

PhD Progress Report 3

Lorand Kedves
2017/2018. II.

Idea Projection

Lorand KEDVES

Doctoral School of Information Science

University of Pannonia

Supervisor: Dr. Botond BERTÓK

Summary

This PhD research emerged from 20+ years of experience in software development and maintenance, requirement analysis and system architecture design: how to extract and emphasize the structure and configuration of any software, favoring an abstract, visual form against thousands of lines of code, and how far this approach can reach. For scientific closure: is it possible to achieve total coverage, when all necessary source code, configuration or binary can be generated directly from the design?

During the past semester, the significance of the previous business-oriented projects has decreased, and I have found more research areas. I temporarily joined the Doug'50 group formed to create a new demo for the 50th anniversary of Douglas Engelbart's Mother of All Demos. I am starting a cooperation with lawyers and publishers focusing on more flexible and networked management of documents using publicly available Hungarian laws as data source. I restarted Dust Platform implementation and found really interesting structural elements while focusing on bootstrapping (initializing the self-contained kernel).

I have chosen 3 courses, if all succeed I will have one left for the 4th semester besides research and publication efforts. I have sent my previous, "Science of Being Wrong" article to Knowledge and Information Systems International Journal (Springer, impact factor 1.7), plan to present the document management research on conferences: Vocal 2018 (as practical graph representation of information and processing) and Jurix 2018 (as an IT application in legal environment) in December, 2018.

Activity and Experience

Follow-up on the Previous Report

Efficient Coding – WYSIWYG Editor

The editor is used by the Government now, with ongoing bugfix/improvement activities. The framework performs well: it is reliable regardless of the inherent complexity, no issues appeared on that level; there were several improvement requests including more complex undo integration of external modules, all requests were managed by at most minimal extensions of existing services.

I currently integrate a formatted HTML editor that also has template service (so the HTML can "look into" the data layer and read data to display), and a general binary storage that support both attached images, HTML templates and created exports and also open for any further extensions. Although this is related to the scope of this PhD research (separating different roles and layers of knowledge), I mostly rely on integrating external modules, so I stop reporting about that progress.

Knowledge Card Management

At the pharmaceutical company, regardless of cancelling the implementation of knowledge card management, we have done a successful internal survey and vendor interview sessions, collected and analyzed the data and made knowledge-driven decisions. However, the scope of the project grew beyond the actual group, therefore the project was cancelled.

Research Activities

Doug'50 Group

As I have learned about the Mother of All Demos and that it was presented in 1968, I thought it would be a good mid-term goal to build an application that resembles some forgotten points of the original. I have found that a group mixed from great internet pioneers, colleagues of Douglas Engelbart and other experienced programmers already exists with the same goal. I have contacted the organizer and after a short discussion, I joined the group.

Among other tasks, we were asked to collect what we think would be important to bring back to light from the original demo, and how we can do it in the current environment and the given time frame. I have created a roadmap (<https://doug-50.info/lorand.html>), focusing on the ultimate goal: a flexible and networked knowledge management environment that also contains and executes itself. I already have partial implementation of this platform, used and presented among others at the TDK contest in 2016.

However, the group focuses on heavily using existing technologies to ensure a working presentation by the event in December, 2018. This includes federation of ontology management tools, HTML communication layer, JavaScript libraries, JSON-LD for data storage etc. In contrast, my research focuses on the essential services and architecture required to store and communicate knowledge and this layer is actually covered and hidden by the existing standards and implementations. I concluded that both my initial concepts and aims were too far from the group and did not allow efficient cooperation.

Patterns

Following the analogies between civil and software engineering that led me to Buckminster Fuller and the very important Tensegrity model, I have found Christopher Alexander. He listed reusable and meaningful patterns that reflect the needs of communities and individuals, and later realized that these patterns, especially their visual representation are really time-proof over any changes of actual technologies and materials. This idea blossomed in software creation, like the GoF Design Patterns that allows defining fundamental system services regardless of the actual tasks and implementation languages (and therefore, are level C tools of Intellectual Augmentation Model from Douglas Engelbart).

This approach continued in the Pattern Language of Programs (PLoP) research community. I contacted the organizer of their upcoming conference (Viking PLoP) and learned that they have an interesting method: the papers should talk for themselves in a 1.5-hour session of pattern experts where the authors appear but not talk. This needs serious preparation called shepherding with a PLoP consultant. However, I have found that their method is based on written patterns while I looked for a more flexible, network or graph-oriented approach, therefore is too far from my current goals.

Change in Research Focus

We had very interesting discussions with Gyuri Lajos, a Hungarian member of the Doug'50 group. He both recommended important results, like further articles from Alan Kay to process, and warned that knowledge representation is a thoroughly defined and researched area. The questions that I work with are out of that scope, so the current research title is misleading. I am also grateful to local faculty members, Botond Bertók, Gyula Simon and István Heckl, who helped a lot with critical questions and requesting further clarification of my aims and the relationship with other areas and ongoing research.

I see that the current activities, from academic Computation Theory to practical software design and quality assurance methods focus on using external tools and notation systems to formalize the “body of knowledge” that they later operate on. My problem is that during the “projection phase”, when we adopt or create our languages and select tools and standards to describe the current state, we lose all information that we can’t represent at that moment: questions, problems, seemingly irrelevant factors. We also lock our vision by the current structures and language, changing our meta-knowledge comes with a high price of instability, data loss, repeated modeling and implementation.

I focus on a network of flexible and minimal domain specific languages (DSL), that we create to formalize our tasks. Formalizing languages, describing data structures or algorithms are also tasks that require languages, so these basic DSLs should be part of the network and should be managed and improved by the same tools, while the managing tools should be defined by them. This approach seems to follow the initial aims that Douglas Engelbart set in his Augmenting Human Intellect report, and the result should be close to Memex as described by Vannevar Bush in the As We May Think article.

Consequently, I have changed the work title of my research to “Idea Projection”: concepts and tools required to efficient, flexible and reliable formalization of ideas both for individual and community use. Another way to see it is that Informatics and programming are different things with opposite motivations, exactly like physics, metallurgy etc. (science) versus engineering (solving problems); or automata (requiring understanding the task) versus the Turing Machine (focusing on the process to execute).

Document Storage Architecture

Another interesting and related research area is text management. After separating from the Memex idea of Vannevar Bush, transforming the existing body of scientific knowledge became the center of research, from JCR Licklider’s Libraries of the Future to Ted Nelson’s Xanadu vision. These approaches focus on storing texts in a cloud-like global medium with bidirectional references, so one can browse the whole global knowledge base following their connections. We attempt to create such a system as this task is a more defined subset of the research target, using Hungarian laws as sources, because they have all the requirements, like complex hierarchical structures, well defined internal and external references, and changes over time.

I have analyzed the existing sources and found that public web Jogtár by Wolters Kluwer has a promising data structure in HTML. A short “proof of concept” Java application using JSoup HTML DOM module and regular expressions is already running that separated headings, stores the texts in a minimal memory “cloud”. The data model is under development, with one thing in focus: the atomic element of the text is not the character, but the referable statement, to which both hierarchy element (with the finest granularity) and text references point to. It is likely that we should do some data cleaning here, so the final visible text will not be the same as the original because of the automatic “reference to text” functions, but semantically equivalent. I also plan to do some experiment with user interface functions allowed by this transparent representation.

Dust2018 Results

I restarted Dust development (<https://github.com/MondoAurora/Dust2018>). The greatest challenge is that this time I focus on the final aim: create a self-containing framework that includes itself, so any Java code is temporal and later should be generated from the graph representation of the required algorithms. This required very serious refactors from the initial Enum-based attribute and link definitions to one that will be equivalent with a dynamic, runtime-modifiable solution. Another interesting finding was the “avatar” component that is required between any active component and the shared knowledge context. This ensures that all participants use the same definition of terms when interacting with the common knowledge, and already required by the persistence components, like the JSON file reader. I also build a minimal (Quick and Dirty) Dust for the text manager because that is necessary for the flexible data storage and need to run before the “real” Dust.

Academic Results

Publication

“The Science of Being Wrong - Unsung Heroes of Informatics” article was rejected from *Information and Software Technology* and later from *Technological Forecasting & Social Change* journals, now submitted to *Knowledge and Information Systems International Journal* (ISSN: 0219-1377 / 0219-3116, impact factor 1.7 in 2016). I concluded that my topic is too broad for a “young” researcher without previous references, this is why we shift our focus to more defined environments.

I prepare to submit my document storage and management architecture to *8th VOCAL Optimization Conference: Advanced Algorithms*, December 10-12, 2018, Esztergom. The paper will focus on the graph representation of the hierarchical text with complex internal and external connections, and that the same tools can be used to store the processing software itself. This approach allows using various analysis and presentation tools, and as a novel result: the same to the data and algorithms processing it.

The same project but focusing on the data and allowed management features will be submitted to *JURIX 2018: The 31st international conference on Legal Knowledge and Information Systems*, December 12–14, 2018 in Groningen, The Netherlands. Here we will present how actual Hungarian law text can be processed, focusing on finer hierarchy levels, two-way references, browsing over the links, “delta-documents” (laws that change other laws), timeline and possible editing features.

Courses

Computation Theory – Dr. István Heckl

This is a fundamental source for my research, not only because I plan to create a more precise representation of software (somewhat similar to the von Neumann architecture), but because the more abstract level related to modeling communication and understanding is nicely shown in the difference between the automata (that requires and represents the understanding of the problem and therefore limited) and the Turing Machine (which does not require the understanding of the system but the process, more capable but does not guarantee a response).

Global Optimization – Prof. Tibor Csendes

In the seminar, we learn the theoretical background, advantages and limitations and practical applications of local and global optimization strategies, including interval arithmetics, formulizing conditions, finding optimums of functions in practice both mathematically and applications (Excel Solver and MATLAB with optimization packages) in lab exercises.

Linear and Nonlinear Programming – Prof. István Maros

Overview of the linear optimization algorithms, their limitations and applications. The material is essential for the mid-term examination.

Progress Summary

At current date, pending in brackets

Course credits: 48 (+8),

Progress reports: 10 (+5),

Publications: 0 (?50: 1 submitted article, ?40: 2 conferences planned).

Veszprém, 2018.06.02

Lorand KEDVES
PhD Student

Dr. Botond BERTÓK
Supervisor