:DatatypeProperty ;
           rdfs:domain :Bookmark ,
                       :Tag ;
           rdfs:range xsd:string .


###  http://www.example.org/#dateCreated
:dateCreated rdf:type owl:DatatypeProperty ;
             rdfs:domain :Bookmark ;
             rdfs:range xsd:dateTime .


###  http://www.example.org/#hasDescription
:hasDescription rdf:type owl:DatatypeProperty ;
                rdfs:domain :Bookmark ;
                rdfs:range xsd:string .


###  http://www.example.org/#hasURL
:hasURL rdf:type owl:DatatypeProperty ;
        rdfs:domain :Bookmark ;
        rdfs:range xsd:string .


###  http://www.example.org/#title
:title rdf:type owl:DatatypeProperty ;
       rdfs:domain :Bookmark ;
       rdfs:range xsd:string .


#################################################################
#    Classes
#################################################################
```

```
###  http://www.example.org/#Bookmark
:Bookmark rdf:type owl:Class .


###  http://www.example.org/#Tag
:Tag rdf:type owl:Class .


#################################################################
#    Individuals
#################################################################

###  http://www.example.org/#empty
:empty rdf:type owl:NamedIndividual ,
                :Bookmark .


###  http://www.example.org/#google
:google rdf:type owl:NamedIndividual ,
                 :Bookmark ;
        :TagString "internetpowerhouse"^^xsd:string ,
                   "search"^^xsd:string ;
        :dateCreated "2001-10-26T21:32:52+02:00"^^xsd:dateTime ;
        :hasDescription "is a search engine"^^xsd:string ;
        :hasURL "https://www.google.com"^^xsd:string ;
        :title "google"^^xsd:string .


###  http://www.example.org/#youtube
:youtube rdf:type owl:NamedIndividual ,
                  :Bookmark ;
         :dateCreated "2001-10-26T19:32:52+00:00"^^xsd:dateTime ,
                      "2001-10-26T19:32:52+00:01"^^xsd:dateTime ;
         :hasDescription "description2"^^xsd:string ,
                         "is a video and streaming website"^^xsd:string ;
         :hasURL "https://www.youtube.com"^^xsd:string ,
                 "https://www.youtube2.com"^^xsd:string ;
         :title "youtube"^^xsd:string ,
                "youtube2"^^xsd:string .


###  Generated by the OWL API (version 4.5.9.2019-02-01T07:24:44Z)
https://github.com/owlcs/owlapi
```

A.2 Spotlight Python script

```
import requests
import bs4
import spotlight ## NPM i PYSPOTLIGHT
import sys

## ADDITIONAL REQUIREMENTS ARE: LMXL

def main():
    ## TAKES INPUT
    ## url = input("Enter or Paste the URL: ")
    url = sys.argv[1:]
    url = '␣'.join(url)
    url.strip()

    try:
        ## GETTING THE URL'S TEXT
        res=requests.get(url)
        souped=bs4.BeautifulSoup(res.text,'lxml')
        print(res.status_code)


        ## FILTERING THE TEXT FROM WEBSITE
        for a in souped(["script","style","link","img","nav","footer","title","meta"]):
            a.decompose()

        text = souped.get_text()
        lines = (line.strip() for line in text.splitlines())
        chunks = (phrase.strip() for line in lines for phrase in line.split("␣␣"))
        txtcontent = '\n'.join(chunk for chunk in chunks if chunk)

        ## ANNOTATION OF THE TEXT
        annotations = spotlight.annotate('https://api.dbpedia-spotlight.org/en/annotate',
                                         txtcontent,confidence=0.5,support=400)

        ## HERE THE ANNOTATIONS ARE FILTERD, CHANGE "URI" AND GET OTHER ITEMS
        annotationsUpdated = []
        for dict_item in annotations:
                item = (dict_item['URI'])
                annotationsUpdated.append(item)

        annotationsUpdated = list(dict.fromkeys(annotationsUpdated))

        ## THESE ARE FOR TESTING IN PYTHON AND NOT FOR ACCTUAL USE
        ## print(annotations)
        print(annotationsUpdated)

        return(annotationsUpdated)

    except:
        print("Oops!␣An␣error␣has␣occurred␣or␣maybe␣you␣just␣entered␣an
␣␣␣␣␣␣␣␣Invalid␣URL␣:D␣Or␣there␣are␣zero␣tags,␣check␣html␣code")
        return ("Oops!␣An␣error␣has␣occurred␣or␣maybe␣you␣just␣entered␣an
␣␣␣␣␣␣␣␣Invalid␣URL␣:D␣Or␣there␣are␣zero␣tags,␣check␣html␣code")

main()
```


```
import requests
import bs4
import spotlight ## NPM i PYSPOTLIGHT
import sys

## ADDITIONAL REQUIREMENTS ARE: LMXL

def main():
    ## TAKES INPUT
    ## url = input("Enter or Paste the URL: ")
    url = sys.argv[1:]
    url = ' '.join(url)
    url.strip()

    try:
        ## GETTING THE URL'S TEXT
        res=requests.get(url)
        souped=bs4.BeautifulSoup(res.text,'lxml')
        print(res.status_code)
```

```
        ## FILTERING THE TEXT FROM WEBSITE
        for a in souped(["script","style","link","img","nav","footer","title","meta"]
            a.decompose()

        text = souped.get_text()
        lines = (line.strip() for line in text.splitlines())
        chunks = (phrase.strip() for line in lines for phrase in line.split("  "))
        txtcontent = '\n'.join(chunk for chunk in chunks if chunk)

        ## ANNOTATION OF THE TEXT
        annotations = spotlight.annotate('https://api.dbpedia-spotlight.org/en/annota
                                    txtcontent,confidence=0.5,support=400)

        ## HERE THE ANNOTATIONS ARE FILTERD, CHANGE "URI" AND GET OTHER ITEMS
        annotationsUpdated = []
        for dict_item in annotations:
                item = (dict_item['URI'])
                annotationsUpdated.append(item)

        annotationsUpdated = list(dict.fromkeys(annotationsUpdated))

        ## THESE ARE FOR TESTING IN PYTHON AND NOT FOR ACCTUAL USE
        ## print(annotations)
        print(annotationsUpdated)

        return(annotationsUpdated)

    except:
        print("Oops! An error has occurred or maybe you just entered an
        Invalid URL :D Or there are zero tags, check html code")
        return ("Oops! An error has occurred or maybe you just entered an
        Invalid URL :D Or there are zero tags, check html code")

main()
```

Bibliography

Abdul Rehman, A., and K. Alharthi (2016), An introduction to research paradigms, *International Journal of Educational Investigation*, *3*. 3.2.1

Agogo, D. (2020), Invisible market for online personal data: An examination, *Electronic Markets*, *31*(4), 989–1010, doi:10.1007/s12525-020-00437-0. 1

Atherton, G. (2023), How solid's "privacy by design" could replace third-party cookies: Blog, https://www.merkle.com/blog/how-solids-privacy-design-could-replace-third-party-cookies. 2.2.3

Berners-Lee, t. (), Design issues for the world wide web, https://www.w3.org/DesignIssues/. 4.3

Biswas, S., K. Sharif, F. Li, Z. Latif, S. S. Kanhere, and S. P. Mohanty (2020), Interoperability and synchronization management of blockchain-based decentralized e-health systems, *IEEE Transactions on Engineering Management*, *67*(4), 1363–1376. 2.4

Bohman, J., J. Flynn, and R. Celikates (2021), Critical Theory, in *The Stanford Encyclopedia of Philosophy*, edited by E. N. Zalta, Spring 2021 ed., Metaphysics Research Lab, Stanford University. 3.1

Bruland, M. (2022), Evaluating solid for social applications with many users, https://bora.uib.no/bora-xmlui/bitstream/handle/11250/3002311/Masteroppgave_Martin_Bruland.pdf?sequence=1. 4.4, 5.5.1

Cadwalladr, C., and E. Graham-Harrison (2018), Revealed: 50 million facebook profiles harvested for cambridge analytica in major data breach. 1.1.1

Charlton, J. (2002), *The Military Quotation Book: More Than 1,200 of the Best Quotations about War, Leadership, Courage, Victory, and Defeat*, Macmillan. 0.1

DavidBooth, PhilA, Anamitra, and TimCole (2013), Rdf validation workshop, https://www.w3.org/2013/09/11-rdfval-minutes. 2.7

Denning, P. J., and T. G. Lewis (2020), Technology adoption, *Communications of the ACM*, *63*(6), 27–29. 5.2.2

for Lifecycle Collaboration, O. S. (2022), https://open-services.net/. 2.7

Fuller, R. B. (1957), *A Comprehensive Anticipatory Design Science*, Royal Architectural Institute of Canada. 3.3

Garcia, L., J. Bolleman, M. Dumontier, S. Jupp, J. E. L. Gayo, T. Liener, T. Ohta, N. Queralt-Rosinach, and C. Wu (2019), Data validation and schema interoperability, *BioHackrXiv*. 6.2

Gayo, J. E. L. (2020), ShEx by Example - Presentation, doi:10.6084/m9.figshare.13174571.v1, https://figshare.com/articles/presentation/ShEx_by_Example_-_Presentation/13174571. (document), 2.9

gkontos (2021), Decentralized applications using solid pods as datastores, https://raw.githubusercontent.com/ippontech/blog-usa/master/images/2021/09/solid-apps-as-views.svg. (document), 2.2

Goodnight, G. T., and S. Green (2010), Rhetoric, risk, and markets: The dot-com bubble, *Quarterly Journal of Speech*, *96*(2), 115–140, doi:10.1080/00335631003796669, https://doi.org/10.1080/00335631003796669. 1

Greenwald, G., and E. MacAskill (2022), Nsa prism program taps in to user data of apple, google and others, https://www.theguardian.com/world/2013/jun/06/us-tech-giants-nsa-data. 2.5

Group, C., and S. Group (), http://schema.org. 4.4.2

GROUP, S. E. C. (2019), http://shex.io/shex-semantics/. 2.9.1, 2.9.1, 2.9.1, 2.9.1

GROUP, S. E. C. (2022), https://www.w3.org/Submission/shex-primer/. 4.3

Herbert, F. (1965), *Dune*, Chilton Books. 0.1

Hevner, A., A. R, S. March, S. T, J. Park, Ram, and Sudha (2004), Design science in information systems research, *Management Information Systems Quarterly*, *28*, 75–. 3.1, 3.4

Hogan, A. (2020), *Web of Data*, pp. 15–57, Springer International Publishing, Cham, doi:10.1007/978-3-030-51580-5_2, https://doi.org/10.1007/978-3-030-51580-5_2". 2.6.1

Iivari, N., and K. Kuutti (2017), Towards critical design science research, *Association for Information Systems, AIS Electronic Library (AISeL)*, iCIS 2017 PROCEEDINGS. 3.1

Johanson, G., and K. Williamson (2018), *Research Methods: Information, Systems, and Contexts*, Routledge. 3.2, 3.3, 5.2, 5.2.2, 5.2.2, 5.2.2, 5.2.2

Klausen, D. (2017), Do we pay with information? 1.1

Klint, F. (), https://www.wired.com/2017/04/tim-berners-lee-inventor-web-plots-radical-overhaul-creation/. 1

Knublauch, H. (), Shacl playground, https://shacl.org/playground/. 4.2

Labra Gayo, J. E., E. Prud'hommeaux, I. Boneva, and D. Kontokostas (2017), *Validating RDF Data*, no. 1 in Synthesis Lectures on the Semantic Web: Theory and Technology, 1–328 pp., Morgan & Claypool Publishers LLC, doi:10.2200/s00786ed1v01y201707wbe016, volume 7. (document), 2.10, 2.10, 4.3

Ltd, J. (2011), https://www.drawio.com/. (document), 2.4

Manola, F., E. Miller, D. Beckett, and I. Herman (2022), Rdf primer — turtle version, https://www.w3.org/2007/02/turtle/primer/. (document), 2.5, 2.6.1

Mendes, P. N., M. Jakob, A. García-Silva, and C. Bizer (2011), Dbpedia spotlight: shedding light on the web of documents, in *Proceedings of the 7th international conference on semantic systems*, pp. 1–8. 4.5.1, 4.5.2

Munroe, R. (2011), Standards, https://xkcd.com/927/. (document), A.1

Musen, M. (June 2015), The protégé project: A look back and a look forward. ai matters. association of computing machinery specific interest group in artificial intelligence, 1(4), doi:10.1145/2557001.25757003. 2.10.1

Prud'hommeaux, E., and J. Bingham (2021a), https://shapetrees.org/TR/specification/. 2.10.2

Prud'hommeaux, E., and J. Bingham (2021b), https://shapetrees.org/TR/primer/. 2.10.2

RealTimeStatisticsProject (2022), Total number of websites, https://www.internetlivestats.com/total-number-of-websites/. 2.1

Richardson, L. (2004), Beautiful soup, https://www.crummy.com/software/BeautifulSoup/. 9

Ryman, A. G., A. Le Hors, and S. Speicher (2013), Oslc resource shape: A language for defining constraints on linked data., *LDOW*, *996*. 2.7

SHAPE-EXPRESSIONS-COMMUNITY-GROUP (October 2019), Shape expressions (shex) 2.1 primer, https://shex.io/shex-primer/index.html. 4.3

snowden, e. (2016), Just days left to kill mass surveillance under section 215 of the patriot act. we are edward snowden and the aclu's jameel jaffer. aua, https://www.reddit.com/r/IAmA/comments/36ru89/just_days_left_to_kill_mass_surveillance_under/crglgh2/. 2.5

Spielberg, S. (1993), Jurassic park, spoken by Dr. Ian Malcolm. 3.1

Statista Research Department, S. (2022), Internet and social media users in the world 2022, https://www.statista.com/statistics/617136/digital-population-worldwide/. 2.1

Team, D. (2022), https://www.dbpedia.org/about/. 2.11

Teijgeler, H. (2016), Shacl portal, https://15926.org/topics/SHACL-Portal/index.htm. (document), 4.3

The_Solid_Team (2021a), https://solidproject.org/origin, the SOLID project website. 0.2

The_Solid_Team (2021b), https://solidproject.org, the SOLID project website. 1, 1.2, 2.2

Tuba, M., S. Akashe, and A. Joshi (2021), *ICT systems and sustainability*, Springer. 1

Tønset, S. (2023), Johannes' digitale mareritt. 2.5

Vaidhyanathan, S. (2012), *The Googlization of everything: (and why we should worry)*, University of California Press. 1.1.1, 3.2

Veres, C. (), Solid pod management for social media, the project proposal given in class. 1.2

Veres, C. (2011), Lexitags: An interlingua for the social semantic web., in *SDoW@ ISWC*. 4

Veres, C. (2021), http://lexitags.dyndns.org/webgui/. (document), 3.3, 4.1, 4.2

W3C (2022), https://www.w3.org/TR/shacl/. (document), 2.7, 2.6, 2.9, 2.9, 2.9, 2.9, 2.9, 2.9, 2.9

Wegner, P. (1996), Interoperability, *ACM Computing Surveys (CSUR)*, *28*(1), 285–287. 2.4

Zeng, M. L. (2019), Interoperability, *KO Knowledge Organization*, *46*(2), 122–146. (document), 2.4, 2.3

Zuboff, S. (2019), *The age of surveillance capitalism: The fight for a human future at the new frontier of power: Barack Obama's books of 2019*, Profile books. 1, 2.5