A Methodology to Develop Agent-Based Models for Policy Support Via Qualitative Inquiry

Vittorio Nespeca[1]**, Tina Comes**[1]**, Frances Brazier**[1]

[1]*Delft University of Technology, Jaffalaan 5, 2628 BX Delft, The Netherlands*
Correspondence should be addressed to *v.nespeca@tudelft.nl*

Abstract: Qualitative research is a powerful means to capture human interactions and behavior. Although there are different methodologies to develop models based on qualitative research, a methodology is missing that enables to strike a balance between the *comparability* across cases provided by methodologies that rely on a common and context-independent framework and the *flexibility* to study any policy problem provided by methodologies that focus on capturing a case study without relying on a common framework. Additionally, a rigorous methodology is missing that enables the development of both theoretical and empirical models for supporting policy formulation and evaluation with respect to a specific policy problem. In this article, the authors propose a methodology targeting these gaps for ABMs in two stages. First, a novel conceptual framework centered on a particular policy problem is developed based on existing theories and qualitative insights from one or more case studies. Second, empirical or theoretical ABMs are developed based on the conceptual framework and generic models. This methodology is illustrated by an example application for disaster information management in Jakarta, resulting in an empirical descriptive agent-based model.

Keywords: Research Design, Simulation Methodology, Empirical Agent-Based Models, Information Diffusion, Information Management, Crisis Management

Introduction

1.1 Qualitative research is a powerful means to capture the human dimension and many studies e.g., from the social sciences provide qualitative insights on human interactions and behavior that can support policy formulation and evaluation (Black 1994; Levina & Vaast 2005; Altay & Labonte 2014). As such, the development of Agent-Based Models (ABMs) for supporting policy formulation and evaluation can benefit greatly from integrating the results of qualitative research (Yang & Gilbert 2008; Adam & Gaudou 2017; Maxwell 2020).

1.2 Translating nuance-rich qualitative data into a computational model is, however, challenging (Janssen & Ostrom 2006). While the contextual richness of the data should be preserved as much as possible, distortions need to be minimized and a transparent approach is needed that ensures replicability (Yang & Gilbert 2008; Edmonds 2015; Bharwani et al. 2015). Several methodologies have been proposed for integrating qualitative data into ABM by (a) using previously developed frameworks to interpret and structure data, and/or by (b) "constraining" the knowledge elicitation and analysis process through clear steps (Edmonds 2015). For instance, Ghorbani et al. (2010) show the potential of using conceptual frameworks developed for institutional (re)design to support the design, implementation and analysis of ABMs in socio-technical systems. Further, Ghorbani et al. (2015) provide an approach for structuring and interpreting qualitative data from ethnographic work on the basis of a previously developed framework (or meta-model). Conversely, Bharwani et al. suggest a mixed-methods research methodology that puts emphasis on the steps adopted to extract and validate agent rules via a participatory and ethnographic process (Bharwani 2004; Bharwani et al. 2015). Within such a methodology, the authors rely on an exploratory phase to design a context-specific game that captures the world views and decisions of the study participants. Such a game is then used to extract agent rules.

1.3 Two gaps were identified concerning methodologies for the development of ABMs for policy support through qualitative data. Firstly, methodologies as (Ghorbani et al. 2015) that rely on pre-existing conceptual frameworks to develop ABMs have the advantage of enabling the study of a specific policy problem across different cases in a rigorous and consistent manner (i.e., through the same framework) which enhances the comparability of results across different cases. However, these methodologies lack in flexibility as they can only be used when an adequate framework already exists that can be applied to the policy problem at hand. For novel policy problems such a framework may not yet be available. Further, in order to support a rigorous and systematic approach, the frameworks used in these methodologies are "prepackaged" with particular agent architectures (Ghorbani et al. 2010, 2013) - i.e., formalized descriptions of agent theories concerning the internal processes occurring within the agents (Wooldridge & Jennings 1994). However, such a design choice reduces the flexibility of these methodologies in terms of enabling to account for different agent architectures.

1.4 Other methodologies such as Bharwani (2004) and Bharwani et al. (2015) rely on specific steps to rigorously develop ABMs tailored to a given case. While such methodologies can be flexibly applied to any case without the need of a pre-existing framework, the lack of a common framework makes it difficult to ensure comparability and consistency across different cases of the same policy problem. In sum, a methodology is missing for developing ABMs for policy support that enables to maintain rigour while striking a balance between (a) comparability as the ability to retain "common ground" among different modelling studies focusing on the same policy problem so that their results can be meaningfully juxtaposed and (b) flexibility with respect to the ability to capture novel policy problems and use different agent architectures.

1.5 Secondly, the process of designing and evaluating policies supported by ABMs may involve the development and use of a series of models with different purposes ranging from theoretical to empirical (Gilbert et al. 2018; Edmonds et al. 2019). Theoretical models can e.g., support the formulation of hypotheses regarding the implications of given mechanisms or policies prior to testing them empirically (Hazy & Tivnan 2003; Altay & Pal 2013). Conversely, empirical models can be used for instance to (a) capture a preliminary description of the currently-available knowledge that is relevant for a considered policy problem and case (Watts et al. 2019), (b) evaluate whether particular mechanisms that are relevant for policy formulation and evaluation actually explain the emergent patterns of a system (see for instance Adam & Gaudou 2017), and (c) explore ex-ante the potential implications of given policies within the context of a case study (Edmonds & ní Aodha 2019; Badham et al. 2021). However, a methodology is missing that is versatile in that it enables the development of ABMs with different theoretical or empirical purposes focusing on the same policy problem.

1.6 In this article, the authors propose methodology that enables to develop ABMs for policy formulation and evaluation based on qualitative inquiry in a flexible and versatile manner. The methodology involves (i) the development of a novel conceptual framework that is centered on the considered policy problem through one or more case studies (when such a conceptual framework in not available), and (ii) the development of empirical or theoretical ABMs guided by the application of this conceptual framework in combination with generic models. While the conceptual framework is designed to enable the identification of the agents and their interactions that are relevant for the considered policy problem, generic models are used to guide the design of the internal processes that drive the agents' interactions found through the conceptual framework. Generic models provide a common language that can capture different agent architectures such as BDI (Rao & Georgeff 1991) in a formalized, abstract, and reusable manner (Brazier et al. 2002). By choosing different generic models capturing different agent architectures, the modeller is provided with flexibility with respect to the choice of the way the agents and their interactions are translated into the internal rules driving the agent's interactions. The paper is structured as follows. Section 2 reviews briefly the use of ABMs to support policy making, and introduces the compositional design of ABMs through generic models. Section 3 outlines the proposed methodology, which has two phases: conceptual framework development and model development. Section 4 shows the application of this methodology to a case study of disaster information management in Jakarta, resulting in a conceptual framework and a descriptive model. Section 5 discusses the findings from the application, presents their implications, outlines directions for future research, and provides the conclusions.

● Background

2.1 This section introduces the phases of the policy making cycle and discusses how ABMs can support at least two of these phases. Next, the compositional design of ABMs through the re-use of generic models is illustrated.

Supporting policy making through ABMs

2.2   Conventionally, policy making has been conceptualized as a cycle characterized by distinct phases (Lasswell 1956; Lindblom 1959, 1979). These phases include: agenda setting; policy formulation and decision making; policy implementation; and policy evaluation and termination. While this cyclic view has received criticism for oversimplifying the complexity of the policy making process, it is a helpful theoretical framework to capture the main components that make up the formulation and evaluation of a policy (Jann & Wegrich 2007).

2.3   Gilbert et al. (2018) suggest that computational models can support at least two of the phases of the policy cycle, namely policy formulation and decision making with models that assess policies ex-ante, and policy evaluation through models that enable the ex-post analysis of policy outcomes. Maxwell (2020) stresses the value of qualitative inquiry in supporting both policy formulation and evaluation and its compatibility with other quantitative approaches. In this article, qualitative research is therefore combined with the quantitative agent-based models to support the formulation and evaluation phase of the policy-cycle.

Compositional development of ABMs through generic models

2.4   Brazier et al. (1997) propose a compositional development method for multi-agent systems (and ABMs[1]) called DESIRE (DEsign and Specification of Interacting REasoning components). According to this method, both the agents and the system as a whole are modelled as a compositional architecture i.e., as series of interacting components that are hierarchically structured and task-based. DESIRE enables to specify both the agents' interactions (or "inter-agent functionality") and the internal processes driving such interactions (or "intra-agent functionality") in an explicit and precise manner. Further, in DESIRE the design of an ABM includes two types of knowledge, namely (a) a *process composition* concerning the tasks that the agents carry out, and (b) a *knowledge composition* capturing the knowledge structures the tasks rely on. The process composition consists of a task hierarchy (i.e., the tasks to be executed and their sub-tasks), tasks' input and output, information exchange among tasks (or information links), sequencing of tasks (or task control knowledge), and task delegation (which agent carries out which tasks). Knowledge composition consist of knowledge structures that capture (a) the ontology or "information types" representing the relevant concepts the tasks rely on and (b) the knowledge bases representing the rules followed by the agents when executing tasks on the basis of the concepts. Central to the DESIRE method is the principle of compositionality, according to which the knowledge composition and process composition are captured at different levels of abstraction i.e., from abstract tasks or knowledge structures to their more and more specialized components. For instance, in the case of process composition, the abstract task "own process control" can be composed of sub-tasks (or components) such as "determine goals and commitments" and "evaluate own processes" (Brazier et al. 1996).

2.5   Instead of designing an ABM from scratch every time, Brazier et al. (2000) suggest that existing generic models can be re-used. Such generic models can be developed for different types of agents and agent architectures thus providing the model developer with a range of options with respect to the type of agent to be considered when designing ABMs (Brazier et al. 2002). Generic models consist of abstracted representations of the process and knowledge composition of ABMs (according to the principle of compositionality discussed above). In the case of process composition, a generic model is abstracted with respect to the tasks that the agents can carry out. These tasks can be *specialized* from the abstract categories provided in the generic model, to more specific sub-tasks that are the required for the considered ABM. Further, in the case of knowledge composition, a generic model is abstracted with respect to the knowledge structures the tasks rely on. The knowledge composition included in the generic model can be *instantiated* by (a) finding additional if, then, else statements that are nested in those captured by the knowledge bases of the generic model or (b) by providing new knowledge bases for the sub-tasks introduced in the specialization of the generic model's process composition. The knowledge composition can also be instantiated by introducing new categories in the ontology of information types accessed by the tasks.

2.6   In the seminal work by Wooldridge & Jennings (1994), Wooldridge & Jennings (1995), a weak notion of agent is introduced which is meant to capture some of the most general and widely recognized features that characterize an agent (e.g., "autonomy", or "social ability"). Brazier et al. (2000) build on this weak notion of agent to design a generic model called Generic Agent Model (GAM). GAM provides a unified and formalized language which can be specialized and instantiated to capture a wide variety of agent types and architectures in a consistent, comparable and reusable manner. An overview of GAM is provided in the Appendix. Several applications of GAM have been proposed that capture different agent architectures (e.g., normative, cooperative, or BDI) (Brazier et al. 2002). These applications enabled the design of new generic models that are more specialized than GAM as they apply to specific agent architectures and classes of problems. For instance, GAM was re-used

to design a new generic model called Generic Cooperative Agent Model (GCAM) that can be used to develop ABMs capturing distributed project coordination and assuming joint intentions (Brazier et al. 1996).

Proposed Methodology

3.1 The methodology introduced in this article involves two interlinked phases: conceptual framework development and model development, see Figure 1. In phase one, a conceptual framework capturing a specific policy problem is developed based on existing theory (literature and ABMs) and case studies. Case studies are carried out mainly through qualitative inquiry. However, quantitative data may also be collected e.g., on demographics. In phase two, a model is developed based on the conceptual framework, along with further insights from theory, empirical data from the case studies (analyzed through qualitative and possibly also quantitative data analysis), and generic models. Each phase is explained in detail in the following sections.
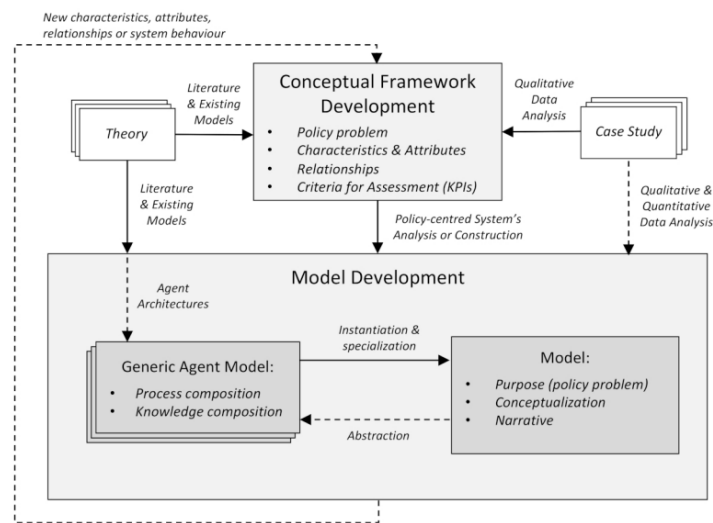

Figure 1: Methodology for developing agent-based models for policy formulation and evaluation based on qualitative research. The dashed line symbolizes activities that are optional.

Phase 1: conceptual framework development

3.2 In this phase, existing theory and qualitative research from one or more case studies is used to design a policy-centered conceptual framework. By policy-centered conceptual framework the authors intend a "list" of categories of meaning and their relationships that are relevant when developing ABMs focusing on a specific policy problem[2],[3]. In the following, firstly the composition and use of such a conceptual framework is discussed. Secondly, the steps suggested for the development of conceptual frameworks are presented.

Conceptual framework use and composition

3.3 ABMs can support policy formulation and evaluation by enabling to simulate and systematically compare the behavior (or performance) of a system given the system's configuration and possible alterations of this configuration through policy interventions (Gilbert et al. 2018). Studying the results of such simulations can support the formulation and evaluation of a theory of change i.e., a theory of how and why particular policy interventions produce the intended outcomes (Weiss 1995, 1997; Connell & Kubisch 1998). Essential elements of a theory of change are the policy's intended outcomes, the policies that may be introduced to achieve these outcomes, and the contextual factors that may affect the policy's implementation and outcomes. Given the above, the authors provide the following definitions of system's configuration, change and performance. A *system's configuration* consists of the agents, their activities and interactions with other agents and their environment that represents the context in which given policy interventions are introduced (Maxwell 2020). A *system's change* refers to shifts in the configuration of a system both as a direct consequence of the agents' choice, or through

emergent, self-organized and bottom-up processes generated by the agent's interactions (Kauffman 1993; Comfort 1994; Anzola et al. 2017). A *system's performance* is intended as the extent to which a system reaches the desired behavior (i.e., the policies' intended outcomes). System's performance can be influenced both by the system's configuration and its change. The seminal work by Levina & Vaast (2005) can provide an example of a study capturing a system's configuration, change and performance. In this case, system's performance is measured as the volume of knowledge exchanged across an organizational boundary. Levina & Vaast (2005) study how such a system's performance is affected by the system's change intended as the emergence of a new organizational community (or "joint field of practice") across the organizational boundary. The emergence of such a community is facilitated by actors who adjust their role to become "boundary spanners in practice" depending on contextual factors captured in the system's configuration. Such contextual factors include the formal nomination of the actors as boundary spanners, their inclination to engage in boundary spanning activities, and their recognition as legitimate participants and negotiators on both sides of the organizational boundary.

3.4 The conceptual frameworks designed with this methodology are meant to enable to capture the key agents and their interactions that are relevant for the development of both theoretical and empirical ABMs that can support policy formulation and evaluation with respect to a considered policy problem. Specifically, the conceptual frameworks provide the means to carry out both (a) a policy-centered analysis of an existing system (e.g., a case study) by structuring qualitative data and translating it into an empirical model (Ghorbani et al. 2015), and (b) a policy-centered construction of an abstract system to be studied via a theoretical model. To this end, conceptual frameworks are required to capture a system's configuration, change, and performance.

3.5 Firstly, to capture a system's configuration, a conceptual framework needs to provide the *system's characteristics* and their *attributes*. A system's characteristics are the fundamental components of a system (e.g., agents and their environment) that are relevant for the policy problem considered as they constitute the context in which the policy interventions are introduced. Attributes are the features that distinguish different instances of a given characteristic. In the case of the study by Levina & Vaast (2005), system's characteristics are for instance the organizations as defined by their boundaries, the actors who belong to such organizations, and their negotiation and knowledge sharing activities (interactions). Attributes of the characteristic "actor" are for example an actor's recognition as a legitimate participant and negotiator on both sides of the boundary, and his/her inclination towards participating in boundary spanning activities.

3.6 Secondly, in order to capture the system's change, a conceptual framework is required to include the *relationships* among characteristics. Such relationships represent the way the characteristics interact, possibly leading to (emergent) changes in the system's configuration. An example of relationship from the article by Levina & Vaast (2005) is that between actors belonging to different organizational units who can interact via knowledge sharing activities. Through such interactions actors may develop an inclination towards boundary spanning activities, leading them to gradually assume the role of boundary spanners in practice.

3.7 Thirdly, the *criteria for assessment* are the indicators used to measure the performance of a system. In the case of Levina & Vaast (2005), the criterion for assessment is the volume of knowledge exchanged across the considered organizational boundary. Figure 2 shows the composition and use of a conceptual framework devised to guide the development of both empirical and theoretical ABMs for policy design by capturing a system's configuration, change, and performance.
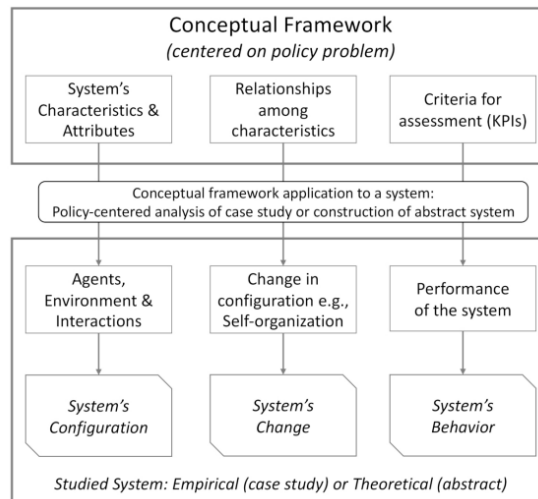
Figure 2: Composition (top) and use (bottom) of a conceptual framework tailored to a specific policy problem that enables to analyze an existing system (or case study) or construct an abstract one from the perspective of the considered policy problem. This analysis informs the development of an ABM for policy support (Phase 2).

Conceptual framework development steps

3.8 Developing conceptual frameworks that enable the design of ABMs for policy formulation and evaluation requires an approach that can capture the complexity of social and socio-technical systems. Brazier et al.'s approach was chosen for developing such conceptual frameworks given it can be applied to both social and socio-technical systems (Brazier et al. 2018). Based on the chosen approach, the conceptual framework development phase follows the steps shown in Figure 3. In the following sections, each of the steps is described in detail.
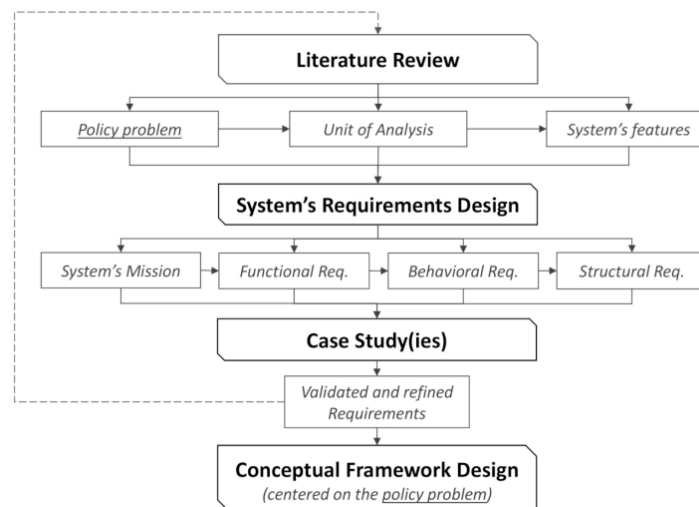

Figure 3: Development of a conceptual framework tailored to a specific policy problem in four steps. The steps are shown in bold and their results are presented below in italics. The dashed line illustrates possible iterations through the steps.

Literature review

3.9 The literature and existing models (including ABMs) related to the type of system in question are studied to identify (a) the type of problem to be addressed via policy design (policy problem), (b) the unit of analysis for the given policy problem, and (c) a list of relevant system's features from the perspective of the unit of analysis. (a), (b) and (c) are only preliminary at this stage and may be refined or changed based on case study research (Eisenhardt 1989).

3.10 As already mentioned, the unit of analysis is carefully chosen to reflect the policy problem at hand. This unit refers to the micro-level entity that is going to the center of the agent-based models the researchers intend to design. Indeed, given the generative nature of agent-based modelling (Epstein 1999), it is crucial that the conceptual framework takes the perspective of the intended model's most elementary unit. Examples of unit of analysis are a person, a household, an organization, or an entire region.

Requirements design

3.11 Brazier et al.'s approach entails the design of the system's mission and of the associated functional, behavioral and structural requirements (Brazier et al. 2018). The mission of the system is its intended purpose, that, in this case, is to address the chosen policy problem. The functional requirements are the functions that the system has to fulfill to achieve the mission. Behavioral requirements define the desired system's behavior associated with the fulfilment of the functional requirements, and the criteria for assessment that can be used for measuring the extent to which the desired behavior is achieved. Structural requirements are the components of the system that are put in place to fulfill the behavioral requirements.

3.12 At this stage, the system's mission, and the functional, structural and behavioral requirements are designed based on the results of the previous step. Specifically, the mission is designed based on the policy problem. Then, the functional requirements are designed based on the system's characteristics and the mission. Next, the behavioral requirements are derived from the functional requirements. Finally, the structural requirements are designed based on behavioral requirements and the list of relevant systems' features. This design process results in a preliminary list of requirements.

Case study

3.13 In this step, the preliminary list of functional, structural and behavioral requirements is verified and expanded based on one or more case studies (Eisenhardt 1989; Flyvbjerg 2006). First, the case study is designed. This includes the selection of a case study, data collection techniques (e.g., interviews and focus groups, participant observations and archival data), and sampling strategies all of which are summarized in a data collection plan (Yin 2009; Miles & Huberman 1994). The collected data is then analyzed through *coding*. The way such analysis is carried out depends on the type of data collection techniques chosen (Miles & Huberman 1994). However, in all cases the analysis begins with the preliminary list of requirements from the previous step.

3.14 In the case of interviews, focus groups, and participant observations the collected data is analyzed with a hybrid deductive and inductive coding approach (Fereday & Muir-Cochrane 2006). Initially, a coding schema is defined based on the preliminary characteristics, attributes, relationships and criteria for assessment from the previous step. More specifically, codes of the first level are defined as (a) the preliminary system characteristics and (b) the system's behavior (or performance). The codes of the second level for (a) are defined as the attributes and relationships, whereas the codes of the second level for (b) are defined as the criteria for assessment. During the coding process, not only instances of the pre-defined codes are identified, but also an open (inductive) coding approach is adopted to find new system characteristics, attributes, relationships, desired system's behavior and criteria for assessment.

3.15 In the case of archival data or documents, the summative content analysis approach is adopted (Hsieh & Shannon 2005). This approach is divided in two levels, namely manifest and latent. The manifest level entails finding in the archival data occurrences of the codes associated with the preliminary characteristics, attributes, relationships and criteria for assessment. At this stage, new characteristics, attributes, relationships and criteria for assessment may also be found through open coding. Next, the latent level focuses on analyzing the context in which the code occurrences were found to study and revise their meaning. In the process, further instances of the codes may be found, and also new codes may be introduced. Typically, an iterative process is required between the manifest and latent levels to determine how well the meaning extrapolated from given contexts fits that associated with the codes and solve potential conflicts. When such conflicts occur, they can be addressed through an in depth inspection of their meaning, leading to the definition of new meanings associated with the code or to new codes which reconcile the conflict. The content analysis can be considered completed when no new codes or conflicts are found in the data.

3.16 The results of this step are refined and validated behavioral and structural requirements. Further, the policy problem identified earlier may also be validated and refined together with the requirements. For instance, a new promising direction may be found from the data which may require to adjust the policy problem and align it with the new or modified requirements . Given the modifications to the requirements and policy problem,

new fields of literature may be found to be relevant which were not considered in the first step. When literature confirms the finding from the case study this provides further confidence in the findings. Further, when literature is in contrast with the findings from the case study, this is an opportunity to further probe into the nature of this contrast and bring a deeper insight into both the literature and the requirements (Eisenhardt 1989). As such, some iteration between case study research, literature review and requirements design is likely to be necessary.

Conceptual framework design

3.17 The design process of the conceptual framework is based on the refined and validated requirements from the previous step. More specifically, the structural and behavioral requirements are considered. Structural requirements provide the list of the system characteristics, attributes, and relationships to be included in the conceptual framework. Each system characteristic is considered as an independent conceptual framework component, with its own attributes and relationships. When the relationships found between the system's characteristics are vertical, such as those of the type "is a part of", "can have one or more" or "contains", then the corresponding characteristics are organized hierarchically. If the relationships among the characteristics are horizontal, such as those of the type "interacts with", "causes", "performs" and "affects", these characteristics are linked with an arrow labeled with the corresponding relationship. Additionally, the behavioral requirements are used to capture the systems performance through the criteria for assessment.

Phase 2: Model Development

3.18 It is good practice to set a clear modelling purpose, as the way a model is developed, justified and also scrutinized by the scientific community depends on its purpose (Boero & Squazzoni 2005; Edmonds et al. 2019). Therefore, the model development process in this article takes different forms depending on the purpose of the model. More specifically, a distinction is made between models with an empirical or a theoretical purpose[4,5]. Such a distinction affects the way the conceptual framework and generic models are used in the development process to analyze an existing system or to construct an abstract one (cf. Section "Conceptual framework use and composition" above).

Model development steps

3.19 Several methodologies have been proposed in the literature for developing and describing ABMs. In this article, the approach proposed in Nikolic & Ghorbani (2011) and Dam et al. (2013) is chosen and extended to include the use of (a) the conceptual framework from the previous phase and (b) generic models such as the Generic Agent Model (GAM) (Brazier et al. 2000) to guide the model development process. However, other ABM development methodologies for instance based on the ODD protocol (Grimm et al. 2006) could be possibly extended in a similar manner.

3.20 The resulting approach involves the following *iterative* model development steps: Conceptual Framework Application, Problem Formulation, System Identification and Composition, Model Concept Formalization, Model Narrative Development, Software Implementation, Model Evaluation, and Abstraction to Generic Model. This last step is optional. In the following sections, each step is described briefly, stressing how the conceptual framework and generic models are used to guide model development for both empirical and theoretical models. Figure 4 shows an overview of how the conceptual framework and generic models support the model development steps and their outcomes.
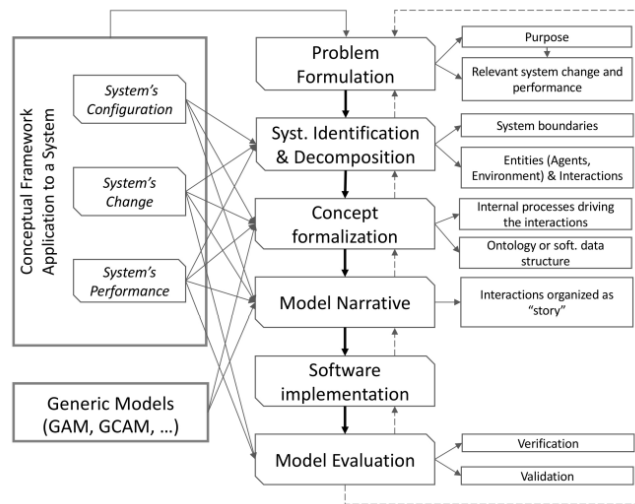
Figure 4: Use of the conceptual framework and generic models (left) to support the steps for the development of an Agent-Based Model (center) with details on the result of each model development step (right). The dashed lines show possible iterations among the different steps of the model development process.

Conceptual framework application

3.21 As already mentioned in Section 3.3, the conceptual frameworks developed with this methodology can be applied both to construct an abstract systems to be investigated via a theoretical ABM, or to analyze an existing system so that the system's essential features can be captured with an empirical ABM. The modelling purpose defines *how* the conceptual framework is applied. Specifically, in the case of theoretical models, the conceptual framework can be used to provide an inventory of relevant system' characteristics, attributes, relationships and criteria for assessment that the researchers may want to consider in the model. With regards to empirical models, the conceptual framework provides the means to analyze the system's configuration, change and performance (see Section "Conceptual framework use and composition") that inform the following model development steps. The use of the conceptual framework application for each of the model development steps shown in Figure 4 is discussed in detail in the sections hereinafter.

Problem formulation

3.22 The problem formulation entails making decisions about (a) the modelling purpose and (b) the system's performance and change of interest to be captured respectively by criteria for assessment[6] and other indicators designed to study possible changes in the configuration of the system. These choices are made based on the policy problem and the application of the conceptual framework as shown in the following.

3.23 In the case of empirical models, the choice of a modelling purpose and of the system performance and change of interest is guided by the results of the conceptual framework application to the case study (i.e., the system's configuration, change, and performance) with respect to the considered policy problem. Empirical models can be employed to provide a description of the current configuration and dynamics of a given system based on the knowledge gathered through the system's analysis, literature, and existing models. While descriptive models do not aim to exactly reproduce or explain specific system's performance or change, they can be a first step for the development of future models aimed at supporting policy formulation and evaluation for the given case and considered policy problem (see Watts et al. 2019 as an example of description). In other cases, the analysis of system's performance uncovers that the system performs poorly or particularly well in terms of specific criteria for assessment or that the configuration of the system changes in particular and unexpected ways. As such, the focus may be placed on providing explanations with respect to the mechanisms that lead to such system performance or change. The results of these explanations can inform policy formulation and evaluation (see for instance Adam & Gaudou 2017). Finally, empirical models may be chosen with the purpose of exploring the implications of future policy interventions e.g., aimed at changing the configuration of the system to improve the system's performance (Edmonds & ní Aodha 2019). The choice of relevant indicators of system's performance and/or change follows from modelling purpose.

3.24 Theoretical models can be developed to abstract from the context of a given case study and to capture a range of systems (Boero & Squazzoni 2005). In this case, the researchers can choose a modelling purpose within the broader scope of achieving a desired system's performance via policy design. Theoretical models can, for instance, have the purpose of illustrating or exploring relationships between given system characteristics or policies and the resulting system's change or performance. The results of such modelling efforts can e.g., support the formulation of hypotheses (to be tested empirically) concerning (a) the success of specific policy interventions in reaching the desired system's performance or (b) explanations for particular system's change that can be relevant for policy design (Comfort et al. 2004; Zagorecki et al. 2009; Altay & Pal 2013; Bateman & Gralla 2018). Relevant indicators of system's performance and change are chosen on the basis of the modelling purpose. In the case of system's performance, the researcher may decide among the assessment criteria provided in the conceptual framework.

System identification and composition

3.25 System identification involves defining the boundaries of the considered system. System composition consists of capturing the relevant system's configuration and change for the chosen modelling policy problem and modelling purpose. More specifically, the key entities (agents and the environment in which they are embedded) and their interactions to be captured in the model are defined conceptually at this stage.

3.26 In the case of empirical models, the boundaries of the system can be those of the case study to which the conceptual framework was applied. However, the considered system can be narrowed down to a specific area. The system composition is derived from the analysis of configuration and change obtained through the conceptual framework application to the identified system.

3.27 With regards to theoretical models, system identification and composition are meant to capture an abstract system, rather than a specific case study. The conceptual framework can support system composition by providing an inventory of system characteristics (e.g., key entities), attributes and relationships that are relevant for the considered policy problem and can be instanced and included in the abstract system's configuration and change.

Model concept formalization

3.28 In this step, the system composition is formalized in a format that can be translated into software. The entities that will become agents in the model are formally defined, together with the tasks (or activities) they carry out, and their properties and state variables. Also the environment in which the agents are placed is considered as an entity characterized by tasks, properties, and states. The entities found are organized hierarchically from general classes including common properties, states, and tasks, to the entities representing the agents and environment actually considered in the model. A concept formalization can be implemented directly as a software data structure or as an ontology (which is then translated into a software data structure).

3.29 Next, the concept formalization of a model requires capturing not only the agents and the tasks (or activities) that they carry out in interaction with other agents or the environment, but also the internal processes that occur within the agents that enable these interactions. While the system composition obtained at the previous step includes the key entities and their interactions that can be translated into the corresponding agents into the model, it does not provide the level of detail required to capture the agents' internal processes. To maintain a rigorous approach, another framework focused on the agents' internal processes is required to structure the implementation of a system's composition into a concept formalization. The design of the agents' internal processes may also require further analysis of the qualitative data through the lens of a framework capturing the agents' internal processes.

3.30 Agent architectures such as BDI (Rao & Georgeff 1991) provide frameworks focused on capturing the agents' internal processes. However, agent architectures are often described in a qualitative manner, making their implementation in the model formalization open to different interpretations and, hence, difficult to reproduce. Generic models such as the Generic Agent Model (GAM) can provide a formalized understanding the internal processes occurring within the agents e.g., according to a particular agent architecture. *The advantage of using generic models is that different agent architectures can be captured and compared through the same formal language*[7]. For instance, several applications of GAM have been proposed that capture different agent architectures (e.g., normative, cooperative, or BDI) in a formalized and non-ambiguous way (Brazier et al. 2002) (cf.

2.5). These applications of GAM provide the model designer with a series of options in the way the agents' internal processes are designed and formalized through the selected generic model, given the key agents and their interactions captured by a particular system composition.

3.31 Developing a model concept formalization entails the following steps. First, a generic model is selected. GAM only assumes some of the general characteristics of an agent (Wooldridge & Jennings 1994, 1995) and could be selected in most cases if its general assumptions are shared by the researchers who decide to use it. However, more specific generic models capturing particular agent architectures may also be selected depending on the chosen modelling purpose and the adopted modelling strategy as discussed by Edmonds et al. (2019). Second, the selected generic model is used to interpret and structure a) the system composition from the previous step, (b) the results of the framework application (analysis of system's configuration, change, and performance), (c) previously or newly collected qualitative data[8] to develop a model's concept formalization. This process occurs in two iterative steps: (a) specialization of the generic model's process composition and (b) instantiation of the information types included in the generic model's knowledge composition (cf. P. 2.5).

3.32 The *specialization* of a generic model's process composition based on the system composition is carried out by individuating a task hierarchy, task inputs and outputs, information exchange among tasks, and the delegation of the tasks[9]. First, the task hierarchy is refined based on the system composition. The activities and interactions of the entities found in the system's composition are assigned as sub-tasks to the matching categories of tasks included in the selected generic model. For instance, tasks that set the goals of an agent (e.g., by assuming or changing roles) or carry out decision making activities, are assigned to the agent's Own Process Control (OPC), whereas tasks that involve interaction with the world or environment such as moving through the landscape or collecting information from the surroundings are assigned to the Management of World Interaction (MWI) task (cf. Appendix).

3.33 Second, *additional sub-tasks are added to the task hierarchy which provide the internal processes required to enable an agent's activities and interactions*. These sub-tasks are designed considering the sub-processes that are required for the agent to make choices with respect to some of the activities found in the system composition e.g., selection of information exchange partners. In the case of empirical models, the chosen generic model can also be used to further structure and interpret the previously collected data to guide the design of the agent's internal processes. The researchers can also collect further data specifically meant to guide the design of the agent's internal processes on the basis of the generic model. Such data can then be structured and interpreted through the lens of the generic model to confirm sub-tasks designed based on the system composition, find additional sub-tasks, or verify if any of the task included in the generic model are actually (not) required. In the case of theoretical models, such sub-tasks are designed purely based on the previously obtained system composition, literature, and the tasks included in the chosen generic model.

3.34 Third, the *input and output* information of the sub-tasks is defined respectively by considering the information required by the task and the information that the task can provide to other tasks.

3.35 Fourth, the *information exchange* among tasks is defined by adding information links across the tasks based on the input and output information. Some of the links required may already be provided by the generic model among the tasks included in such model (e.g., OPC and MWI). In other cases, some of the links included in GAM may not be necessary for the considered application and as such they may be excluded from the concept formalization.

3.36 Fifth, the tasks are *delegated* (assigned) to the entities (different agent types and the environment) found in the system composition phase.

3.37 The *instantiation* of the information types entails finding new refined information types that are instances of the generic information types included in the generic model. In the case of GAM, the generic information types are world information, agent information, agent identification, domain actions, and domain agent characteristics. Both for theoretical and empirical models, the generic information types can be instantiated based on the internal processes designed in the specialization of the process composition (e.g., when the new sub-tasks introduced require information types that are more refined than the generic ones). Additional information types can be instantiated by analyzing the results of the framework application (systems configuration, change and behavior), literature, and existing models through the lens of the generic information types. In the case of empirical models, the previously collected qualitative data can also be analyzed through the lens of the generic information types to find new instances. If the data collected does not provide sufficient evidence, the researcher may decide to collect further data aimed at verifying, refining and expanding upon the information types found.

3.38 Given the specialization of the process composition and instantiation of the knowledge composition obtained above, the agents, their tasks, and information types can be defined in a software implementable way (e.g., via UML diagrams).

3.39 At this stage, the researcher can rely on quantitative data (e.g., demographics) to quantify the number and spatial locations of the instances of the agents and environment captured in the concept formalization.

Model narrative development

3.40 At this stage, all the tasks that agents carry out including their interactions with other agents and with the environment are organized into a narrative. First, the specialization of the task control knowledge is carried out. Task control knowledge dictates when particular tasks are executed by the agents and by tasks that have sub-tasks (cf. Appendix). Secondly, the instantiation of the knowledge bases is carried out in order to define the rules followed by the agents when particular tasks are executed. Specifically, knowledge bases are specified for the tasks that are primitive (i.e., that do not have sub-tasks).

3.41 In the case of empirical models, the selected generic model provides a framework to further structure and interpret the collected data, the results of the conceptual framework application, the system composition, and concept formalization to specialize task control knowledge and instantiate knowledge bases. At this stage, further data may be collected to specifically inform the design of task control knowledge and knowledge bases. In the case of theoretical models, the specialization of task control knowledge and instantiation of knowledge bases is based on the system composition and on the chosen generic model.

3.42 Given the specialized task control knowledge and instantiated knowledge bases it is possible to *assemble a narrative* to be captured in the model e.g., via pseudo-code or through a flow chart. The task control knowledge determines the order of execution of the tasks and their sub-tasks given particular triggering "events" or knowledge states (e.g., incoming information to the task or the successful completion of another task, cf. Appendix). However, the order of occurrence of the events that trigger the tasks is a deliberate choice of the modeller. As such, a number of narratives can be assembled given the same task control knowledge depending on the considered order of triggering events. The knowledge bases determine the rules followed by the agents within the most elementary tasks (primitive tasks i.e., tasks which do not have sub-tasks).

Software implementation

3.43 In this step, the model conceptualization and narrative are implemented in a modelling environment such as NetLogo, Repast Symphony or GAMA. A detailed review to guide the choice of a modelling environment is provided by Abar et al. (2017).

Model evaluation

3.44 Model evaluation is an activity that occurs throughout the development of a model. Evaluation can take different forms including verification and validation. Verification focuses on assessing whether the model corresponds to the intentions of the modeller. Validation is concerned with evaluating if the model corresponds to the reality it aims to capture (Nikolic & Ghorbani 2011; Calder et al. 2018). Depending on the modelling purpose, validation and verification assume different relative importance (Edmonds et al. 2019). Theoretical models are not directly connected to a particular case study. As such, there is a stronger focus on verification rather than validation. Conversely, empirical models aim to capture a given case study, and therefore typically require a stronger emphasis on validation. Descriptive empirical models do not aim to reproduce a system's performance or change but only to combine knowledge gathered through the case study with previously available knowledge and models. Therefore, such models require solely a validation in terms of their model conceptualization and narrative (structural validation). Other empirical models that aim at reproducing the system performance or change need to be validated not only in terms of their structure, but also with regards to their ability to reproduce the system's performance or change. In such cases, the results of the analysis of system performance and/or change from the conceptual framework application (see Section "Conceptual framework use and composition") can be used here as the output to be matched by the model.

Abstraction of a new generic model

3.45 When a generic model is used in the concept formalization and for the development of a model narrative, new tasks and knowledge structures may be introduced which can be abstracted and constitute a new generic model. The resulting generic model can then be re-used in similar situations. For instance, the application

of GAM to develop a model for distributed project coordination in engineering consultancies resulted in the definition of a new generic model, namely the Generic Cooperative Agent Model or GCAM (Brazier et al. 1996). GCAM may be re-used for any situation in which distributed and cooperative project management is of interest, assuming joint intentions, limited time resources, and non-urgent problems. In the case of ABMs designed to target a particular policy problem in a given system, abstraction entails the design of a new generic model that applies to the considered class of policy problems. As such, this model can then be re-used to develop models that aim at addressing the same policy problem in different systems.

Iterative model and conceptual framework development

3.46   The model development phase is likely to produce new knowledge regarding relevant systems characteristics, attributes, relationships or system's performance that are not included in the conceptual framework designed in phase one. As such, this knowledge can be incorporated back in the conceptual framework for future use.

Methodology Application: A Case Study on Disaster Information Management in Jakarta

4.1   In this section, a case study of disaster Information Management (IM) in Jakarta, Indonesia is used to illustrate the use of the methodology. The following sections provide information on the case study, and show how a conceptual framework was designed based on the case study (phase 1), and how this conceptual framework together with a generic model was used to develop an empirical model (phase 2).

Case study

4.2   When disasters such as floods and storms hit, both formal and informal professional and volunteer organizations and communities need to adapt to ever-changing and often unexpected conditions (Comes et al. 2020). The ability of affected communities and disaster responders alike to self-organize, coordinate and respond to the situation strongly relies on the timeliness and quality of the information available (Nespeca et al. 2020; van de Walle & Comes 2015). Yet, with the dynamically evolving situation, the roles and information needs of the actors continually change (Turoff et al. 2004; Meesters et al. 2019). Designing for coordination and self-organization in disasters thus mandates the design of information management (IM) policies to ensure that information of good quality reaches the actors who need it when they need it. Such policies need to take into account the socio-technical nature of disaster response systems, as the way information is collectively managed often depends on the interplay between human behavior and the use of technology (e.g., mobile phones and social media) (Starbird & Palen 2011; Silver & Matthews 2017). The authors in this case were specifically interested in the design of bottom-up IM policies, i.e., policies that take into account and build on the decentralized and collective nature of disaster IM.

4.3   Jakarta is a critical case for bottom-up disaster IM as (a) the city is affected by frequent flooding due to its rapid subsidence and urbanization processes, (b) because of such frequent floods, many bottom-up IM initiatives have been initiated, often aided by social media and messaging apps (van Voorst 2016), and (c) the city presents a great diversity of actors, groups, roles and activities relevant for the development of a conceptual framework than can capture such diversity. Another reason for choosing Jakarta was that at the time of data collection (in 2018) many international organizations were in the city due to the humanitarian response to the Sulawesi Earthquake. This provided the opportunity to interview their representatives in person[10]. More information on the case study can be found in Nespeca et al. (2020).

Phase 1: Conceptual framework development

4.4   This section summarizes how the conceptual framework was developed through the steps shown in Figure 5.

Literature review

4.5   The review of relevant literature and existing ABMs was narrative. Specifically, literature from the field of multi-actor systems, self-organization and information management in crisis response was considered. Additionally,

a review of the existing ABMs on disaster IM was carried out. This lead to the identification of the *policy problem* as the design of bottom-up IM policies that can support both coordination and self-organization in disaster response by satisfying the continually shifting information needs of individual actors. As such, the *unit of analysis* chosen for the conceptual framework is that of an individual person (or actor). A list of relevant *system's features* was derived from the current literature and existing ABMs[11] as shown in the following (Nespeca et al. 2020):

- Multi-Actor Systems:

    - Actors: skills, experience & knowledge, preferences;
    - Roles: responsibilities with related rules & norms, capabilities, domain of expertise, status (formal or informal), information (needs and access);
    - Groups with their structures & networks;
    - Environment;
    - Operations.

- Self-organization: self-organization and coordination activities, namely Role & Structural change and Networking (building new connections, and establishing or joining groups)

- Information Management:

    - Information Management Activities: collecting, evaluating, processing, sharing & storing information;
    - Information Characteristics: Information quality (Relevance, Timeliness, Accessibility, Interoperability, Reliability, and Verifiability), and (cognitive) load;

Requirements design

4.6 Based on the results from the previous step, the system's mission and requirements were identified. Considering the chosen policy problem and the information characteristics derived from the information management literature (cf. previous section), the mission for bottom-up disaster IM policies was defined as in the following:

Mission for bottom-up disaster IM policies: "*to provide relevant, reliable and verifiable information to the actors who need it, when they need it in an accessible manner.*" (Nespeca et al. 2020).

Functional requirements were designed as the functions that the system has to provide in order to achieve the mission, taking into account the relevant system's features. Four examples of functional requirements are provided hereinafter.

Diversity (functional): "*the system has to cater for the great diversity of actors, roles and groups involved in and affected by the disaster, and to consider the way this diversity affects the activities carried out by the actors.*" (Nespeca et al. 2020).

Relevance (functional): "*irrelevant information contributes to overload. The actors should therefore receive information that matches their intended use.*" (Nespeca et al. 2020).

Timeliness (functional): "*due to the dynamic nature of disaster response, information received and made available for the actors should be kept up to date to keep decision making and coordination attached to reality.*" (Nespeca et al. 2020).

Accessibility (functional): "*information shared with the actors should be accessible for them in terms of language and format.*" (Nespeca et al. 2020).

The behavioral requirements found were designed as the desired behavior of the systems associated with the fulfillment of the functional requirements. Three examples of behavioral requirements associated with the three previously-listed functional requirements are shown in the following.

Relevance (behavioral): "*the degree to which the information that reaches the actors matches their intended use.*" (Nespeca et al. 2020).

Timeliness (behavioral): "*the degree to which the information received by actors is up to date.*" (Nespeca et al. 2020).

Accessibility (behavioral): "*the degree to which information is provided in such a way that the actor can easily use its content.*" (Nespeca et al. 2020).

Finally, structural requirements were designed as the system's components that need to be put in place to fulfill the behavioral requirements. The structural requirements designed included the following three examples.

Distinction between actors and roles (structural): "*Actors can change roles and assume additional ones. The way roles are carried out depends on the personal attributes of the actors who assume them*" (Nespeca et al. 2020);

Actors (structural): "*Actors are characterized by their Skills, Experience, Knowledge, and Preferences (e.g., willingness to share information)*" (Nespeca et al. 2020);

Roles (structural): "*Roles are characterized by the Responsibilities and Capabilities to carry out specific activities, the Information needs (characterized by Relevance, Timeliness, Accessibility, Reliability and Verifiability) and access, the domain of expertise, and status (officially mandated or not)*" (Nespeca et al. 2020);

Requirement validation and refinement based on the case study of Jakarta

4.7   The data collection techniques chosen were interviews and focus groups. The protocols for both interviews and focus groups were designed to capture the requirements in four stages: biographical, situations, information and obtaining information. Each stage was designed to elicit knowledge from the participants with regards to one or more of the requirements. The protocol stages, their contents, and targeted requirements are shown in Table 1.

Stage	Contents	Targeted requirements
Stage 1: Biographical	Introduction, Biographical Information & Role of the Interviewee	Actors, Groups, Roles (formal), Environment
Stage 2: Situations	Selecting a specific (disruptive) event that triggered the need for information.	Environment, Activities
Stage 3: Information	Information needed to address the situations, as well as the information available that could be shared.	Information Quality (Relevance, Timeliness, and Accessibility) and Load.
Stage 4: Obtaining Information	How was the information obtained? From what sources and which activities, methods, and tools were involved.	Activities, Groups, Roles (formal and informal), Operations, and Environment

Table 1: Stages of the interview protocol, their contents, and requirements they target. Adjusted from (Nespeca et al. 2020).

4.8   The data collection plan was designed on the basis of an exploratory interview with a government official from Jakarta carried out in preparation for the research on-site. Additional participants were identified through snowballing during the data collection. In total, 9 semi-structured interviews and 3 focus groups were carried out. Altogether, 25 participants were involved in the data collection, ranging from the information managers of national and international governmental and non-governmental organizations, to the members of highly affected communities in the city. Specifically, the Marunda and Kampung Melayu communities were selected.

4.9   The recordings from the interviews and focus groups were then transcribed and analyzed through a hybrid deductive and inductive coding approach. First, an initial coding scheme was designed to capture the requirements designed based on literature (cf. P. 4.6). This approach was meant to validate the requirements designed (deductive approach). Further, while thematically analyzing the interviews, also open coding was carried out in parallel to refine the existing requirements and possibly find new ones (inductive approach). Sample quotes were selected from the interviews and focus groups to show supporting evidence for the validated and refined list of requirements.

4.10   The findings showed that no additional requirements were individuated through open coding and that most of the requirements from literature were confirmed. However, some differences were found between the literature and the findings from the data with regards to the definitions associated with some of the requirements.

In such cases, the definitions from literature were updated to match those found from the data. Table 2 shows an excerpt of the findings from the field by Nespeca et al. (2020) with respect to structural and behavioral requirements and their sub-requirements, definitions, code count, and sample quotes.

Table 2: Excerpt of results table by Nespeca et al. (2020): list of requirements, sub-requirements, code count and sample quotes. The sub-requirements which definition changed via open coding are underlined.

Req.	Sub-requirements	Num. Codes	Sample Quotes	Quote ID
Actors (structural)	Skills: ability to carry out activities within a given time. Skills can be transferable across roles.	30	'I think it was a kind of a natural progression to then take some of that work and apply it (…) it just made a lot of sense. Because the skills were transferable' UN-OCHA Community Liaison Officer	1
	Knowledge: non-procedural knowledge from info gathered during disasters or from education.	30	'if the height in Depok is three meters there will be no flood in here' Kampung Melayu Community Leader	2
Roles (structural)	Responsibilities, Rules and Norms an actor should comply with given his/her role.	84	'if somebody notices that the sea level rises, they directly inform it by sending text through WhatsApp' Marunda Community Member	3
	Status (Formal or Informal): availability of a mandate or not.	35	'So yes, the government is helping us, but more than that communities (…) and also NGOs'. Kampung Melayu Community Member	4
	Information: actors have information needs and access because of the roles they assume.	29	'We can always provide you with information for example on assessment registry. What kind of assessment has been done, where is it, what sort of sector did they do the assessment'. UN-OCHA Information Management Officer	5
Activities (Structural)	Networking: build new connections and create new groups.	40	'sometimes after the meeting I need to chase people that have so much information (…). after the meeting I approach them to talk'. UN-OCHA Information Management Officer	6
	Role & Structural Change: assume roles or change structural relationships among them.	13	'I was becoming a reference for everyone for asking about mailing lists, who is working in certain area or what sort of maps are available (…) So that's the role that I have done.' UN-OCHA Information Management Officer	7
	Information Management: Collect, Evaluate, Process and Share info.	132	'I check information updates through Twitter. If, there is still no electricity I stay at home'. Other Community Member	8

Req.	Sub-requirements	Num. Codes	Sample Quotes	Quote ID
Information quality (behavioral)	Relevance: the degree to which the information received by the actors matches their intended use, required level of aggregation, and spatial location.	50	'When you open the map, you might not click on every point. But you would immediately have a sense of the areas that are flooded, enabling you to make decisions about areas to avoid.'. Petabencana Community Liaison Officer	9
	Timeliness: the degree to which information reaches the actors before the expiration of their information needs.	18	'Sometimes we don't know whenever the flood finishes and then we can clean up our house. Then suddenly it floods again. We don't have any information'. Kampung Melayu Community Member	10
	Accessibility: the degree to which information is provided in such a way that the actor can easily use its content.	26	'In a lot of the communities I've worked with there's no literacy and that's why face to face and oral communication is much more effective' UN-OCHA Community Liaison Officer	11

4.11 Two of the behavioral requirements, namely relevance and timeliness had to be updated as their definitions from literature did not match the views of the case study participants (cf. Table 2). Relevance is intended as "*the degree to which information by the actors matches their intended use*". The case study showed that such a definition is valid. However, two additional factors were found that determine the relevance of information. Such factors are (a) the level of aggregation of information (e.g., summarized for a region or, point by point) and (b) the spatial location to which the information refers. For example, when asked about the way information is displayed in their crowdsourcing platform, the Community Liaison officer from the NGO Petabencana mentioned how information is provided as a summary over an area in order to give an overview. The user is also able to zoom to areas of interest in order to get the information required at specific locations (cf. Quote 9, Table 2). As such, the definition of Relevance was extended to include the aggregation and spatial location of information as shown in Table 2.

4.12 In the case of Timeliness, the definition was completely revisited based on the findings from the case study. The literature points to timeliness as "*the degree to which information is up to date*" as designed in P. 4.6 based on e.g., Van de Walle & Comes (2015); Bharosa & Janssen (2015). Such definition is independent from the context in which the receiver is placed and only depends on the currency of information. However, the data analysis revealed how the participants mentioned timeliness as the need to obtain information by the time they require it given the context in which they are placed. For instance, when asked about the way information quality could be improved, one of the community members from Kampung Melayu shared that in some cases a second wave of flooding can occur after they already started their recovery activities (clearing their house from debris) (cf. Quote 10, Table 2). In this context, information concerning a second flood wave would be timely if provided by the time community members start their recovery activities. To match this perspective, the definition of timeliness was updated as shown in Table 2.

Conceptual framework design

4.13 As introduced in the section "Conceptual framework use and composition", a conceptual framework that enables the analysis of an existing system or construct an abstract one to inform the development of ABMs for policy support should include the system's characteristics and their attributes, relationships among characteristics, and criteria for assessment (cf. Figure 2). This section shows how a conceptual framework with its system's characteristics, attributes, relationships, and criteria for assessment was designed based on the refined and validated list of structural and behavioral requirements obtained in the previous section. Firstly, structural requirements provided the *characteristics* of the system to be captured in the conceptual framework, and the sub requirements provided the attributes of such characteristics. Each characteristics was included as an

independent component of the conceptual framework with the associated attributes. For instance, "actors" and "roles" are characteristics that were included as conceptual framework components. The attributes of the "actors" characteristic include "skills" and "knowledge" given these were found to be sub-requirements of "actors" (cf. Table 2). Secondly, structural requirements also provided the *relationships* among characteristics. When the relationships were vertical as in the case of "belongs to" or "can have one or more" then the characteristics were organized hierarchically. For instance, actors and roles have a vertical relationship dictated by the "distinction between actors and roles" structural requirement according to which actors "can have one or more" roles. Conversely, when the relationships were horizontal as in "affects" or "performs", then such relationships were linked with an arrow and labelled with the corresponding relationship. For example, the actor's activities may lead to the emergence of new groups, and coordination structures and networks for information sharing[12]. As such, actors' activities "affect" groups. Thirdly, behavioral requirements provided the *criteria for assessment* necessary to evaluate the extent to which the system performs according to the desired behavior or performance. Specifically the criteria for assessment are information relevance, timeliness, accessibility, reliability, verifiability and load. Figure 5 shows the resulting conceptual framework.
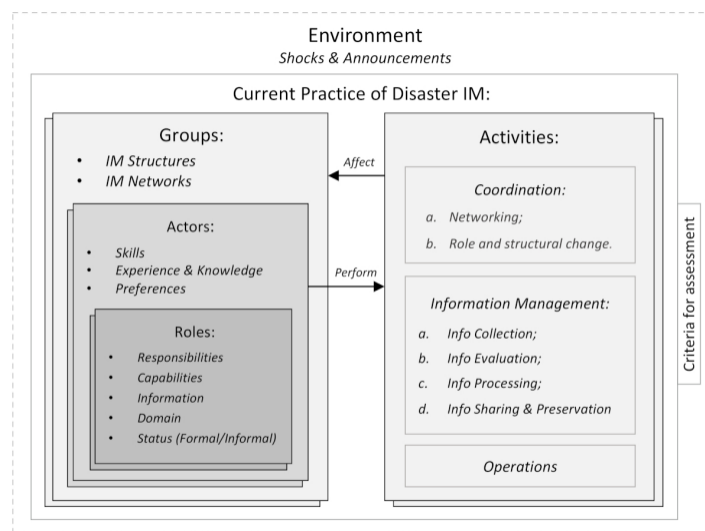

Figure 5: Developed Conceptual Framework, adapted from (Nespeca et al. 2020). IM = Information Management

4.14   This conceptual framework has the threefold purpose of providing the means to analyze (a) the current practice of disaster IM in a case study representing the way information is collectively managed in a system (system's configuration), (b) the way such practice changes through self-organized bottom-up processes (system's change), and (c) the extent to which the current practice supports bottom-up disaster IM as measured by the criteria for assessment (system's performance). This conceptual framework was further validated by Slingerland et al. (2022) who applied it to the case of Information Management during the COVID-19 pandemic response in the neighbourhood of Bospolder-Tussendijken (BoTu) in Rotterdam, The Netherlands. Specifically, Slingerland et al. (2022) used the conceptual framework to analyze the system's configuration, change and performance before and during the pandemic. Such an analysis enabled the ex-post evaluation of resilience policies implemented prior to the pandemic in BoTu from an information management perspective. The study resulted in a list of implications for the design of policies aimed at fostering crisis resilience in urban neighbourhoods.

Phase 2: Model Development

4.15   In this phase, the conceptual framework was used to develop an empirical model. The following sections explain the model development process in detail.

Conceptual Framework Application

4.16   Even though the specific purpose of the model was not clear yet at this stage, the authors already decided to develop an empirical model. As such, the conceptual framework was applied to the case study. The authors relied on the previously collected data that had already been used during the conceptual framework development

phase. The data analysis consisted in going back to the codes assigned during the requirements validation to find instances of the characteristics, attributes, relationships, and criteria for assessment found in the conceptual framework. This allowed the authors to identify the system's configuration (current practice of Disaster IM), change (alteration of the practice) and performance (in terms of the criteria for assessment) as briefly shown in the following. For a thorough discussion please refer to (Nespeca et al. 2020, pp. 6-10).

4.17   The analysis of **system's configuration** enabled to identify the instances of the characteristics and attributes that represent the current practice of disaster information management. The characteristics identified were the key actors and roles they assume, groups they belong to, IM structures and networks through which they share information, activities they carry out, and the environmental factors that play a role in disaster IM (shocks and announcements) (cf. FigURE 5). For instance, the data showed how the community members in Marunda rely on their informal connections (IM networks) to exchange information e.g., aided by a WhatsApp group (cf. quote 3 in Table 2).

4.18   During the analysis of system's configuration, one system characteristic was found that had not been included in the conceptual framework, namely that of objects. Objects are any non-human entities that can support IM and coordination activities of the actors both within and across groups. Examples of objects found in the considered case study are social media (cf. Quotes 8, Table 2) and a WhatsApp group used to share and receive flood warnings in the Marunda community (cf. Quote 3, Table 2).

4.19   The analysis of **system's change** was carried out by analyzing the way the activities of the actors affect the configuration of the groups and the actors' roles (cf. relationship "affect" in Figure 5). The analysis showed how role change can occur not only because of a deliberate choice of the actors, but also through a self-organized process triggered by interactions among the actors. The Information Management Officer (IMO) from UN-OCHA observed how s/he assumed the role of an information exchange hub that would provide the information requested by other actors. However, this role change did not occur via a direct choice, but because of gradually increasing requests for information made by other actors (cf. Quote 7 in Table 2). When other actors realized that the s/he had access to information (cf. Quote 5 in Table 2) and knowledge on the type of information available, they gradually required more and more of the IMO's "services". As such, the IMO gradually assumed the (informal) role of information exchange hub across different organizations.

4.20   Concerning the analysis of **system's performance**, the performance of the system related to information reliability, verifiability, accessibility and load were found to be acceptable from the perspective of the case study participants. However, the timeliness and relevance criteria were found to be unsatisfactory. For example, one of interviewees mentioned how in some cases information concerning flood warnings was not received on time (cf. quote 10 in Table 2). Specifically, the analysis of performance showed that the relevance and timeliness of information were lacking especially for those information needs of which the actors were not aware (e.g., flood warnings). To capture the discrepancy in performance among information needs of which the actor were aware or not aware of, a distinction was drawn between latent and known information needs. Known information needs are those that the actors are aware of (e.g., an update on the current water level in a flooded area). As such, actors can search for the information they need. In the case of latent information needs this is not possible as the actors are not aware that they need information (e.g., in the case of a flood early warning).

Problem formulation

4.21   A model with a descriptive purpose was chosen to capture some of the main characteristics and dynamics of the current practice of disaster IM in the considered system with a focus on bottom-up practices. This is a first step in building a simulation environment that can be used for designing and evaluating bottom-up IM policies aimed at supporting self-organization and coordination in the considered system.

4.22   Given the findings of the analysis of system's performance (cf. P. 4.20), information relevance and timeliness were chosen as the assessment criteria representing the *relevant system's performance* to be captured in the model output. Further, the analysis of system's performance highlighted the need to distinguish between latent and known information needs. As such, the model output includes indicators of information relevance and timeliness for both latent and known information needs.

System identification and composition

4.23   The purpose of the model is to capture disaster information management practices within a community, and its interactions with other relevant actors from a bottom-up perspective. The Marunda community and specifically

its most affected community units RW 07, 10, and 11 were chosen as the conceptual framework application revealed that these units presented a rich array of bottom-up information management practices. Further, to capture other relevant actors the system boundary was extended to include also the governmental and non-governmental organizations and groups that (may) exchange information with the community.

4.24 The system composition was designed based on the results of the analysis of system's configuration and change refined and focused within the chosen system's boundaries (narrower compared to those of the system considered for the initial conceptual framework application). As such, only the system's characteristics, their attributes and relationships found within the system boundaries were accounted for. For instance, only deliberate role change was considered for the Marunda community, as no evidence was found that emergent role change occurs within the chosen system's boundary (cf. P. 4.19). Deliberate role change can occur when community members become aware of a given shock and decide to change their to role to that of responder. Through the analysis, information was also found on the number of the actors in the considered community units (RWs) and their administrative subdivisions or neighbourhood units (RTs).

4.25 The system composition also included modelling choices informed by literature and previous models e.g., regarding the level of abstraction at which particular system's characteristics needed to be captured in the model. For instance, according to the conceptual framework, the environment generates disruptive events (shocks and announcements). *Shocks* represent the cascading effects that occur in the disaster-affected area where the community members are located (Meijering 2019) (e.g., community members are stuck on their roofs without food or water). *Announcements* represent information produced outside the disaster affected area (e.g., a flood early warning) (Watts et al. 2019; Watts 2019). Capturing shock and announcements is key as they provide the information that the actors need. However, while shock and announcements could have been captured in a more detailed way e.g., by distinguishing among different types of announcement and shocks found in the data, this was not considered relevant for the model. As such, only the two generic types of information, namely shocks and announcements were included in the ABM. Such a choice was based on previously existing disaster information management ABMs that consider only one generic type of information e.g., Altay & Pal (2013), Zagorecki et al. (2009), Meijering (2019). Further, the number of events occurring every day of simulation were conceptualized as *environmental turbulence* defined as the *"as the frequency with which new information is introduced into the environment"* (Hazy & Tivnan 2003). The frequency of shocks and announcements per day represent two parameters in the model capturing environmental turbulence.

4.26 Figure 6 shows the resulting system composition for the Marunda community [13] with the exception of the activities of actors, objects, and the environment which are shown in Table 3.
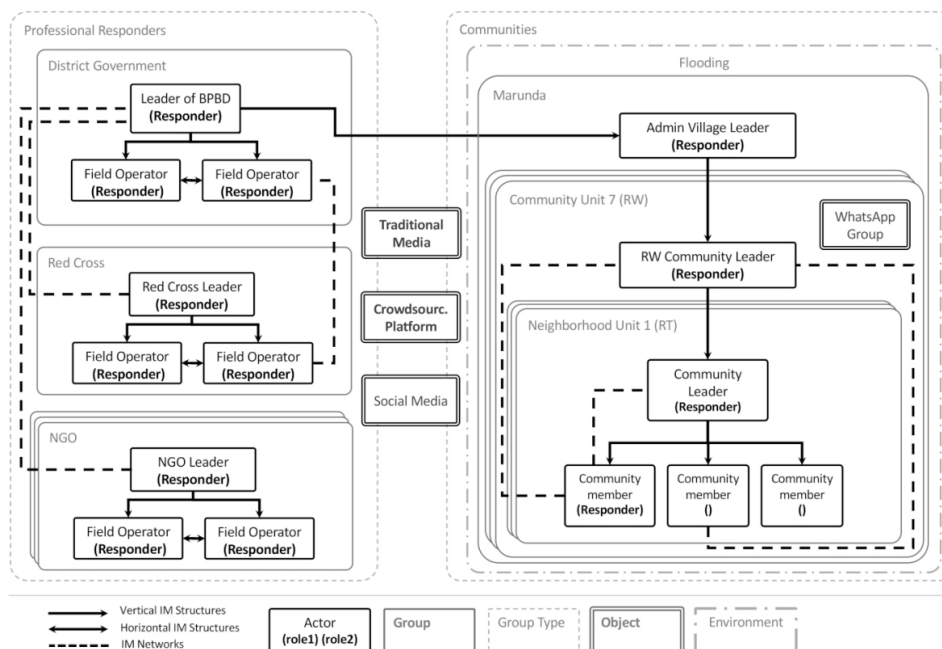

Figure 6: System Identification and Composition: configuration of the current practice of Disaster IM in Marunda, Jakarta. Adapted from Nespeca et al. (2020).

Entity	Category of Activity	Activity	Sub-activities
All Actors	Information Management	Collect info	from Environment, Actor or Object
		Receive info	from actor
		Evaluate Info	Check addressed info needs, track shocks found, track info gaps.
		Process Info	Choose from info pile
		Share & Preserve Info	Share to actor, Store in Object, Maintain Info needs
	Coordination	Networking	Manage Collection
	Operations	Move in field	-
		Become Affected	-
Actor: NGO & BNPB Leader	Coordination	Role & Structural Change	Deploy field Operator
			Join Response as group
Actor: Field Operator	Operations	Deploy to field	-
		Leave field	-
Actor: Community Leader	Information Management	Evaluate Info	Assess if required assistance
		Share & Preserve info	Request Assistance to BPBD
Actor: Community Member	Coordination	Role & Structural Change	Manage Responder Role
Object: Traditional Media	Information Management	Process Info	Filter Info
Object: Social Media	Information Management	Share & Preserve Info	Publish Noise (irrelevant info)
Environment	Generate Event	Generate Shocks	Release shocks in affected areas
		Generate Announcements	Release announcements (e.g., info from (flood) monitoring posts)

Table 3: System Identification and Composition: activities and sub-activities carried out by the actor and object entities according to the categories of activities introduced in the conceptual framework (cf. Figure 5), and by the environment.

Model concept formalization

4.27   At this stage, a generic model was used to guide the design of the concept formalization based on the system composition from the previous step. GAM was chosen as the generic model to be specialized an instantiated for this application. This choice was made as the more specialized generic models available in Brazier et al. (2002) were not found to the be suited for the particular case of disaster information management[14]. The concept formalization was obtained by specializing and instantiating GAM (cf. P. 2.5) through the knowledge captured in the system composition and by analyzing the results of the conceptual framework application through the lens of GAM. Specialization and instantiation proceeded in parallel through subsequent refinement and iterations. In the following, an example of specialization and instantiation of GAM based on the system composition is shown with respect to three of the generic model's abstracted tasks, namely "Own Process Control" (OPC), "World Interaction Management" (WIM), and "Maintenance of World Information" (MWI) (cf. the Appendix).

4.28   A *specialization* of GAM was developed to individuate a task hierarchy, tasks input and output, information exchange among tasks, and task delegation to different agent types. First, a task hierarchy was designed by specializing the abstract categories of tasks included in GAM (a) through the activities (i.e., agent's actions and interaction) found in the system composition (cf. Table 3) and (b) by individuating the internal processes required to support such activities. For instance, an activity from the system composition that was chosen to be a specialized task of the OPC was "Manage Responder Role". This task defines whether an actor who was not directly involved in crisis response decides to do so. The choice of specializing the OPC with "Manage Responder Role" was made as this activity can alter an actor's goal, a task that is typically associated with the OPC in GAM (cf. the Appendix). In other cases, further activities were added which were not directly found in the system composition, but that *provided the internal processes required to support the activities and interactions found in the system composition*. For instance, a task was missing among those found in the system composition to make decisions with respect to when and how to carry out the Information Management (IM) tasks "Collect info", "Process info", and "Share & preserve info" (cf. Table 3). To this end, a new task was introduced as a sub-task of the OPC called "Determine IM Actions". This task was further specialized with the sub-tasks "Determine IM focus" and "Prepare Information Collection". "Determine IM focus" decides whether the actor agent focuses on processing the received information [15] or on collecting new information. "Prepare Information Collection" assesses whether the actor actually collects information on the basis of the availability of known information needs (cf. P. 4.20). Further, "collect information", "process information, and "share & preserve information" were added as sub-tasks of the world interaction management component of GAM. The result is shown in figure 7.
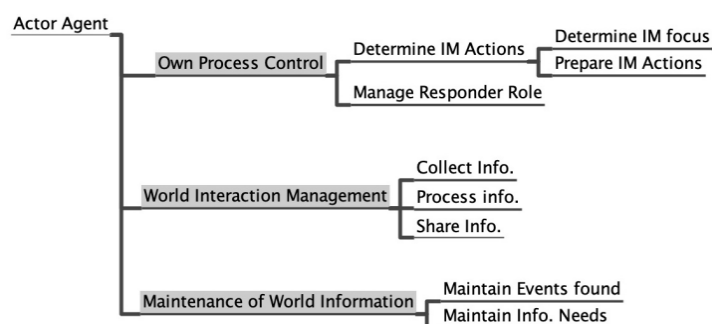

Figure 7: Task hierarchy obtained through specialization of the GAM components "Own Process Control", "World Interaction Management", and "Maintenance of World Information" (highlighted in grey in the picture) based on the system composition.

4.29   Next, the *tasks input and output* information were designed respectively by considering (a) the information required by each task and (b) the information that the task provides (potentially to other tasks). In the case of "Determine IM actions", the required inputs were the currently known information needs of the actor as actors can only look for information associated with known information needs (cf. P. 4.20). The output information provided by "Determine IM actions" were the decisions made in terms of information collection, processing, and sharing. Then, to capture the information exchange among tasks, *information links* were designed by connecting the tasks that provide particular outputs with the tasks that require such information as an input. For instance, in the case of "Determine IM Actions", information had to be provided concerning the currently known information needs of the actor. This information is provided by the task MWI and, specifically, by its sub-task "maintain information needs". As such, information links were added to connect "maintain information needs" with "determine IM actions". First, an information link called "known information needs to output" was added between "maintain information needs" and the output of its parent task MWI. Second, the link between MWI and OPC was already included as part of the generic model GAM (cf. Appendix A). Third, another link called "known info needs to dim" was added between the input of the OPC and the input of the OPC's sub-task "Determine IM Actions". With regards to *task delegation*, the tasks were assigned to the entities or agents i.e. the actors, objects, and environment that were found to carry out the specific activities in the system composition as shown in Table 3. For instance, "Manage Responder Role" is a task that is delegated to community members as found in the system composition. All the other tasks discussed in this section are associated not only with community members but with all agents representing an actor (cf. Table 3). Figure 8 shows the resulting specialization of GAM with respect to its OPC, WIM, and MWI components and for the specific case of a community responder.
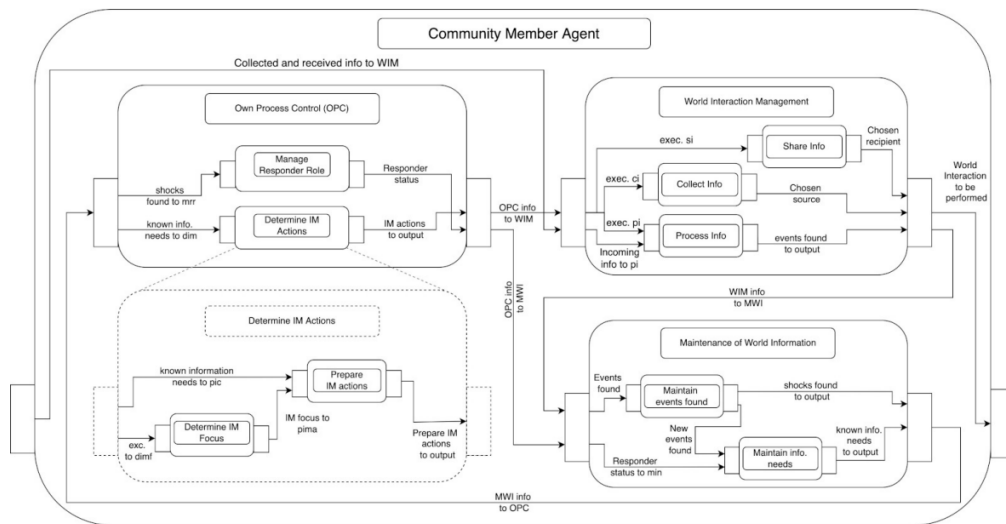
Figure 8: Example of tasks specialization for community member agents. The example shown focuses specifically on the specialization of GAM's generic tasks "Own Process Control" (OPC), "World Interaction Management" (WIM), and "Management of World Information" (MWI). In the figure, the dashed line is used to highlight the specialization of the task "Determine IM actions" with the sub-tasks "Determine IM Focus" and "Prepare IM Actions". The full lines in the picture show the links used for information exchange among tasks.

4.30 The *instantiation* of GAM was carried out by defining domain-specific instances of the of generic information types included in GAM (namely, world information, agent information, agent identification, domain actions, domain agent characteristics). This was carried out by analyzing the (a) the system composition, (b) the specialization presented above, and (c) the results of the conceptual framework application through the lens of the generic information types. For example, the conceptual framework application showed that, in the considered system, two types of information are generated by the environment (or world) namely, "shocks" and "announcements" (cf. P. 4.17). As such, shocks and announcements were introduced as instances of the information type "world info" in GAM (this is the type of information shared and collected by the agents in interaction with other agents and the environment). Additionally, the information type "domain agent characteristics" was instantiated (refined) by assigning state variables and properties to the agents. *States* (or state variables) were introduced when a task required a particular variable to keep track of the dynamic state of an agent e.g., associated with its goals (information needs), condition (e.g., affected by a flood), or roles (responder or not). In other cases, state variables were included to store, receive and share information according to the devised information flows (e.g., in the case of the list of known information needs). *Properties* of the agents were also introduced to account for the static characteristics of the agents necessary to determine how particular tasks were going to be carried out by given agents (e.g., group). Table 4 shows the result.

Generic information types from GAM	Instances for the domain of disaster IM
World info	Shock, Announcement, Noise[16]
Other agent info	Connections
Other agent identification	N.A.
Domain actions	N.A.
Domain agent characteristics	Known information needs (state), Latent information needs (state), Responder? (state), Affected? (state), IM actions (state), IM focus (state), Group (property), Altruism (property)

Table 4: Instantiation of the domain-specific information types from GAM for the community member agent.

4.31 Given the agents, their tasks, states, properties, and interactions a model conceptualization was built. When the agents had common tasks, properties and states and could therefore be seen as instances of a more abstract entity, a new entity was introduced with the given common tasks, properties and states. Figure 9 shows the resulting conceptualization including both the general entities and their instances for the specific case of Marunda.
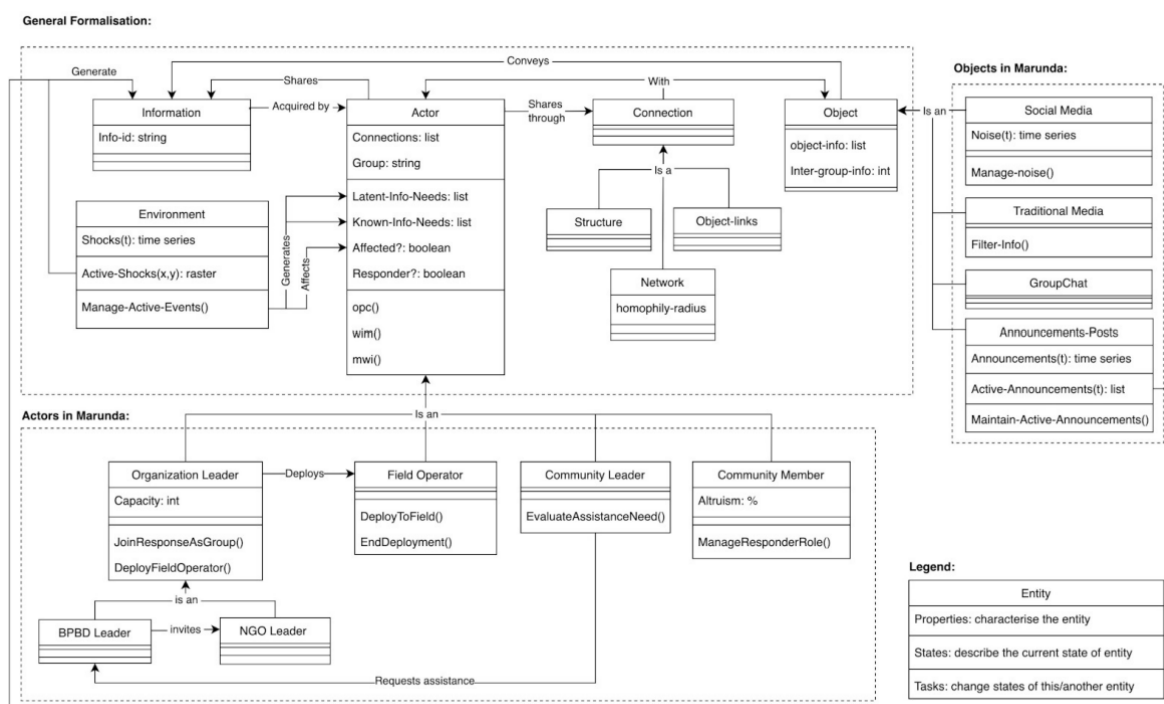

Figure 9: Concept Formalization: description of entities, their properties, states and tasks in bottom-up disaster IM both in general and for the case study of Marunda. Among these entities, the actors, objects and environment are agents.

4.32 Then, quantitative data on the number of population per age in the Marunda administrative subdivisions (RT and RWs cf. Figure 6) was used to capture in the model the number of residents that actually live in Marunda and their spatial locations.

4.33 Next, to capture the *relevant system's performance* selected in the problem definition (cf. P. 4.22), the authors developed direct indicators of information quality and specifically of timeliness and relevance to be provided as the model output. These quantitative indicators were designed based on the definitions obtained through qualitative inquiry during the conceptual framework development (cf. P. 4.11 and 4.12). Developing direct indicators of information quality is a novel approach as the existing ABMs capturing disaster IM rely on indirect (or proxy) operational indicators such as the correct and timely allocation of resources to measure information quality (Comfort et al. 2004; Zagorecki et al. 2009; Altay & Pal 2013). Using direct indicators of information quality has the advantage of allowing to focus on the information management challenges at hand while abstracting from the context of a specific operational problem. Additionally, the authors chose to measure the *gap in information relevance and timeliness* rather than relevance and timeliness, as this was found to be a better measure of the on-going performance of the system (or lack thereof) during the response to a disaster. Specifically, information gaps represent the average amount of information needs that the actors were not yet able to address at a given time of simulation. In the case of relevance, this gap is measured without considering when the information needs were addressed. Conversely, the timeliness gap takes into account time and it can only be reduced when the information needs were addressed before their expiration. Given these consideration the indicators for relevance and timeliness gaps were defined as follows.

$$Relevance\ Gap\ (t) = \frac{\sum_{k=1}^{n_{actors}} (info\ needs_k\ (t) - info\ needs\ addressed_k\ (t))}{n_{actors}} \tag{1}$$

$$Timeliness\ Gap\ (t) = \frac{\sum_{k=1}^{n_{actors}} (info\ needs_k\ (t) - info\ needs\ addressed\ on\ time_k\ (t))}{n_{actors}} \tag{2}$$

Where:

- n_{actors} = total number of actors in the simulation;

- $info\ needs_k(t)$ = total information needs received by the actor k at the simulation time (t)

- $info\ needs\ addressed_k(t)$ = total information needs addressed for the actor k at the simulation time (t)

- $info\ needs\ addressed\ on\ time_k(t)$ = total information needs addressed before their expiration for the actor k at the simulation time (t)

4.34 From this definition it follows that the timeliness gap is always higher than or equal to the relevance gap. This is due to the fact that the number of information needs addressed on time "$info\ needs\ addressed\ on\ time_k(t)$" is always less or equal to the information needs owned by the actor "$info\ needs\ addressed_k(t)$". The timeliness and relevance gap indicators were computed as a total and with specific reference to known and latent information needs as discussed in the section "Conceptual Framework Application".

Model narrative development

4.35 Starting from the general conceptualisation and the conceptual framework application (specifically the analysis of system's change and performance P. 4.17-4.19) from the previous steps, a narrative was developed by specializing the task control knowledge and instantiating the knowledge bases included in GAM. In the following, the narrative development is illustrated through the example of the OPC and one of its sub-task "Determine IM actions" as shown in previous section (cf. P. 4.28).

4.36 First, **task control knowledge** was specialized for each of the tasks introduced in the model conceptualization that have sub-tasks (cf. the Appendix). This included for instance the OPC with its sub-tasks manage_responder_Role and determine_IM_actions. Task control knowledge had to be provided also for determine_IM_actions to manage the execution of its sub-tasks determine_IM_focus and prepare_IM_actions (cf. Figure 8). In the case of the OPC, task control knowledge was specialized as in the following.

```
if start
then next−component−state (determine_IM_actions , awake)
and next−target−set (determine_IM_actions , IM_actions)
```

The above means that if the OPC is started by the task control knowledge of the community member agent, the sub-task determine_IM_actions is activated with the target (or goal) to provide the IM_actions as its output. Similarly, also the manage_responder_role task is managed by the task control knowledge of the OPC, as shown in the following.

```
if start
then next−component−state (manage_responder_role , awake)
and next−target−set (manage_responder_role , responder ?)
```

In the above, it can be noticed how the information type "responder?" obtained through instantiation at P. 4.29 is used by the task control knowledge of the OPC to set a target for the sub-task become_responder.

Next, the task control knowledge of the the task "determine IM actions" was specialized as in the following.

```
if start
then next−component−state (determine_IM_focus , awake)
and next−target−set (determine_IM_focus , IM_focus)
```

The above determines that when the determine_IM_actions task is awakened by the task control knowledge of its parent task (the OPC), the sub-task determine_IM_focus is awakened with the target of producing the information management focus as its output.

```
if evaluation (determine_IM_focus , IM_focus , succeeded)
then next_component_state (prepare_IM_actions , active)
and next−target−set (prepare_IM_actions , IM_actions)
and next_link_state (IM_focus_to_pima , awake)
```

4.37 The task control knowledge shown above determines that, once the determine_IM_focus task succeeded in its target, the prepare_IM_actions task is awakened with the target of providing the IM_actions as its output. It must be noted that the IM_actions are transferred from the task determine_IM_focus to the task prepare_IM_actions via the link IM_focus_to_pima. Such information exchange is possible given the link is awakened by the task control knowledge (see the "next_link_state" statement above).

4.38 Second, **knowledge bases** were instantiated for each of the tasks that do not have sub-tasks. Task control knowledge looks at the sub-tasks of a task as black boxes that need to be managed based solely on their inputs and outputs. Knowledge bases define the rules implemented within a task once it is activated by the task control knowledge. Specifically, the knowledge bases define the way a task's output information is obtained given the input information (if any). Knowledge bases were instantiated based on the concept formalization and on the results of the analyses of system's change and performance (cf. section "Conceptual framework application" P. 4.19-4.20). For example, the knowledge base used by the task Prepare_IM_actions was defined based on the analysis of system's performance. Two distinct types of information needs, namely latent and known information needs were found (cf. P. 4.20). These two types of information needs present implications not only for the performance of the system, but also for the information collection behavior of the actors. Specifically, actors can collect information only when they have information needs of which they are aware i.e., when they have know information needs. As such, the knowledge base for the task Prepare_IM_actions enables the actors to collect information only when their list of known information needs is not empty. This knowledge base is defined as in the following.

```
if (IM_focus, collect)
    if not (known_info_needs_list, empty)
    then (IM_actions, collect)
    if (known_info_needs_list, empty)
    then (IM_actions, process)

if (IM_focus, process)
then (IM_actions, process)
```

4.39 The knowledge base above specifies that when the IM_focus obtained through the determine_IM_focus task is that of collecting information, then two options are available. If the agent's list of known information needs is not empty, then the agent proceeds to collect information by setting the IM_actions information type to "collect". However, if the list of known information needs is empty, then the agent proceeds to process the received information by setting IM_actions to "process". Additionally, when the IM_focus obtained through the determine_IM_focus task is that of "processing", the agent will again proceed to process the received information by setting IM_actions to "process".

4.40 Next, a **narrative was assembled**. Firstly, the task control knowledge informed the order of execution of the tasks in the narrative. Starting from the most abstracted tasks and gradually proceeding to their sub-tasks, it was possible to delineate the sequences of tasks triggered by particular events or activation conditions (cf. "Task control knowledge" in Appendix A). However, it must be noted that the occurrence of task-triggering events was not known. The order of occurrence of task-triggering events was a deliberate choice of the modeller. For the case of a community member agent, the most abstracted tasks were the OPC, WIM and MWI. Among these, MWI was chosen as the initial task. MWI, and specifically its sub-task maintenance_of_information_needs, is triggered every pre-defined amount of time[17] to check whether a new shock or announcement is generated by the environment, and possibly add an information need when such an event is relevant for the considered actor. Next, the actor determines what kind of information management activities to execute through the OPC's sub-task determine_IM_actions (and its sub-tasks determine_IM_focus and prepare_IM_activities). The task determine_IM_focus chooses whether to collect new information or process the received information. Then, determine_IM_actions assesses whether the actor has known information needs before proceeding to collect information, or proceeds to process the received information otherwise. Once the agent has collected or processed information, the agent proceeds to share information. The knowledge bases provide the rules executed by the most elementary tasks (i.e. those that do not have sub-tasks). The resulting narrative for the considered example of a community member agent is shown in figure 10.
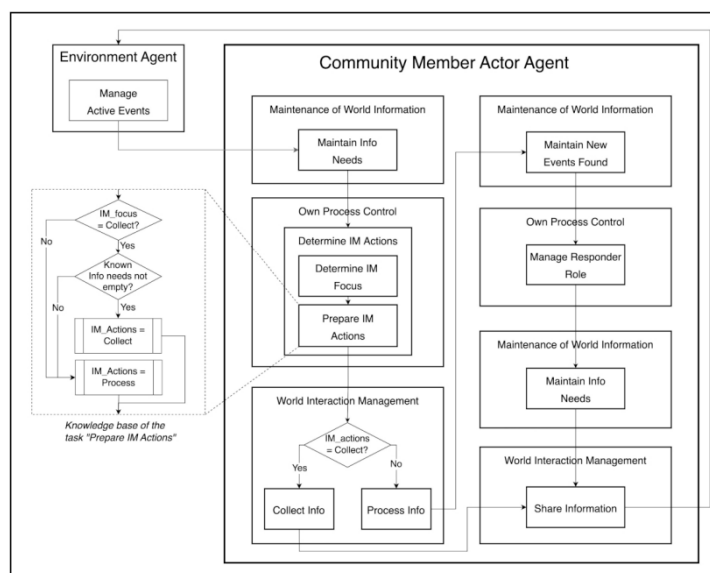
Figure 10: Developing a Model Narrative: narrative for the community member actor agent developed based on the model conceptualization, the specialization of the task control knowledge, and on the instantiation of the knowledge bases included in the generic model GAM. In the figure, the task control knowledge is represented by the order of execution of the tasks shown by the full arrows. An example of a knowledge base is also illustrated for the task "prepare IM Actions" through dashed lines.

Software implementation

4.41 The model was implemented in Netlogo 6.1.1 as this modelling environment provides the means to implement reasonably complex models with a relatively low time investment required for the software implementation (Abar et al. 2017). Figure 11 shows part of the GUI of the resulting implementation. *The source code, input data, and its description based on the ODD protocol (Grimm et al. 2006, 2010, 2020) can be found on the COMSES Net Computational Model Library* at `https://www.comses.net/codebases/a7c3bb63-258b-4a4c-9f58-4f42ec0dfd86/releases/1.0.0/` (Nespeca et al. 2022).
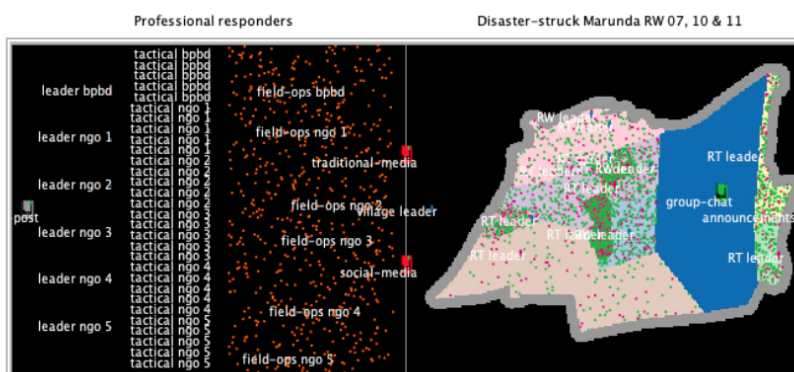

Figure 11: GUI resulting from the software implementation of the concept formalization and narrative.

Model evaluation

4.42 The empirical model was verified thoroughly through single agent, interaction and multi-agent testing as suggested in (Nikolic & Ghorbani 2011). To provide an example of multi-agent testing the model was run a total of 1215 times considering different scenarios of environmental turbulence. Environmental turbulence was conceptualised as the amount of shocks and announcements occurring during one day of simulation (cf. P. 4.25).

The results of the data analysis are shown in the following. In all figures below, the confidence levels were calculated via bootstrapping.

4.43   Firstly, it can be noticed that the information gaps associated with both timeliness and relevance tend to increase over time (see Figure 12). This shows that, while the number of information needs grows due to the occurrence of more disruptive events (shocks and announcements), the actors are not receiving all the information they need, leading to the cumulation of information gaps. However, it can also be observed that in some cases (e.g., after 10 hours of simulation) the gap is being reduced as the actors' information needs are being addressed at a rate that's higher compared to the increase in information needs due to new disruptive events.

4.44   Secondly, the data analysis confirmed that, as follows from definition, the median of the timeliness gap is always equal or above that of the relevance gap (cf. P. 4.36). Further, at some point of the simulation the information needs of the actors start to expire. When this occurs, it becomes possible for the actors to address their information needs not on time. As such, the value of "$info\ needs\ addressed\ on\ time_k\ (t)$" in equation (2) becomes smaller than "$info\ needs\ addressed_k\ (t)$" in equation (1), leading to an increase of the timeliness gap compared to the relevance gap and thus to their divergence as shown in Figure 12. For the same reason, it can be observed that in some cases the relevance gap is reduced to a greater extent compared to the timeliness gap.
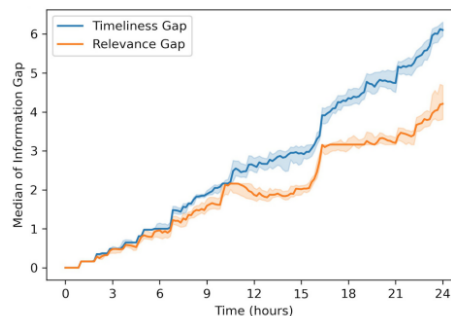

Figure 12: Comparison between the median of the timeliness and relevance information gaps at each time step across all simulations. The vertical axis represents the number of pieces of information that the actors still need on average.

4.45   Further, the data analysis demonstrated that both the number of shocks per day and announcements per day contribute to increasing the cumulative relevance and timeliness gaps (see Figure 13). These results confirm the expectation that higher levels of environmental turbulence (i.e., more things to keep track of) make it more difficult for the actors to address their information needs and thus reduce the information gaps. It can also be observed that the confidence intervals around the median are much wider at specific points of the simulation compared to others. This uncertainty in the median can be attributed to the stochasticity used to (a) initialize the model's the structures and networks (connections cf. Figure 9), and (b) make the agents choose which other agents to collect information from and share information to. The fact that this uncertainty varies over the simulation is a consequence of the model structure. Specifically, it is only after a given disruptive event is released, that information needs are assigned to the actors, which on turn increases the information gaps. How rapidly (if at all) the associated information needs are addressed will depend on the connections available and on how effective the random chains of information exchange across the actors turn out to be.
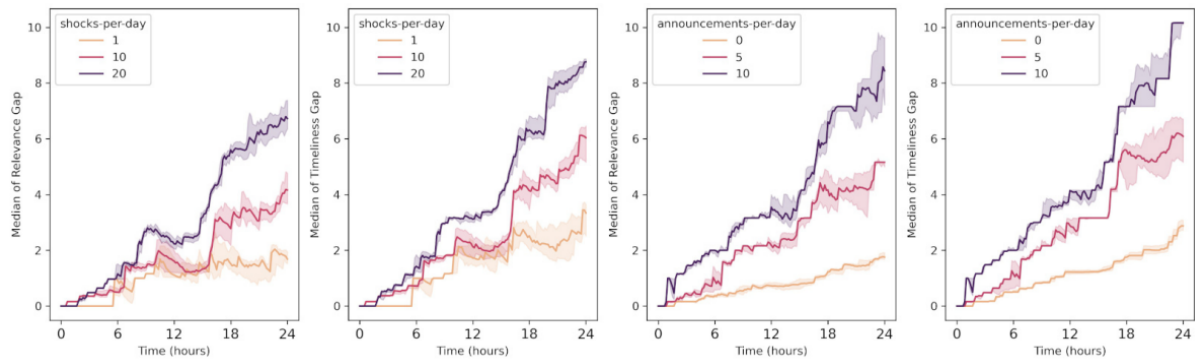
Figure 13: Median of Relevance and Timeliness Gaps for different levels of environmental turbulence (shocks per day and announcements per day).

4.46 Finally, latent and known information needs were compared in terms of the associated relevance and timeliness gaps. The results show that information gaps associated with latent information needs have medians higher than those of known information needs (see Figure 14). Such a system's performance is also consistent with the authors' expectations that latent information needs are more difficult to address as the actors are not aware of them and therefore cannot actively search for the related information (cf. P. 4.20). Figure 14 also shows that the difference between information gaps associated with latent and known information needs is particularly pronounced for the timeliness gap. This behavior is exemplified by the information gaps for latent needs occurring after 18 hours of simulation. The relevance gap peaks and then start decreasing, while the timeliness gap simply keeps growing. This suggests that the impact of being unaware of the information needed affects more heavily the ability of actors to get such information on time, rather than simply receiving it at any time.
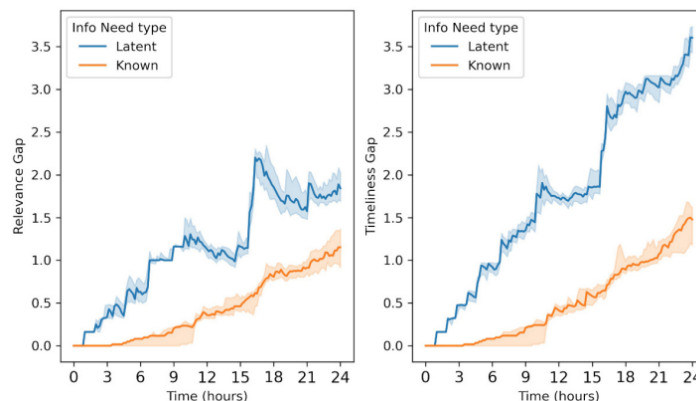

Figure 14: Comparison between the median of information gaps for latent and known information needs. The figure shows such a comparison for both the relevance gap (left figure) and the timeliness gap (right figure).

4.47 A validation was not performed at this stage and will be carried out as a future step. Given its descriptive purpose, the model developed is not intended to precisely reproduce the system's performance observed in reality. The goal is rather to formalize and combine the knowledge gathered through a study for the Marunda community together with previously available theory and models on crisis IM. As such, the validation in this case will not focus on assessing whether the model reproduces the performance observed at the macro level in the Marunda community (through the analysis of system's performance, cf. P. 3.43). The validation will rather aim at ensuring that the model conceptualization and narrative match the way information is managed at the micro level in the case study. This will be achieved by discussing the model conceptualization and narrative with the members of the Marunda community and other organizations captured in the model. Alternatively, a comparison of the model with the literature available for the case study could also be used to validate the model.

Iterative conceptual framework development

4.48 In the model development phase a new system characteristic was found, namely that of "Objects". Objects represent non-human (technological) entities that can support the actors belonging to one or more groups in their IM and coordination activities. Given this definition obtained through the conceptual framework application (cf. P. 4.18), the following relationships are found between objects and other system's characteristics. Activities can be carried out *through* objects. Additionally, objects *connect* the actors within a group and also actors that belong to different groups so that they can exchange information and coordinate. The objects and their relationships with the system's characteristics "Activities" and "Groups" were integrated in the conceptual framework as shown in Figure 15. This updated conceptual framework can be used in substitution of the one developed in (Nespeca et al. 2020) (Figure 5).
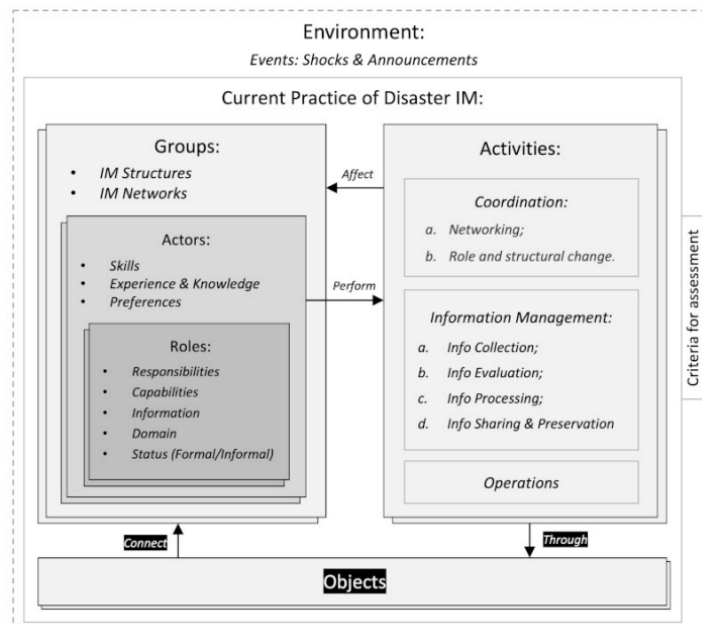

Figure 15: Updated conceptual framework with the new systems characteristics and relationships (highlighted in the figure).

Discussion & Conclusions

5.1 This article proposes a methodology that enables to rigorously develop ABMs for policy formulation and evaluation based on qualitative inquiry. The methodology fills a gap in the literature by providing the means to (a) strike a balance between the comparability of framework-based methodologies and the flexibility of case-based methodologies and (b) to provide versatility by enabling to develop ABMs with both theoretical and empirical purposes.

5.2 The methodology is structured in two phases. In the first phase, a conceptual framework centered on a specific policy problem is designed based on literature and the findings from one or more case studies. In the second phase, the conceptual framework together with generic models are used to guide the development of an ABM. The way the conceptual framework and generic models support model design depends on whether the model is directly related to a case study (empirical) or not (theoretical) (Edmonds et al. 2019). In the case of *empirical models* the conceptual framework is applied to a case study to analyze the system's configuration, change, and performance. Based on these analyses, a context-specific conceptual model (a system composition) is developed which captures the agents, their interactions with other agents, and the indicators of system's performance and change that are key for the considered policy problem. Next, a generic model (Brazier et al. 2002) capturing a particular agent architecture is selected depending on the modelling purpose and strategy (Edmonds et al. 2019). Such a generic model is then used to analyze the results of the conceptual framework application, system composition, and possibly additional qualitative data from the case study to inform the design of the internal processes that drive the agents' interactions. The result is an empirical ABM. In the case

of *theoretical models*, the conceptual framework provides an inventory of (a) systems characteristics, attributes and relationships of which the researchers may decide to introduce instances in the considered abstract system and (b) a list of criteria for assessment as indicators of the system's performance that the researchers may wish to study. Once a conceptual model (the system composition) capturing the agents, their interactions and indicators of system's performance is designed, a generic model is selected and used to guide the development of the internal processes that support the agents' interactions. The result is a theoretical ABM.

5.3 The use of this methodology was illustrated through a case study of disaster IM in Jakarta (Indonesia). The considered policy problem was in this case the design of bottom-up disaster IM policies. First, a conceptual framework centered on the design of bottom-up disaster IM policies was designed based on available literature, case study interviews, and focus groups. The following mission was individuated: "*to provide relevant, reliable and verifiable information to the actors who need it, when they need it in an accessible manner*". The criteria for assessment found were: information relevance, timeliness, accessibility, reliability, verifiability, and load. Second, the conceptual framework was applied to the case study to carry out the analyses of system's configuration, change, and performance based on the qualitative data collected via interviews and focus groups. These analyses were instrumental in the development of a conceptual model (the system's composition) capturing the agents, their interactions and performance that are key for the considered policy problem (i.e., the current practice of disaster IM) in the Marunda community. Next, the Generic Agent Model (GAM) was chosen given that other more specialized generic models such as GCAM were not suited for the considered policy problem of bottom-up disaster IM. GAM was used to structure and interpret the results of the conceptual framework application, and the system composition to design the internal processes driving the agents' interactions. This process, called specialization and instantiation, enabled the formalization of the system composition in a software-implementable manner (or model concept formalization) and the development of a model narrative (or model narrative development), resulting in an empirical descriptive ABM. Finally, during model development new systems characteristics and relationships were found. These were integrated in the conceptual framework for future use.

5.4 The case study showed how the proposed methodology and specifically the use of the conceptual framework and of the generic model GAM enabled the translation of qualitative data into an empirical ABM by balancing comparability and flexibility while maintaining rigour. Firstly, the methodology enabled to retain some degree of *comparability* with future studies by developing and centering the model development process on a conceptual framework tailored to the policy problem of bottom-up disaster Information Management (IM). Such a conceptual framework provides a clear mission, categories of meaning, and criteria for the assessment of policy "performance" so that future simulation studies focusing on bottom-up disaster IM will be able to interpret and build upon the results of this study through the same conceptual framework. However, while future simulation studies focusing on the same policy problem will be able to rely on the same conceptual framework, these studies may potentially choose different generic models compared to the one used here (GAM). As such, comparability with future studies was retained only to a certain degree. Other methodologies such as Ghorbani et al. (2015) that fully rely on the same framework across different studies (framework-based) provide a greater level of comparability. However, this comparability comes at the cost of flexibility i.e., the researchers are not able to choose among different agent theories or architectures that e.g., are adequate for the particular policy problem considered.

5.5 Secondly, the proposed methodology provided some degree of *flexibility*. Firstly, the methodology enabled to capture a novel policy problem (bottom-up disaster information management) for which an adequate conceptual framework was not initially available. Secondly, the methodology provided the means to choose a generic model capturing a particular agent architecture that was suited to the considered policy problem and modelling purpose. However, flexibility was provided only to a degree as the model development process was constrained by the use of the conceptual framework developed in the first phase of the methodology. Methodologies that tailor an ABM to a particular case without relying on a common framework (case-based) as in Bharwani et al. (2015) provide a higher degree of flexibility in that they can capture the nuances and details that characterize a particular case study without being limited by the availability and constraint of pre-existing frameworks or agent architectures. However, such methodologies are not designed to retain common ground that can be used to compare the results of different case studies focusing on the same policy problem.

5.6 Thirdly, the proposed methodology enabled to maintain rigour by systematically structuring and interpreting qualitative data through the lens of two frameworks (Edmonds 2015), namely the conceptual framework (developed in the first phase) and the GAM generic model.

5.7 In sum, while the methodology proposed in this article does not provide the same degree of flexibility of case-based methodologies, nor the same level of comparability provided by framework-based methodologies, it enables to provide both flexibility and comparability to a certain degree, thus striking a balance between them

while maintaining rigour. Such a balance is obtained by providing some standards (the conceptual framework developed to capture a particular policy problem and the use of generic models) but not too many (i.e., the researchers can choose different generic models for the same policy problem).

5.8   With regards to *versatility*, the process of designing policies supported by ABMs may involve the development and use of a series of models with different theoretical and empirical purposes (Edmonds et al. 2019). In this study, an empirical ABM with a descriptive purpose was developed which is meant to capture and formalize the knowledge gathered on the current practice of disaster IM in the Marunda Community of North Jakarta. Based on this model, another empirical model with an explanatory purpose could be developed at a later stage to support policy design. Theoretical models may also be needed e.g., with the purpose of illustrating the implications of given policy designs in an abstract system prior to testing them empirically. While a theoretical model was not developed in this article, the proposed methodology provides indications for such a model to be designed via the conceptual framework obtained in phase 1 and generic models. Specifically, instead of being used as tools to structure and interpret qualitative information, the conceptual framework and generic models provide templates for the design of the agents, their interactions, performance, and internal processes. As such, the proposed methodology is versatile as it enables to develop both theoretical and empirical ABMs (Edmonds et al. 2019) for policy formulation and evaluation (Gilbert et al. 2018) through the lens of the same policy-centered conceptual framework.

5.9   The proposed methodology presents the following implications for the development of ABMs for policy support through qualitative inquiry. Firstly, the methodology enables to uncouple the policy-centered analysis of the key agents, their activities, and interactions (captured in the system composition) from the agent-centered analysis of the agent's internal processes that drive their activities and interactions (captured through generic models). As such, via this methodology, researchers can rigorously interpret the same system composition through the lens of different generic models representing e.g., different agent architectures. This can enable for instance to systematically develop a series of comparable ABMs for the same case each including a different agent architecture. A potential application could be testing alternative explanations for a particular system's change or performance of interest that is relevant for policy formulation (as in (Adam & Gaudou 2017)). Secondly, in the absence of other more refined generic models that are suited to the considered policy problem, the researchers can rely on the Generic Agent Model (Brazier et al. 2002). GAM only assumes general features that characterize an agent (Wooldridge & Jennings 1994, 1995), and as such it can be widely applied to different policy problems.

5.10   Further, when a generic model is used in the model concept formalization and narrative development, additional tasks and knowledge structures may be introduced that can be abstracted to develop a new generic model tailored to the considered (class of) policy problems (Brazier et al. 1996). Such an abstraction process requires additional effort from the researchers. However, it has the advantage that future studies focusing on the same policy problem are able to rely on a generic model that is refined for the considered policy problem (instead of e.g., using GAM). Finally, according to the proposed methodology not only empirical but also theoretical models are developed on the basis of a conceptual framework that is designed empirically (through qualitative inquiry). Then, while theoretical models are not specific with respect to any case study (Edmonds et al. 2019), those developed with this methodology still reflect empirically-embedded system's characteristics, attributes, relationships and criteria for assessment of performance that are relevant for the considered policy problem and are captured in the conceptual framework. While the findings of these theoretical models e.g., hypotheses regarding the effectiveness of particular policies still require to be tested empirically, these models are more likely to reflect issues that are relevant for policy formulation and evaluation compared to models developed through conceptual frameworks that are designed in a purely inductive manner.

5.11   Despite its advantages, the methodology introduced in this article presents limitations providing ground for further research. Firstly, the proposed methodology requires a considerable investment in time and resources required to develop a novel conceptual framework and possibly a new generic model before actually developing a model. Secondly, the proposed methodology focuses solely on the use of qualitative inquiry. However, combining the current qualitative approach with quantitative research methodologies provides an opportunity for enhanced rigour in the development of empirical models (Bharwani et al. 2015; Antosz et al. 2022). For instance, the model conceptualization and narratives designed through the use of generic models could inform the design of quantitative research tools (e.g., surveys) that aim at capturing the choices made by the actors statistically.

Acknowledgements

Appendix: Generic Agent Model (GAM)

This appendix provides a brief description of the Generic Agent Model designed by Brazier et al. (2000) based on the weak notion of agent introduced by Wooldridge & Jennings (1994, 1995). This appendix focuses specifically on GAM's process composition. Further details including the generic model's knowledge composition can be found in Brazier et al. (2000). A generic model's process composition includes the models' task hierarchy (abstracted tasks and their sub-tasks), tasks' inputs and outputs, tasks' information exchange (executed through information links among tasks), and task control knowledge (capturing the sequencing of tasks).

Task hierarchy

The task hierarchy of GAM includes only two hierarchical levels, namely that of the agent and the abstracted tasks (or components) carried out by the agent. Such abstracted tasks are the Own Process Control (OPC), Agent Interaction Management (AIM), World Interaction Management (WIM), Maintenance of Agent Information (MAI), Maintenance of World Information (MWI), and Agent Specific Tasks (AST) (see Figure 16).
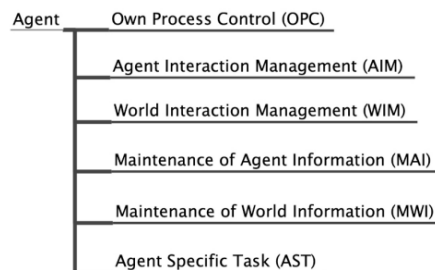

Figure 16: Task hierarchy of the Generic Agent Model (GAM).

The OPC component maintains a self-model of the agent (i.e., a model of the agents' goals and, in general, of the characteristics that distinguish them), and carries out decisions to trigger further tasks that are performed by other components of GAM. The AIM component executes interactions with other agents such as receiving or sharing information e.g., when requested to do so by the OPC. The WIM component carries out interactions with the environment or world such as performing observations of the environment. The MAI component maintains information regarding other agents that the agent may use to carry out other tasks (such as deciding who to cooperate with, cf. Brazier et al. 1996) and provides this information to other tasks when requested to do so. The MWI task stores information from the environment (e.g., shocks and announcements) that the agent finds relevant and provides it to other tasks or agents when this information is requested by other tasks. Finally, the AST task is associated with domain-specific tasks an agent carries out such as combining different pieces of information to obtain new information (Brazier et al. 1996).

Task input and output

The tasks' inputs and outputs for GAM are shown in Table 5.

Task (or process)	Input information types	Output information types
own process control	belief info	own characteristics
agent interaction management	incoming communication info, own characteristics, belief info	outgoing communication info, maintenance info
world interaction management	observation result info, own characteristics, belief info	observation info, action info, maintenance info
maintenance of agent information	agent info	agent info
maintenance of world information	world info	world info

Table 5: GAM's input and output information types, from (Brazier et al. 2000).

Information exchange

The information exchange among tasks is enabled by information links among tasks. Figure 17 shows the tasks included in GAM (according to the task hierarchy) and their information links used for information exchange.
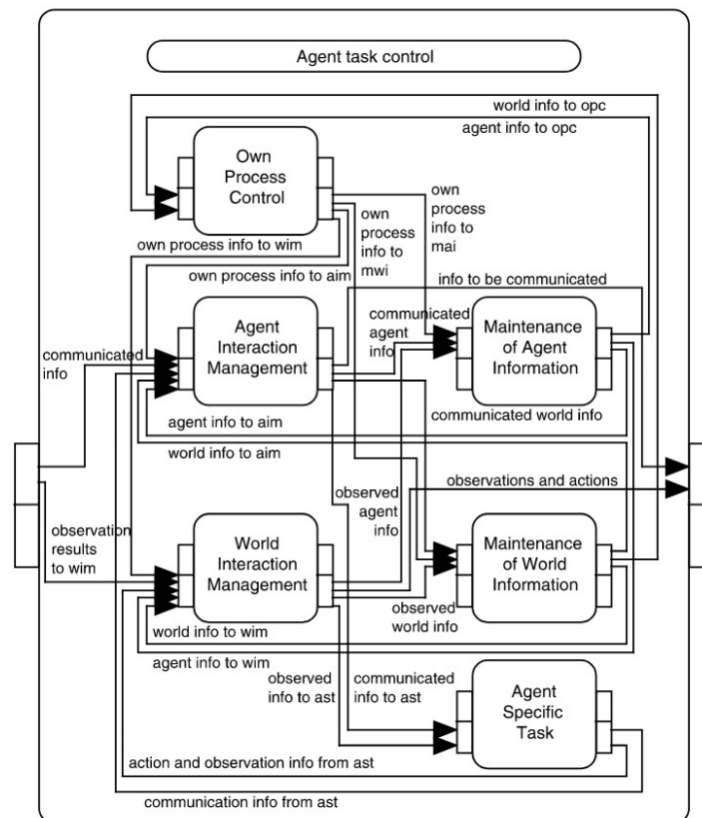

Figure 17: Information exchange among GAM's tasks (Brazier et al. 2000).

Task control knowledge

Task control knowledge defines the conditions under which particular tasks are carried out, and the goals or targets associated with the activation of tasks. In practice, such a knowledge is represented as a set of rules capturing the triggering knowledge states (or activation conditions) under which a task is executed, the task that is executed, the target (or goal) that determines when the task is concluded, and possibly other statements capturing e.g., the activation of information sharing links that enable to share information among tasks. Triggering

knowledge states can e.g., be associated with the activation of a parent task or agent, or the availability of new input information provided by an information link. The following pseudo code shows the general structure of task control knowledge.

```
if activation_condition
then next_component_state(task_to_execute_name, active)
and next_target_set(target_name)
```

In the case of GAM, only the agent has sub-tasks (e.g., OPC, AIM, etc.). All of these tasks are primitive, meaning they do not have sub-tasks. As such, the task control knowledge included in GAM only pertains to the agent (represented as "Agent task control" in Figure 17). GAM's task control knowledge dictates that all tasks and information links are activated at the beginning of the simulation. This means in practice that all tasks process incoming information as soon as it is received as an input by the task (in an asynchronous manner). The following pseudo code shows the task control knowledge for two tasks and one information sharing link among them (cf. Figure 17). Task control knowledge for the remaining tasks and links (shown in Figure 17) is specified according the same structure as that shown in the example below.

```
if start
then next_component_state(own_process_control, awake)

if start
then next_component_state(agent_interaction_management, awake)

if start
then next_link_state(own_process_info_to_wim, awake)
```

Notes

[1] While DESIRE was initially conceived to design multi-agent systems, it can as well be applied to the design of ABMs.

[2] It must be noted that the conceptual frameworks developed in phase 1 enable part of the methodology, but they are not the methodology. Through the methodology, it is possible to develop different conceptual frameworks for different policy problems when such frameworks are not already available.

[3] As such, by "policy-centered conceptual framework", the authors do not mean a conceptual framework that provides relevant design dimensions and options for policies that can be implemented to tackle a given policy problem e.g., as proposed in (Bharosa et al. 2011). Rather, the conceptual frameworks developed with this methodology are strictly meant to enable the development of ABMs for policy formulation and evaluation.

[4] *Empirical models* are those which have a direct relationships with a specific case study. Descriptions, explanations and predictions are examples of empirical modelling purposes. *Theoretical models* are those which do not have a direct relationship with any given case study or specific system. Illustrations and theoretical expositions are examples of theoretical modelling purposes. (Edmonds et al. 2019)

[5] With this distinction, the authors do not imply that theoretical models cannot be used in practical settings. However, theoretical model can be applied in practice only if they have been empirically tested in terms their micro assumptions and macro implications (Flache et al. 2017).

[6] New or more detailed criteria for assessment may be introduced at this stage compared to those presented in the conceptual framework.

[7] Ontologies can also be used to formalize an agent architecture in a rigorous way. However, ontologies can capture solely the declarative knowledge of an agent architecture, while overlooking the procedural knowledge (i.e., the rules followed by the agents when carrying put particular tasks)) (Zack 1999). Conversely, generic models can capture both the declarative and procedural knowledge of an agent architecture (Brazier et al. 2002).

[8] In some cases, the researcher may decide to collect further data that is specifically meant to capture the internal processes occurring within the agents. In such cases, the generic model can provide a framework that informs not only the data analysis, but also the data collection (Ghorbani et al. 2015).

[9]According to Brazier et al. (2000), task control knowledge is also obtained via the specialization of a generic model's process composition. However, given that task control knowledge defines procedures related to when and how particular tasks are carried out by an agent, the specification of task control knowledge is placed in the later Model Narrative Development step of this methodology.

[10]Case-study research in the field of disaster management has to often take advantage of the opportunities that arise when new cases occur (see for instance Comes et al. 2015; Baharmand et al. 2020).

[11]For instance, the conceptualization of the environment in a crisis as a series of cascading shocks producing information needs was introduced as in (Meijering 2019).

[12]Here called Information Management (IM) structures and networks.

[13]The attributes of these characteristics are not presented at this stage, but directly in the model conceptualization due to limitation in space.

[14]One of GAM's applications, namely the Generic Cooperative Agent Model (or GCAM) presented in (Brazier et al. 1996, 2000, 2002), has similarities with the distributed and (partly) cooperative nature of disaster information management. However, while disaster response is intrinsically characterized by urgency, GCAM is meant for non-urgent problems in which the agents have time to make joint plans. As such, GCAM is not a suitable generic model for developing ABMs that focus on capturing disaster response situations.

[15]The design choice of having a backlog of information that the actor can access and process at will was made to capture asynchronous communication e.g., through instant messaging apps.

[16]Intended as irrelevant information.

[17]This is another deliberate choice of the modeller. It could be executed e.g., at every time step.

References

Abar, S., Theodoropoulos, G. K., Lemarinier, P. & O'Hare, G. M. P. (2017). Agent based modelling and simulation tools: A review of the state-of-art software. *Computer Science Review*, *24*, 13–33

Adam, C. & Gaudou, B. (2017). Modelling human behaviours in disasters from interviews: Application to Melbourne Bushfires. *Journal of Artificial Societies and Social Simulation*, *20*(3), 12

Altay, N. & Labonte, M. (2014). Challenges in humanitarian information management and exchange: Evidence from Haiti. *Disasters*, *38*(s1), 50–72

Altay, N. & Pal, R. (2013). Information diffusion among agents: Implications for humanitarian operations. *Production and Operations Management*, *23*(6), 1015–1027

Antosz, P., Bharwani, S., Borit, M. & Edmonds, B. (2022). An introduction to the themed section on 'Using agent-based simulation for integrating qualitative and quantitative evidence'. *International Journal of Social Research Methodology*, *25*(4), 511–515

Anzola, D., Barbrook-Johnson, P. & Cano, J. I. (2017). Self-organization and social science. *Computational and Mathematical Organization Theory*, *23*(2), 221–257

Badham, J., Barbrook-Johnson, P., Caiado, C. & Castellani, B. (2021). Justified stories with agent-based modelling for local COVID-19 planning. *Journal of Artificial Societies and Social Simulation*, *24*(1), 8

Baharmand, H., Comes, T. & Lauras, M. (2020). Supporting group decision makers to locate temporary relief distribution centres after sudden-onset disasters: A case study of the 2015 Nepal earthquake. *International Journal of Disaster Risk Reduction*, *45*, 101455

Bateman, L. & Gralla, E. (2018). Evaluating Strategies for Intra-Organizational Information Management in Humanitarian Response. Proceedings of the 15th ISCRAM Conference, Rochester, NY

Bharosa, N. & Janssen, M. (2015). Principle-based design: A methodology and principles for capitalizing design experiences for information quality assurance. *Journal of Homeland Security and Emergency Management*, *12*(3)

Bharosa, N., Janssen, M. & Tan, Y.-H. (2011). A research agenda for information quality assurance in public safety networks: Information orchestration as the middle ground between hierarchical and netcentric approaches. *Cognition, Technology & Work*, *13*(3), 203–216

Bharwani, S. (2004). Adaptive knowledge dynamics and emergent artificial societies: ethnographically based multi-agent simulations of behavioural adaptation in agro-climatic systems. Working Paper, University of Kent, UK. Available at: `https://kar.kent.ac.uk/94209/`

Bharwani, S., Coll Besa, M., Taylor, R., Fischer, M., Devisscher, T. & Kenfack, C. (2015). Identifying salient drivers of livelihood decision-making in the forest communities of cameroon: Adding value to social simulation models. *Journal of Artificial Societies and Social Simulation*, *18*(1), 3

Black, N. (1994). Why we need qualitative research. *Journal of epidemiology and community health*, *48*(5), 425

Boero, R. & Squazzoni, F. (2005). Does empirical embeddedness matter? Methodological issues on agent-based models for analytical social science. *Journal of Artificial Societies and Social Simulation*, *8*(4), 6

Brazier, F., van Langen, P., Lukosch, S. & Vingerhoeds, R. (2018). Complex systems: Design, engineering, governance. In H. L. M. Bakker (Ed.), *Projects and People: Mastering Success*, (pp. 35–60). NAP - Process Industry Network

Brazier, F. M., Dunin-Keplicz, B. M., Jennings, N. R. & Treur, J. (1997). DESIRE: Modelling multi-agent systems in a compositional formal framework. *International Journal of Cooperative Information Systems*, *6*(01), 67–94

Brazier, F. M., Jonker, C. M. & Treur, J. (1996). Formalisation of a cooperation model based on joint intentions. International Workshop on Agent Theories, Architectures, and Languages

Brazier, F. M., Jonker, C. M. & Treur, J. (2000). Compositional design and reuse of a generic agent model. *Applied Artificial Intelligence*, *14*(5), 491–538

Brazier, F. M., Jonker, C. M. & Treur, J. (2002). Principles of component-based design of intelligent agents. *Data & Knowledge Engineering*, *41*(1), 1–27

Calder, M., Craig, C., Culley, D., De Cani, R., Donnelly, C. A., Douglas, R., Edmonds, B., Gascoigne, J., Gilbert, N., Hargrove, C., Hinds, D., Lane, D. C., Mitchell, D., Pavey, G., Robertson, D., Rosewell, B., Sherwin, S., Walport, M. & Wilson, A. (2018). Computational modelling for decision-making: Where, why, what, who and how. *Royal Society Open Science*, *5*(6), 172096

Comes, T., van de Walle, B. & van Wassenhove, L. (2020). The coordination-information bubble in humanitarian response: Theoretical foundations and empirical investigations. *Production and Operations Management*, *29*(11), 2484–2507

Comes, T., Vybornova, O. & van de Walle, B. (2015). Bringing structure to the disaster data typhoon: An analysis of decision-makers' information needs in the response to Haiyan. AAAI Spring Symposia

Comfort, L. K. (1994). Self-organization in complex systems. *Journal of Public Administration Research and Theory: J-PART*, *4*(3), 393–410

Comfort, L. K., Ko, K. & Zagorecki, A. (2004). Coordination in rapidly evolving disaster response systems: The role of information. *American Behavioral Scientist*, *48*(3), 295–313

Connell, J. P. & Kubisch, A. C. (1998). Applying a theory of change approach to the evaluation of comprehensive community initiatives: Progress, prospects, and problems. *New approaches to evaluating community initiatives*, *2*(15–44), 1–16

Dam, K. H., Nikolic, I. & Lukszo, Z. (2013). *Agent-Based Modelling of Socio-Technical Systems*. Berlin Heidelberg: Springer Science & Business Media

Edmonds, B. (2015). Using qualitative evidence to inform the specification of agent-based models. *Journal of Artificial Societies and Social Simulation*, *18*(1), 18

Edmonds, B., Le Page, C., Bithell, M., Chattoe-Brown, E., Grimm, V., Meyer, R., Montaola-Sales, C., Ormerod, P., Root, H. & Squazzoni, F. (2019). Different modelling purposes. *Journal of Artificial Societies and Social Simulation*, *22*(3), 6

Edmonds, B. & ní Aodha, L. (2019). Using agent-based modelling to inform policy – What could possibly go wrong? In P. Davidsson & H. Verhagen (Eds.), *Multi-Agent-Based Simulation XIX*, (pp. 1–16). Cham: Springer International Publishing

Eisenhardt, K. M. (1989). Building theories from case study research. *Academy of Management Review*, *14*(4), 532–550

Epstein, J. M. (1999). Agent-based computational models and generative social science. *Complexity*, *4*(5), 41–60

Fereday, J. & Muir-Cochrane, E. (2006). Demonstrating rigor using thematic analysis: A hybrid approach of inductive and deductive coding and theme development. *International Journal of Qualitative Methods*, *5*(1), 80–92

Flache, A., Mäs, M., Feliciani, T., Chattoe-Brown, E., Deffuant, G., Huet, S. & Lorenz, J. (2017). Models of social influence: Towards the next frontiers. *Journal of Artificial Societies and Social Simulation*, *20*(4), 2

Flyvbjerg, B. (2006). Five misunderstandings about case-study research. *Qualitative inquiry*, *12*(2), 219–245

Ghorbani, A., Bots, P., Dignum, V. & Dijkema, G. (2013). MAIA: A framework for developing agent-based social simulations. *Journal of Artificial Societies and Social Simulation*, *16*(2), 9

Ghorbani, A., Dijkema, G. & Schrauwen, N. (2015). Structuring qualitative data for agent-based modelling. *Journal of Artificial Societies and Social Simulation*, *18*(1), 2

Ghorbani, A., Ligtvoet, A., Nikolic, I. & Dijkema, G. (2010). Using institutional frameworks to conceptualize agent-based models of socio-technical systems. Proceeding of the 2010 Workshop on Complex System Modeling and Simulation

Gilbert, N., Ahrweiler, P., Barbrook-Johnson, P., Narasimhan, K. P. & Wilkinson, H. (2018). Computational modelling of public policy: Reflections on practice. *Journal of Artificial Societies and Social Simulation*, *21*(1), 14

Grimm, V., Berger, U., Bastiansen, F., Eliassen, S., Ginot, V., Giske, J., Goss-Custard, J., Grand, T., Heinz, S. K., Huse, G., Huth, A., Jepsen, J. U., Jørgensen, C., Mooij, W. M., Müller, B., Pe'er, G., Piou, C., Railsback, S. F., Robbins, A. M., Robbins, M. M., Rossmanith, E., Rüger, N., Strand, E., Souissi, S., Stillman, R. A., Vabø, R., Visser, U. & DeAngelis, D. L. (2006). A standard protocol for describing individual-based and agent-based models. *Ecological Modelling*, *198*(1), 115–126

Grimm, V., Berger, U., DeAngelis, D. L., Polhill, J. G., Giske, J. & Railsback, S. F. (2010). The ODD protocol: A review and first update. *Ecological Modelling*, *221*(23), 2760–2768

Grimm, V., Railsback, S. F., Vincenot, C. E., Berger, U., Gallagher, C., DeAngelis, D. L., Edmonds, B., Ge, J., Giske, J., Groeneveld, J., Johnston, A. S. A., Milles, A., Nabe-Nielsen, J., Polhill, J. G., Radchuk, V., Rohwäder, M.-S., Stillman, R. A., Thiele, J. C. & Ayllón, D. (2020). The ODD Protocol for Describing Agent-Based and Other Simulation Models: A Second Update to Improve Clarity, Replication, and Structural Realism. *Journal of Artificial Societies and Social Simulation*, *23*(2), 7

Hazy, J. K. & Tivnan, B. F. (2003). The impact of boundary spanning on organizational learning: Computational explorations. *Emergence*, *5*(4), 86–123

Hsieh, H.-F. & Shannon, S. E. (2005). Three approaches to qualitative content analysis. *Qualitative Health Research*, *15*(9), 1277–1288

Jann, W. & Wegrich, K. (2007). Theories of the policy cycle. *Handbook of Public Policy Analysis: Theory, Politics, and Methods*, *125*, 43–62

Janssen, M. A. & Ostrom, E. (2006). Empirically based, agent-based models. *Ecology and Society*, *11*(2)

Kauffman, S. A. (1993). *The Origins of Order: Self-Organization and Selection in Evolution*. Oxford: Oxford University Press

Lasswell, H. D. (1956). *The Decision Process: Seven Categories of Functional Analysis*. College Park, MD: University of Maryland

Levina, N. & Vaast, E. (2005). The emergence of boundary spanning competence in practice: Implications for implementation and use of information systems. *MIS quarterly*, *29*(2), 335–363

Lindblom, C. E. (1959). The science of "muddling through". *Public Administration Review*, *19*(2), 79

Lindblom, C. E. (1979). Still muddling, not yet through. *Public Administration Review*, *39*(6), 517–526

Maxwell, J. A. (2020). The value of qualitative inquiry for public policy. *Qualitative Inquiry*, *26*(2), 177–186

Meesters, K., Nespeca, V. & Comes, T. (2019). Designing disaster information management systems 2.0: Connecting communities and responders. ISCRAM

Meijering, J. (2019). Information diffusion in complex emergencies: A model-based evaluation of information sharing strategies. Available at: `https://github.com/JasperCM/information-diffusion`

Miles, M. B. & Huberman, A. M. (1994). *Qualitative Data Analysis: An Expanded Sourcebook*. Thousand Oaks, CA: Sage

Nespeca, V., Comes, T. & Brazier, F. (2022). Share: bottom-up disaster information management v1.0.0. *CoMSES Computational Model Library*. doi:$10.25937/3dbz\text{-}qv52$. Publisher: CoMSES.Net

Nespeca, V., Comes, T., Meesters, K. & Brazier, F. (2020). Towards coordinated self-organization: An actor-centered framework for the design of disaster management information systems. *International Journal of Disaster Risk Reduction*, *51*, 101887

Nikolic, I. & Ghorbani, A. (2011). A method for developing agent-based models of socio-technical systems. 2011 International Conference on Networking, Sensing and Control

Rao, A. S. & Georgeff, M. P. (1991). Modeling rational agents within a BDI-architecture. Proceedings of the Second International Conference on Principles of Knowledge Representation and Reasoning

Silver, A. & Matthews, L. (2017). The use of Facebook for information seeking, decision support, and self-organization following a significant disaster. *Information, Communication & Society*, *20*(11), 1680–1697

Slingerland, G., Edua-Mensah, E., van Gils, M., Kleinhans, R. & Brazier, F. (2022). We're in this together: Capacities and relationships to enable community resilience. *Urban Research & Practice*, *0*(0), 1–20

Starbird, K. & Palen, L. (2011). "Voluntweeters" self-organizing by digital volunteers in times of crisis. Proceedings of the SIGCHI Conference on Human Factors in Computing Systems

Turoff, M., Chumer, M., van de Walle, B. & Yao, X. (2004). The design of a dynamic emergency response management information system (DERMIS). *Journal of Information Technology Theory and Application (JITTA)*, *5*(4), 3

van de Walle, B. & Comes, T. (2015). On the nature of information management in complex and natural disasters. *Procedia Engineering*, *107*, 403–411

Van de Walle, B. & Comes, T. (2015). On the Nature of Information Management in Complex and Natural Disasters. *Procedia Engineering*, *107*, 403–411. doi:$10.1016/j.proeng.2015.06.098$

van Voorst, R. (2016). Formal and informal flood governance in Jakarta, Indonesia. *Habitat International*, *52*, 5–10

Watts, J. (2019). CHIME ABM Hurricane Evacuation Model v1.4.0. Publisher: CoMSES Computational Model Library

Watts, J., Morss, R. E., Barton, C. M. & Demuth, J. L. (2019). Conceptualizing and implementing an agent-based model of information flow and decision making during hurricane threats. *Environmental Modelling & Software*, *122*, 104524

Weiss, C. H. (1995). Nothing as practical as good theory: Exploring theory-based evaluation for comprehensive community initiatives for children and families. *New Approaches to Evaluating Community Initiatives: Concepts, Methods, and Contexts*, *1*, 65–92

Weiss, C. H. (1997). How can theory-based evaluation make greater headway? *Evaluation Review*, *21*(4), 501–524

Wooldridge, M. & Jennings, N. R. (1994). Agent theories, architectures, and languages: A survey. International Workshop on Agent Theories, Architectures, and Languages

Wooldridge, M. & Jennings, N. R. (1995). Intelligent agents: Theory and practice. *The Knowledge Engineering Review*, *10*(2), 115–152

Yang, L. & Gilbert, N. (2008). Getting away from numbers: Using qualitative observation for agent-based modeling. *Advances in Complex Systems*, *11*(02), 175–185

Yin, R. K. (2009). *Case Study Research: Design and Methods*. Thousand Oaks, CA: sage

Zack, M. H. (1999). Managing codified knowledge. *Sloan Management Review*, *40*(4), 45–58

Zagorecki, A., Ko, K. & Comfort, L. K. (2009). Interorganizational information exchange and efficiency: Organizational performance in emergency environments. *Journal of Artificial Societies and Social Simulation*, *13*(3), 3