Library Carpentry: Software Skills Training for Library Professionals

James Baker

University of Sussex, UK
james.baker@sussex.ac.uk

Caitlin Moore

Sotheby's Institute of Art, UK
moorecaitlina@gmail.com

Ernesto Priego

City University London, UK
efpriego@gmail.com

Raquel Alegre

University College London, UK
r.alegre@ucl.ac.uk

Jez Cope

University of Sheffield, UK
j.cope@erambler.co.uk

Ludi Price

City University London, UK
ludi.price@gmail.com

Owen Stephens

Owen Stephens Consulting, UK
owen@ostephens.com

Daniel van Strien

Homerton University Hospital Foundation Trust
davanstrien@gmail.com

Greg Wilson

Software Carpentry, Canada
gvwilson@software-carpentry.org

Abstract

Librarians play a crucial role in cultivating world-class research and in most disciplinary areas today world-class research relies on the use of software. This paper describes Library Carpentry, an introductory software skills training programme with a focus on the needs and requirements of library and information professionals. Using Library Carpentry as a case study of the development and delivery of software skills focused professional development, this paper describes the institutional and intellectual contexts in which Library Carpentry was conceived, the syllabus used for the initial exploratory programme, the administrative apparatus through which the programme was delivered, and the analysis of data collection exercises conducted during the programme. As many university librarians already have substantial expertise working with data, it argues that adding software skills (that is, coding and data manipulation that goes beyond the use of familiar office suites) to their armoury is an effective and important use of professional development resource.

Key Words: capacity building; software skills; data; Library Carpentry

1. Overview

Librarians play a crucial role in cultivating world-class research and in most disciplinary areas today world-class research relies on the use of software (Hettrick, 2014). Established non-profit volunteer organisations such as Software Carpentry (Wilson, 2010) and Data Carpentry (Teal et al., 2015) offer introductory research software skills training with a focus on the needs and requirements of research scientists. This paper describes Library Carpentry, a comparable introductory software skills training programme with a focus on the needs and requirements of library and information professionals. In its initial exploratory run, Library Carpentry took the form of four three-hour sessions held at the City University London Centre for Information Science across four successive Monday evenings in November 2015. These sessions attracted 59 participants from 14 institutions in London and its environs. Subsequently, there has been interest in Library Carpentry both from outside the United Kingdom and from individuals working in comparable professional contexts outside librarianship (such as the archives sector).

This paper presents Library Carpentry as a case study on the development and delivery of software skills focused professional development. It begins by describing the institutional and intellectual contexts in which Library Carpentry was conceived. It goes on to describe the syllabus used for the initial exploratory programme, the administrative apparatus through which the programme was delivered, and the analysis of data collection exercises conducted during the programme. It concludes with a discussion of what went well, what might need adapting, and future plans. As many university librarians already have substantial expertise working with data, it argues that adding software skills (that is, coding and data manipulation that goes beyond the use of familiar office suites) to their armoury is an effective and important use of professional development resources.

The paper has three aims:

1) To describe how and why the approaches of and resources created by non-library specific software skills training (Software Carpentry, Data Carpentry, and Programming Historian) and existing library specific programmes (British Library Digital Scholarship Training Programme, Data Scientist Training for Librarians) were adapted to create the Library Carpentry exploratory run and lesson materials.

2) To present data on software skills in university libraries that was collected during this exploratory run.

3) To stand as one of a number of activities that builds the foundations of a distributed community model for embracing and sustaining software skills in the library and information profession.

2. Context

2.1. Intellectual contexts

Jonathan Rochkind (2007) argued in his Editorial Introduction to the inaugural issue of The *Code4Lib Journal* that:

> *"This is a decisive time for libraries. In the changing social and technological environment, libraries must adapt to fulfill their missions and satisfy their users. Library technology is acutely involved in this adaptation. Digital services, content and tools have become a part of nearly every aspect of library operations. The "digital library" is here–if you work in a library, you probably work in a digital library" (n.p.).*

Modern librarians have a wide range of roles (RLUK, 2014). Within many of these roles there is potential for applying programming and IT skills, which allow for automation of tasks and the manipulation of data, whether aggregating and harmonising usage statistics from diverse sources or bulk editing of metadata in library catalogues. The research data management and open access services now offered by academic libraries provide additional motivation for librarians to develop such skills (Goben & Raszewski, 2015). Conceptual questions raised when thinking about software design and programming skills often overlap with issues 'traditionally' considered by librarians, potentially leading to new approaches in both. Nevertheless, while the "digital library" has since 2007 fundamentally altered what university libraries do and are for, the integration of software skills into the work of library and information professionals has remained uneven (Dalziel, 2008; Kim, 2011). As Andromeda Yelton (2015) notes in her American Library Association Library Technology Report 'Coding for Librarians: Learning by Example', significant social and political barriers remain:

> *"Many library coders spend a significant amount of time trying to cultivate buy-in, educate their colleagues about technology, or work against siloed organizational structures as they produce inherently cross-departmental work"* [p. 22].

That "buy-in" can be inhibited by a conflation of the desire to automate processes and reduce human involvement with cuts in staffing levels and the resulting increased workloads on individuals (Wilkinson, 2013). In turn, the emergence of software skills among library and information professionals can contribute to existential doubt, to concerns that the traditional and valued skills are no longer enough to justify employment (Wilkinson, 2013).

One factor that has provided fresh impetus for librarians to develop software skills which cut across their existing data management, user support, and research collaboration roles, is the recent emergence of the Digital Humanities, a field in which library and information scientists feature prominently (Baker et al., 2014a; Baker, Williams, Russell, & Rosenblum, 2014b; Keener, 2015). A project funded by the UK public body Jisc made a series of recommendations to Higher Education Institutions or Higher Education clusters looking to build capacity for enabling complex analysis of large-scale digital collections by their non-computationally trained humanities researchers. In particular, it suggested investment 'in training library staff to run these initial queries in collaboration with humanities faculty, to support work with subsets of data that are produced, and to document and manage resulting code and derived data' (Terras et al., 2016).

2.2. Practical contexts

In response to these contexts, Library Carpentry was conceived in Autumn 2014 as an attempt to facilitate the cultivation of software skills in the library and information science community. It built on three activities undertaken by the Library Carpentry PI (Baker) whilst a Digital Curator at the British Library between March 2013 and August 2015. The first activity was his attendance at a Software Carpentry workshop (Greenwich, London, October 2013). Subsequent discussions with Greg Wilson (then director of Software Carpentry) revealed the prominence of library and information science professionals among the cohort of non-scientists who attend Software Carpentry

workshops. This indicated a potential appetite for in-person software skills training tailored to librarians. The second activity was the experience of writing lessons for the open-access, peer-reviewed journal *The Programming Historian* (Crymble et al., 2016). The observation of the crossover between historians and librarians in the use of and contribution to its lessons indicated an interest among librarians in structured and maintained online learning materials on software skills. The third activity was his role as a session lead and programme coordinator for the British Library's Digital Scholarship Training Programme, a hands-on practical training programme designed for British Library staff and delivered as one-day on-site workshops. The programme launched in November 2012 and covers topics from communicating collections and cleaning up data to command line programming and geo-referencing, some of which are available for self-directed learning (Baker, 2014). When presenting this activity to librarians outside of the British Library, it was clear that demand existed in the sector for comparable in-person training programmes.

In light of the contexts and observations described, Library Carpentry had three aims upon inception:

1) To blend non-library specific software skills training (Software Carpentry, Data Carpentry, and Programming Historian) with existing library specific programmes (British Library Digital Scholarship Training Programme, Data Scientist Training for Librarians; Erdmann, von Alstine, Eslao, Durocher, & Wicks 2014) into a public offering aimed at library and information professionals seeking an introduction to software skills.
2) To collect data on software skills in university libraries, organisations that play a crucial role in cultivating world-class research that is increasingly reliant on software.
3) To build the foundations of a distributed community model for embracing and sustaining software skills in the library and information profession.

3. Programme administration

Library Carpentry was conceived in Autumn 2014 as a planned activity for Baker's then proposed Software Sustainability Institute Fellowship. Use

of the 'Carpentry' name was intended to indicate an alignment with core aspects of Software Carpentry, namely: training delivered over four segments; an emphasis on open software and data; encouragement of attendance as groups rather than as individuals to support peer learning; the use of multiple trainers to troubleshoot issues; and training materials published online. The use of the 'Carpentry' moniker was approved by Software Carpentry. Upon being selected for a Fellowship, an outline for Library Carpentry was pitched to the Software Sustainability Institute and a £1,000 budget set aside to cover travel costs for trainers, room hire, and refreshments for attendees. At this stage it was anticipated that 20–30 attendees would be accommodated on a single day. In early 2015, a public call for input into shaping Library Carpentry was issued (Baker, 2015a) and discussions with individuals and groups already involved in relevant pedagogical initiatives (see 'Context') took place. As a result of this work, a decision was made for the Library Carpentry exploratory programme to take the form of a series of short events delivered in late-2015 based around a syllabus comparable to Software Carpentry and Data Carpentry, but with examples, data, exercises that replicated library practice rather than scientific research. The syllabus would include the Unix shell (a command line user interface),[1] Git (a version control tool) and OpenRefine (an interactive data cleanup tool). City University London Centre for Information Science were approached with a view to joining the project and the discussions that followed led to the decision to host Library Carpentry at City University London in alignment with their Master's degree in Library and Information Science. This had two significant consequences. First, a programme of four sessions to take place between 17:30 and 20:30 on consecutive Monday evenings was agreed upon. Doing so meant diverging from the two-day format preferred by both Software Carpentry and Data Carpentry, effectively distributing each half-day session over a four-week period. Second, as room hire costs were waived as part of this arrangement, the budget was in turn adjusted to accommodate 50 attendees.

Having settled on a venue, draft syllabus, and format, in April 2015 a call for participants and volunteers was issued (Baker, 2015b). The response to this established both the demand for Library Carpentry and a group of individuals willing to offer their time to develop and/or deliver the syllabus. In order to maximise the pedagogical benefits of peer learning between sessions (Wilson, 2013), respondents from the first group were asked to bring a cohort from their institution to Library Carpentry.

In Spring/Summer 2015 the programme website and syllabus were developed and hosted on public GitHub repositories. This platform centralised our work and made it transparent: for example, GitHub Issues trackers were used to manage lesson development. Although using GitHub for this purpose is not frictionless – its steep learning curve introduces potential inclusivity issues (Crymble, 2015) – it nevertheless provided a useful platform on which to promote, iterate, and deploy Library Carpentry. In July 2015 lesson plans for each session were in place and proposed attendees were invited to confirm their registration. Due to high demand, institutions were limited to four attendees, with a waiting list for those the programme was unable to support due to capacity restrictions.

During programme delivery in November 2015, attendees worked on their own personal computers. For sessions one, two, and three printed handouts were provided to guide learning. Refreshments and snacks were provided at every session and as the sessions were held in the evening attendees were encouraged to bring more substantial meals.

At the beginning of each session attendees were asked to make a name badge for themselves, to self-assess their confidence level on the topic at hand, and to clearly display this on their badge (see 'Attendance and Feedback'). Each session was coordinated by a session lead and supported by between two and four helpers who undertook administrative tasks and/or were available to troubleshoot problems arising in the room. At the close of each session, attendees were directed to any actions they needed to take before the next session, typically software installation or downloading data. Between sessions matters arising, clarifications, and installation problems were handled via both GitHub Issues and email.

4. Syllabus

4.1. Overview

The Library Carpentry syllabus delivered in during the exploratory programme was aimed at beginners, required no prerequisite knowledge, and was tailored to align with the needs and requirements of library and

information science professionals through the use of relevant examples, data, and exercises. The programme was split into four parts, each of which was delivered in a single three-hour session that was sub-divided into three sections each roughly 45 minutes in length (timetables used for each session are noted below, though note that they do not correspond exactly to what happened at each session). Dependencies between each session were minimised. This reduced the need for refreshers and meant that attendees who were unable to attend a session were not disadvantaged.

Two sets of learning outcomes informed the development and delivery of the syllabus. First, that attendees understood the value of command line interfaces, regular expressions, plain text file formats and consistent naming conventions, and would be equipped by Library Carpentry to apply these in their own professional context. Second, that attendees understood the importance of openly licensed software with strong and diverse user communities and how these characteristics could support both self-directed learning and professional development. To support the delivery of these learning outcomes, the syllabus was also built around use cases with clear relevance to library practice and functionality found in multiple software tools. The latter was particularly important in syllabus construction. The choice to offer a session that focused on the interactive data cleanup tool OpenRefine, for example, was made not only because OpenRefine is a powerful software tool for manipulating tabulated data that is well liked by librarians and is underpinned by a strong user community, but also because OpenRefine queries are built on both regular expressions and programming languages (Jython, Clojure, and the bespoke General Refine Expression Language) that introduce learners to clear, well-documented, and well-constructed programming syntax.

4.2. Sessions

Session one began with an introduction to basic programming concepts. It drew upon a wide literature, with elements adapted from British Library Digital Scholarship Training Programme lesson materials. Thereafter attendees were asked to reflect on words and phrases associated with programming, code, and software that they believed they would benefit from

knowing more about. Attendees were organised into small peer groups to encourage honesty, peer support, and a better understanding of confidence levels among their peers. Discussions in these small groups were collated by the session lead and an open discussion held that aimed to underscore the learning outcomes of Library Carpentry, to flag areas that the sessions would not cover and where help on those could be sought, and to ameliorate concerns over prerequisite knowledge. The session concluded with an introduction to regular expressions. This included an exercise designed to both help attendees understand what regular expressions do and to encourage attendees to build their own regular expressions. These were conducted on paper with personal computers used to check answers, resolve queries, and experiment.

Week One Timetable

> Arrival and Introduction (30 minutes)
> Jargon Busting exercise (40 minutes)
> Foundations presentation (35 minutes)
> Regular Expressions practical (45 minutes)

Session two covered the Unix shell. It was adapted from Software Carpentry and the Programming Historian lesson materials. In this hands on-session, attendees were introduced to the basic commands required to navigate the filesystem, to count (wc) and mine (grep) metadata for journal articles in tabular form, and to clean and manipulate a text for the purposes of counting words. A demonstration of how to run Named Entity Recognition software from the Unix shell sought to underscore the wider applications of the interface and commands introduced. The session concluded with some advice on and recommendations of resources that can support learning and use of the Unix shell and programming languages.

Week Two Timetable

> Arrival and Introduction (15 minutes)
> Unix Shell Basics (40 minutes)
> Counting and Mining in the Unix Shell (50 minutes)
> Cleaning and Transforming in the Unix Shell (45 minutes)

Session three covered version control with Git. It was adapted from Software Carpentry lesson materials. The session began by introducing attendees to

the concept of version control, to the terms used to control versions in Git, and how to collaborate and publish Git repositories using GitHub. A pen and paper group exercise was used to reinforce the latter. Thereafter, attendees acquired hands-on experience of using Git in the unix shell to version control a file and explored the use of GitHub within a collaborative versioning workflow. Finally, attendees were encouraged to consider use cases for Git/GitHub workflows including using GitHub Pages to create simple websites to host a blog or details of an event.

Week Three Timetable

Arrival and Introduction (15 minutes)
Introducing Git and GitHub (40 minutes)
Exercise and Using Git (60 minutes)
Git and GitHub Use Cases (30 minutes)

Session four covered cleaning and transforming data in OpenRefine. It was adapted from British Library Digital Scholarship Training Programme lesson materials. After a brief introduction to the history of OpenRefine, its purpose, and the community that supports and maintains it, attendees were led through a series of exercises that cleaned and normalised a real-world dataset from an Institutional Repository. The session concluded with some examples of more advanced uses of OpenRefine (such as interactions with web based services) and a discussion of the capabilities of OpenRefine, its appropriateness for certain use cases, and how to integrate it into existing workflows.

Week Four Timetable

Arrival and Introduction (15 minutes)
Introduction to OpenRefine (15 minutes)
Basic OpenRefine Functions (80 minutes)
Advanced OpenRefine Functions (35 minutes)

Each of the four sessions began by situating what was to follow within the wider context of the programme, and sessions two, three, and four began with an articulation of which of the basic programming concepts encountered in session one would be covered. Every session provided opportunities for attendees to ask questions of the session lead and helpers both during and at the conclusion of the session.

Training materials for each session were available to attendees on Github and have been subsequently archived: the materials available include lesson plans, slides, data used in exercise, handouts, and answer sheets (Baker, 2016).

5. Attendance and feedback

One of the aims of the Library Carpentry exploratory programme was to collect data on software skills in university libraries. This took the form of gathering attendance data and feedback from attendees. Starting with attendance, across the four sessions, a total of 59 individuals from the library and information science community attended Library Carpentry, nine of whom were trainers and/or supported the programme in an administrative capacity (often, in addition to attending as learners). 15 organisations were represented (see Table 1).

5.1. Feedback: self-assessment

During each Library Carpentry session feedback was gathered from attendees. It was anticipated that this data would support understanding of both whether the syllabus had been pitched correctly so as to deliver the stated learning outcomes and the potential barriers to building a distributed community model for embracing and sustaining software skills in the library and information profession. Three mechanisms were used to gather feedback from attendees.

Table 1: Composition of Library Carpentry attendance by location and affiliated organisation.

Location	N	Affiliated Organisation
London	8	Birkbeck University of London, British Film Institute, British Library, City University London, Imperial College London, UCL, University of London Computing Centre, The Wellcome Trust
South East of England	3	Cambridge University, University of Reading, University of Sussex
Rest of England	3	De Montfort University, University of Leicester, University of Sheffield
International	1	Software Carpentry

First, at the beginning of each session, attendees were asked to self-report their skill level based on the topic at hand. In order to do this, attendees were asked to complete the sentence 'In relation to the topic this week I know...' with one of four options: 'Nothing!' (1), 'A little!' (2), 'Lots!' (3), and 'Lots and Lots!' (4). The rationale for this exercise was threefold: first, it enabled the session leads to gain a quick sense of the confidence in the room each week; second, as attendees were asked to clearly display this on their badge, other attendees were able to identify people nearby in the room who might be able to support them if they got stuck; and third, it provided data on what software skills attendees thought they had before attending Library Carpentry.

The data displayed in Table 2 shows that at the beginning of sessions two, three, and four more than three-quarters of attendees reported knowing nothing about the topic at hand. Attendees were more knowledgeable on concepts (session one), suggesting the existence of a basic set of knowledges in the library and information science community upon which software skills could build. OpenRefine (session four) provided the lowest self-assessment, although this was expected given that it is specialist, bespoke software. Session three (which covered Git) had the lowest attendance. This data suggests that the integration of software skills into the work of library and information professionals remains uneven.

5.2. Feedback: anticipated use of Library Carpentry

The second mechanism by which feedback was collected took place at the beginning of the third session, the programme mid-point (for this data see

Table 2: Attendee self-reporting on skill level at the start of each week.

In relation to the topic this week I know…	Nothing!	A little!	Lots!	Lots and Lots!!
Week One	22	17	2	0
Week Two	31	10	2	2
Week Three	27	7	0	0
Week Four	32	7	0	0
Mean (n)	*28*	*10.25*	*1*	*0.5*

week two materials; Baker, 2016). On this occasion attendees were asked to articulate ways in which what they had learnt could be used in their daily practice. Attendees were asked to write on a sticky note a scenario in which they could imagine that they or a member of their team might be able to use what they had learnt at Library Carpentry in the workplace.

27 attendees responded. These responses were grouped by theme to identify clusters. Analysis of these responses shows a strong cluster (n=11) that anticipated using software skills to improve their search capabilities:

- 'Ability to combine large documents and search across them'
- 'Find all VERSIONS of a word (misspelt due to dyslexia) within a database'
- 'Possibly pulling lines out of a horrid csv file our open access system produces'.

Concurrent with these were clusters of responses that anticipated using skills learnt at Library Carpentry to manipulate large scale data (n=7) or to review library data (n=10):

- 'Use regular expressions to help create review files of metadata'
- 'Comparing e-book data (ie. cat records) with physical records. Overlaps? Gaps? Subject heavy in one or other?'
- 'Cleaning up data after exporting it from a) LMS b) Institutional Repository'
- 'Reformatting shelfmarks (title, author, etc) in an exported database of item/bib records. Quite possibly in transforming data to different systems e.g. alt LMS etc'.

Attendees also reported anticipating using the skills learnt at Library Carpentry to better understand software related possibilities in their workplaces (n=4):

- 'Understanding arguments/possibilities for software'
- 'BUT MOSTLY—talking to the Tech people in clearer, more useful ways'
- 'Given me confidence, reminded me what I know/can learn so I can make online 'sandbox' environment to test tools & share courses'.

Smaller overlapping clusters of potential uses of the skills learnt included research support (n=4), local capacity building (n=2), and task automation (n=2). This data suggests demand in the library and information profession for acquiring software skills.

5.3. Feedback: barriers to learning software skills

The third feedback mechanism took place at the close of the final session (for this data see week four materials; Baker, 2016). On this occasion attendees were asked to articulate what they might need to pass on to colleagues the skills they had learnt at Library Carpentry. Again they were asked to write their responses on a sticky note. 18 attendees responded. These responses were grouped by theme to identify clusters. Analysis of these responses shows time (n=6) and practice (n=7) as important requirements to passing on skills learnt at Library Carpentry. A smaller cluster (n=4) reported the need for more worked examples:

- 'Practice with data I am familiar with—applying each week's lessons practically'
- 'I'd be really keen but think I'd need time and actually apply this to something practically myself (I am hoping to do this) to feel more confident in it (+ seeing other people's practical applications too)'
- 'Any websites that act as a simple reminder for me + to give others'.

Organisational barriers to passing on skills learnt were reported. These were described in two forms. One group (n=3) reported having insufficient IT permissions to apply the skills developed at Library Carpentry in their workplace. A second more nebulous group (n=4) reported that their organisational culture was either unsupportive or lacked the communities they needed to embed software skills in their practice:

- 'An appropriate channel: knowing the people/group to disseminate to'
- 'some of my colleagues are not very technologically adept, so not sure how much can pass on', 'Might be able to sell OpenRefine

to colleagues but the organisation I work for hates Open Source Software :('
- 'I certainly need more time to practice and a community where I am not afraid of asking stupid questions'.

This data suggests that in spite of demand for software skills, the integration of core software skills into the work of library and information professionals is inhibited by significant cultural, social and organisational barriers.

5.4. Review

Taken together, these three sets of data (drawn, it must be noted, from a self-selecting audience) point to three provisional findings and recommendations:

1) Library and information science professionals both value the acquisition of software skills as part of their professional development and report a low competency in such skills (findings corroborated by two attendee blogs that reflected on attending Library Carpentry: Playforth, 2015; Sykes, 2016). More work is needed corroborate these findings in an international context.
2) Library and information science professionals report undertaking activities that could be improved by their acquisition of software skills. More work is needed to map these activities to the roles performed by library and information professionals.
3) Library and information science professionals face various challenges to acquiring software skills and to embedding those skills in their workplace. More work is needed to tease out commonalities between these challenges.

An organisational challenge not captured by these feedback mechanisms was one of timing. Taking into account the target audience and the host institution, it was important to strike a balance between fulfilling everyday responsibilities and the need for ongoing professional development. Complementing the positive feedback regarding content and delivery, some participants provided informal negative feedback that the sessions were too

long, especially as they took place after a day of work. The pragmatic challenge for organisers and participants was that this time of the day was the most feasible for information professions with core working hour responsibilities and stretched professional development capacity. Work that builds on the initial exploratory run of Library Carpentry must remain attentive to this important professional dynamic.

6. Next Steps

Starting with what went well, the decision to spread Library Carpentry across four weekly evening sessions (as opposed to Software Carpentry's two-day format) delivered significant benefits: the intensity of learning was distributed, the time between sessions was used to reinforce skills and field queries, nascent peer support communities were built (especially through the logging of issues on GitHub and the use of the Twitter hashtag #librarycarpentry during and after sessions; Priego, 2016), the syllabus was revised mid-programme based on the progress of previous sessions, and a wider pool of expert trainers could be drawn upon (that is, the individual trainers and helpers were unlikely to all have been available simultaneously). The decision to encourage institutions to bring a cohort to Library Carpentry across a four-week period was a notable success: attendees reported discussing sessions during journeys home and subsequently in the office, thus deepening knowledge acquisition and facilitating self- and peer- learning. Attendance remained consistent throughout, indicating that attendees were sufficiently satisfied with the learning experience to return the next week.

As the lessons were new – if adapted in most cases from existing resources – areas were encountered where revisions focused on clarity and concision were required. The necessity to troubleshoot problems encountered by a large group meant that all four lessons fell behind schedule, and most did not cover the entirety of the planned lesson content. A higher ratio of trainers and helpers to attendees might alleviate this problem, though as many of the problems encountered were not lesson related but caused by attendees working on their own personal computers (such as failed WiFi logins, software installation faults, and operating systems interoperability issues) an IT training suite might also ensure more lesson content is covered. That said,

we observed clear benefits of having learners able to leave a training session with a configured environment on their own personal computers: that is, they could continue developing their skills immediately (Wilson, 2013). Better support for learners who wish to reinforce learning between sessions could be offered by providing exercises for completion between lessons.

Finally, attendees struggled most with Git workflows and terminology, this in spite of attendees using GitHub Issues to discuss or ask questions from week one (see for example the especially large number of OpenRefine use case queries raised on GitHub Issues after prompting from the session four lead   https://github.com/LibraryCarpentry/week-four-library-carpentry/ issues). Introducing Git through a GUI such as GitHub Desktop might be preferred, although this would reduce the opportunity to reinforce learning of shell commands. Note however this is a finding of comparable training programmes and is a reason for Data Carpentry not teaching Git and GitHub.

7. Conclusion

The future plans for Library Carpentry, however, go beyond iterating lesson content. Library Carpentry set out to build the foundations of a community model for embracing and sustaining software skills in the library and information profession, with the intention that this model would emerge from the experience of delivering the initial exploratory programme. Spreading the programme over four weeks had the unexpected benefit of alerting a wider range of non-attendees to Library Carpentry, a number of whom reached out during and since Library Carpentry to find out more and to enquire into running iterations of Library Carpentry in their local area.

Since the Library Carpentry exploratory run, 12 further workshops have since been organised in 7 countries across 4 continents. Each workshop has adapted the lesson materials and timings to suit the needs of requirements of their local audience. Alongside and as a result of these workshops, the Library Carpentry lesson materials have been substantially revised and improved. During the Mozilla Science Lab Global Spring (2–3 June 2016), a team from the USA, Australia, Canada, UK, the Netherlands, and South Africa developed lesson materials, added a new lesson on SQL (a relational

database management language), assigned administrative roles required to support a distributed management and maintenance structure, and republished the materials using the Data Carpentry lesson template (Weaver, 2016). Building on the momentum created by these medium-term successes, we hope that in the long-term Library Carpentry will develop a sustainable model comparable to Software Carpentry and Data Carpentry – that is, towards an evolving set of centralized lessons delivered globally by individuals who have undergone instructor training, the latter in particular having proved crucial to building communities around both Software Carpentry and Data Carpentry. To achieve this, we must first note that Software Carpentry and Library Carpentry have different audiences: the former being scholars (and primarily scientists) who self-identify with the benefits of developing their own software skills in relation to their research; the latter being professionals who self-identify with the benefits of developing their own software skills in relation to their organisational needs. With this in mind, and in lieu of substantial funds, the next step is to establish Library Carpentry along a distributed model, to create a body of materials that libraries as organisations can drawn on, adapt, and reuse as appropriate in their local contexts, thereby achieving medium-term sustainability through an organisation to organisation community. To achieve this, investment priorities for future work are:

- Evaluate the short- to medium-term benefits of attending Library Carpentry workshops to both attendees and their libraries through semi-structured interviews with attendees.
- Develop a set of resources to enable Library Carpentry attendees to pass on software skills in their libraries. It is anticipated that these resources would be predicated on the idea that the best way to reinforce your own software skills is to teach others.
- Disseminate the results of that evaluation and the resources developed in appropriate venues.

This case study has described how the Library Carpentry exploratory programme was conceived and delivered, the syllabus of software skills training materials that were used and subsequently developed, the analysis of data collection exercises that were conducted, and developments that have occurred in subsequent months. The Library Carpentry exploratory programme and the activities that have built on it confirm that there is demand,

appetite and a will among library and information professionals to acquire software skills. As librarians and software skills are both vital components of world-class research, Library Carpentry is a timely intervention into the role of librarians in the research lifecycle.

Acknowledgements

References

Baker, J. (2014). *British Library Digital Scholarship Training Programme: a round-up of resources you can use.* British Library Digital Scholarship. Retrieved October 20, 2016, from http://britishlibrary.typepad.co.uk/digital-scholarship/2014/10/british-library-digital-scholarship-training-programme-round-up-of-resources-you-can-use.html.

Baker, J. (2015a). *What would Library Carpentry look like?* British Library Digital Scholarship blog. Retrieved October 20, 2016, from http://britishlibrary.typepad.co.uk/digital-scholarship/2015/02/what-would-library-carpentry-look-like.html.

Baker, J. (2015b). *Library Carpentry: call for volunteers, call for participants*. British Library Digital Scholarship blog. Retrieved October 20, 2016, from http://britishlibrary.typepad.co.uk/digital-scholarship/2015/04/library-carpentry-call-for-volunteers-call-for-participants.html.

Baker, J. (2016). *Library Carpentry. Weeks One-Four. November 2015.* figshare. Retrieved October 20, 2016, from https://dx.doi.org/10.6084/m9.figshare.3082990.v2.

Baker, J., Bourg, C., Gil, A., Hettel, J., Lindblad, P., Miller, L., & Stack, P. (2014a). Methods for empowering library staff through digital humanities skills'. Workshop given at Digital Humanities 2014, Lausanne, Switzerland, July 7 – 12, 2014. Retrieved October 20, 2016, from https://dh2014.files.wordpress.com/2014/04/dh-2014-workshop-010.pdf.

Baker, J., Williams, H., Russell, J., & Rosenblum, B. (2014b). *Teaching digital humanities in the library*. Talk given at Data Driven: Digital Humanities in the Library, Charleston, SC, June 20–22, 2014.

Crymble, A. (2015). How can we make the PH more friendly for women to contribute?. *The Programming Historian, 152*, n.p. Retrieved October 20, 2016, from https://github.com/programminghistorian/jekyll/issues/152.

Crymble, A., Gibbs, F., Hegel, A., McDaniel, C., Milligan, I, Taparata, E., & Wieringa, J. (Eds.) (2016). *The programming historian.* (2nd ed.). Retrieved from http://programminghistorian.org/.

Dalziel, K. (2008). *Why every library science student should learn programming.* nirak.net. Retrieved October 20, 2016, from http://nirak.net/2008/12/why-every-library-science-student-should-learn-programming/.

Devenyi, G.A., Koch, C., & Srinath, A. (Eds) (June 2016) *Software Carpentry: The Unix Shell*. Version 2016.06. Retrieved October 21, 2016, from https://github.com/swcarpentry/shell-novice, http://dx.doi.org/10.5281/zenodo.57544.

Erdmann, C., von Alstine, C., Eslao, C., Durocher, M., & Wicks, S. (2014). Data scientist training for librarians. International Data Curation Conference 2014, San Francisco, California, February 24–27, 2014. Retrieved from http://www.dcc.ac.uk/sites/default/files/documents/IDCC14/215DST4L-idcc_14.pdf.

Goben, A., & Raszewski, R. (2015). Research data management self-education for librarians: a webliography. *Issues in Science and Technology Librarianship*, Issue 82, n.p. Retrieved October 20, 2016, from http://dx.doi.org/10.5062/F4348HCK.

Hettrick, S. (2014). *It's impossible to conduct research without software, say 7 out of 10 UK researchers.* Software Sustainability Institute. Retrieved October 20, 2016, from http://www.software.ac.uk/blog/2014-12-04-its-impossible-conduct-research-without-software-say-7-out-10-uk-researchers.

Keener, A. (2015). The arrival fallacy: collaborative research relationships in the digital humanities. *Digital Humanities Quarterly*, *9*(2), n.p. Retrieved Octoer 20, 2016, from http://www.digitalhumanities.org/dhq/vol/9/2/000213/000213.html.

Kim, B. (2011). *Why not grow coders from the inside of libraries?* Library Hat. Retrieved October 20, 2016, from http://www.bohyunkim.net/blog/archives/1099.

Playforth, C. (2015). *Why the information profession needs Library Carpentry*. Software Sustainability Institute. Retrieved October 20, 2016, from http://software.ac.uk/blog/2015-12-03-why-information-profession-needs-library-carpentry-0.

Priego, E. (2016). *An Archive of #LibraryCarpentry [04/11/2015 11:48:10 to 25/02/2016 07:42:45 GMT].* figshare. Retrieved October 20, 2016, from https://dx.doi.org/10.6084/m9.figshare.3081478.v1.

RLUK. (2014). *Powering Scholarship: RLUK Research Libraries UK Strategy 2014–17.* Retrieved October 20, 2016, from http://www.rluk.ac.uk/wp-content/uploads/2014/02/RLUK-Strategy-2014-online.pdf.

Rochkind, J. (2007). Editorial Introduction — Issue 1, n.p. *The Code4Lib Journal*, 1, 2007. London: Research Libraries UK. Retrieved October 20, 2016, from http://journal.code4lib.org/articles/39.

Sykes, T. (2016). *C: Print "Library Carpentry". Making connections with code*. Code and the Librarian. Retrieved Octobre 20, 2016, from https://codeandthelibrarian.wordpress.com/2016/01/08/library-carpentry-poster/.

Teal, T., Cranston, K., Lapp, H., White, E., Wilson, G., Ram, K., & Pawlik, A. (2015). Data Carpentry: workshops to increase data literacy for researchers. *International Journal of Digital Curation, 10*(1), 135–143. Retrieved October 24, 2016, from http://dx.doi.org/10.2218/ijdc.v10i1.351.

Terras, M., Baker, J., Hetherington, J., Beavan, D., Welsh, A., O'Neill, H., …, Farquhar, A. (2016). *Enabling complex analysis of large-scale digital collections: Humanities tesearch, high performance computing, and transforming access to British Library digitalcCollections*. Digital Humanities 2016, Kraków, Poland, July 12–16, 2016.

Weaver, B. (2016). *Updating Library Carpentry*. Software Carpentry blog. Retrieved October 20, 2016, from http://software-carpentry.org/blog/2016/06/library-carpentry-sprint.html.

Wilkinson, L. (2013). *Is coding an essential library skill?* Sense and Reference. Retrieved October 20, 2016, from https://senseandreference.wordpress.com/2013/03/08/is-coding-an-essential-library-skill/.

Wilson, G. (2010). *Software Carpentry. Teaching basic lab skills for research computing*. Blog. Retrieved October 24, 2016, from http://software-carpentry.org/.

Wilson, G. (2013). *Software Carpentry: lessons learned*. Retrieved October 20, 2016, from https://arxiv.org/pdf/1307.5448v2.pdf.

Yelton, A. (2015). Coding for librarians: learning by example. *Library Technology Reports, 51*(3), 1–30. Retrieved October 20, 2016, from https://journals.ala.org/ltr/article/view/5675/7040.

Note

---

[1] As the Software Carpentry Unix Shell lesson describes: 'The most popular Unix shell is Bash, the Bourne Again SHell (so-called because it's derived from a shell written by Stephen Bourne). Bash is the default shell on most modern implementations of Unix and in most packages that provide Unix-like tools for Windows' (Devenyi, Koch, & Srinath, 2016). Popular examples of 'modern implementations of Unix' are Mac OS X and Linux operating systems such as Ubuntu.