Investigating the usability and utility of tangible modelling of socio-technical architectures

Dan Ionita
SCS Group, University of Twente

Roel Wieringa
SCS Group, University of Twente

Jan-Willem Bullee
SCS Group, University of Twente

Alexandr Vasenev
SCS Group, University of Twente

29th May 2015

Contents

List of Figures

List of Tables

Abstract

Socio-technical models are models that represent social as well as technical elements of the modeling subject, where the technical part consists of both physical and digital elements. Examples are enterprise models and models of the target of assessment used in risk assessment. Constructing and validating these models often implies a challenging task of extracting and integrating information from a multitude of stakeholders which are rarely modelling experts and don't usually have the time or desire to engage in modelling activities.

We investigate a promising approach to overcome this challenge by using physical tokens to represent the model. We call the resulting models **tangible** models.

In this paper we illustrate this idea by creating a tangible representations of a socio-technical modelling language used in Risk Assessment and provide an initial validation of the relative usability and utility of tangible versus abstract modelling by an experiment and a focus group, respectively. We discuss possible psychological and social mechanisms that could explain the enhanced usability and utility of tangible modelling approaches for domain experts. Finally, we discuss the generalizability of this approach to other languages and modelling purposes.
Keywords: participatory modelling, tangible modelling, socio-technical, enterprise models, usability experiment

Chapter 1

Introduction

A conceptual model consists of concepts and relations among the concepts, by which people understand a part of the world. The subject of the model may be an individual system or a class of phenomena. Conceptual models are usually represented by (software) modelling tools using abstract graph-like structures containing boxes, arrows, and other symbols. The problem with these abstract representations is that stakeholders whose input or feedback is needed to construct an adequate model may be unfamiliar with the notation.

The issue is exacerbated for socio-technical conceptual models, which are multi-layered by nature and require input from domain experts who may not possess technical knowledge. Extracting and integrating information from a multitude of stakeholders, from different fields, with different knowledge, some of which don't speak formal languages can then be a challenge. Parts of this challenge are inconsistent descriptions provided by various stakeholders or documents and the intricate nature of an organization's internal socio-technical structure. Typically, modelling experts interview domain experts and consult technical documentation in order to build conceptual models.

To facilitate this, we propose to communicate with domain experts using a tangible representation of a conceptual model, that we call a tangible model. This stimulates participation of and collaboration among domain experts and modelling experts, and the evidence presented in this paper suggests that the resulting models are more accurate than abstract representations of conceptual models, and cost less effort to build. This contrasts with traditional approaches, where the modelling task is left up to - often external - consultants with insufficient knowledge of the organisation, by stimulating involvement, interaction and agreement of non-technical stakeholders (such as managers, process owners and specialists).

We have developed and tested this idea in the context of the TREsPASS[1] Information Security project. The goal of this project is to create a toolkit that takes as input a model of an organisation's social, technical and physical infrastructure and outputs a ranked list of potential attack vectors. In order to allow for the kind of model checking required by such an analysis, the model itself needs to be formally specified and reasonably complete. This results in a conceptually and syntactically complex modelling language with a steep learning curve. To facilitate non-technical domain experts' participation in the creation of these models, we experimented with tangible modelling.

In Section 2, we describe related work on tangible modelling and tangible user interface prototyping. In Section 3 we put forward a hypothesis that attempts to decompose the mechanism

[1] Technology-supported Risk Estimation by Predictive Assessment of Socio-technical Security, www.trespass-project.eu

explaining why tangible modelling can leads to more accurate models with less effort and summarize theories of cognitive psychology that could support it. In order to empirically validate our hypothesis, in Section 4 we describe an illustrative conceptual socio-technical modelling language, define a tangible mapping of it and describe an experiment in which we compare the usability of the tangible and abstract representations of this language (Sections 5, 6). We evaluate the utility and applicability of the tangible modelling approach in a focus group with several enterprise modelling experts (Section 7). Finally, we discuss conclusions about generalizability to other languages and further work in Section 8.

Chapter 2

Related work

Barijs [1] claims that collaboration, participation and interaction are directly linked to the completeness and accuracy of conceptual (enterprise) models as well as to the speed and efficiency of the modelling effort itself. *Collaboration* of modellers, analysts and potentially domains experts is needed to construct accurate models of complex socio-technical phenomena in a timely manner. *Participation* of key employees is crucial for capturing the shared understanding of how an enterprise operates in practice, verifying and validating conceptual models of existing systems. *Interaction* should be facilitated by intuitive, easy to use tools which are powerful enough to capture complex interactions while allowing easy manipulation of the constructed models so as to easily observe, for example, the effects of changes.

Grosskopf [6] has shown that tangible modelling could have these effects in process modelling. He used physical tokens to represent abstract process modelling concepts and empirically measured the effects. He provided empirical evidence that by tangible process modelling, practitioners achieved a better understanding, higher consensus and a higher rate of adoption of the results. Fleischmann et al. [3] claim that tangible modelling of organisational processes stimulate stakeholder engagement and ensure coherence of representations on the organisational level, thus paving the way towards acceptance of the results. Garde [4] reports that domain experts with no design skills or process modelling skills successfully successfully designed and specified a complex new procedure within a few hours by means of a participatory "board game", while reporting high levels of commitment and even enjoyment.

Tangible User Interfaces (TUIs) are dedicated to giving physical form to abstract, digital concepts. Early work on TUIs by Fitzmaurice et al. [2] on what they dubbed "Graspable User Interfaces" relied on three key principles: (1) physical artefacts which act as handles for control, (2) the advantage of leveraging people's lifelong experience with the physical world, and (3) space-multiplexed vs. time-multiplexed devices. A TUI gives physical form to digital information, letting it serve as the representation and controls for its digital counterparts. TUIs make digital information directly manipulatable with our hands and perceptible through our peripheral senses through its physical embodiment. Ishii [17] claims TUIs to take advantage of our evolved dexterity and skills in manipulating physical objects. An in-depth analysis of Tangible User Interfaces [21] found that, despite some TUIs being evaluated as less usable than their GUI counterparts, most users prefer the TUI version due to its highly interactive nature, rich feedback and realism.

Rettig [14] describes *low-fidelity* (tangible) prototypes as a means of simulating user interaction with a software tool before actually coding it. He claims this improves the quality of the UI while reducing the number of later, more costly modifications. Furthermore, such lo-fi prototypes are easier to change which makes it easier to try out various layouts.

To sum up, tangible modelling is claimed to speed up the modelling process, to enhance participation and collaboration of domain experts and to lead to more accurate models, all of which promotes a higher rate of adoption of the results. However, the applicability of tangible modelling in multi-layered models in general or in socio-technical models specifically has not been evaluated in existing literature.

In the next section we formulate a causal hypothesis with regard to the effects tangible modelling and review established theories that could back up these claims.

Chapter 3

Theoretical background

We focus on the hypotheses that a tangible collaborative modelling approach can speed up the modelling process and that it can improve the quality of the resulting models, especially when these models need to integrate knowledge from various fields and various stakeholders.

We decompose these overall hypotheses into three more detailed hypotheses (Figure 3.1):

H1 Physical representations of a conceptual model are easier to understand and manipulate than abstract representations;

H2 The participatory aspect encourages engagement, collaboration and - more importantly - agreement between stakeholders;

H3 Physical and participatory modelling, similar to board games, increases engagement while reducing repetitiveness.

H1 implies that modelling will take less effort. H2 and H3 imply that modelling produces more engagement of all stakeholders than in abstract modelling, and this implies in turn that there is a bigger chance that it will result in an accurate agreed-upon model. (We assume here that more agreement implies higher accuracy of the required model.) We have identified several cognitive theories that support our detailed hypotheses:

Constructionism is based on the ideas of Seymour Papert, who built in turn on the Constructivist theories of Papert's colleague Jean Piaget. Papert [9] argued that learning about a product improves when people are engaged in constructing it. Thus, understandability of both

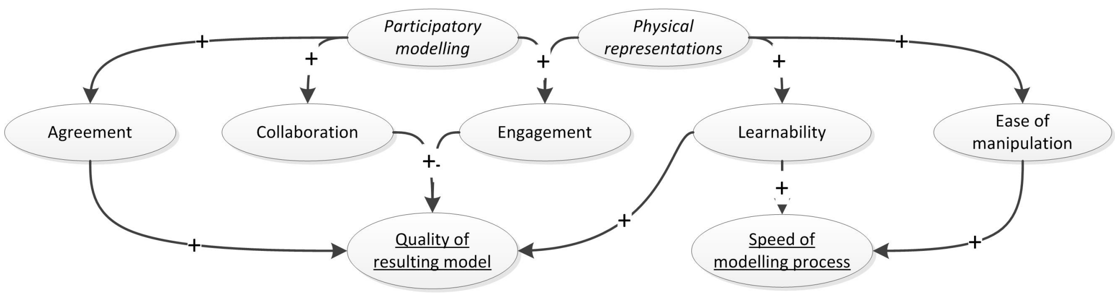

Figure 3.1: Causal graph describing our hypotheses. The nodes in *italics* are the variables we want to influence. The underlined nodes are the target variables

the modelling concepts and the model itself can be enhanced by engaging a group of people in the construction of the model. Increased understandability by people normally not actively involved in the modelling effort can, in turn, facilitate collaboration and agreement and provides a platform for promoting awareness. The underlying mechanism is that a shared focus external and a shared activity, enhances collaboration. This is supported by the empirical findings of Garde [4], Fleischmann [3] and Ishii [17].

Gamification is defined by Marczewski [11] as "the application of gaming metaphors to real life tasks to influence behaviour, improve motivation and enhance engagement". By building the models around a table instead of in front of a screen, we hope to make the task more similar to a board game than to creating a diagram. Furthermore, we suggest selecting game-like tokens, such as Lego characters, playing card shapes and colours to further enhance the board-game metaphor. As confirmed by the findings of Garde [4], participants have more fun doing this than they have when manipulating abstract representations, thus creating more participation and engagement.

Cognitive load is to the total amount of mental effort required to perform a task. The theory of cognitive load suggests that performance is increased when conditions are aligned with the human cognitive architecture [15]. The mechanism is that this reduces the strain on short-term memory. Miller [12] claims that the ability to remember and discriminate information can be dramatically expanded by adding dimensional stimuli (such as color, sound, material & space). Hecht et al. [7] further demonstrated that adding a haptic signal to visual and audio stimuli enhance perception performance. The authors claim that tri-modal (visual, auditory and haptic) interaction enable users to pay attention to a wider range of details and subtle cues. This supports the hypothesis that physical tokens are easier to understand, remember and reason about, due to the multitude of identifiable features such a token exhibits when compared to a two-dimensional uni-model (visual) representation.

Cognitive fit is the reduction of cognitive load by "fitting the representation to the problem". According to the theory of cognitive fit, this leads to the use of familiar problem-solving process as there is no need to transform the mental representation to the problem. Vessey [18] showed that when the representation of the concepts or information matched the task, the problem solving performance for symbolic tasks was drastically increased. This provides further support that using physical representations of concepts can enhance the modelling abilities of the participants, as tangible models potentially bear more similarities to the modelling target world than abstract representations. Their evidence also supports our hypothesis that intuitive physical representations are easier to understand and manipulate by domain experts than abstract representations. Teets et al. [16] goes even further by claiming that cognitive fit is a reliable way of evaluating the effectiveness of information visualisations to support decision making.

Chapter 4

Mapping a conceptual language to a tangible representation

To validate hypotheses about the relative advantage of tangible modelling with regard to abstract modelling, we test them for one particular modelling language, described here. After the tests, we will first discuss to what extent our evidence supports the more our hypotheses for this modelling language. Next, we will discuss if the evidence also supports our hypotheses for conceptual modelling in general.

4.1 The TREsPASS modelling language

TREsPASS[1] is a European project aimed at with automating the Information Security Risk Assessment process. The input of any risk assessment process is a conceptual model of the socio-technical architecture of the Target of Assessment (ToA) and its environment. This model should define the existing business roles, physical locations, physical objects, assets, computing devices and IT networks, as well as the (access) relationships among them in a formal manner. Creating such a model requires in-depth knowledge of ToA and its environment, as well as of the modelling language. Knowledge of the language cannot be assumed to be present in domain experts. However, these experts are also needed to build an accurate TREsPASS model of the ToA and its environment.

The TREsPASS modelling language contains the following concepts:

- **Actors** are persons or roles relevant to the model.
- **Locations** are containers. Currently, three types of locations are defined:
  - *Physical locations* are 3-dimensional containers such as buildings or rooms;
  - *Network locations* are fixed hardware such as servers, or databases;
  - *Device locations* are mobile hardware such as phones or laptops.
- **Assets** are items and data relevant to the model:
  - *Item assets* are three-dimensional objects one can physically have and/or move around, such as a card or a key;

[1] http://www.trespass-project.eu/

  - *Data assets* represent a piece of knowledge and/or digital asset, such as a PIN or a password.

- **Relationships** are complex concepts that can be used to represent:
  - *Position* of actors. For example, an actor might be positioned in a room, or outside the building.
  - *Possession* of assets. For example, an access-card or password could be in possession of an actor.
  - *Containment* of network/device locations. For example, a network or device location could be located inside a physical location.
  - *Connection* of locations. For example, two physical locations can be connected if it's possible to move from one into the other. Similarly, network and device locations can be connected if they can communicate.

- **Access Policies** enable access to (physical, network or device) locations, given the right credentials. Credentials can be data assets (such as a password) or item assets (such as a key).

The current version of the TREsPASS toolkit provides abstract representations of these concepts, as listen in the second column of Table 4.1 below.

4.2 The tangible TREsPASS modelling language

To cross the gap between this abstract language and the knowledge of domain experts, we defined a tangible representation of the same concepts as follows, as listed in the third column of Table 4.1.

1. We started off with a whiteboard as the basis for our model in order to allow drawing and easy erasing. Furthermore, whiteboards are commonplace, are metallic (as to allow magnetic tokens to be attached) and can be hanged on the wall once the model is complete.

2. We selected a specific colour marker for drawing the physical layer. This is because we assumed it is intuitive to start with modelling the physical layer and add the other layers on top afterwards.

3. Lego characters were selected to represent actors due to the low price, large diversity, high availability and familiarity.

4. Cards representing various network components, part an open-source Network Security board game [5] were printed. These can be customized with relevant components. There are even blank cards which users can customize on-the-fly.

5. To specify an access policy, a sticky-note is attached to the desired (physical/network) location. On the sticky-note, a colour is written, indicating the credential needed for access. Optionally, a legend can be used to map colours for credentials to their real-life meaning.

6. Textual notes can be written next to any physical token, for example for naming.

Table 4.1: Mapping of concepts to representations

Concept	Software representation	Tangible representation
Actor	Stickman	LEGO®character
Asset (item)	Solid circle	LEGO®item[a]
Asset (data)	Dotted Circle	LEGO®mini-brick[b]
Location (network)	Box (green)	Hexagonal card
Location (physical)	Box (yellow)	Box
Location (device)	Box (Blue)	Card
Access policy	Text-box	sticky-note
Relationship (position)	Solid line	Physical overlap
Relationship (possession)	Dotted line	Physical attachment
Relationship (containment)	Directional arrow	Physical overlap
Relationship (connection)	Bi-directional arrow	Line

[a] A LEGO®item resembles a real-world object and can be placed in a LEGO®characters' hand

[b] A LEGO®mini-brick is the smallest LEGO®brick available, usually of circular or cylindrical shape and can be placed on a LEGO®characters' head

7. Finally, we placed magnetic tape on the back of the cards and attached all LEGO®elements to magnetic LEGO®bricks so as to allow the finished model to also be placed vertically (for example to be hanged on a wall).

Figure 4.1 shows a TREsPASS tangible modelling toolkit and Figure 4.2 shows part of a tangible model created using these conventions.

Figure 4.1: Tangible TREsPASS modelling kit

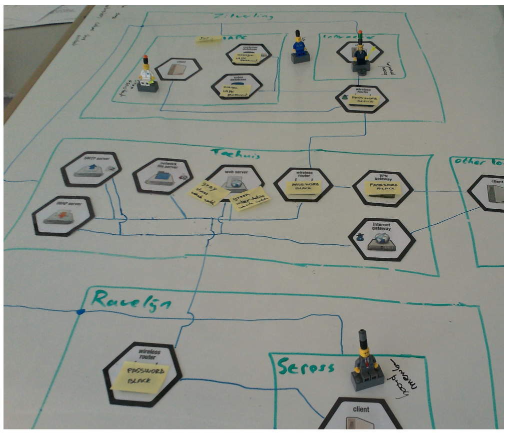

Figure 4.2: Part of a tangible TREsPASS model

Chapter 5

Usability experiment

Following a pilot proof-of-concept experiment with Information Security PhD students from our department, we made some small improvements (such as deciding to add magnets to all the elements and differentiate between each color marker) and we decided the approach is feasible. We have set up an experiment in order to answer the research question:

Q *"Is tangible modelling more usable than abstract modelling?"*

The following description of the experiment follows the design science methodology guidelines outlined by Wieringa [20].

Since, in order to get an accurate measure of usability, everything else had to be maintained constant, we could not measure the utility provided by tangible modelling within the same experiment. Rather, a focus group of enterprise modelling experts was set-up to evaluate the utility of the approach (Section 7).

5.1 Object of study

The target population, which is the ultimate target of our generalizations, consists of the tuple <group of experts; modelling target>. We are interested in the difference between the effects of using tangible and abstract representations of conceptual models in this population. For the experiment, we selected a sample of this population, namely <group of student volunteers; Study associations>.

Choice of volunteers

We invited students from the Business Administration and Computer Science faculties at the University of Twente. A total of eight volunteers were split into two independent groups, each containing a mix of students from various tracks.

This choice was motivated by the following external validity considerations: The intended users of complex, multi-layered models are inter-disciplinary teams of domain experts. For enterprise models, these teams typically consist at least of Business Experts and IT stakeholders. The choice of Business Administration and Computer Science student simulates this division of expertise to some extent.

Students are not necessarily representative of experienced domain experts. However, we assume the cognitive theories identified in Section 3 are equally applicable to both experts and students. If the experimental outcome can be explained by these general theories, then this provides some support for the claim that similar outcomes will occur for domain experts. Such a claim would of course have to be substantiated by further research.

While we tried to balance the groups as much as possible, there is still the possibility that some of the participants had more modelling experience, lay-outing expertise or were simply more skilled. This is a threat to internal validity, as it is a possible cause of differences in group outcomes, unrelated to the difference in modelling approaches. To measure this threat to validity, any variations in group behaviour and dynamics were noted throughout the experiment.

Modelling target

The modelling target should ideally be a socio-technical system the participants are familiar with, such as their own organization. Thus, we asked the student volunteers to create a model showcasing the physical layout, network infrastructure (both servers and clients), important roles and associated access policies of their own Student Associations. Students from Computer Science might be less aware of the structure of the student association of Business Administration, and vice-versa. This simulates the disjunct knowledge individual domain experts might have with regard to the model and therefore improves external validity of the choice of modelling target. To limit variation due to lack of knowledge, as well as the effect of pre-existing knowledge, each group was provided with a half-page description listing the core components of each association.

Since we are measuring how well the tools and concepts perform, not how familiar each participant is with the modelling target, the participants were allowed to ask questions with regard to the modelling target at any time during the exercise to mitigate the effect lack of familiarity with the system might have on the resulting model.

5.2 Treatment design

Each group was shown a brief description of the system and an outline of the task they have to perform. These were identical for both groups. The groups are given as much time as they need to understand this description.

Each group was then given a specification of the modelling concepts of the TREsPASS language, using the definitions listed in Section 4. One group received the mapping of concepts to conceptual representations available in the software tool (the first two columns of Table 4.1), the other a mapping to tangible tokens (the first and last column of Table 4.1). Thus, we define C as the independent variable representing the different possible treatments that are to be applied to the OoS, and can take the value of *tangible* or *software*.

Each group was allowed to ask questions pertaining to the concepts. We measured the time it took each group to declare that they understand the concepts and their representations and counted the number of questions they ask with regard to the modelling language description. Once each group declared the concepts are clear and that they are ready to start, they were given an unlimited amount of time to create a model of the given system – to the best of their abilities – using the only the concepts provided. Once the modelling started, the moderator only intervened when the participants had questions about the modelling target or when the group agreed that they were done. An exit questionnaire is administered and the resulting model is photographed. At the end of the experiment, each student was rewarded with a 50 Euro gift-card, with bonus movie tickets awarded by means of a raffle.

5.2.1 Treatment validity

Participants were only informed of the treatment that is to be applied to their group. They were informed of their task before-hand, as well as of the measurements which will be taken. However, the goal of the experiment, the research it is part of, the task of the other group and any other questions that might influence the outcome are only revealed during de-briefing, thus mitigating the threat that knowledge of the modelling approach of the other group could influence their own modelling performance.

By using a modelling language that is still under development we minimized the chance that the participants have pre-existing knowledge of the concepts, thus eliminating another possible cause of differences in outcome.

To eliminate experimenter expectancy, we told each group that their goal is to finish as fast as possible while maintaining consistency with the system description.

A threat to validity we were unable to mitigate was the quality of the software tool. The ability of the tool to manipulate diagrams, as well as the developer's choice on how to represent the concepts directly impact its usability and thus may have biased the results in favour of the tangible approach.

5.3 Measurement design

Our aim is to measure and compare the utility and usability of tangible modelling versus computer-based modelling. Utility is commonly defined as how well a method satisfies a user's needs. Since, within the scope of the experiment, both methods take the same input and aim at producing the same result, essentially providing identical functionality, we cannot make a claim about comparative utility of the two approaches from this experiment.

Based on recommendations of Nielsen [13], we operationalize usability in terms of the following indicators:

- Learnability: Time needed to understand the modelling language and number of question asked with respect to it
- Efficiency: Time needed to construct model
- Correctness: How well the concepts were applied
- Self-reported usability

Learnability was measured as the time interval between being presented with the concepts and definitions, and the time the group declares they are ready to start. We also counted the number of questions the group needed answers to before starting.

Efficiency was measured as inverse of the total time needed to construct the model.

Correctness is measured as the number of errors at the end of the modelling process. We distinguish three types of errors: (1) Placing en element where none was expected, (2) Missing element where one was expected and (3) Using a wrong concept to represent an element. We used a model created in advance as a standard against which to measure correctness.

Satisfaction was measured via an exit-questionnaire, containing 4 questions, to be answered on a 5-point semantic dfferential scale with the end-points labeled i(e.g. 1: Very Difficult to 5: Very Easy). The questionnaire design was based on the ASQ (After-Scenario Questionnaire) described by Lewis [10], containing questions on ease of task, time on task, tool satisfaction and group agreement. To maximize accuracy, the questionnaires were administered to each participant individually right after the group declared they are finished. The answers were averaged per group. The full questionnaire is given in Appendix A.

5.3.1 Measurement validity

In order to make sure that we are indeed measuring the effects of the method, and not something else, we need to control variation due to other causes. Such causes might be internal (for example, due to improper description of the concepts), or contextual [19] . We controld for internal causes by providing the same definitions to all groups, irrespective of the method used.

Contextual causes are always present and more difficult to control for. According to Vriezekolk [19] contextual causes for variation are due to the subjects applying the method, the environment during application or other aspects related to the context in which the method was applied. We tried to minimise these by using the same room, layout and instructions for both groups. Variations might still appear due to different skill levels within each group, which are harder to control for.

Chapter 6

Results

6.1 Measurements and observations

The quantitative results of the experiment are listed in Table 6.1. Photos of the models created by the two groups can be found in Appendix B. In terms of the indicators defined in the previous section, we can observe several improvements when using the tangible method:

- Learnability: similar time but less questions
- Efficiency: 52% faster
- Errors: half as many
- Self-reported usability:
  - 12.5% increased agreement
  - 54% more satisfied with the tools
  - 25% easier
  - 24% faster

Further, qualitative observations, made by the supervisor of the experiment:

- The tangible modelling group worked quite fluently, and did not spend to much time on any particular phase.
- The software group split up into two sub-groups in an attempt improve the process but later had trouble defining the relationships between concepts.

Table 6.1: Results of the student experiment

	Learnability		Efficiency	Errors	Self-reported usability			
	Time	Questions	Time	#	Agreement	Satisfaction	Difficulty	Length
Software modelling	11min	7	1h42min	7	4	2.75	4	4.25
Tangible modelling	12min	4	1h7min	3	4.5	4.25	3	3.25

6.2 Discussion

Some group members provided reasons behind the usability scores reported in the exit questionnaire. Many rationales of the group using the abstract notation mentioned weaknesses of the Graphical User Interface or the way concepts are represented in the software tool as obstacles, and this confirms the bias against the abstract notation due to the quality of the software tool, that may be present in this experimental setup.

However, this group also mentioned other factors that impeded their modelling: (1) cluttering due to small screen size and tendency to try fitting everything into the screen to avoid scrolling, (2) lines and arrows overlapping with each other or with objects (due to inability to easily trace custom line shapes), (3) when many components are added, the model becomes hard to understand and de-bug and (4) making changes is hard and can have adverse effects on the understandability of the model. These factors remain present even if another tool were used, and this lends some support to the claim that the difference in outcomes was not only due to the quality of the software tool of the TREsPASS notation but also to the advantages of tangible modelling.

The tangible group provided less explanations for their scores, mostly discussing the following: (1) not all elements were used/useful, (2) the model exploded in size, (3) the modelling itself was not difficult if the system description is clear and (4) most of the time was spent discussing details with the team.

Chapter 7

A focus group to assess utility

In order to gain insight with regard to the applicability of the method in practice, we conducted a demo session at BiZZdesign[1], a company providing consultancy on the enterprise architecture modelling language ArchiMate [8], an industry standard in Enterprise Modelling.

Eight consultants participated in a 2-hour demo and workshop aimed to generating feedback and discussions on the topic of tangible modelling for enterprise architectures. After a presentation describing the overall approach, introducing the TREsPASS language and its tangible mapping, they were asked to collaboratively create a tangible TREsPASS model and later to envision the possible usefulness of a similar approach to ArchiMate.

They indicated several application scenarios where tangible modelling of enterprise architectures might provide utility:

- Architecture modelling sessions with domain experts (not modelling or architecture experts). They base this claim on their observation that, during modelling, less technical people tend to have a stronger impact on the model, as they can now manipulate the concepts themselves, and do not rely on a "modeller" to parse their input.
- Early stages of design where different types of stakeholders have to come up with an architecture; They claim this is because the participative aspect increases collaboration and encourages imagination.
- Models built with the goal of increasing awareness and feeling of involvement of employees with regard to the internal structure of the company. They claim this is because non-technical people can more easily understand the tangible model, which could be displayed somewhere within the company. Potentially, employees could be allowed to tweak it, thus taking enterprise architecture out of the architecture department.

[1] www.bizzdesign.nl

Chapter 8

Conclusions and future work

To answer our research qustion, in our experiment, the tangible modelling approach outperformed conceptual modelling on almost all of the measured indicators.

The tangible representations resulted in slightly increased learnability. Users of tangible modelling also reported lower task difficulty and higher satisfaction with the tools. Overall, this reduction of the modelling effort led to a significant reduction in both the perceived and measured time needed to construct the model.

Furthermore, the tangible group did not divide tasks, suggesting increased collaboration. While we were unable to measure discussion, we did observe a decrease in the number of errors in the final model. The tangible group also reported higher agreement with the resulting model.

Both measured and self-reported usability was higher when using tangible tokens versus software. This provides support that such an approach is more intuitive and understandable by domain experts and encourages collaboration and engagement while fostering discussion and ultimately agreement.

While we were unable to control for all other contextual causes (such as participant skill or limitations of the software tool), the comments by the subjects that there are important factors favouring a tangible modelling approach over a tool-supported approach with abstract notations. This is especially the case when the (group of) experts in possession of the real-world knowledge needed as input for the model are not familiar with the ontology and syntax of the modelling language itself.

The improvements noticed were similar to the ones of tangible process modelling reported in [6, 3, 4], which provides further validity as well as leading us to believe that the approach could be further generalized to other complex, practice-oriented modelling notations.

Consistency with the theoretical explanations provided in Section 3 also lends support to the generalizability of these results.

8.0.1 Generalizability

Because the positive effect of tangible modelling on usability can be explained by general theories of human cognition, we expect similar effects in similar situations, i.e. situations in which a group of experts with different knowledge, some of which not familiar with modelling languages and not willing to learn it, need to jointly produce a model where each of their knowledge is needed.

An approach to creating a tangible mapping, similar to the one described in Section 4.2 could theoretically be applied to virtually any conceptual framework or modelling language. Even when intuitive physical representations cannot be found or are not available for some of the concepts, one can fall back on blank cards or even color-coded sticky-notes, similar to the tangible process modelling approach described by Grosskopf [6].

It is, however, clear that tangible modelling is not useful for all socio-technical modelling application scenarios, and the approach needs to be further validated to refine our generalizability claims. For example, lay-outing experts might overcome the limitations of the software tool. Simple modelling targets or very effective tooling would also reduce the added benefit or creating a tangible model. If the desired end-result is a formal (software) model, this benefit is further reduced, due to the need for conversion to an abstract model that can be manipulated by software.

However, our focus group indicates that a tangible modelling approach may be useful when the awareness and commitment of domain experts is required. The need for awareness and commitment of domain experts exists at the start of enterprise architecture processes. We intend to explore this further by replicating this experiment with business analysts with an enterprise modelling language.

Another scenario in which tangible modelling may be useful is IT security risk assessment, as it seems to be easier to come up with attacks when looking at a tangible model compared to an abstract diagram. Furthermore, a similar approach might also be useful for teaching: training of people unfamiliar with a modelling language.

To enhance the utility of a tangible modeling approach in these scenarios, we plan to explore tool support for automatic conversion from a tangible model and an abstract representation of the model.

8.1 Future work

- Repeating experiment with other languages (such as ArchiMate) to verify validity as well as improve quality and scope of generalizability.
- Mixing technical with business people and see if the business people's opinion is reflected in the model. In software modelling, probably not (or less). In tangible modelling, more so.
- Automation of tangible <-> software conversion. For example, via an over-head camera.
- Investigating value in teaching

8.1.1 Acknowledgment

The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement ICT-318003 (TREsPASS) as well as from the Joint Program Innitiative (JPI) Urban Europe via the IRENE project. This publication reflects only the authors' views and the Union is not liable for any use that may be made of the information contained herein.

Appendix A

Exit-questionnaire

1. How would you describe the difficulty of the task you just completed?
   1(Very easy) 2 3 4 5(Very difficult)

2. How satisfied are you with the tools provided to complete the task?
   1(Very unsatisfied) 2 3 4 5(Very satisfied)

3. How would you rate the amount of time it took to complete the task?
   1(Very little time) 2 3 4 5(Too long)

4. How much do you agree with the final version of the model?
   1(Don't agree) 2 3 4 5(Fully agree)

Appendix B

Models created during the experiment

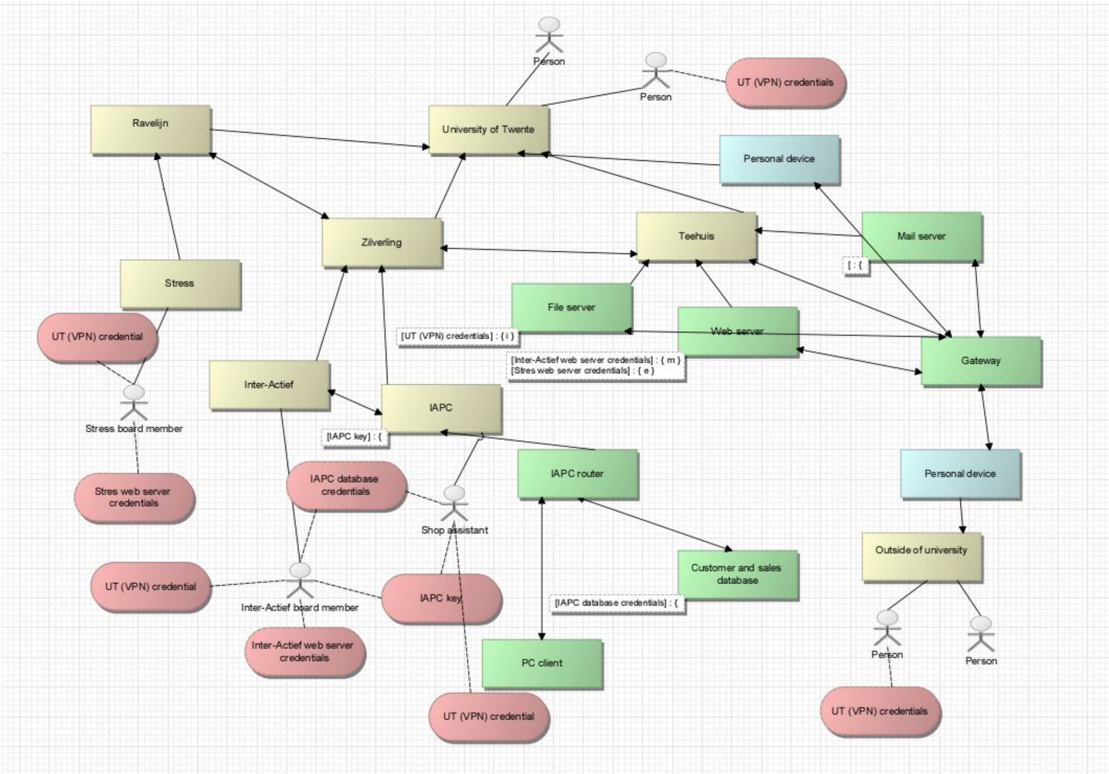

Figure B.1: Model created by the group working with the software tool

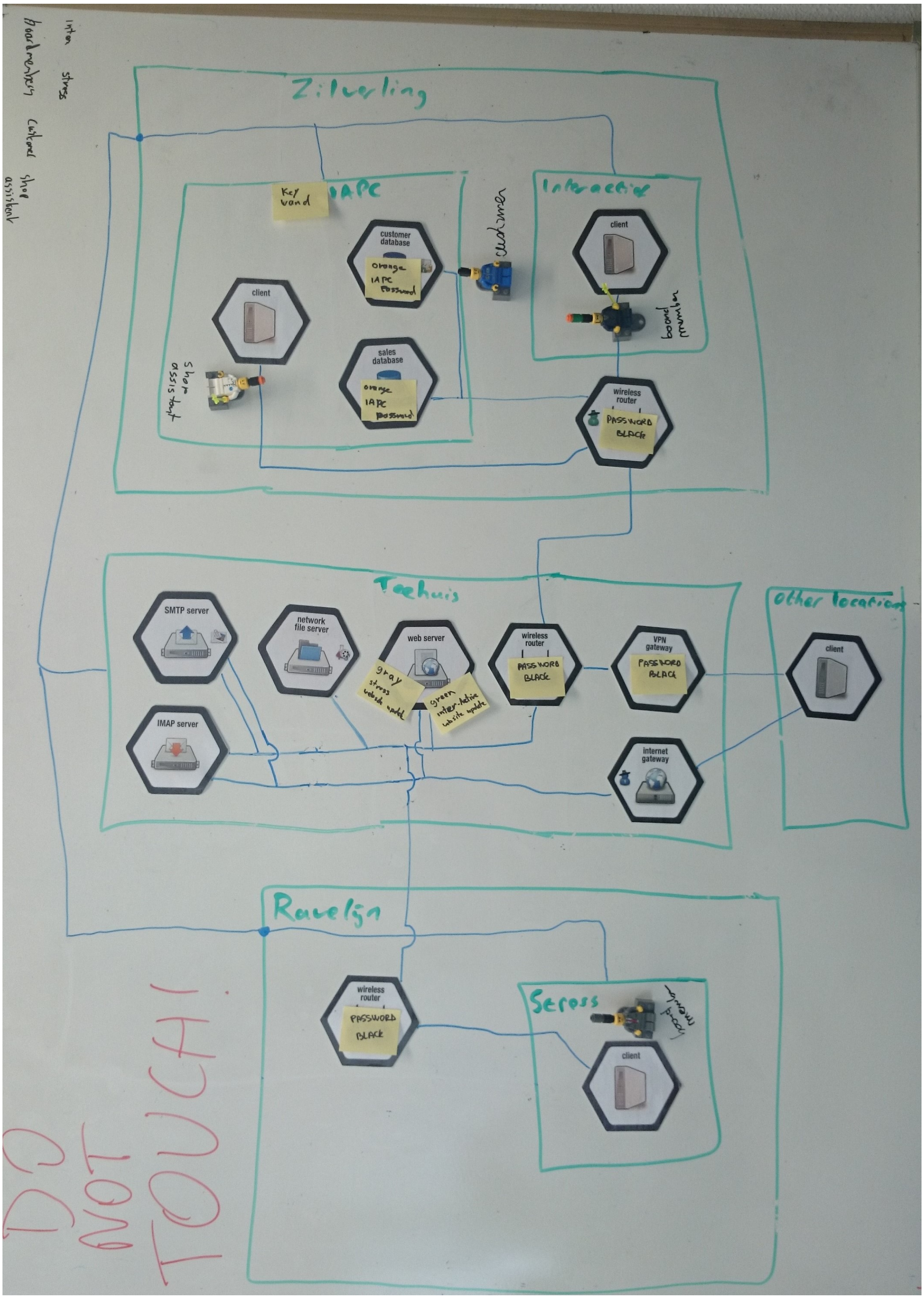

Figure B.2: Model created by the group working with tangible tokens

Bibliography

[1] J. Barjis. Collaborative, participative and interactive enterprise modeling. In Joaquim Filipe and José Cordeiro, editors, *Enterprise Information Systems*, volume 24 of *Lecture Notes in Business Information Processing*, pages 651–662. Springer Berlin Heidelberg, 2009.

[2] G. W. Fitzmaurice, H. Ishii, and William A. S. Buxton. Bricks: Laying the foundations for graspable user interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '95, pages 442–449, New York, NY, USA, 1995. ACM Press/Addison-Wesley Publishing Co.

[3] A. Fleischmann, W. Schmidt, and C. Stary. Tangible or not tangible – a comparative study of interaction types for process modeling support. In *Human-Computer Interaction. Advanced Interaction Modalities and Techniques*, volume 8511 of *Lecture Notes in Computer Science*, pages 544–555. Springer International Publishing, 2014.

[4] J.A. Garde and M.C. van der Voort. The procedure usability game: A participatory game for development of complex medical procedures & products. In *Proceedings of the CIRP IPS2 Conference 2009*, 2009.

[5] M. Gondree and Z. N.J. Peterson. Valuing security by getting [d0x3d!]: Experiences with a network security board game. In *Presented as part of the 6th Workshop on Cyber Security Experimentation and Test*, Berkeley, CA, 2013. USENIX.

[6] A. Grosskopf, J. Edelman, and M. Weske. Tangible business process modeling – methodology and experiment design. In *Business Process Management Workshops*, volume 43 of *Lecture Notes in Business Information Processing*, pages 489–500. Springer Berlin Heidelberg, 2010.

[7] D. Hecht, M. Reiner, and A. Karni. Enhancement of response times to bi- and tri-modal sensory stimuli during active movements. *Experimental Brain Research*, 185(4):655–665, 2008.

[8] M.E. Iacob, Dr. H. Jonkers, M.M. Lankhorst, E. Proper, and Dr.ir. D.A.C. Quartel. Archimate 2.0 specification: The open group. Van Haren Publishing, 2012.

[9] Y. Kafai and I. Harel. Learning through design and teaching: Exploring social and collaborative aspects of constructionism. In I. Harel and S. Papert, editors, *Constructionism*, pages 85–110. Ablex, Norwood, NJ, 1991.

[10] J. R. Lewis. Psychometric evaluation of an after-scenario questionnaire for computer usability studies: The asq. *SIGCHI Bull.*, 23(1):78–81, January 1991.

[11] A. Marczewski. *Gamification: A Simple Introduction.* Andrzej Marczewski, 2013.

[12] G. Miller. The magical number seven, plus or minus two: Some limits on our capacity for processing information, 1956. One of the 100 most influential papers in cognitive science: http://cogsci.umn.edu/millennium/final.html.

[13] J. Nielsen. Usability 101: Introduction to usability. *Jakob Nielsen's Alertbox*, 2003.

[14] Marc Rettig. Prototyping for tiny fingers. *Commun. ACM*, 37(4):21–27, April 1994.

[15] J. Sweller. Cognitive load during problem solving: Effects on learning. *Cognitive Science*, 12(2):257–285, 1988.

[16] J. M. Teets, D. P. Tegarden, and R. S. Russell. Using cognitive fit theory to evaluate the effectiveness of information visualizations: An example using quality assurance data. *IEEE Transactions on Visualization and Computer Graphics*, 16(5):841–853, 2010.

[17] J. Underkoffler and H. Ishii. Urp: A luminous-tangible workbench for urban planning and design. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '99, pages 386–393, New York, NY, USA, 1999. ACM.

[18] I. Vessey and D. Galletta. Cognitive fit: An empirical study of information acquisition. *Information Systems Research*, 2(1):63–84, 1991.

[19] E. Vriezekolk, S. Etalle, and R. Wieringa. Experimental validation of a risk assessment method. In *21st International Working Conference on Requirements Engineering: Foundations for Software Quality (REFSQ 15)*. Springer, 2015.

[20] R. J. Wieringa. *Design Science Methodology for Information Systems and Software Engineering*. 2014.

[21] O. Zuckerman and Ayelet Gal-Oz. To tui or not to tui: Evaluating performance and preference in tangible vs. graphical user interfaces. *International Journal of Human-Computer Studies*, 71(7–8):803 – 820, 2013.