Environmental Modelling and Software 207 (2027) 107211

Contents lists available at ScienceDirect

Environmental Modelling and Software

journal homepage: www.elsevier.com/locate/envsoft

Fourteen reusable building blocks (RBBs) for representing decision-making in agent-based models: a compilation on AGENTBLOCKS providing practical guidance and ways forward

Julia Kunkel [a,b,*], Volker Grimm [a,c], Christian Stindt von Dohm [d,e], Leon Grothmann [d], Luana Schwarz [f,g,h], Ronja Hotz [i], Robert Huber [j], Yongchao Zeng [i], Kaja Jurak [a,b], Uta Berger [k], J. Gareth Polhill [l], Liz Verbeek [m], Tatiana Filatova [m], Matthias Meyer [d,e], Birgit Müller [a,b,n], Jürgen Groeneveld [a,o]

[a] Helmholtz-Centre for Environmental Research - UFZ, Dept. Ecological Modelling, Permoserstraße 15, Leipzig, 04318, Germany
[b] Brandenburg University of Technology (BTU) Cottbus-Senftenberg, Modelling of Human-Environment-Systems, Erich-Weinert-Str. 1, Cottbus, 03046, Germany
[c] University of Potsdam, Plant Ecology and Nature Conservation, Zeppelinstraße 48A, Potsdam, 14471, Germany
[d] Hamburg University of Technology, Institute of Management Accounting and Simulation, Am Schwarzenberg-Campus 4, Hamburg, 21073, Germany
[e] United Nations University Hub on Engineering to Face Climate Change, Am Schwarzenberg-Campus 3, Hamburg, 21073, Germany
[f] Max Planck Institute of Geoanthropology, Integrative Earth System Science, Kahlaische Str. 10, Jena, 07745, Germany
[g] Potsdam Institute for Climate Impact Research, Earth Resilience Science Unit, Telegrafenberg A 31, Potsdam, 14473, Germany
[h] Potsdam University, Institute of Environmental Science and Geography, Karl-Liebknecht-Straße 24 – 25, Potsdam, 14476, Germany
[i] Institute of Meteorology and Climate Research, Atmospheric Environmental Research (IMK-IFU), Karlsruhe Institute of Technology, Garmisch-Partenkirchen, 82467, Germany
[j] ETH Zürich, Agricultural Economics and Policy, Sonneggstrasse 33, Zürich, 8092, Switzerland
[k] TU Dresden, Faculty of Environmental Sciences, Institute of Forest Growth and Computer Sciences, Dresden, 01062, Germany
[l] The James Hutton Institute, Information and Computational Sciences Department, Craigiebuckler, Aberdeen, AB15 8QH, UK
[m] Delft University of Technology, Multi Actor Systems Department, Faculty of Technology, Policy and Management, Jaffalaan 5, Delft, 2628BX, the Netherlands
[n] iDiv German Centre for Integrative Biodiversity Research Halle-Jena-Leipzig, Puschstraße 4, Leipzig, 04103, Germany
[o] Julius Kühn-Institute (JKI) Institute for Forest Protection, Erwin-Baur-Str. 27, Quedlinburg, 06484, Germany

ARTICLE INFO

Keywords:
Reusable building blocks
Decision-making
Socio-environmental systems
Modular coding
Agent-based modelling
Individual-based modelling
Model reuse

ABSTRACT

Agent-based models (ABMs) are widely used to study socio–environmental systems. By integrating environmental modelling with behavioural theories, they enable to analyse how decision-making processes shape environmental outcomes. To speed up model development and enable cumulative theorising, it has been proposed to modularise components relevant across applications as reusable building blocks (RBBs). This paper describes an example of adopting the idea in practice using AGENTBLOCKS, a new open access platform for publishing RBBs. We present the first coordinated interdisciplinary effort of eleven modelling groups to identify, extract, review, and publish 14 decision-making RBBs from existing ABMs using AGENTBLOCKS. From this compilation, we extracted a decision tree to identify and scope reusable model components. Using a mixed-methods approach combining RBB criteria assessment, thematic coding of editorial feedback, and contributor surveys, we suggest empirically informed practices and next steps to lower the barrier for reusable coding practices in socio–environmental applications.

1. Introduction

Agent-based models (ABMs) are widely used to investigate complex socio-environmental systems by capturing heterogeneity, feedback, and adaptive behaviour of individuals and groups within the environment (Filatova et al., 2013; Schlüter et al., 2019; Schulze et al., 2017). ABMs are used across many environmental applications, from purely ecological (Grimm, 1999), to land use (Matthews et al., 2007), in climate impact research such as on migration (Thober et al., 2018) and flood risk (Anshuka et al., 2022; Taberna et al., 2020), or in modelling of energy

* Corresponding author. Helmholtz-Centre for Environmental Research - UFZ, Dept. Ecological Modelling, Permoserstraße 15, 04318, Leipzig, Germany.
E-mail address: julia.kunkel@ufz.de (J. Kunkel).

https://doi.org/10.1016/j.envsoft.2026.107211
Received 8 April 2026; Received in revised form 11 September 2026; Accepted 12 October 2026
Available online 12 October 2026
1364-8152/© 2026 The Authors. Published by Elsevier Ltd. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).
Open access funding enabled and organized by Projekt DEAL.

systems (Castro et al., 2020). To speed up model development, enhance model comparability, and enable cumulative theory-building, the implementation of reusable building blocks (RBBs) and modular coding practices has been proposed (Bell et al., 2015; Berger et al., 2024), following up on various traditions and attempts to promote modular coding practices in different modelling communities (Section 2).

With the conceptualisation of Berger et al. (2024) and the launch of the AGENTBLOCKS repository (Filatova et al., 2025), https://www.agentblocks.org/, both a definition and a template and infrastructure for RBBs in the context of ABMs became recently available. Several example RBBs have been published at AGENTBLOCKS at its launch, yet the process of isolating such RBBs and documenting them has not yet been tested at scale. More importantly, following a series of conference workshops between 2021 and 2025 (2021 Volkswagen Foundation, iEMSs, 2022 Congress, Social Simulation Conferences 2022-25, Social Simulation Fest, 2023), it became clear that a practical guidance on deciding whether a specific functionality of a large complex model can be demarcated and described following the proposed concept and template, will be useful. This study aims to fill these gaps by providing the first systematic exercise to isolate RBBs based on the AGENTBLOCKS template from large ABMs across a wide range of disciplines and programming languages and by consolidating these experiences to develop practical guidance to ease the adoption of RBBs.

Towards this goal, we present a community effort of eleven modelling teams that apply the proposed RBB definition using the provided template and repository. We focus on decision-making mechanisms, as for those a particular need on standardised, well-documented and reusable components has been identified (Groeneveld et al., 2017; Grothmann et al., 2026; Magliocca et al., 2024; Wijermans et al., 2023). We present 14 RBBs, which span a range of socio–environmental modelling contexts, from ecological ABMs capturing animal behaviour (e.g., Groeneveld et al., 2024; Martin et al., 2013) to coupled human–environment systems addressing decision-making in land-use and conservation contexts (e.g., Huber et al., 2022; Hotz et al., 2026; Kunkel et al., 2026; Zeng et al., 2025). Based on developed RBBs and the experiences throughout their development, we answer the following research questions: RQ1) How does the application of RBBs benefit the ABM community? RQ2) How can the RBB definition proposed on AGENTBLOCKS be applied in practice? RQ3) What are lessons learned for developing and implementing RBBs? RQ4) What are next steps for advancing the RBB approach?

To answer these questions, we systematically documented the whole process of defining, implementing and publishing RBBs throughout this community effort. First, we derived potential benefits for the community from the characteristics of the published RBBs. Subsequently, we analysed the alignment of the RBBs with the suggested criteria (Filatova et al., 2025) of reusability, independence, compactness and meaningfulness to identify challenges in demarcating RBBs and provide guidance on their application in practice. We coded the editorial feedback provided through the review process to collect thematic challenges and emerging practices throughout developing and implementing RBBs. Drawing on a survey of contributors and community exchange, we identified promising next steps for further advancing the adoption of the RBB ideas among the ABM community. The obtained insights can also be helpful to related efforts in fostering reusable modelling practices in socio-environmental systems.

The paper is structured as follows: Section 2 summarizes the current state of the RBB concept and the AGENTBLOCKS repository design. Section 3 presents the organisation of the community effort, assessment criteria for the published RBBs, a coding scheme for the editorial feedback and the survey of contributors. Section 4 presents our results, focusing on what benefits the effort brought to the ABM community, what we learned about the concept of RBBs, what practices emerged from our effort, as well as what were contributors' motivation and experiences. A discussion and outlook of these results is given in Section 5.

2. Background

2.1. Reusable model code for simulation models

Reusing and sharing pieces of software code and following modular coding practices is a long-established principle in software engineering (e.g., Parnas, 1972; Krueger, 1992) and has been adopted in various modelling communities, including environmental modelling (Laniak et al., 2013). Commercial simulation environments such as AnyLogic and Simio provide libraries of reusable modelling components that can be assembled into a larger simulation model. Likewise, platforms such as NetLogo or GAMA (Taillandier et al., 2019) offer libraries of complete example models from various disciplines that users can adapt for their own applications. Reuse in simulation modelling thus encompasses attempts at different levels of granularity, ranging from individual software components and modelling objects to complete models.

Different strands of the simulation literature have discussed and proposed definitions and approaches to model reuse that vary in their granularity, representation, and intended purpose - ranging from fast and efficient model development, comparability of implementations, to theory development (e.g., Robinson et al., 2004; Balci et al., 2011; Tucker and Gross, 2013; Bell et al., 2015; Parker et al., 2008). At the same time, several challenges associated with model reuse have been discussed, including the actual time saving, incentives, technical barriers, the risk of creating "integronsters" (Voinov and Shugart, 2013) through the integration of independently developed modules, and reduced opportunities for learning when modelling is reduced to "plug-and-play" assembly (Belete et al., 2017; Berger et al., 2024; Robinson et al., 2004; Voinov and Shugart, 2013).

The specific idea of RBBs as presented in (Berger et al., 2024) emerged from discussions across a series of symposia on theory development, which can therefore be seen as their main motivation (Grimm et al., 2024; Lorscheid et al., 2019). This was inspired by existing practices in other fields than computer science. In particular, the widespread use of established and validated measurement scales in fields such as marketing and psychology have become established measures across studies (Bearden et al., 2011). Deviations from those standard measures typically require explicit justification (Teeluckdharry et al., 2021), which may enhance the reliability in measurements and reduce researcher subjectivity, while it enables comparability and cumulative knowledge development across studies (Hulland et al., 2026).

Next to the question of how to define RBBs, there is the question of which processes are worthwhile to create building blocks for as previous studies suggest that the actual reuse of available model components remains limited in practice (Vendome et al., 2020). Studies address this, by systematically analysing commonalities of existing models to identify recurring themes as candidate building blocks. For example, Schroeder et al. (2022) apply thematic analysis to 28 NetLogo COVID-19 models, while Cheng et al. (2023) examine 39 models of human migration and rumour spread. Both identify recurring themes and discuss how each might be transformed into RBBs.

Taken together, the literature reflects considerable diversity in what constitutes a reusable element of a simulation model and how such elements are identified, represented, and shared. Against this heterogeneous background, this study adopts the RBB approach operationalised by Filatova et al. (2025), which is based on the definition of Berger et al. (2024). This conception focuses on meaningful mechanisms and processes rather than purely technical software components or complete models. It further separates the generic description of a building block, including pseudocode, from software-specific implementations, facilitating its representation across programming languages and modelling platforms. Finally, RBBs are intended to originate from existing, published ABMs and to be contributed and reviewed by the modelling community. The following section describes how this concept is implemented in AGENTBLOCKS through RBB criteria, a template and a shared repository.

2.2. AGENTBLOCKS: A template and repository for RBBs in ABMs

In this study, we refer to the definition of Berger et al. (2024) and its further development by Filatova et al. (2025). Here, an RBB is conceptualised as an element of a larger complex model, that represents a particular mechanism or process that is relevant across many (agent-based) models and applications and, hence, can be verified and reused.

To operationalise this definition, the article by Filatova et al. (2025) introduces four practical criteria to determine whether a model component qualifies as an RBB. These criteria are briefly summarised below - for their full definitions and underlying considerations, we refer to Filatova et al. (2025). An RBB should be: 1) **Reusable** – the process has clear application across multiple modelling applications that justify efforts in sharing it 2) **Independent** – it has defined inputs and outputs, allowing it to be isolated from and integrated into broader models with minimal changes. 3) **Compact** – it is concise and transparent, balancing simplicity with usefulness beyond a single equation. (4)) **Meaningful** – it has a standalone theoretical interpretation and connects to domain knowledge, making it interpretable by experts outside the modelling community or across disciplines, and hence transferable to other models and applications. Alternatively, processes that have a technical benchmark, e.g. for model intercomparison or repetitive technical tasks, like initializing synthetic populations, are considered as 'meaningful'.

The AGENTBLOCKS platform provides a community-based infrastructure for the development, peer review, and publication of RBBs. The platform provides a structured template that guides contributors in extracting and documenting their building blocks consistently and transparently. Following this template, each RBB is composed of two main components (Filatova et al., 2025):

- **RBB Main Description**: a verbal, conceptual description of the underlying mechanism, theory or process captured by the respective block of code. Here each code module is described by clarifying its background and purpose, key concepts and definitions and a generic specification of inputs, outputs and the interaction level the RBB represents (e.g. agent-to-agent, agent-to-environment, etc.). The Main Description refers to the concept of computation-independent models as used in model-driven software engineering, and provides a formalisation of the RBB, independent of its specific code implementation (Miller and Mukerji, 2003).
- **Implementations**: one or more technical realisations of the RBB, including implementation-specific descriptions, a process flow chart, the position in the overall model, pseudocode, as well as the reusable code block, written in a specific programming language. This refers to the concept of a platform-independent as well as a platform-specific model in software engineering, as it includes formalisations of both the (programming language-independent, but context-specific) workflow of a certain implementation of the RBB, as well as the programming language- (or platform-) specific code implementation.

This dual-pillar structure enables contributors to grasp the general purpose of an RBB from the 'description' part without a need to delve into the details of the specific code implementation that may be ABM-specific. The dual-pillar system also offers alternative implementations of the same conceptual building block and facilitates comparison across modelling languages and application domains. RBBs in AGENTBLOCKS are further categorised by disciplinary domain (with multiple tags signalling interdisciplinary work), programming language, interaction level (e.g., single agent, multi-agent, agent–environment), and application-specific keywords, enhancing discoverability and reusability across communities.

To make sure that the RBBs have been tested and went through a thorough design circle, the platform requires that they originate from a fully functional ABM. This ABM should be published on a public repository (like COMSES.net) alongside material documenting its design and application, e.g. a comprehensive model documentation or an academic publication or preprint that can be linked to. The code block of the RBB should demonstrate the process as it is employed in this original ABM and its specific programming language, but should be extracted as a working example (see Supplementary Material S.5 for detailed requirements).

Besides the administrators, the AGENTBLOCKS repository integrates three types of roles: contributor, editor, and reviewer. As a contributor, one can upload and revise RBB main descriptions as well as implementations. Upon submission of an RBB, an editor is assigned to guide the review- and publishing-process of submitted RBBs. The editorial team is originally composed of the AGENTBLOCKS team, but guest editors may also be invited to guide submissions of a specific thematic domain.

3. Methods

Our overall process of applying the RBB concept across disciplines, learning to isolate an RBB from a complex model code, and deriving practical guidelines for future contributors combined several methods. The baseline builds an international community effort that was organised to facilitate the RBB development and systematic collection of data on the process (Fig. 1).

The community effort involved several roles (lanes). We assigned one group member (JK) as a guest editor to coordinate the overall effort, facilitate exchange and technical support, guide revisions, and manage the collection, publication, and evaluation of the RBBs and related data (upper lane). Contributors extracted components from their existing ABMs and revised their submissions based on feedback from the editor, the admins, and mutual peer-review (bottom lane). To promote exchange and a shared understanding of the process, contributors engaged in online meetings and peer-reviewed each other's work (mid lane).

Conceptually, the procedure can be divided into three project phases (columns). First, the initial set of RBBs was defined and developed. Second, these RBBs went through a peer-review and revision process prior to publication. Third, the collected data was evaluated, including the published RBBs, the editorial comments and the online survey on the contributors.

Throughout this process, multiple data sources were collected and evaluated using different methods: 14 RBBs have been developed and published (Section 3.1.). These published RBBs (blue) were assessed against the underlying criteria characterising an RBB, that is, reusability, independence, compactness and meaningfulness (Section 3.2); editorial feedback (purple) was thematically coded to identify recurring challenges and emerging practices from designing and implementing RBBs (Section 3.3); finally a contributor survey (green) was conducted to capture participants' motivations, benefits and workload (Section 3.4).

3.1. RBB development, peer-review and contribution

This study used the AGENTBLOCKS platform to develop and publish 14 RBBs on decision-making. The platform distinguishes between *RBB Main Descriptions* and its *Implementations* as explained in Section 2.2. Contributors could either contribute full RBBs, including both a *Main Description* and one or more *Implementations*, or only provide implementations to existing RBBs. The template of the AGENTBLOCKs repository served as the guiding instrument. For each field of the repository, it provides a short explanation on the expected input (see Supplementary Material S.5). For the selection of the processes presented in this study, we adopted a broad definition of decision-making. Contributors could submit RBBs that describe both human and animal behaviour.

For the peer-review process, we distributed the submissions among the authors. Following the reviewer guidelines of the AGENTBLOCKS

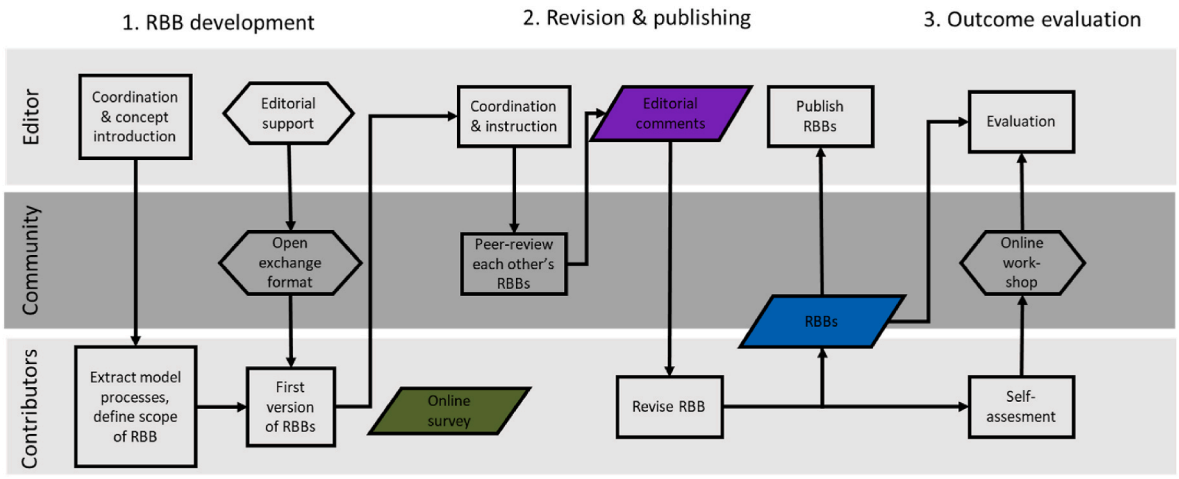


Fig. 1. Process flow chart illustrating the different roles and phases within the presented project. Grey lanes distinguish between the project roles of editor, community and contributors. Colors and shapes indicate the type of workflow. Rectangles represent processes, parallelograms represent processes that produce data for this article, and hexagons represent meetings and exchange. The colors distinguish the resulting data sources which are used for the evaluation process. Numbered headlines divide the workflow into project phases. (For interpretation of the references to color in this figure legend, the reader is referred to the Web version of this article.)

platform (Supplementary Material S.6), reviewers provided structured feedback on each contribution. The editor of the compilation (JK) reviewed both the contributions and the reviewer feedback, consolidated the comments, checked whether the requirements of the AGENTBLOCKS template and guidelines were addressed, and provided additional editorial feedback where needed to ensure consistent documentation and layout across contributions.

To support the community process and ensure consistent interpretation of the template several exchange formats took place. In a kick-off meeting the concept was introduced, including a hands-on session by the admins (LV and TF). Afterwards, a series of four optional online meetings was offered for open exchange. These sessions provided a space for discussing questions, sharing challenges, and aligning documentation standards. The editor of the compilation (JK) was present in each meeting to provide guidance and facilitate exchange and was additionally available for technical or content-related questions at any point during the development. The admins (LV and TF) consulted the editor and the group on tailored questions. Throughout the meetings, we developed rules to identify and scope an RBB, as presented in Section 4.2, and agreed on practices to address common challenges. These outcomes of the meetings have been shared with all contributors via e-mail as additional guidance and have been considered by the editor in the review process.

3.2. Assessment of RBB alignment with criteria

To determine the applicability of the RBB concept in practice, we assessed whether the published RBBs are in line with the operationalised RBB criteria of reusability, independence, compactness and meaningfulness, as introduced in Section 2. This assessment was conducted only after the RBBs had undergone the full publication process, including peer review. It served to evaluate the extent to which the existing guidance for contributors, reviewers, and editors on AGENTBLOCKs supported the development of RBBs that aligned with the four criteria underlying the RBB concept. Misalignment could thus indicate that a criterion was not sufficiently clear or operationalised in the existing guidance, or that the criterion itself may require further refinement. For each criterion, we therefore classified the alignment per RBB on an ordinal scale including a range of five levels: low, low-medium, medium, medium-high and high.

As the RBB criteria are formulated narratively in Filatova et al. (2025), i.e. do not specify thresholds for different levels of alignment, we first translated the criteria into an assessment rubric to guide the post-hoc assessment of contributions (for details, see Supplementary Material Table S2). For the reusability criterion, we assessed two dimensions: 1) whether the described process is generally applicable to other contexts than the provided implementations and 2) whether it has already been employed in different ABMs. The descriptions and references within the RBB main description served as the evidence for this evaluation.

For the independence criterion, we applied a three-step evaluation. First, we checked the implementation description to see if the RBB only covers a single mechanism. Secondly, we scanned if there are conceptual dependencies to the outer model, which need to be restructured when applying this to another context. Finally, we assessed the technical aspects of the code, such as how well it is encapsulated, and whether it has clear interfaces for inputs and outputs. This will allow the RBB to be easily integrated into another model from a technical point of view.

To assess the compactness of each RBB, we analysed several code-based metrics. Referring to the compactness criterion defined by Filatova et al. (2025), an RBB is expected to be longer than one equation but shorter than 10-100s of thousands of lines of code, so that it remains transparent and can be communicated and reused easily. To make this tangible for the contributions collected in our effort, we first quantified the size of each contribution by counting the source lines of code (SLOC), excluding blank lines and comment lines. Furthermore, we measured structural complexity as an indicator how easily one can grasp the code snippet, by calculating the cyclomatic complexity of each RBB, as commonly applied in software engineering to determine the complexity of code (MacCabe, 1976).

Here, cyclomatic complexity was defined as the number of binary control-flow elements (e.g., conditional statements and loops) and the number of functions defined within a code snippet (Mccabe, 1976). To avoid penalising RBBs that follow a modular or functional programming style - characterised by a higher number of small, clearly structured functions - we analysed the number of binary branches and the number of functions separately. This distinction allows us to differentiate between logical complexity arising from decision structures and structural modularity intended to improve readability and reusability.

To assess whether an RBB is meaningful, we scanned the RBB main description and the references herein for underlying theories and empirical grounding of the presented processes.

The assessment was conducted post hoc by the coordinator of the effort (JK), who applied the developed assessment rubric

(Supplementary Material, Table S2) consistently across all RBBs. Following the initial assessment, the contributors responsible for the respective RBBs were asked to review the assigned ratings and to flag potential inaccuracies. Their comments were subsequently reviewed by the coordinator, and ratings were adjusted where inaccuracies were identified.

3.3. Coding of editorial feedback on challenges in RBB contribution

To systematically evaluate the challenges and approaches to tackle them that emerged during the development and implementation of the RBBs, we analysed the editorial feedback which was provided throughout the review process. Each RBB submitted to the AGENTBLOCKS platform must undergo a structured peer-review process, following the instructions for RBB for reviewers (Filatove et al., 2025; see Supplementary Material S.6). After each review round, the editor (JK) issued comments that synthesized the reviewers' evaluations and included additional observations on the use of the template and the potential reusability of the building blocks to ensure consistency across submissions. Building up on these editorial comments, we conducted a thematic analysis to extract recurring challenges and strategies to overcome these.

We therefore applied an inductive coding approach, which means that coding categories were not predefined but developed based on recurring patterns in the comments (Braun and Clarke, 2006). Using these identified thematic challenges, each editorial comment was manually reviewed and tagged in MAXQDA 24 (VERBI Software, 2023). The analysis was conducted by the editor (JK), drawing on contextual insights gained throughout the review and revision process. The occurrence of each identified challenge across the 14 RBBs was subsequently quantified, and the associated challenges and suggested strategies were synthesized to identify recurring issues related to the RBB concept and its implementation in practice. For two contributions provided by the editor herself, the comments of the respective reviewers are used.

To ensure transparency and traceability of the analytical process, all coded segments and their thematic categorization are provided in the Supplementary Material (Table S4). The underlying feedback originated from multiple reviewers, who had already discussed recurring challenges and potential solutions during the online exchange meetings. These challenges were further discussed during the final online workshop, and the resulting thematic challenges and suggested practices were reviewed by all co-authors throughout the manuscript development process.

3.4. Survey on contributor experiences with RBB development

To collect the motivation, benefits and workload among contributors, we designed a short online survey. Each contributor was asked to fill this in after finalising and uploading a first version of the RBB to the AGENTBLOCKS platform. The timing prior to the editorial and review process was chosen to ensure that participants reflected solely on their personal experience with creating an RBB, without being influenced by reviewer or editorial feedback. Characteristics of survey participants, including modelling experience, number of published ABMs, programming languages used, and disciplinary background, are provided in the Supplementary Material (Table S1).

The survey combined structured and open-ended questions, capturing contributors' motivations, time investment and the perceived difficulty of key subtasks (e.g., conceptualisation, implementation, documentation), as well as a reflection on what was learned during the process (see Supplementary Material, Figure S1.). Quantitative responses were analysed descriptively, while qualitative comments were thematically grouped and summarised.

The survey involved only co-authors of this study and concerned their experiences in creating RBBs. All participants were informed about the purpose of the survey and the intended use of their responses in this publication beforehand. The survey involved no vulnerable participants, deception, or risks to participants. Only aggregated results are reported, preventing individual responses from being attributed to specific participants and thereby minimising potential professional or reputational risks. Based on the applicable guidance of the German Research Foundation (DFG), formal ethical review was therefore deemed not required.

4. Results

In the following, we present the results of this community effort, which aims to contribute to available RBBs on decision making as well as practical guidelines on how to adopt the concept of RBBs in practice. To this end, we first present the RBBs compiled in this study and what this enables for the ABM community (Section 4.1). Second, we reflect on ambiguities and tensions regarding the definition of an RBB and how we tackled this in the practical process of demarcating RBBs (Section 4.2). Third, we share our challenges in developing and implementing RBBs and propose practices that emerged useful during this community effort (Section 4.3). Finally, we examine contributors' motivations and experiences, focusing on the factors that shape their willingness and ability to extract mechanisms from their models and make them available as RBBs, as well as opportunities to lower barriers for future contributors (Section 4.4).

4.1. What did we contribute to the ABM community?

As a result of the coordinated community effort, a diverse collection of decision-making RBBs has been published on the AGENTBLOCKS platform (Table 1, Fig. 2). These contributions span multiple disciplines, interaction levels, and programming languages, and include both novel RBBs, including RBB Main Descriptions and Implementations, as well as alternative implementations of existing RBBs. This structured diversity provides new opportunities for reuse, comparison and theorising, as well as progressing within the ABM community:

First, we provide a curated and documented set of ready-to-use decision-making RBBs. By extracting commonly used mechanisms from full ABMs and formalising them within a shared template, this has the potential to reduce redundancy in model development and lower entry barriers for researchers seeking to implement decision-making processes in socio-environmental systems. While RBBs are not intended as fully plug-and-play components – their reusability depends on the fit between interaction level, discipline and context, as well as programming language – they provide a structured and transparent starting point to integrate established mechanisms into new models.

Second, the inclusion of multiple RBBs representing similar or identical theoretical mechanisms can potentially facilitate alternative and cumulative theorising. Presenting alternative behavioural mechanisms for the same decision process (e.g., profit maximisation and cumulative prospect theory) provides modellers with the opportunity to explore and test alternative behavioural theories in their own modelling context (alternative theorising), e.g. via intercomparison studies. As illustrated in Fig. 2, the occurrence of multiple decision-making RBBs at the same levels of interaction and from similar disciplinary backgrounds further points to potential candidates for such intercomparison studies, although their actual interchangeability was not assessed here.

Furthermore, different implementations of the same theory (e.g., Descriptive Norm and Norm Sensitivities) can be examined side by side more easily in the format of an RBB, rather than embedded in case-specific models. This comparability is facilitated by the repository's dual structure, which distinguishes between a shared, computer-independent main description of the described mechanism and multiple platform-specific implementations. This presentation may provide a starting point for discussing modelling assumptions and design choices which could ultimately contribute to advancing the formalisation of behavioural theories (cumulative theorising).

Table 1
Overview of the RBBs contributed throughout this study. In order to refer to the RBBs throughout this study, we use abbreviated forms, indicated in bold.

RBB	Level of Interaction	Main Discipline	Contribution	Language	Authors & Link
Activity Dissimilarity Index	Multiple agents (different types)	Sociology	Main + implementation	R	Huber & Gadhammar: https://www.agentblocks.org/rbb/activity-dissimilarity-index
Attitude formation through relational learning process	Agent(s) and the environment	Psychology	Main + implementation	Python	Schwarz et al.: https://www.agentblocks.org/rbb/attitude-formation-through-relational-learning-processes
Cumulative Prospect Theory	Single agent	Economics	Main + implementation	R	Kunkel & Finger: https://www.agentblocks.org/rbb/cumulative-prospect-theory
Demand driven foraging decision in honey bees	Single agent	Ecology	Main + implementation	NetLogo	Groeneveld & Becher: https://www.agentblocks.org/rbb/demand-driven-foraging-decision-in-honey-bees
Descriptive Norm and Norm Sensitivities	Multiple agents (same type)	Sociology	Main + implementation	NetLogo	Stindt et al.: https://www.agentblocks.org/rbb/descriptive-norm-and-norm-sensitivities
			Implementation	NetLogo	Hotz: https://www.agentblocks.org/rbb/descriptive-norm-and-norm-sensitivities
Dynamic Energy Budget (DEB) theory for animals	Single agent	Ecology	Main + implementation	NetLogo	Grimm et al.: https://www.agentblocks.org/rbb/dynamic-energy-budget-deb-theory-for-animals
Endogenous Institutional Model (EIMOD)	Agent(s) and the environment	Governance	Main + implementation	Java	Zeng: https://www.agentblocks.org/rbb/endogenous-institutional-model-eimod
Hegselman-Krause Opinion Dynamics of Social Influence	Multiple agents (same type)	Sociology	Main + implementation	Java	Kunkel: https://www.agentblocks.org/rbb/hegselman-krause-opinion-dynamics-of-social-influence
Heuristic Satisficing, Experimenting and Imitative Decision-Making Algorithms	Single agent	Psychology	Main + implementation	NetLogo	Polhill & Gotts: https://www.agentblocks.org/rbb/heuristic-satisficing-experimenting-and-imitative-decision-making-algorithms
Profit Maximisation	Single agent	Economics	Main + implementation	R	Jurak: https://www.agentblocks.org/rbb/profit-maximization
Theory of Planned Behaviour for individual decision-making	Single agent	Psychology	Implementation	Python	Schwarz et al.: https://www.agentblocks.org/rbb/theory-of-planned-behavior-for-individual-decision-making
Zone-of-influence (ZOI)	Multiple agents (same type)	Ecology	Main + implementation	NetLogo, Python	Piou et al.: https://www.agentblocks.org/rbb/zone-of-influence-zoi
			Implementation	Python	Piou et al.: https://www.agentblocks.org/rbb/zone-of-influence-zoi

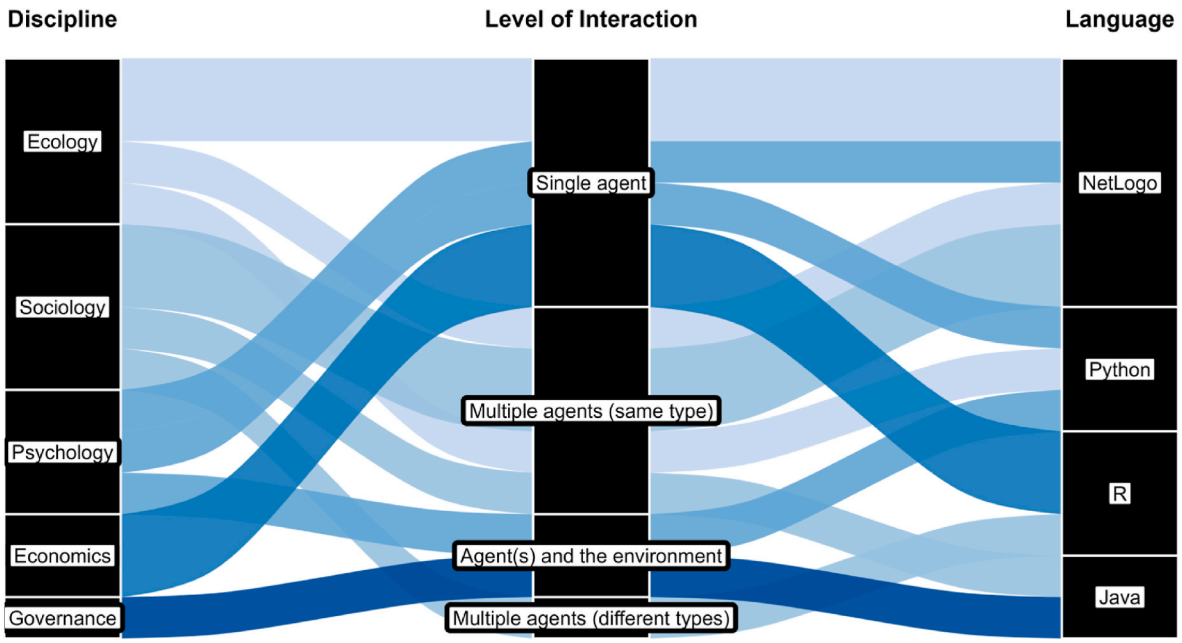


Fig. 2. Overview of the 14 examples of RBB across the three dimensions: main discipline, level of interaction and programming language of the provided code. Colour shade indicates discipline. Size of blocks and bars reflect the number of contributions to which this category applies to. (For interpretation of the references to color in this figure legend, the reader is referred to the Web version of this article.)

Third, the diversity of contributions allowed us to test the RBB concept and its infrastructural implementation across heterogeneous modelling contexts. This large-scale application exposed structural tensions, clarified ambiguities in the implementation process, and ultimately enabled the derivation of practical guidelines and concrete next steps, which we detail in the following sections.

4.2. What did we learn about the RBB concept?

During the extraction and formalisation of decision-making processes as RBBs, contributors followed the template and instructions on the AGENTBLOCKS repository, which build on the four criteria operationalised by Filatova et al. (2025): reusability, independence, compactness, and meaningfulness. In the following, we present the

alignment of the contributed RBBs with these criteria after peer-review and publication (Fig. 3). In this way, we can identify those criteria, for which we did not reach a high alignment, despite the RBBs having been implemented, peer-reviewed and published in following the AGENT-BLOCKs repository. We further present the assessment for each criterion and discuss underlying procedural and conceptual challenges that potentially caused this gap between the practical implementations and the underlying concept. As an output from these results, we developed a guidance for future contributors, reviewers and editors in the form of a decision tree that supports assessing whether a process (i) qualifies as an RBB, (ii) should be represented as a single building block, or (iii) should be split into multiple RBBs (Fig. 4).

For the reusability criterion, the alignment of the published RBBs was mostly high (Fig. 3). This suggests that the existing contribution and review process generally resulted in RBBs that met the intended reusability criterion. Most of our contributed RBBs capture generic modelling tasks – such as social influence, decision-making under risk, or resource allocation – that recur across a wide range of ABMs and application domains and are grounded in well-established theories or modelling traditions (e.g., Theory of Planned Behaviour, Descriptive Norm, or Prospect Theory). In such cases, extensive prior use of such processes already exists in modelling literature and is often accompanied by scholarship from outside modelling, e.g. via empirical data that supports the parameterisation of RBBs across diverse contexts. This suggests a rather homogeneous interpretation of reusable components amongst the contributors in this effort. Yet, we see reusability as a critical criterion, as producing an RBB that is not reusable takes up time of contributor, reviewer and editor, while potentially clogging the repository.

Lower alignment with the reusability criterion mainly occurred for context-specific processes (e.g., Foraging specifically for honey bees) or more innovative modelling approaches (e.g., Endogenous Institutions). Context-specific RBBs are closely tied to particular application domains, like behaviour of specific species in ecology, which can be reused in other ecological models of bees while their transfer to other applications is hindered. In turn, innovative modelling approaches, like modelling decisions of an institutional policy maker, though highly demanded in general in environmental modelling (Peng et al., 2021), are not yet widely adopted across models. This suggests they demand more time to demonstrate and verify their reusability and/or utility across applications. Nevertheless, when such contributions articulated potential application domains beyond the original model, we considered them suitable RBB candidates, as their formalisation may facilitate broader uptake and further development of these processes. Based on these observations, we make two suggestions: i) Verify reusability at an early stage in the RBB development process to avoid investing unnecessary time. ii) Interpret reusability broadly to cover existing and potential applications for reuse (see Fig. 4).

For the criterion of meaningfulness most of our contributions are grounded either in established theories (e.g., Social Norm Theory or Theory of Planned Behaviour) or in well-recognised stylized ecological processes (e.g., zones of influence or foraging behaviour), leading to high alignment (Fig. 3). Again, lower alignment was primarily reached where contributors proposed context-specific or innovative operationalisations of broader concepts (e.g., satisficing implemented via heuristic decision-making or similarity theory implemented via dissimilarity index), or where the underlying concepts are well established but more abstract and difficult to translate directly into empirical observations (e.g., Opinion dynamics). Purely technical processes have not been among the submitted contributions in our set. This aligned with the intended focus on theory-, concept-, or empirically grounded RBBs. To ensure alignment with the criterion of meaningfulness as proposed by Filatova et al. (2025), we suggest actively assessing whether an RBB has a meaning outside of the modelling world throughout the contribution and peer-review process (Fig. 4). This distinction intends to exclude purely technical modules or helper functions where reusability relates primarily to technical questions, such as implementation efficiency. Such elements may therefore be better suited as extensions, libraries, or programming-language-specific packages rather than RBBs as characterised by Berger et al. (2024).

With respect to independence, the alignment of published RBBs was mixed. Several RBBs either consist of several subprocesses or show conceptual interlinkages to the outer model. This indicates challenges in defining the scope of an RBB and demarcating its boundaries. Already throughout the submission and review process, the scoping of an RBB turned out as a major challenge for contributors (see also Section 4.3). To decide if a decision-making process should constitute a single or several RBBs, we therefore developed and applied the following decision guidance (Fig. 4): Any subprocess that is meaningful and reusable in its own regard or that offers several conceptually different formalisations should constitute its own RBB. For example, the formation of attitudes is a subprocess of the theory of planned behaviour. However, there is an own branch of theories on how to form attitudes and this process is also integrated in other modelling contexts, such as in the Technology Acceptance Model (Davis, 1985). We therefore decided that each theoretical branch of attitude formation (e.g., through learning) should constitute its own RBB.

In contrast, we suggest that several subprocesses should be

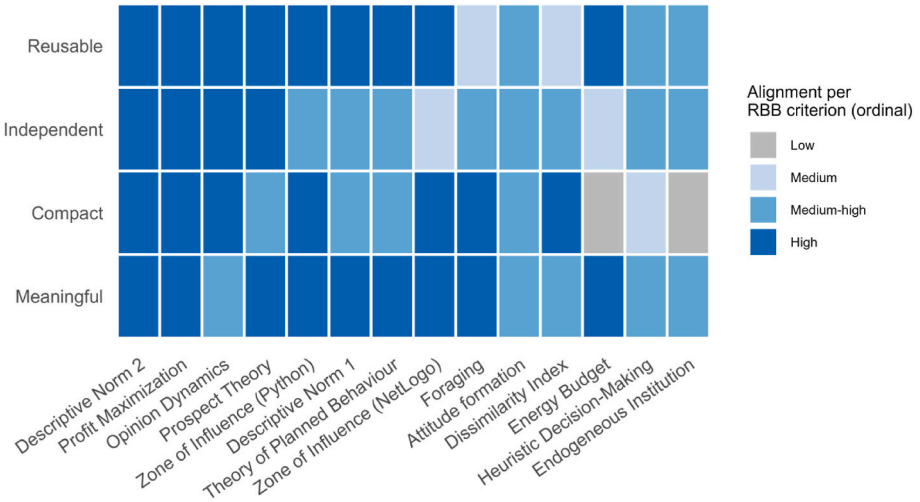


Fig. 3. Alignment of each published RBB contribution with the criteria defined in Filatova et al. (2025). The detailed assessment of each contribution along these dimensions is presented in the Supplementary Material Table S3.

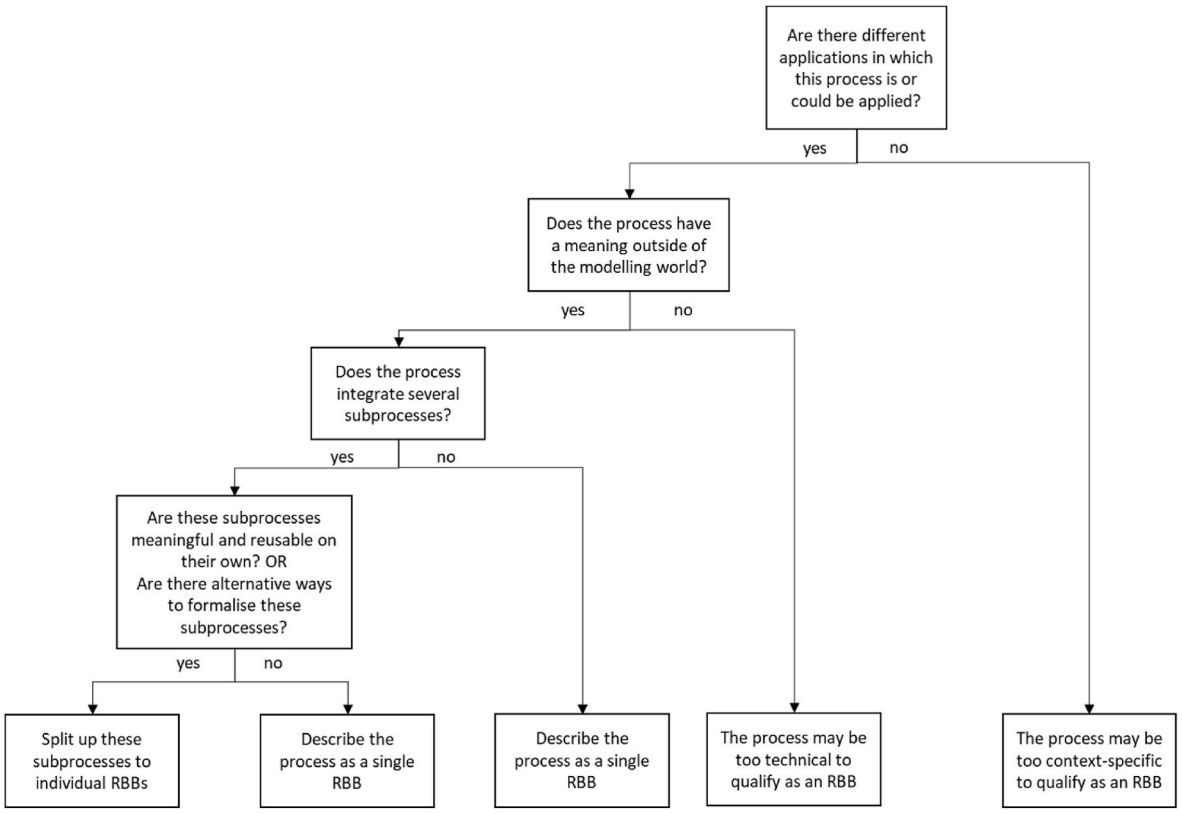


Fig. 4. Decision tree for identifying and scoping RBBs. It guides contributors in assessing whether a mechanism qualifies as a RBB and whether it should be represented as a single RBB or decomposed into multiple building blocks. Boundary examples this tree emerged from, and which can be used to demonstrate how to apply this decision tree, are presented in the Supplementary Material S.3.

integrated as one RBB if they are reusable and meaningful solely in combination. An example is the Energy Budget RBB. This RBB contains several subprocesses, such as the maturation or reproduction of individuals. However, these subprocesses are explicitly tailored to the overall process of how individuals allocate their energy budget and are tightly interlinked, so that the single subprocesses are not reusable outside of the energy budget theory. No alternative formalisations were identified for these subprocesses within the scope of this contribution that would justify presenting them as separate RBBs.

The assessment in Fig. 3 shows that applying this decision rule did not consistently result in high alignment with the independence criterion. This reveals a trade-off between the RBB criteria. Splitting every submodel into the smallest possible RBBs to maximise independence may result in components that are no longer meaningful or reusable on their own and may unnecessarily increase the number of RBBs in the repository. Conversely, retaining too many interconnected subprocesses within a single RBB may reduce independence and make the RBB more similar to a case-specific submodel rather than a reusable mechanism. Our decision rule aims to balance this trade-off and can provide guidance for future contributors facing similar boundary decisions. The boundary cases encountered during this effort and the subsequent development and application of the decision rules are presented in the Supplementary Material S.3.

Another source of misalignment with the independence criterion was a lack in technical independence of the published RBBs. Certain structural assumptions were inherent to most ABMs and could not be fully removed, e.g., the types of agents involved (e.g., individuals, superorganisms, or institutions) or the characteristics of the surrounding model environment (e.g., social or spatial networks, grid-based environments, time steps). Therefore, complete technical independence could not be achieved. However, as RBBs are not supposed to be plug-and-play model components (Berger et al., 2024), a transparent documentation of the structure and modular coding practices could be sufficient to fulfill this criterion. The AGENTBLOCKS platform already provides dedicated fields to indicate which inputs and outputs need to be provided, and most importantly – their conceptual meaning and implications when integrating an RBB to another model. A consistent implementation of this on the code-basis — especially providing the RBB as an encapsulated module with a clear interface for inputs and outputs — will increase the independence and should be thoroughly checked by the reviewers as suggested in the review guidelines.

For the criterion ‘compactness’, we obtained mixed alignment across RBBs, while the reasons for lower alignment differed. Some implementations contained comparatively extensive code but implemented conceptually straightforward mechanisms (e.g., Prospect Theory). Others were syntactically concise yet included many conditional branches and intertwined logical steps (e.g., Attitude formation). Additionally, the assessment revealed an interdependence with the decision rules set up for scoping an RBB above. If RBBs with many subprocesses are retained as single RBBs – as they only form a meaningful, reusable process in this way – this potentially leads to quite long and complex code (e.g. Energy Budget Theory). Therefore, we did not exclude or split up any RBB due to overly long or complex code, unless they comprised stand-alone reusable and meaningful mechanisms. Subsequently, we suggest not splitting RBBs into their subroutines solely based on code length or complexity and therefore did not include this as a separate branch in the decision tree.

Our results suggest that the AGENTBLOCKs repository generally supports the implementation of RBBs that align with the underlying criteria defining an RBB according to Filatova et al. (2025). A main challenge remains in identifying the scope and demarcating an RBB from a larger model that aligns with the RBB criteria. The decision tree presented in Fig. 4 summarizes the key takeaways regarding the criteria and the scoping of RBBs obtained in this study. It is supposed to serve as a

practical decision tool for future RBB development. Boundary cases from which these decision rules evolved can serve as examples on how to practically apply this tool, but also as basis to further discuss and specify what constitutes an RBB (see Supplementary Material S.3). We suggest no changes in the criteria, but a hierarchy among them. Reusability and meaningfulness emerged as the core criteria to be verified first: if these are not fulfilled, providing a module as an RBB does not appear useful for the community. Independence, in contrast, could be improved through re-scoping and re-structuring of RBBs. This indicates that additional guidance in the RBB concept and the adoption of more modular coding practices could substantially improve contributions. Finally, compactness appeared to function primarily as an initial diagnostic indicator, prompting a closer assessment of the other criteria, rather than as a standalone condition that would disqualify a contribution from being considered an RBB.

4.3. What suggestions for developing and implementing RBBs emerged?

During the process of developing and implementing RBBs, several recurring challenges emerged. The feedback from the editor and the admins that was produced during the peer-review process provides valuable insights into which challenges occurred and which strategies were developed to address them. By systematically coding the editorial feedback, we identified seven recurring thematic challenges together with suggested practices developed during the review process (Table 2).

Fig. 5 shows how many contributions encountered each thematic challenge throughout the review process. Content clarity emerged as the most pervasive challenge. Many submissions did not sufficiently distinguish between theoretical concepts, their formalisation, and model-specific implementation choices. Using the implementation description field to explain how generic key concepts (main description tab) are formalised in the specific use case proved helpful in addressing this issue. In addition, following basic documentation standards - such as using consistent terminology across text, figures, and code - further improved clarity.

In addition to content clarity, code reusability constituted a recurring challenge across nearly all contributions (Fig. 5). A frequent issue was the lack of explicit documentation of assumptions and context-specific elements that would need to be adapted by end users of the RBB. Improving tractability by referring to how parameters or context-dependent functions were selected, and improving adaptability by clearly indicating which parts of the code must be modified when applying the RBB in another model, were identified as key approaches to address this challenge.

Applying the template of the AGENTBLOCKS repository revealed several ambiguities. This has occurred despite testing the template across multiple interdisciplinary workshops between 2022 and 2025, and already simplifying the first version of the template (Berger et al., 2024). On the one hand, it has to do with the human nature of having different mental models - especially across disciplines - on what the standard way of describing something should look like. Consequently, providing a metadata following a specific standardized structure always needs time to get used to. On the other hand, contributors frequently assigned content to inappropriate fields, omitted required elements, or interpreted template components inconsistently. Over the course of the review process, a more consistent understanding of the requirements for the RBB description and visualisation elements emerged, which can now serve as guidance for future contributors. Notably, since our effort was the first systematic test of the RBB concept after the launch of the beta-version of the AGENTBLOCKS repository, the admins have been working closely with the group and revised the platform. For example, the template was extended to include a choice between a permissive (MIT) and copyleft (GPL, LGPL) license and to explicitly attribute copyright and authorship independent from each other.

From a technical perspective, providing a working code snippet proved to be a non-trivial task. While the AGENTBLOCKS repository does not impose strict requirements regarding what constitutes working code, we aimed to ensure that each RBB includes an executable minimal example to demonstrate its functionality and facilitate reuse to other models. Encapsulating the RBB within a main function that provides a clear interface for inputs and outputs proved particularly effective in simplifying reuse. To provide a minimal runnable example without introducing unnecessary complexity, dummy functions and data were used where appropriate. In addition, clear documentation of parameters that need to be adapted, as well as of external libraries and packages, proved crucial for enabling executability. For NetLogo implementations, additional documentation or refactoring is necessary to include those parameters and functions that are defined through the graphical interface also in the code block provided in the repository.

Finally, several challenges related to the scope and dual-pillar structure of an RBB occurred. Given the difficulties with demarcating an RBB from the original complex ABMs, some submissions defined RBBs that were too broad, covering entire models or multiple submodules, while others were so context-specific that reuse beyond the original application was limited. As a result of such a large spread, some contributors struggled to separate the generic underlying concept of an RBB from implementation-specific details. This mainly has to do with imagining generic usage of a proposed RBB outside the original model. To address the first issue, we developed a decision tree to guide the identification of an appropriate RBB scope, as introduced in the previous section (Fig. 4). To address the second issue, we formulated guiding questions helping to distinguish the main description and the implementation part of an RBB (Table 2). Applying these guidelines, two RBBs with implementations of different contributors could be published (Theory of Planned Behaviour and Descriptive Norm), which can serve as examples, showcasing how the dual-pillar structure can separate a generic description accessible for wide audiences from a particular code implementation as suggested by best practices in software engineering and intended by the developers of the concept for RBBs (Berger et al., 2024; Filatova et al., 2025).

4.4. What were contributors' motivations and experiences?

While the preceding sections focused on how to tackle conceptual, technical, and infrastructural challenges throughout the RBB development and implementation, the broader uptake of both the RBB concept and the AGENTBLOCKS repository ultimately depends on individuals' motivation, perceived benefits and available time to extract mechanisms from their models and make them available as RBBs. To better understand how these factors apply here, we analyse results from a contributor survey examining motivations for participation, perceived benefits, and the time required to develop RBBs (Fig. 6). Based on these insights, we identify opportunities to lower barriers for future contributors and strengthen community engagement around the RBB concept.

Enhancing the reusability of code was a key motivation for all participants (Fig. 6a). Many respondents (10 out of 12) also reported a desire to improve modelling standards and to support more efficient development of ABMs (9 out of 12). Traditional academic incentives in terms of citations were less central but still relevant for three contributors, such as contributing to a joint publication, and increasing the visibility of their research. Additionally, motivations covered under "group dynamics" were mentioned, including opportunities to exchange with others or receive feedback on modelling choices.

Participants further reported several learning effects resulting from the RBB development process (Fig. 6b). Eight participants indicated that engaging with other contributions exposed them to alternative ways of implementing similar processes. Half of the contributors also noted improvements in their own code quality and a deeper understanding of the underlying theory because of defining and documenting the RBB. Respondents also highlighted that the process helped them to more clearly separate decision-making mechanisms from other model components and to adopt the perspective of an external user when

Table 2
Coding scheme applied to the editorial feedback of each contributed RBB and suggested practices. Challenges are sorted according to how often they occurred throughout the editorial feedback. Note: coded segments (n = 147) refers to text passages of the feedback that referred to one of these challenges. All coded segments are provided in the Supplementary Material Table S4.

Challenges	Explanation	Example	Suggested practices
CONTENT-CLARITY	Insufficient explanation of concepts, context or inconsistency of contents between different fields	"Your overview figure, includes terms, which are not included in your background description"	**Implementation Description:** Use this field to explain how the implementation formalises the key concepts from the main description (e.g., as a "written flow chart"). Provide equations where these are central to the process, and structure complex processes into clear steps that can be referenced in the (pseudo-)code. **Documentation standards:** Use consistent terminology across text, figures, and code (including comments). Provide an overview figure even if it is optional. Indicate potential application domains and clearly distinguish which formalisations or algorithms are taken from the literature and which elements are newly introduced.
REUSABILITY	Missing instructions for re-use (parameterisation, assumptions, examples)	"How do you come up with the specific numbers and thresholds? Are these specific to your context and need to be adapted by the end-users?"	**Tractability:** Provide references or explanations for how variables were parameterised or how specific formalisations were chosen (implementation description). Describe each parameter, variable, or function of the code either in the description or in metadata. Document alternative options to formalise processes and explain how you decided between them. **Adaptability:** Highlight in the code instructions and the code itself which parameters or functions (beyond the main inputs) are context-specific and need to be adapted when the RBB is applied in other contexts or models.
TEMPLATE-USE	Misunderstanding or misuse of template fields or structure	"What is in your Pseudocode, is more what is expected in "main > implementation description" field."	**Template fields:** Read the description provided for each template field carefully. Use the implementation description to provide details about the implementation (not more than one page). Note the difference between a flow chart (describing the internal logic of the RBB) and a sequence diagram (showing the position of the RBB within the overall model). **Technical hacks:** Avoid unsaved changes: every pop-up window and tab must be saved before switching. The presentation of formulas can be improved by adjusting the HTML under the "source code" button (large language models can assist with this). **Licencing & Authorship:** Decide between a permissive license (MIT) and a copyleft license (GPL or LGLP). Provide authors for each implementation; the overall RBB authorship will include the authors of all implementations.
CODE-EXECUTABILITY	Problems providing a working example of the code	"Please explain under code instructions, which sliders or inputs must be set with which ranges in the Netlogo interface to run your code"	**Encapsulation:** Use a main function or a functional/class-based coding style with clear interfaces for model inputs and outputs. Define parameters and variables in one place and avoid hard-coded values throughout the code. **Runnable:** Provide a working runnable (dummy) example. Random values or placeholder data can be used, but clearly indicate which assumptions are only made for demonstration purposes. **External dependencies:** If external libraries or packages are required, provide detailed instructions for integration (in the code instructions) and state the versions used. **NetLogo GUI:** If a NetLogo RBB relies on elements defined via the GUI (buttons, sliders), three options exist: i) Use comments and code instructions to describe what must be set up in the interface (e.g., Descriptive Norm RBB). ii) Define globals in the code and provide a command-line statement to run the code (e.g., Foraging RBB). iii) Paste the full NetLogo source code (open the. nlogo file in a text editor) including the GUI (e.g., Energy Budget RBB).
ABSTRACTION-SCOPE	Unclear level of abstraction/generalisation, missing applicability of RBB to other contexts	"You upload a whole model, which seems to also include submodules and processes, which […] go beyond an RBB"	**RBB scope:** Use the decision tree in Fig. 4 to determine whether a component should be represented as an RBB and whether it should be described as one RBB or split up to several RBBs.
VISUALISATION	Suggestions to improve overview figures, flow charts, sequence diagrams	"Although the flow chart is understandable, it would be clearer if some modifications could be made according to flow charting conventions"	**Level of detail:** Flow charts should include all content-related processes and decision branches of the implementation. Clearly indicate inputs and outputs in both overview figures and flow charts. Use the figure descriptions in case of ambiguities. **Visualisation standards:** Use the suggested figure templates and follow common conventions for flow charts. Align figures with their textual descriptions.
ASSIGN-TO-TAB	Assigning content to the wrong tab (main description vs. implementation) of the template	"Therefore, please remove all implementation specific information to the "Implementation …"	**Guiding question for main description:** Does this description (including terminology) also hold for applications of this theory in other models and contexts? **Guiding question for implementations:** Is it clear how this implementation is formalised, why it is formalised in this way and how this can be included and adjusted to fit another modelling context?

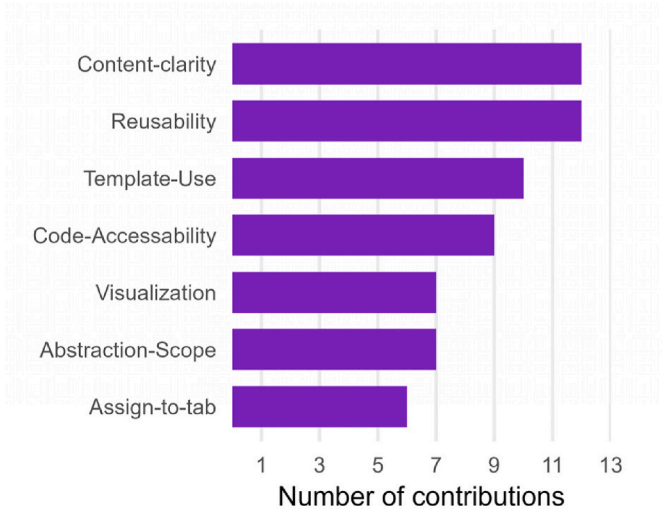


Fig. 5. Number of contributions in which the respective challenge was identified during the review process at least once. Note: n = 13, as the ZOI RBB with two implementations was submitted and reviewed as one contribution.

structuring their code and documentation. These results reveal substantial learning effects for individual contributors, also motivating the use of RBBs in teaching contexts.

The initial working time for extracting an RBB from an existing model and preparing the corresponding description required on average two and a half working days (Fig. 6c). Variations reflect differences in prior model structure and modularity, and subsequently, the complexity of separating decision-making processes from surrounding model components. Participants reported that the most time-consuming tasks were defining the appropriate scope of the RBB, restructuring code to function as a standalone unit, and designing ways to handle complex inputs and outputs. Across all contributions, one to two revision rounds were required before acceptance. Peer exchange was perceived as helpful throughout the whole process of the RBB development.

5. Discussion & outlook

This study contributes to ongoing efforts to advance reusable coding practices in socio-environmental systems modelling. Through a coordinated community effort involving eleven modelling teams, we used the AGENTBLOCKS template and repository (Filatova et al., 2025) - based on the RBB concept of Berger et al. (2024) - to identify, document, review, and publish reusable components derived from existing models. Moving from conceptual and technical developments towards practical application, we provide a collection of RBBs and an empirically grounded evaluation of their operationalisation, including the challenges encountered in this process.

By covering RBBs that represent similar or identical theoretical mechanisms across different application domains, the collection facilitates comparison of alternative decision-making processes and conceptual variants of the same process. Such comparisons can support modellers in justifying the fit of particular decision-making representations to a given socio-environmental context (Wijermans et al., 2023) and, in the longer term, provide a basis for alternative and cumulative theorising (Grimm et al., 2024). In this way, decision-making RBBs complement existing resources, including code of complete ABMs published in repositories such as CoMSES.net and conceptual frameworks of decision-making (Schlüter et al., 2017), by facilitating comparison of the assumptions, inputs, outputs, and implementations of individual decision-making processes.

We assessed the published RBBs, developed following the repository template, against the underlying RBB criteria. This assessment revealed important interdependencies among the four criteria. Reusability and meaningfulness emerged as exclusion criteria: if a model component is not useful beyond its original model context or does not represent a meaningful process or mechanism, providing it as an RBB offers little added value. In contrast, independence and compactness did not emerge as stand-alone exclusion criteria. Strictly separating dependent subprocesses may result in numerous small components that are less meaningful or useful in isolation, while reusable and meaningful mechanisms may comprise dependent subprocesses and be less compact in terms of code length and complexity. Maximising independence and

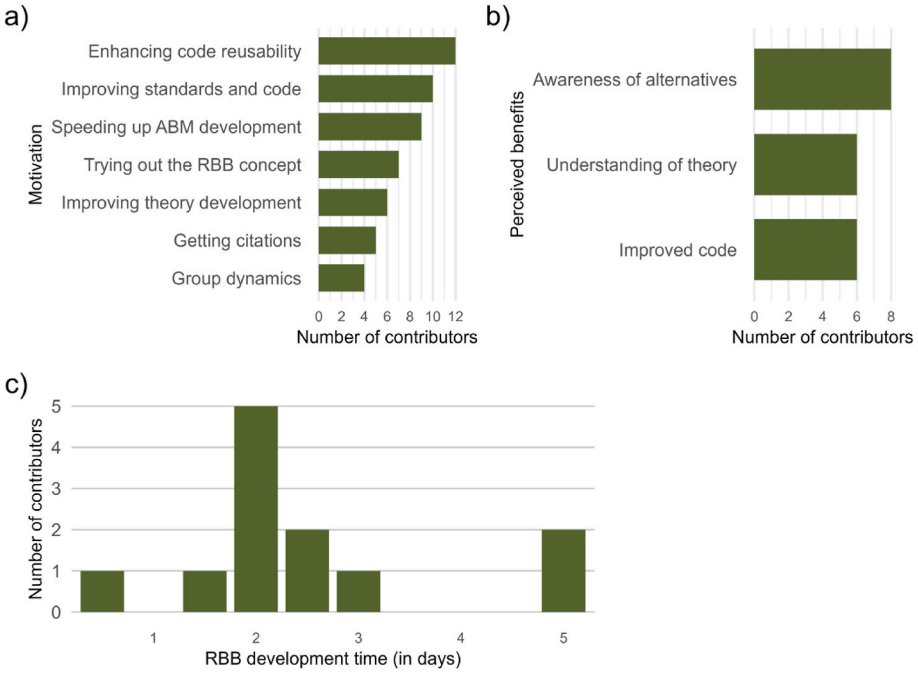


Fig. 6. Results from the contributor survey regarding their: a) Motivation, b) Benefits and c) Workload to contribute an RBB until first submission including group calls and discussion. Number of respondents n = 12 (as some modelling teams provided two RBBs). Characteristics of the survey participants are summarised in the Supplementary Material (Table S1).

compactness may therefore conflict with preserving meaningfulness and reusability. Based on these findings, we propose a hierarchy among the four RBB criteria, summarised in a decision tree (Fig. 4) as practical guidance for future contributors. This prioritisation should be further discussed and tested as the RBB concept is applied by the wider modelling community.

At the same time, our practical application revealed that demarcating and implementing modelling processes as RBBs is non-trivial. We document various challenges that contributors encountered throughout the process. By offering working examples and suggesting practices and guidance on how to contribute to an RBB we address typical barriers for newcomers to contribute to open collaborative projects (Steinmacher et al., 2015). Beyond facilitating individual contributions, our suggested practices and working examples aim to promote a consistent application of the RBB template across contributors facilitating comparison across RBBs. Similar processes of iterative application and refinement have accompanied the establishment of documentation standards in the ABM community. For example, the ODD and ODD + D protocols have been repeatedly refined in response to experiences from their application and broader community use (Grimm et al., 2006, 2020, 2026; Müller et al., 2013; Szangolies et al., 2024).

Many of the practices identified in this study build on established principles for modular coding and documentation from software engineering, which have discussed standards for modular code for decades already (e.g., Gamma et al., 1995). The challenges encountered therefore point less to a lack of suitable technical principles than to their transfer into socio-environmental modelling, where interdisciplinary backgrounds and the need to align technical modularity with conceptual transparency pose specific requirements. Our guidance builds on this established knowledge and may provide RBB contributors with accessible entry points to relevant coding and documentation principles.

The contributor survey revealed a considerable workload involved in extracting and documenting RBBs from existing models, highlighting the need to both reduce contribution effort and strengthen incentives for participation. AI-assisted code extraction and documentation, screening for similar RBBs, or translation between programming languages may help reduce this workload (Filatova et al., 2025), while scientific exchange and contributing to standardization and theory development emerged as important motivations for contributors. Importantly, interaction within our community effort also helped harmonise RBB scope, resolve technical questions, and develop shared practices. Future uptake may therefore benefit from opportunities for community exchange and learning, for example through the existing CoMSES RBB forum (https://forum.comses.net/t/about-reusable-building-blocks/8996), community events, training formats, and ultimately the integration of reusable modelling practices into ABM teaching.

The community effort also provided a first larger-scale practical test of the AGENTBLOCKS repository. The repository provided guidance and a functional infrastructure for documenting, reviewing, and publishing RBBs, resulting in contributions that were generally well aligned with the underlying RBB definition. Beyond suggesting additional guiding material on applying the template, contributors also identified opportunities to further develop the repository to facilitate RBB contribution. These included clearer differentiation between conceptual variants and programming-language implementations, improved communication across user groups, and a more streamlined contribution and review workflow. These and other suggestions were compiled and shared with the AGENTBLOCKS development team. Further application by independent groups will be important for distinguishing individual preferences from recurring community needs and prioritising future developments.

As AGENTBLOCKS grows, its further development becomes not only a question of platform design, but also a question of governance and standardisation. Responsibilities for moderation, review, quality assurance, maintenance, and succession currently rely substantially on voluntary efforts and will require more sustainable structures, either through dedicated funding or distributed community governance, such as rotating editorial roles or reviewer pools (Garijo et al., 2022; Lin et al., 2020; Ram et al., 2019; Smith et al., 2018). At the same time, iterative development needs to be balanced against the benefits of a stable community standard. The AGENTBLOCKS template has evolved through several years of interdisciplinary discussion among agent-based modellers and already represents a refinement of the initial version presented by Berger et al. (2024), while optional fields and space for context-specific information provide flexibility for individual contributions. As comparability across RBBs depends on maintaining a consistent baseline structure, we recommend that future adaptations respond to recurring community needs while preserving continuity of the underlying template, which is an important prerequisite for its wider establishment as a standardised way to document RBBs throughout the community.

Several limitations qualify the conclusions that can be drawn from this study. First, participating modelling teams and contributed examples were not randomly selected but emerged from a coordinated community effort, potentially overrepresenting modellers already interested in reusable modelling practices. Moreover, overlaps between the group of contributors, the authors of the underlying RBB concept, and the developers of the AGENTBLOCKS platform may have influenced how the contribution process and encountered challenges were assessed. In particular, the thematic analysis of the editorial challenges was conducted by the coordinating editor, who was directly involved in the editorial process. While this involvement provided detailed contextual knowledge of the review and revision process, the dual role may also have influenced the categorization and interpretation of the feedback.

Second, the study focused on intended positive cases, self-selected by experienced modelling teams. Our findings can therefore inform how well the AGENTBLOCKS template and the additional guidance developed in this study support the development and documentation of intended RBBs, but provide limited evidence on how reliably it distinguishes RBBs from non-RBBs. Future research could address this limitation by systematically screening larger and more diverse collections of existing model code, such as the CoMSES.net archive, for recurring processes and implementation patterns, and testing the consistency with which independent reviewers classify candidate components as RBBs or non-RBBs.

Such broader screening could also include recurring ABM-specific technical processes, such as generating synthetic populations or initializing social networks from empirical data. Given our focus on decision-making and its potential to contribute to alternative and cumulative theorising, our collection and resulting decision tree did not include such technical processes. However, the community discussions informing the criteria defined in Filatova et al. (2025) considered recurring ABM-specific technical tasks to be ‘meaningful’. These were distinguished from generic technical programming tasks, such as import/export or scheduling utilities, for which established software libraries and platform-specific modules already provide reusable solutions. Whether the decision rules and prioritisation of RBB criteria derived in this study require adjustment when applied to such ABM-specific technical RBBs remains to be tested.

Further limitations result from the scope of the study. Our empirical basis comprises 14 RBBs focused specifically on decision-making processes in socio-environmental applications. RBBs representing ecological, spatial, biophysical, or technical processes may pose different challenges. Finally, our assessment primarily concerns the creation, documentation, and review of RBBs rather than their subsequent reuse by independent modelling teams. We can therefore assess whether model components can be demarcated according to the RBB concept, but not yet determine their practical utility and transferability across modelling contexts. Future research should test whether independent modellers can identify, understand, adapt, and integrate existing RBBs into new models, how much effort this requires, and how well RBBs can be transferred and scaled across applications. Nevertheless, by providing

a collection that includes alternative formalisations of decision-making processes operating at the same levels of interaction, this study provides potential candidates for future intercomparison studies examining how the choice of decision-making representation affects model outcomes.

Overall, this study moves the RBB concept from conceptual development towards practical application by developing, reviewing, and publishing RBBs across multiple modelling teams and applications. Our findings demonstrate that reusable components can be systematically derived from existing ABMs, while also showing that their demarcation, implementation, and documentation require careful interpretation of the RBB criteria and shared practices among contributors. The empirically grounded guidance, decision tree, and collection of RBBs developed through this effort provide a basis for more consistent future contributions. The next step is to move from demonstrating that RBBs can be created and shared towards testing their actual reuse: whether independent modellers can efficiently integrate them across modelling contexts and use alternative implementations for systematic model comparison. Such efforts will ultimately determine the contribution of RBBs to more cumulative and reusable socio-environmental modelling.

CRediT authorship contribution statement

Julia Kunkel: Writing – original draft, Visualization, Validation, Software, Project administration, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Volker Grimm:** Writing – review & editing, Validation, Software, Methodology, Investigation, Conceptualization. **Christian Stindt von Dohm:** Writing – review & editing, Validation, Software, Investigation. **Leon Grothmann:** Writing – review & editing, Validation, Software, Investigation. **Luana Schwarz:** Writing – review & editing, Validation, Software, Investigation. **Ronja Hotz:** Writing – review & editing, Validation, Software, Investigation. **Robert Huber:** Writing – review & editing, Validation, Software, Methodology, Investigation. **Yongchao Zeng:** Writing – review & editing, Validation, Software, Investigation. **Kaja Jurak:** Writing – review & editing, Validation, Software, Investigation. **Uta Berger:** Writing – review & editing, Validation, Software, Investigation. **J. Gareth Polhill:** Writing – review & editing, Validation, Software, Investigation. **Liz Verbeek:** Writing – review & editing, Software, Project administration. **Tatiana Filatova:** Writing – review & editing, Software, Resources, Project administration, Methodology, Funding acquisition, Conceptualization. **Matthias Meyer:** Writing – review & editing, Validation, Software, Investigation. **Birgit Müller:** Writing – review & editing, Validation, Supervision, Funding acquisition. **Jürgen Groeneveld:** Writing – review & editing, Validation, Supervision, Software, Methodology, Investigation, Conceptualization.

Software and data availability

In the course of this study, 14 reusable building blocks (RBBs) were implemented and published. All RBBs are publicly available through the open-access platform AGENTBLOCKS. Survey data were collected from contributors participating in this study. Aggregated survey results are presented in the study and participant characteristics are provided in the Supplementary Material. Individual-level survey data are not publicly available to protect participant anonymity, as the small and defined group of contributors may allow individuals to be identified from combinations of responses. Editorial feedback generated during the RBB review process was used for the thematic analysis reported in this study. The anonymised coded segments are provided in the Supplementary Material.

- Name of Software: Activity Dissimilarity Index. Impl.1[R]
- Developers: Robert Huber and Katja Gadhammer
- Contact: rhuber@ethz.ch
- Publish Date: August 18, 2025
- Programming language: R
- Source code and documentation at: https://www.agentblocks.org/rbb/activity-dissimilarity-index
- Name of Software: Attitude formation through relational learning process. Impl. 1 [Python]
- Developers: Luana Schwarz
- Contact: luana.schwarz@pik-potsdam.de
- Publish Date: November 13, 2025
- Programming language: Python
- Source code and documentation at: https://www.agentblocks.org/rbb/attitude-formation-through-relational-learning-processes
- Name of Software: Cumulative Prospect Theory. Impl. 1 [R]
- Developers: Julia Kunkel and Robert Finger
- Contact: julia.kunkel@ufz.de
- Publish Date: August 20, 2025
- Programming language: R
- Source code and documentation at: https://www.agentblocks.org/rbb/cumulative-prospect-theory
- Name of Software: Demand driven foraging decision in honey bees. Impl. 1 [NetLogo]
- Developers: Jürgen Groeneveld and Matthias Becher
- Contact: juergen.groeneveld@ufz.de
- Publish Date: November 4, 2025
- Programming language: NetLogo
- Source code and documentation at: https://www.agentblocks.org/rbb/demand-driven-foraging-decision-in-honey-bees
- Name of Software: Descriptive Norm and Norm Sensitivities. Impl. 1 [NetLogo]
- Developers: Christian Stindt, Leon Grothmann and Matthias Meyer
- Contact: christian.stindt@tuhh.de
- Publish Date: December 17, 2025
- Programming language: NetLogo
- Source code and documentation at: https://www.agentblocks.org/rbb/descriptive-norm-and-norm-sensitivities
- Name of Software: Descriptive Norm and Norm Sensitivities. Impl. 2 [NetLogo]
- Developers: Ronja Hotz
- Contact: ronja.hotz@kit.edu
- Publish Date: December 17, 2025
- Programming language: NetLogo
- Source code and documentation at: https://www.agentblocks.org/rbb/descriptive-norm-and-norm-sensitivities
- Name of Software: Dynamic Energy Budget (DEB) theory for animals. Impl. 1 [NetLogo]
- Developers: Volker Grimm, Benjamin T. Marin and Tjalling Jager
- Contact: volker.grimm@ufz.de
- Publish Date: August 18, 2025
- Programming language: NetLogo
- Source code and documentation at: https://www.agentblocks.org/rbb/dynamic-energy-budget-deb-theory-for-animals
- Name of Software: Endogenous Institutional Model (EIMOD). Impl. 1 [Java]
- Developers: Yongchao Zeng
- Contact: yongchao.zeng@kit.edu
- Publish Date: December 1, 2025
- Programming language: Java
- Source code and documentation at: https://www.agentblocks.org/rbb/endogenous-institutional-model-eimod
- Name of Software: Hegselman-Krause Opinion Dynamics of Social Influence. Impl. 1 [Java]
- Developers: Julia Kunkel
- Contact: julia.kunkel@ufz.de
- Publish Date: December 17, 2025
- Programming language: Java
- Source code and documentation at: https://www.agentblocks.org/rbb/hegselman-krause-opinion-dynamics-of-social-influence

- Name of Software: Heuristic Satisficing, Experimenting and Imitative Decision-Making Algorithms. Impl. 1 [NetLogo]
- Developers: Gary Polhill and Nick Gotts
- Contact: Gary.Polhill@hutton.ac.uk
- Publish Date: November 11, 2025
- Programming language: NetLogo
- Source code and documentation at: https://www.agentblocks.org/rbb/heuristic-satisficing-experimenting-and-imitative-decision-making-algorithms
- Name of Software: Profit Maximisation. Impl. 1 [R]
- Developers: Kaja Jurak
- Contact: kaja-alina.jurak@ufz.de
- Publish Date: August 6, 2025
- Programming language: R
- Source code and documentation at: https://www.agentblocks.org/rbb/profit-maximization
- Name of Software: Theory of Planned Behaviour for individual decision-making. Impl. 2 [Python]
- Developers: Luana Schwarz, Jannes Breier and Ronja Hotz
- Contact: luana.schwarz@pik-potsdam.de
- Publish Date: August 20, 2025
- Programming language: Python
- Source code and documentation at: https://www.agentblocks.org/rbb/theory-of-planned-behavior-for-individual-decision-making
- Name of Software: Zone-of-influence (ZOI). Impl. 1 [NetLogo]
- Developers: Cyril Piou, Marie-Christin Wimmler, Uta Berger
- Contact: uta.berger@tu-dresden.de
- Publish Date: November 4, 2025
- Programming language: NetLogo
- Source code and documentation at: https://www.agentblocks.org/rbb/zone-of-influence-zoi
- Name of Software: Zone-of-influence (ZOI). Impl. 2 [Python]
- Developers: Cyril Piou, Marie-Christin Wimmler, Uta Berger
- Contact: uta.berger@tu-dresden.de
- Publish Date: November 4, 2025
- Programming language: Python
- Source code and documentation at: https://www.agentblocks.org/rbb/zone-of-influence-zoi

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Given her role as an associate editor, TF had no involvement in the peer review of this article and had no access to information regarding its peer review. Full responsibility for the editorial process for this article was delegated to another journal editor. If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

We thank the members of the workshop “Speeding-up the development of useful agent-based land-use models” at Helmholtz Centre for Environmental Research - UFZ in Leipzig in 2024 for initiating this project. We gratefully acknowledge technical support by the current AGENTBLOCKS development team and for their realisation of first extensions to the platform. We thank the participants of the workshop “Reusable building blocks – turning individual models into a collective resource” at the Social Simulation Conference 2025 in Delft for valuable discussions on challenges and opportunities of the RBB concept in practice. JK gratefully acknowledges feedback from Robert Finger and the Agricultural Economics and Policy group for feedback on this work during her research stay at ETH Zürich. We thank the ERC StG SCALAR project for the core funding making the AGENTBLOCKS platform possible (European Research Council under the European Union's Horizon 2020 Research and Innovation Program, grant no 758014). JK and BM acknowledge funding by the D-A-CH project InsuranceGrass from the DFG (460738920). JGP gratefully acknowledges funding from the Scottish Government Rural and Environment Science and Analytical Services Division (project reference JHI-C5-1), which supported his involvement in this work. LS acknowledges financial support by the Generation Foundation, the Global Challenges Foundation, and Partners for a New Economy via the Earth4all project, funding from the European Union's Horizon 2.5 - Climate Energy and Mobility programme under grant agreement No 101081661 (project WorldTrans), and financial support from the project “CZS Research Groups for Earth System Modelling” funded by the Carl-Zeiss Stiftung. We thank three anonymous reviewer for their helpful and constructive comments that substantially improved this paper.

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.envsoft.2026.107211.

Data availability

We added the data in anonymised form to the Supplementary material and added a data availability statement.

References

Anshuka, A., van Ogtrop, F.F., Sanderson, D., Leao, S.Z., 2022. A systematic review of agent - based model for flood risk management and assessment using the ODD protocol. Nat. Hazards 112 (3). https://doi.org/10.1007/s11069-022-05286-y.

Balci, O., Arthur, J.D., Ormsby, W.F., 2011. Achieving reusability and composability with a simulation conceptual model. J. Simulat. 5 (3), 157–165.

Bearden, W.O., Netemeyer, R.G., Haws, K.L., 2011. Handbook of Marketing Scales: Multi-Item Measures for Marketing and Consumer Behavior Research, third ed. SAGE Publications, Thousand Oaks.

Belete, G.F., Voinov, A., Laniak, G.F., 2017. An overview of the model integration process: from pre-integration assessment to testing. Environ. Model. Software 87, 49–63. https://doi.org/10.1016/j.envsoft.2016.10.013.

Bell, A.R., Robinson, D.T., Malik, A., Dewal, S., 2015. Modular ABM development for improved dissemination and training. Environ. Modell. Software 73, 189–200. https://doi.org/10.1016/j.envsoft.2015.07.016.

Berger, U., Bell, A., Barton, C.M., Chappin, E., Dreßler, G., Filatova, T., Fronville, T., Lee, A., van Loon, E., Lorscheid, I., Meyer, M., Müller, B., Piou, C., Radchuk, V., Roxburgh, N., Schüler, L., Troost, C., Wijermans, N., Williams, T.G., et al., 2024. Towards reusable building blocks for agent-based modelling and theory development. Environ. Modell. Software 175, 106003. https://doi.org/10.1016/j.envsoft.2024.106003.

Braun, V., Clarke, V., 2006. Using thematic analysis in psychology. Qual. Res. Psychol. 3 (2), 77–101. https://doi.org/10.1191/1478088706qp063oa.

Castro, J., Drews, S., Exadaktylos, F., Foramitti, J., Klein, F., Konc, T., Savin, I., van de Bergh, J., 2020. A review of agent-based modeling of climate-energy policy. WIREs Clim. Change 11 (4), e647. https://doi.org/10.1002/wcc.647.

Cheng, S., Giabbanelli, P.J., Kuang, Z., 2023. Identifying the building blocks of social simulation models: a qualitative analysis using open-source codes in NetLogo. In: Blas, M.J., Alvarez, G. (Eds.), Proceedings of the 2023 Annual Modeling and Simulation Conference (ANNSIM'23). IEEE, pp. 306–317. https://ieeexplore.ieee.org/abstract/document/10155219.

Davis, F.D., 1985. A Technology Acceptance Model for Empirically Testing end-user Information Systems: Theory and Results. Dissertation. Massachusetts Institute of Technology.

Filatova, T., Verbeek, L., Warnier, M., Ghorbani, A., Nikolic, I., Grimm, V., Berger, U., Barton, M., Bell, A., Lee, A., Magliocca, N., Wagenblast, T., 2025. AGENTBLOCKS: a community platform for sharing , comparing , and improving reusable building blocks for (agent-based) models. JASSS 28 (4). https://doi.org/10.18564/jasss.5831.

Filatova, T., Verburg, P.H., Parker, D.C., Stannard, C.A., 2013. Spatial agent-based models for socio-ecological systems: challenges and prospects. Environ. Modell. Software 45, 1–7. https://doi.org/10.1016/j.envsoft.2013.03.017.

Gamma, E., Helm, R., Johnson, R., Vlissides, J., 1995. Design Patterns: Elements of Reusable object-oriented Software. Addison-Wesley, pp. 1–52.

Garijo, D., Ménager, H., Hwang, L., Trisovic, A., Hucka, M., Morrell, T., Allen, A., Task Force on Best Practices for Software Registries, SciCodes Consortium, 2022. Nine best practices for research software registries and repositories. PeerJ Comput. Sci. 8, e1023. https://doi.org/10.7717/peerj-cs.1023.

Grimm, V., 1999. Ten years of individual-based modelling in ecology: what have we learned and what could we learn in the future? Ecol. Model. 115, 129–148. https://doi.org/10.1016/S0304-3800(98)00188-4.

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., et al., 2006. A standard protocol for describing individual-based and agent-based models. Ecol. Model. 198 (1–2), 115–126. https://doi.org/10.1016/j.ecolmodel.2006.04.023.

Grimm, V., Berger, U., Meyer, M., Lorscheid, I., 2024. Theory for and from agent-based modelling: insights from a virtual special issue and a vision. Environ. Modell. Software 178, 106088. https://doi.org/10.1016/j.envsoft.2024.106088.

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. JASSS 23 (2). https://doi.org/10.18564/jasss.4259.

Grimm, V., Rosello, C., Hamilton, S., Gallagher, C., Micheletti, T., Szangolies, L., et al., 2026. Twenty years of the ODD protocol: development, usage, extensions, and lessons learned about modelling standards development. Preprint. ESS Open Archive. https://doi.org/10.22541/essoar.15005893/v1.

Groeneveld, J., Müller, B., Buchmann, C.M., Dressler, G., Guo, C., Hase, N., Hoffmann, F., John, F., Klassert, C., Lauf, T., Liebelt, V., Nolzen, H., Pannicke, N., Schulze, J., Weise, H., Schwarz, N., 2017. Theoretical foundations of human decision-making in agent-based land use models – a review. Environ. Modell. Software 87, 39–48. https://doi.org/10.1016/j.envsoft.2016.10.008.

Groeneveld, J., Odemer, R., Requier, F., 2024. Brood indicators are an early warning signal of honey bee colony loss — a simulation-based study. PLoS One 19 (5), e0302907. https://doi.org/10.1371/journal.pone.0302907.

Grothmann, L., Meyer, M., Stindt von Dohm, C., 2026. Formalizations of bicchieri's social norm theory in agent-based models: a qualitative systematic literature review. https://doi.org/10.2139/ssrn.7305078.

Hotz, R., Brown, C., Zeng, Y., Schmitt, T., Rounsevell, M., 2026. Modelling socio-psychological drivers of land management intensity. https://doi.org/10.48550/arXiv.2602.02347.

Huber, R., Xiong, H., Keller, K., Finger, R., 2022. Bridging behavioural factors and standard bio-economic modelling in an agent-based modelling framework. J. Agric. Econ. 73 (1), 35–63. https://doi.org/10.1111/1477-9552.12447.

Hulland, J., Sample, K.L., Houston, M.B., 2026. Standards for scale development in marketing: elevating the role of theory. J. Acad. Market. Sci. 54 (2), 359–380. https://doi.org/10.1007/s11747-026-01153-5.

Krueger, C.W., 1992. Software reuse. ACM Comput. Surv. 24 (2), 131–183.

Kunkel, J., Groeneveld, J., Finger, R., Müller, B., 2026. Can extensive grassland management pay off for farmers facing drought risks? A cumulative prospect theory-based approach. J. Agric. Econ. 77 (2), 652–676. https://doi.org/10.1111/1477-9552.70037.

Laniak, G.F., Olchin, G., Goodall, J., Voinov, A., Hill, M., Glynn, P., et al., 2013. Integrated environmental modeling: a vision and roadmap for the future. Environ. Modell. Software 39, 3–23. https://doi.org/10.1016/j.envsoft.2012.09.006.

Lin, D., Crabtree, J., Dillo, I., 2020. The TRUST principles for digital repositories. Sci. Data 7, 144. https://doi.org/10.1038/s41597-020-0486-7.

Lorscheid, I., Berger, U., Grimm, V., Meyer, M., 2019. From cases to general principles: a call for theory development through agent-based modeling. Ecol. Model. 393, 19–22. https://doi.org/10.1016/j.ecolmodel.2018.10.006.

Magliocca, N.R., Pathak, R., Price, A.N., Tanveer, H., Kumar, M., Moradkhani, H., 2024. Increasing behavioral richness and managing structural uncertainty in social-ecological system agent-based models. Socio-Environ. Syst. Modell. 6, 18749. https://doi.org/10.18174/sesmo.18749.

Martin, B.T., Jager, T., Nisbet, R.M., Preuss, T.G., Grimm, V., 2013. Predicting population dynamics from the properties of individuals: a cross-level test of dynamic energy budget theory. Am. Nat. 181 (4), 506–519. https://doi.org/10.1086/669904.

Matthews, R.B., Gilbert, N.G., Roach, A., Polhill, J.G., Gotts, N.M., 2007. Agent-based land-use models: a review of applications. Landsc. Ecol. 22, 1447–1459. https://doi.org/10.1007/s10980-007-9135-1.

Mccabe, T.J., 1976. A complexity measure. IEEE Trans. Software Eng. SE-2 (4), 308–320.

Miller, J., Mukerji, J., 2003. MDA Guide Version 1.0.1. OMG Group.

Müller, B., Bohn, F., Dreßler, G., Groeneveld, J., Klassert, C., Martin, R., Schlüter, M., Schulze, J., Weise, H., Schwarz, N., 2013. Describing human decisions in agent-based models - ODD+D, an extension of the ODD protocol. Environ. Modell. Software 48, 37–48. https://doi.org/10.1016/j.envsoft.2013.06.003.

Parker, D.C., Brown, D.G., Polhill, J.G., Deadman, P.J., Manson, S.M., 2008. Illustrating a New'Conceptual Design Pattern'For Agent-based Models of Land Use via Five Case Studies—The MR POTATOHEAD Framework.

Parnas, D.L., 1972. On the criteria to be used in decomposing systems into modules. Commun. ACM 15 (12), 1053–1058.

Peng, W., Iyer, G., Bosetti, V., Chaturvedi, V., Edmonds, J., Fawcett, A.A., Hallegatte, S., Victor, D.G., Vuuren, D. Van, Weyant, J., 2021. Climate policy models need to get real about people — here's how. Nature 594 (7826), 174–176.

Ram, K., Boettiger, C., Chamberlain, S., Ross, N., Salmon, M., Butland, S., 2019. A community of practice around peer review for long-term research software sustainability. Comput. Sci. Eng. 21 (2), 59–65. https://doi.org/10.1109/MCSE.2018.2882753.

Robinson, S., Nance, R.E., Paul, R.J., Pidd, M., Taylor, S.J.E., 2004. Simulation model reuse: definitions, benefits and obstacles. Simulat. Model. Pract. Theor. 12 (7–8 SPEC. ISS.), 479–494. https://doi.org/10.1016/j.simpat.2003.11.006.

Schlüter, M., Baeza, A., Dressler, G., Frank, K., Groeneveld, J., Jager, W., Janssen, M.A., McAllister, R.R.J., Müller, B., Orach, K., Schwarz, N., Wijermans, N., 2017. A framework for mapping and comparing behavioural theories in models of social-ecological systems. Ecol. Econ. 131, 21–35. https://doi.org/10.1016/j.ecolecon.2016.08.008.

Schlüter, M., Müller, B., Frank, K., 2019. The potential of models and modeling for social-ecological systems research: the reference frame ModSES. Ecol. Soc. 24 (1), 31. https://doi.org/10.5751/ES-10716-240131.

Schroeder, S.A., Vendome, C., Giabbanelli, P.J., Montfort, A.M., 2022. Towards reusable building blocks to develop COVID-19 simulation models. Proceedings of the 2022 Winter Simulation Conference, pp. 569–580. https://doi.org/10.1109/WSC57314.2022.10015340.

Schulze, J., Müller, B., Groeneveld, J., Grimm, V., 2017. Agent-based modelling of social-ecological systems: achievements, challenges, and a way forward. JASSS 20 (2). https://doi.org/10.18564/jasss.3423.

Smith, A.M., Niemeyer, K.E., Katz, D.S., Barba, L.A., Githinji, G., Gymrek, M., Huff, K.D., Madan, C.R., Mayes, A.C., Moerman, K.M., Prins, P., Ram, K., Rokem, A., Teal, T.K., Guimera, R.V., Vanderplas, J.T., 2018. Journal of open source software (JOSS): design and first-year review. PeerJ Comput. Sci. 4, e147. https://doi.org/10.7717/peerj-cs.147.

Steinmacher, I., Graciotto Silva, M.A., Gerosa, M.A., Redmiles, D.F., 2015. A systematic literature review on the barriers faced by newcomers to open source software projects. Inf. Software Technol. 59, 67–85. https://doi.org/10.1016/j.infsof.2014.11.001.

Szangolies, L., Rohwäder, M.-S., Ahmed, H., Jahanmiri, F., Wagner, A., Souto-Veiga, R., Grimm, V., Gallagher, C.A., 2024. Visual ODD: a standardised visualisation illustrating the narrative of agent-based models. JASSS 27 (4), 1. https://doi.org/10.18564/jasss.5450.

Taberna, A., Filatova, T., Roy, D., Noll, B., 2020. Tracing resilience, social dynamics and behavioral change: a review of agent-based flood risk models. Socio-Environ. Syst. Modell. 2, 17938. https://doi.org/10.18174/sesmo.2020a17938.

Taillandier, P., Gaudou, B., Grignard, A., Huynh, Q.-N., Marilleau, N., Caillou, P., Philippon, D., Drogoul, A., 2019. Building, composing and experimenting complex spatial models with the GAMA platform. GeoInformatica 23 (2), 299–322. https://doi.org/10.1007/s10707-018-00339-6.

Teeluckdharry, N.B., Teeroovengadum, V., Seebaluck, A., 2021. Scale development in marketing research. In: Nunkoo, R., Teeroovengadum, V., Ringle, C.M. (Eds.), Handbook of Research Methods for Marketing Management. Edward Elgar Publishing, pp. 17–50. https://doi.org/10.4337/9781788976954.00007.

Thober, J., Schwarz, N., Hermans, K., 2018. Agent-based modeling of environment-migration linkages: a review. Ecol. Soc. 23 (2), 41. https://doi.org/10.5751/ES-10200-230241.

Tucker, W.V., Gross, D.C., 2013. What more do we want in modeling and simulation interoperability and reuse?. In: Proceedings of the 2013 Grand Challenges on Modeling and Simulation Conference, pp. 1–8.

Vendome, C., Rao, D.M., Giabbanelli, P.J., 2020. How do modelers code artificial societies? Investigating practices and quality of netlogo codes from large repositories. In: 2020 Spring Simulation Conference (Springsim). IEEE, pp. 1–12. https://doi.org/10.22360/SpringSim.2020.HSAA.007.

VERBI Software, 2023. *MAXQDA 2023* (No. 24). maxqda.com.

Voinov, A., Shugart, H.H., 2013. 'Integronsters', integral and integrated modeling. Environ. Modell. Software 39, 149–158. https://doi.org/10.1016/j.envsoft.2012.05.014.

Wijermans, N., Scholz, G., Chappin, É., Heppenstall, A., Filatova, T., Polhill, J.G., Semeniuk, C., Stöppler, F., 2023. Agent decision-making: the elephant in the room - enabling the justification of decision model fit in social-ecological models. Environ. Modell. Software 170, 105850. https://doi.org/10.1016/j.envsoft.2023.105850.

Zeng, Y., Raymond, J., Brown, C., Byari, M., Rounsevell, M., 2025. Simulating endogenous institutional behaviour and policy implementation pathways within the land system. Ecol. Model. 501, 111032. https://doi.org/10.1016/j.ecolmodel.2025.111032.