predict
prioritise
prevent

TRESPASS

Technology-supported Risk Estimation by Predictive Assessment of Socio-technical Security

Deliverable D4.1.2

Final requirements for visualisation processes and tools

Project:	TRESPASS
Project Number:	ICT-318003
Deliverable:	D4.1.2
Title:	Final requirements for visualisation processes and tools
Version:	1.0
Confidentiality:	Public
Editor:	L. Coles-Kemp, C. Heath
Cont. Authors:	L. Coles-Kemp, C. Heath, P. A.Hall, R. Trujillo, J. Barendse
Date:	2015-10-30

Part of the Seventh Framework Programme
Funded by the EC-DG CONNECT

Members of the TRESPASS Consortium

1.	University of Twente	UT	The Netherlands
2.	Technical University of Denmark	DTU	Denmark
3.	Cybernetica	CYB	Estonia
4.	GMV Portugal	GMVP	Portugal
5.	GMV Spain	GMVS	Spain
6.	Royal Holloway University of London	RHUL	United Kingdom
7.	itrust consulting	ITR	Luxembourg
8.	Goethe University Frankfurt	GUF	Germany
9.	IBM Research	IBM	Switzerland
10.	Delft University of Technology	TUD	The Netherlands
11.	Hamburg University of Technology	TUHH	Germany
12.	University of Luxembourg	UL	Luxembourg
13.	Aalborg University	AAU	Denmark
14.	Consult Hyperion	CHYP	United Kingdom
15.	BizzDesign	BD	The Netherlands
16.	Deloitte	DELO	The Netherlands
17.	Lust	LUST	The Netherlands

Document History

Authors		
Partner	Name	Chapters
RHUL	Lizzie Coles-Kemp, Claude Heath, Peter Hall	All
UL	Rolando Trujillo	1,2
LUST	Jeroen Barendse	Appendix A

Quality assurance		
Role	Name	Date
Editor	Lizzie Coles-Kemp	2015-09-30
Reviewer	Henk Jonkers	2015-10-30
Reviewer	Lorena Montoya	2015-10-30
Task leader	Lizzie Coles-Kemp	2015-10-30
WP leader	Lizzie Coles-Kemp	2015-10-30
Coordinator	Pieter Hartel	2015-10-30

Circulation	
Recipient	Date of submission
Project Partners	2015-10-30
European Commission	2015-10-30

Acknowledgement: *The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement no. 318003 (TRESPASS). 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.*

Contents

List of Figures

List of Tables

Management Summary

This deliverable lists the final selection of visualisation requirements for the TRESPASS project. The deliverable also explains how the requirements were selected, the rationale behind the requirements selection and shows the progress to date in meeting those requirements.

Key takeaways:

- Visualisation requirements have been identified.
- Visualisation requirements have been validated through a number of activities.
- Good progress has been made on operationalising those requirements.

As this report shows, the requirements for the visualisation component of the TRESPASS tool chain are reasonably mature and the focus is now on refining the visualisations and evaluating these through user panels.

As the examples in this deliverable show, several of these requirements are operationalised at any one time when visualising an aspect of information security risk. The operationalisation of these requirements can be further seen in the Attack Navigator Map prototypes that have been produced to date. The visualisation requirements contribute to the development of the Attack Navigator Map by informing the visual aesthetics of the interface.

The prototype together with the visualisations presented in this deliverable is available at trespass.lustlab.net, in a specially devised demo environment. Here the various demos can be tried out, comments on demos can be left, and demos are explained by a read-me text file. The login details to this prototype platform are provided with the deliverables.

Once you are logged in you can browse all available prototypes through the demo interface, including older prototypes and visualisation demos (Fig. 1). The visualisations presented in this deliverable can be found online at: http://trespass.lustlab.net/proto/anm-wizard-workflow You can access the Attack Navigator directly via: http://trespass.lustlab.net/proto/an and the Attack Navigator Map via: http://trespass.lustlab.net/proto/anm

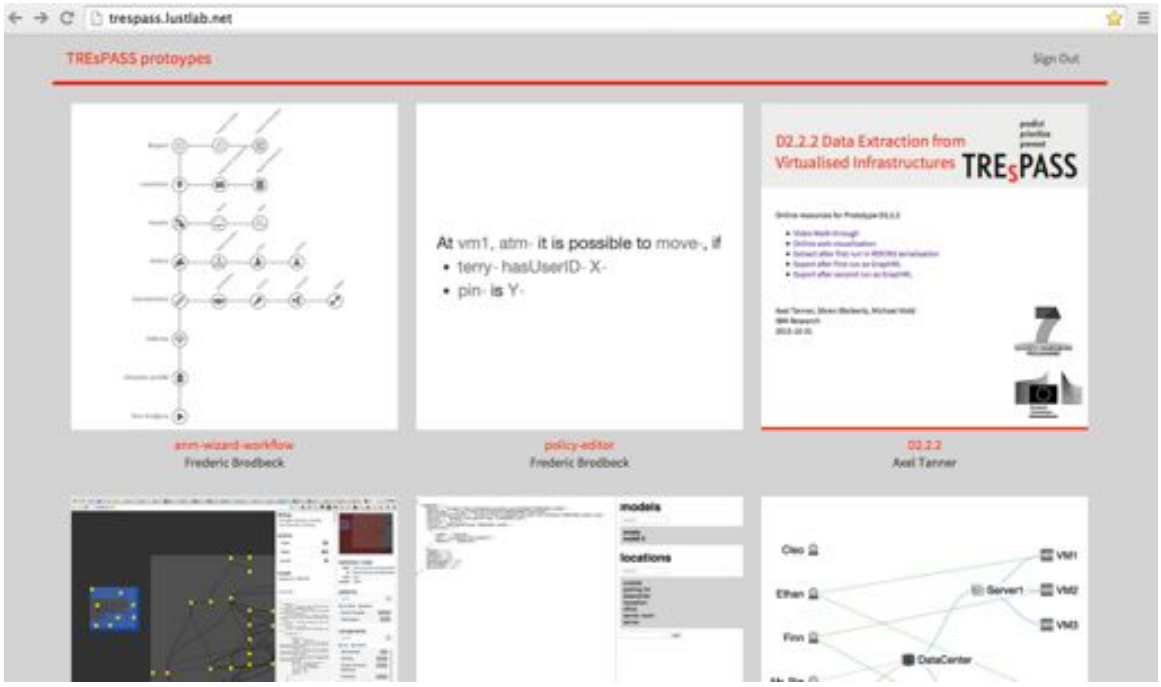

Figure 1.: LUSTLab prototype index, screenshot.

1. Introduction

1.1. Goals

The goal of this deliverable is to present the final requirements for visualisation in the TRESPASS project. These requirements address both the visualisation processes and tools. Two other goals are also identified: validation of these requirements, and showing the progress to date in the operationalisation of the requirements.

1.2. Choices made

The visualisation requirement choices that are presented in this deliverable have been made based on a number of inputs:

- Review of the state of the art.
- Inputs from the TRESPASS community, including TRESPASS reviewers, peer-review and consortium review.
- Practitioner inputs via WP4's practitioner panel and dissemination programme.
- TRESPASS requirements identification task-force.

In this deliverable report we remind readers of the choices that we made based on the state of the art, we present the visualisation requirements resulting from the deliberations of the TRESPASS community and we explain how we have engaged with the security practitioner community to evaluate and validate the visualisation requirements produced as a result of the state of the art review and engagement with the TRESPASS community.

1.3. Foreground and background

The current prototypes of the TREsPASS user interface, namely the Attack Navigator and one of its main tools, the Attack Navigator Map, are foreground to the project. Early versions of some TREsPASS tools (ADTool (v1.1), DFTCalc, Approxtree etc.) are background of the project. The individual tools have been improved by their development teams and constitute foreground of the project. TREsPASS.js is foreground to the project, other

libraries are background to their respective developers (d3, angular, react, bootstrap, keystone). Everything else in this document and on the prototype website[1] is foreground.

1.4. Document structure

The document is structured into four parts:

- An overview of the history and rationale behind the visualisation requirements (Chap. 2).
- A presentation of the final set of visualisation requirements (Chap. 3).
- A presentation of progress to date in operationalising the requirements (Chap. 4).
- An overview of the validation activities to date (Chap. 5).

Appendix B provides the context for this deliverable in the TRE$_S$PASS project. It describes the overall summary of the project and the TRE$_S$PASS workflow. As can be seen from the workflow, the work presented in this deliverable contributes to the visualisation of the results of the TRE$_S$PASS model, the visualisation of the attack trees and the visualisation of the e3 fraud models.

[1] demo environment at `trespass.lustlab.net`

2. History and rationale of requirements selection

This chapter explains the process of requirements selection, the philosophy that underpins the selection and the methods used.

2.1. Participatory design and the requirements selection process

The development of visualisations to articulate the TRESPASS outputs and concepts is founded on the principles of human-centred participatory design. As far as possible, the design decisions that have been taken and the requirements that have been identified have been carried out in conjunction with the target user community (see *Requirements for Visualisation Principles* Table 3.1). The target user community was identified in deliverable (The TRESPASS Project, D5.1.1, 2013) as being information security practitioners where it was recognised that there were many types of information security practitioners ranging from auditors to risk assessors and technical implementors. Whilst humans are understood to be a relevant part of security design and implementation (ISO/IEC 27002:2005, 2005; Sherwood, Clark, & Lynas, 2005), it is seldom that users play an active role in the design of security technologies. Typically the focus of user engagement is on implementation of security technologies rather than on the initial design. We have taken a different position in the development of TRESPASS visualisations because of the recognised importance of the user community and the knowledge that this community of risk assessment processes.

Participation in design takes many forms. Vines et al. (Vines, Clarke, Wright, McCarthy, & Olivier, 2013) describe participation in the following way: “The term ‘participation’ is traditionally used in HCI to describe the involvement of users and stakeholders in design processes, with a pretext of distributing control to participants to shape their technological future." Vines et al. point out that 115 papers in the ACM’s 2012 Computer Human Interaction Conference (CHI 2012) included the term ‘participatory’. Vines et al. claim that, whilst retaining its original philosophy of empowering users to be active decision makers and shapers in the design process, the concept of participatory design has been developed by the Human Computer Interaction community to include a means of designing technologies that enable decision-making processes across an organisation or communities and that stimulate active involvement in creative processes and knowledge sharing activities. The five-stage requirements stage outlined below reveals a number of points at which we

have worked together with practitioners and co-designed the visualisation requirements for TRESPASS. We initially engaged with practitioners when working with the TRESPASS consortium and we then worked with practitioners on several practitioner panels.

Based on this participatory philosophy, a five stage process was followed to develop the TRESPASS visualisation requirements. The stages were as follows:

- Systematic literature and state-of-the-art review.
- Review and validation of proposed visualisation requirements by TRESPASS consortium.
- Production of paper and digital prototypes to represent the operationalisation of the requirements.
- Survey of WP3 outputs (visualisation of analysis results) and the visualisation of model content and input data, in order to further the operationalisation of requirements.
- Review and validation of proposed requirements by practitioner panels.

The systematic literature and state of the art review was published in deliverable (The TRESPASS Project, D4.1.1, 2013). The visualisation requirements have been published in the internal deliverable I4.1.1 and reviewed by TRESPASS consortium members at that point. In addition the visualisation requirements were listed in deliverable 6.2.2. Participatory tools and techniques were developed in order to engage with the practitioner community to elicit their feedback on the requirements. The requirements were refined through discussion with WP3 and a more detailed description of the analysis tools that WP3 will produce. The final set of requirements is iteratively evaluated through the practitioner panels that WP4 has set up.

2.2. Systematic literature and state of the art review

A systematic literature and state of the art review of approaches to security visualisations was undertaken in the first six months of the TRESPASS project. The literature review systematically considered:

- Risk assessment methods literature
- Risk and threat reports
- Environmental risk assessment literature
- Usable security design literature
- Interaction design literature
- Infographics design literature

In order to fully evaluate methods of infographics and interaction design, the following digital and software artefacts were also evaluated:

- Digital mapping projects
- Risk assessment tools
- Digital interaction projects

From this systematic review, the following core requirements were identified:

- Define a visualisation process.
- Develop a set of thinking tools with which to explore new areas of risk.
- Define methods of visualisation evaluation.

In order to further systematise this search, we identified the following sub-field areas in which to conduct the literature and artefact search:

- Visualisation design process
- Visualisation expression
- Visualisation interaction

In the literature review it was identified that in order to articulate the information security risk, the visualisations must be capable of expressing the complexity of information security risk, the spatial and temporal location of information security risk and the interaction between the different elements that constitute an information security risk.

The following requirements were derived from the literature review process and published in the deliverable (The TRESPASS Project, D4.1.1, 2013):

- **R4.1** - The visualisation must have a particular goal
  - **R4.1.a** - The visualisation must be created and implemented using tools that are appropriate for the communication of the intended narrative
  - **R4.1.b** - The visualisation design process must address the temporal, complexity, spatial, and interaction properties of the narrative
  - **R4.1.c** - The visualisation tools must be capable of expressing the temporal, complexity, spatial, and interaction properties of the narrative
- **R4.2** - The visualisation must be usable and accessible for the intended audience.
  - **R4.2.a** - The audience for the visualisation must be identified as part of the visualisation design
  - **R4.2.b** - The visualisation tools must be appropriate for the visual literacy of the intended user community
  - **R4.2.c** - The operational constraints under which the visualisation must operate (for example the technical constraints and the cognitive processing constraints) must be identified as part of visualisation design
  - **R4.2.d** - The visualisation tools must be appropriate for the operational constraints under which a visualisation is generated and used

- **R4.3** - Tools and methods for adjusting images to achieve different types of expressivity.
  - **R4.3.a** - The visualisation tools must be able to express the relevant strata at work within the narrative
  - **R4.3.b** - The visualisation tools must be able to adjust the visualisation to present the visualisation for different strata

These requirements reflect the importance of:

- Designing with and designing for particular user communities
- Direct engagement with the target user communities and the tool designers to develop and adjust visualisations
- Adjustment of visualisation to reflect different levels of expressivity and to communicate different aspects of the narrative

3. Final visualisation requirements set

WP4 took part in the requirements task-force activity to consolidate the core visualisation requirements and to re-visit commitments to other work packages. As part of this process the abstract requirements published in deliverable (The TRESPASS Project, D4.1.1, 2013) were made more concrete by grounding the risks in risk scenarios relevant to the target user community.

3.1. TRESPASS consortium review and validation of requirements

In the task-force work, the three core WP4 requirements were confirmed as:

- Define a visualisation process.
- Develop a set of thinking tools with which to explore new areas of risk.
- Define methods of visualisation evaluation.

As part of the task-force work, the following clarifications were added to the requirements:

Table 3.1.: What WP4 requires from other WPs

#	Requirement	Source	Target	Goals	Refer to
R12	Develop a visual language	WP4	WP4,6	Needed for development of interface	See 4.1.1
R13	Define a visualisation process	WP4	WP4,5,6	Needed for development of interface	See 2.1
R14	Develop visual thinking tools	WP4	WP2,4,5	Needed to produce meaningful analysis results	See 4.1
R15	Define methods of visualisation evaluation	WP4	WP2,4	Needed to ensure robust tool	See 5.3
R49	End user visualisation requirements	WP4	WP7	A prioritised list of visualisation requirements from the case studies would be useful additional input during Year 3 as WP4 concentrates on end-user requirements.	See 5.2.1 and App. A

Table 3.2.: What other WPs require from WP4

#	Requirement	Source	Target	Goals	Example
R09	Classification of data types	WP2	WP4	Needed to produce visualisation toolkit	Figure 4.19
R10	Interface between analysis tools and visualisation tools	WP6	WP4,6	Needed to generate visualisations	Figure 4.25
R12	Develop a visual language	WP4	WP4,6	Needed for development of interface	Sect. 4.1.1
R13	Define a visualisation process	WP4	WP4,5,6	Needed for development of interface	Sect. 2.1
R14	Develop visual thinking tools	WP4	WP2,4,5	Needed to produce meaningful analysis results	Sect. 4.1
R15	Define methods of visualisation evaluation	WP4	WP2,4	Needed to ensure robust tool	Sect. 5.3
R33	The tool needs to provide adequate visualisation for technical correlations within different misuse scenarios	WP7	WP4	Helps discussion with practitioners. Task 7.3 depends on this - from a WP4 perspective this depends on data coming through from WP2 and the analysis engines provided by WP3	Figure 4.14
R34	The tool needs to provide adequate visualisation for risks related to features of the involved products and services of the telecommunications company	WP7	WP4	Needed for case studies. From a WP4 perspective this depends on the data from WP2 and the analysis from WP3.	Figure 4.9
R35	The tool needs to provide adequate visualisation for evaluation of risks	WP7	WP4	Needed for case studies. From a WP4 perspective, this depends on the data coming from WP2 and the analysis from WP3.	Figure 4.15
R40	Support for attack tree visualisation	WP3	WP4,6	Improved presentation of tool output	Figure A.7
R42	Data analysis model relevant to case studies	WP3	WP4	Require capabilities to sufficiently analyse data for case study (WP7) purposes	Figure 4.10
R81	Visualisation of social and technical data, maps, scenarios, countermeasures.	MT	WP4,6	“Derived from P7 (TRESPASS tools should be able to visualise maps, scenarios, and countermeasures). Supports Use-case 1 (Security Investment), Use-case 2 (Audit), Use-case 4 (Product-service system) and Use-case 5 (Quick scan), specifically U1.2, U2.6, U2.13, U4.6, U5.9 and U5.15.”	Figure 4.21

Table 3.2.: What other WPs require from WP4

#	Requirement	Source	Target	Goals	Example
R86	Users can select base models from a Model Template Library.	MT	WP4,5,6	Supports Use-case 4 (Product-service system) and Use-case 5 (Quick scan), specifically U4.2, U5.4	Figure 4.18

4. Operationalising requirements: progress report

The visualisation requirements can be placed into three groups, each operationalised and validated with a a group of techniques :

- Requirements for visualisation principles
- Requirements for risk visualisations
- End-user requirements

The requirements in the first group, requirements for visualisation principles, are enacted in all aspects of the visualisations in TRESPASS and are evaluated through the use of digital and paper prototypes. The requirements in the second group, requirements for the visualisations of risks, are developed in conjunction with the tool makers, particularly those who form part of TRESPASS WP3. The requirements in the final group, enduser requirements, are developed in conjunction with our user panels and are evaluated through the use of digital and paper prototypes. These requirements gathering and evaluation processes are outlined in the sections below.

The following figures provide examples of how the different requirements have been applied. In the legend for each figure, the requirement number is given so that the specification of the requirement can be looked up in Tables 3.2 and 3.3.

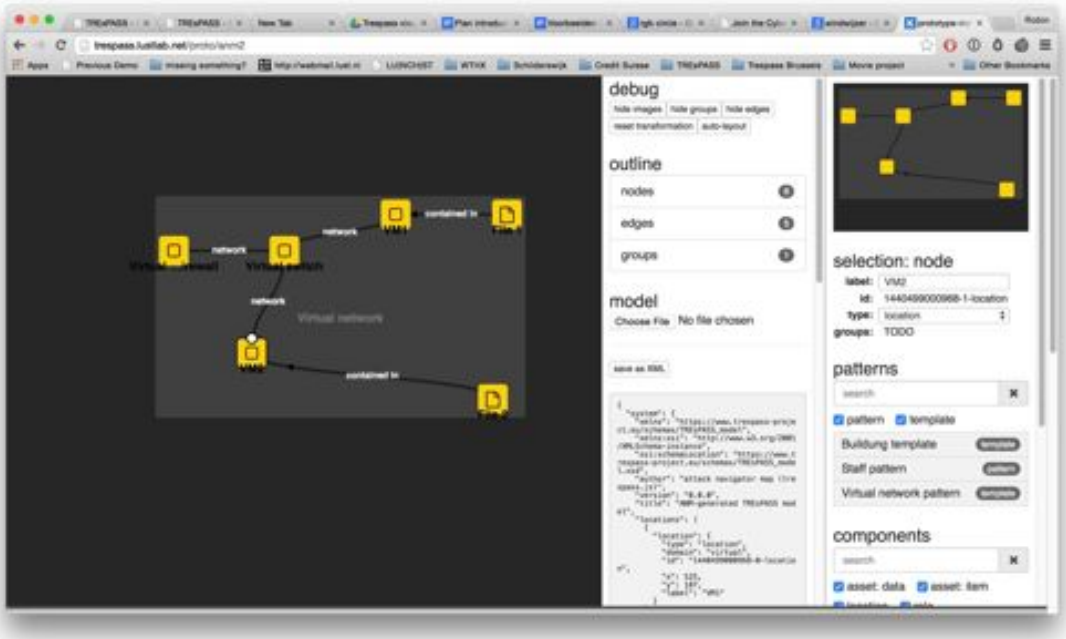

Figure 4.1.: **ANM**, screenshot showing how users can drag and drop standard elements onto the map.

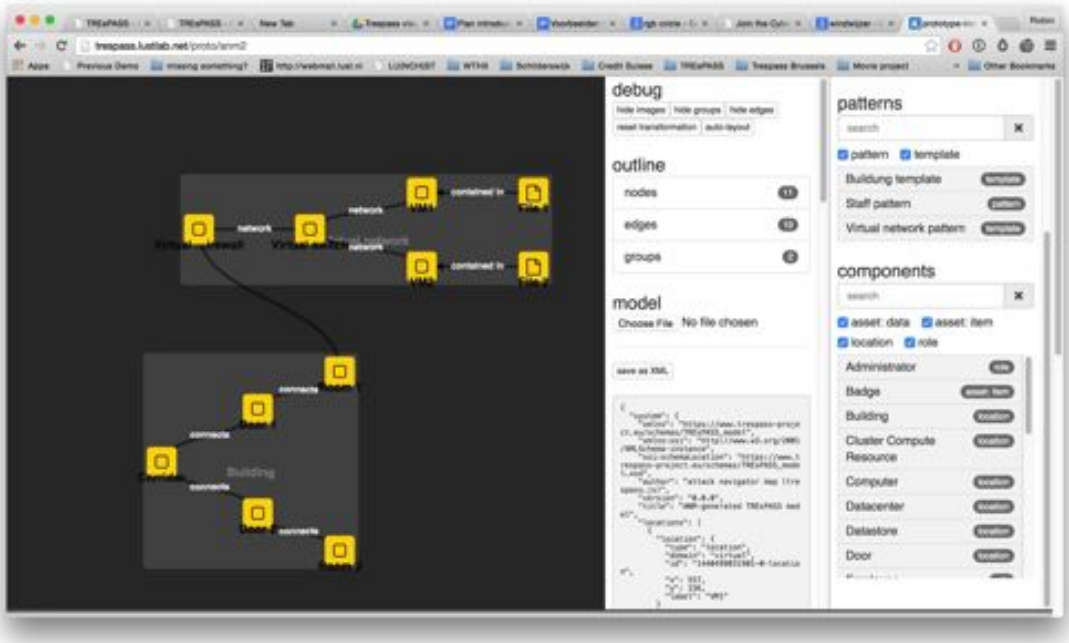

Figure 4.2.: **ANM**, screenshot showing how users can add connections between elements on the map.

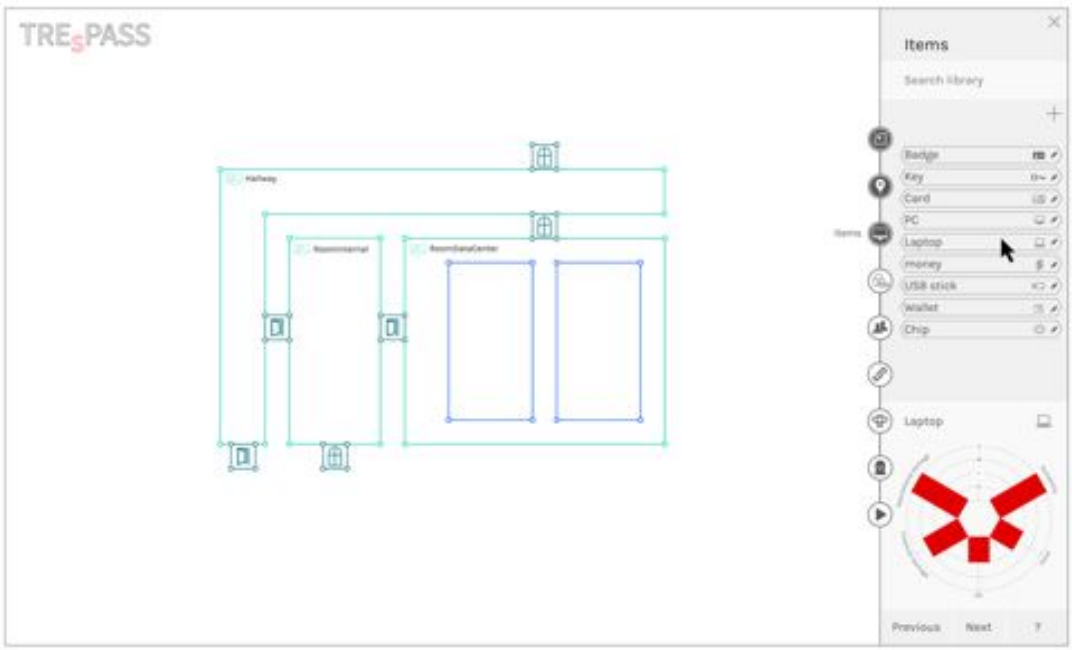

Figure 4.3.: Using the ‘Data’ overview and editing side-pane, the user selects Laptop from the list of ‘Items’ (assets).

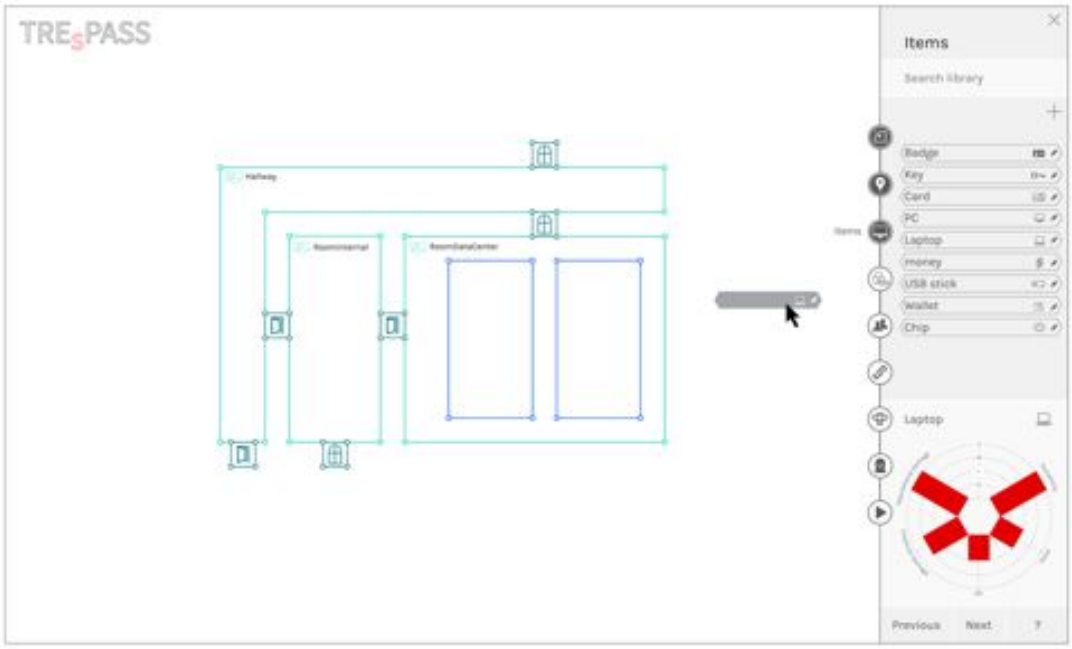

Figure 4.4.: The user drags the ‘Laptop’ to the relevant part of the map.

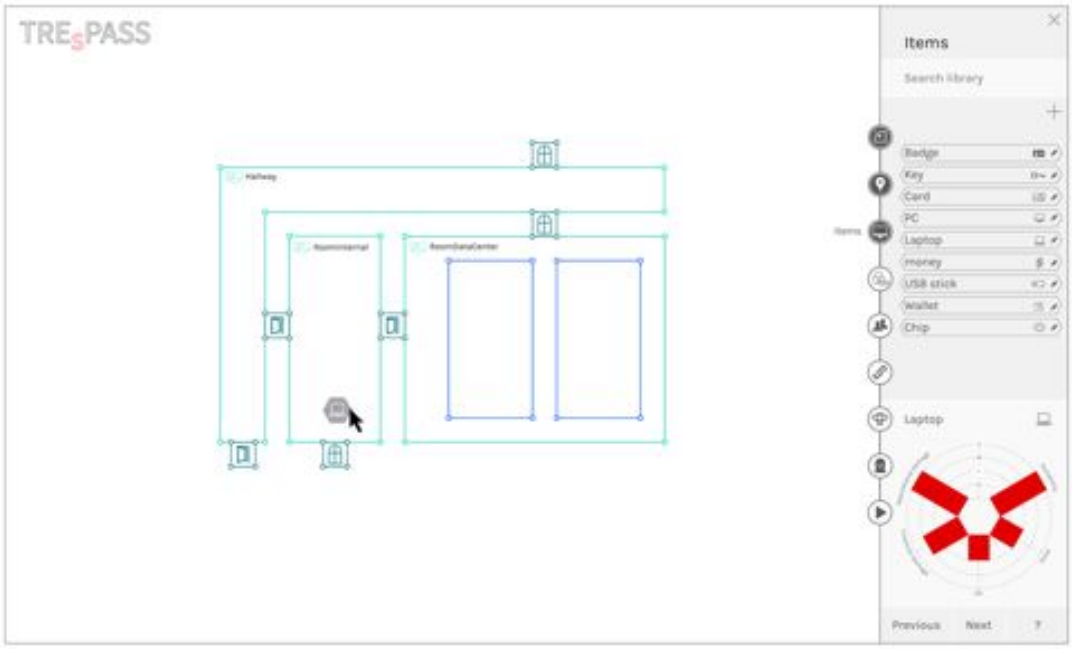

Figure 4.5.: The user drops the ‘Laptop’ into the relevant part of the map.

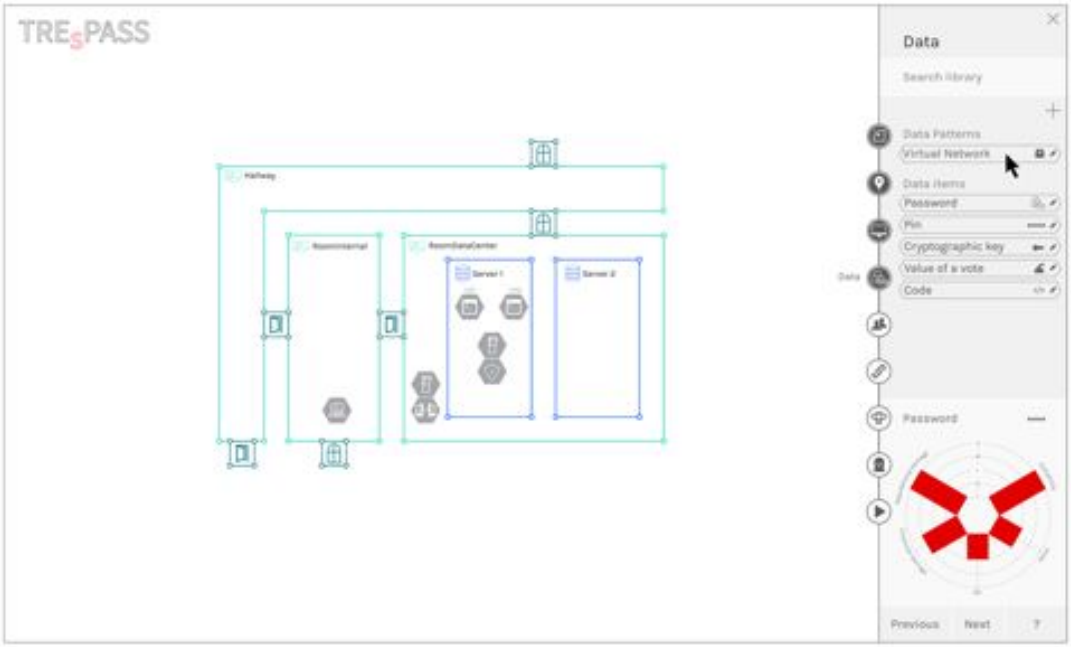

Figure 4.6.: The user selects 'Data Patterns' > 'Virtual Network', and the relevant already existing 'Items' (assets) appear on the map, or can be added to the map by dragging (see next image). Note the inclusion of design features that can be adapted to many different types of scenario, including virtual networks and other data exchange patterns, including those relevant as features that are involved in the telecommunications case study (R 34). See Table 3.2, *What other WPs require from WP4*.

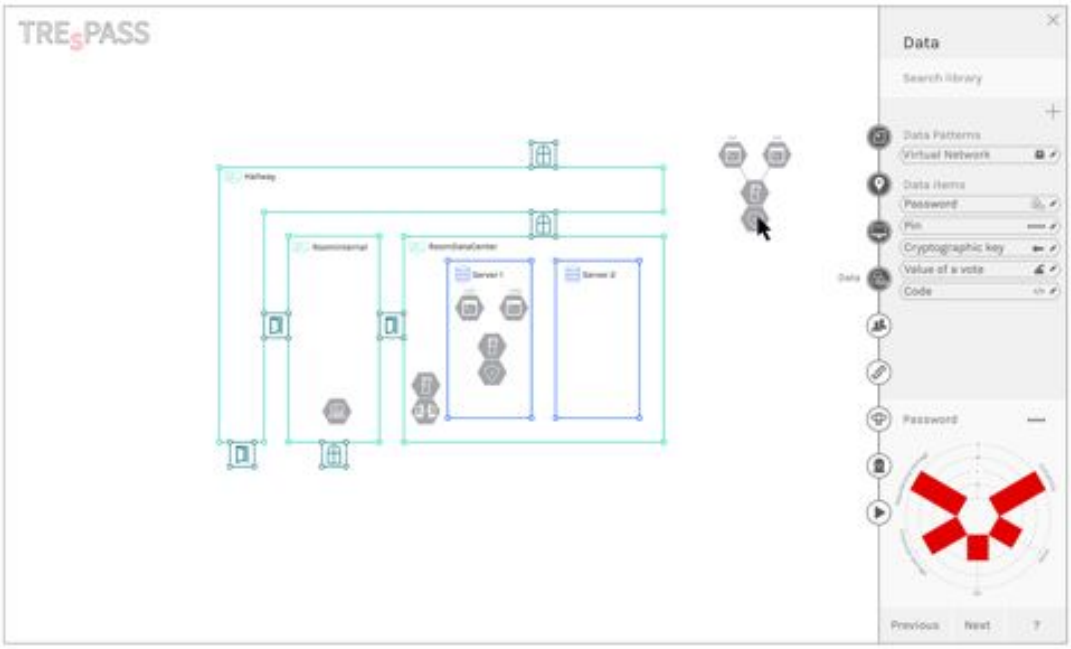

Figure 4.7.: The user drags ‘Virtual Network’ pattern onto the relevant part of the map, adding to those previously added.

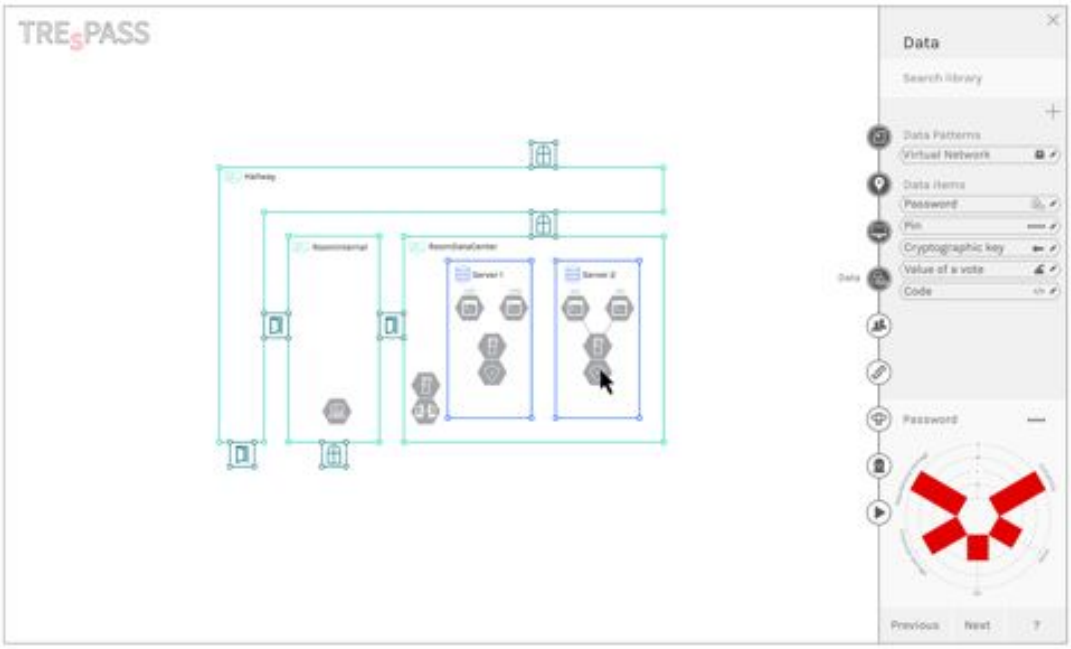

Figure 4.8.: The user drops ‘Virtual Network’ pattern onto the relevant part of the map, adding to those previously mapped. Once dropped this network can be customised by the user.

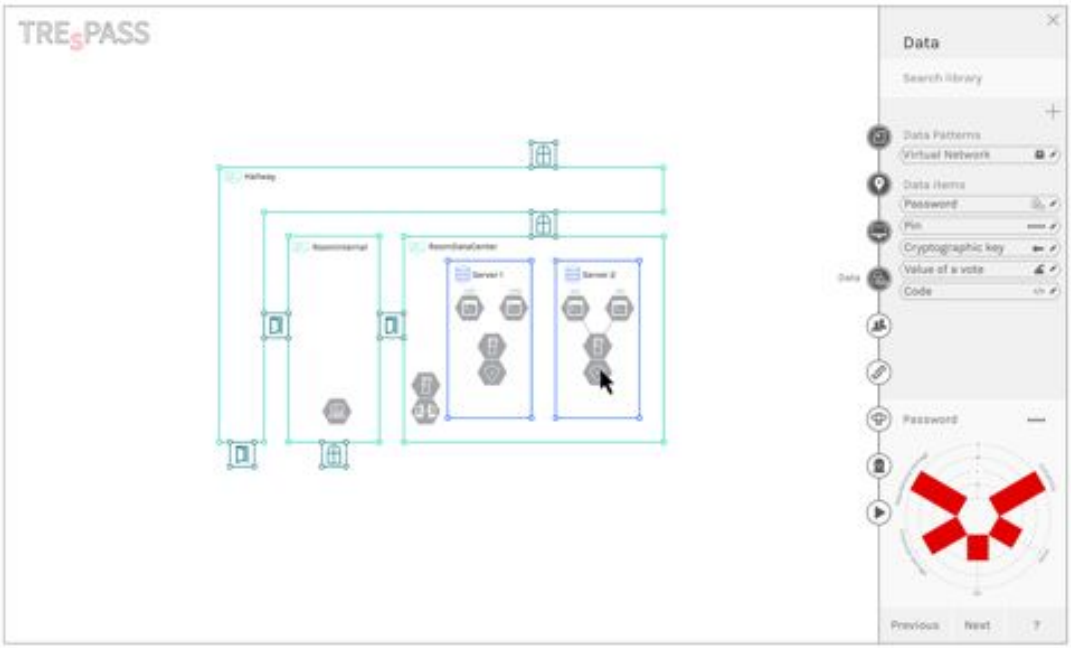

Figure 4.9.: The user drops 'Virtual Network' pattern onto the relevant part of the map, adding to those previously mapped. Once dropped this network can be customised by the user.

Figure 4.10.: Key to colour codes: this shows the narrowing of the search for a solution to not constantly show all connections. All the items on the map will have a selection of offset-paths, which indicate what kind of connections belong to this actor or item. The types of connections are colour-coded. Note that this presents data to the analyst user in a way that can be applied to all case studies (R 42). See Table 3.2, *What other WPs require from WP4*.

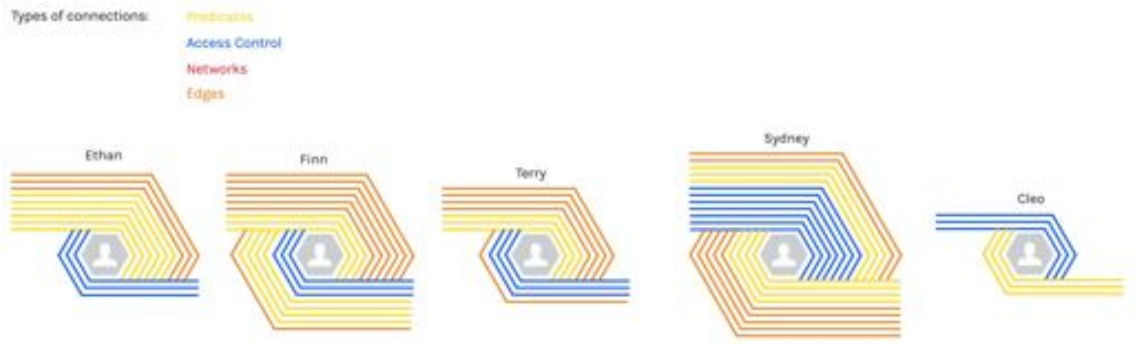

Figure 4.11.: **ANM**. On this slide all the 'Actors' are aligned. By showing the different types of connections and the amount of incoming and outgoing connections, the user immediately gets an overview of which actors might be more vulnerable than others. For example, the user just has to look for the actor with the most blue paths to find out who has the most access to other location and objects.

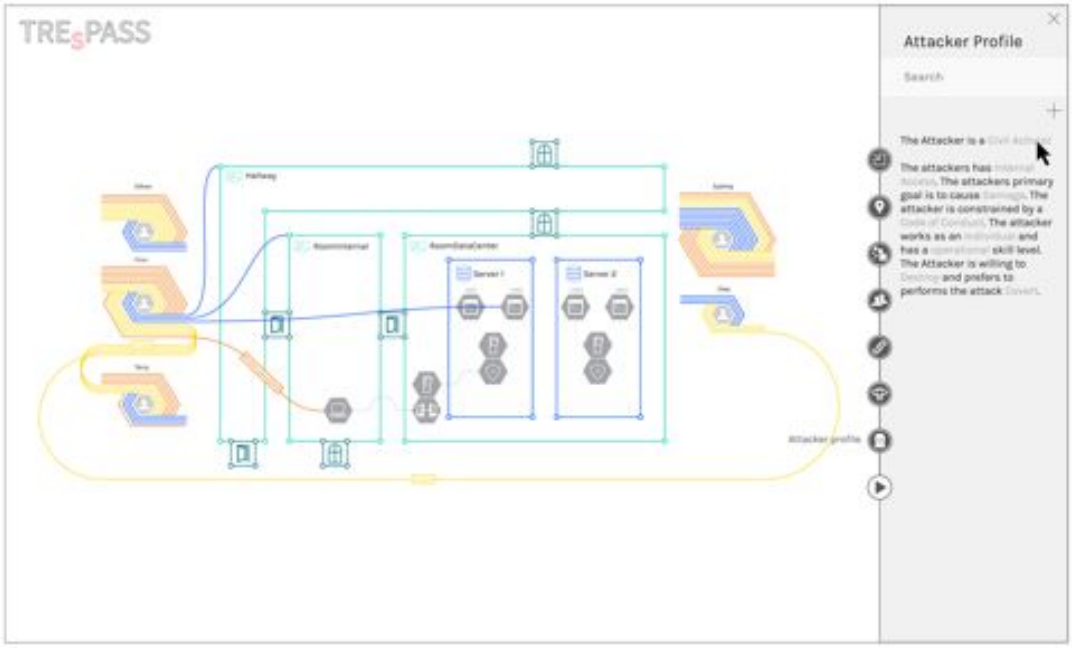

Figure 4.12.: The 'Actors' are surrounding the map. And the connections between these actors and their linked Items (assets) are only shown when items or actors are hovered-over by the user with their mouse. The actor 'Finn' (In the middle at left) is hovered-over, which results in the connections belonging to Finn to becoming visible, showing the connections with other objects on the ANM. The user now changes the side-panel to address 'Attacker Profiles'. The user is able to read a text describing the attributes of an attacker persona. Certain points in this description are highlighted and clickable, giving the user the option of interacting with and editing this description of the attacker. See Table 3.2, *What other WPs require from WP4*.

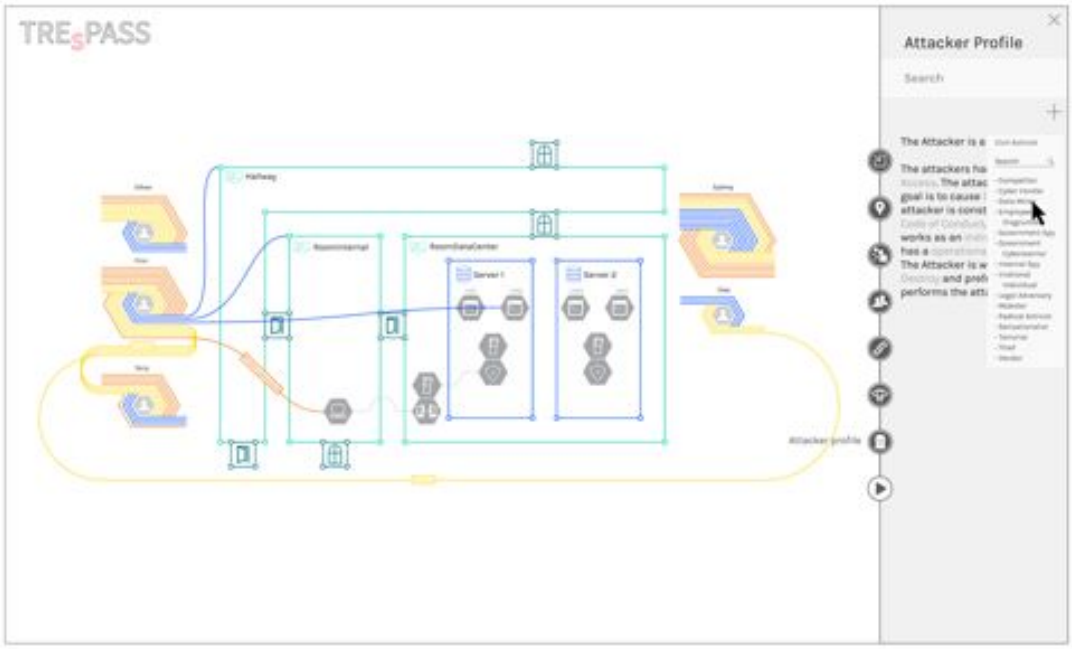

Figure 4.13.: The user is able to click on the name of the attacker and adjust to any of a drop-down list of alternatives that are available.

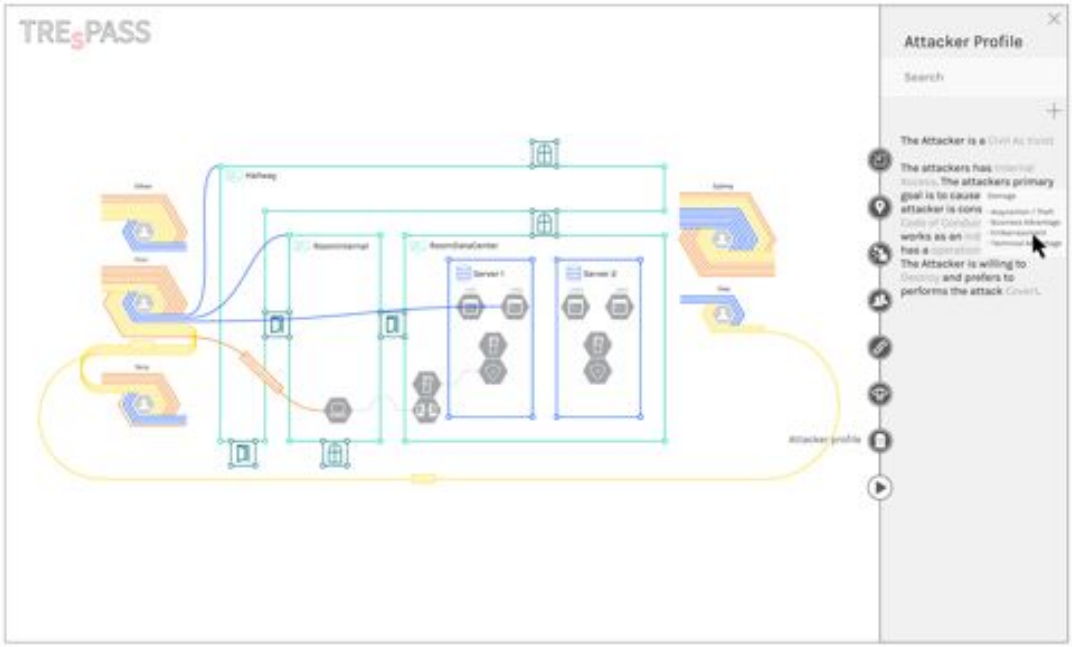

Figure 4.14.: The user is able to click on the property of the attacker that they wish to edit, in this case the attackers' motivation to cause Damage can be refined to that of 'Business Advantage' or 'Embarrassment' for example. Note the ability to correlate one of a number of potential misuse scenarios with the technical and organisational infrastructure (R 33). See Table 3.2, *What other WPs require from WP4*.

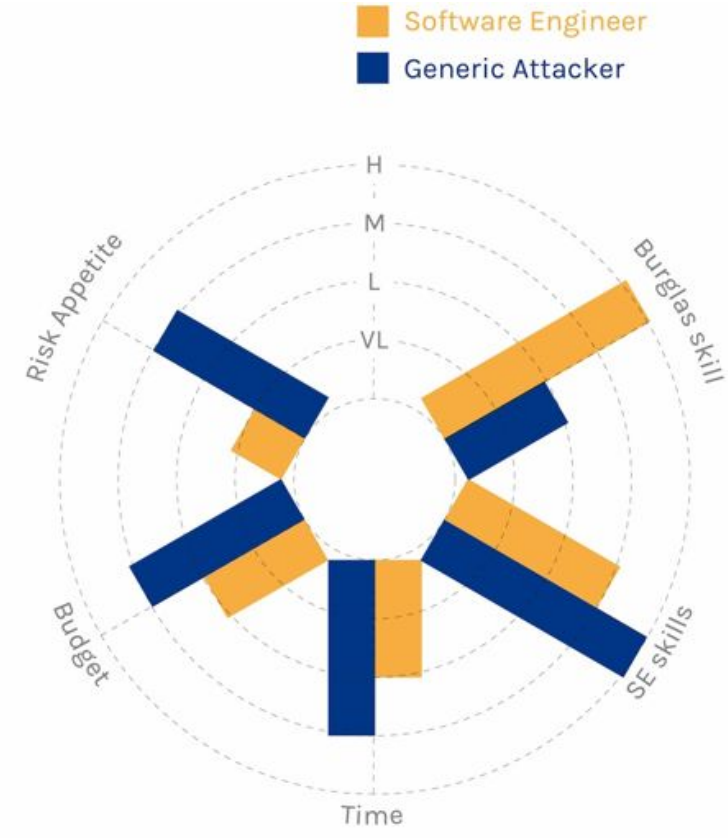

Figure 4.15.: This sample visualisation from a recent prototype shows how attacker profiles can be used to differentiate different categories of attack types graphically. Shows the new graphing approach to visualising an attackers profile, and its ability to show multiple attacker profiles at the same time, which creates the possibility to compare attacker profiles and types of attackers. Note the provision of a visualisation/tool for the analysis of different types of risk, which is here based upon different attacker profiles (R35). See Table 3.2, *What other WPs require from WP4*.

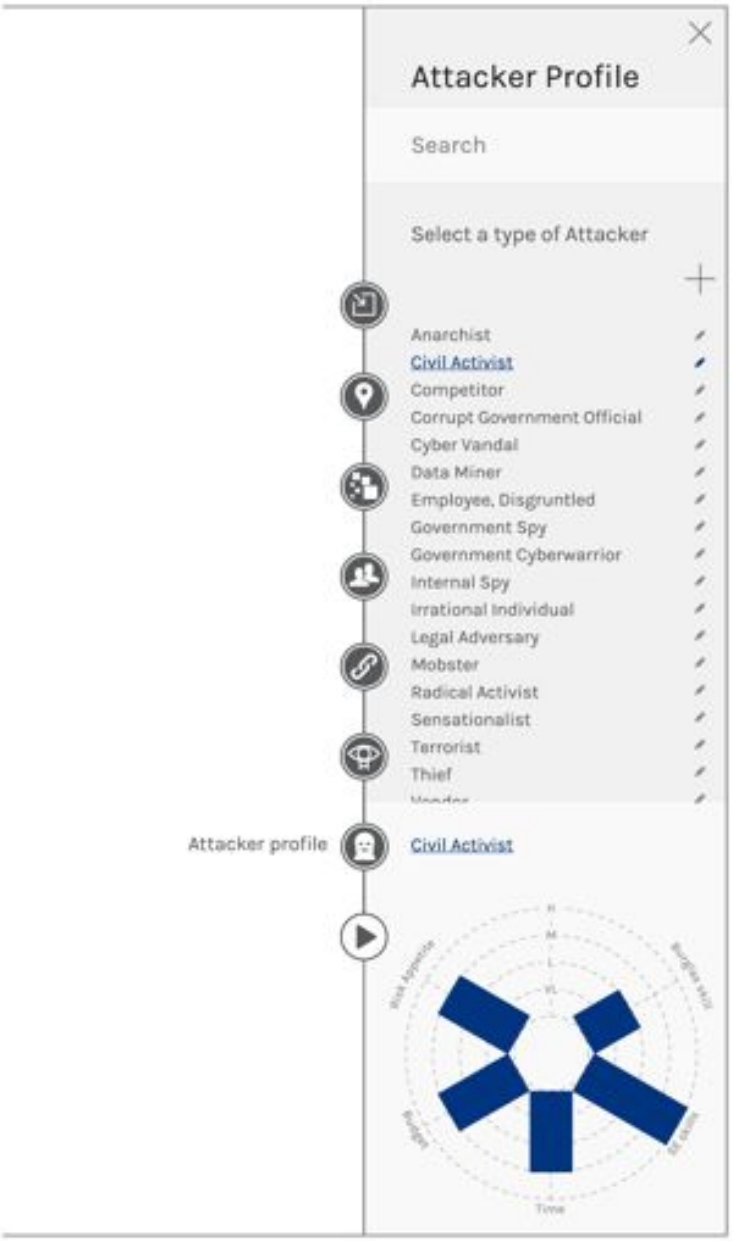

Figure 4.16.: **ANM**, shows how the new wizard and attacker profiles can be embedded into the ANM interface. This also shows in more detail how the design of the attacker profile wizard has been changed in response to practitioner feedback comments, which had suggested that the previously used 'radar graph' was misleading, in that it implies and indeed represents a continuous surface that exists between different parameters, which may or may not in fact exist. (This slide shows a previous Library approach for the attacker profiles, which has since been replaced by a text-based approach, where the user is able to click on the property of the attacker that they wish to edit).

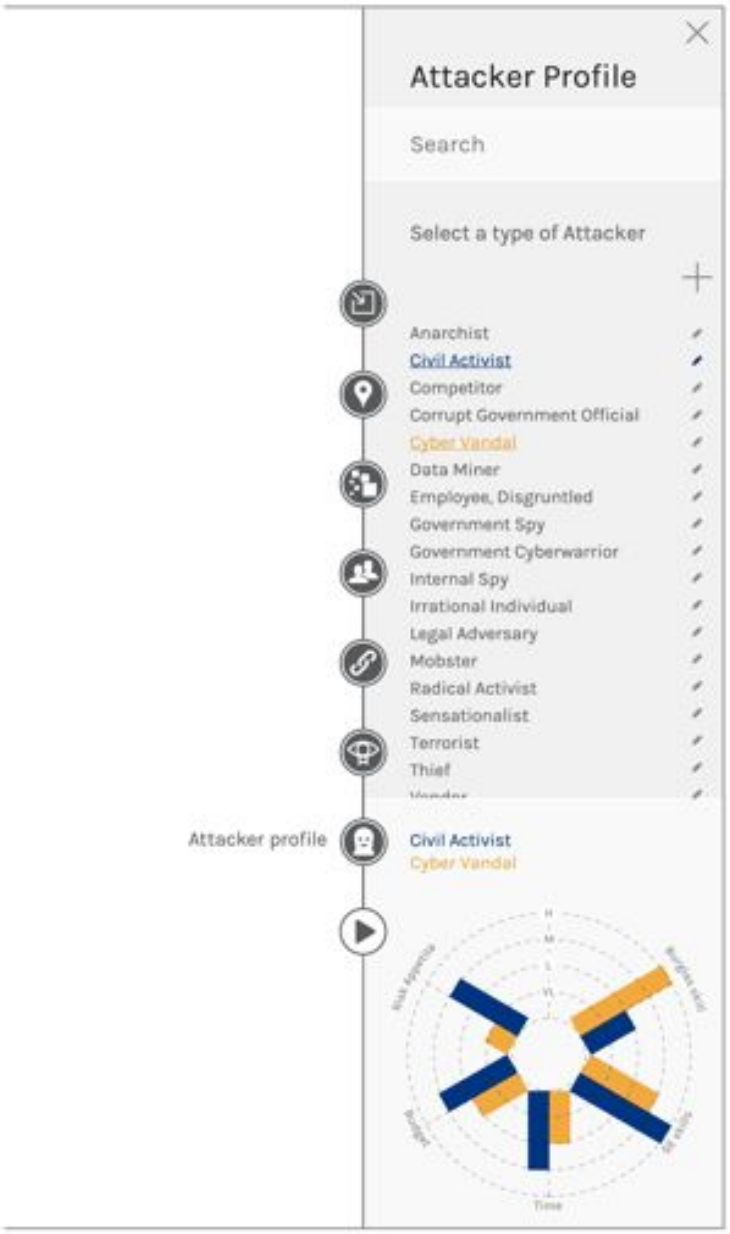

Figure 4.17.: **ANM**, shows how the new wizard and attacker profiles can be embedded into the ANM interface, and how entities may be selected and duplicated to be included in the model. (This slide shows a previous Library approach for the attacker profiles, which has since been replaced by a text-based approach, where the user is able to click on the property of the attacker that they wish to edit).

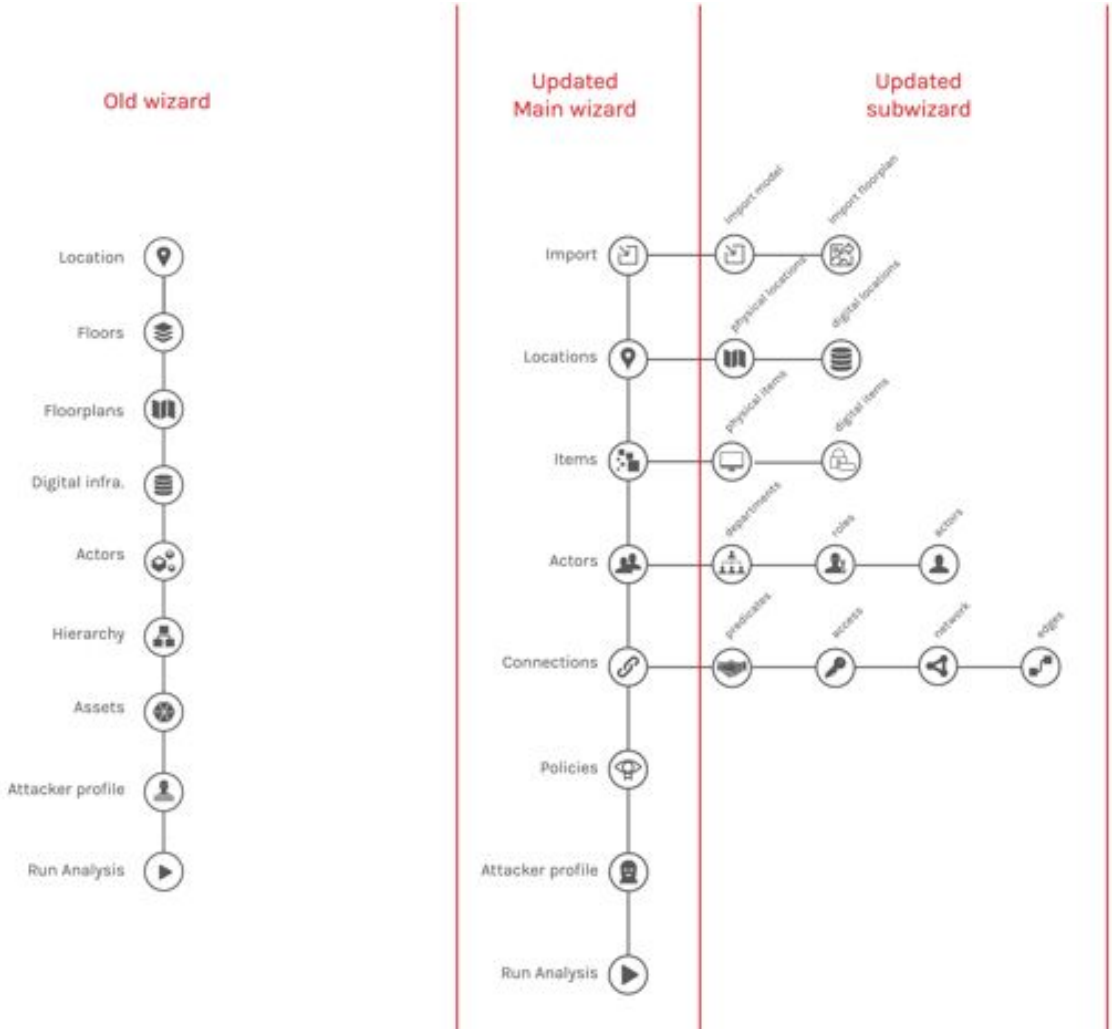

Figure 4.18.: Updates to the **ANM**. On the left is shown the previous version, including Wizard and sub-Wizards (light grey). Note the 'Import' models button (R86). See Table 3.2, *What other WPs require from WP4*.

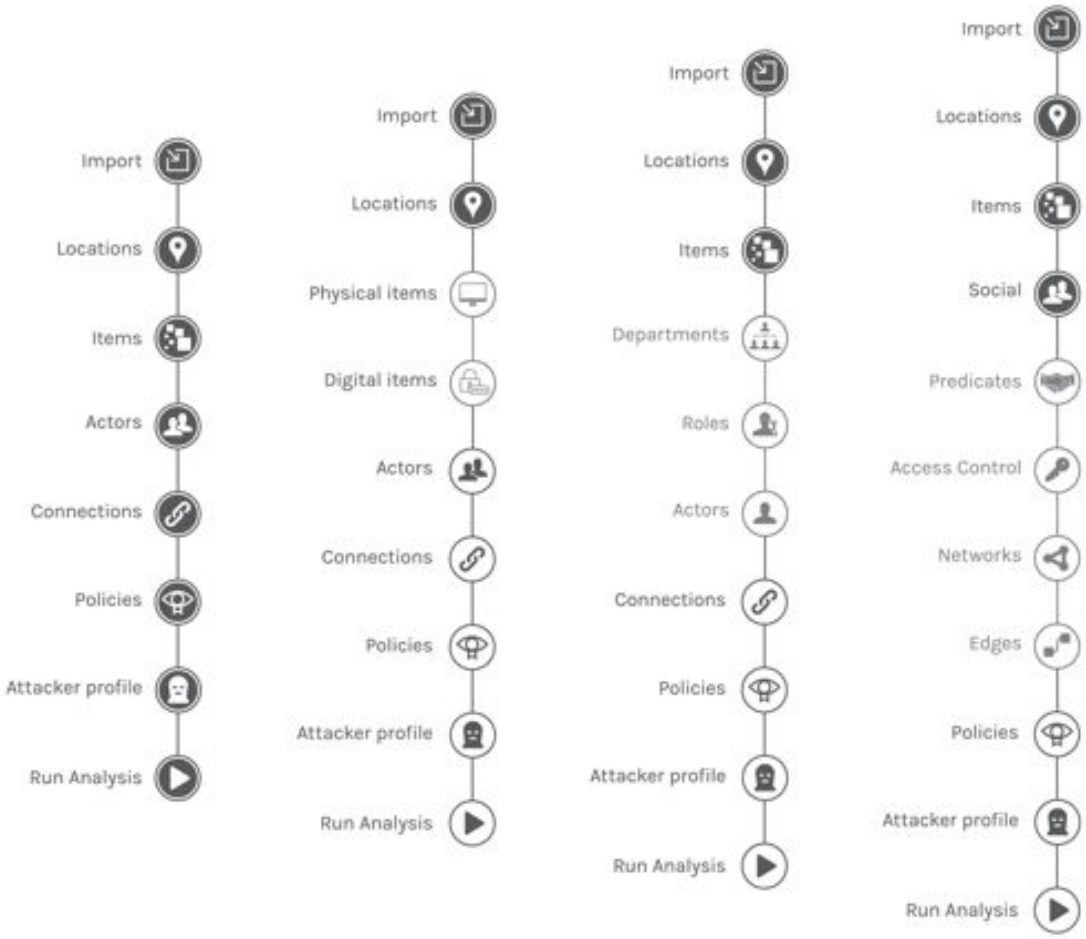

Figure 4.19.: **ANM**, showing the unexpanded side-bar (at left), and each successive expansion, 'Locations', 'Items', 'Social'. Note that the classification of different data types is afforded by the menu options that are available to the user here (R10). See Table 3.1, and *What WP4 requires from other WPs*, Table 3.2, *What other WPs require from WP4*.

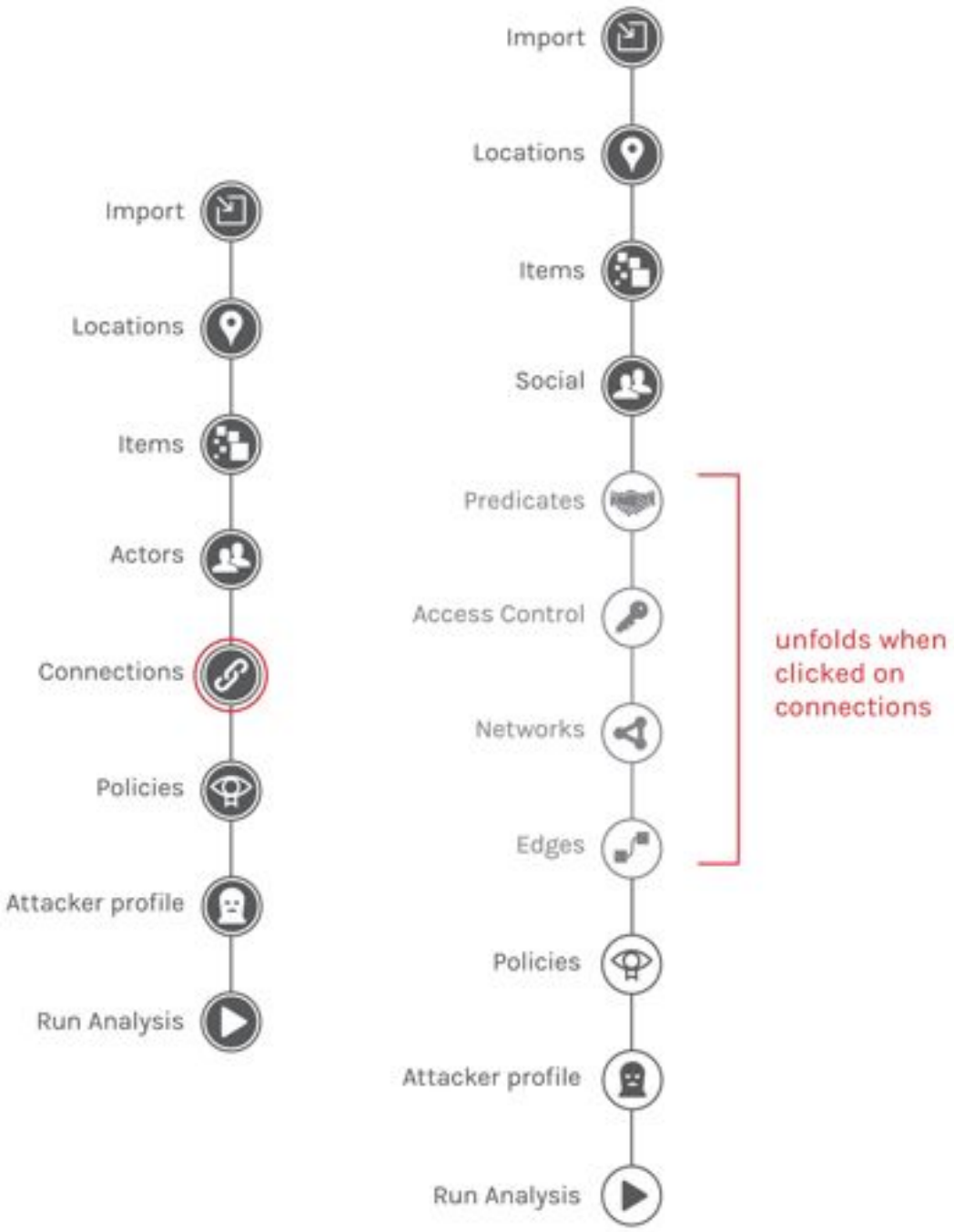

Figure 4.20.: **ANM**, showing the way that the side-bar expands and collapses at command. When one clicks on the Items button, this one button will split into 'physical items' and 'Digital Items'. The same is true for clicking 'Actors' and 'Connections'.

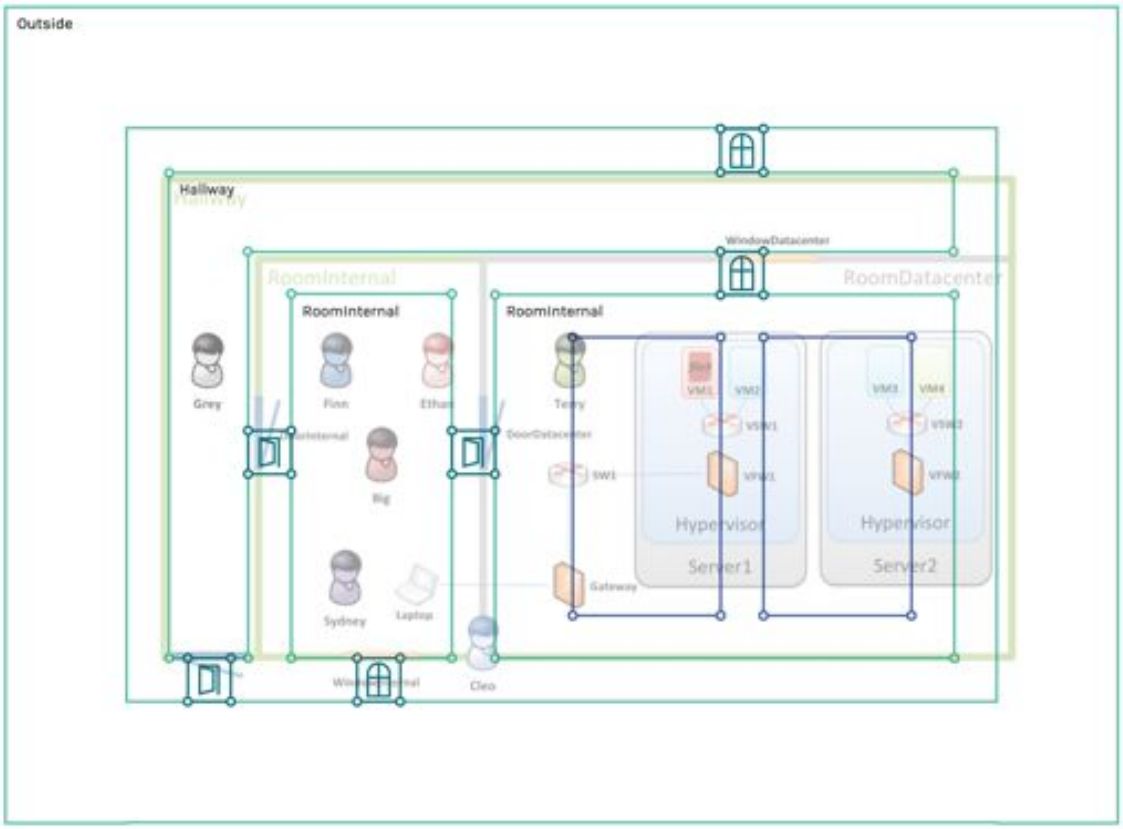

Figure 4.21.: **ANM**, showing the way that the the user can import a floor plan to work from, for example, and can trace over it, using the import to guide the construction of a new or existing model with the tools of the ANM. Locations have anchor-points that can be dragged around to match the locational object with the physical spaces on the floor plan. As well as using floor plans and technical diagrams, the user can also import photographs of rich-pictures models, including LEGO co-design work for the 'stage-zero' phase of risk assessment and model development. Note that the visualisation of social and technical data, maps, scenarios, countermeasures is afforded by the inclusion of this facility (R81). See Table 3.2, *What other WPs require from WP4*.

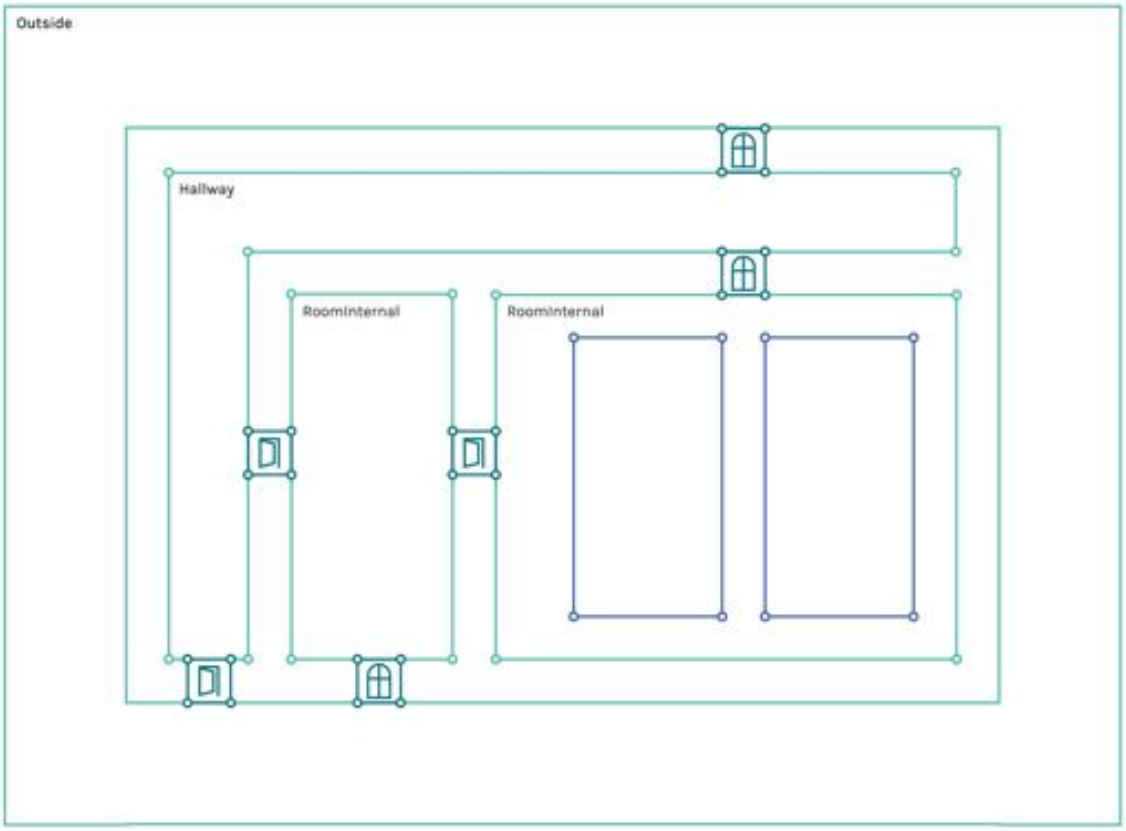

Figure 4.22.: **ANM**, shows the result of this tracing actions from the previous Figure. Now the floor plan in the background is removed. Doors and windows are seen as physical locations that connect one space with one other. Outside is also considered to be a location.

4.1. Survey of WP3 outputs

As a refinement of the task force process, further discussion was sought with the tool-makers in WP3. The resulting visual requirements apply to both the visualisation of the attack navigator map and to the outputs of the analysis tools developed in WP3 (see *Requirements for Visualisation Principles* Table 3.1). The meta requirements of the capability to provide a visual thinking tool, a visual methods of evaluation and technical correlation were initially tested with the paper prototyping. The detailed requirements of the visualisation of the outputs of the different analysis tools could only be developed from close collaboration with the tool makers. A survey was therefore conducted within WP3 of all the analysis tools that are to be made and the following results came back:

Tool Partner & contact	Description	Input	Output	Status & Comments
treemaker DTU Christian Probst	Evaluates TRE$_S$PASS model files and creates attack trees to be used for analyses, visualisation, and for the TRE$_S$PASS process.	Model file in TRE$_S$PASS model file format (XML)	Attack trees in ADTree XML format	available for IPTV case, cloud case under development
AttackTreeEvaluator DTU Zara Aslanyan	The tool evaluates attack trees with multiple parameters based on Pareto efficient solutions. It computes the maximum probability of success of an attack, and the Pareto set of pairs with maximum probability and minimum cost. Uses two methods for evaluating attack-defence trees: a topdown semantic evaluation (more intuitive), and a bottom-up algorithmic evaluation (lower complexity)	*binary* attack trees in ADTree format (XML)	• maximum success probability in text format • set of Pareto efficient solutions with corresponding assignment (set of basic actions) for the main attack in text format	available; extension to non-binary trees under development

ADTool UL Olga Gadyatskaya	The Attack-Defense Tree Tool (ADTool) allows users to model and display attack-defense scenarios, through the use of attack-defense trees (ADTrees) or an alternative term-based representation of ADTrees called attack-defense terms (ADTerms). ADTool allows to perform quantitative analyses on ADTrees/ADTerms. This means that a user is able to answer questions such as: What are the costs of an attack, what is the minimal skill level required for the attacker, how long does it take to implement all necessary defenses or who is the winner of the considered attack-defense scenario, and many others.	• Input and editing of ADTrees and ADTerms can be done by using the user graphic interface provided with the tool (See a picture attached). • Starting from Version 1.2, ADTool supports exporting and importing ADTrees written in the XML form (See an example attached)	A user can export ADTrees to image, LaTeXfiles, and XML files.	• Available (but not actively maintained). Extensions planned for handling sequential AND/ORs. • Analysis does not output attack traces (yet).
ATCalc UT Rajesh Kumar	ATCalc is a tool for efficient Attack Tree Analysis using priced timed automata models. It computes the system unreliability for each mission time, i.e. the probability that the system fails within the mission time. Further it is capable of computing the mean time to failure (MTTF), i.e. the expected time that the system will fail. ATCalc uses CADP, a proprietary software owned by Inria.	Hierarchical information about gates and basic steps (basic steps decorated with CDF and P-success)	• graph showing evolution of attack(P-goal vs. time), text data • can potentially output a set of attack traces, if you sample the output at specific values for P-goal or time	available

Attack Tree Analyzer CYB Aleksandr Lenin	Assess “attractiveness” as a measure for security for given attack trees. The attack tree computation tool can be used to calculate optimal attack vector (from the attacker point of view) taking attacker profile into account. The tool is capable of analysing if the system is secure against rational profit-oriented adversaries, as well as can derive the most profitable attack vector in the system. It can perform calculations according to Failure-Free Model and JW Parallel Model (former ApproxTree+) and supports ‘slow’ precise algorithms as well as ‘fast’ imprecise stochastic algorithms.	Attack Tree in XML format (ADTool compatible)	• Attack traces with associated attractiveness number (simple textual format) “or” • ADTool compatible tree parts (XML) with associated attractiveness number ML file containing the most profitable attack path (if any), the value of the expected adversarial profit and optional comments. Output varies depending on the outcome. If profitable attack vectors were found, output is represented as a subtree with utility annotation on the root node. If no profitable attack vectors were found, text can be found in the <comment> element.	• Successor of ApproxTree++ (including its functionality). • Can output an arbitrary number of attacks, not only the most attractive one. • Available in iTrust platform.
UPPAAL SMC AAU Rene Rydhof				-no input received-

e3Fraud UT Dan Ionita				Specific for Telco case study

Table 4.1.: Overview of WP3 Analysis Tools

4.1.1. Applying the visualisation requirements to meet WP3 visualisation needs

All of these analysis methods produce a set of attack traces (sub-trees + annotations) as an output and this commonality will be used when developing the visualisations. The subtrees could be shown on their own (potentially overlaid, producing a sort of heat map), or highlighted in the original attack tree. Attack traces annotated with values could also be visualised in radar / spider web graphs. Paper and digital prototypes will be used to evaluate which visualisations are most meaningful and accessible for particular user communities.

One possible envisioning of the Attack Navigator Map is a connected split-screen results view (see Fig. 4.24), where custom visualisations, sub-trees, and a tabular ranking of the attacks are visible at the same time; and these are also sortable and filterable throughout. Another view could show the attack navigator map, and highlight relevant parts of the model therein (see Fig. 4.25).

The image below (Fig. 4.23) shows the central problem that is addressed by the present interface design, that of over-complexity in simultaneous graphic representation. Inset at top-right is a simple floor plan representation of the cloud-based Data Centre scenario, and to its side is a dense map of the same scenario which is presented in full, without aid of simplifying devices that enable the map to be read in instalments. By collapsing wizard menus and by allowing the user to hover of actors and assets of interest, the relevant connections to different parts of the entire map are highlighted and brought to attention visibly.

4.1.2. Placing these visualisations within the Attack Navigator Map

In this section are presented a sequence of images from the visualisation prototyping that should be cross-referenced to the Tables at the beginning of this Chapter, relating to *What WP4 requires from other WPs* 3.1, *What other WPs require from WP4* Table 3.2. Discussion of the prototyping and requirements procedures is given in a separate place, Section 5.

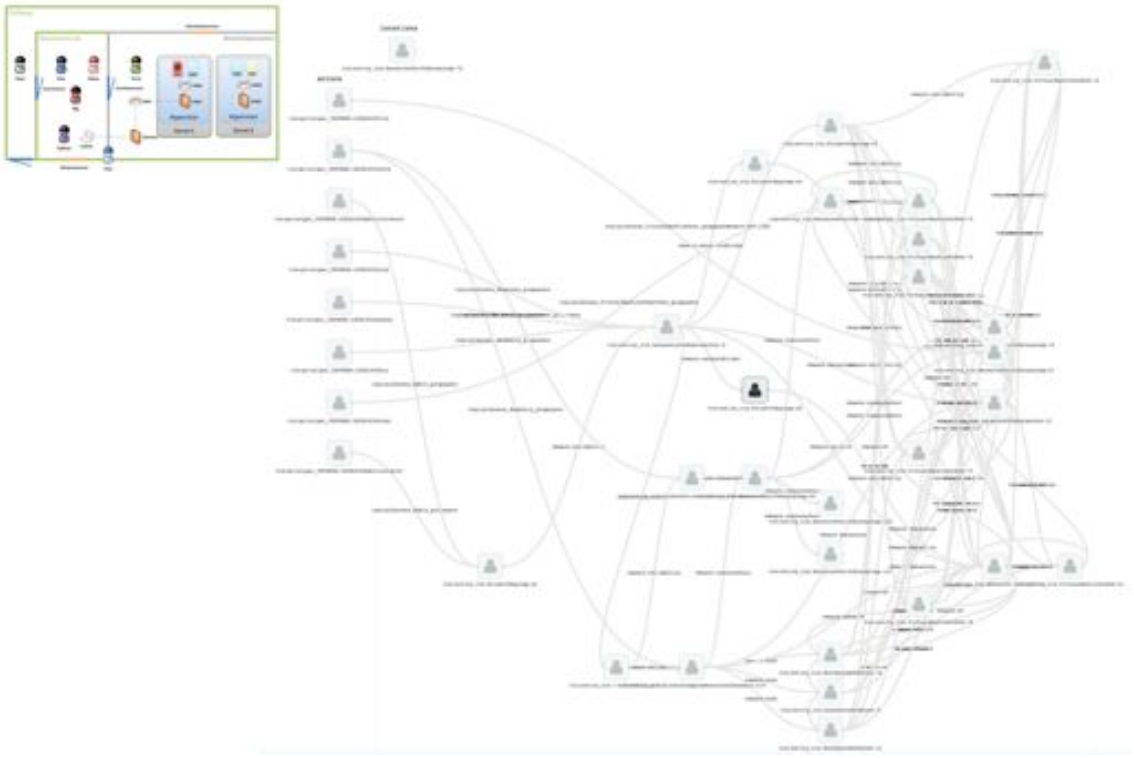

Figure 4.23.: The high number of connections in this image make it hard to read this mapping of infrastructure and actor relationships in the model. This shows the results of attempting to simultaneously show all connections, as modelled in a graph editor such as the present ANM. Even for this very basic Data Centre scenario. We try to solve this problem in the current design of the **ANM**.

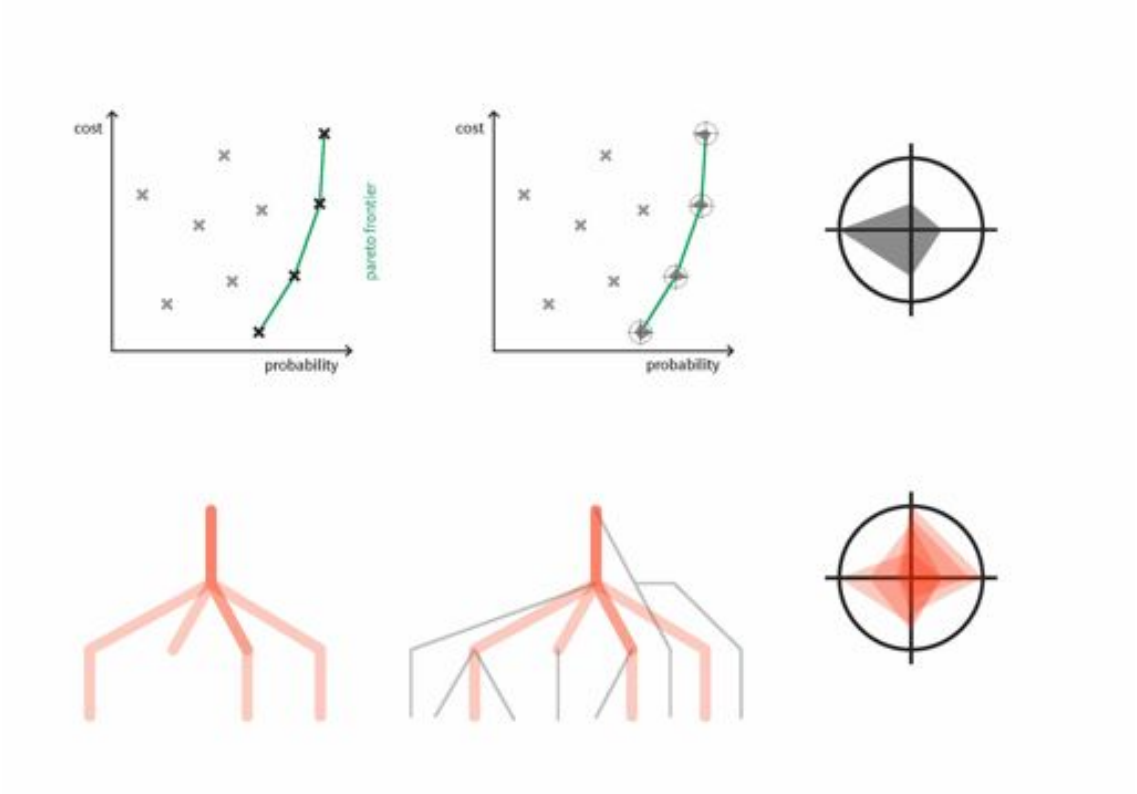

Figure 4.24.: ANM: Analysis results, prototype sketch of data views in separate but interrelated panels.

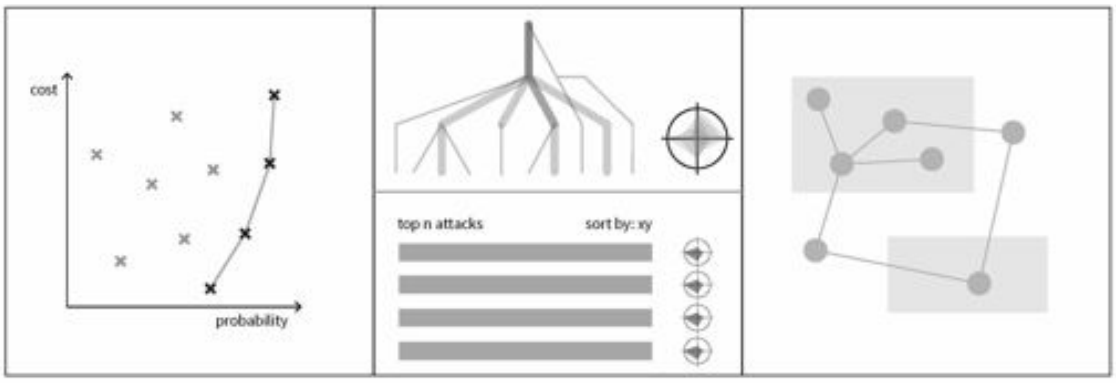

Figure 4.25.: ANM: Analysis results, prototype sketch of split screen dashboard. Note the facility for the user to interface between the analysis tools and the visualisation tools (R10). See Table 3.1, *What WP4 requires from other WPs*.

5. Validating the interpretation of the requirements

Three strategies have been used to validate the interpretation of the requirements. As the requirements can be interpreted in many ways, we have chosen to deploy a critical review approach that involves assessing the operationalisation of the requirements from the perspective of different stakeholders that are TRE$_S$PASS. Taking a critical approach that deploys a range of methods to assess the deployment of the requirements from different perspectives enables us to develop more robust visualisations that can be engaged with by different communities.

5.1. Participatory methods

The first strategy has been to purposely extend the concept of visualisation to accommodate participatory methods used to gather research on current perceptions of risk and practices of information exchange. Drawing from the participatory mapping practices that have emerged in geography and planning, the methods provide tools (in the case of TREsPASS, *LEGO* bricks for analogue 3-D modelling, see Fig. 5.1) to assist discussion among groups of their specific social and technical networks (Heath, Coles-Kemp, Hall, et al., 2014). This approach also requires consideration of the key questions of agency raised by Gandy, namely how to avoid advanced security tools from putting specific populations at a cumulative disadvantage? Or, more simply, whose security are we concerned about?

The results from these sessions can be successfully incorporated into TREsPASS methodology. In the first instance an abstracted version of the representation can be constructed, with reference to core organisational goals and alignments of values. This results in a clustering of the elements into different communities within a relational service design (Fig. 5.2). Finally the representation can be carried over into a Universal Modelling Language (UML) format (Fig. 5.3), so that more formal analysis of data-sinks and other data sharing patterns can be extracted and imported into the Attack Navigator Map. A *LEGO* representation can therefore be visually analysed and the data emerging from this can be transformed into a variety of appropriate formal formats.

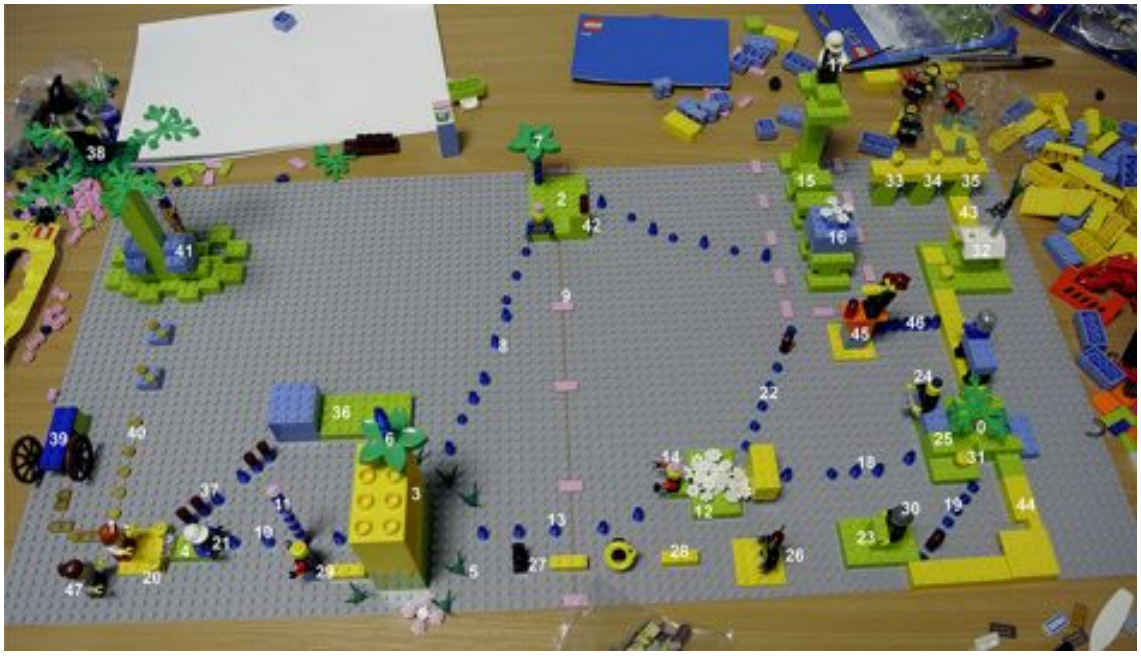

Figure 5.1.: *LEGO* model from participatory sessions. Key: **0** = LRS; **1** = Client; **2** = Card; **3** = TV; **4** = Remote; **5** = Client's sphere of interest; **6** = Antenna on TV; **7** = Antenna on Card; **8** = Data TV to Card; **9** = Boundary between Client and LRS; **10** = Data Remote to TV; **11** = Raspberry Pi; **12** = Cloud; **13** = Data TV to Cloud; **14** = Protection on Cloud; **15** = Bank; **16** = Account; **17** = Security on Bank; **18** = Data Cloud to LRS; **19** = Data LRS to Partner 23; **20** = Children; **21** = Security on Remote; **22** = Data Bank to Cloud; **23** = Partner 23; **24** = LRS Data management; **25** = LRS Server; **26** = Partner 26; **27** = Intervention in progress; **28** = Intervention pathway; **29** = Partner 29; **30** = Staff at Partner 23; **31** = Staff at LRS; **32** = Partner HA; **33** = Partner 33; **34** = Partner 34; **35** = Partner 35; **36** = Energy provider; **37** = Data Bill to Client; **38** = Governmental welfare agencies; **39** = Income source; **40** = Welfare benefits; **41** = Government systems; **42** = Additional cards; **43** = Partner bridges1; **44** = Partner bridges2; **45** = Troubleshooter; **46** = Data Troubleshooter to Partners; **47** = Carer.

Figure 5.2.: The elements of the *LEGO* model rearranged in a digital collage, making it easier to see the flow of the relational service. The client (located between the upper and middle circles), has been connected to the notion of 'impact' in the data file, and is highlighted in red as are other nodes. The central area defines the essential relationships that are required for the smooth transaction of the service, and this is supported by the outlying banking (bottom) and state systems (top).

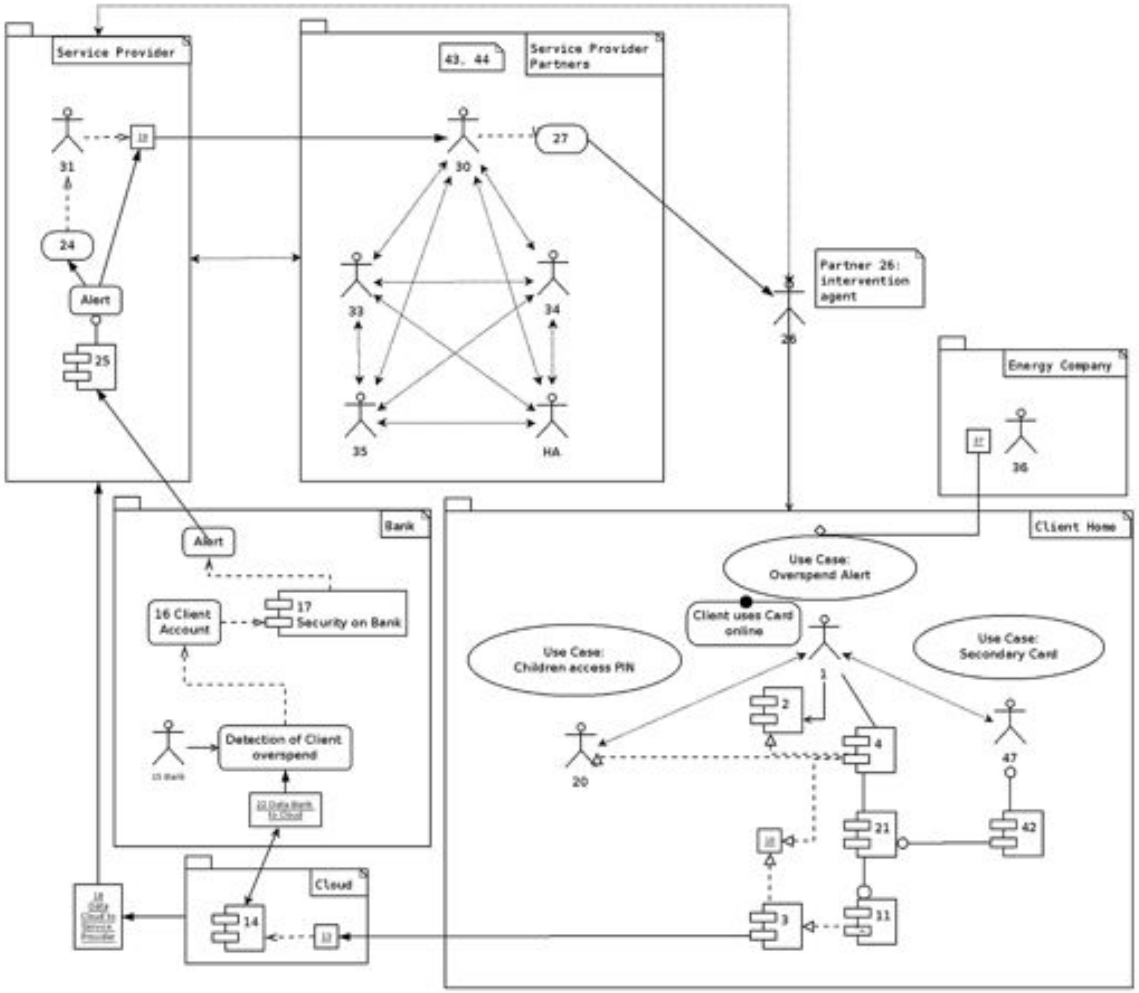

Figure 5.3.: Mapping the *LEGO* model's elements into UML format (Universal Modelling Language). This is a *UML use-case diagram*, representing three main facets of the scenario that was discussed at length during the session. Some locations are grouped as packages and connections between them are annotated with the IDs and descriptions of the corresponding parts of the *LEGO* model. This particular diagram has been arranged so that the relational service is seen as a rough circular clockwise movement of data from the client (right) to service provider and partners (left). This is a similar layout to the digital collage of the *LEGO* model.

5.2. Paper prototyping

The second strategy has been to build in extensive evaluation and review of visualisation tools before decisions about how risk is identified are black-boxed, encoded and made opaque. Prototyping is an important way of evaluating requirements and assessing whether the requirements have been meaningfully implemented. In Year 3 of TRESPASS both digital and paper prototypes of the Attack Navigator Map have been developed and evaluated. The prototypes have been used to gather requirements, further define existing requirements and evaluate instantiations of the requirements. In particular, the paper prototyping has been used to evaluate the intended visual attributes of the Attack Navigator Map.

In this section we focus on paper prototyping. Paper prototyping is a means of creating a paper version of a digital interface and inviting a participant group to engage with the paper prototype simulating the use of the digital interface. This has placed emphasis on taking paper prototypes to user groups to explore how they perceive risk through successive spheres: organisational, physical, digital and social. The importance of the spheres is to steer participants toward awareness of four significantly different views of the same issue, for example the differences between the views that security is about compliance to protocol (organisational), locking the office door (physical), changing passwords frequently (digital) or trusting a colleague with sensitive information (social).

Four main paper prototyping sessions have been undertaken in year 3: two paper prototype sessions took place in Australia, one session in Brussels and one in the UK. A mapping kit was developed for these sessions. The mapping kit was composed of:

- A map of a geographical location (in most cases a room)
- Icons for physical assets and people
- Icons representing boundaries
- Colouring pens
- Tape

An example of the map can be seen below (see Figs. 5.6, and 5.7).

In each session the same process was followed. The process steps were as follows:

- Introduce the TRESPASS project and the role of visualisation within the project.
- Present participants with a scenario and a mapping kit and explain how to use the mapping kit.
- Ask participants to identify the assets, the connections between the assets and the possible attack paths.
- Place a likelihood on the success of each attack (represented by an attack path).
- Rank the risks based on the likelihoods.

The results were recorded through photography, note-taking and the collection of the completed paper prototyping.

5.2.1. Insights from the evaluation sessions

The key insights from the evaluation sessions are as follows:

- Narratives are needed to make the map understandable.
- Risks need to be visually categorised in order to make the map usable.
- Those using the map focused on the left-hand side of the map and not on the right.

The insight about narrative has led us to consider how we can include narrative in the Attack Navigator Map. One possible avenue of innovation is to incorporate 3-D modelling in analogue methods such as *LEGO* into the more mathematically abstracted Attack Navigator Map. This will be further explored in Year 4 of the TRESPASS project.

The insight regarding risk categorisation led us to use spider diagrams and visual techniques related to the use of colour to categorise risks. Further visualisation of risk categorisation will be forthcoming as we work with the outputs of the WP3 analysis.

The insight about importance of layout and position of elements on the Attack Navigator Map will be the focus of user testing in Year 4.

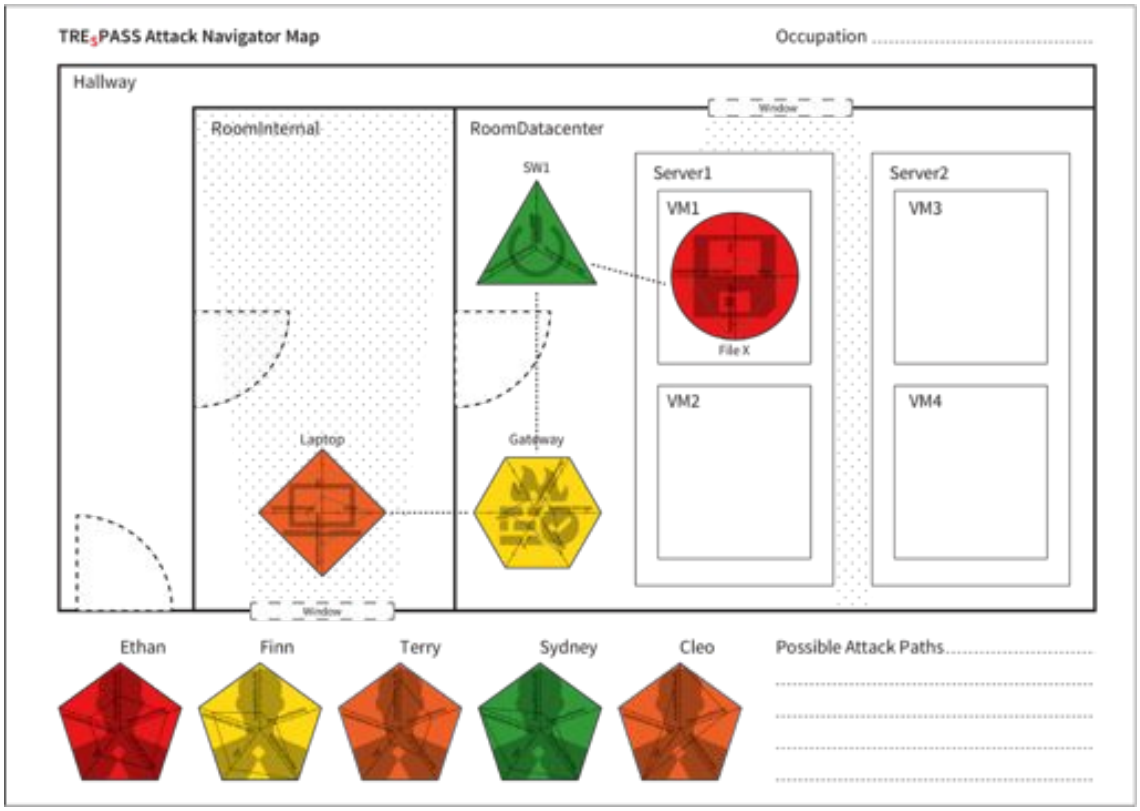

Figure 5.4.: The graphical template that was used for paper prototyping.

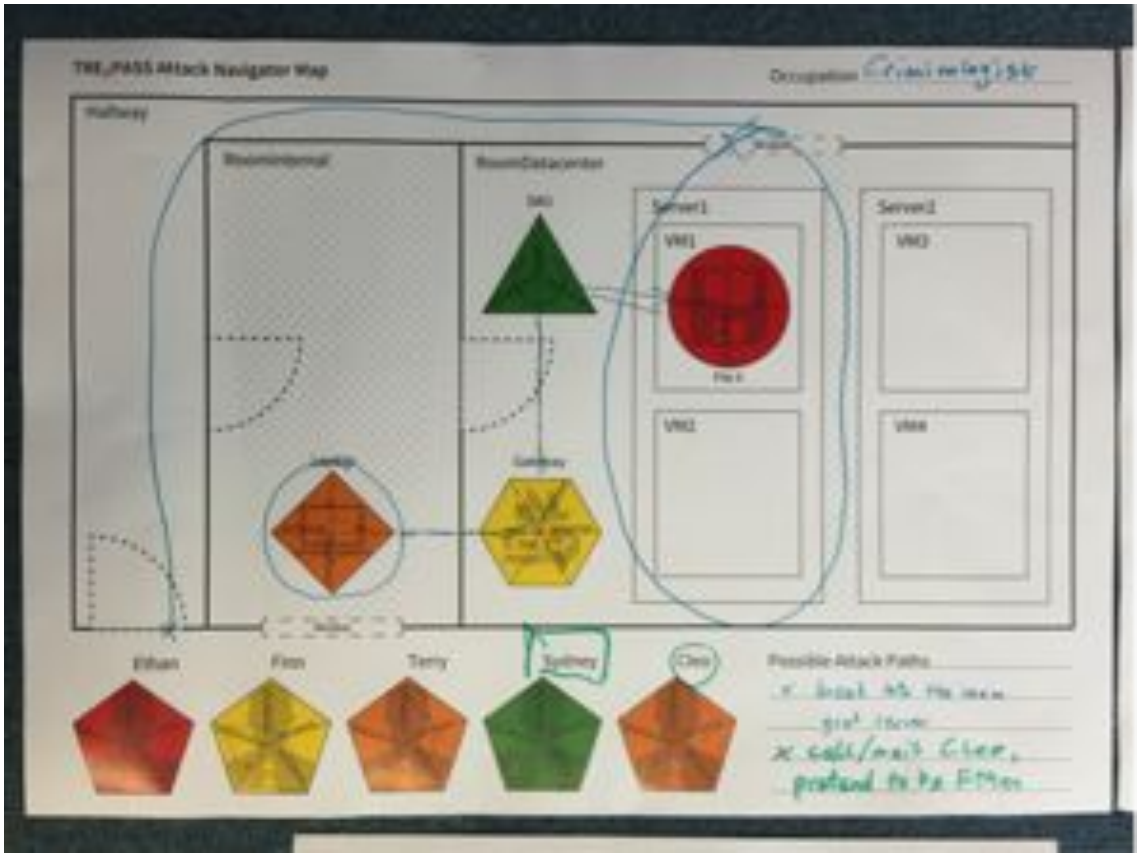

Figure 5.5.: The completed template, a sample result from the paper prototyping session.

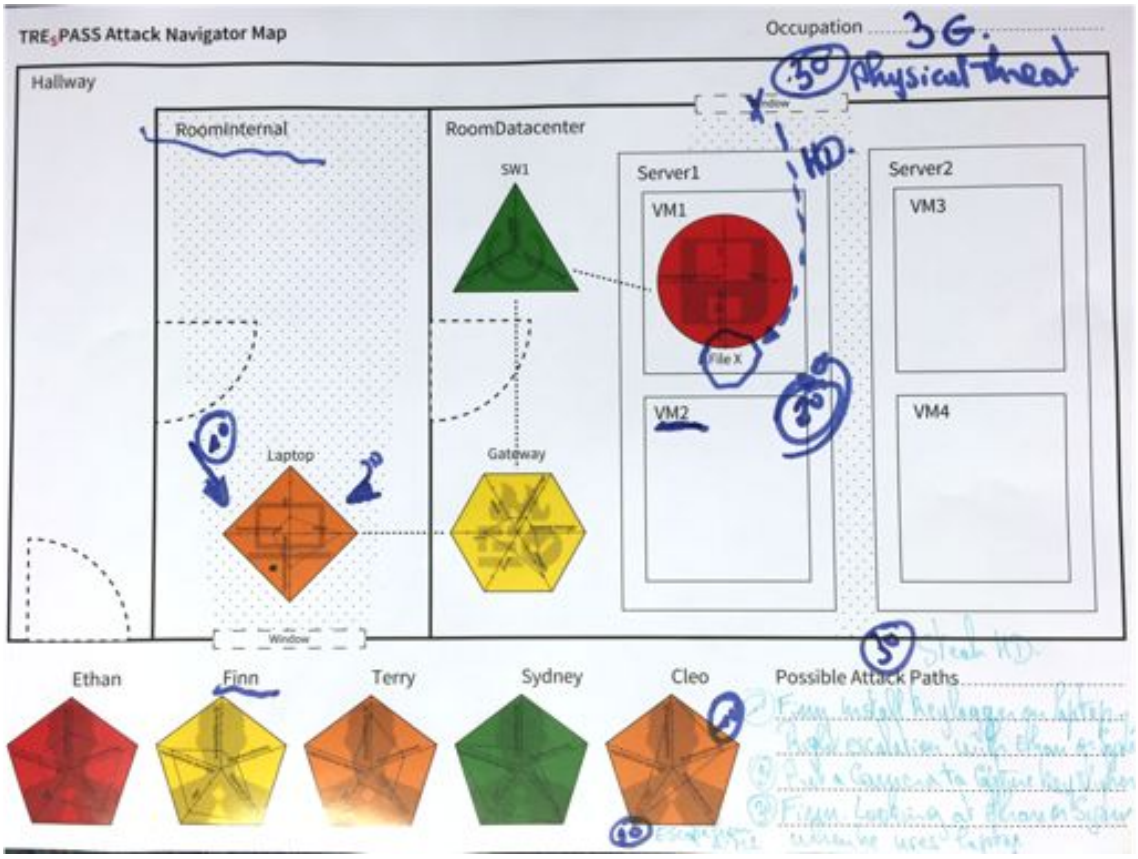

Figure 5.6.: The completed template, a further result from the paper prototyping session.

Figure 5.7.: One of the several groups at the paper prototyping session.

Figure 5.8.: The paper prototyping session, as groups are formed and begin to work on the brief.

5.3. Observe, make, and reflect - practitioner panels

The third strategy has been to use recursive design methods, such as Dubberly's "observe-make-reflect" cycle, which suggests a constant return "to the drawing board" to calibrate prototypes further in light of observations, the process of development and critical reflection (Dubberly & Evenson, 2008). Visualisation tools such as a "navigator" to facilitate understanding of a given network or computing cloud, its physical and digital layout along with access controls, the relationships between machines and social actors and perceived security of each element, are designed and evaluated by practitioners. Evaluation is an important aspect of visualisation review. Evaluations should reveal the most important of the tacit, haptic, and other design affordances, as a critique applied to the current visualisations but transferable to future iterations as well. The appropriation of the visualisations in the field needs to be understood in order to ensure that the visualisation platform is adequate and meets the needs of the TRE$_S$PASS stakeholder communities (see Table 3.1).

One insight to have emerged from this process, for example, is that of the "security onion," suggesting a multi-layered approach to organisational security, as opposed to a fixation on the defensive controls at a network's perimeter. A sequential focus on "people, process and technology" implies a more resilience-focused approach than the dominant surveillance paradigm.

The practitioner panels were run using provocations and stimulus material from the current Attack Navigator Prototypes (please refer to Annex A for details). The panel was structured in such a way that A critical visualisation practice insists, in fact, on the situatedness of what is being visualised. So rather than pushing always to abstract and generalise patterns of data exchange, it supports the representation of specific social practices toward a local understanding of risk and vulnerability. It may be that the information gleaned from a particular group during the process of visualising their information-sharing practices cannot be imported into a computer model for the predictive analysis of risk. So be it. If the net result of the participatory visualisation is to have fostered greater understanding of communication and vulnerability among the participating members, then the visualisation process has achieved a potentially greater job than that of a visualisation tool in perpetual risk of being out-manouevered by a cyber-criminal.

5.3.1. Practitioner panel feedback

In March and June 2015, two practitioners panels were held at LUST's offices in the Hague. In these panels the current state of the work on attack trees, visualisation of cloud scenarios and attack navigator map were presented, recent tools and prototypes and discussed by practitioners in the field of security. The outcomes are input for further development with regards to visualisation aspects, but also more general comments are regarded as input for the model.

The feedback provides important information not only for the visualisation process but also for the wider TRE$_S$PASS development. In addition to using the feedback to set the

direction of the visualisations and the visual look and feel of the ANM, feedback was also sent to the technical co-ordinators and priorities were set as to how the feedback will be used.

The feedback from the second panel was broadly in-line with the feedback from the first panel. Both panels were of the opinion that it was less useful to visually place the social, physical and virtual space in one map and that separate maps were more beneficial to the analyst. In response to this valuable feedback, we have designed the ANM so that you can create the physical structures (rooms, doors etc.) and then in most cases load the extract of the virtual infrastructure (constructed in for example in Graphml/yED and export this to the TREsPASS XML format. This means that the physical elements are displayed in the physical map, but all relations between the virtual elements would be displayed in in a different map.

All panelists emphasised that risk impact should be the focus of visualisations as this is crucial to an analyst's activities. This is a good example of where visualisation provides a useful feedback to the modelling activities in TRE$_S$PASS as impact is not something that is currently highlighted either in the WP1 model, in the WP3 tools or in the WP5 processes. The approach is therefore to evaluate the inputs to the visualisation process to see where impact does feature and to also identify where it might feature. Where impact is addressed, the visualisations ensure that it is highlighted and where it is not addressed this gap is communicated to the TRE$_S$PASS technical co-ordinators.

The paper prototyping and discussion work with the practitioners highlighted that most practitioners seem to start with the asset and asses what they need to do to protect these. As a result, the ANM uses maps that are asset-centric so that practitioners have a degree of familiarity with the visual layout.

Processes were also identified as another form of asset. Process hierarchies were specified as a way to describe the way the different departments, roles and actors rely on each other in doing their job within the company. This potentially provides a way of visualising the interaction between the attack pattern libraries and the physical, virtual and social assets.

6. Conclusions

In a paper in the geography journal *Cartographica*, the designer and critical cartographer Denis Wood argues that aside from the variety of instruments at his disposal for measuring the environment (tape measure, yardstick, stopwatch, goniometer, a clinometer, map measure, compass, a wall thermometer, etc), there is another important instrument in his studio that measures time, humidity and pressure in a different way: *himself* (Wood, 2010). This meditative opening provides Wood's segue for a discussion of two sciences that in the 1950s and 1960s, both of which "used humans as instruments for learning about the environment," both of which used the term psychogeography (p.194-5).

Parallels to psychogeography can be found in the associated discipline of data visualisation. In its rush to achieve scientific objectivity and what Donna Haraway called the "God-trick of seeing everything from nowhere" visualisation has sacrificed a critical disclosure: Who (or what) gathered the data, where and when and under whose orders? (Haraway, 1988). Our use of multiple assessment methods and our early engagement with practitioners enable us to create as rounded a view as possible as to what of the TRE$_S$PASS model should be visualised and how it should be visualised. As we move into the final stage of developing the TRE$_S$PASS prototypes, we shall aim to bring this rounded view into all that we create and contribute to the development of the final TRE$_S$PASS tool chain.

References

Dubberly, H., & Evenson, S. (2008). On modeling the analysis-systhesis bridge model. *interactions*, *15*(2), 57–61.

Haraway, D. (1988). Situated knowledges: The science question in feminism and the privilege of partial perspective. *Feminist studies*, 575–599.

Heath, C. H. P., Coles-Kemp, L., Hall, P. A., et al. (2014). Logical lego? co-constructed perspectives on service design. *DS 81: Proceedings of NordDesign 2014, Espoo, Finland 27-29th August 2014*.

ISO/IEC 27002:2005. (2005).

Sherwood, J., Clark, A., & Lynas, D. (2005).

The TRE$_S$PASS Project, D4.1.1. (2013). *Initial requirements for visualisation processes and tools.* (Deliverable D4.1.1)

The TRE$_S$PASS Project, D5.1.1. (2013). *Initial requirements for process integration.* (Deliverable D5.1.1)

Vines, J., Clarke, R., Wright, P., McCarthy, J., & Olivier, P. (2013). Configuring participation: On how we involve people in design.

Wood, D. (2010). Lynch debord: About two psychogeographies 1. *Cartographica: The International Journal for Geographic Information and Geovisualization*, *45*(3), 185–199.

A. Feedback from security practitioners

The feedback panel consisted of seven information security practitioners ranging from risk analysts to auditors and consultants. The panelists gave the following feedback when presented with the prototype visualisations:

Attack trees

- Attack paths are accepted as being important, but it also necessary to show defences.
- Can certain properties be prioritised, and perhaps be weighted differently (with direct probable consequences for the visualisation process)?

Visualisation feedback The following points were identified as important to note for our future development:

- Countermeasures are missing, either existing ones or those planned.
  - Representing the impact of attacks should also be considered as a priority.
  - Also the cost of mitigation as compared to possible impacts.
- The ability to show more than a single parameter at a time - the combination of parameters, leading to vulnerabilities.
  - and therefore also to find a way to aggregate these parameters.
- Spider graphs: in edge-cases spider graphs can result in data not being represented in the right manner (for example, when two axes have the value 0, and one axis a very high value, this will almost not be visible - this can be solved partially by not starting at point 0).
  - Bundling similar properties is another point to look further at, creating a distinguishable “shape” in a spider graph.
  - One suggestion was made that it would more appropriate to use a Venn-diagram.
  - The practitioners felt the user of the TREsPASS model would be given too much freedom if all the parameters were editable directly through the interface. Instead the emphasis should be to be able to select types of actors (‘Stepping-stone’, ‘Ostrich’, and so on).
- ‘The Security Onion’: a layered approach to security.

  - Despite investing heavily in their security defences many organisations are still finding their systems regularly compromised. The problem these organisations face is they are focusing too much on the defensive controls at their network perimeter in the false belief that this makes it difficult for their systems to be compromised. However, time and time again we see that once the perimeter controls fail attackers have easy access to the organisation's sensitive assets.
  - The Security Onion is an approach how to secure corporate assets by implementing a multi-layered approach to security incorporating the key cornerstones of people, process and technology. With this multi-layered approach should one security layer fail the other layers can compensate and continue to secure key organisational assets.
  - The "onion" might be interesting from a visualisation point of view, it was suggested.
- Cloud example: physical location of data is either unknown or irrelevant. Access control is interesting, however, and maybe there can be more focus on those aspects.
- The practitioners suggested to highlight the distinction between social-engineered / non social-engineered attack-steps for both the digital attack-steps and the physical attack-steps as we presented. This would mean this would need to be annotated in the attack tree decoration.
- Quantifying can happen within a range. But this range should be shown or explained.
  - On an ordinal scale, what does "high" precisely mean?

Attack paths

- They all doubted if showing all possible attack paths was useful. From a business point of view the top ten attack paths would be enough.
  - We think a top ten view can be one of the visualisations. This would be the one that you export and take to the next meeting with your supervisor or manager.
  - We think that showing all possible attack paths can somehow be useful if the user is looking for the best or most efficient countermeasure. By showing all countermeasures and indicating their control strength (in keeping the attacker from reaching his goal).
- Minor comments:
  - Why do the radial attack-trace visualisations start from 3 o'clock instead of 12 o'clock?
  - Why do we read the radial attack-trace visualisations counterclockwise?

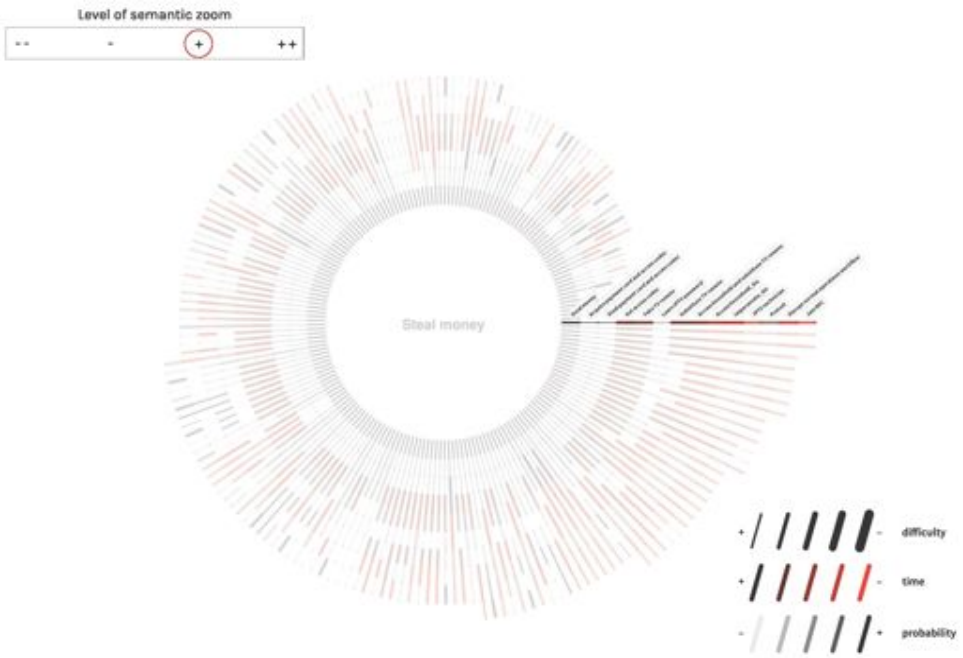

Figure A.1.: WP4 Visualisations feedback group: many kinds of user operation were presented to the group. Note the support for the visualisation of attack trees in a particularly novel way (R 40).

Building the Attack Navigator Map

- The practitioners panel preferred a visual for the Attack Navigator Map (**ANM**) that was like a floor-plan, rather than a more abstract translation of it, although they liked the aesthetics of the latter.
- The nested-ness of the digital sphere within all the physical boundaries seemed confusing to the panel. Showing both these realms suggests that it is really hard to get to the digital assets. Whereas there might be ways to get to those digital assets without leaving your own house.
- One suggestion was to split the physical map from the digital map, and make it possible to view those separately.
- A common case was mentioned, what happens if you're not the only tenant in a building, and how the policies connected with this fact are to be represented?
- Using the very same graphical 'Legend', or 'Key', and the same visualisation principles for the **ANM**, for example, might not be a good idea, since it is natural to read a floor-plan that has thicker wall sections as being more protected, whereas the opposite might be case.
  - In a building.
  - In a datacenter.

  - On a server?

- There was doubt how useful it is to show physical and digital space within a single map. If this is to be the case, they should somehow be represented so that their differences will be apparent.

  The panel was very much interested in this relationship described above, between the physical map and the digital map. They wondered in which ways physical access leads to digital access? Except for very-well encrypted data (where also the keys are not available in the same cloud infrastructure), physical access gives you complete access to digital data. With physical access you can very often boot into root-level access to servers.

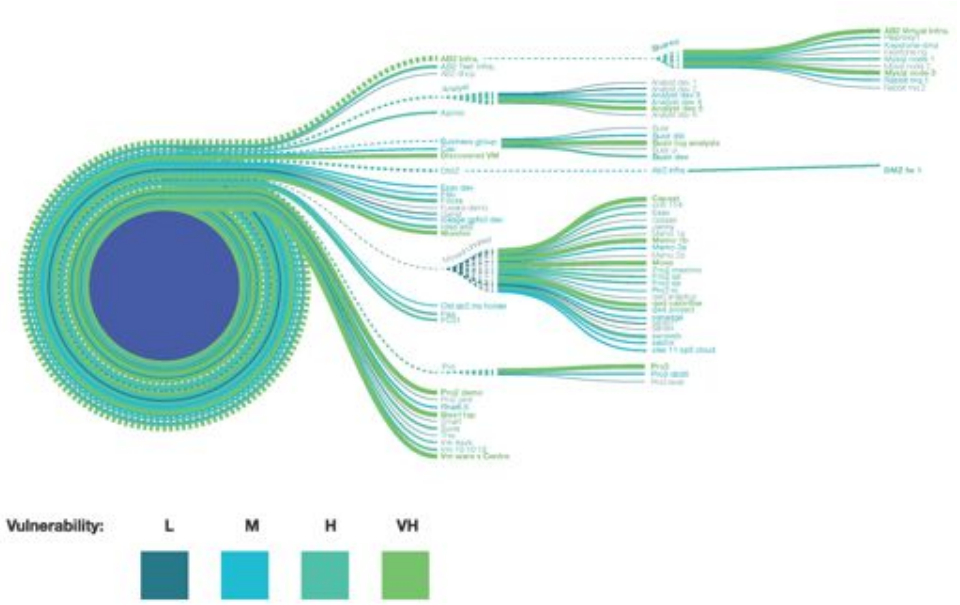

Figure A.2.: WP4 Visualisations feedback group, an example of stimulus material given to the group: combining physical and digital maps. LUST, 04-06-2015

Building ANM using the Wizard

- One thing that was missing, according to one of the panelists (Rob Cordes), was the possibility of modelling interpersonal relationships between actors:

  - Process hierarchies.

    * Can these be compared to policies? They describe the way the different departments, roles and actors rely on each other in doing their job within the company, and how do they work together.

    * Similar to BPML, and machine readable, but able to refer to these relationships.

- Alternative approach:
  - Starting point: identify and provide an inventory for the assets.
  - Estimate impacts whereupon the asset was compromised (how much damage would it cause financially and in terms of business reputation, and so on).
    * An alternative is to work outwards, building different types of controls and countermeasures around the assets that have been identified.

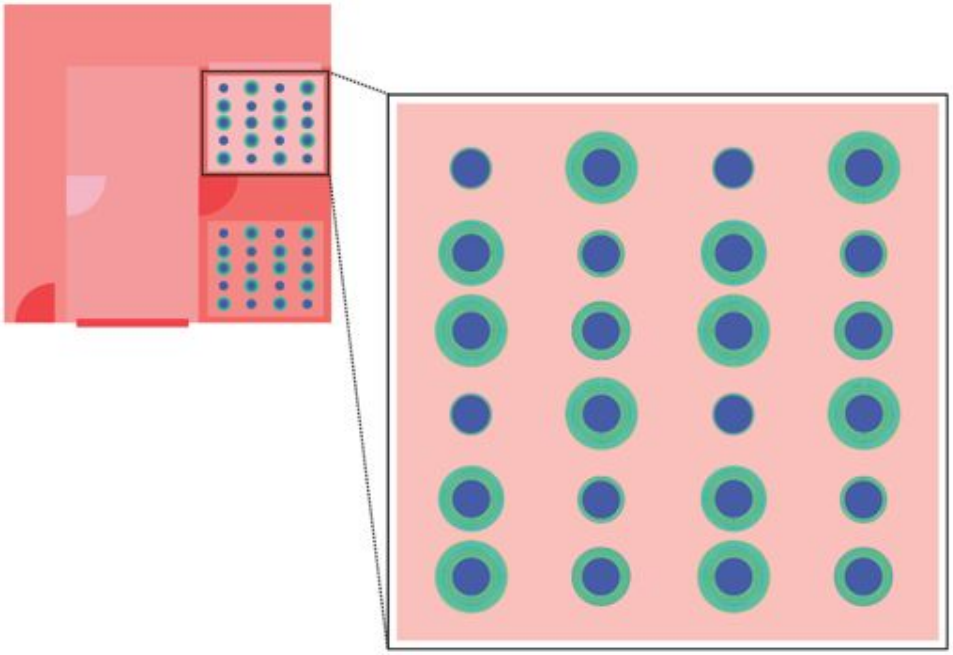

Figure A.3.: WP4 Visualisations feedback group, an example of stimulus material given to the group: combining physical and digital maps. LUST, 04-06-2015

Properties - Actors

- We looked at a few properties and agreed to send full list to practitioners for further feedback (pending). Some of their initial feedback was that the use of actor properties such as “age”, “nationality” and “gender” should be explored in greater depth. Having said this, they also had the feeling that in many companies this is a sensitive area. On the other hand, IBM for instance (as well as other US-based organisations), has a list of embargoed countries, hence this could also be a good property to develop.
- The panel suggested “seniority” as an alternative property to investigate.
- “Personality colour” can represent persona character traits (where a colour code can be used to distinguish a particular type of personality) and this is potentially interesting from a visualisation perspective.

- The panel mentioned The CALUWE test[1] where companies evaluate their employees and categorise them by using colours (see personality colour).

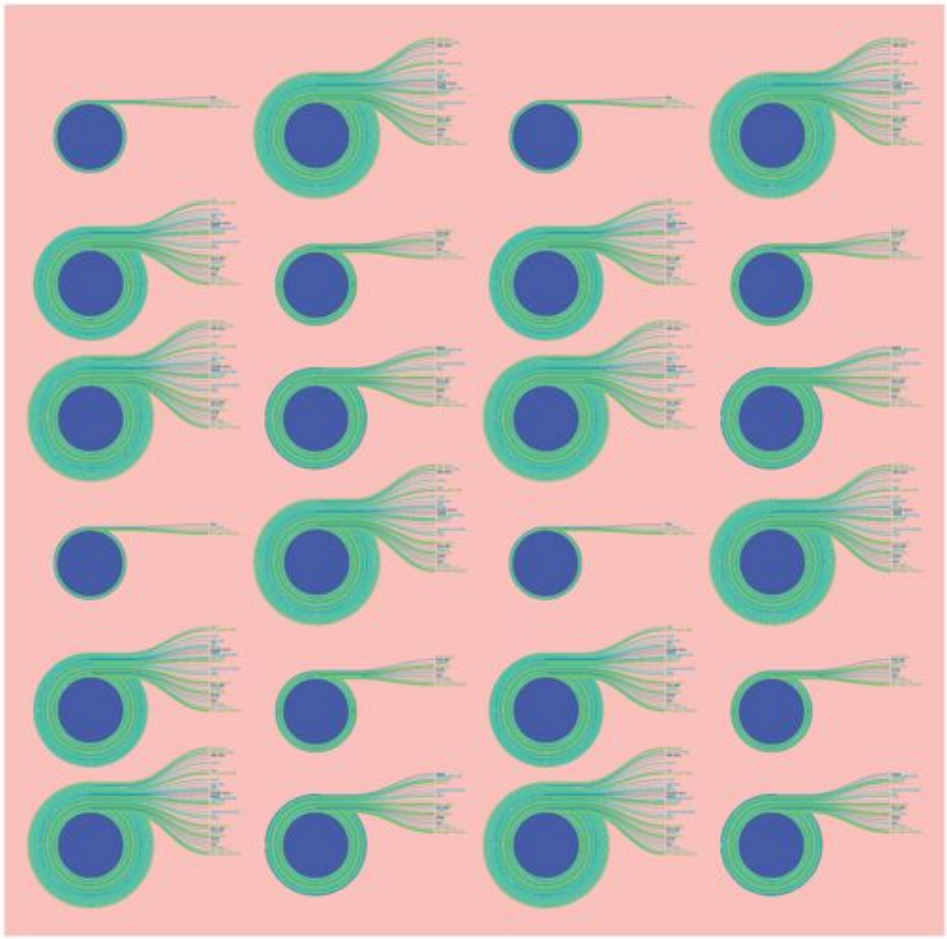

Figure A.4.: WP4 Visualisations feedback group, an example of stimulus material given to the group: combining physical and digital maps. LUST, 04-06-2015

Data

- 'Up-to-date' solutions, an essential but significant challenge.
- Quality: "garbage in, garbage out".
  - certainty: a lot is based upon professional judgement and "guess-culation".
    - One issue is how to acknowledge and communicate this.
- There should be a clear message to the user about the process is: initial modelling, leading to a first analysis, then to visualisation, and thus to iteration of this process.

[1] https://www.managementdrives.com/ and The CALUWE test

  - for them it all seemed to happen in the same view, should fix that in UI.

- Some of the practitioners felt we were forgetting about data and the movement/relocating of data within the digital infrastructure. They felt we focused too much on representing the data on one physical place (VM1), while in practice the data is likely dispersed over many servers, VMs and even physical locations.

Figure A.5.: WP4 Visualisations feedback group, an example of stimulus material given to the group: colour-coding the social usages of organisational space. LUST, 04-06-2015

Determining the dominant narrative: the approach of Risk assessment in *Shell* organisation

1. Determine which assets you most want to protect.
2. What is it's value? Determine the value of an asset based on the possible damage when an attacker gets hold of the asset (in other words, to bring about a state of affairs not desirable for your business).
3. Determine, based on this value, who possibly would get hold to this asset (attacker).
   - The attacker profile plays an important role:
     a) Who wants my assets, and why?
     b) What does the attacker want to achieve?
        i. Define your attacker profile ("know your enemy").

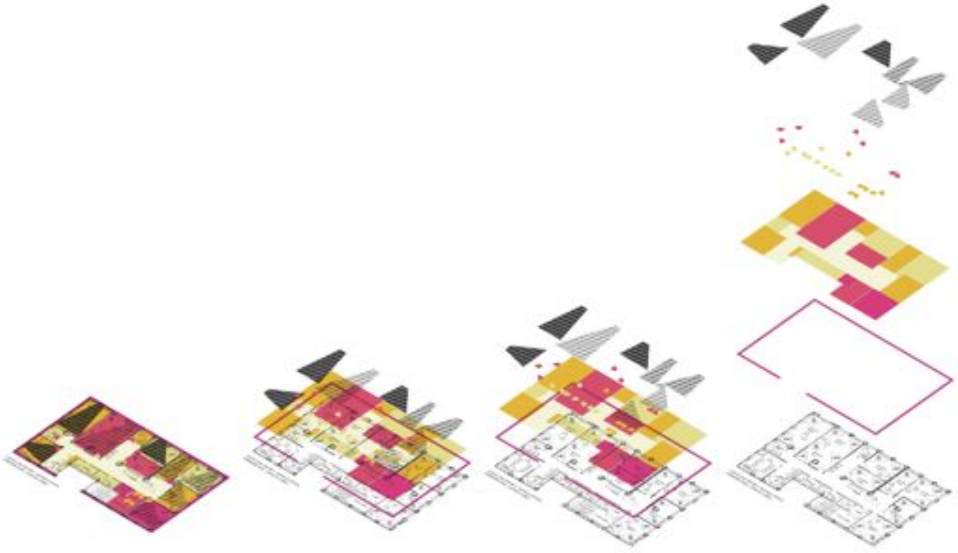

Figure A.6.: WP4 Visualisations feedback group, an example of stimulus material given to the group: the layering of spatial, human, and technical dimensions. LUST, 04-06-2015

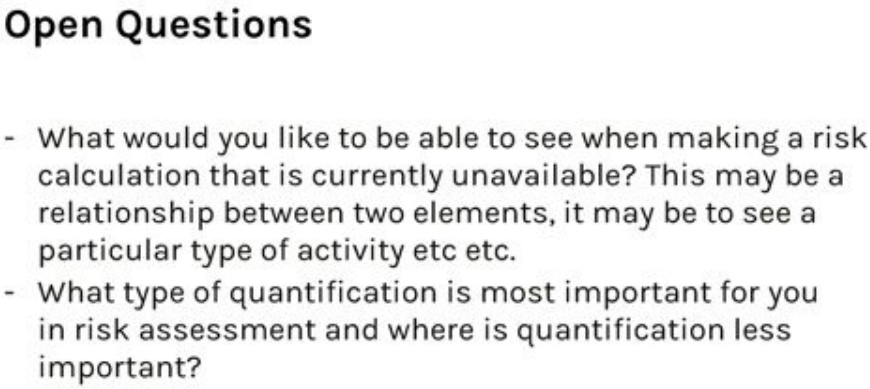

Figure A.7.: WP4 Visualisations feedback group: the presentation of open questions to the group. LUST, 04-06-2015

Purpose of the TREsPASS model The practitioners panel had some questions regarding the use of the model. They question the way this system can be updated to do the analysis on a more regular basis.

General feedback / Open questions

- Risk evaluation for the complete practitioner's panel is more based on the possible impact an attack causes or might cause. This is still a very important aspect from a business point of view.
- For the practitioners panel, risk consist of three elements:
  - Impact.
  - Vulnerability.
  - Likelihood.
- Analysis shouldn't be an unknown quantity, or a 'black box'.
  - people should know why specific countermeasures are needed

B. Project Summary

This chapter gives an overview of the TRE$_S$PASS project and its use cases. The section is shared by the public deliverables to provide the necessary background and to put the current deliverable in context.

Information security threats to organisations have changed completely over the last decade, due to the complexity and dynamic nature of infrastructures and attacks. Attacks like StuxNet involve technical and human factors, and they damage physical infrastructure. The recent attack on a German steel mill [1] was a combination of both targeted phishing emails and social engineering attacks. The phishing helped the hackers extract information they used to gain access to the plant's office network and then its production systems. As a result, the technical infrastructure of the mill suffered severe damage.

The attack on the German steel mill illustrates that we need to integrate the social and technical aspects of systems in assessing their security - and we need to do so today. Socio-technical systems pose new challenges by combining parts for which we often understand the security issues; the combined system is however much more complex due to interactions between these parts.

The main innovation of the TRE$_S$PASS project is the attack navigator, a tool and metaphor that enables defenders to predict and preventing attacks on socio-technical systems. The attack navigator supports current risk-assessment techniques with the TRE$_S$PASS process (developed in Work Package WP5), an analytical approach to identifying attacks and evaluating their impact.

The four main stages in the TRE$_S$PASS process are *data collection*, *modelling*, *analysis*, and *visualisation*. Data collection (WP2) is vital to understanding the nature of a scenario and providing input to subsequent tasks of modelling, analysis and visualisation. Within the project, the focus has been on collection and analysis of social, technical and physical data and the ways in which these relate to one another. Within each of these domains, different approaches have been taken to provide different viewpoints on the nature of the organisation being investigated.

The models (WP1) developed in TRE$_S$PASS can be adapted to the application scenario. We have developed physical modelling techniques in order to understand where further investigation may usefully be targeted. The TRE$_S$PASS model describes relevant aspects of the organisation and their connections. To explore contractual and commercial relationships, the e3value method has been adopted.

[1] BBC News, *Hack attack causes 'massive damage" at steel works*, http://www.bbc.com/news/technology-30575104, last visited October 31, 2015.

The analysis methods (WP3) developed in TRE$_S$PASS identify attacks in models and identify the most effective controls to prohibit these attacks. The analyses are supported by tools and together they provide the defender with a comprehensive understanding of properties attacks, *e.g.*, cost for the attacker, required skills, or required time.

The innovative visualisations (WP4) developed in TRE$_S$PASS focus focus particularly on visualising elements of the analysis, as this is key to the overall project goal of providing "decision support" to practitioners. However, visualisations contribute also to model development and data gathering.

Practitioners can access the TREsPASS toolkit via the attack navigator map interface, which provides an intuitive means of selecting appropriate tools (WP6) for data gathering, modelling, analysis and visualisation. These can be used, individually or in combination, to strengthen operational and strategic decision-making.

B.1. Case Studies

The TREsPASS process and tools are validated by means of case studies (WP7) in the area of cloud infrastructure, telecommunications infrastructure, ATM infrastructure, and an organisation processing privacy sensitive data.

A cloud infrastructure shares infrastructure within or across organisations, giving the cloud services provider and its employees full physical and logical access to all resources across the different consumers. In TRE$_S$PASS we formalise typical components in cloud infrastructures as well as human actors and their interrelationships, to identify their contribution to attacks on the organisation.

In telco infrastructure new products need to be launched under significant time pressure, often opening up loopholes for so-called knowledge insiders who know the market very well, trying to make as much monetary gain from the new products as possible. In TRE$_S$PASS we model both the infrastructure and contractual relationships to identify physical and monetary attacks.

The ATM infrastructure connects machines that are composed of a money safe and a computer that controls the ATM's devices. There are well protected ATMs installed inside bank branches, while others are deployed in the street and some are not even embedded in a wall. ATM attacks are common and include classic physical attacks and emerging digital attacks. In TRE$_S$PASS we model ATM installations, and identify attack likelihoods using geospatial data.

The organisation processing privacy sensitive data develops a system supporting primarily elderly and disabled people in performing online payments and managing their own money from their home. This case study involves from strictly technical security aspects, such as how information is protected while stored or transmitted, to socio-technical security aspects covering security issues arising from the use of and interaction with the technology. In TRE$_S$PASS we identify social-engineering and trust-based attacks on such systems.

B.2. Overview of TRESPASS Integration

The TRESPASS workflow involves several stages with various activities, some of which are optional. Figure B.2 shows the architecture diagram and Figure B.1 shows a visual description of the notation used. In practice, stages may not follow a linear order. For example, depending on the goal of the risk assessment, new data requests may be issued later in the process, or automatic updates of data may be supported.

The **Data collection stage** prepares for analysis and modelling steps, and may require the gathering of one or more of the following kinds of data.

Physical data collection provides knowledge about the physical layout of the organisation including locations, buildings, rooms, doors, windows, etc.

Digital data collection gathers information about the organisation's IT infrastructure.

Social data collection focuses on organisational and individual data, and results in actor profiles containing, *e.g.*, attributes of employees, stakeholders, or potential attackers.

Commercial data collection gathers information required for *e3fraud* analyses, which focus on potential fraud.

Stakeholder goal collection identifies assets and policies the protection of which is critical to one or more stakeholders.

The **model creation stage** handles the creation of the TRESPASS model and associated actor profiles. The e3value model creation process is complementary to the main TRESPASS model, for cases requiring a more specific financial focus:

TRESPASS model creation is a key activity result in a system model that can be further extended and analysed.

- **Components customisation (optional)** takes place before or during the TRESPASS model creation to create specialised custom model components.
- **Attacker profile creation** creates the attacker profile that the TRESPASS model analysis should consider, based on ready-made attacker profiles.
- **Defender/target profile creation** creates similar profiles for the other actors in the model based on the social data gathered in the social data collection activity.

e3value model creation This interactive activity involves using the *e3value toolkit* [2] to create business value models. These models structure the commercial information gathered in the data collection stage in a formal way.

In the **analysis stage** different analyses are possible depending on the model chosen. The analysis of the TRESPASS model involves these steps:

1. In the **attacker profile selection**, the user selects the attacker profile to use in the analysis.

[2] http://e3value.few.vu.nl/tools/

2. The **attacker goals creation** provides the attack generation with the attacker goals. These can be derived by hand from the stakeholder goals or deduced automatically from the selected attacker profiles.
3. The **scenario selection** selects a scenario, consisting of a single pair of attacker and attacker goal, to run the TRESPASS analysis on.
4. To extend attack trees, **attack pattern creation and sharing** provides libraries with known attack steps. The attack tree generation can only reach a certain level of abstraction, which may not be sufficient for quantitative analyses.
5. **Attack generation** transforms the TRESPASS model to an attack tree.
6. **Attack tree annotation & augmentation** then extends the attack tree with attack patterns and decorates leaf nodes with parameter values from the data collection stage for quantitative analysis.
7. The **attack tree analyses** compute quantitative properties of attacks, *e.g.*, utility for the attacker or success probability of the attack.

The analysis of the **e3value model** is complementary to the core TRESPASS analysis and has only one step:

1. For the **fraud model generation**, the user needs select an attacker and an interval of expected occurrence rates of the commercial transactions specified by the e3value model. The e3fraud tool then identifies all possible violations of contracts, the loss for actors, and the delta in profit for the other actors.

The **visualisation stage** can be used continuously to provide practitioners with feedback regarding the results of their activities:

1. **Fraud model visualisation** shows the generated attacks as a ranked list of textual descriptions of the attack steps and displays charts showing the profitability for each actor.
2. **Attack tree visualisation** shows the intermediary attack trees.
3. **Attack tree analysis visualisation** visualises analysis results.

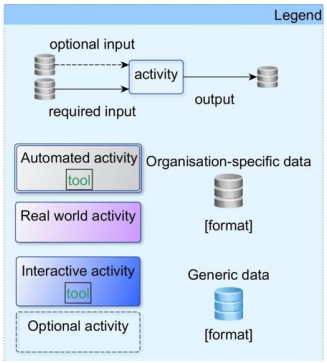

Figure B.1.: Legend for the Integration diagram in Figure B.2.

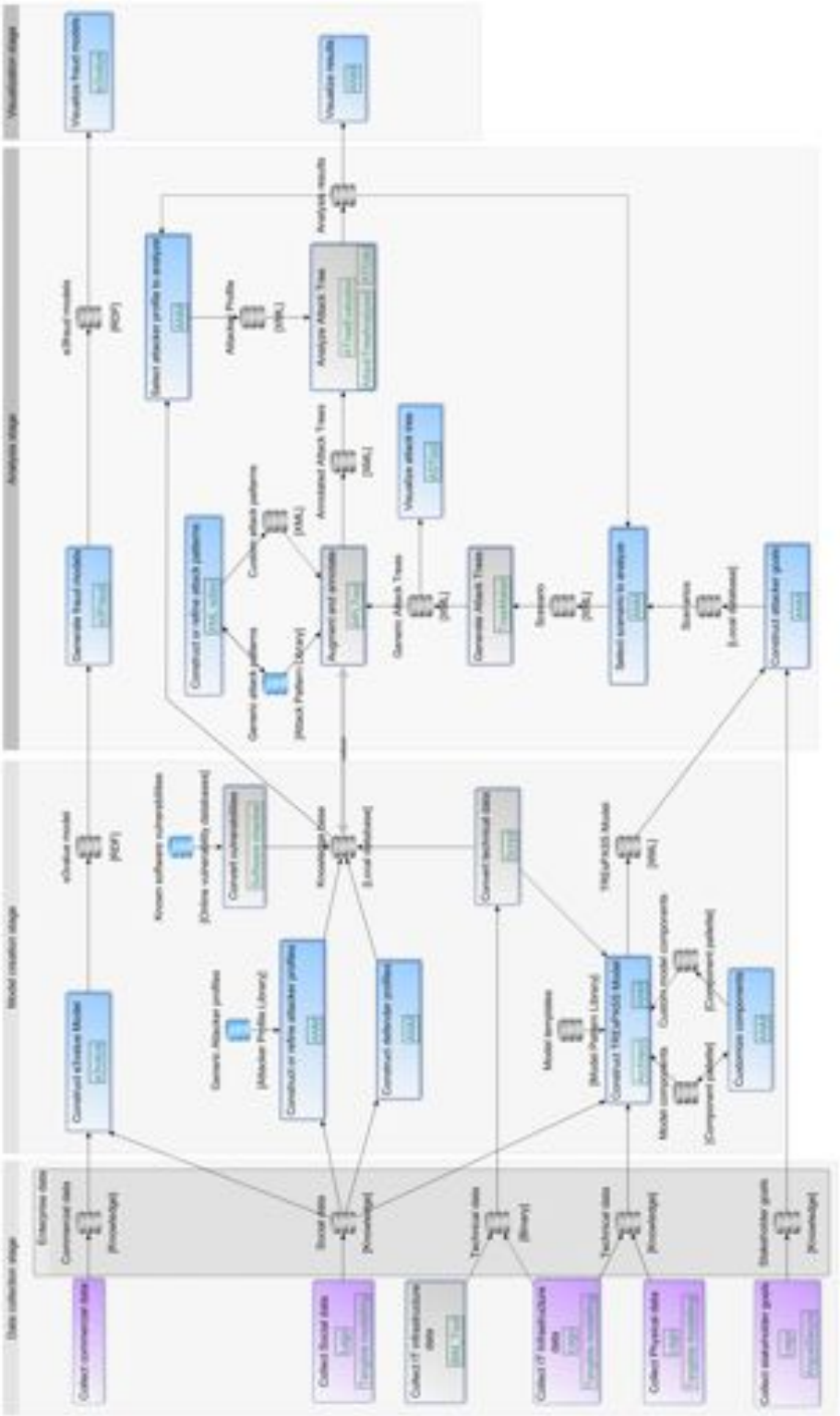

Figure B.2.: Integration diagram for the TRESPASS project.