



EXTENDED EDITORIAL

Simplicity in IT: a Chance for a New Kind of Design and Process Science

Tiziana Margaria^{a*} and Barry D. Floyd^a

^a Chair Service and Software Engineering, University of Potsdam, Potsdam, Germany

^b Orfalea College of Business, California Polytechnic State University, San Luis Obispo, CA USA

Abstract The philosophy of *simplicity as a key value*, underlying and driving design and decisions, is strategically important yet poorly understood and rarely systematically applied in IT. Instead, design principles attempt to focus predominantly on increased functionality within thinly disguised complexity, often at the expense of life cycle costs and total cost of ownership issues (e.g., training, system malfunctions, system upgrades). Designers and researchers alike must be challenged in their current thinking in order to make salient these trade-offs and impacts. This special issue presents five points of view of how simplicity focused work can provide intrinsic and long lasting gains in improving the philosophy, methods, and tools behind the practice of design and processes in IT systems. With the increased use of IT in socially critical areas as healthcare, transportation, banking, and international telecommunication infrastructure, society can no longer afford systems that do not perform as specified, causing costs and grief, and are designed in such a fashion that system modifications turn out to be administratively and economically infeasible. We examine here aspects of simplicity in the domains of design, deployment and provision, and evolution along the IT lifecycle.

Keywords: Simplicity, total cost of ownership, simple design process

1. Simplicity as a Value

Despite all the promises, working with IT and especially with software products rarely proceeds as expected. Business organizations find that software projects are rarely on budget or on time and rarely deliver the promised value. Research has shown that less than 25% of software development projects meet projected goals. Difficulties and frustrations are also very salient when inevitable upgrades, migrations, or version changes are involved. We seem to be unable to build systems that are easily adaptable to changes in our needs.

In addition to the day to day difficulties, the IT landscape is sprinkled with high visibility failures. The difficulties of installing an ERP application at Hershey Chocolate, Ford's abandonment of a purchasing system, the sinking of the US Navy's AEGIS cruiser Valley Forge due to the inability to integrate new software, Microsoft's difficulties with the Surface RT with \$900 million of inventory written off, and most recently the difficulties experienced with the launch of the medical insurance portal in the US are

* Corresponding author. Email: margaria@cs.uni-potsdam.de

appalling examples of inefficiency and waste – at odds with the image we have of our design abilities and technological prowess.

These negative outcomes with IT lead to an enormous loss of personal, organizational, and societal performance and value. The inability of understanding the (mostly untold) technical issues leading to failure, malfunction, or delay, diminish the trust citizens have in IT and in the socio-political system that supports projects that are perceived as resulting in failure. While there are many causes for the inability of the IT world to deliver, a key factor is that *overly complex* systems are designed.

Complex systems in many cases mirror the complexity of the environment in which they are embedded. The world is per se complex and as such there is inherent complexity in the way in which these systems provide the needed functionality (Hinchey & Coyle, 2012). However, as consumers, we often add to the problem by demanding complexity or complication. We seek out products that have high levels of functionality and features when so very often we use so very few of them. 20% seems to be the usual number of features we use on any given product). User models derived from log traces are well known to be insufficient to represent a system's or component's functionality. This is one of the central limitations of passive model extraction that is usually the basis for monitoring, testing, as well as regression testing.

In spite of these difficulties with handling complexity, we also find stunning successes where companies have identified ways to design around inherent complexities and deliver high performing products and services. The designers in these companies follow the vision of leaders who embrace the concept of simplicity, removing the unnecessary and providing elegance. Companies such as Apple consistently turned out products that became standards in their industry. There are also other examples that do not involve devices, like Skype as a simple IP-based alternative to roaming-based telephony, or PayPal as the first ubiquitously functioning low-cost instantaneous online payment system.

In their book, "Simplicity Wins: How Germany's Mid-Sized Industrial Companies Succeed" (Rommel *et al.*, 1995), authors Rommel, Kluge, Kempis, Diederichs and Bruck discuss their understandings of why in the late '80s certain German medium sized manufacturing companies outperformed others in the marketplace. They place most of the success on the notion of simplicity. For them, simplicity is measured in how manufacturing companies organize their competitive strategy and, how they consequently align key managerial decisions to coincide with this strategy. Simplicity leads to lean, which is the new notion of simplicity in the production-oriented economy. In the successful companies, managerial decisions were made on issues such as product offering, number and variety of customers targeted and served, the organizational system involved, and the types of business systems. For example, the more successful companies focused their marketing efforts on a smaller subset of suppliers and customers and worked to optimize their interaction with those customers. Internally, the more successful organizations focused efforts on simple goals that were in direct alignment with understanding how each business operation provided value to their customers. The striving for simplicity, coherence, cohesion, alignment pervaded the entire organization, becoming a holistic cultural trait.

While not always applied, there are widespread understandings of simplicity based design. The field of human-computer interaction has flourished over the past 20 years. Researchers in HCI successfully provide insight and guidance on usability, intuitive associations of symbols to functions, and coherence within a user interface. There are guidelines which have been published such as the SAP Simplicity Guidelines¹

In "Living with Complexity" (Norman, 2010), the noted HCI researcher Don Norman discusses how good design can be created given a simplicity perspective, though he suggests that simplicity in and of itself is not the answer but how simplicity is approached in the landscape of design and use. The foundational book by John Maeda provides 10 laws of simplicity based design (Maeda, 2006). Importantly, some designers find a way to make even the complex appealing and accessible. They have a

¹ See for example <http://www.sapdesignguild.org/goodies/simplification/ConPrinciples.htm>

vision that leads their design along a different path. Hence we have successful products that create new trends and re-base the state-of-the-art of the technology as well as the look and feel.

Given the knowledge base and the many success stories, why do we continue to build overly complicated systems? It seems to us that simplicity should not be an ‘add-on’ in terms of thinking of design. As it cannot be retrofitted upon existing artefacts and methods, it needs to be more deeply embedded in the practice, in the individual, in the culture, at much earlier stages.

2. Research on Simplicity

This special issue is devoted to the fundamental concept of *simplicity in design and process science*. Surprisingly, it is a rarely addressed topic. Simplicity in IT has been particularly addressed through the work carried out in the context of the ITSy Support Action² (Margarita & Hinchey, 2013; Merten & Steffen, 2013), that aimed at establishing whether Simplicity in IT is a good candidate Future and Emerging Technology (FET), that is one of the far reaching up and coming technologies or movements that the European Commission funds as long term high impact research topics. In the course of this action we have discovered that both concepts, complexity and simplicity, while related, should not be conceived of as two sides of the same coin nor is there a win-lose game involved. We have gradually discovered that these concepts build upon integrative issues, where the able designer resorts to complexity / simplicity as higher-level concepts and uses them as a means to add concrete specific value for the various constituencies involved in system design, development, and use while at the same time reducing costs. There are numerous ways of adding value through simplification and streamlining, and numerous potential recipients, like the end users, the developers, the testers, the validators or certifiers. Together, they define a multidimensional space of collaborative design decisions that need to be taken knowingly and in concert by some or all of these constituencies. Added value is here the foremost design objective. Some added value results in ease of use by end users, other added value manifests itself through more robust systems for system developers. Other simplicity based decisions in design will reduce the risk of uptake of various aspects of the design process including choice of methodologies and techniques, or in the selection of third party products by partners. Through this, simplicity techniques and strategies, once accepted as core values of the IT ecosystem, may play a role in the many stages of IT design.

In our research into simplicity, we collected war-stories from IT developers and IT researchers. We asked about their experiences and perspectives on how simplicity fits within the fabric of managing complexity (Floyd & Bosselmann, 2013). We have also started to create a simple ontology of the complexity management frameworks and techniques which inform design in specific areas of IT, as for instance in resource constrained and embedded systems design (Margarita *et al.*, 2012), in formal methods (Margarita, 2012), and in ERP systems (Margarita & Steffen, 2012). Two concurrent themes run through our work on simplicity and relate directly with the culture and encompassing aims of the design and process science community:

- A key focus of the design and process science community is on *transformative research and education through interdisciplinary means*. Our conception of this research perspective originates within the research community but extends into IT practice and, indeed, into education. Although many believe simplicity to be of *intrinsic value* in both IT research and education, an interesting finding so far is that there are no communities of practice within IT that adopt this concept as an explicit cornerstone to their work. Additionally, the transdisciplinary character of simplicity is clearly affirmed by the need to align the values of all the interdisciplinary stages that an idea has to go through to impact practice as a product: the idea, the design, the realization, the organization around the production of the product, the marketing, the communication strategy, the adoption, the added value that this product brings in its situation of use, and the training, maintenance, evolution. Like in a clean room

² IT/[t] Simply Works! ITSy Support Action – European Commission FP7 ICT FET contract 258058.

environment, as soon as one of these stages disrupts the basic property of simplicity, the damage is done and the cost of recovery or repair enormous.

The transdisciplinary character of simplicity is well aligned with the description of transdisciplinarity and interdisciplinarity of (Carbone & Crowder, 2011), and we see that the enforcement of simplicity as an intrinsic value is tightly connected with the resolution of conflicts, as described in (Yan & Zeng, 2011) and with the management of tradeoffs.

- The second and technical theme, specifically highlighted in this special issue, is the identification of *frameworks and techniques for simplicity enabled design and development*. A number of framework have addressed design as a discipline of responsible and motivated choices and tradeoffs, as described for TRIZ in (Cascini, 2012) and for the relation of design activities with creativity and stress in (Nguyen & Zeng, 2012). They did not however target specifically simplicity. Simplicity has been prevalent in the HCI community for some time, and, in fact, it was identified very early as a critical design goal. However, it has not found a home in other critical areas of IT research and practice. We believe that the philosophy of simplicity is strategically important towards ensuring global leadership in technology. Designers and researchers alike must be challenged in their current thinking in order to make salient these trades-offs and impacts. To this aim, we have also collected the contributions and positions of a selected group of researchers and practitioners in a special issue of IEEE Computer entitled “Simplicity in IT” (Margarita & Hinchey, 2013). There, specifically the introduction states the philosophical viewpoint (Margarita & Hinchey, 2013) that less is more.

With the increased use of IT in such socially critical areas such as healthcare, administration, transportation, communication, society can no longer afford systems that do not perform as specified and are designed in such a fashion that change is administratively and economically infeasible. We believe that an understanding of simplicity and its benefits and how the concept of simplicity is engaged is key. Simplicity is foundational, its essence fundamental to many desired characteristics of IT systems such as reliability, flexibility, usability and trust. Thus, several communities need to reshape key research disciplines with the goal of influencing IT practice. The central action field is the big and diverse area of process design and maintenance: it impacts and shapes all the value chain of IT.

3. Simplifying Process Design and Maintenance of IT Systems

In the overall goal of achieving simplicity, the adequate handling of processes in and around the products is of paramount importance. In this special issue, we see five examples of how systematic thinking involving simplicity as a core concept can lead to better designs.

3.1. Handling simplicity at design time

How to represent the processes easily at design time (what the system under development should do) and how to handle this design phase in a simple, intuitive and adequate way is dealt with in the first contribution: *Simplicity Driven Application Development* by Maik Merten and Bernhard Steffen (Merten & Steffen, 2013). They illustrate a specific case study on how to support agility in process design and evolution within a context of the co-creation of services and service orchestrations. The perspective is that of design tool designers: how should tools look like to be intuitively understandable by the users, which instances of simplicity and simplification play a key role at designing tools for large-scale adoption by non-IT experts? The central value here is the ease of change of the services and of the orchestrations with a special attention to executability (for rapid prototyping) and to verifiability.

How to deal with real-life procedures followed by healthcare professionals is dealt with in the second contribution: *Simple Modeling of Executable Role-Based Workflows: An application in the healthcare domain* by Tiziana Margarita, Steve Boßelmann, and Bertold Kujath (Margarita *et al.*, 2013). They show on a concrete case study (the clinic everyday processes in the outpatients’ treatment in a large German hospital) the evolution of process modelling techniques and tools from the verbal description to modern executable processes. The perspective here is from the point of view of the engaged professionals in

healthcare (doctors, nurses, administrative personnel) and their need to own the processes and be empowered to analyze and change them at need. The added value of simplicity here is at the end-user side.

3.2. Handling simplicity at deployment and provision time

Complementary to the design, the deployment and provisioning are central islands of complexity and potential bottlenecks to growth and success.

The cloud computing paradigm leverages the idea of Application Service Provision (ASP), meanwhile 15 years old towards complete virtualization and elastic scalability. The third contribution, *Simplifying Software Development and Distribution – Platform as a Service* by Andreas Holubek and Barry D. Floyd (Holubek & Floyd, 2013), addresses the new potential and practice of Software-as-a-Service projects. The paper provides an insight into the development process of applications and their architecture in one "as-a-service" universe using the example of the Force.com platform. It also discusses the limits of this development paradigm - what is already possible and what is the future of this technology.

Security in a networked environment is centrally implemented by using firewalls. The current technology, based on packet filtering, is local, fine granular, and highly technical in its configuration and management. The fourth contribution, *Simplifying Firewall Setups by Using Offline Validation* by Stephan Windmüller (Windmüller, 2013) moves the design, configuration and management of firewalls from the usual turf of complicated domain specific and often cryptic commands stored in some text file, out of reach from the application experts who know the behaviour from a global perspective to a model driven approach that employs a graphical depiction of resources and operations and allows a fit-for-purpose validation of the policies prior to enforcement. Again, abstraction here is a simplicity-generating step that aids comprehension and enables codesign and participation by the application owners.

3.3. Handling simplicity at re-engineering and evolution time

When only the running system is available, or when code and models/documentation have gone so out of synchronicity that the latter have little predictive value for the behaviour of the current system, process mining helps find out what is the real behaviour of the system, for example as a basis for simulation or for migration/substitution. In *Simplifying Translation Validation via Model Extraction* by Falk Howar, Christian Wagner, and Tiziana Margaria, (Howar *et al.*, 2013) the authors show that Behavioural Mining, a technique that extracts analyzable behavioural models from other artefacts (specifications or code), is a practicable and very simple way to obtain a process-oriented description of third-party black box systems. They revisit their previous case study on the NASA's Voyager space mission to automatically discover its behaviour by means of model transformation and automata learning and investigate the conformance of three structurally different types of specification of the case study: (1) a formal specification given in ASSL, (2) a derived implementation in Java, and (3) two behavioural models, one derived from the ASSL specification and one learned from the Java implementation. Validation via model checking of properties of the models shows alignment with the observed behavioural issues of the real system and helps pinpoint their specific cause.

The aim of research on simplicity and work on projects such as ITSy is to reshape key research disciplines with the intent of influencing IT practice. We have meanwhile found support at the highest levels of the European research associations: in the Strategy Paper of the European Forum for Information and Communication Science and Technology (European Forum for Information Science Communication and Technology (EFICST), 2013), we read

- *“Simplicity as a design objective in software and its use within the complex technological products on which we depend daily in our professions and our lives, is gaining paramount importance for the entire society” (p. 5), and*

- “The software in our digital future is yet to be created, but we already know that it will be more complex than today’s. It will contain more opportunities for errors, failures, and security weaknesses. To master this complexity and strive towards simplicity wherever possible, we need to address research challenges in modelling, design, usability, and software economics throughout the software and product lifecycle.” (p. 9)

Indeed, the first recommendation on “Recognising Software as a key enabling technology for Europe’s digital future” is

Recommendation 1: *Software is a key technology of Europe’s digital future. Designing software is a rigorous endeavour guided by scientific principles. The European Forum for ICST will further the development of technologies for (large) software systems that are simpler and more robust under the challenges of the digital future.*

Other communities are already following along these statements of value recognition, providing concrete examples for simplicity-oriented interventions and stating their professional need for an urgent simplicity-driven rethinking in the IT that concerns their professions. One of the most agile communities is the European Society for Emergency Medicine³: its Special Interest Group on Technology and Process of Care has recently endorsed the above strategy paper. But not only researchers and practitioners are inspired by this vision, others are as well. Examples include the “Health as an Asset” global thought leader forum founded by Rutgers State University (New Jersey, US), the Academy of Business in Society and the Johnson & Johnson Corporate Citizenship Trust. These entities all focus on enhanced sharing, use and interpretation of data on global health challenges, as well as building capacity for coordination and collaboration across public, private, and civil sectors and all identify simplicity in IT systems and processes as a key enabler for this to happen.

Supported by these developments, we believe that knowledge gained through research on simplicity by an interdisciplinary effort and the dissemination of this knowledge through our educational institutions will improve the fields of research and the situation of the stakeholders who rely on our efforts for individual and society growth.

References

- Carbone, J. N. & Crowder, J. A. (2011). Transdisciplinary Synthesis and Cognition Frameworks. *Journal of Integrated Design and Process Science (JIDPS)*, 15(1), 35-57.
- Cascini, G. (2012). Triz-Based Anticipatory Design of Future Products and Processes. *Journal of Integrated Design and Process Science (JIDPS)*, 16(3), 29-63.
- European Forum for Information Science Communication and Technology (EFICST). (2013). Shaping the Digital Future of Europe. 16. Retrieved from <http://www.cs.uu.nl/groups/AD/sp-2013.pdf>
- Floyd, B. D. & Bosselmann, S. (2013). Itsy--Simplicity Research in Information and Communication Technology. *Computer*, 46(11), 26-32.
- Hinchey, M. & Coyle, L. (2012). *Conquering Complexity*: Springer-Verlag London, Inc.
- Holubek, A. & Floyd, B. D. (2013). Simplifying Software Development and Distribution - Platform as a Service. *Journal of Integrated Design and Process Science (JIDPS)*, Special Issue on "Simplicity in Information Technology - a chance for a new kind of process science", 17(3), 47-57.
- Howar, F., Wagner, C., & Margarita, T. (2013). Simplifying Translation Validation Via Model Extraction. *Journal of Integrated Design and Process Science (JIDPS)*, Special Issue on "Simplicity in Information Technology - a chance for a new kind of process science", 17(3), 71-91.
- Maeda, J. (2006). *The Laws of Simplicity (Simplicity: Design, Technology, Business, Life)*: MIT Press.
- Margarita, T. (2012). *Formal Methods in the Era of Service Oriented Design*. Proc. COMPSAC 2011, 35th IEEE Annual Computer Software and Applications Conference.

³ EuSEM: <http://www.eusem.org/>. EuSEM covers emergency and disaster medicine, and as such it is also the premiere partner for civil protection issues in every European nation.

- Margaria, T., Boßelmann, S., & Kujath, B. (2013). Simple Modeling of Executable Role-Based Workflows: An Application in the Healthcare Domain. *Journal of Integrated Design and Process Science (JIDPS), Special Issue on "Simplicity in Information Technology - a chance for a new kind of process science"*, 17(3), 25-45.
- Margaria, T., Floyd, B. D., & Steffen, B. (2012). *It Simply Works: Simplicity and Embedded Systems Design*. Proc. Computer Software and Applications Conference Workshops, 35th IEEE COMPSAC.
- Margaria, T. & Hinchey, M. (2013). Simplicity in It: The Power of Less. *Computer*, 46(11), 23-25.
- Margaria, T. & Steffen, B. (2012). Service-Oriented: Conquering Complexity with Xmd. In M. Hinchey & L. Coyle (Eds.), *Conquering Complexity* (pp. 217-236): Springer London.
- Merten, M. & Steffen, B. (2013). Simplicity Driven Application Development. *Journal of Integrated Design and Process Science (JIDPS), Special Issue on "Simplicity in Information Technology - a chance for a new kind of process science"*, 17(4), 9-23.
- Nguyen, T. A. & Zeng, Y. (2012). A Theoretical Model of Design Creativity: Nonlinear Design Dynamics and Mental Stress-Creativity Relation. *Journal of Integrated Design and Process Science (JIDPS)*, 16(3), 65-88.
- Norman, D. (2010). *Living with Complexity*: MIT Press.
- Rommel, G., Kluge, J., Kempis, R.-D., Diederichs, J., & Bruck, F. (1995). *Simplicity Wins: How Germany's Mid-Sized Industrial Companies Succeed*: Harvard Business Press.
- Windmüller, S. (2013). Simplifying Firewall Setups by Using Offline Validation. *Journal of Integrated Design and Process Science (JIDPS), Special Issue on "Simplicity in Information Technology - a chance for a new kind of process science"*, 17(3), 59-69.
- Yan, B. & Zeng, Y. (2011). Design Conflict: Conceptual Structure and Mathematical Representation. *Journal of Integrated Design and Process Science (JIDPS)*, 15(1), 75-89.

Author Biographies

Tiziana Margaria is Chair of Service and Software Engineering at the Institute of Informatics, Universität Potsdam, Germany. Her research interests include model-based system and service engineering as well as process science for healthcare and life sciences. Margaria received a PhD in computer and systems engineering from the Politecnico di Torino, Italy. She is a member of the board of the European Forum for Information and Computation Science and Technology, a Fellow of the Society for Design and Process Science, and vice president of the European Association of Software Science and Technology. Contact her at margaria@cs.uni-potsdam.de.

Barry D. Floyd is a Professor of Information Systems and Management at the Orfalea College of Business, California Polytechnic State University, San Luis Obispo CA, USA. He has worked in industry and academia as a programmer, manager, director and educator both domestically and internationally. He earned his MBA and PhD from The University of Michigan.