AN ALCTS MONOGRAPH

LINKED DATA FOR THE PERPLEXED LIBRARIAN

SCOTT CARLSON
CORY LAMPERT
DARNELLE MELVIN
AND
ANNE WASHINGTON

ALA Editions
CHICAGO | 2020

ISBNs
978-0-8389-4746-3 (paper)
978-0-8389-4712-8 (PDF)
978-0-8389-4710-4 (ePub)
978-0-8389-4711-1 (Kindle)

Library of Congress Control Number: 2019053975

Cover design by Alejandra Diaz. Text composition by Dianne M. Rooney in the Adobe Caslon Pro and Archer typefaces.

♾ This paper meets the requirements of ANSI/NISO Z39.48–1992 (Permanence of Paper).

Printed in the United States of America

23 24 22 21 20 5 4 3 2 1

CONTENTS

EPILOGUE

INTRODUCTION

Since the mid-2000s, the greater GLAM (galleries, libraries, archives, and museums) community has proved itself to be a natural facilitator of the idea of linked data—that is, a large collection of datasets on the Internet that is structured so that both humans and computers can understand it. With our specialized needs in discovery, precise searching, authority control, and disambiguation, along with our lengthy history of producing complex, structured metadata, we in the GLAM community could hardly have asked for a better position to be in with regard to the topic. Over the last ten years, our community has published countless articles about experimentation with GLAM linked data; GLAM conferences have gained a reliable presentation topic; the Andrew W. Mellon Foundation provided close to $8 million in funding for a series of linked data collaborations between a group of Ivy League university libraries; and the Library of Congress announced BIBFRAME, a linked data format intended to replace our long-standing MAchine Readable Cataloging (MARC) format.[1] The outcomes of each individual project may vary, but you can't deny that it has been a very exciting decade for linked data in GLAM.

Yet despite this activity, linked data has become something of a punchline in the GLAM community. For some, *linked data* is one of this era's hottest technology buzzwords; but others see it as vaporware, a much-hyped project that will ultimately never come to fruition. The former may be true, but the latter certainly is not. To quote Tim Williams, a proponent of linked data use in the pharmaceutical industry, the rebuttal to the idea that linked data will never happen is the fact that it is, indeed, happening.[2] Google, Facebook, the Wikimedia Foundation, and others are already putting the underlying standards of the Semantic Web to use, often in websites you use on a daily basis (even though you might not know it). Libraries are also part of the wave, with linked data being delivered through the discovery web pages of library service platforms by SirsiDynix, iii Innovative, and EBSCO. Comparing this unfolding reality to the GLAM community's negative perception of it reveals a significant rift, so what's the problem?

First, only a few in the GLAM community use (and evangelize) linked data, compared with the many who do not. Research and experiments in linked library data are costly, in terms of both technological support and staff time spent away from ongoing library work. Visit a GLAM conference's sessions on linked data and you will chiefly find attendees who have the institutional and financial support to play with linked data (not to mention the institutional and financial support to simply attend conferences). The result is a small assemblage of linked data enthusiasts that rarely grows or changes, which in turn stagnates the technical infrastructure that would welcome others into Linked Data Land.

Our failure to grow this audience is only matched by our field's inability to communicate the practical opportunities of linked data to others in the GLAM community. With all due respect to our technically minded kin, not everyone in GLAM Land has the technical background that is implicitly required to understand linked data concepts; likewise, not everyone in the GLAM linked-data community is able or willing to explain those concepts in a nontechnical fashion. When the GLAM community does ask our enthusiasts to slow down and explain the topic of linked data, too often the explainers fall back on technical jargon, an unintentional (or, in some cases, *very* intentional) form of gatekeeping. If the GLAM community cannot adequately communicate linked data principles to its members, then we either don't actually understand the subject or we don't have an interest in effectively communicating it. The result is a pervasive myth that linked data is too complex for nontechnical GLAM audiences to understand.

In response, this book aims to smash that myth into a thousand jagged shards by presenting the basics of linked data. It is written with the perspective of the GLAM community in mind—specifically librarians, and even more specifically, librarians whose background may not be traditionally considered "technical." The politics of such a statement are (quite rightly) a minefield, since members of departments aligned with cataloging, metadata, and patron-focused catalog searches are often disregarded as "technical, but, you know, not *technical*." This book intends to strip away that pretense and present basic information about linked data in a clear, jargon-minimized way.

That being said, due to the nature of its subject matter, this book inevitably becomes increasingly technical as it wears on. For example, chapter 5 includes samples of a linked data search tool called SPARQL, which may appear unintelligible to those without some experience in SQL (Structured Query Language), which is used to interact with data inside relational databases, or programming languages such as Python. The inverse situation is that library staff with a broad technical background may find portions of this book redundant or overly simplistic. And that's okay! If anything, it means you may be more skilled with linked data than you might have thought.

While we're on the subject of experience, please be aware that while this book is a great primer on linked data basics, it is not an exhaustive dive into the topic, nor is it intended to make you an expert. Rather, its purpose is to get you up to speed and conversant on a (relatively) new technology that could jostle libraries (and archives, galleries, and museums) into new cultural and technological territories. Once you've finished this book, if you find yourself still interested, or better yet, energized by the discussion, there are plenty of opportunities to get involved and work on linked library data, with chapter 7 to get you started.

NOTES

1. LD4L, "LD4L: Linked Data for Libraries," www.ld4l.org/.
2. Tim Williams, "Overcoming Resistance to Technology Change: A Linked Data Perspective," PHUSE EU Connect 2018, www.phusewiki.org/docs/Frankfut%20Connect%202018/TT/Papers/TT01-tt04-19214.pdf.

1

ENQUIRE WITHIN UPON EVERYTHING

The Origins of Linked Data

> *The hardest part of being a developer isn't the code, it's learning that the entire internet is put together with peanut butter and goblins.*
>
> —Sarah Drasner, developer advocate at Microsoft[1]

Chances are, you are reading (or listening to) this book because you have questions about *linked data*, that prevalent, somewhat inscrutable term that has been buzzing around the GLAM-o-sphere for the better part of a decade. Perhaps you work in a position that doesn't have much of a technical component. Or maybe your job is highly technical, but you still aren't sure what to make of the voluminous conference presentations and webinars on linked data that you've attended. Maybe you're just looking for a refresher. Whatever your background, we welcome you to this book.

In preparing this book, we spoke to countless librarians about what they really wanted to know when they began learning about linked data, and what some of them still want to know. Not surprisingly, the single most recurring answer comes down to: why? Why are we still talking about linked data for libraries when there seems to be so little progress in the field? What makes this metaphorical White Whale of a technology so special? What does it have to do with us?

To be able to answer these questions, we need to tell the short, very recent history of the technologies that power linked data. As it turns out, talking about the history of linked data means also talking about the World Wide Web. In fact, linked data's existence was, and continues to be, inextricably intertwined (or do we mean *linked?*) with the creation of the web, which was the most important invention of the late twentieth century. This chapter covers the chronological footpath between these two technologies, allowing us to understand the *why* of linked data before we attempt to tackle the *how* in the rest of this book.

To do this, we need to jump back a few decades to the 1980s. What we think of today as the Internet did not exist then, but there was a primordial version, born in the 1960s. Contrary to popular belief, the internet was not designed by the U.S. government as a communications backchannel in the event of nuclear war. In fact, the program—the Advanced Research Projects Agency Network (ARPANET), funded in 1966 by the U.S. Department of Defense—was actually an experiment in distributing computing power among geographically separate research institutions; in this way, a facility that needed more processing power than was available on-site could reach out to use someone else's computers. The ARPANET was eventually shut down in 1990, but its side projects—for instance, research on long-distance communication and secure data transfer—effectively brought us closer to what we recognize as today's internet.[2]

By the early 1980s, there was finally a capacity for computers to connect to a network of different networks—the definition of an "internet"—but it was certainly not for everyone. The online interfaces were almost entirely text, which meant that navigating the network required some degree of command-line proficiency. Today, it is comically easy to click a link to read an article online, but in the pre-web days, simply acquiring a text document was a somewhat cumbersome task. Putting aside the knowledge needed to point a file-transfer program at a specific server, the user would have needed to know (1) that the file existed and was available to download; (2) the appropriate log-in and password to access the server; and (3) where on the server (i.e., in what folder) the file was kept. Since the first internet service resembling a search engine did not appear until late 1990, finding stuff online required being on the right electronic discussion lists or accessing Usenet newsgroups (the precursors to modern online forums) at just the right time.[3]

THE WORLD WIDE WEB

Into this background steps the main character of our story. In June 1980, an Englishman named Tim Berners-Lee spent six months as a consultant at CERN, the European Organization for Nuclear Research, known today for operating the largest and most powerful particle collider in the world. (Less renowned but no less awesome was the Cernettes, a long-running doo-wop band of women working at CERN.)[4] The son of computer builders and programmers, Berners-Lee took the consulting job to work on systems that would enable instant data acquisition and distribution. Another project, however, would soon capture his attention.

Historically, CERN has employed vast numbers of people at a time. During Berners-Lee's tenure there, the interlacing job functions, projects, and software needs of thousands of other CERN researchers became too much for him to remember on his own. In response, he created a program designed to store this web of random, associated information. The program—dubbed ENQUIRE after his cherished childhood almanac, *Enquire Within Upon Everything*—was capable of describing documents, people, internal working groups, and other real-world things at CERN using roughly a dozen two-way relationship markers.[5] (Figure 1.1 contains a handful of example diagrams imagining how these objects and relationships might have been described in

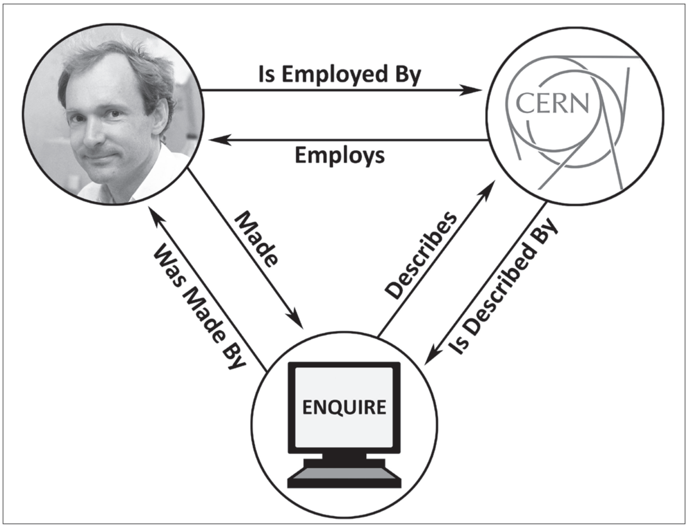

FIGURE 1.1
Enquire example diagram featuring Tim Berners-Lee

ENQUIRE, which now no longer exists.) Berners-Lee would later write that because ENQUIRE stored its information differently than how most people store files in directories today, intuitive leaps from random associations were possible; these connections could then be cross-referenced as new connections with other entities within the ENQUIRE program, automatically connecting back to the original entity.[6]

Part of ENQUIRE's structure was informed by *hypertext*, a term coined by information technology pioneer Ted Nelson as a Harvard graduate student in the early 1960s. Hypertext was a nonlinear, nonsequential form of writing in which sections of text—either whole pages, paragraphs, or short bursts of words—could be connected to text elsewhere, not unlike footnotes. Users reading hypertext on a computer could click these links to follow whichever informational rabbit hole they wanted. Nelson spent most of his professional life pursuing "Project Xanadu," a piece of conceptual software that would use hypertext to facilitate out-of-sequence reading, visualize changes between documents, and source quotations between texts using—wait for it—two-way links running between text and information points. (After more than fifty years in development, Nelson finally delivered a working version of Project Xanadu in 2014.)

His consulting job finished, Berners-Lee left both CERN and ENQUIRE, which was admired by colleagues but not used. He rejoined CERN in 1984, facilitating communication between CERN's computers and networks, and he also resurrected the idea of ENQUIRE. "In addition to keeping track of relationships between all the people, experiments, and machines, I wanted to access different kinds of information, such as a researcher's technical papers, the manuals for different software modules, minutes of meetings, hastily scribbled notes, and so on," he later wrote. "Furthermore, I found myself answering the same questions asked frequently of me by different people. It would be so much easier if everyone could just read my database."[7] But, he soon realized, the last thing he wanted was to create an actual database of information and links, especially one that was centralized. A distributed, decentralized system, on the other hand, could not only scale to the demands of a large number of users, but also guaranteed that anyone could access it without special privileges.

Intrigued by that nascent network-of-networks, Berners-Lee decided to magnify ENQUIRE into a larger system that he could pitch to CERN as a documentation system that would not disrupt his colleagues' organizational styles. The goal was to present information universally to the user, no matter the computing platform. Documents would be written in a standardized way that, upon access, would convey the textual content and structure of a written document, including links to other documents. Without needing to manage a central database of links, adding new documents would be ridiculously easy; anyone could link to anything else, so long as they knew where it was located in the system.

On the advice of his boss, Berners-Lee collected his ideas into a proposal in early 1989, concluding that "a universal linked information system" should be a technical goal for managing information inside and outside of CERN. The proposal read: "The aim would be to allow a place to be found for any information or reference which one felt was important, and a way of finding it afterwards. The result should be sufficiently attractive to use that the information contained would grow past a critical threshold, so that the usefulness of the scheme would in turn encourage its increased use."[8] A collective shrug from the CERN community stalled the project until late 1990, when Robert Cailliau, a Belgian engineer, joined Berners-Lee as a project manager. Cailliau's first task was to hone the proposal and ask for a catchier name. Berners-Lee retitled his project "WorldWideWeb: Proposal for a HyperText Project."[9]

With Cailliau organizing, Berners-Lee was free to get to work on the underlying structure of the web, inventing a suite of protocols and tools allowing the exchange of digital files. The Hypertext Transfer Protocol (HTTP) would allow hypertext documents to be requested and exchanged through transactions with a server, while Uniform Resource Locators (URLs) specified the locations of documents on the server. Finally, the exchanged documents were written in Hypertext Markup Language (HTML), which would be decoded and displayed to the user. Inspired by a preexisting language called Standard General Markup Language (SGML), HTML is built on a standard set of components, the most important of which are tags, or descriptors set between angle brackets. The following, extremely simple example still stands as a working HTML document:

```
<!DOCTYPE html>
<html>
      <head>
            <title>Web page title</title>
</head>
      <body>
            <p>Hello world!</p>
      </body>
</html>
```

The text between the `<head>` tags describes basic information about the page we don't see; our example includes a title, but this could also have links to other HTML or text documents with additional information. Meanwhile, the `<body>` section is the visible content that shows up when we access a web page. Note that the body text of this page—"Hello, world!"—is set within a `<p>` or paragraph, tag. This tag effectively functions as two different kinds of markup language: first as *descriptive markup*, explicitly describing the text as a unit of human writing; and second, it is also *procedural markup*, instructing the web browser to treat this text as one would display a paragraph in other kinds of human writing.

By the end of 1990, a proto-web was up and running at CERN, featuring a simple, text-based website about the project.[10] As a bigger proof of concept, Berners-Lee and Cailliau converted CERN's massive telephone directory of 10,000 employees to HTML, eliminating the need for print copies.[11] (Berners-Lee also published a web page about the Cernettes, making them the first musical group to have a web page, as well as the subjects of the first-ever photograph published to the web.)[12]

Bolstered by CERN's response, in August 1991, Berners-Lee responded to a message on the alt.hypertext newsgroup, announcing the World Wide Web (WWW) project.[13] From there, CERN began offering its source code to anyone who wanted it and urged people to improve upon their work. In 1992 several physics labs, including the National Center for Supercomputing Applications (NCSA) at the University of Illinois, set up their own web servers; and by January 1993 there were fifty known web servers.[14] After seeing a local demonstration of the web, Marc Andreessen, a bored undergraduate student working at the NCSA, partnered with programmer Eric Bina to work on a web-browsing app that, unlike the software distributed by Berners-Lee and Cailliau, did not require extensive command-line experience to run. Their browser, Mosaic, simplified the process of getting the web onto home computers while adding personal flourishes, including modifying HTML to better handle images and graphics; a year after Mosaic's release in 1993, the number of web servers had rocketed to 1,248.[15] The World Wide Web had begun to ascend with particle-accelerator speed.

THE LIMITS OF HTML

Most of the history books written about the internet end here, often because a good chunk of them was written before the end of the twentieth century. The web has grown rapidly and evolved significantly since its accidental birth in the early 1990s, with much shepherding credited to Berners-Lee; upon leaving CERN in 1994, he founded (and still leads) the World Wide Web Consortium (W3C). The W3C continues to be an international standards organization to improve the quality of the web, periodically updating HTML while endorsing other emerging web standards, such as Extensible Markup Language (XML) and Cascading Style Sheets (CSS). However, we may be forgiven for wondering if, behind the scenes, the success of the web rang hollow with its creator. Indeed, almost none of the relationship aspects from ENQUIRE were brought forward to the World Wide Web, with the exception of hyperlinks between web pages; even then, a link to another page carries no intrinsic significance, except to signify that we expect to safely arrive at another HTML page. For instance, imagine looking at someone's personal website; somewhere in the page, a link is anchored to a piece of text: "Click here to see my sister's online store!" In the World Wide Web as it was designed, the only thing that truly signifies that link as (a) an online store or (b) that it is connected to you through a family relationship is a series of four words—"my sister's online store"—written by an English-speaking human and intended for other English-speaking humans to read it. By itself, the HTML has no way of codifying either piece of information; your link, and thus those two pieces of information, are not understandable to a machine that has no ability to comprehend the concepts of "online store" and "sister." Once that link is clicked, any context about the relationship between those two web pages dissipates like smoke in the atmosphere.

As the web began to swell with HTML content, that kind of contextual information became a necessity. In the early days of the web, there was so little content that the totality of entire regions of web content could be collected, described, and published on a web page like an annotated bibliography. (One such "web directory" from 1992, created by Berners-Lee, listed thirty of the known web servers across the world.)[16] But as the internet took off, the total estimated number of websites jumped to a couple thousand in 1994 and had reached (roughly) 17 million by the turn of the century.[17] (There are now about 1.5 billion websites and, depending on whom you ask, somewhere between

5 and 60 billion web pages.) With no feasible way for humans to manually index and describe millions of HTML documents, internet companies responded by inventing search engines. Search engines deploy web crawlers—automated "bots"—to systematically scour the internet, looking for new and updated web pages. The discovered pages are then consumed by the search engine (a process called *indexing*) and digested into an internal database, where information from the pages is associated with search terms drawn from the information itself. When someone uses a search engine, it examines its indexed content and returns its best guesses as to what the user was looking for. The companies that own major search engines spend quite a lot of money on researching, developing, and sharpening the algorithms that process HTML content, much of which is, again, written by humans for other humans to read. Consequently, those algorithms are highly prized company secrets; after all, while descriptive markup can be part of HTML tags—for example, `<blockquote>` and `<cite>` intuitively describe HTML content as block-quotes and citations, respectively—the value of indexing a web page is in being able to machine-process what its content is actually about. And while the World Wide Web has proven to be extremely handy in connecting documents and displaying them to human users, there were no methods built-in to describe the intrinsic "aboutness" of a web page. The `<head>` tags in a website may identify it as an online music store, but does the algorithm that's indexing the site know what a store is? Or, for that matter, what music is?

Not that people didn't try to fix this. As early as 1995, researchers and developers began working on a `<meta>` tag that "defin[ed] a set of words to use to allow document cataloging."[18] Placed among the other information in the `<head>` tag, meta tags allowed web authors to embed information that was previously undefined in HTML, including keywords and description of the page's content, language, and authors. By 1996, a common standard existed for some `<meta>` tags, but not for including keywords.[19] Nevertheless, a handful of web search engines began supporting and recommending the use of keywords in the `<head>` tag, which could then be used in the retrieval and ranking of web content. Search engines gambled that web developers and companies could be trusted to provide accurate and honest descriptions of their content in exchange for web indexing. Naturally, webmasters quickly figured out that hundreds of unrelated terms and words could be stuffed into the keyword tag in an effort to game the ranking systems. Within a few years, search-engine support for a keyword `<meta>` tag dwindled, meaning that search engines would have to develop their own methods of mechanically parsing the content of web pages to fill up their indexes.

To sum up, for almost twenty years, search engines have relied on increasingly complex machine methods of sifting through billions of web pages, and then used complex algorithmic systems to intuit their meanings from human-created text. Consider that over that same time frame, our use of internet search has evolved from the basic research and navigation of content to relying on the internet to facilitate discourse, record history, and even help us remember basic facts. This vast universe of uses depends on machines being able to divine meaning from things we wrote for other humans in an assortment of written and spoken languages. No wonder the search engine companies keep those processes secret.

ENTER THE SEMANTIC WEB

Less than a decade after the web's creation, Berners-Lee published an abridged autobiography that focused on the development of the web. In it, he envisioned the future of his invention: an extension of the World Wide Web that could be processed, directly or indirectly, by machines. The web, as he saw it, had become dependent on machines to find and understand content that was created primarily for human consumption. "If HTML and the Web made all the online documents look like one huge book," Berners-Lee argued, then this new phase would "make all the data in the world look like one huge database."[20] He laid out a theoretical framework in which data on the web was packaged in a way that could be understood by machines—computers, bots, and other automated processors—just as easily as by humans. He termed this new-phase internet the *Semantic Web*, so named to highlight the expectation of unambiguously crafted data. The old web—the web of documents—was implicit, murky, and unpredictably structured. Data on the Semantic Web would be explicit and structured.

For a moment, consider what you know when you hear the name *Charlie Chaplin*. You may not be a historical expert, but through cultural osmosis, you have probably retained some information about him, such as his persona as "the Little Tramp" and at least one of his professions (an incredibly famous comedic actor, as well as a highly regarded director, writer, and composer). Hard-core fans or film scholars will know much more about him, including the names and release dates of his films, his frequent co-stars and collaborators, and biographical details, such as his childhood spent in a London workhouse, the controversial paternity lawsuit filed against him, allegations of Communist ties, and his de facto expulsion from the United States.

Whether your knowledge about Chaplin is finite or bottomless, your brain already understands certain underlying concepts because it has spent the bulk of your lifetime connecting the dots in the background. In fact, in the process of reading some of these extremely topical details that might be new to you, your brain is adding value and context to your recognition without you knowing it. Without really considering it, you probably

1. have ascertained that Chaplin was a human being who expressed a masculine gender identity;
2. possess a basic understanding of comedy and humor, even if you cannot explicitly explain them other than knowing what "funny" is;
3. understand what movies are, and that actors portray roles in films;
4. know that London is a city in the United Kingdom, which itself is a collection of countries, which are politically defined territories.

And so on and so on. Your brain works very hard, and has for the entirety of your life, to register these concepts and make connections, so that when someone mentions the words *Charlie Chaplin* to you, you recognize the concept of a human male who made movies without your having to manually connect to the concepts of human beings, comedians, filmmakers, and Londoners.

To a computer, *Charlie Chaplin* is a 15-character string (or sequence) of symbols—mostly letters from the Roman alphabet. That particular pattern of symbols represents a series of sounds that are meaningful to humans; these sounds elicit a series of thoughts and concepts in the human brain which help humans recognize and understand, conceptually, what a "Charlie Chaplin" was. But to a computer, the sum total of those symbols is devoid of any larger meaning, other than as a pattern of text symbols that can be used to match other strings of text symbols. The early-to-middle history of internet searching was built, in part, on bots scouring HTML data for words and text strings that could be matched with search queries. Proprietary search algorithms, like Google's, mix in other elements to answer a query, but to a large extent, any page with a high rate of matching text strings becomes a viable internet search result. Under these conditions, web pages about Chaplin the actor-filmmaker become potential matches for search queries about his son, Charles Chaplin Jr.; the French painter Charles Joshua Chaplin; and Richard Patrick Bennett, a Jamaican ragga DJ who adopted the stage name Charlie Chaplin.

As an extension of the existing web, the Semantic Web offers a solution to this situation: the ability to create databases that represent facts, concepts, places, times, and people in a way that a computer can intuitively process. One of these databases could contain a *dataset* (a subset collection of data within a database) with a *node* (a specific, unambiguous entry, otherwise known as a "record" in Library Land) that represents our human being, Charlie Chaplin. Verifiable facts and measurable values about this entry—his dates of birth and death, his occupations, the places where he was born, lived, and died—could be added to this node, distinguishing it from, say, other nodes representing his son or Chaplin the ragga musician.

Of course, a human being rarely exists as an island in space and time; we are always connected, through friendships, bloodlines, geographic locations, events, artistic works, and shared experiences. These connections can enhance and augment our database, binding otherwise disconnected nodes by explicitly stating relationships. Our Chaplin node could be connected to that of his son, Charles Jr., along with his other children, his spouses, and the films he created, which then implicitly present the possibility of connecting Chaplin to his costars and crew members. The unspoken premise here is that one node almost always leads to another, and that one to another, ad infinitum, forming a true web of data. With sufficient connections between each node, a computer could leverage the aggregated knowledge to interpret a data search and deliver a set of results directly related to the actual query subject, rather than hoping to just match a string of text.

CASE IN POINT
Following a Path from Chaplin to James Bond

Figure 1.2 is an example of how even an extremely simple web of connections between nodes of a semantic database potentially offers users a smorgasbord of serendipitous conceptual connections. Starting at the top, our Charlie Chaplin node is explicitly connected to Douglas Fairbanks. Fairbanks, a superstar of swashbuckling movies in the 1910s and 1920s, was a close friend of Chaplin's for much of his life. Fairbanks eventually married Mary Pickford, another superstar of the silent film era, making them an early Hollywood power couple. In 1917 Chaplin, Pickford, and Fairbanks, along with D. W. Griffith, cofounded the United Artists film studio, which released some of their most acclaimed movies of the era, such as Chaplin's *The Gold Rush* and *City Lights*;

Fairbanks's *The Mark of Zorro* and *Robin Hood*; and Pickford's *Sparrows* and *Dorothy Vernon of Haddon Hall*. Through both direct and indirect connections, our Chaplin node allows us to drill through other nodes representing people, events, companies, and films.

But we can go even further than what is pictured in figure 1.2: we can start at Chaplin and end up in a James Bond movie. United Artists still exists a century later, albeit in a slightly different form and as a subsidiary of MGM TV Group and Digital (which is itself a subsidiary of Metro-Goldwyn-Mayer, another of the ancient Hollywood studios). United Artists served as distributor of the 1960s and 1970s James Bond films; it is now co-owner of the copyright and trademarks for the classic James Bond film properties, as well as the sole copyright holder of the rebooted Bond series that began in 2006 with *Casino Royale*. The methods may be convoluted, but in a semantic system with reasonably high-quality metadata, we can follow a clear-cut pathway between three completely different film eras.

However, there is an immediate problem with this idea. According to Berners-Lee, the Semantic Web is "a universal space for anything which can be expressed" using computer-accessible representations and logic.[21] In practice, expressing every conceivable thing in computer-processable terms is a laughably grandiose goal, since it would be subject to standard time and money limitations. Because of real-world limitations, we are more likely to end up with either (a) a collection of unconnected databases encompassing small, highly specific areas of subject matter, or (b) exorbitantly large databases with a shallower level of detail. To be of the greatest possible value, the collections of information on the Semantic Web need to be connected.

In 2006 Berners-Lee put up a personal note on the W3C site, cheekily marked as "imperfect but published." In it, he described his technical vision for "a serious, unbounded web in which one can find [all] kinds of things, just as on the hypertext web we have managed to build."[22] At the time, Berners-Lee was focused on making sure that nodes within a single database shared as many connections as possible; he soon widened his scope to joining disparate datasets together on shared information. He called this idea *linked data*. "Letting your data connect to other people's data is a bit about letting go," he wrote in a 2007 blog post. "It is not about giving to people data which they don't have a right to. . . . It is about getting excited about connections, rather than nervous."[23]

One of the biggest stumbling blocks to understanding linked data is, arguably, the term itself. *Linked data* refers to an ideal scenario in which individual

Figure 1.2
Diagram connecting Charlie Chaplin, Mary Pickford, Douglas Fairbanks, their films, and United Artists

concepts (nodes) of datasets are connected to other concepts, both inside the dataset and outside in external datasets and databases. The reality, however, is that the vernacular use of the word *link* refers to a URL address, which has led to confusion about linked data in general. Adding URLs to a website does not create linked data; linked data is the result of expressing relationships across the web. In our Chaplin example, we bolster both the completeness of our information, as well as our data's credibility, by taking the time to state that our Chaplin is the same person—the same conceptual entity—that can be found in other datasets. We include specific references to those other database nodes in our example, knowing full well that machines will use the same Semantic Web standards to follow those links and consume their data. We can even tell the machines that our method of linking data—our ability to say that "our thing is the same as this other thing"—can also be described using the same semantic terms that other databases use. It boils down to finding shared relationships between unconnected dots. Just like ENQUIRE did.

At this point, you may be (rightfully) wondering what the history of the World Wide Web has to do with libraries. In the next chapter, we will delve into the current state of library data and try to explain why a still-nascent data format has enraptured a small section of GLAM Land.

NOTES

1. Sarah Drasner (@sarah_edo), Twitter, March 22, 2016, https://twitter.com/sarah_edo/status/712482904090128387.
2. If you are interested in the early history of the primordial internet, there are a few excellent books and resources to draw from: Janet Abbate, *Inventing the Internet* (MIT Press, 2000); Katie Hafner and Matthew Lyon, *Where Wizards Stay Up Late* (Simon & Schuster, 1998); and Ian Peter's website, NetHistory (www.nethistory.info).
3. "The First Search Engine, Archie," last modified September 21, 2002, archived from the original at https://web.archive.org/web/20110719211803/; http://www.isrl.illinois.edu/~chip/projects/timeline/1990archie.htm.
4. The Cernettes, "Cernettes: The Bios," https://cernettes.wixsite.com/cernettes/the-bios.
5. Tim Berners-Lee, "A Brief History of the Web," World Wide Web Consortium, www.w3.org/DesignIssues/TimBook-old/History.html.
6. Tim Berners-Lee, *Weaving the Web* (San Francisco: HarperSanFrancisco, 1999), 10–11.
7. Berners-Lee, *Weaving the Web*, 15.
8. Tim Berners-Lee, "The Original Proposal of the WWW, HTMLized," World Wide Web Consortium, www.w3.org/History/1989/proposal.html.
9. Walter Isaacson, *The Innovators* (New York: Simon & Schuster, 2014), 412–13.
10. CERN has since preserved that first website at http://info.cern.ch/.
11. Johnny Ryan, *A History of the Internet and the Digital Future* (London: Reaktion, 2010), 107.
12. Andrew Hough, "How the First Photo Was Posted on the Web 20 Years Ago," *Telegraph*, July 11, 2012, www.telegraph.co.uk/technology/news/9391110/How-the-first-photo-was-posted-on-the-Web-20-years-ago.html.
13. Isaacson, *The Innovators*, 414.

14. John Naughton, *A Brief History of the Future* (London: Weidenfeld & Nicolson, 1999), 239.
15. Berners-Lee, *Weaving the Web*, 217.
16. Tim Berners-Lee, "World Wide Web Servers," World Wide Web Consortium, www.w3.org/History/19921103-hypertext/hypertext/DataSources/WWW/Servers.html.
17. Internet Live Stats, "Total Number of Websites," last modified August 2019, www.internetlivestats.com/total-number-of-websites/.
18. Davide Musella, "The META Tag of HTML," IETF Tools, last updated December 20, 1995, https://tools.ietf.org/html/draft-musella-html-metatag-01.
19. Danny Sullivan, "Meta Keywords Tag 101: How to 'Legally' Hide Words on Your Pages for Search Engines," Search Engine Land, September 5, 2007, https://searchengineland.com/meta-keywords-tag-101-how-to-legally-hide-words-on-your-pages-for-search-engines-12099.
20. Berners-Lee, *Weaving the Web*, 186.
21. Tim Berners-Lee, "Axioms of Web Architecture," World Wide Web Consortium, last modified August 27, 2009, www.w3.org/DesignIssues/Rules.html.
22. Tim Berners-Lee, "Linked Data: Design Issues," World Wide Web Consortium, last modified June 18, 2009, www.w3.org/DesignIssues/LinkedData.html.
23. Tim Berners-Lee, "Giant Global Graph," Decentralized Information Group, Massachusetts Institute of Technology, last modified November 21, 2007, archived from the original at https://web.archive.org/web/20160713021037/; http://dig.csail.mit.edu/breadcrumbs/node/215.

INDEX