Research Proposal

- PhD student: Jaume Nualart Vilaplana
- Degree: Doctor of Philosophy in Communication
- Faculty: Arts and Design (FAD)
- Institution: University of Canberra (UC)

B) Provisional title:

"Artisanal software development for text visualisation in real scenarios".

C) Abstract

This practice led research project studies the practical and applied visualisation and exploration of texts. Practical because the main outputs —the artifacts— of the research, will be a number of developed tools published under free licenses; applied because the goal of this project is to demonstrate the utility of the tools developed to visualise and explore real world texts.

While data visualisation seems like a useful and necessary tool to handle the digital environment, few visualisation and exploration tools are used in real world cases. This project aims to address this gap between great ideas developed and few implementations of text visualisation tools.

The methodology includes being prepared to establish strong relationships between the developer(s) and the client(s), and the investigation of phases of the process: design, development, evaluation, and implementation.

The methodology of development is a customised tool design, accompanied by personal motivation from the developer for each software project. In addition to creating software, ancillary aims include:

- Engage/encourage groups of people to work with text visualisation tools in real scenarios.
- Propose a methodology for the successful implementation of text visualisation tools.
- Help to identify cases in which text visualisation can be adopted and implemented.

D) Final Submission:

- Artifacts (75%).
- Written Exegesis (25%).

E) Supervisory Team Primary Supervisor:

- Supervisor: Dr Mitchell Whitelaw.
- Secondary Supervisor: Dr Stephen Barrass.
- NICTA Supervisor: Dr Wray Buntine.

Contents

1 Introduction

My research project is a creative production PhD which aims to develop several text visualisation software tools and apply them to real world scenarios. As set out below, data visualisation is a multidisciplinary field and the Faculty of Arts and Design (FAD) at UC is an ideal place.

My scholarship is kindly funded by the Machine Learning Research Group (MLRG) at National Information and Communication Technology of Australia (NICTA). The combination of FAD and MLRG allows me to draw on the expertise of artists, writers and digital humanities researchers, as well as mathematicians, statisticians and computer science researchers. This context is as multidisciplinary as the research project is itself.

The formal reasons for conducting a creative production research are:

1. My scholarship with the NICTA has a project agreement that defines a list of milestones and deliverables with the project's partner, UC (see 7). The scholarship milestones and deliverables include a report on state-of-the-art text visualisation tools, software, and documentation.

2. My prior experience as an experimental software developer who, for the last decade, has been coding free licensed software following a non-corporative development model. I try to participate only in applied and real world projects.

Besides the main project aim —the creation of innovative software for text visualisation— this project also explores the way in which the relationship between the developer(s) and the client(s) can influence the software end product. It is a clear aim of this project to demonstrate a methodology that supports and enables a productive relationship between software developer(s) and client(s). I present a formal explanation of the proposed methodology in 4.

Data as textual documents

As the amount of data that is publicly available is expanding almost exponentially the field of data visualisation itself is also rapidly developing. Nowadays, seven out of the top ten universities, in the Times Higher Education ranking (TSL Education Ltd. [2012]), have departments or research groups related to data visualisation. Data visualisation has developed in a wide range of departments, such as computer science, statistics, linguistics, graphical design, chemistry physics, genetics, history. Recently data visualisation has emerged as a distinct field in its own right with masters programs and departments dedicated to it (see 1).

institution	rank 2012	Department/Course	URL
Harvard University	1	Broad Institute of Harvard and MIT	http://www.broadinstitute.org/vis
Massachusetts Institute of Technology	2	Broad Institute of Harvard and MIT	http://www.broadinstitute.org/vis
University of Cambridge	3	—	—
Stanford University	4	Stanford Vis Group	http://vis.stanford.edu/
University of California, Berkeley	5	VisualizationLab	http://vis.berkeley.edu/
University of Oxford	6	Visual Informatics Lab at Oxford	http://oxvii.wordpress.com/
Princeton University	7	PrincetonVisLab	http://www.princeton.edu/researchcomputing/vis-lab
University of Tokyo	8	—	—
University of California, Los Angeles	9	IDRE GIS and visualization	https://idre.ucla.edu/visualization
Yale University	10	—	—

Table 1: Top universities and data visualisation departments

I refer to data types as the formats that data can have in order to be processed for later representation. In the visualisation process, data can be transformed several times before it takes on the desired or required format according to the visualisation technique used. Here again there is no a single classification of kinds of data or data-types. We are going to use Shneiderman's classification (1996) called Task by data Type Taxonomy (TTT) that divides data types in seven groups: 1-, 2-, 3-dimensional, Temporal, Multi-dimensional, Tree and Network.

- 1-dimensional data: textual documents, program source code, alphabetical lists, etc.
- 2-dimensional data: geographic maps, floorplants, newspaper layouts, 2-axes diagrams, etc.
- 3-dimensional data: real world objects such as molecules, bodies, buildings, etc.
- Temporal data: all kinds of timelines.
- Multi-dimensional data: most relational and statistical databases are conveniently manipulated as multi-dimensional in which items with n attributes become points in a n-dimensional space.
- Tree data: collections of hierarchical data where each item (except the root) links to a parent item. For example: genealogical data, file systems trees, genetic data,

etc.

- Network data: any collection of items and their relationships, for example: social relationships, computer networks, etc.

This study focuses on visualisation tools applied to textual documents and collections of textual documents. According to Shneiderman's classification, regular texts would be considered 1-dimensional data. A text is a sequential data that goes right-to-left or left-to-right and line by line, top to bottom. However a text can have multiple internal structures, e.g. according to morphology it can have paragraphs, sentences and words. According to the information structure, a text can be ordered by chapters, parts, sections, subsections, etc. If the text has a format like HTML, then it can be ordered by HTML tags, like <body>, , <p>, etc. In those examples the text is transformed to a tree structure as a data type.

Text visualisation is not often typically considered as a subfield of data visualisation. Illinski (2013) asserts that text cannot be considered as a data type. Silić (2010) says that "unstructured text is not suitable for visualisation". In fact, as mentioned above, most text visualisations transform the initial unstructured textual data into a new structured and, usually, a reduced dataset. This new dataset is no longer a 1-dimension data type, but a categorical or a network dataset. And it can be represented with a wide range of tools not specific to natural text representation (Hearst, 2009, Grobelnik and Mladenic, 2002).

The literature review (see next section) shows that text visualisation is valid field and reveals some gaps providing a rich context for the project.

2 Literature review Introduction

As part of this research project, and as part of the deliverables agreement with NICTA, last July (July 2013) I have published the document "How we draw texts: a literature review on text visualisation tools". This document has two parts: a report on the field of text visualisation: origins, definitions, and challenges (first part). And a literature review on text visualisation tools (second part). I'm presenting here an adapted extract of it.

As I often cite the document, I strongly recommend obtaining a copy of the document.

- The document can be accessed directly here: `http://research.nualart.cat/JaumeNualart_2013-How_we_draw_texts.pdf`
- Or in a Github repository: `https://github.com/jaumet/myacademydata`
- A visualisation/exploration tool of the reviewed cases made with AREA (Case #25 in "How we draw texts" document) is accessible at: `http://research.nualart.cat/texvisttools`.

Since I could not find text visualisation literature reviews less than five years old and the field changes so fast, I analysed all the cases I found whilst endeavouring not to

repeat concepts. In case of repetition the older or the original case was taken. This methodology ended up in forty-nine reviewed cases.

The literature review document created will continue being revised during the PhD research. The accessible version is the first version and it will be a key tool in shaping the research project developments.

I am sure that my current literary review has not covered all the existing ideas related to text visualisation. Certainly data and text visualisation is a relatively new field, therefore there is not a single consolidated list of dedicated publications and sources. This presents a challenge in making a consistent review of the field. Some of the cases we are presenting have been found in very specific publications; for example Joel Deshaye and Peter Stoicheff and their works on William Faulkner's visualisations (cases #11, #12 and #13). Reading Stoicheff's notes it can be seen that they developed the tools just to assist in a specific study of William Faulkner's narrative timelines. There are no references to applications of these interesting ideas to other texts, suggesting that more works remain hidden in the depths of other fields.

Gaps and insights

I considered two main aspects when reviewing the cases: the proposed visual language used for each case, and the choice of the dataset. By visual language I mean the concept and features of the representation, its business, its advantages and limitations. Most of the forty-nine cases (74.5%) are visualisation experiments with limited real world scenario application. Another point that I found remarkable is that cases older than five years tend to go offline (42%) or only a static version is online, with no more interaction (50%). The research project aims are to fulfill the gap between experimentation and interest and the implementation and consolidation of text visualisation tools.

Single text visualisation cases have mainly been developed for examining literature. Literature is an unstructured text; apart from complex combinations of words it can have high levels of human abstraction and freedom of structures and experimentation. I contend that it is more effective to apply visualisation techniques used to analyse literature to other kinds that have a more formal register and/or predefined style, such as legal texts, scientific papers, template based texts and communications, etc.

According to my proposed classification of text visualisation tools (see next subsection), I have found only one single/part text visualisation case which is sequential (case #22 "Document Arc Diagrams" by Jeff Clark (2007)). Most part-text-visualisations extract the essence of the text based on some criteria and the original sequence of the text is lost. Since sequential visualisation tools have some advantages, it seems there is room to develop part visualisation tools that maintain the original text sequence.

Text collection visualisations use methods that can be found in data visualisation in general. This idea invites experimentation in bringing more standard data visualisation methods and tools to the specific text visualisation subfield.

Collections of text aggregations is the category that has developed more specific designs and ideas. More work needs to be done in order to find some patterns in this kind of visualisation.

Text visualisation tools: a proposed classification

A remarkable challenge when doing the literature review was to suggest a classification framework of the cases. There is not a clear classification widely accepted or even established as the standard. So finally one of the first outputs of this research project is a proposed classification tree for text visualisation tools.

Since there are few papers that review text visualisation tools, I referenced the classic Shneiderman data visualisation review (1996) as well as new ones (Collins et al. 2009). In these cases the classifications were based on tasks that the visualisation tool can solve rather than on the explicit aspects of the visualisation. This is why we decided to propose our own classification that, while far from perfect, is hopefully a useful approach to a classification based on visual features.

Single texts

- Whole <-> Part
- Sequential <-> Non sequential
- Discourse structure <-> Syntactic structure
- Search
- Time

Text collections

- Items <-> Aggregations
- Landscape
- Search
- Time

Table 2: Proposed Text visualisation classification

The ground level of classification of text visualisation tools according to the type of data has two categories:

- Single texts: a single text is a sequence of words ordered according to the hierarchy: document > paragraphs > sentences > other punctuation marks (“:”. “;”, “?”, “!”. “,”) > words > syllable and phonemes. In cases where the text is a book or other

kind of structure, then, it is possible to have more granularities including: chapters > sections > subsections > ... Also we assume the metadata of the text and other attached texts: title, authors, publishers, copyright notes, acknowledgement, dedication, preface, table of contents, forward, glossary, bibliography, index, etc.

- Text collections: groups of texts in which each item is a clearly differentiable entity. Usually when talking about collections of texts, we talk about texts that have some similarity, either in register, length, or structure. All the cases we have reviewed are collections of the same kind of texts. Heterogeneous collections of texts are also referenced in the literature (Meeks, E. 2011), especially for a representative analysis of a field of knowledge, in which cases the goal of the collection is to include the maximum variety of expressions and a wide vocabulary. In these cases the dataset is heterogeneous according to its structure and register.

Starting from these two differentiated kinds of visualisations, I have added several subjective subdivisions to each case (see "How we draw texts" document). The aims of this qualitative classification is to help describing and explaining the reviewed cases, as well as to suggest key features of text visualisation tools.

For further information the presentation and definition of the classification is in section 2.1 (p. 18) of the document "How we draw texts: a literature review on text visualization tools".

3 Research questions

I present a main research question and two specific ones.

Main question

How can we create text visualisation tools that are used in real scenarios?

To answer this question I will design, develop and implement software applied to a number of real case scenarios, mainly in web environments. The development style will be an artisanal software development. I use this term inspired by Mark Bernstein (2004, Atzenbeck [2008]) and Christopher Schanck (Schanck 2009).

Artisanal in the sense of handmade, taking care as if the project were a personal project, feeling part of the project. Living and assuming the project as a personal challenge.

Artisanal software in the sense of open source, open techniques, handmade coding, and the originality of the work.

Artisanal software development in the sense of a hyper customised project, A piece made step by step in a non necessarily perfectly optimized working process. An organic, and spontaneous-like project based in a non-corporate style of working and influenced by creative labs, hackerspaces and free software communities.

Another choice is to work only in real world scenarios, makes the project realistic in resources and in timing. Obviously, limited time will affect the degree of project experimentation. My position is to see this as an advantage. If this project cannot create too experimental tools it is only because one of its goals is to reach implementation in-production systems.

The literature review presents good ideas on text visualisation, most of them only experimental, with no real applications yet. Here the challenge is to generate new implementations more than new concepts. Despite that, of course, every project has its creative and experimental parts.

Since I will have freedom deciding which projects to develop, the ones in which the development of the tool will be a new feature will have more chances of been chosen. This is because a new tool creates excitement and its implementation will be likely seen for users.

4 Research methodology, methods and process

I present a description of the general methodology and, afterwards, the list of most important methods proposed linked to their disciplines. Finally I present a diagram of the life-cycle development model and a comment on the status of the research project ethics application.

General Methodology

This is a multidisciplinary practice led research through practice (practice as an element of the research design). This concept is an adaptation of the classification that Frayling defines in visual arts (Frayling 1993).

The methodology is based on the study and lessons learnt from a number of text visualisation tools that will be developed as part of the project artifacts. Following Linda Candy (Candy et al. 2006), since the main focus of this research is to advance knowledge about practice and it seeks to learn about practice through making the artefacts it can be considered a practice led research rather than a practice-based one, where the contribution to knowledge is demonstrated through artefacts such as images, music, designs, models, software. For this reason the project cycle reword for each artifact's development is critical.

The developments involve not only computer science knowledge, but team and project management, and collaborative design. Researchers and professionals from a wide range of disciplines will work together, forming temporary teams of development.

Several successful cases of data visualisation developments and software development in general have influenced the chosen methodology for this research project. Through my endeavour to work with inspiring people, I am inspired, among others, by: Moritz Stefaner: artist & developer that works as a freelance from Germany (Stefaner, M.), Stamen studio: maps specialists from L.A. (sta), And in general all kinds of hackerspaces and community labs.

These examples share several characteristics:

- Size is important:you can work effectively in small or very small teams.
- Dialogue design: the project design is the result of the communication between the developers team and the so-called customer. Communication is critical in collaborative project.
- Multidisciplinarity: the participants in each project come from a number of backgrounds and/or skills. There is always presence of people from science and humanities.
- Freelance culture: they look more like a creative studio, arty, maybe hipster than a computer systems corporation.

Two more examples have inspired this project. First: the way that education is evolving in Finland. How important is the level of challenge in the motivation of students or developers, and how the free knowledge way works better that the proprietary one (Triki). Second: William J Turkel and his great book "Programming Historian" (Turkel and MacEachern 2007). Turkel is a digital humanities developer that shows cleverly how computing can help humanities -historians- on managing and analysing data. This is a tangible example of the unavoidable multidisciplinarity of knowledge.

The analysis and the comparison of each developed case will be the base to propose a concrete development methodology that can contribute to increase the use of text visualisation tools in real case scenarios.

Specific Methods

Methods used according to the disciplines related to each phase of the development project are:

1. Software engineering
    a) Free software development (GNU Manifesto, Stallman et al. 1985): this garantees the projects to be truly open and knowledge shareble ready in an standard way.
    b) Open to collaboration: version control software makes easy to include collaborators in coding projects, among other advantages, like working in multiple places, multi editing files, side-tools for project management and documentation,etc.
    c) Third party libraries: when appropriate, the incorporation of free licensed third party libraries and external modules will speed up the project and increase the quality and robustness of the code.
2. Humanities & creative arts: as a student of the Faculty of Arts & Design.
    a) Reflective practice: the project is based on continuous learning through every development. My previous experience in developing software will be taken as an advantage, as well as the experience of collaborators. I will conduct

a reflective practice management in order to improve the project during its development and my own skills as well (Schön 1983, Brookfield 1998)

b) Graeme Sullivan's braid metaphor methodology (2005): since this project is based in spontaneity and organic organization it follows the Sullivan's braid metaphor in which the project evolution is based on questioning, exploring, analysing and discussing ideas.

3. Usability testing: due to the interface design and the user evaluation of the tools. The method selected will depend on the specificities of each development: time and tool availability.

   a) User evaluation during development: think-aloud, interviews and/or questionnaires.

   b) User evaluation at the end: automatise user actions record and/or questionnaires.

4. Machine Learning & Natural Language Processing: usually visualization projects requires data transformations. For state-of-the-art data transformations I will work with collaborators, mainly researchers at Machine Learning Research Group (MLRG) at NICTA. Two scheduled methods are:

   a) Text meaning extraction & text summarise: with Dr. Gabriela Ferraro (NICTA).

   b) Text topic modeling: with Dr Wray Buntine (NICTA), co-supervisor of my PhD.

Case life-cycle

The development life-cycle I propose is represented in Figure 1. I propose a nine steps timeline process. 40% of the steps involve people (collaborators, clients, test users). The rest involves only coders and computers. The proposed life-cycle follows the standard waterfall model. It is a sequential design process, in which the progress is a linear list of steps that includes Conception, Initiation, Analysis, Design, Construction, Testing, Production/Implementation, and Maintenance. Waterfall model criticism is about the inflexibility and difficulty integrating unforeseen tasks and it advocates for more modern models, like Agile, Spiral and Iterative and incremental development model. Waterfall works well for smaller projects where requirements are very well understood (ISTQB [2013], Amlani [2012]).

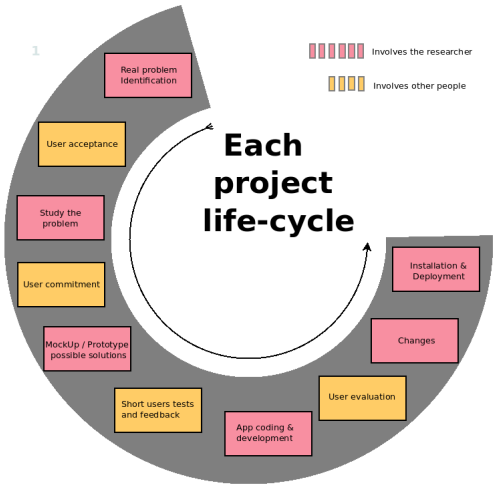

Figure 1: process cycle for the development of the software artifacts of this research project.

Research project ethics

I will apply to the ethics committee because the evaluation tests involve users. Since I'm not improving an existing visual method, but introducing a tool where none was used, it will not be necessary to measure the usability of the tool, but to validate that the initial requirements for each development have been accomplished.

Ethics application is being prepared at the time of the publication of this text. NICTA committee has assessed the ethics risks of the research proposal as negligible risk. Next step will be to apply to the UC Ethics committee.

5 Timeline

- 2013
  - 1st semester
    - Jan-Jun: attended HDR-UC Seminars (writing, research methodologies,etc).
    - Mar 20th-23th: Presentation "Data visualisation tools" at Machine Learning Research Group yearly meeting (Kyoloa NSW).
    - Jul: Paper published in peer-review open access journal Information Research (Nualart and Perez-Montoro, 2013).
    - May 16th: UC-HDR lecturer in the seminar: Introduction to LyX

    * Mar-Jul: How we draw texts: a literature review on text visualization tools
  - 2nd semester
    * May-Oct: PhD software development (I), Visference, sofware development at NICTA as part of the practice based PhD: Visference.
    * Sep: Annual Progress Report
    * Aug-Oct: Research proposal.
    * 1st Oct: Confirmation seminar.
    * Nov: release of Visference. Reserach proposal.
    * Jul-Des: PhD software development (II),
    * Oct-Dec: AREA applied to the open access journal Information Research. Outputs: software to explore the papers of the journal.
    * Aug-Des: PhD software development (III), UClaims, software development with NICTA and Lens.org about one-patent visualisation + paper/conference.
    * Submit ethics application
- 2014
  - 3rd semester
    * Aug-Feb: PhD software development (IV), Treebars for patents exploration. Outputs: software development with NICTA/Lens + paper/conference.
    * Write/compile notes of the I, II and III developments for the exegesis.
    * User evaluation of AREA software (II).
    * Jan-April: submit/review AREA paper.
    * User evaluation for the developments III, IV.
    * Write & submit Patents paper
    * Submit to conferences (not decided yet)
  - 4th semester
    * Annual Progress Report.
    * Doctoral Working Progress Seminar.
    * Write/compile notes of the IV developments for the exegesis.
    * User evaluation for the development I.
    * PhD software development (V) (not decided yet).
    * Conferences presentations (not decided yet).
- 2015

- 5th semester
  - User evaluation for the development V.
  - First exegesis draft.
  - Conferences presentations (not decided yet).
- 6th semester
  - Annual Progress Report.
  - PhD administrative work.
  - Final submission of exegesis.
  - Verify that developed software are online and accessible.
  - Final Seminar.

6 Bibliography

References

Stamen.com. URL `http://stamen.com/`. Accessed: 2013-09-26. (Archived by WebCite at http://www.webcitation.org/6Jv9xzRP8).

Radhika D Amlani. Advantages and limitations of different sdlc models. *IJCAIT*, 1(3): 6–11, 2012.

Claus Atzenbeck. Interview with mark bernstein. *SIGWEB Newsl.*, 2008(Summer): 4:1–4:5, June 2008. ISSN 1931-1745. doi: 10.1145/1377501.1377505. URL `http://doi.acm.org/10.1145/1377501.1377505`.

Mark Bernstein. Neovictorian computing, 2004. URL `http://www.markbernstein.org/NeoVictorian.html`. Accessed: 2013-09-26. (Archived by WebCite at http://www.webcitation.org/6Jv77F9QB).

Stephen Brookfield. Critically reflective practice. *Journal of Continuing Education in the Health Professions*, 18(4):197–205, 1998.

L. Candy, S. Amitani, and Z. Bilda. Practice-led strategies for interactive art research. *CoDesign*, 2(4):209–223, December 2006. ISSN 1571-0882. doi: 10.1080/15710880601007994. URL `http://www.tandfonline.com/doi/abs/10.1080/15710880601007994`.

Christopher Collins, Sheelagh Carpendale, and Gerald Penn. Docuburst: Visualizing document content using language structure. In *Computer Graphics Forum*, volume 28, pages 1039–1046. Wiley Online Library, 2009.

Christopher Frayling. *Research in art and design*, volume 1. Royal College of Art London, 4 1993.

M. Grobelnik and D. Mladenic. Efficient visualization of large text corpora. In *Proceedings of the seventh seminar. Dubrovnik, Croatia*, 2002. URL http://ailab.ijs.si/dunja/SiKDD2002/papers/GrobelnikSep02.pdf.

Marti a Hearst. Search user interfaces. *Search User Interfaces*, 54(Ch 1):404, November 2009. ISSN 00010782. doi: 10.1145/2018396.2018414. URL http://searchuserinterfaces.com/book/.

ISTQB. What is waterfall model- advantages, disadvantages and when to use it?, 2013. URL http://istqbexamcertification.com/what-is-waterfall-model-advantages-disadvantages-and-when-to-use-it/. Accessed: 2013-09-26. (Archived by WebCiteÂ® at http://www.webcitation.org/6Jw3P8qfK).

Meeks, E. Documents | digital humanities specialist. https://dhs.stanford.edu/comprehending-the-digital-humanities/documents/, 2011. URL https://dhs.stanford.edu/comprehending-the-digital-humanities/documents/.

Noah Iliinsky. *Choosing visual properties for successful visualizations*. s IBM Software - Business Analytics, 2013. URL http://public.dhe.ibm.com/common/ssi/ecm/en/ytw03323usen/YTW03323USEN.PDF.

J. Nualart and M Perez-Montoro. Texty, a visualization tool to aid selection of texts from search outputs. *Information Research*, 18(2), jun . ISSN 1368-1613.

Christopher Schanck. Don t be a coder, engineer, or developer: be a software artisan, 2009. URL http://designbygravity.wordpress.com/2009/10/03/dont-be-a-coder-engineer-or-developer-be-a-software-artisan/. Accessed: 2013-09-26. (Archived by WebCite at http://www.webcitation.org/6Jv5UeHco).

Donald A Schön. *The reflective practitioner: How professionals think in action*, volume 5126. Basic books, 1983.

Ben Shneiderman. The eyes have it: A task by data type taxonomy for information visualizations. In *Visual Languages, 1996. Proceedings., IEEE Symposium on*, pages 336–343, 1996.

Artur Silic and Bojana Dalbelo Basic. Visualization of text streams: A survey. In Rossitza Setchi, Ivan Jordanov, RobertJ. Howlett, and LakhmiC. Jain, editors, *Knowledge-Based and Intelligent Information and Engineering Systems*, volume 6277 of *Lecture Notes in Computer Science*, pages 31–43. Springer Berlin Heidelberg, 2010. ISBN 978-3-642-15389-1. doi: 10.1007/978-3-642-15390-7_4. URL http://dx.doi.org/10.1007/978-3-642-15390-7_4.

Richard Stallman et al. The gnu manifesto, 1985.

Stefaner, M. Moritz stefaner, truth and beauty operator. URL http://stefaner.eu/. Accessed: 2013-09-26. (Archived by WebCite at http://www.webcitation.org/6Jv9PUq6u).

G. Sullivan. *Art Practice as Research: Inquiry in the visual arts*. SAGE Publications, 2005.

Salah Triki. Graduate students in finland solve real problems beyond the classroom. URL https://opensource.com/education/12/10/graduate-students-finland-solve-real-problems. Accessed: 2013-09-26. (Archived by WebCite at http://www.webcitation.org/6JvAq8xXX).

TSL Education Ltd. World university rankings 2012-2013 - times higher education. http://www.timeshighereducation.co.uk/world-university-rankings/2012-13/world-ranking, 2012. URL http://www.timeshighereducation.co.uk/world-university-rankings/2012-13/world-ranking.

William J Turkel and Alan MacEachern. The programming historian. 2007.

7 Appendix

7.1 NICTA scholarship milestones

SCHEDULE 4 TO PROJECT AGREEMENT: MILESTONES AND DELIVERABLES

NICTA Milestones / Deliverables

No.	Deliverable / Milestones	Delivery Date
1.	nil	

Project Partner Milestones / Deliverables

UC will undertake the work and deliver the results through a series of reports and related software releases which will include source code and documentation.

The software is to be developed to a professional standard and is to include documentation to a level sufficient to allow a third party to be able to modify the software, as well as appropriate revision control and coding guidelines.

The parties will hold formal liaison meetings at the commencement of the project and from time to time thereafter. More precise standards for the software will be agreed in writing by the parties at the first formal liaison meeting. Detailed work plans for the subsequent 6 months will also be agreed in writing at each formal liaison meeting

No.	Milestones	Deliverable	Delivery
1.	Report and evaluation of existing state of the art on large corpus text visualisation	Report	1/9/2012
2.	Report on browser-based visualisation frameworks, feasibility of text visualisation in browser	Report	1/3/2013
3.	Report on initial text visualisation prototypes - rich overview	Report & Software	1/9/2013
4.	Report on text visualisation prototypes - visualisation of topic / document space	Report & Software	1/3/2014
5.	Report on technical implementation / integration of visualisation techniques	Report & Software	1/9/2014
6.	Final report - research outcomes and implications	Report & Software	1/3/2015

Figure 2: NICTA milestones and deliverables