CrimeSeen: An Interactive Visualization Environment for Scenario Testing on Criminal Cocaine Networks

F. Oetker[a], L.A.S. Roelofsen[a], R.G. Belleman[a], R. Quax[a,b]

[a] *Computational Science Lab, University of Amsterdam, Netherlands*
[b] *Institute for Advanced Study, University of Amsterdam, Netherlands*

Appendix A. Citadel

Citadel is a web application created with React [1]. The website is available to anyone with the link and is accessed through a start page. On this page, a session can be started or joined (see Figure A.2). To start a session users can either choose one of the example graph URLs or supply an existing graph in JSON or GraphML format by specifying a URL that points to that graph (see the upper part of Figure A.2). Thus, the graph must be hosted online, e.g. on GitHub or Paperbin. To join an existing session, users need to provide the session ID of that session or connect to a previously accessed session. When the user has entered a session, Citadel's view will look like Figure 3 in the main text. We will now discuss Citadel's four menu tabs as shown in Figure A.4.

Table A.1: *Comparative analysis of existing network visualization applications based on several desirable features for criminal network visualization. Here, '+' means the feature is supported, 'o' means it is incorporated to some extent but not to an accepted standard, and '-' means the feature is not supported by the application. The empty values indicate that this feature could not be judged due to a lack of available information.*

	Network Visualization Tool					
Desired Feature	**CrimeNet Explorer**	**Dynalink**	**PEVNET**	**Gephi**	**Cytoscape**	**Citadel**
Visual Mapping		o	o	+	+	+
Layout Algorithms	+	+	+	+	+	+
Filtering		+	+	+	+	-
Zooming/Panning			+	+	+	+
Selection	+	+	+	+	+	o
Collaboration	-	-	-	-	-	+
Crime Application	+	+	+	o	o	+
SNA Metrics	+		o	+	+	o
Analysis Graphs	-		+	+	+	-
Dynamics	o	+	+	+	o	o
Simulation	-	-	-	-	o	+
User Friendliness	+		+	+	o	o
Scalability	o	+	+	o	+	+
Accessibility	-	-	-	+	+	+

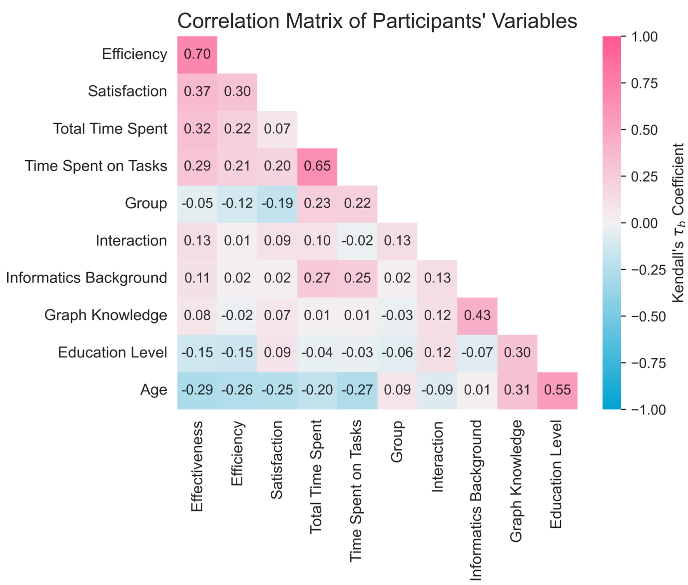


Figure A.1: *A correlation matrix created with the Kendall τ_b test of the main observed metrics, time spent, test settings, and the ordered attributes of participants' demographics.*

Mapping. In the Mapping tab under "Visual", Citadel allows the user to apply one or multiple visual mappings of node or edge attributes to visual attributes (see Figure A.4a). The possible visual attributes are radius (or width for edges), lightness, saturation, opacity, hue and exclusively for nodes: shape, text and x- and y-position. The visual mapping options for nodes are shown in Figure A.3. Under "Layout", Citadel provides the possibility to apply a range of standard layout algorithms as provided by the JavaScript library Cytoscape.js with the option to adapt the default parameters if desired.

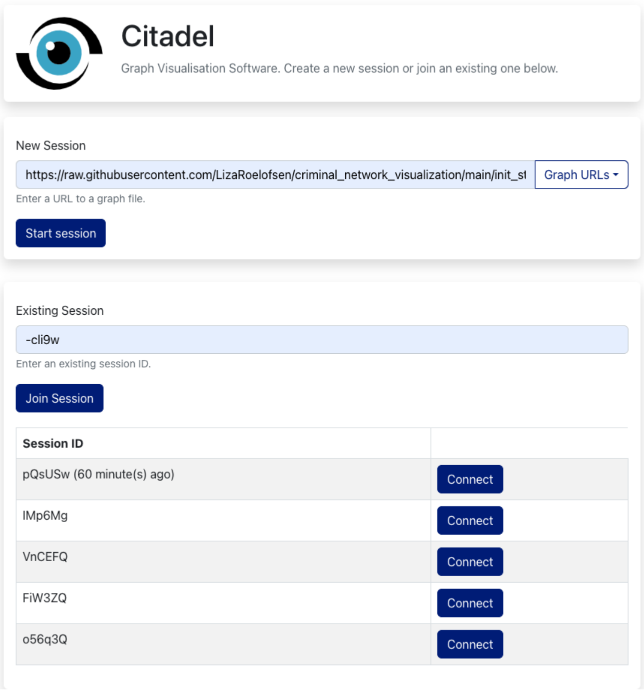


Figure A.2: *The starting page of Citadel in its current version. It shows the two options to enter a session. At the top, users can start a new session by providing a graph URL. At the bottom, users can join a session by providing a session ID or clicking "Connect" behind a previously accessed session that is still active. An example of both options is provided in the text bars to illustrate the input format.*

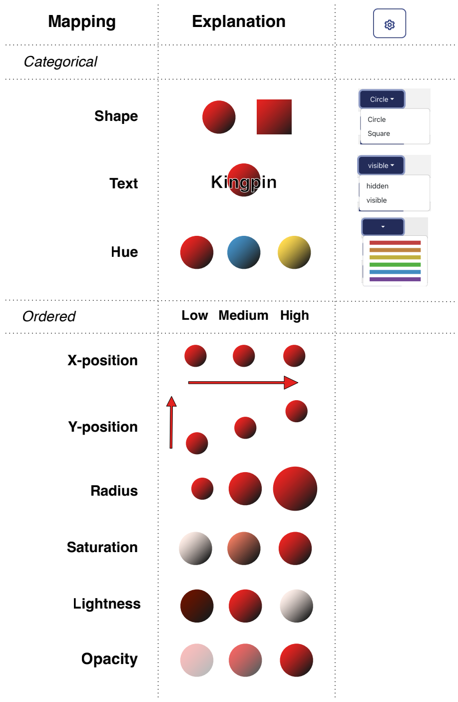


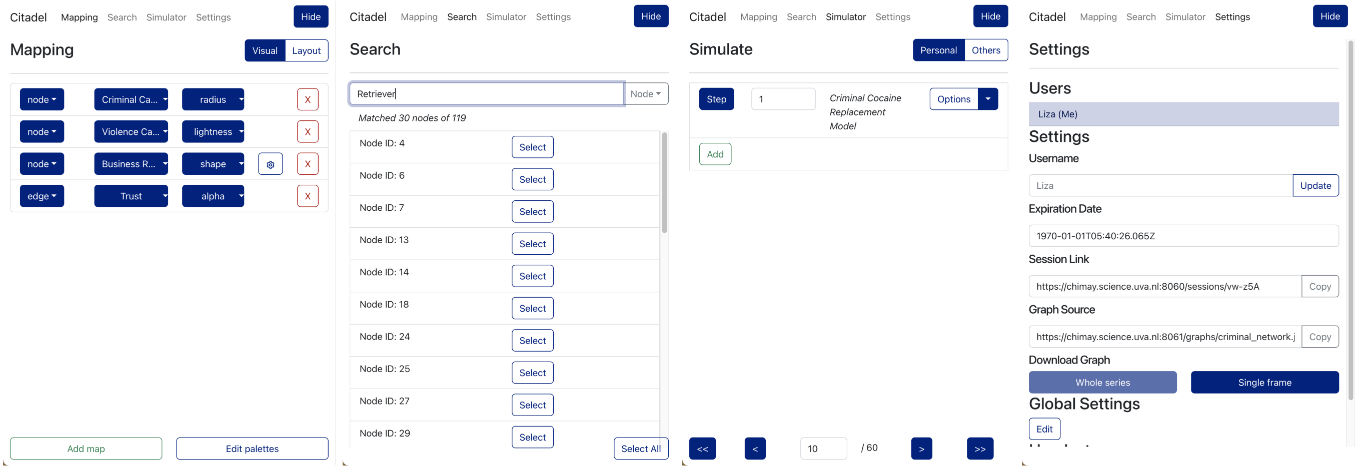
Figure A.3: *The visual attributes that can be mapped for node attributes including a visual explanation of these attributes and where applicable, the extra preferences to be set.*

Search. In the Search tab, the user can search for nodes or edges through their attribute values (see Figure A.4b). The algorithm applied to this is noisy where it accepts any node or edge that contains a similar attribute value. Subsequently, the user can select one or the whole selection of nodes or edges that match the search query through the "Select" and "Select All" buttons respectively.

Simulation. A key feature of the current application is the Simulation tab (see Figure A.4c). It allows users to connect a simulation step function through an API, by providing the Citadel URL, port, session id and simulator key as arguments to the script through which the function can be run (see Figure A.5a). To execute a simulation, the user indicates the number of steps to be performed by the simulator and clicks "Step" (see Figure A.5b). This button then changes to "Stop" and if during the simulation steps the user feels the simulation should be ended earlier, the simulation can be stopped by clicking this button. The simulation will then be ended after finishing the current iteration. After a number of steps have been completed, the user can move between the time slices and inspect the changes. The simulator can have user parameters, which can alter the simulation step and once one or multiple steps are run, the user can flip through the different time steps.

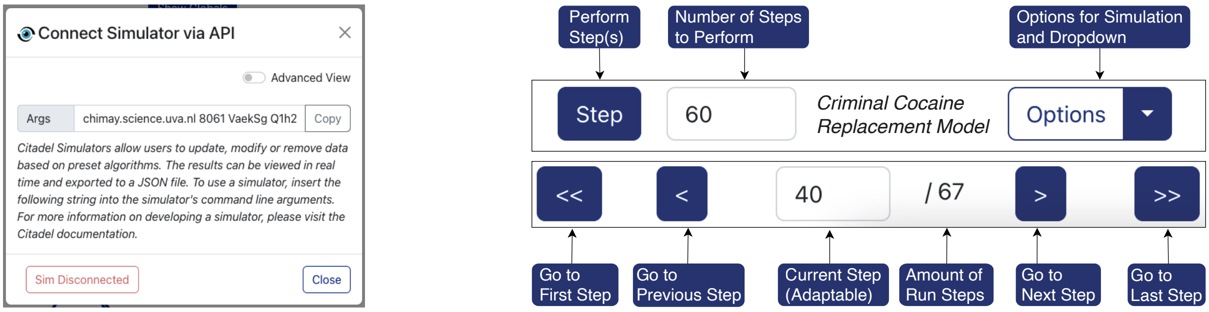
Visual interaction. Citadel has most of the standard visual interaction possibilities of modern graph visualization tools. Users can zoom and pan, center the graph and select and deselect single or multiple nodes. When the user has selected a node or edge, an inspection tab opens with all the attributes and their values (see the right panel in Figure 3 in the main text). A recent feature was the implementation of 'hidden attributes', which can be hidden or shown. The inspection tab further allows users to change the attribute values or delete the node and its accompanying edges.

Globals. A feature of Citadel which was added throughout the implementation process of this project is the possibility of providing global graph variables. These variables say something specific about the graph as a whole, which are not particular to a node or edge. It can for example contain graph statistics, a title and a description. It is especially useful for parameters used and updated by the simulator, which in the use case of the CCRM are the model parameters at initialization, e.g. time step. To inspect these "Globals", there is a button in the middle at the top of the screen (see Figure 3 in the main text). When opening this window, the user can see the variables and their values including an option to search for the variables and hide/show the hidden ones.



(a) *Mapping Tab* (b) *Search Tab* (c) *Simulation Tab* (d) *Settings Tab*

Figure A.4: *The different tabs of Citadel in its current version. In the Mapping tab (a), visual mappings can be applied by choosing "node" or "edge", an object attribute, a visual attribute and for categorical mappings specify further what to map in the settings tab. Another option here is layout, in which the user can apply a layout algorithm. In the Search tab (b) users can search in the nodes or edges list for attribute values. In the Simulation tab (c) simulations can be added, connected and run. Further, the user can flip through the simulated time steps or define some simulator parameters in "Options". Lastly, the Settings tab (d) contains session information and the option to download the graph state, change some global settings or add an AR headset.*



(a) *Pop-up modal for connection arguments and status.* (b) *Simulation controls and their meanings.*

Figure A.5: *A more detailed explanation and view of the simulation tab including the connection modal and the simulation controls.*

Appendix B. Criminal Cocaine Replacement Model (CCRM)

The CCRM is a computational ABM on a network created with the python library Mesa [2]. It simulates "the recovery of a hypothetical, but realistic, sub-segment" of a Dutch criminal cocaine network "after removal of a central node, e.g. a kingpin or a specialist", and the associated selection process for finding a replacement [3, 4]. The model was created using both quantitative and qualitative data, i.e. case files, databases created from arrest data and interviews held with domain experts [3]. The dynamics of the model can be described in phases: from an initially undisturbed network, in which an intervention occurs, giving a reaction from the network, to a recovered network (see Figure B.7).

Appendix B.1. Initial network

The model is initialized with a graph of an adaptable number of criminal agents (N) and the density of the network (edge probability). For this project, much work was put into the creation of a realistic initial network, as this is an important part in determining the applicability of the system. Since the model was evaluated on small case files, the initial conditions and model parameters were defined, but the model also needed to create a realistic initial network when no initial network was specified. The initial network requires several tags (business role, mindset, and activity) and attributes (criminal, financial, and violence capital) to be defined for each node, or agent, and which edges should be formed and with what "trust" level and relationship type. To make the initialization of these tags and attributes realistic, we created a Value Network. A Value Network is a graph itself in which the roles are encoded as nodes and the dependencies between these roles as directed edges [5]. The roles were chosen as the defining factor as this is the most important aspect for distinguishing agents. Table B.2 provides an overview of the roles and their contributions to a criminal cocaine network. For each role, their relative frequency of occurring and capital values associated with that role are defined. The dependencies contain the increase in these capital values if a connection between those roles exists and based on this increase in capital a probability of connection between the role is found to either form or not form a connection. The Value Network used for creating the initial network is displayed in Figure B.6.

Business Role	Description
Kingpin	Most authoritative and important person in the criminal network
Organizer	Organizes operations within the cocaine network in different phases
Coordinator	Coordinates the transport within the country of origin and the country of import
Financer	Finances cocaine operations
Supplier	Produces cocaine and offers it to the market
Exporter	Exports the cocaine from the country of origin (usually in South America)
Customs Officer	Ensures that the cocaine is not detected when entering the import country
Retrievers Broker	Organizes the cocaine retrieval, knows corrupt civil servants and hires retrievers
Distributor	Distributes cocaine through the network
Gatekeeper	Decides what and who gets through certain gates at (air)ports
Transporter	Transports the cocaine from the country of origin to the import country
Murder Broker	Organizes assassinations and knows and hires assassins
Placer Inland	Coordinates the amount of cocaine to be brought to each place within the import country
Stasher	Stores the cocaine until it is ready to be sold
Driver	Transports the cocaine to or from the (air)ports
Frontman	Represents the criminal organization and tries to make its activities seem acceptable to the public
Cutter	Cuts cocaine and mixes it with other substances to increase profits or change the drug's effect
Security Guard	Protects the stash houses with weapons for little money (often minors)
Retriever	Take out the drugs from containers for criminal organizations (often minors)
Assassin	Person liquidating other agents
Dealer	Deals cocaine to users or other dealers

Table B.2: *An overview of the roles used in the CCRM extracted from semi-structured interviews with and documentation provided by domain experts. The order of the roles is loosely based on their importance within a criminal cocaine network.*

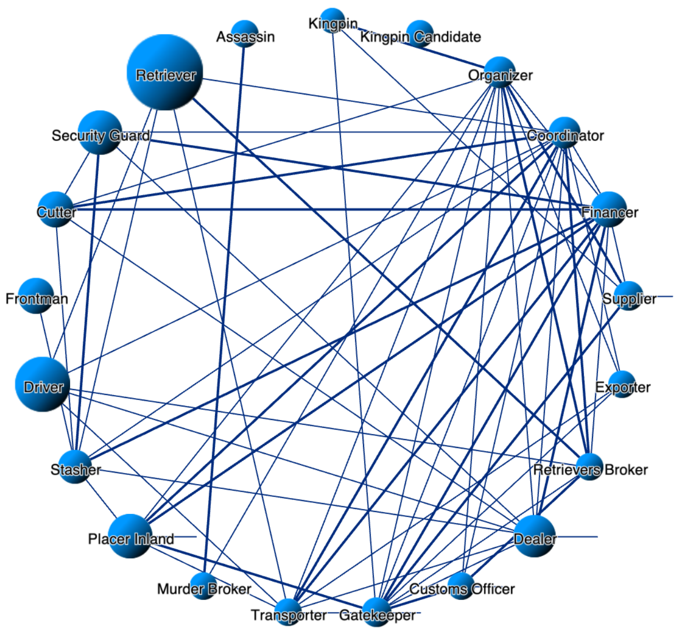


Figure B.6: *The Value Network of the different roles and their dependencies, visualized by Citadel. The size of the nodes indicates the relative frequency of each role occurring in the network. Thick edges indicate a mutual dependency and the short edges to the right that do not connect to another node indicate a self-edge, thus that role is dependent on knowing others with the same role.*

Appendix B.2. Dynamics

Using this initial state, the CCRM investigates the dynamics surrounding the removal and replacement of a ‘kingpin’ in criminal cocaine networks in the Netherlands. A minimal example of these dynamics is shown in Figure B.7. The initial network is undisturbed, after which the “kingpin” is removed from the network. Sometime after this removal, usually several days, the criminal agents that were directly connected to the original kingpin - the so-called ‘orphans’ - will try to find a replacement for the kingpin by holding a conclave among the important orphans in which the perceived candidates are considered. One of these candidates is then chosen based on the relational attribute “trust” and the candidate’s criminal, financial and violence capital. If no suitable candidate could be found the conclave is rescheduled to a later time. After a candidate is picked, there is a trial period in which the kingpin candidate has to prove himself and if the trial period has finished and he was not killed or demoted, he becomes the new kingpin.

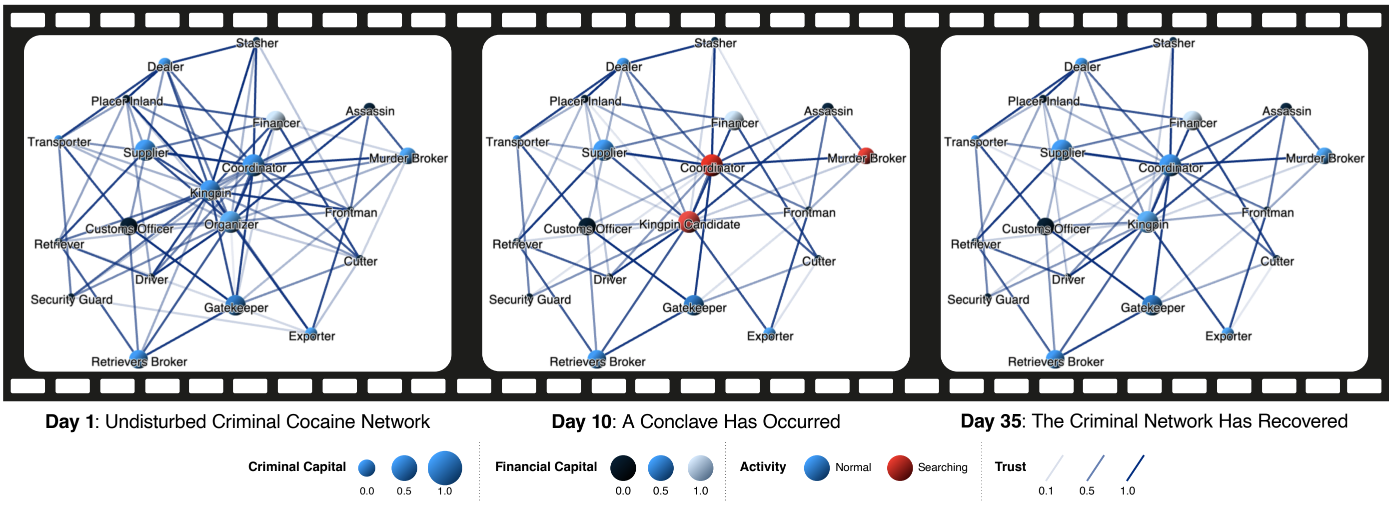


Figure B.7: *An example of the CCRM dynamics triggered by the removal of the kingpin from the network. The image displays three time steps, each in a different phase of the network during a particular simulation run.*

Appendix B.3. User parameters

The user parameters are encoded in the simulator, but the model itself drives these values. Any parameters in the model, which are not fixed and the developer wants to make adaptable to the end user, should be implemented here. Table B.3 displays these parameters for the CCRM which are incorporated in the simulator.

Appendix C. Experimental Setup

The test consisted of a Qualtrics questionnaire containing all necessary information and links to complete the test. The participants were further instructed on their test settings. These are discussed in more detail in Section **??**. Participants were encouraged to ask questions for clarification or help. The questionnaire was divided into different sections:

1. Introduction. This consisted of an introduction to the aim of the project, a consent form, a tutorial on Citadel and instructions to enter a session in Citadel.

2. Network Questionnaire. This consisted of four tasks about finding certain information about the criminal network in the initial phase and thus a static network (Appendix D, Table D.4, task IDs starting with “N”). In each of the sessions, the initial graph consisted of 120 nodes with a medium density between 0.4 and 0.65.

3. Running a Simulation. To test the ease of observing dynamics, the participants were asked to connect and run a simulation on the network. Here, a split was made between the interactive and non-interactive test settings. The participants in the interactive test setting were asked to play out the usage scenario as outlined above, mainly consisting of the removal of the kingpin themselves and another action, which slightly differed between the sessions. The participants in the non-interactive test setting were asked to simply run the model for 60 steps and observe the outcomes. For this group, automatic intervention occurred, so the kingpin still got removed and possibly replaced.

4. Simulation Questionnaire. After the simulation had finished participants received the simulation questionnaire, which consisted of seven tasks about the important dynamics and inner workings of the model (Appendix D, Table D.4, task IDs starting with “S”).

Table B.3: *The user parameters for the main simulator of the CCRM including their default values and meaning.*

Parameter Name	Default	Meaning
Automatic Intervention	True	If True, the kingpin will be automatically removed by the model. If False, then the network remains unaffected and interventions only occur through actions taken by the user.
Time to Kingpin Removal	5	Defines the number of steps after the initial state, before the kingpin node will be removed from the network. Note that this parameter is only used when ”Automatic Intervention” is True.
Time to Conclave	4	Defines the number of steps from the time step that the kingpin is removed to holding a “conclave” (or meeting) among the important agents that were connected to this kingpin to decide who will be the candidate for replacement. Note that this parameter is only used when ”Automatic Intervention” is True.
Max Number of Steps	365	The number of steps that can be run before the simulation is ended if it wasn’t already.
Update Positions?	True	Whether the x- and y-coordinates of the nodes should be recalculated based on the current edges and trust values (it uses a force-directed layout).
Fix Node Position with ID		The ID of a Node which will keep the same x- and y-coordinates around which the other nodes can change position. Note that this parameter is only used when “Update Positions?” is True.

5. Demographics Questionnaire. This was a short questionnaire consisting of six mainly multiple-choice questions about their age, educational level and field, graph knowledge, the browser used and the operating system of the device used.

6. SUS and Usability Issues. The end of the test consisted of the previously discussed 10 statements of the SUS and an open question in which participants could write down any usability issues they encountered (see Appendix E for the SUS statements and issues question).

Appendix D. Tasks

The different tasks with their ID and the posed sub-questions are displayed in Table D.4. Accompanying these questions is the intended goal behind measuring that particular task. These were chosen to be a range of different tasks that might be of interest to analyzing the results of the CCRM and thus understanding the best way to intervene in the network. The goals of each task are displayed in Table D.5.

Appendix E. Usability Questions

This appendix includes the usability questions posed to the participants. The 10 SUS statements as used in our testing session were:

1. *I think that I would like to use this system frequently when working with graphs.*
2. *I found the system unnecessarily complex.*
3. *I thought the system was easy to use.*
4. *I think that I would need the support of a technical person to be able to use this system.*
5. *I found the various functions in this system were well integrated.*
6. *I thought there was too much inconsistency in this system.*
7. *I would imagine that most people would learn to use this system very quickly.*
8. *I found the system very complicated to use.*
9. *I felt very confident using the system.*
10. *I needed to learn a lot of things before I could get going with this system.*

To make the SUS appropriate for our testing, these statements include two small adaptations to the standard questionnaire. Firstly, since the focus of the CrimeSeen is highly specific and would not generally be a system that is frequently used, "when working with graphs" was added to the end of the first statement. Secondly, since "cumbersome" is a cumbersome word and not always familiar to participants [6], it was replaced by "complicated". Bangor et al. [6] used "awkward", but "complicated" seemed more appropriate in the sentence and expected understanding of the participants.
In addition, participants received the following open question about usability issues:

Please write down any usability issues you encountered. Usability issues can be anything that:

1. *Prevents task completion (missing features)*
2. *Creates confusion (information that can be misinterpreted, unintuitive navigation)*
3. *Produces an error*

A breakdown of the scores on the individual items of the SUS is shown in Figure E.8. It displays the score distributions for each item of the SUS per test session. It can be observed that the first session only has four responses for the highest score, which increased in the subsequent sessions to 18 and 16 occurrences. Further, we considered the mean scores in comparison to the average score of 2. In the first session, only items 5 and 10 had an average score while the other items scored below this with a minimum of 1.2 for item 3, which was the item about ease-of-use. In the second session, only item 9 still averaged below 2 with a score of 1.8 for item 9, about confidence in using the system. In the third session, items 4 and 9 scored below average with a score of 1.75. Item 4 is about the learnability of the system without the need for a technical person. Overall the ranges of the mean scores per item were between 1.2 and 2.5. Thus, there were no items with a particularly extreme score, but rather most scores averaged around the expected score if the answers were randomly chosen.

Figure E.8: *Histograms displaying the frequency of each score based on participant's responses for each of the ten statements of the SUS shown per testing session.*

Appendix F. Significance Tests

This appendix includes a number of significance tests. Table F.6 displays the effects of the session on the time spent on the test in total and on the tasks. The significance results support the observed trend from Figure F.9. It shows the trend that the higher the session number, the longer participants spent on completing the questionnaire on average (Figure F.9). For session three, also the time spent on doing and answering the tasks increased compared to session one.

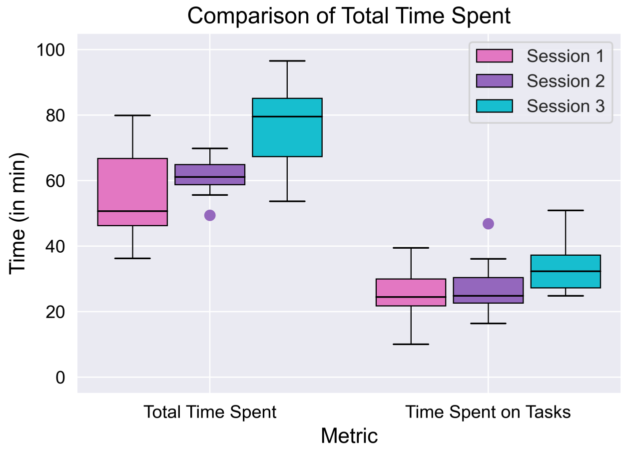


Figure F.9: *The mean total time spent on the testing for the participants across testing sessions. Each box plot represents at least 10 observations, with the third session representing 12 observations.*

Table F.7 is on the effects of the demographics and devices of the participants on the outcome metrics.

Table F.6: *The results of the Kruskal-Wallis H test on the comparison of the total time spent and the time spent on the tasks per testing session. The H-statistics are shown including their p-value between brackets and if applicable, the significance indicated with* *$\alpha < 0.05$, **$\alpha < 0.01$, ***$\alpha < 0.001$.

	All Sessions	Session 1 and 2	Session 1 and 3	Session 2 and 3
Total Time Spent	11.35 (0.003)**	1.65 (0.199)	7.67 (0.006)**	7.67 (0.006)**
Time Spent on Tasks	6.93 (0.031)*	0.37 (0.545)	5.95 (0.015)*	3.66 (0.056)

Table F.7: *The result of the Kruskal-Wallis H test for comparing the effectiveness, relative efficiency and satisfaction value for the different demographic groups. The H-statistics are shown including their p-value between brackets and if applicable, significance indicated with* *$\alpha < 0.05$, **$\alpha < 0.01$, ***$\alpha < 0.001$.

	Age	**Education Level**	**Education Field**	**Graph Knowledge**	**OS**	**Browser**
Effectiveness	3.94 (0.139)	1.22 (0.749)	2.27 (0.686)	5.98 (0.201)	0.57 (0.750)	3.48 (0.481)
Relative Efficiency	3.54 (0.170)	1.58 (0.665)	0.61 (0.962)	5.93 (0.204)	0.58 (0.747)	4.99 (0.288)
Satisfaction	3.49 (0.175)	2.95 (0.400)	1.89 (0.756)	3.38 (0.497)	1.73 (0.421)	1.40 (0.844)

Table F.8: *The result of the Kruskal-Wallis H test for the effectiveness score between the test sessions per user task. The H-statistics are shown including their p-value between brackets and if applicable, significance indicated with* *$\alpha < 0.05$, **$\alpha < 0.01$, ***$\alpha < 0.001$. *N1 is left empty since all scores were identical which means the test could not be performed and the results are not significant.*

Task ID	**Effectiveness Sessions**	**Task Time Sessions**
N1	-	0.92 (0.632)
N2	13.72 (0.001)**	8.16 (0.017)*
N3	4.39 (0.111)	6.30 (0.043)*
N4	17.09 (0.000)***	16.15 (0.000)***
S1	17.51 (0.000)***	0.35 (0.841)
S2	2.67 (0.263)	2.87 (0.239)
S3	14.38 (0.001)***	2.08 (0.354)
S4	8.18 (0.017)*	5.37 (0.068)
S5	7.72 (0.021)*	1.01 (0.604)
S6	19.71 (0.000)***	18.36 (0.000)***
S7	1.65 (0.438)	4.73 (0.094)

Appendix G. Usability Issues and Improvements

Based on the usability testing sessions, many usability issues were outlined. We provide an overview of these issues and list some additional useful information about them in Table G.9. In total, we classified these usability issues into 37 general issue topics, to clear out overlapping and recurring issues. The most outlined issue topics were issues with the visual mapping of attributes, visual clutter, slow performance and edge cluster selection. These first three are relatively broad issues, but the last one was a singular issue, that presented itself to at least five participants by trying to solve task S4. Further, most issues that were not resolved between the session in the improvements stage, were highlighted again by participants in the next testing session (Appendix G, Table G.9).

In this list we excluded comments that were not usability issues or did not regard an issue of the system, i.e. these were often considered missing explanations in the experiment setup, but could not be changed between the sessions due to reasons of comparability. Since the latter are useful issues to avoid, we list these in a separate table (see Table G.10).

Aside from the usability issues, the effectiveness and efficiency scores inspired some improvements in the implementation of the model and the simulator to aid tasks which were found to be particularly difficult to get right or took a long time to find. The improvements made are shown in Table G.11.

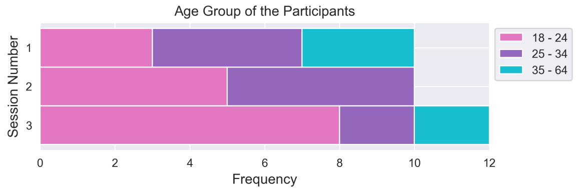
Appendix H. Participants' demographics

The demographics of the users in the test sessions are shown in Figure H.10. Participants, ranging from 18 to 64 years old, were divided into age groups as displayed in Figure H.10a. The majority of participants were students in the 18 to 24 age group. Figure H.10b displays the completed education levels, with a focus on highly educated participants. Participants who only finished high school were either pursuing a bachelor's degree or demonstrated the capacity to obtain one. For reasons of availability, 56.25% of the participants had an educational background in "Informatics", indicating a high understanding of computer systems. This group was supplemented by participants from diverse backgrounds, reflective of the police setting (Figure H.10c). The distribution of graph knowledge among participants, as given in Figure H.10d, displays a nearly equal likelihood of selecting all options throughout the sessions and an equal average level of graph knowledge with only a slight increase for the last session.

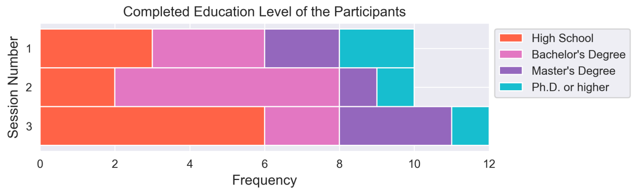
Given the small group sizes for some of the demographics due to non-selective sampling, we focused on correlations rather than comparing different groups individually. Age shows a moderate negative correlation with all outcome metrics ($\tau > 0.20$), indicating that older participants generally had lower usability scores. Nevertheless, the Kruskal-Wallis H test indicates that there is no significant relationship between any of the demographics and the output metrics (Appendix F, Table F.7). Further analysis reveals that age is strongly correlated with both education level ($\tau_b = 0.55$) and graph knowledge ($\tau_b = 0.31$) and they with each other ($\tau_b = 0.30$). Additionally, participants with an informatics background exhibit a strong positive correlation with graph knowledge ($\tau_b = 0.43$) and a moderate correlation to the total time spent on the questionnaire ($\tau_b = 0.27$).

Appendix I. dumping from main text into appendix

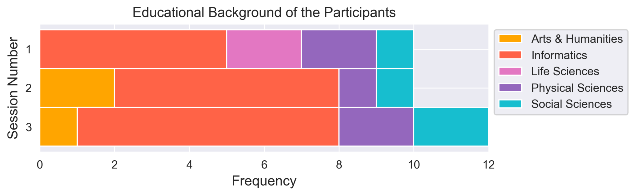
Juxtaposed timelines show the graphs next to each other (Figure I.12a). This method is very structured, and one can see changes well. Unfortunately, one will quickly run out of space as all time steps are next to each other and dense graphs are difficult to show in this way. Super-imposed timelines show each time step as a slice of a stack of network visualizations (Figure I.12b). Although it leaves relatively much space for each graph, this method has a lot of occlusion and cluttering, even with few time steps. The integrated approach combines the different time steps into one graph representation (Figure I.12c). Several ways to do this are showing the time steps within the edges, the edges next to each other or a central timeline of an ego node in which its neighbours are drawn at the point in time where the edge between them first appeared. No matter the approach, integrated timelines leave much space for the graph itself, yet the visualization quickly becomes confusing as the number of changes increases. Thus, depending on the graph and simulation run, either of these methods can be best suited.



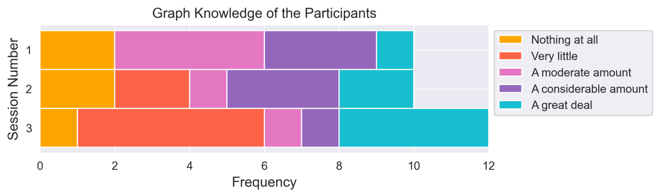
(a) *Age Group*



(b) *Education Level*



(c) *Educational Background*



(d) *Graph Knowledge*

Figure H.10: *Histograms of the different demographic categories of the participants per test session. The figure includes (a) Age, (b) Educational Level, (c) Educational Background, and (d) Graph Knowledge.*

Glossary

ABM Agent-Based Model. A computational model consisting of heterogeneous agents that act according to some predefined rules within an environment and undergo local interactions. 4

AR Augmented Reality. Superimposes a computer-generated image on a user's view of the real world, usually through special glasses or smartphones. 4

CCRM Criminal Cocaine Replacement Model. A computational ABM which simulates how a hypothetical, but realistic, criminal cocaine network recovers after a central node was removed from the model. This recovery includes the selection process for finding a replacement. 4–7

SUS System Usability Scale. A widely used and validated self-reporting questionnaire containing ten statements of which on each a respondent needs to indicate their experience on a five-point Likert-scale from strongly disagree to strongly agree. 8

Value Network A graph in which roles are encoded as nodes and the dependencies between these roles as directed edges [5]. 4, 5

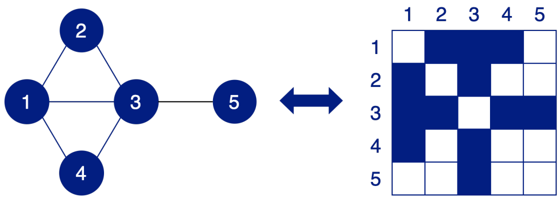
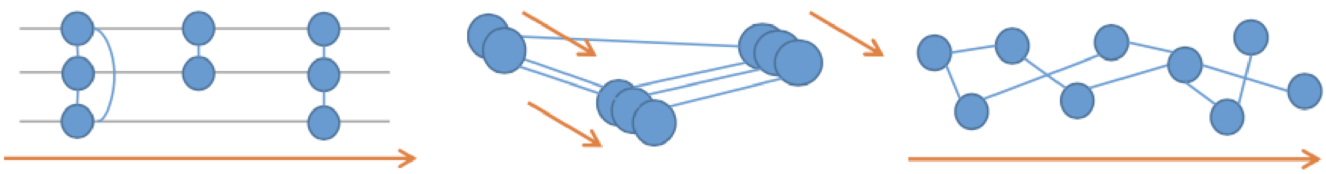


Figure I.11: *Two common methods to represent networks. The node-link representation (left) shows the nodes as circles and their connections (or edges) as the lines between them. The adjacency matrix representation (right) presents the nodes as the rows and columns of the matrix and the connections by the colours in the matrix, where blue indicated the nodes are connected and white that they are not. Both representations show an undirected simple graph.*



(a) Juxtaposed

(b) Superimposed

(c) Integrated

Figure I.12: *The three general ways to do timeline visualization with the node-link representation taken from Beck et al. [7].*

Table D.4: *The tasks posed to the participants in the form of questions with sub-questions to be answered. Between the sessions, some questions changed slightly which is indicated by the test month behind the (sub-)question.*

ID	Task
N1	*Which node is the kingpin?*
	Node ID
	How did you find the kingpin? / Citadel features used to find the kingpin
N2	*Aside from the kingpin, can you identify the other most central members in the network? (session 1, 2)*
	Can you identify the most central members in the network? (session 3)
	1. Node ID and Business Role
	2. Node ID and Business Role
	3. Node ID and Business Role
	4. Node ID and Business Role
	5. Node ID and Business Role
	Attribute/feature used to find the central members.
N3	*What is the most common business role in the network?*
	Business Role
	Business Weight of this business role (session 1) / Frequency of this business role (session 2, 3)
N4	*What is the size and density of this network?*
	Number of edges
	Number of nodes
	Density
S1	*When did each event happen?*
	1. The original kingpin was removed.
	2. A potential kingpin was (first) chosen. (session 1)
	A first kingpin candidate was suggested by an undercover agent. (session 2)
	A second node was removed. (session 3)
	3. The potential kingpin was not picked. (session 1)
	A first conclave was held. (session 2)
	A first conclave was held, but no kingpin candidate was picked. (session 3)
	4. The potential kingpin was picked. (session 1)
	A kingpin candidate was picked during a conclave. (session 2)
	A first conclave was held and a kingpin candidate was picked. (session 3)
	5. A picked kingpin was removed. (session 1)
	A kingpin candidate was demoted or removed. (session 2, 3)
	6. The kingpin was deemed worthy for 30 days. (session 1)
	A new kingpin was picked. (session 2, 3)
S2	*Who is the new kingpin, if there is one?*
	Node ID
	Kingpin's Business Weight / Kingpin's Criminal Capital
S3	*What changes to the network structure occurred? Options: none, one, or multiple.*
	Number of nodes added
	Number of nodes removed
S4	*For each attribute below, indicate how its mean value changed from the first to the last time step.*
	Options: increases, stays the same, or decreases.
	Trust (edge)
	Violence (edge) / Violence (node)
	Business Weight (node) (session 1) / Criminal Capital (node) (session 2, 3)
S5-7	*Predict who will likely replace the old kingpin.*
S5	*Select 2 potential kingpin roles.*
	1. Business Role
	2. Business Role
S6	*What agents in the model decide who the new kingpin becomes?*
	Number of agents deciding
	Business role(s) of these agents
S7	*What edge attribute(s) are used to choose the kingpin? (session 1)*
	Which of these attribute(s) are used to choose a new kingpin? (session 2, 3)

Table D.5: *The intended measuring goals for each of the tasks posed to the participants.*

ID	**Goal of the Task:** *Measures the ability of users to...*
N1	...find a node of interest within Citadel.
N2	...identify important nodes in the network (an often used task to evaluate graph visualization software).
N3	...find statistics of attribute values and inspect the statistic.
N4	...find statistics of the graph in general.
S1	...detect important events that occurred in the network.
S2	...find whether a person of interest exists and if so, find its attributes.
S3	...find what important changes occurred.
S4	...find how some attributes developed.
S5-7	...understand the inner dynamics of the CCRM.

Table G.9: *A list of the usability issues that were outlined by participants, grouped in issue topics to clear out any overlapping issues. Ordered by first appearance in the results.*

Addressed in Session(s)	Issue Nr.	Issue Topic	Primary Issue Type	Number of Mentions	(When partly) Resolved?
1 and 2	1	UI scaling	Prevents task completion	4	After session 1
All	2	Visual mapping	Missing feature	7	After session 1
1 and 3	3	Visual clutter	Creates confusion	5	
1	4	Rendering during simulation	Produces an error	1	After session 1
1	5	General Rendering	Produces an error	1	After session 1
1 and 2	6	Timeline	Missing feature	3	After session 1
1	7	Overview graph information	Missing feature	2	
1	8	User-friendliness	Prevents task completion	2	After session 1
1	9	User levels	Missing feature	1	After session 1
1	10	Edge selection	Missing feature	1	
All	11	Search of attribute values	Missing feature	4	After session 1
2 and 3	12	Simulator error	Produces an error	4	After session 2 and 3
2 and 3	13	Slow performance	Prevents task completion	5	After session 2
2 and 3	14	Edge cluster selection	Produces an error	5	
2 and 3	15	Ordering	Creates confusion	4	After session 2
2 and 3	16	Complicated to run a simulation	Prevents task completion	3	
2	17	Rendering during simulation	Produces an error	1	
2 and 3	18	Select all (between time steps)	Creates confusion	4	
2 and 3	19	System crash	Produces an error	5	
2 and 3	20	Shared session	Creates confusion	2	
2 and 3	21	Time step of inspected values	Creates confusion	3	
2	22	Inconsistency	Creates confusion	1	
3	23	Selection in Search tab	Creates confusion	3	
3	24	Selection short cuts	Missing feature	3	
3	25	Changing node position	Missing feature	1	
3	26	Difficulty bulk selection	Unintuitive navigation	2	
3	27	Information not shown	Prevents task completion	4	
3	28	Too technical	Prevents task completion	2	
3	29	Unintuitive navigation	Creates confusion	2	
3	30	Different graph representation	Missing feature	2	
3	31	Warning creates confusion	Creates confusion	2	
3	32	Graph not centered	Creates confusion	2	
3	33	Selection highlight	Creates confusion	4	After session 3
3	34	Difficulty quick controls	Creates confusion	1	
3	35	Referencing in attribute values	Missing feature	1	
3	36	Model confusion	Creates confusion	1	
3	37	Complicated to perform analysis	Prevents task completion	1	

Table G.10: *A list of the seven issues about the way the test itself was conducted as outlined by participants including in which session the issue was addressed.*

Addressed in Session	**Issue Name**	**Description**
1	Visual mappings explanation	Inital instructions on what the mapping options are and the effects of altering them, could have used improvement. e.g., I knew “how” to add mapping but not “why”.
2	Layout algorithm explanation	It was unclear that you could change the format of the graph, some more explanation about that and the parameters it entails would’ve been nice.
2	Too many complicated words	The instructions used a bit too many words that needed an explanation, it would benefit from either more visual instructions (screenshots/screen recording) or simpler terminology.
2	Start-up period required	I’d probably need someone to help out with the program for the first week or so, but I think it becomes easier to use after that
2	Step-by-step introduction	If you want inexperienced people to interact with this system, you need a better step-by-step introduction to this method as a whole.
3	Simulation connection explanation	In the tutorial, I missed an explanation of how to run a simulation, i.e. that you need a simulator and the website to interact with each other.
3	Include subtitles in tutorial	Add subtitles to the tutorial.

Table G.11: *A list of the improvements made before and between the testing sessions.*

When	Improvement	Description
Before session 1	Graph selection without URL	Be able to start a session without the need for a graph URL that is hosted online. This feature allows the user to click one of a selected number of graphs that are listed on Citadel's start page.
	Node removal visualized	Fixed the bug that when a node was deleted, the visualization still showed it.
	Additional visual mappings	Included the option to visualize nodes attribute values with the visual mapping, text and shape.
After session 1	Default Layout	Created a default layout visualization
	(based on trust levels)	Based on trust levels, the higher the trust the closer together and map nodes in chains where applicable
	Indicated when mapping (not) selected	Given by a different button colour when a selected mapping is mapped or not.
	Search of attribute values	Allows users to find nodes with specified attribute values and bulk selection.
	Annotation settings	Provides the global setting to scale the annotation to a desired size.
	Improving Bulk-selection	Provides an easy way to select all nodes/edges in the graph through the long click and drag and through the search tab in which the matching results can be selected or when nothing is searched to select all nodes or edges.
	Show and Hide buttons	Provides users with the possibility to hide the current menu tab and/or inspection tab when an object or objects are selected to utilize the full space of the screen for the visualization of the graph itself and to show these again when the user wants to change or inspect something.
	Highlight current value mapping dropdown	To allow users to see what value was selected, it is highlighted with blue. This can be of use when not the full name of the attribute is visible in the mapping button itself and the user wants to check this value by opening the dropdown.
	Stop simulating button	Allows for stopping the simulation during the generating process of multiple steps. When clicked, it finishes the current step, but will not generate the next one. This is especially useful when a simulator is very slow, so when the user accidentally started the simulator or when after a few steps it is already clear that something is going wrong, the user can stop the simulator from generating more steps and save a lot of time.
	Connection modal	Provides the status of the connection and all connection arguments in one place, so they can be copied at once, which wasn't possible before and required getting the information from four separate places.
After session 2	Globals	Allows for global variables to be used in the simulator which are not part of the graph state, but are updated throughout the simulation. This is an essential feature in running models with parameters that are updated during the simulation.
	Resolve button inconsistency	Since the "Remove" button for a single node or edge selection and the "Deselect All" button for the deselection of a cluster nodes or edges is located at the same location, their difference was made apparent with a red and grey colour respectively. After the third session, this visual difference was improved through a different position for the "Remove" button.
	Order attributes alphabetically	Especially useful for new users that are searching for particular attributes and can now easily find them as alphabetical is a clear order to most users.

References

[1] M. Van Der Lely, A Novel Visualisation Paradigm Using AR Superimposition, Bachelor Thesis, University of Amsterdam (July 2022).
URL https://scripties.uba.uva.nl/search?id=728606

[2] J. Kazil, D. Masad, A. Crooks, Utilizing Python for Agent-Based Modeling: The Mesa Framework, in: R. Thomson, H. Bisgin, C. Dancy, A. Hyder, M. Hussain (Eds.), Social, Cultural, and Behavioral Modeling, Vol. 12268, Springer International Publishing, 2020, pp. 308–317. doi:10.1007/978-3-030-61255-9_30.

[3] F. Oetker, V. Nespeca, T. Vis, P. Duijn, R. Quax, Framework for Developing Quantitative Agent-Based Models Based on Qualitative Expert Knowledge: An Organised Crime Use Case, [Unpublished manuscript] (2023).

[4] L. A. S. Roelofsen, F. Oetker, M. Van der Lely, R. G. Belleman, R. Quax, CitadelPolice: An Interactive Visualization Environment for Scenario Testing on Criminal Networks, in: EuroVis 2023 - Posters, The Eurographics Association, 2023, pp. 17–19. doi:10.2312/evp.20231057.

[5] W. Kenton, Value Networks: Definition, Benefits and Types (December 2022).
URL https://www.investopedia.com/terms/v/value-network.asp

[6] A. Bangor, P. T. Kortum, J. T. Miller, An Empirical Evaluation of the System Usability Scale, International Journal of Human-Computer Interaction 24 (2008) 574–594. doi:10.1080/10447310802205776.

[7] F. Beck, M. Burch, S. Diehl, D. Weiskopf, A Taxonomy and Survey of Dynamic Graph Visualization, in: Computer Graphics Forum, Vol. 36, Wiley Online Library, 2017, pp. 133–159. doi:10.1111/cgf.12791.