2022-01
第 1 期

数字人文

Digital Humanities

清华大学
中华书局
主办

中华书局

目　录

■ 编辑导言

■ 研究性论文

■ 数据库与工具

CONTENTS

Leaping (and Bridging) the Digital Gorge: Development, User-Experience, and the China Historical Christian Database(CHCD)

Alex Mayfield（马飞立） / Department of Social Science&History, Asbury University
Daryl Ireland（艾德恩） / Boston University
Eugenio Menegon（梅欧金） / Boston University

Introduction: The Digital Gorge

Humanists have long recognized that liberal arts are under attack in higher education, and that Asian humanities are particularly vulnerable. In a recent essay, Peter Bol argues that there are three main challenges facing Asian humanities in the United States. First, a general malaise among Western humanists prevents them from looking beyond the Mediterranean for sources of inspiration. Second, there is a decrease in foreign-language acquisition in the United States, making the pool for potential Asian humanists much smaller. Lastly, humanists have been slow to adapt to digital technologies which might increase "opportunities for reaching a broader audience."[①] Bol's technical essay goes on to make the case that humanists are not using these technologies to their fullest potential. His argument also suggests the tantalizing possibility that by addressing the third challenge, Asian humanists might make headway in addressing the other two. Though perhaps a touch dramatic, the mythic folklore of the *Hutiaoxia (The 'Tiger Leaping Gorge'of Shangri La)* provides an apt metaphor: Asian humanists can be described as tigers being hunted on the edge of a digital cliff. If they could only leap across the narrow gorge to the other side, they might have a fighting chance to survive.

Bol's own career has shown that such bounds can and should be made. Successful digital humanities projects like the China Biographical Database (CBDB) Project and the China Historical GIS (CHGIS) have demonstrated the value of large-scale digital projects in the field of Chinese humanities.[②] These projects provide rich and complex datasets that humanists can utilize in new modes of inquiry that prior historiographic and humanistic

①Peter Bol, "How the Digital is Changing Research and Teaching on Asia," *ASIANetwork Exchange*, vol. 25, no. 2, 2018, pp. 7-28.

②See "Welcome," China Biographical Database Project, https://projects.iq.harvard.edu/cbdb/home, accessed October 23, 2020; China Historical GIS, http://chgis.fas.harvard.edu/, accessed October 23, 2020.

approaches rarely considered.[①] Among these modes of inquiry is network analysis to which this issue is dedicated. As a relatively new approach in the digital humanities, this methodology has the potential to bring humanities into a more robust conversation with fields as diverse as mathematics, biology, and physics. Networks are complex and evolving aspects of reality and, as such, are as alive in historical sources as they are in molecular phase transitions or the internet.[②] With resources like the ones above, digital humanities in Chinese studies are positioned to provide a much needed historical and cross-cultural contribution to the interdisciplinary conversations about networks.

Yet, these projects face many of the same hurdles that digital humanities projects face generally: they only benefit technically adept humanists. While valuable, the real fruits of many digital projects remain inaccessible to most humanist scholars. Mining their depths usually requires learning advanced technological skills, mastering coding languages, or rifling through complex and hard-to-use user interfaces. In short, many of the leaps taken by digital humanists do not translate back across the digital gorge.[③] Unfortunately, such an approach threatens to undermine the digital humanist's goal of reaching broader audiences and demonstrating the value of digital humanities amid the attack on liberal arts.[④] As such, early adopters of the digital humanities might have a sort of evolutionary imperative for survival. Our approaches to digital humanities must not just leap the digital gorge; they must help bridge it.

To that end, this essay will look at two newer technological toolsets which can aid digital humanists as they seek to leap and then bridge the digital gorge: graph databases and JavaScript data visualization coding libraries. While these technologies are powerful, they also prioritize flexibility, visualization, and accessibility that translates to a much gentler learning curve and a much shorter development time frame for digital humanists.

---

①Joshua Sternfield, "Historical Understanding in the Quantum Age," *Journal of Digital Humanities*, vol. 3 no. 2, Summer 2014, http://journalofdigitalhumanities.org/3-2/historical-understanding-in-the-quantum-age/.

②The early work of Barabasi and Frangos provide an interesting cross-disciplinary look at networks and graph theory. History is embedded within the narrative of their popular book, *Linked*. See Albert-laszlo Barabasi and Jennifer Frangos, *Linked: How Everything Is Connected to Everything Else and What It Means for Business, Science, and Everyday Life*, New York: Basic Books, 2002.

③In this estimation we are not alone. Digital humanists who are pedagogically minded and heedful of user experience have been making this observation for some time. See, for example, Sean Michael Morris, "Digital Humanities and the Erosion of Inquiry," *Disrupting the Digital Humanities*, eds. Dorothy Kim and Jesse Stommel, Earth, Milky Way: punctum books, 2018, pp. 217-226; Klaus Thoden et al., "User-Centered Design Practices in Digital Humanities-Experiences from DARIAH and CENDARI," *BI Technik*, 2017, vol. 37, no. 1, 10.1515/abitech-2017-0002; Fred Gibbs, Trevor Owens, "Building Better Digital Humanities Tools: Toward Broader Audiences and User-centered Designs," *Digital Humanities Quarterly*, vol. 6, no. 2, http://www.digitalhumanities.org/dhq/vol/6/2/000136/000136.html.

④Hitchcock suggests that digital historians have "the undeniable choice to create new forms of history that retain the empathetic and humane characteristics found in the old generic forms." We concur and add that it is, in fact, more than a choice, but an imperative. See Tim Hitchcock, "Academic History Writing and the Headache of Big Data," *Historyonics*, last modified January 30, 2012, https://historyonics.blogspot.com/2012/01/academic-history-writing-and-headache.html.

As a case in point, this essay will utilize the China Historical Christian Database (CHCD), a new prosopographical and geographic database for the study of Christianity in Chinese history, to demonstrate the benefits provided by these technologies. At key junctures, the essay will provide examples from the CHCD, and it will conclude with a brief discussion on the development and design of the project.[①] Built on a graph database platform, the project structures its data as an intuitive network graph which allows for complex querying. At the same time, the project utilizes JavaScript data visualization libraries to prioritize end-user experience and accessibility. While still in the first phase of development, the project already illustrates that a concern for a research product's reception can enhance the project's goals and publicly illustrate the value of digital humanities research.

Leaping and Bridging: Graph Databases and JavaScript Data Visualization

It might seem strange to group graph databases and JavaScript development packages together as essential toolsets for digital humanists interested in network analysis: the former is an underlying technological structure, and the latter is a front-end development tool for web applications. Each, however, holds unique advantages for humanists who want to leap and bridge the digital gorge. Both make it far easier to develop innovative projects that appeal to broader audiences.

Graph Databases

Graph databases provides several benefits for digital humanists interested in leaping the digital divide. Specifically, graph databases make it far simpler for humanists to develop database projects, to apply the statistical methodologies like social network analysis, and to learn the requisite skills to implement their digital projects. In effect, graph databases create a gentler learning curve for database design and implementation by rendering database design more natural and data analysis more comprehensible to the average humanist scholar.

As a newer form of database, graph databases are essentially a reconceptualization of database structures which abandon the more established format of standard relational databases. Rather than relying on a series of interconnected tables, graph databases embrace triples as the standard organizing element. G. F. Hurlburt provides a description of the structure of graph databases:

Graphs are expressed in node-arc-node (subject-predicate-object) triples. This notion of a graph is fundamentally straightforward. Nodes generally represent physical or conceptual objects, typically associated with objects as represented in a programming

①More information about the project and a preview of the current prototype of the project's online platform can be found at https://chcdatabase.com/.

language. Nodes can typically have one or many descriptive properties ascribed to them. Arcs, or edges, represent metaphysical constructs that connect or create relationships between nodes or properties. In some graph representations, properties can also be assigned to arcs.①

In short, graph databases apply network thinking to the core structure of a database. Or perhaps it might be said the other way around, "social networks are usually represented as graphs."② Rather than using the unfamiliar abstractions of rows, columns, and join tables, graph databases situate data within the structure of a network itself, and triples become the basic storage block of information. This decreased amount of abstraction can be seen in Figure 1 below which depicts the graph schema of the CHCD alongside a comparable relational database schema.

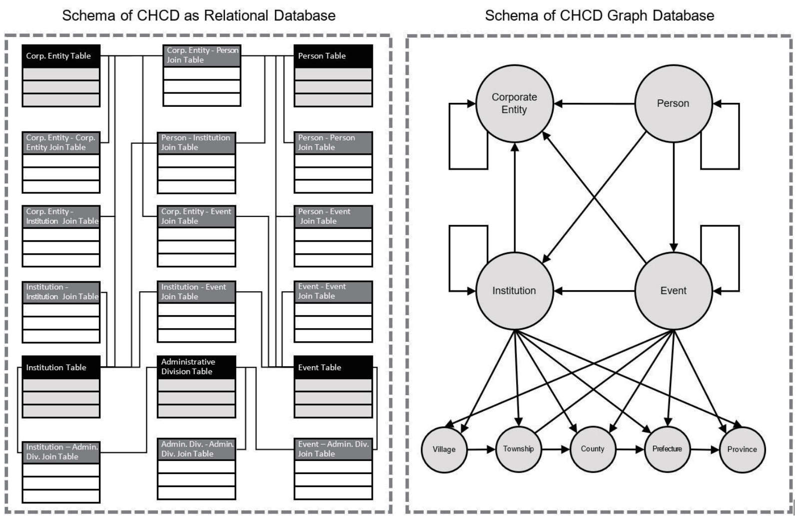

Figure 1　Comparative Design of the CHCD

Both schemas can record the same kind of information and to the same degree, but the graph schema organizes its data using triplets and is far more readable to the average person. While this seems like a peripheral design choice, it translates to advantages on both a technical and practical level.

Technically, graph databases are ideal for representing and analyzing complex network data, and in many ways are the "logical fallout" of the ongoing trend towards network

①G. F. Hurlburt, G. K. Thiruvathukal, M. R. Lee, "The Graph Database: Jack of All Trades or Just Not SQL?," *IT Professional*, vol. 19, no. 6, 2017, p. 22.
②Nagehan Ilhan, Sule Gündüz-Ögüdücü, A. Sima Etaner-Uyar, "Introduction to Social Networks: Analysis and Case Studies," *Social Networks: Analysis and Case Studies*, eds. Sule Gündüz-Ögüdücü, A. Sima Etaner-Uyar, Vienna: Springer, 2014, pp. 3-4.

thinking.① This shift benefited greatly from the long mathematical history of graph theory. Graph theory itself goes back as far as the eighteenth century to the famed mathematician Leonard Euler. When Euler offered his solution to the Königsberg Bridge problem in 1753, he utilized a graph (i.e. a network diagram) to solve his problem. In doing so, Euler noticed that graphs had distinctive properties which were useful for solving mathematical problems.② Over the centuries, graph theory remained a small subfield of mathematics until developments in computational science and the advent of the World Wide Web caused researchers in multiple fields to reassess the value of graph theory. Within the realm of data management, this brought forth the graph database.

While graph databases are not as efficient when performing end-to-end transactions and associative summation, they do offer three technical benefits which could be of use to humanists interested in network analysis.③ First, graph databases are performance powerhouses for organizing and analyzing the data of large complex networks. In certain use cases and data structures, query latency can improve by one or more orders of magnitude.④ Second, just as real-world networks are dynamic and constantly evolving, graph databases are "naturally additive."⑤ This means that data structures are flexible and can adapt to the changes in network structures or the research question in view. As a consequence, graph datasets are highly integrative and allow related datasets to communicate with one another more easily. Last, graph databases might also be described as technically "agile." On a practical level, researchers can start developing their database without a complete sense of the final data structure in mind.⑥ This can help cut down on development time for any project because it frees researchers to ask the important questions that they discover along the way without having to start their database from scratch every time they modify their research questions.

Graph databases also offer a bevy of practical benefits to researchers interested in social network analysis. The largest benefit is the aesthetic sensibility of graph databases: they simply make more sense. While any form of data modeling is a form of abstraction, the logical model of graph databases closely follows our linguistic representation of relationships (e.g., "Sally is married to Tom," "Tom is a colleague of Charlie," etc.). In a traditional relational database, the logic of linguistic expression is forced into a more abstract physical model of tables, columns, and rows.⑦ Graph databases remove a certain

---

①Hurlburt, Thiruvathukal, Lee, "The Graph Database," pp. 21-22.

②Robin J. Wilson, "Section 1.3. History of Graph Theory," *Handbook of Graph Theory*, eds. Jonathan L. Gross et al., Boca Ratonm FL: CRC Press LLC, 2013, pp. 31-51; Barabasi and Frangos, *Linked: How Everything Is Connected to Everything Else*, p. 13.

③As such, graph databases may not be the best choice for text-oriented humanities projects, or—perhaps further afield—humanities projects which seek to sell anything, Hurlburt, Thiruvathukal, and Lee, "The Graph Database," p. 22.

④Ian Robinson, James Webber, Emil Eifrem, *Graph Databases*, Sebastopol, CA: O'Reilly Media, 2013, p. 8.

⑤Robison, Webber, Eifrem, "Section 1.3. History of Graph Theory," p. 9.

⑥Robison, Webber, Eifrem, "Section 1.3. History of Graph Theory," p. 9.

⑦Robison, Webber, Eifrem, "Section 1.3. History of Graph Theory," pp. 22-23.

degree of abstraction from the database, thus requiring a smaller conceptual leap for humanists who are new to database design and implementation. Further, this network structure is even more apparent on the database user-end: the graphical interfaces for most graph database platforms utilize network graphs as the primary mode for viewing and editing data. In adopting the graph database, humanists are also better situated to benefit from the mathematical toolsets which are employed by network scientists: graph databases are ready out of the box to measure key network metrics such as centrality, clustering coefficients, density, and structural cohesion. This also brings humanities projects into conversation with network-based research in other fields.①

The final benefit that graph databases offer humanists is popularity and ease of access. Graph database platforms such as Neo4j, JanusGraph, and DGraph are used by industry giants like Intel, Hewlett Packard, Comcast, eBay, Microsoft, and IBM.② While those companies utilize more advanced paid features, the major platforms are open-source and have free community versions which would be enough for virtually any digital humanist. This hybrid community-commercial model means that development teams behind various platforms tend to be well-funded and are actively working to make the product better and more accessible. In practice, this ensures that documentation for major platforms is actively maintained, easy to use, and filled with tutorials for potential users/customers. While there is still a learning curve, these training resources help to significantly decrease the slope of that curve.

JavaScript Data Visualization Libraries

Graph databases make it easier to create valuable datasets but, without data visualization, they still cannot effectively bridge the digital divide between the general humanist and complex data. Scholars usually link graph databases to more powerful software like Gephi to create network visualizations and conduct quantitative analysis,③ but many humanists do not have such advanced technical skills. Therefore, it is necessary for graph databases to employ some form of data visualization interface that allows database users to see and read the data in an intuitive manner.

Databases are complex webs of interconnection, and the data stored therein can be modeled and conceptualized in multiple ways and for multiple purposes. Digital humanities projects which rely on this sort of complex data could exponentially increase their value and impact if they ensure more people could comprehend and engage with that data. JavaScript data visualization libraries are a key tool for accomplishing this goal. Bridging the digital

①See, for example, Ruth Ahnert, Sebastian E. Ahnert, "Protestant Letter Networks in the Reign of Mary I: A Quantitative Approach, *ELH*, vol. 82, no. 1, 2015, pp. 1-33.

②For more on each platform, see Neo4j, https://neo4j.com/, accessed October 23, 2020; JanusGraph, https://janusgraph.org/, accessed October 23, 2020; Dgraph, https://dgraph.io/, accessed October 23, 2020.

③Gephi, for example, has a streaming plugin which connects to several graph database platforms and exports data in the correct format. For more, see "Supported Graph Formats," Gephi, https://gephi.org/users/supported-graph-formats/, accessed October 23, 2020.

gorge, however, does not require a massive re-design of humanities education more generally (though this would indeed help). Rather, digital humanities projects could expand their impact and bridge the divide by making the strategic choice to prioritize the end-user experience of their project. To this end, it might behoove digital humanists to make room in their grant budgets for expenses related to front-end development.

Bridging to the general public in this way, however, does require an additional leap by the digital humanities project team. Thankfully, advances in the world of application development have dramatically cut development time and lowered the technical threshold for creating customizable user interfaces and mobile applications. In other words, the leap is far easier than it used to be. The various coding languages used for application development abound, and a project team should undertake a cost-benefit analysis before proceeding with any of these languages. With that said, programming languages like C#, C++, Java, and JavaScript have an assortment of useful and often open-source code libraries. These can be used to create interfaces for humanities projects that allow end-users (i.e., researchers, students, and the general public) to explore the data from a digital project. The remainder of this section will focus on the benefits of JavaScript code libraries as those are most familiar to the authors.

JavaScript remains one of the most accessible programming languages for application development, and it also offers several coding libraries which prioritize data visualization, in general, and network visualizations in particular. These libraries progress along an axis which runs from simple, standardized visualizations and easier implementation on one end to complex, non-standard visualizations and more difficult implementation on the other end. This range is helpful to digital humanities projects, as project goals and development timelines can vary dramatically. On the simple-easy end of the axis, we find JavaScript libraries like Vis.js, Springy.js, and Sigma.js.[①] Each of these libraries provides developers with easy parameters that help transform project data into web-ready visualizations in a few keystrokes. Furthermore, Springy.js and Sigma.js are dedicated to — and thus optimized for — network visualizations. On the complex-harder end of the spectrum, coding libraries like D3.js and Vega allow for a higher level of control over how data appears.[②] These libraries offer much more than network visualization capabilities. On the downside, they are not dedicated to network visualizations and therefore require a bit more work by humanist developers to get up and running. On the upside, however, they offer access to a bevy of other forms of data visualization, and thus allow project data to be viewed and explored in other complementary forms, such as Sankey diagrams, arc diagrams, and dendrograms, just to name a few.

①For more on each library, see Vis.js, https://visjs.org/, accessed October 23, 2020; Springy.js, http://getspringy.com/, accessed October 23, 2020; Sigma.js, http://sigmajs.org/,accessed October 23, 2020.

②For more on each library, see “D3: Data-Driven Documents,” D3.js, https://d3js.org/, accessed October 23, 2020; “A Visualization Grammar,” Vega, https://vega.github.io/vega/, accessed October 23, 2020.

```
const links = data.links.map(d => Object.create(d));
const nodes = data.nodes.map(d => Object.create(d));
const width = 500;
const height = 600;

const simulation = d3.forceSimulation(nodes)
    .force("link", d3.forceLink(links).id(d => d.id))
    .force("charge", d3.forceManyBody())
    .force("center", d3.forceCenter(width / 2, height / 2));

const svg = d3.create("svg")
    .attr("viewBox", [0, 0, width, height]);

const link = svg.append("g")
    .attr("stroke", "#999")
    .attr("stroke-opacity", 0.6)
    .selectAll("line")
    .data(links)
    .join("line")
      .attr("stroke-width", d => Math.sqrt(d.value));

const node = svg.append("g")
    .attr("stroke", "#fff")
    .attr("stroke-width", 1.5)
    .selectAll("circle")
    .data(nodes)
    .join("circle")
      .attr("r", 5)

node.append("title")
      .text(d => d.id);

simulation.on("tick", () => {
  link
      .attr("x1", d => d.source.x)
      .attr("y1", d => d.source.y)
      .attr("x2", d => d.target.x)
      .attr("y2", d => d.target.y);

  node
      .attr("cx", d => d.x)
      .attr("cy", d => d.y);
});

invalidation.then(() => simulation.stop());

return svg.node();
```


Figure 2 D3.js Code for a Force-Directed Graph①

For example, the CHCD utilizes the D3.js library to let users create customizable force-directed ego network graphs. D3.js ensures that these complex visualizations are controlled by a relatively small amount of JavaScript code. Figure 2 shows a basic example of the code for a D3 force-directed graph. This basic code accepts any correctly formatted data file as the input and automatically adjusts the graph layout to provide the best display of the data. Additional features and display options (such as customized coloring or sizing of nodes, dragging, panning, and zooming, etc.) can be added easily through additional attributes and short functions. The CHCD utilizes a very similar codebase for its data visualization.②

Using and developing such libraries, however, have one major drawback for digital humanists. While interactive visualizations for digital projects significantly lowers the learning threshold for non-technical users, it also places an additional burden on digital humanities projects. By and large, most humanists do not have a working knowledge of JavaScript or other application development languages. In the traditional humanities mode of the "lone researcher," such a project would be impossible unless the researcher happens also to have the requisite coding skills. Even if this were the case, there is always the matter of time, and many tenure review boards remain hesitant to recognize the worth and effort of such digital projects. As such, the easiest, and perhaps best, way to overcome this additional technical hurdle is for digital humanities projects to embrace an approach that partners more fully with students and scholars in computer science, graphic design, and other related fields.③

For project teams willing to take this leap, the bridge for the general public must be

---

①This example is adapted from Mike Bostock, "Force-Directed Graph," *Observable*, last modified November 15, 2017, https://observablehq.com/@d3/force-directed-graph.

②The CHCD also utilizes React.js for reasons that lie outside the scope of this article. Therefore, the code used in the CHCD is slightly less compact than the generic example given above.

③This approach is alive and well in places like the Yale's Digital Humanities Lab, The Center for Digital Humanities at Princeton, and the Humanities+Design Research Lab at Stanford University. The approach of these larger labs, however, is also possible in smaller, less well-funded research units.

a sturdy one. JavaScript libraries are so valuable because they can dramatically increase the degree of interactivity with a humanities project. While traditional digital humanities projects might utilize desktop software to create static images, JavaScript applications can be developed to allow end-users to create their own visualizations of network graphs that are interactive and highly customizable. This interactivity is possible because these code libraries preserve the visual effects of a graphical representation while its underlying data changes. This means that as end-users filter or alter the data that underlies dynamic JavaScript-powered network graphs, they are freed from having to continually make complex design choices. Thus, instead of creating static visualizations to tell one story, digital humanist developers can encourage interaction with project data and allow users to create visualizations which tell a multitude of stories. Of course, the hand of the research team would still be present. Nonetheless, by establishing semiotic cues, making distinct visual design choices, and carefully curating filter parameters, digital projects could allow end-users to explore the data under the careful — albeit somewhat invisible — guidance of the project team. Just as with written words, design choices are also a form of narrative, though they work in a much more indirect manner.[①] Still, the open-ended nature of data visualization offers the opportunity for end-users to discover and parse data in ways that the original research team may not have anticipated, thereby extending the project's usefulness and potential for novel insights.

The China Historical Christian Database as a Leap and a Bridge

While either of the above technologies could offer benefits to a project alone, they provide even more benefits when used together. The native graph environment of a graph database allows researchers to develop their databases more quickly and make it easier to convert data into formats that can be used by coding libraries to create interactive network visualizations. Together, these two technologies can drastically cut down on development timelines while also expanding the reach of any given project. As an example, the China Historical Christian Database (CHCD), currently under development, employs both technologies.[②] This paper will conclude with a brief discussion of the project's design and how graph databases and JavaScript data visualization libraries have been integral to achieving project goals.

---

①The narrative choices involved in data visualization design cannot be ignored. As Applen and Stephens have noted, "Both technical communication and digital humanities researchers have discussed the importance of maintaining a critical perspective on using computing tools and displaying data—for example, by adapting visualizations to specific audiences and rhetorical situations, analyzing the power relations embedded in computing technologies and code, and interrogating positivist assumptions about data and their relations." For more, see J. D. Applen, Sonia H. Stephens, "Digital Humanities, Middleware, and User Experience Design for Public Health Applications," *Communication Design Quarterly*, vol. 5, no. 3, 2017, pp. 3, 24-34.

②More information about the project and a preview of the current prototype of the project's online platform can be found at https://chcdatabase.com/.

The CHCD has the ambitious goal of mapping Christian presence in China from 1550 to 1950. It aims to achieve this goal by tracing Christian people, institutions, organizations, and events over these four centuries. More specifically, the CHCD synthesizes information from thousands of primary source documents in multiple languages to record tens of thousands of Chinese Christians and missionaries, as well as the schools, hospitals, missions, and churches which they frequented. In doing so, the project is creating a large treasure trove of historic data that integrates the spatial, temporal, and relational dimensions of the history of Christianity in China. However, this trove of data is not just about "religious" networks. As Nicolas Standaert has observed, the modern category of "religion" is often problematic for understanding the complex webs of cultural interaction, scientific exchange, diplomatic relations, and ritual life that come under the heading of "Christianity in China."[①] As such, the CHCD provides a resource which is valuable for anyone interested in fields such as sinology, religious studies, or intercultural studies, as well as various historical subfields. Of course, providing such a resource requires a great deal of human-power, but it also requires technological innovations which are commensurate with the scope and importance of the project. To that end, the project has adopted a three-pronged approach which aims to make the database possible and its data accessible to the large body of scholars and students who might find it useful.

First, the CHCD focuses on relational and geographic historical data and seeks to capture as wide a swath of material as possible. This is accomplished through the creation of a flexible graph database structure which connects people, institutions, and event nodes to specific geographic nodes. This mixture of node-types allows for multiple forms of spatial and network analysis and can be used by digital humanists to explore historical Chinese Christian networks from interpersonal, institutional, and geographic perspectives. The second prong of the CHCD approach is a focus on end-user accessibility. Rather than create a publicly available database that would only be usable by the digitally adept, this project has sought to bridge the digital gorge by creating an online platform that allows users to explore the CHCD data using interactive geographic, network, and statistical data visualizations. While more advanced forms of analysis will remain the purview of specialists who utilize more complex software, the online platform will be a way to introduce researchers and students to the toolsets and frameworks of digital humanities by allowing them to analyze project data via intuitive filters and preset visualization parameters. The final prong of the CHCD's approach is forming key partnerships between Western and East Asian research universities and archives to collect and analyze a rich diversity of historical data. While technological advances will be utilized to cull data when possible, this final prong represents the human-power that drives the project.

Two of these three prongs would be impossible without leveraging the two technologies outlined above. Currently, the CHCD has completed its initial proof-of-concept stage and is moving into full-scale development and implementation. This second

①See Nicolas Standaert, "Christianity as a Religion in China. Insights from the Handbook of Christianity in China: Volume One (635-1800)," *Cahiers d'Extrême-Asie,* 2001, no. 12, pp. 1-21.

stage has been made possible by generous private donations and a Digital Advancement Grant from the National Endowment of the Humanities. Although still at an early stage, the project has produced major learning outcomes. As the CHCD is so interested in connection, the team chose to utilize a graph database schema and used Neo4j graph platform to do so. This choice offered several advantages. The platform is well documented, uses a unique query language that is fast to learn, and offers a large suite of add-on plugins that make it easy to import and export data to various formats. Using this structure, the research team created and populated a proof-of-concept database that focused on the historical Christian presence in Shanxi province. This sample database included 1,763 nodes and 3,408 edges. Once the database was populated, the project team exported the data into JSON (JavaScript Open Notation) formatted files which could be used to power a public-facing, online JavaScript application.

Concurrent with the collection of data, the project team began building an online application using several well-known coding libraries, namely React.js, Leaflet, and D3.js. This online application was developed with the intention of allowing users to search and explore the Shanxi data through both geographic and network visualizations, as well as to grant users the power to pull up customized profiles for the major entities listed in the database. These profiles list all the pertinent information contained in the database on that entity. The network visualization portion of the platform was developed to allow users to create force-directed network graphs which depict customizable levels and types of connections within the database. This network visualization component was built entirely with the open-source D3.js coding library.

The achievements of this initial project phase were satisfactory and still serve as a source of potential knowledge creation. Yet, the most important finding of this exploratory phase was the timeline. Working as a team of three, the entire proof-of-concept project was able to be implemented within a 12-month period and using only a small amount of internal, institutional funding. Much of this funding consisted of undergraduate research assistants who primarily focused on data collection. While some outside consultations were instrumental, the bulk of the work was completed by a three-member team of humanists with specialties in Protestant Chinese history, Catholic Chinese history, and JavaScript development, respectively. Moreover, this project was completed without any major course releases, sabbaticals, or temporary reprieves from faculty duties. In short, these technologies allowed for the team to achieve an outsized degree of productivity and developmental efficiency without sacrificing traditional academic requirements or depending on enormous grant-funded budgets.

Additionally, this phase demonstrated that newer technologies make it far easier to combine a concern for in-depth humanities research with concerns about accessibility and end-user experience. From its inception, the project had to balance the wishes of both digital humanists who want rich, complex datasets and the more non-technical scholars who might desire more basic tools that allow them to explore the data. Rather than splitting the attention of the project team, this dual focus enriched the project. Indeed, by developing toolsets alongside the database, the potential impact of the project was kept at the forefront; the research team constantly had to consider hypothetical use-cases and structure the database schema and data-collection goals according to standards which went beyond the narrow confines of a single specialist. Moreover, the JavaScript application enabled the project team to quickly access its own data in a form that was more intuitive and aesthetically pleasing. This allowed the project team to engage in a more basic level of analysis, without having to write lengthy queries, and it helped to conceptualize questions which had not theretofore been considered.[①]

Conclusion: Leaping and Bridging the Digital Gorge

For humanists, leaping across the digital gorge can be a great undertaking. In dedicating time to learning new technologies and methodologies — such as social network analysis — digital humanists open a range of new possibilities for themselves and the field in general. Though leaping across the divide is a task in itself, digital humanities projects might do well to consider how their digital "leaps" can also become "bridges" that others in the scholarly community can use in order to cross the digital gorge. Graph databases and JavaScript development packages are powerful, flexible, and time-saving toolsets that can help humanists create projects that are more public-facing and readily understandable. They reduce the levels of abstraction needed to capture and display data, and they prioritize access, aesthetics, and interactivity for end-users. The China Historical Christian Database is an ongoing attempt that leverages these two technologies to create a humanities resource that is as academically rigorous and technologically sophisticated as it is accessible and intuitive to non-technical end-users. This dual focus, rather than an impediment, is an enriching approach that helps to keep the human at the center of the digital humanities. To borrow once more from the *Hutiaoxia*, if we are to leap across the digital gorge, it is imperative that we help as many others do the same as well.

（编辑：肖爽）

①In many ways, this focus on multiple end-users is in keeping with the observation made by Charlie Edwards: "[Digital Humanities] and its possible futures are likely to be shaped, delimited, or advanced by how [digital humanists] choose to design and build their conceptions of the user, and the extent to which all users can participate in that process." See Charlie Edwards, "The Digital Humanities and Its Users," *Debates in the Digital Humanities*, ed. Matthew K. Gold, Minneapolis, MN: University of Minnesota Press, 2012, pp. 213-232.