

SupplyNet: Supporting Visual Exploratory Learning in Supply Chain via Contextual Multi-Agent Simulation

Yanjia Li^{a,1}, Kelcy Kexin Han^{a,1}, Tianrui Hu^b, Yi-Fan Cao^a, Huamin Qu^a, Sicheng Song^{a,*}

^aThe Hong Kong University of Science and Technology, Hong Kong, China

^bCity University of Hong Kong, Hong Kong, China

Abstract

Simulation has long supported supply chain management instruction by letting learners observe network behavior and test decision strategies. Recent progress in LLM-driven agents opens new possibilities for richer, more adaptive simulations, but many existing systems still present abstract, opaque data that overwhelms learners and discourages active exploration. We introduce *SupplyNet*, a gamified visual simulation system built on a contextual graph-based LLM multi-agent framework that models interdependent supply chain dynamics and provides responsive feedback through tiered challenges. *SupplyNet* turns the simulation into a manipulable decision space by integrating an interactive network view of system state, a branching timeline for “what-if” exploration and comparison, and a task-oriented analysis console for structured performance breakdowns. Together, these visual components support counterfactual exploration, causal tracing, and comparative reasoning about outcomes. A user study suggests that *SupplyNet* increases engagement and supports users’ perceived understanding of supply chain dynamics, highlighting the potential of pairing contextual multi-agent simulation with visualization to advance operational comprehension.

Keywords: Supply Chain Visualization, Multi-Agent Simulation, Intelligent Learning Environment

1. Introduction

Supply Chain Management (SCM) is a fundamental course in university curricula that equips students with uncertainty-aware decision making skills under the interdependency of supply chain dynamics Stadler (2014). Yet it is difficult to learn from lectures alone: key concepts (e.g., disruption propagation, inventory oscillation, and trade-offs between service level and cost) emerge from the interaction of entities and time-varying constraints. Therefore, agent-based simulations have become a common pedagogical approach for SCM Swaminathan et al. (1998); Xue et al. (2005), where business entities are represented as autonomous agents Swaminathan et al. (1998), allowing learners to experiment with decisions and observe dynamic network behaviors. Xue et al. (2005).

However, existing simulation systems are primarily designed for narrow concept demonstration, often overlooking realism and learner engagement. These systems typically represent complex networks Song et al. (2023a) as static diagrams Song et al. (2023b, 2024) and confine critical data to tables, creating a disconnect between abstract numbers and intuitive understanding Pfeffer and Fong (2002). Such limitations are further compounded by the simplistic, rule-based agents, which fail to reflect the inner motivation of human beings in the real-world

business context. Consequently, there is a lack of free space for students’ self-exploratory learning of supply chain knowledge, ultimately restricting the cultivation of their practical application and decision-making capabilities Pettigrew and Starkey (2016).

To better understand these learning barriers, we conducted a formative study with 7 university students and 3 instructors, which informed the design of a contextual, feedback-rich, and comparison-driven exploratory simulation environment.

Based on these insights, we present *SupplyNet*, a gamified visual simulation system designed for university students studying SCM, supporting them in exploring supply chain dynamics for engaging concept learning. Powered by a Large Language Model (LLM)-based multi-agent framework, the system models the interdependent supply chain dynamics under customizable contexts and difficulty tiers. The system integrates gamified onboarding where users select roles and challenge levels, then define supply chain structures. During simulation, users can observe state changes, inject disruptions, make decisions, and receive responsive feedback.

SupplyNet provides visual interfaces that render the environment as a manipulable decision space. The **Graph Structure View** presents business entities and logistics as an intuitive network. For counterfactual reasoning, the **Timeline Tree Map** tracks simulation progress and profitability and allows users to branch at any point to create parallel “what-if” scenarios for comparison. The **Analysis Console** provides structured breakdowns of key performance metrics to reduce analytic burden. By combining contextual multi-agent simulation with decision-

*Sicheng Song is the corresponding author.

Email addresses: ylif@connect.ust.hk (Yanjia Li),
khanac@connect.ust.hk (Kelcy Kexin Han),
tianruhu-c@my.cityu.edu.hk (Tianrui Hu), caoyifan@ust.hk (Yi-Fan Cao),
huamin@ust.hk (Huamin Qu), csescsong@ust.hk (Sicheng Song)

¹These authors contributed equally.

oriented visualizations, *SupplyNet* supports learners in tracing causal propagation, testing strategies, and building operational understanding through sustained exploration.

We conducted a comprehensive user study to evaluate *SupplyNet*. Participants reported improved learning support and a more engaging learning experience compared to a classic baseline simulation. We further observed various exploration styles and a reciprocal relationship between learning support and engagement, suggesting that contextual simulation and visual exploratory tools can reinforce one another in SCM learning.

Our contributions are threefold:

- We characterize key barriers that limit exploratory learning in existing supply chain simulation tools, informed by a formative study with students and instructors.
- We propose *SupplyNet*, a gamified visual simulation system for exploratory learning in supply chains, supported by contextual multi-agent simulation. *SupplyNet* enables counterfactual exploration and causal reasoning through an interactive network view, a branching “what-if” timeline, and a task-oriented analysis console.
- We conduct a rigorous evaluation that demonstrates *SupplyNet*’s effectiveness in enhancing learning support and engagement, outperforming the baseline.

2. Related Work

2.1. Supply Chain Simulation

Supply chain simulation has been indispensable for complex cost management, revealing downstream-upstream cascading effects and supporting decisions in inventory management, resource allocation and risk management Datta and Christopher (2011); Persson and Olhager (2002).

Agent-Based Modeling (ABM) is widely adopted for supply chain simulation due to its strength in modeling entities’ attributes and behaviors. Mohaddesi et al. (2022) models human-simulation interaction in supply chain shortage through statistical models to demonstrate how information sharing affects human decisions. Integrated with machine learning, ABM has greatly enhanced agents’ adaptability to dynamic environments. For example, Kotecha and Chanona (2024) and Liu et al. (2022) apply reinforcement learning in multi-agent systems to optimize inventory management amid ever-evolving market conditions. However, learning-based methods suffer from the challenge of acquiring large training datasets in practice Cannas et al. (2024). Furthermore, existing frameworks often neglect agent-user interactivity and agents’ decision interpretability, which are vital for users to better understand, trust, and influence agents’ decisions, thus promoting effective collaboration in dynamic supply chain contexts.

2.2. LLM-based Multi-Agent Framework for Simulation

LLMs have emerged as a powerful technology for creating autonomous agents in multi-agent frameworks, featuring strong task performance, zero-shot adaptability to domain-specific context, interpretable reasoning processes, and natural language interaction with humans. In human-AI collaboration, LLM-based agents have been proven effective in boosting user productivity and understanding by facilitating tasks like writ-

ing Zhang et al. (2023, 2026) or mitigating filter bubbles with diverse perspectives Zhang et al. (2024).

Building upon these foundational capabilities, researchers have increasingly applied LLM agents to simulate complex social and economic systems. Studies have shown that LLM agents can exhibit human-like behaviors in macroeconomic decision-making Li et al. (2024) and visualization understanding Chen et al. (2025); Song et al. (2026) consistent with established principles, develop emergent competition dynamics aligned with sociological theories Zhao et al. (2024); Lin et al. (2024); Han et al. (2024), and demonstrate realistic cooperative behaviors in social and educational simulations Park et al. (2022); Gao et al. (2023); Chen et al. (2026). This body of work establishes that LLM-based agents are not merely task-solvers but can also be promising proxies for human actors in complex dynamic environments.

A key advantage of LLM agents for such simulations is their ability to adopt diverse personas and behavioral patterns via simple prompt engineering Choi and Li (2024); Jiang et al. (2024). Their natural language outputs allow them to articulate their “motivation” and “feelings” in response to environmental stimuli, enhancing their human-likeness. For instance, LLM agents have displayed positive or concerned attitudes in simulated financial markets based on social media information, mirroring human emotional reactions Yang et al. (2025). This interactivity improves simulation engagement and agent reliability, laying the groundwork for engaging simulated worlds.

The potential of LLMs has also been extended to SCM. Quan and Liu (2024) introduce an LLM-based multi-agent framework for a four-echelon chain-structured model, demonstrating the feasibility of SCM application with emergent dynamics and high flexibility. Our work builds upon this basic skeleton and further implements a graph-structured supply chain framework. We leverage the unique human-likeness, interpretability, and interactivity of LLM agents, and exploit their potential for gamified visual design, to develop an interactive tool that enhances learning engagement.

2.3. Gamification in Educational Design

Gamification applies game design elements in non-game contexts to increase engagement and motivate goal-oriented behavior. Prior work has demonstrated its effectiveness in improving learning efficacy Feger et al. (2019). By creating interactive and immersive learning experiences, gamification enriches system design and supports sustained user participation.

Existing studies have incorporated a variety of game elements into interactive systems to facilitate learning. For instance, contextualization frames learning process as a narrative-driven experiences to intensify immersion and engagement. In Du et al. (2024), users assume roles to explore different life trajectories within a simulated world. Reward mechanisms, including *points*, *badges*, and *unlockables*, are frequently used to reinforce progress and encourage consistent task completion; in Kirchner-Krath et al. (2024), such mechanisms help users overcome procrastination. Social elements, such as *competition* and *leaderboards*, further promote active exploration through

connections as demonstrated in Schade et al. (2023).

Gamification also has strong potential for learning SCM, where it facilitates active participation in simulations under uncertainty. Yet engagement and interaction design have received limited attention in this domain. SCM involves numerous quantitative factors, such as inventory levels, costs, prices, and delivery times—alongside abstract representations of relationships and performance Stadler (2014). Current SCM simulation systems often represent these variables as raw numerical values in static tables and charts. A typical example is the Beer Game simulation Harvard Business School (2023), which presents order, shipment, and cost information in lengthy, time-ordered tables. This challenges users to track business performance and grasp the underlying supply chain dynamics, thereby narrowing usability and analytical insights. Although some systems, such as Skill Dynamics (2023), offer more visually expressive interfaces, limited interactivity still keeps learners in a largely observational role, making it difficult to test hypotheses, revise decisions, and connect actions to downstream consequences through iterative exploration.

SupplyNet addresses the gap by integrating game elements into the design, providing an engaging and responding learning environment for users to better perceive dynamics and refine decision-making strategies through repeated interaction.

3. Formative Study

We conducted a formative study to identify limitations in existing supply chain simulation systems and derive design considerations for improved engagement, usability, and interaction. Our study investigated: (1) challenges faced by students when learning SCM; (2) how current simulation systems support this process; and (3) desired features of an ideal simulation system.

3.1. Participants

We recruited 7 students (4 female, 3 male; S1-S7), including 3 undergraduates and 4 graduate students majoring in Accounting (S1, S3), Finance (S2), Marketing (S4, S5), and Operations Management (S6, S7). All participants had completed at least one supply-chain-related course and had relevant foundational knowledge. Students were compensated at \$20/hour. Additionally, 3 instructors (2 female, 1 male; E1-E3) with extensive cross-departmental experience in teaching supply-chain-related courses were invited from publicly accessible faculty directories at top-tier universities. Instructor details are shown in Tab. 1. They were compensated at \$50/hour.

Participant ID	Academic Title	Department	Experience (Years)	Location
E1	Associate Professor	Operations Management	10+	Hong Kong
E2	Assistant Professor	Marketing	3	United States
E3	Associate Professor	Industrial Engineering	7	Hong Kong

Table 1: Demographic and professional background of instructor participants.

3.2. Procedure

The study followed standard ethical guidelines. Participants were informed of the study purpose, procedures, and their right to withdraw at any time. Informed consent was obtained prior to participation. All data were anonymized and used solely for research purposes. Each participant completed a one-hour

semi-structured interview, conducted either in person or via Zoom. Sessions were audio-recorded with permission, transcribed, and stored with restricted access. The interview followed a predefined outline with three phases. First, we collected background information on participants’ prior exposure to supply chain concepts and simulation systems. Students reflected on course experiences and past use of simulations, including perceived value and limitations, while instructors shared their professional background and observations of common challenges. Second, participants interacted with the “Root Beer Game”, a single-chain simulation that illustrates the bullwhip effect and remains widely used Harvard Business School (2023). Each participant was given 15 minutes to explore the system in the role of a retailer, after which they assessed usability, interaction experience, and overall value. Finally, participants discussed expectations for an ideal supply chain simulation system, focusing on desired functionality, feedback mechanisms, and visual representations.

3.3. Findings

Despite acknowledging the utility of simulations in illustrating core SCM fundamentals, participants prominently reported five key challenges that impede students’ learning and exploration, which we summarized as follows:

C1: Limited Contextual Cues Lead to Disengagement. Students consistently reported that the rigid, linear structure and random fluctuation of current simulations quickly create a disconnect from real-world complexity, leading to disengagement and boredom once the basic concepts are mastered (S3, S4, S6). S3 stated, “*I would totally lose the interest to keep playing after mastering the basic concepts and demand patterns.*” This disconnect is lessened when the environment is contextualized. S4 echoed this, “*I would have more fun when playing if the environmental changes are contextualized. For example, the demand fluctuation is subject to seasonal change or quality concerns.*” Instructors (E1-E3) confirmed that the lack of an engaging narrative or evolving complexity undermines the goal of long-term skill reinforcement (E1: “*As a simulation game, it is important to tell a fun and interesting story to keep students engaged.*”). Advanced learners also expressed a desire for systems that could support new and challenging content to maintain engagement over time (S1, S2, S5, S6). As S2 stated, “*It would be much more interesting if some recent events in life can be added to the simulated environment.*”

C2: Insufficiency of Timely and Actionable Feedback. A major source of student frustration was the absence of timely and specific feedback on their in-game decisions, hindering the learning process. Students felt “lost” and were often unaware of their performance mid-game, which led to random decision-making as the simulation progressed (S3: “*I feel lost in the middle of the game for not knowing how good I perform.*” S4: “*I received little hints on whether I have been making good or bad decisions in the game.*”). While final summary statistics on earnings were available, most participants (S2-S6) found these numbers insufficient for evaluating their overall business management performance. Instructors acknowledged this critical gap, noting the impossibility of providing detailed, personal-

ized feedback to every student due to high labor and time costs (E2: “*However, it is almost impossible for us to provide detailed feedback to everyone due to high labor and time costs.*”). Consequently, students felt little incentive to replay the simulation with an objective of self-improvement.

C3: Constraint of Alternative Decision Exploration. All student participants (S1-S7) expressed a strong, inherent curiosity to explore how alternative planning decisions would lead to different consequences, but they felt constrained by the restricted operative space of existing systems. A primary limitation was the difficulty of comparing outcomes between different playthroughs. S5 commented that “*Although I could always restart the simulation, it is challenging for me to compare the outcomes between different decision-making without a dedicated feature to place different timelines side-by-side for direct analysis.*” This finding aligns directly with instructors’ teaching objectives. E2 confirmed that explicit support for varying one or two key factors (e.g., price or order quantity) and observing the differential outcomes is a valuable component of the counterfactual analysis prized in business studies (E2: “*It contributes to the counterfactual analysis we valued in school.*”).

C4: Opaque Agent Reasoning and Interdependencies. Student learning was hampered by the opacity of the supply chain network’s internal dynamics. Simply observing evolving numerical factors was deemed insufficient for underlying understanding (S1, S3). Participants desired insight into the mutual effects within the simulation and the rationale driving computer-controlled agents’ behavior (S3: “*I am not only curious about how decisions affect the environment, but also how business entities are motivated to react to the environmental changes.*”). S6 specifically wished for agents to “*think aloud,*” elaborating that visibility into an agent’s mind makes the simulation more realistic and helps grasp a manager’s thinking. Instructor E2 highlighted that making agent reasoning transparent serves a dual educational purpose: it helps students “*understand the inter-dependent nature of supply chain networks*” and the explanations themselves “*serve as a helpful reference for students to manage their own business.*”

C5: Inadequate Visualizations for Analysis. The visual design and presentation of information were a consistent concern (S1-S7, E1-E2). Instructors noted that supply chain environments are information-dense, posing considerable challenges for students’ information processing (E2). S6 shared that previous simulations suffered from “*messy data organization, which brought me much trouble to find environmental changes and check business performance when playing.*” Beyond basic data organization, most students (S3-S7) and instructors (E1-E2) expressed a strong need for task-oriented data visualization, including time-series and correlation analyses. S3 explicitly requested, “*I wish to observe the evolution of key metrics over time, not only to understand the short- and long-term effects of decisions but also to compare the correlation between factors.*” E1 noted that intuitive visualization is essential to facilitate the financial and operational analyses that instructors require students to conduct for evaluating management efficiency.

These findings highlight the evolving learning needs associ-

ated with supply chain simulations and underscore the significance of more advanced visual-dynamic graph agent simulations designed to support exploratory learning.

3.4. Design Goals

Drawing upon the challenges identified in the formative study (C1–C5), we formulated three high-level design goals to bridge the gap between abstract supply chain concepts and practical experimental learning, guiding our subsequent system implementation. These goals articulate the necessary interaction strategies and visual capabilities, directly informing the subsequent system design.

DG1: Provide Immersive Context and Immediate Feedback. Our formative study revealed that static, numerical simulations detach students from the urgency of real-world supply chains (C1), while delayed feedback hinders their ability to evaluate decisions (C2). An effective learning environment would ground decision making in realistic operational contexts and evolving challenges rather than isolated numerical updates. Such context could help learners remain engaged beyond initial concept acquisition and would encourage iterative experimentation. The system should also provide timely and interpretable feedback so that learners could quickly connect actions and external disruptions to downstream consequences, adjust strategies, and maintain a sense of progress.

DG2: Support Counterfactual Exploration and Comparative Analysis. Participants reported frustration with the linear progression of traditional simulations, which prevents them from revisiting past decisions or directly comparing outcomes (C3). Therefore, we aim to lower the cost of experimentation and facilitate “what-if” analysis through iterative trial-and-error learning. The design should not only enable learners to test hypotheses over time (e.g., by changing ordering policies or responding to disruptions), but also create a structured way to compare decision paths and make alternative outcomes easy to notice and reflect on. This would support counterfactual analysis and the development of transferable decision heuristics.

DG3: Intuitively Visualize Network Structure and Dynamics. Given the multi-echelon dependencies (C5) and opaque interactions between entities (C4) in information-dense supply chains, traditional tabular interfaces often cause cognitive overload. To assist students in forming accurate mental models, the interface would present network structure and evolving system state in a clean and intuitive manner. It would make flows of materials and information between entities visually traceable, helping learners grasp the overall state at a glance and follow how changes propagate through the network. Furthermore, computer-controlled business managers would be assigned human-like personas that can vividly articulate their reasoning and motivations, making interactions more authentic and insightful. Learners should also benefit from data analytic support for trend inspection and structured comparison across alternative operations, helping them reason about correlations among factors and relate decisions to long-term impacts.

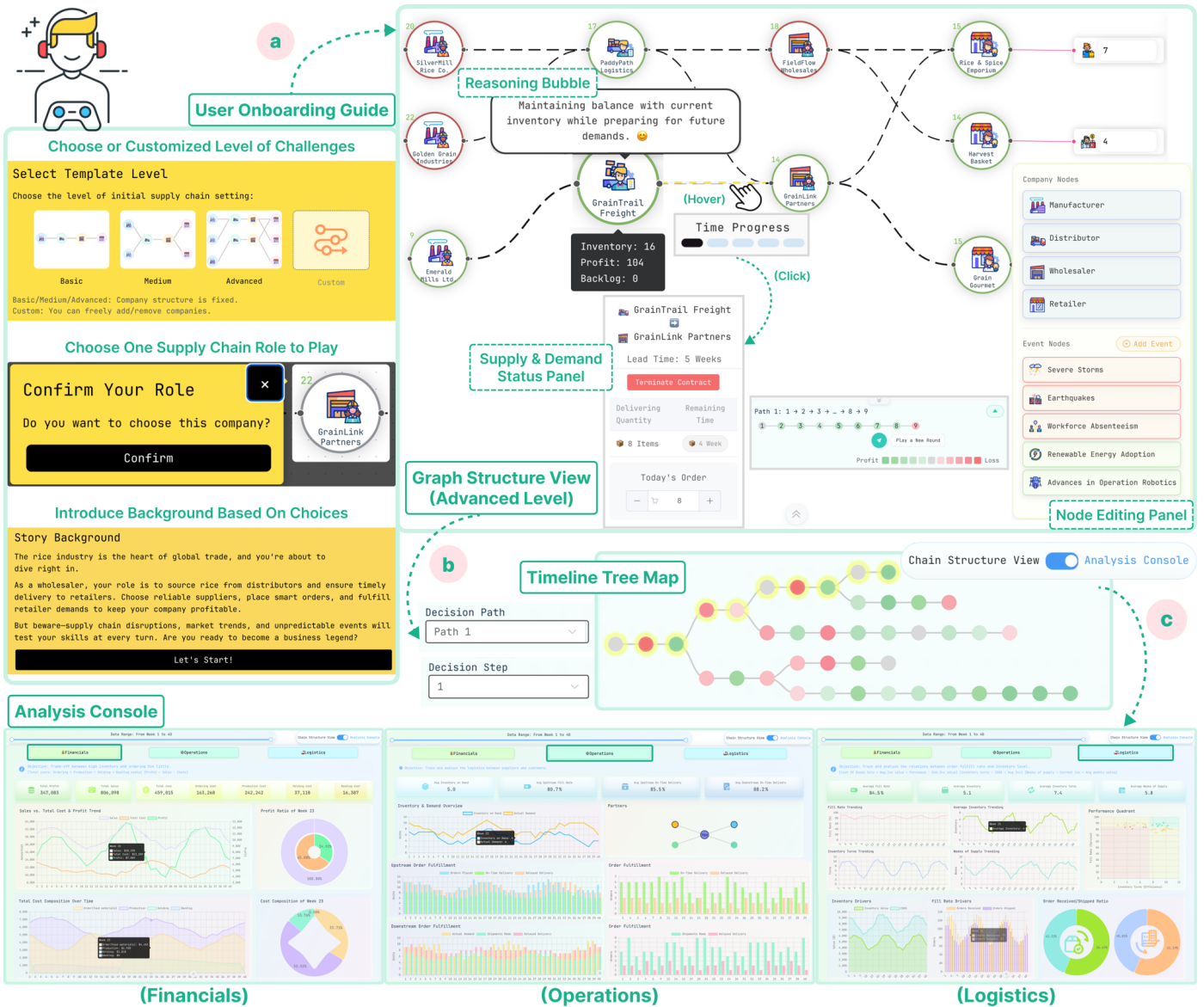


Figure 1: Overview of the *SupplyNet* interface. (1) Users begin with the onboarding guide, selecting a challenge level and company role before entering the *Graph Structure View*. (2) The *Timeline Tree Map* records decision paths and supports branching for “what-if” exploration. (3) Users switch to the *Analysis Console*, which provides three dashboards: (a) financials, (b) operations, and (c) logistics, for inspecting performance over time and comparing outcomes across timelines.

4. System Overview

4.1. System Pipeline

SupplyNet’s learning objective centers on inventory management, guiding students to understand dynamic supply chain interdependencies by making sequential decisions on order placement and supplier selection and observing the resulting trade-offs in multiple factors.

The pipeline follows an explicit bidirectional loop between an LLM-based multi-agent simulation core and an interactive visual interface. The simulation core (*i.e.*, the multi-agent framework) maintains the evolving supply chain state as a directed graph: nodes encode entity attributes (inventory, price, and production capacity), and edges encode supply relationships and logistics (lead times, downstream demand, and arriving deliveries). In each period, agents make sequential decisions on order placement and supplier selection; they are

queried from downstream to upstream (retailers → wholesalers → distributors → manufacturers) to reflect demand propagation. To keep agent prompting scalable and context-aware, the simulation core performs agent-specific subgraph retrieval by extracting a localized context around each entity, including direct partners, relevant attributes, and active events. The retrieved subgraph is then textualized and combined with heuristic “Golden Rules” as agent input, enabling agents to produce role-specific actions along with brief motivation explanations. The agents return structured outputs containing decision variables (*e.g.*, order quantities, supplier IDs). Together with the updated system states, it is saved as temporal snapshot to the database. The front-end reads data from the database to present real-time visualization on the user interface. Conversely, user decisions made on the interface are sent back to the database as an update of the system states. It is fetched by the simulation core when querying agents for decision-making. The

overall data flow is illustrated in Fig. 2. At a more abstract level, this pipeline can be viewed as a reusable simulation architecture for dynamic relational learning environments. The graph represents domain entities, dependencies, and time-varying states; agents and learners observe relevant subgraphs, take domain-specific actions, and trigger state transitions; and the database stores temporal snapshots that support replay, branching, and comparison. In SupplyNet, this architecture is instantiated with business entities, logistics links, inventory-related states, ordering decisions, disruption events, and operational KPIs. Other subjects can reuse the same architecture by replacing the domain schema, transition rules, agent heuristics, event types, and learning metrics. We return to this abstraction in Sec. 8.3 to discuss cross-subject instantiations.

4.2. Workflow and Coordinated Views

SupplyNet’s interface is organized as a workflow across four views (Fig. 1). Users begin with user onboarding, where they select a company role and challenge level and receive a narrative context, grounding subsequent decisions in a realistic setting (**DG1**). They then manage the simulation in the *Graph Structure View*, which serves as the primary sandbox for monitoring network status, inspecting entities and links, and executing decisions (Fig. 1(1)); the *Node Editing Panel* supports constructing custom networks and injecting contextual events, while the *Supply & Demand Status Panel* provides link-level details for inspecting orders, deliveries, and fulfillment. After each step, updated states and lightweight outcome cues are surfaced to help learners connect actions and disruptions to downstream consequences (**DG1**), while maintaining interpretable network reasoning about propagation (**DG3**).

As the simulation progresses, actions and outcomes are recorded in the *Timeline Tree Map*, which preserves decision paths and enables users to revisit earlier states and branch parallel trajectories for “what-if” exploration and comparison (Fig. 1(2)), enabling counterfactual reasoning (**DG2**). Users can switch to the *Analysis Console* to examine structured dashboards of key performance indicators over time and compare outcomes across timelines (Fig. 1(3)), supporting diagnosis and reflection on longer-term impacts (**DG2, DG3**). Learners iterate by returning to the *Graph Structure View* to revise decisions or introduce disruptions based on insights from the *Timeline Tree Map* and the *Analysis Console*.

5. Visualization and Interaction Design

In this section, we first introduce the common network visual substrate that makes supply chain structure and dynamics legible, and then present the contextualization and analysis tools that build upon it (Fig. 1).

5.1. Graph Structure Workspace

To support learners in reasoning about complex dependencies and interdependent dynamics (**DG3**), SupplyNet centers interaction around the *Graph Structure View*. Inspired by the “Visual Information Seeking Mantra” Shneiderman (2003), the view prioritizes network overview and supports progressive disclosure of details through interaction. This shifts learners from isolated tables to a topological representation that makes structure,

propagation, and local constraints simultaneously inspectable.

The *Graph Structure View* serves as the primary workspace for operating the simulated supply chain. It visualizes the environment as a directed graph, where nodes represent business entities across stages and edges represent logistical relationships. To reduce cognitive overload in information-dense environments, the view adopts lightweight visual status encodings Song et al. (2025) that provide immediate summaries while preserving access to detail. Nodes are color-coded to reflect real-time operational outcomes: green indicates profitability, red represents losses due to backlog or other operational issues, and gray signals a shutdown state (Fig. 4(a)). These status encodings also function as lightweight visual reward cues Feger et al. (2019), giving learners an immediate sense of progress or concern without requiring them to open detailed panels. Edges visually distinguish active versus planned relationships and animate to indicate transportation progress: active orders are shown with solid lines, and planned partnerships with dashed lines. When hovering over an edge, a compact progress indicator appears to support rapid scanning without opening additional panels (Fig. 4(b)).

The view supports progressive disclosure through details-on-demand, allowing users to scan the simulation at multiple levels of granularity. Selecting a node reveals its local operational state, such as inventory, capacity, backlog-related costs, and other decision-relevant attributes. Selecting an edge opens the *Supply & Demand Status Panel*, which focuses on the relationship between two connected entities by exposing lead time and fulfillment progress. Together, these interactions allow learners to quickly locate bottlenecks and follow how local constraints propagate through upstream and downstream dependencies.

To support scenario construction and controlled interventions, the *Node Editing Panel* enables users to modify network structure and attach contextual disruptions directly onto entities. Users may add entities to construct alternative hierarchies, or inject events by dragging event icons onto nodes. Injected events are visually highlighted on the graph, helping users connect a local shock to subsequent network-wide changes as the simulation evolves.

Finally, to mitigate the perceived opacity of computer-controlled managers, SupplyNet exposes agent decision rationale through an on-demand *Reasoning Bubble*. When users click an automated entity, the bubble appears above the entity and presents a brief textual explanation of the agent’s motivation for its current action and a compact affective cue that reflects current business performance. Specifically, the motivation is shown as a concise sentence, while the affective state is mapped to an emoji to preserve interface cleanliness. This design keeps the primary workspace lightweight while still providing interpretable hooks for learners to connect agent behaviors to evolving system conditions.

5.2. Onboarding, Challenges, and Contextual Feedback

To transform abstract operations into a situated learning experience (**DG1**), SupplyNet integrates narrative framing, tiered challenges, and real-time feedback loops into the interaction

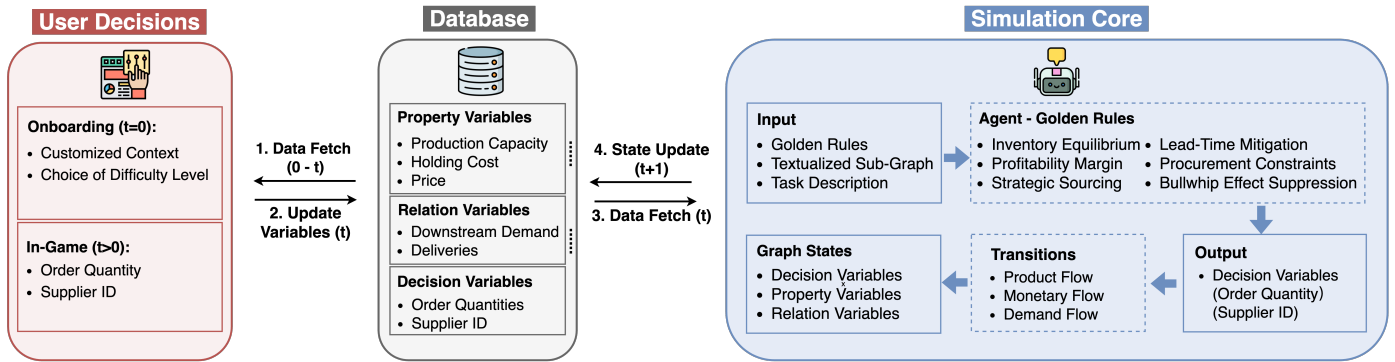


Figure 2: The bi-directional data flow in *SupplyNet*.

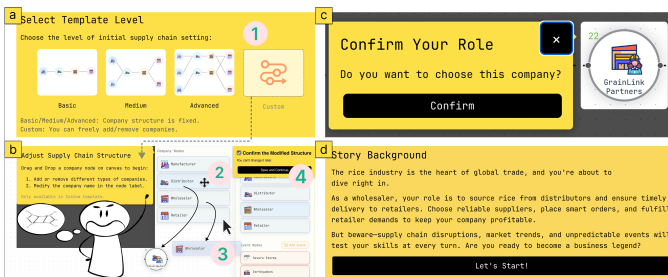


Figure 3: Onboarding Process. (a) Challenge selection. (b) Custom network assembly. (c) Role selection with narrative context.

flow. This design aims to sustain engagement beyond initial concept acquisition and help learners rapidly connect decisions and disruptions to downstream consequences.

Immersive Onboarding and Tiered Challenges. Before the simulation begins, the *User Onboarding Guide* (Fig. 3) establishes context and calibrates complexity to learners’ needs. During onboarding, users first choose a challenge level that matches their experience. The predefined levels progressively increase structural complexity, while a sandbox option supports free-form network assembly for open-ended exploration. A basic level introduces a linear four-stage chain to focus attention on core inventory concepts without overwhelming learners. Intermediate and advanced levels introduce a hub-and-spoke network and additional customization, encouraging strategic sourcing and resilience-oriented responses. Users then select a specific company role, and the system provides a short role-specific narrative that frames the simulation as a first-person managerial task with an explicit mission, making decision making feel purposeful and situated rather than purely numerical. This narrative contextualization is a core gamification strategy Domínguez et al. (2013) that transforms abstract operational tasks into situated, goal-directed activities.

Events and Real-time Feedback. Contextual volatility is introduced through dynamic events. At the lower section of the *Node Editing Panel* there is an event palette offering a set of predefined scenarios, such as natural disasters, workforce shortages, and technological upgrades, with positive or negative effects on factors such as inventory and logistics. Dragging event icons onto specific nodes simulates their effects, with immediate visual feedback highlighting positive impacts in green and negative impacts in red (Fig. 4(c)). Furthermore, users can

create their own disruptions by clicking the “Add Event” button. This allows them to define a custom event’s name, its specific impact, magnitude, and duration. This functionality helps users understand how disruptions propagate through the network while evaluating possible mitigation strategies.

SupplyNet provides real-time feedback tightly coupled with each decision cycle. When a user confirms a decision for a given week, the backend advances the simulation and updates deliveries, inventory, and sales-related outcomes; key outcome indicators such as profit and backlog signals are refreshed immediately in the interface. When a user switches to the *Analysis Console* to examine the history, the system retrieves historical data from the database and performs real-time calculations to generate statistical summaries. This immediate feedback loop mirrors the tight action–reward cycles characteristic of gamified learning environments, supporting iterative learning and creating an incentive mechanism by enabling learners to quickly evaluate decision quality and maintain a sense of progress in evolving contexts (DG1).

5.3. Timeline Tree Map for Counterfactual Exploration

A key contribution of *SupplyNet* is enabling non-linear exploration of decision consequences. Drawing on counterfactual thinking Roesse (1997) and external cognition Scaife and Rogers (1996), the interface externalizes simulation history so learners can compare “what happened” with plausible alternatives without relying on memory.

The *Timeline Tree Map* (Fig. 5) records simulation progression as a branching structure rather than a linear log. Each step node represents a decision point, visually encoded with an outcome summary to support fast scanning of successful versus problematic periods. Users click any past step to revert the simulation state to that point, modify a decision, and resume the simulation to spawn a new parallel branch. When a past decision is modified, downstream steps on the affected branch are reset to preserve consistency, ensuring that subsequent outcomes reflect the revised history. This branching mechanism functions as an explicit “save and branch” interaction, echoing the save and reload mechanic familiar from digital games, lowering the cost of experimentation while maintaining provenance of alternative strategies. By preserving parallel trajectories, the timeline makes comparison concrete and repeatable, assisting structured “what-if” exploration rather than ad hoc replay.

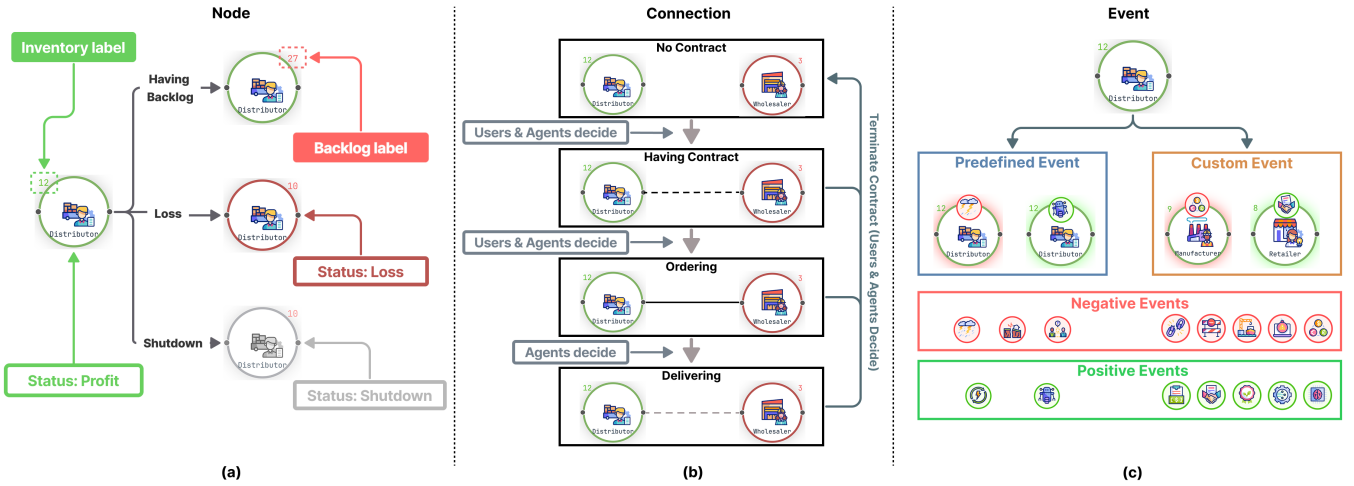


Figure 4: Visual Encodings. (a) Node status summary. (b) Link relationships and logistics progress. (c) Event attachment and impact visualization.

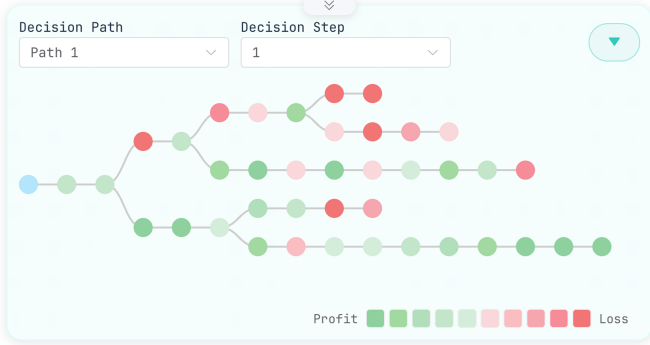


Figure 5: The *Timeline Tree Map* visualizes the simulation history, enabling users to revert to past states and branch new timelines for comparison.

The *Timeline Tree Map* is designed for both backtracking and comparison. Users can switch between branches to examine how alternative decisions shape subsequent outcomes. This design helps direct contrast across strategies and encourages users to articulate why outcomes diverge, strengthening transferable heuristics for inventory and fulfillment management (DG2).

5.4. Task-Oriented Analysis Console for Metric-driven Diagnosis

While the graph workspace facilitates situated monitoring and interventions, learners also need metric-driven evidence to diagnose delayed effects and assess strategies. The *Analysis Console* (Fig. 1) provides an organized analytic space that complements visual causal tracing in the network view.

The console organizes performance indicators into three dimensions that align with common learning objectives in SCM: financial performance (Fig. 1(a)), operational efficiency (Fig. 1(b)), and logistics efficiency (Fig. 1(c)). Each panel introduces the meaning and formula of key concepts at the top, and then presents interactive charts that visualize metric evolution over time. This design combines conceptual explanation with live evidence, helping learners interpret metrics in the context of ongoing dynamics rather than as detached summaries.

All concepts presented in the Analysis Console follow standard SCM definitions and formulas in textbooks to ensure consistency with the long-established principles. We provide the complete set of mathematical formulas and variable definitions in Appendix D.

The console is synchronized with the active branch selected in the *Timeline Tree Map*. Switching branches updates the console views accordingly, enabling learners to compare alternative trajectories using consistent metric definitions and temporal alignment. This branch-aware linkage supports diagnostic reasoning about why a strategy succeeds or fails, and helps learners relate local interventions in the graph to longer-term, system-level impacts observed in KPI trends (DG2, DG3).

5.5. Implementation Details

The prototype of SupplyNet is implemented using a modern web technology stack, with the frontend built in Vue 3.5 and the backend simulation core in Python environment. We developed a custom multi-agent framework where all agents are powered by OpenAI’s gpt-4o model. Detailed information regarding the frameworks, libraries, database design, system requirements, and token cost considerations is provided in Appendix F. The system demo video can be found in the supplementary materials.

6. User Study

We conducted a within-subjects user study to evaluate *SupplyNet* against a standard baseline simulation, focusing on usability, engagement, and learning support.

6.1. Participants

We recruited 14 participants ($N = 14$, 4 female, 10 male, aged 21–28) via university platforms. The cohort comprised 11 graduate and 3 undergraduate students majoring in business or industrial engineering. To ensure domain literacy for evaluating the learning content, all participants had foundational knowledge of SCM: 54.8% had completed one related course, 38.5% had completed two, and 7.7% had completed three. Participants received \$20/h for their contribution.

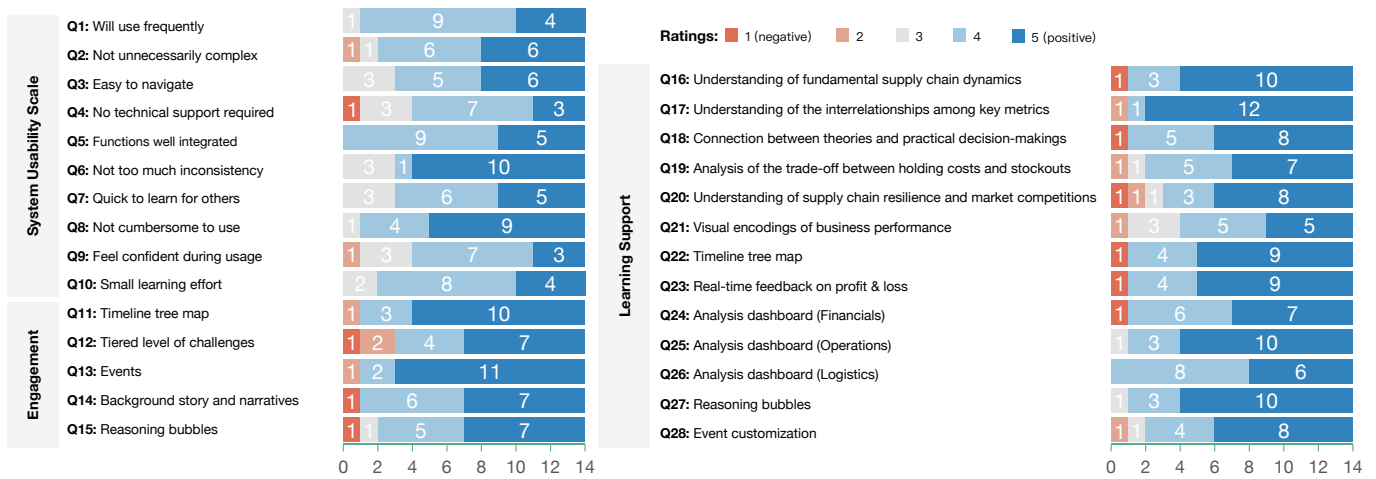


Figure 6: Results of questionnaires for the system's evaluation metrics, assessing three key dimensions: the System Usability Scale (SUS) (Q1-Q10), Engagement (Q11-Q15), and Learning Support (Q16-Q28). The Learning Support items include participants' perceived learning gains and decision-to-outcome understanding. Ratings range from 1 (negative) to 5 (positive), with the bar chart showing the number of respondents for each score across all questions. For consistency, the points for negative questions in the SUS scale have been reversed.

6.2. Procedure

The study employed a within-subjects design with counterbalanced order to mitigate learning and fatigue effects. Participants completed two conditions: a baseline system (HBS Beer Game Harvard Business School (2023)) and *SupplyNet*.

At the beginning of each session, we introduced the study purpose and data privacy practices. In the baseline condition, participants followed the built-in instructions (approximately 15 minutes) and then played the simulation for 30 minutes. In the *SupplyNet* condition, participants first received a 15-minute interactive tutorial covering the *Graph Structure View*, *Timeline Tree Map*, and *Analysis Console*. They then completed a 30-minute task session with the goal of maximizing profit while verbalizing their thoughts using a think-aloud protocol. A mandatory 5-minute break was enforced between conditions.

After both sessions, participants completed a comparative questionnaire using five-point Likert items, followed by a semi-structured interview about their experience with *SupplyNet*. Each session lasted about 90 minutes and was audio-recorded.

6.3. Data Collection and Analysis

We collected quantitative ratings from questionnaires and interaction logs from both conditions. For qualitative data, two authors conducted an inductive thematic analysis on the interview transcripts following Braun and Clarke's six-step framework Braun and Clarke (2006). Codes and themes were iteratively refined through discussion; disagreements were discussed until consensus was reached to ensure that the reported themes were grounded in participant feedback.

6.4. Quantitative Results

The results provide evidence for *SupplyNet*'s effectiveness in fulfilling design goals and outperforming the baseline.

Evaluation Against Design Goals. User ratings (Fig. 6) suggest that key design mechanisms were effective:

- **Context and Feedback (DG1):** Participants rated the tiered challenge design (Q12), dynamic events (Q13), and background narratives (Q14) highly (avg. > 4.5/5), supporting the effectiveness of contextualized gameplay. Based on interaction logs, 64% of participants voluntarily selected the Intermediate or Advanced difficulty levels, and the average active session time was 28 minutes within the 30-minute task, suggesting sustained engagement during decision cycles.
- **Exploration and Comparison (DG2):** Participants reported that the *Timeline Tree Map* supported exploratory analysis (Q22; 13/14 rated ≥ 4) and helped maintain engagement (Q11-Q15). Real-time profit/loss feedback (Q23) was viewed as important for guiding iterative trials. Ratings also indicate that the comparison-oriented analysis support (Q21) helped learners contrast outcomes across alternative operations.
- **Visual Understanding and Interpretability (DG3):** Participants rated the *Analysis Console* panels (Q24-Q26) positively for clarifying system dynamics and performance breakdowns. The Reasoning Bubbles (Q27) were also rated positively by most participants, suggesting that brief agent explanations improved interpretability. Usability ratings for navigation (Q3) remained high despite the multi-view interface.

Comparative Performance. As shown in 7, *SupplyNet* showed statistically significant improvements ($p < 0.01$) over the baseline:

- **Learning Support:** Participants rated *SupplyNet* significantly higher in helping them learn core SCM concepts (Q29), understand network structure (Q30), and clarify entity roles in the supply chain (Q31). They also reported stronger support for reasoning about consequences, including tracing cause-and-effect relationships (Q32), con-

in the *Analysis Console*. This visual curiosity motivated him to examine the underlying figures and understand the insights behind them. Through a few rounds of play, he became familiar with complex cost compositions and professional metrics. Similarly, P6 was first attracted to the *Timeline Tree Map*, playfully creating multiple timelines to observe their consequences. The resulting color variations on the timelines motivated him to think more critically. In these cases, features designed for exploration and immediate feedback served as a bridge, naturally guiding users toward analytical learning.

For other participants (P2-3, P12), the act of learning itself becomes the primary driver of engagement. This was particularly evident among those with science or engineering backgrounds, who were immediately drawn to the *Analysis Console* and the wealth of information it provided. Engagement was achieved through mastery and understanding. P12 was initially unsure of his grasp of “inventory turns”. He then used the provided formula to manually calculate the value from the data shown on the interface. After confirming his self-calculated number matched the one on the chart, he reported a profound sense of joy. As he articulated, “*Understanding in detail gives me a sense of achievement, and it further leads to joy.*”

8. Discussion

In this section, we discuss the key findings from our case study and their implications for designing effective educational simulations. First, we explore the divergence in user preferences for learning support, attributing this to the interplay between prior domain knowledge and cognitive load. Then we will examine the reciprocal relationship our system successfully creates between engagement and learning, and discuss how this dual-pathway approach can guide the future development of adaptive learning systems. We will conclude by discussing the limitations of our system and outlining future research directions.

8.1. Different preferences for learning support

The observed divergence in preferences for intuitive versus analytical learning support in Sec. 7.1 can be attributed to two interconnected factors: the participants’ background knowledge and their cognitive load.

The level of prior domain knowledge determines the learning behaviors. Participants who had taken more courses possessed a more systematic understanding of supply chain. Thus, they are capable of interpreting inter-related variables and engaging with the *Analysis Console* directly without feeling disoriented. However, participants at a beginner level lacked the knowledge foundation, making them more likely to feel overwhelmed by the density of graphs and metrics presented in the console.

This difference in background knowledge directly correlates with the second factor: cognitive load. For advanced users, their expertise automated the interpretation of basic concepts, freeing up cognitive resources to focus on higher-level strategic thinking and data analysis. This made the analytical process feel empowering rather than burdensome. In contrast, novices had to process every piece of information from terminology to graphical representations, which rapidly consumed their cognitive capacity and led to feelings of being overwhelmed.

This distinction highlights the need for progressive disclosure of information and analytical learning support. To avoid overwhelming beginner-level users, the system should initially offer a streamlined, visual interface that minimizes information density and fosters motivation. As users gain expertise, they can be introduced to the richer analytical tools and data granularity that advanced learners require to achieve mastery and make the simulation a more robust training environment. This adaptive approach helps create a scaffold that guides all users from initial engagement toward deeper analytical understanding.

8.2. Reciprocal Relationships between Engagement and Learning Support

Engagement and learning support are often treated as distinct, sometimes competing, dimensions in educational design. Some systems, such as Capitalism Lab prioritize engagement through rich visual stimuli and playful design, but at the expense of substantive learning. Others, like the baseline system in this study, focus heavily on learning by presenting dense data and complex analytical tasks, which can sacrifice a fun and engaging user experience.

In contrast, the findings in Sec. 7.2 demonstrates that our system builds a successful reciprocal relationship between these two elements, where each one strengthens the other. We observed two primary pathways into this positive feedback loop. Some participants were initially captivated by the visual appeal and interactive elements, which served as an effective gateway to exploring its deeper educational content. For them, engagement led to learning. Others found engagement through the act of learning itself, deriving a sense of mastery and intellectual satisfaction from analyzing data and mastering the simulation’s mechanics. For them, learning was inherently engaging.

This dual pathway highlights that an effective educational simulation should provide entry points for both types of users. It must cater to those drawn in by playful interaction and visual appeal, as well as those motivated by the intellectual satisfaction of analysis and mastery. By doing so, all students, regardless of their starting point, can enter a rewarding cycle where engagement fuels learning, and successful learning, in turn, deepens engagement.

8.3. Generalizability Across Subjects

Although SupplyNet is designed and evaluated in the context of supply chain management, its broader contribution is not limited to the direct reuse of SCM-specific entities, actions, or metrics. Rather, the system embodies a reusable interaction pattern for subjects that can be modeled as dynamic relational systems. Such subjects typically involve multiple connected entities, local decisions or disturbances, delayed propagation of effects, and trade-offs that learners need to diagnose over time. Under this abstraction, SupplyNet separates a reusable layer from a domain-specific layer. The reusable layer consists of graph-based state representation, contextualized sub-graph retrieval and role/task framing, event injection, temporal snapshots, branchable histories, coordinated visual views, and metric-driven reflection. The domain-specific layer consists of node and edge semantics, valid learner actions, agent heuris-

tics, event types, transition rules, and performance indicators. Therefore, the framework is most transferable to subjects that share three learning properties: knowledge emerges from interactions among multiple entities, learners benefit from testing interventions over time, and learning objectives can be expressed through interpretable state changes and performance indicators.

This separation clarifies how each view and interaction technique can be adapted beyond SCM. The onboarding guide functions as a role-and-complexity calibration mechanism: company roles, challenge levels, and narrative missions can be replaced by domain roles such as public-health officers, urban planners, ecosystem managers, or grid operators, while difficulty can be controlled through network size, uncertainty level, number of controllable policies, or severity of external events. The Graph Structure View functions as the main inspection-and-intervention workspace. Its nodes and edges can be redefined from companies and logistics links to regions and mobility flows, intersections and roads, species and ecological interactions, or generators and transmission lines. Node and edge inspection remains a details-on-demand interaction, but the inspected attributes change from inventory, capacity, backlog, and lead time to domain-specific states such as infection prevalence, congestion level, population size, habitat quality, energy load, or system reliability.

The Node Editing Panel and event palette can be preserved as tools for controlled perturbation. Instead of adding suppliers or injecting storms and shortages, learners may add transportation links, allocate medical resources, restore habitats, adjust grid capacity, or introduce policy shocks and environmental disturbances. The Reasoning Bubble can also be generalized as an interpretability layer. In human-centered domains, it can explain the decision logic of simulated actors such as hospitals, commuters, local authorities, or households. In non-human domains, such as ecology or energy systems, it can instead summarize the local rule or model mechanism that explains why a state changed. The Timeline Tree Map provides a branchable history for counterfactual reasoning, allowing learners to return to a prior state, alter an intervention, and compare alternative futures. The Analysis Console remains a metric-aligned reflection space, where SCM dashboards for financial, operational, and logistics performance are replaced with indicators aligned with the target subject and learning objectives.

Building on this component-level mapping, Table 2 illustrates how the abstract framework can be instantiated in representative subjects by replacing the graph schema, learner interventions, contextual events, and metric dashboards while preserving the same exploratory workflow.

The instantiations in Table 2 suggest a practical adaptation workflow. First, designers identify whether the target subject can be represented as a dynamic relational system and define its graph schema, including entities, relationships, state variables, and controllable actions. Second, they encode domain rules or agent heuristics that constrain how the system evolves and how autonomous actors respond. Third, they define contextual events that perturb local states and make propagation mechanisms observable. Fourth, they replace SCM-specific

dashboards with metrics aligned with the target learning objectives. Finally, they preserve the same exploratory workflow: role-based onboarding, graph-based inspection and intervention, event-driven feedback, branchable counterfactual comparison, and dashboard-based reflection.

This abstraction also clarifies the boundary of generalization. SupplyNet is less suitable for learning tasks dominated by static memorization or one-shot procedural training unless those tasks can be embedded in a meaningful relational process. Cross-subject deployment would still require collaboration with domain instructors to validate the graph schema, transition rules, agent behavior, event design, and assessment metrics. The generalizable contribution of SupplyNet is thus not a universal plug-and-play simulator, but a reusable design framework for transforming domain-specific dynamic systems into interactive, visual, and counterfactual learning environments.

8.4. Limitations and Future Work

The proposed framework has certain limitations that lead to future research directions.

Further eliciting the potential of LLMs in educational simulations. Beyond their role as business managers, the capabilities of LLMs can be harnessed in other aspects of the simulation to enhance engagement and learning support. One significant potential lies in simulating complex, human-like interactions such as negotiations, leveraging capacity for understanding and reasoning. Furthermore, LLMs can be integrated with the *Analysis Console* to act as an on-demand analytical assistant. This contextual just-in-time aid would bridge the gap between observing data and understanding core SCM concepts, making the learning process more intuitive and effective. By leveraging these capabilities, future educational simulations can become more interactive, responsive, and interesting. Beyond supply chain education, the framework’s interaction design—gamified onboarding, branching timeline exploration, and task-oriented analysis dashboards—could potentially be adapted to other subjects involving dynamic systems with interdependent decision-making, such as public health policy or urban planning, provided that domain-specific graph schemas, agent heuristics, and performance metrics are defined.

Benchmarking user performance in educational simulations. The simulation is designed to motivate students to explore alternative strategies and explore “what-if” scenarios via *Timeline Tree Map*. However, the system currently offers limited support in helping students benchmark their decisions. For instance, students may maximize profits through “trial and errors”, but lack insights into optimal outcomes they could achieve. To enhance learning support, the system could integrate optimization algorithms to calculate the upper and lower bounds of potential business performance, motivating students with a clearer understanding of achievable results.

Reducing computational cost. The computational complexity of the framework increases significantly with the number of agents. Specifically, the complexity scales as $T(M \cdot N)$, where M represents the number of stages and N represents the total number of agents in the framework. The computational

Table 2: Representative instantiations of the generalized SupplyNet framework beyond SCM.

Subject	Graph schema	Learner interventions and events	Metrics and learning tasks
Public health and epidemiology	Nodes represent regions, hospitals, or population groups; edges represent mobility, contact, or patient-transfer links.	Learners allocate vaccines, testing resources, hospital beds, or mobility restrictions. Events may include new variants, public gatherings, seasonal changes, or vaccine shortages.	Dashboards can summarize infection prevalence, hospital utilization, intervention cost, mortality, and regional equity. Learning tasks include understanding disease propagation, delayed intervention effects, resource-allocation trade-offs, and policy timing.
Urban planning and transportation	Nodes represent districts, intersections, stations, or public facilities; edges represent roads, transit lines, commuting flows, or accessibility links.	Learners adjust signal timing, transit frequency, zoning policies, road capacity, or emergency routing. Events may include accidents, weather disruptions, population growth, or infrastructure failures.	Dashboards can summarize travel time, congestion, accessibility, emissions, budget, and service reliability. Learning tasks include identifying network bottlenecks, reasoning about induced demand, and comparing efficiency-equity trade-offs.
Ecology and environmental management	Nodes represent species, habitats, water bodies, or resource pools; edges represent predation, competition, migration, or nutrient flows.	Learners introduce conservation policies, harvest limits, habitat restoration, or pollution controls. Events may include droughts, invasive species, wildfires, or climate shocks.	Dashboards can summarize population stability, biodiversity, resource availability, extinction risk, recovery time, and intervention cost. Learning tasks include exploring trophic cascades, resilience, delayed ecological feedback, and conservation-resource trade-offs.
Energy systems and smart grids	Nodes represent generators, substations, storage units, consumers, or regions; edges represent transmission lines and energy flows.	Learners dispatch generators, schedule storage, adjust demand-response policies, or repair failures. Events may include heat waves, renewable fluctuations, transmission failures, or price shocks.	Dashboards can summarize reliability, load shedding, generation cost, carbon emissions, renewable utilization, and peak demand. Learning tasks include understanding supply-demand balancing, cascading failures, renewable intermittency, and cost-emission trade-offs.

cost for simulation at the current scale is acceptable. However, this could become a concern in the custom mode, where users have the freedom to construct networks at a much larger scale. It may lead to prohibitive computational costs and long waiting times, making the simulation unaffordable. To mitigate this, techniques such as quantization, model compression, or parallel computing may be required to reduce the computational costs.

9. Conclusion

This work focused on the challenge of creating educational simulations for SCM that are often disengaging and fail to provide sufficient learning support. Based on insights from our formative study, we developed *SupplyNet*, a gamified visual simulation system built on a contextual graph-based LLM multi-agent framework that models complex, interdependent supply chain dynamics. The *SupplyNet* interface visualizes and transforms the simulated supply chain into an interactive learning environment, where students can experiment with alternative strategies, and conduct analysis on business performance metrics. Our evaluation demonstrated that the proposed system significantly enhanced the learning support for students while promoting a fun and engaging learning experience. Furthermore, it successfully creates a reciprocal relationship between engagement and learning, effectively supporting students at different learning stages.

Acknowledgement

This work is supported by RGC GRF Grant 16210321.

References

- Braun, V., Clarke, V., 2006. Using thematic analysis in psychology. *Qualitative research in psychology* 3, 77–101. doi:10.1191/1478088706qp0630a.
- Cannas, V.G., Ciano, M.P., Saltamacchia, M., Secchi, R., 2024. Artificial intelligence in supply chain and operations management: a multiple case study research. *International Journal of Production Research* 62, 3333–3360. doi:10.1080/00207543.2023.2232050.
- Chen, Z., Song, S., Shum, K., Lin, Y., Sheng, R., Wang, W., Qu, H., 2025. Unmasking deceptive visuals: Benchmarking multimodal large language models on misleading chart question answering, in: *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 13767–13800.
- Chen, Z., Zeng, Y., Song, S., Lin, Y., Xu, X., Qu, H., Xia, M., 2026. Vizqstudio: Iterative visualization literacy mcqs design with simulated students. *IEEE Transactions on Visualization and Computer Graphics*, 1–18doi:10.1109/TVCG.2026.3695959.

- Choi, H.K., Li, Y., 2024. PICLe: eliciting diverse behaviors from large language models with persona in-context learning, in: Proceedings of the 41st International Conference on Machine Learning.
- Datta, P.P., Christopher, M.G., 2011. Information sharing and coordination mechanisms for managing uncertainty in supply chains: a simulation study. *International Journal of Production Research* 49, 765–803. doi:10.1080/00207540903460216.
- Domínguez, A., Saenz-de Navarrete, J., De-Marcos, L., Fernández-Sanz, L., Pagés, C., Martínez-Herráiz, J.J., 2013. Gamifying learning experiences: Practical implications and outcomes. *Computers & education* 63, 380–392. doi:10.1016/j.compedu.2012.12.020.
- Du, W., Zhu, Z., Xu, X., Che, H., Chen, S., 2024. Career-Sim: Gamification design leveraging LLMs for career development reflection, in: Extended abstracts of the CHI conference on Human Factors in Computing Systems. doi:10.1145/3613905.3650928.
- Feger, S.S., Dallmeier-Tiessen, S., Woźniak, P.W., Schmidt, A., 2019. Gamification in science: A study of requirements in the context of reproducible research, in: Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems, p. 1–14. doi:10.1145/3290605.3300690.
- Gao, C., Lan, X., Lu, Z., Mao, J., Piao, J., Wang, H., Jin, D., Li, Y., 2023. S3: Social-network simulation system with large language model-empowered agents. *arXiv preprint arXiv:2307.14984* doi:10.2139/ssrn.4607026.
- Han, X., Wu, Z., Xiao, C., 2024. "guinea pig trials" utilizing GPT: A novel smart agent-based modeling approach for studying firm competition and collusion. *arXiv:2308.10974*.
- Harvard Business School, 2023. Global supply chain management simulation v3. URL: <https://hbsp.harvard.edu/product/7908-HTM-ENG?ntt=supply+chain+simulation>.
- Jiang, H., Zhang, X., Cao, X., Breazeal, C., Roy, D., Kabbara, J., 2024. PersonaLLM: Investigating the ability of large language models to express personality traits, in: Findings of the Association for Computational Linguistics: NAACL 2024. doi:10.18653/v1/2024.findings-naacl.229.
- Kirchner-Krath, J., Schmidt-Kraepelin, M., Schöbel, S., Ullrich, M., Sunyaev, A., Von Korfflesch, H.F., 2024. Outplay your weaker self: A mixed-methods study on gamification to overcome procrastination in academia, in: Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems. doi:10.1145/3613904.3642048.
- Kotecha, N., Chanona, A.d.R., 2024. Leveraging graph neural networks and multi-agent reinforcement learning for inventory control in supply chains. *arXiv preprint arXiv:2410.18631* doi:10.1016/j.compchemeng.2025.109111.
- Li, N., Gao, C., Li, M., Li, Y., Liao, Q., 2024. EconAgent: Large language model-empowered agents for simulating macroeconomic activities, in: Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). doi:10.18653/v1/2024.acl-long.829.
- Lin, R.Y., Ojha, S., Cai, K., Chen, M., 2024. Strategic collusion of LLM agents: Market division in multi-commodity competitions, in: Language Gamification-NeurIPS 2024 Workshop.
- Liu, X., Hu, M., Peng, Y., Yang, Y., 2022. Multi-agent deep reinforcement learning for multi-echelon inventory management. *Production and Operations Management*, 10591478241305863doi:10.2139/ssrn.4262186.
- Mohaddesi, O., Griffin, J., Ergun, O., Kaeli, D., Marsella, S., Hartevelde, C., 2022. To trust or to stockpile: Modeling human-simulation interaction in supply chain shortages, in: Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems. doi:10.1145/3491102.3502089.
- Park, J.S., Popowski, L., Cai, C., Morris, M.R., Liang, P., Bernstein, M.S., 2022. Social simulacra: Creating populated prototypes for social computing systems, in: Proceedings of the 35th Annual ACM Symposium on User Interface Software and Technology. doi:10.1145/3526113.3545616.
- Persson, F., Olhager, J., 2002. Performance simulation of supply chain designs. *International Journal of Production Economics* 77, 231–245. doi:10.1016/s0925-5273(00)00088-8.
- Pettigrew, A., Starkey, K., 2016. From the guest editors: The legitimacy and impact of business schools—key issues and a research agenda. *Academy of Management Learning & Education* 15, 649–664. doi:10.5465/amle.2016.0296.
- Pfeffer, J., Fong, C.T., 2002. The end of business schools? less success than meets the eye. *Academy of management learning & education* 1, 78–95. doi:10.5465/amle.2002.7373679.
- Quan, Y., Liu, Z., 2024. InvAgent: A large language model based Multi-Agent system for inventory management in supply chains. *CoRR*.
- Roese, N.J., 1997. Counterfactual thinking. *Psychological bulletin* 121, 133. doi:10.1037/0033-2909.121.1.133.
- Scaife, M., Rogers, Y., 1996. External cognition: how do graphical representations work? *International journal of human-computer studies* 45, 185–213. doi:10.1006/ijhc.1996.0048.
- Schade, E., Savino, G.L., Niess, J., Schöning, J., 2023. MapUncover: Fostering spatial exploration through gamification in

- mobile map apps, in: Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems. doi:10.1145/3544548.3581428.
- Shneiderman, B., 2003. The eyes have it: A task by data type taxonomy for information visualizations, in: The craft of information visualization. Elsevier, pp. 364–371. doi:10.1016/b978-155860915-0/50046-9.
- Skill Dynamics, 2023. Supply chain beer game - virtual simulation. URL: <https://skilledynamics.com/supply-chain-beer-game/>.
- Song, S., Chen, J., Li, C., Wang, C., 2023a. GVQA: Learning to answer questions about graphs with visualizations via knowledge base, in: Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems. doi:10.1145/3544548.3581067.
- Song, S., Li, C., Li, D., Chen, J., Wang, C., 2024. Graphdecoder: Recovering diverse network graphs from visualization images via attention-aware learning. *IEEE Transactions on Visualization and Computer Graphics* 30, 3074–3088. doi:10.1109/TVCG.2022.3225554.
- Song, S., Li, C., Sun, Y., Wang, C., 2023b. Vividgraph: Learning to extract and redesign network graphs from visualization images. *IEEE Transactions on Visualization and Computer Graphics* 29, 3169–3181. doi:10.1109/TVCG.2022.3153514.
- Song, S., Zhang, Y., Chen, Z., Qu, H., Wang, C., Li, C., 2026. Vizdefender: Unmasking visualization tampering through proactive localization and intent inference. *IEEE Transactions on Visualization and Computer Graphics* 32, 4720–4730. doi:10.1109/TVCG.2026.3694448.
- Song, S., Zhang, Y., Lin, Y., Qu, H., Wang, C., Li, C., 2025. Gvvst: Image-driven style extraction from graph visualizations for visual style transfer 31, 5975–5989. doi:10.1109/TVCG.2024.3485701.
- Stadtler, H., 2014. Supply chain management: An overview. *Supply Chain Management and Advanced Planning: Concepts, Models, Software, and Case Studies*, 3–28doi:10.1007/978-3-540-74512-9_2.
- Swaminathan, J.M., Smith, S.F., Sadeh, N.M., 1998. Modeling supply chain dynamics: A multiagent approach. *Decision Sciences* 29, 607–632. doi:10.1111/j.1540-5915.1998.tb01356.x.
- Xue, X., Li, X., Shen, Q., Wang, Y., 2005. An agent-based framework for supply chain coordination in construction. *Automation in Construction* 14, 413–430. doi:10.1016/j.autcon.2004.08.010.
- Yang, Y., Zhang, Y., Wu, M., Zhang, K., Zhang, Y., Yu, H., Hu, Y., Wang, B., 2025. TwinMarket: A scalable behavioral and social simulation for financial markets, in: ICLR 2025 Workshop on World Models: Understanding, Modelling and Scaling.
- Zhang, Y., Sun, J., Feng, L., Yao, C., Fan, M., Zhang, L., Wang, Q., Geng, X., Rui, Y., 2024. See widely, think wisely: Toward designing a generative multi-agent system to burst filter bubbles, in: Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems. doi:10.1145/3613904.3642545.
- Zhang, Y., Zhu, J., Wu, M., Qu, H., Song, S., 2026. Adapt: Adaptive lesson plan transformer for cross-regional and differentiated instruction. arXiv preprint arXiv:2606.17633 .
- Zhang, Z., Gao, J., Dhaliwal, R.S., Li, T.J.J., 2023. VISAR: A human-ai argumentative writing assistant with visual programming and rapid draft prototyping, in: Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology. doi:10.1145/3586183.3606800.
- Zhao, Q., Wang, J., Zhang, Y., Jin, Y., Zhu, K., Chen, H., Xie, X., 2024. Competeai: understanding the competition dynamics of large language model-based agents, in: Proceedings of the 41st International Conference on Machine Learning, JMLR.org. doi:10.5555/3692070.3694596.

Appendix A. Details of Semi-Structured Interview for the Formative Study

Appendix A.1. Interview Questions to Instructors

PART I: Background and Experience Teaching Background

- Which specific supply chain courses have you taught?
- Based on your experience, which concepts or knowledge points in supply chain management are the most difficult for students to understand and master?
- What do you think are the main reasons for these difficulties? Is it because the concepts are too abstract, or due to a lack of practical environment?
- What methods are you currently using (*e.g.*, case studies, classroom discussions, guest lectures) to help students address the limitations of textbook knowledge? Do you think practicing in a simulation system can help overcome these difficulties?
- In your attitudes toward university education for business students, do you believe students need more hands-on training in a realistic environment, either for the purpose of understanding concepts or for applying them in practice?
- Which simulation tools have you used in your teaching?
- (If the answer to the previous question is “Yes”)
- What is this simulation system? What functions does it support?
- What was the learning objective to achieve by using the educational supply chain simulation system?
- How did you integrate the simulation system into your courses (*e.g.*, in-class practice/tutorial/after-class exercise)
- What determines whether you choose to use a simulation for a particular concept?
- Did it efficiently help you achieve the learning objective?

PART II: Evaluation of the Existing Beer Game System

- Based on your impression on this system, What value or advantages do you think such tools bring to teaching? Which concepts can they help students intuitively understand? (*e.g.*, safety stock, reorder points, lead time, etc.)
- if we take a longer-term perspective, what do you think are the obvious limitations of this system? Or, in other words, which important supply chain knowledge points does it fail to cover?
- For a simple exercise aimed solely at helping students practice the fundamentals of ‘inventory management,’ do you think it is sufficient? Why?
- In your opinion, how much gap exists between this ‘single-chain’ model and a real-world supply chain? Does this gap affect the development of students’ skills or fail to meet your teaching needs?

PART III: Graph-Based Modeling of Supply Chain Networks From Limitations to Complexity Needs

- You just mentioned the limitations of the single-chain model. In the real world, supply chains are rarely linear; instead, they form complex networks with multiple suppliers, production facilities, and distribution channels. How necessary do you think it is for students to engage in simulations within a networked environment for their learning?
- Imagine a more powerful simulator: students can make decisions on a ‘graph’ (network) with multiple companies. Each has their own supply and demand connections. The students can even design or modify the network themselves. In your opinion, what kinds of concepts that are hard to explain through textbooks could be better understood with such a model?
- For instance, concepts like ‘the propagation of the bullwhip effect across a network,’ ‘risk diffusion caused by the disruption of a node (*e.g.*, factory shutdowns or port closures),’ or ‘how to balance inventory across multiple warehouses’—do? Do you think these could be better addressed in such a model?
- In a network-based or graph-based simulator, what types of decisions would you most want students to experience and practice?
- (Providing options for inspiration) “Would it be decisions about network structure (*e.g.*, where to build warehouses, which suppliers to choose), or operational decisions (*e.g.*, designing multi-echelon inventory strategies, planning transportation routes)?”
- In your view, what key features should an ideal supply chain network simulator have to achieve good teaching outcomes? (For example: data visualization, team collaboration, competitive modes, customizable scenarios, random events?)

PART IV: Ideal Simulation System for Teaching Support

- Do you think LLM can benefit the supply chain simulation? What is your attitudes toward a simulation system where the LLM models play the role of business managers? Do you think it could perform better than conventional rule-based models or parametric functions that control the company behaviors?
- Apart from the aforementioned features, what other features/functions would make a supply chain simulation most valuable for learning?

Appendix A.2. Interview Questions to Students

PART I: Background and Experience Educational Background

- What industry do you work in? Does your job involve supply chain management?
- When did you complete your business program?
- Which specific supply chain management courses did you take during your studies?

- Have you used any supply chain simulation tools during your studies?
- Please provide the system name and describe what it looks like and what it can do?
- How did these simulations facilitate learning?
- What frustrated you most when using these simulation tools?

PART II: Evaluation of the Existing Beer Game System Here is a standard supply chain simulation used to illustrate the bullwhip effect. The user can play the role of the distributor in a 4-tier supply chain. You are free to explore it.

- How intuitive was the interface and navigation?
- Do you think that this simulation effectively demonstrated the bullwhip effect? Please provide reasons based your experience.
- Do you think the system is oversimplified? For example, the system represents supply chain as a linear chain structure instead of a graph structure, making a disconnect from reality.
- What are the potential consequences in terms of educational value if the system is oversimplified?
- Is the system capable of handling your learning needs? If yes, can you name some scenarios in workplace that is related to the simulation game.
- What concepts do you think would be better demonstrated through a graph-based complex supply chain network?

PART III: Ideal Simulation System Learning Support

- Do you think it valuable to have two map views to display relation structure and the geographic location? How would it benefit the learning experience.
- Do you think it valuable to have time reverse functions to check the supply chain status in the past time stamps and enable "what if" analysis by changing he actions in the specific time point.
- Do you think it valuable to have the flexibility to modify supply chain networks by adding new companies or shutting down existing companies.
- Do you think it is valuable to introduce events into the environment?
- Do you think it is valuable to have high flexibility to modify the configuration of companies in the simulated environment (*e.g.*, price, cost, contracts, location, etc.)
- Do you expect the simulation system to be capable of serving diverse learning objectives/scenarios? If so, how would you like the system to support it?
- Do you expect the simulation system to have strong interactivity with the users? What kind of interaction do you expect to have?
- How would you prefer to receive feedback on your decisions?

Features and Functionality

- Apart from the above-mentioned features, what other features/functions would make a supply chain simulation most valuable for learning?
- What other aspects of supply chain simulation tools should we consider?

Appendix B. Details of Environment Setup

Appendix B.1. Attribute Variables

The key variables associated with each agent (*i.e.*, business entity) are defined in the system. These variables describe the state, decisions, and interactions of agents at different stages of the supply chain.

Let $A_{m,i}$ represent the i^{th} agent at stage m . The agents in the system are defined by the following variables.

1. The supplier set $S_{m,i}$ represents the suppliers of agent $A_{m,i}$. Manufacturers, as the top tier of the supply chain, are assumed to have a fixed source of raw materials for production. Therefore, their supplier sets are denoted as $S_{A_{M-1,i}} = \emptyset$ in the simulation.
2. Customer set $D_{m,i}$ includes all the customers of agent $A_{m,i}$. Retailers, as the bottom tier of the supply chain, are assumed to have a fixed group of customers at the local market. Thus, their customer sets are denoted as $D_{A_{0,i}} = \text{Customer}_i$ in the simulation.
3. Inventory level is denoted as

$$I = \{I_{m,i} \mid m \in \{0, \dots, M-1\}, i \in \{0, \dots, N_m\}\},$$

where $I_{m,i}$ represents the quantity of materials held by agent $A_{m,i}$. It follows

$$I \sim \text{Uniform}(lb_I, ub_I), \quad I_i \in \mathbb{Z} \quad (\text{B.1})$$

where $lb_I = 20$ and $ub_I = 25$.

4. Production capacity is defined as

$$C = \{C_{m,i} \mid m \in \{0, \dots, M-1\}, i \in \{0, \dots, N_m\}\},$$

where $C_{m,i}$ represents the quantity of materials that can be transformed into outgoing products by agent $A_{m,i}$ within a single

period. It follows:

$$C \sim \text{Uniform}(lb_C, ub_C), \quad C_i \in \mathbb{Z} \quad (\text{B.2})$$

where $lb_C = 20$ and $ub_C = 40$.

5. Profit rate $Pr_{m,i}$ is defined as

$$Pr = \{Pr_{m,i} \mid m \in \{0, \dots, M-1\}, i \in \{0, \dots, N_m\}\},$$

where $Pr_{m,i}$ represents the profit rate of agent $A_{m,i}$ for selling one product unit. It follows:

$$Pr \sim \text{Uniform}(lb_{Pr}, ub_{Pr}), \quad P \in \mathbb{Z}. \quad (\text{B.3})$$

where $lb_{Pr} = 1$ and $ub_{Pr} = 2$.

6. Sale price P is defined as

$$P = \{P_{m,i} \mid m \in \{0, \dots, M-1\}, i \in \{0, \dots, N_m\}\},$$

where P_i represents the price of a unit product that agent i offers to its downstream customers. It is determined by the order cost, production cost and the profit rate as follows:

$$P = (PC_{m,i} + \max_{j \in S_{m,i}} OC_{i \rightarrow j}^m) \cdot Pr_i^m \quad \forall m \in \{0, \dots, M-2\}. \quad (\text{B.4})$$

7. Backlog cost B is defined as

$$B = \{b_{m,i} \mid m \in \{0, \dots, M-1\}, i \in \{0, \dots, N_m\}\},$$

where $b_{m,i}$ denotes the cost incurred by agent $A_{m,i}$ for each unit of product that has not yet been fulfilled to the downstream customers for one time period. It follows:

$$B \sim \text{Uniform}(lb_B, ub_B), \quad B \in \mathbb{Z}. \quad (\text{B.5})$$

where $lb_B = 1$, $ub_B = 3$.

8. Holding cost H is defined as

$$H = \{h_{m,i} \mid m \in \{0, \dots, M-1\}, i \in \{0, \dots, N_m\}\},$$

where $h_{m,i}$ denotes the cost incurred by agent $A_{m,i}$ for storing each unit of product in inventory for one time period. It follows:

$$H \sim \text{Uniform}(lb_H, ub_H), \quad H \in \mathbb{Z}. \quad (\text{B.6})$$

where $lb_H = 1$ and $ub_H = 3$.

Appendix B.2. Relational Variables

The key variables associated with the interactions or relationships between agents are defined in the system. The generation details are described below.

1. Order cost OC is defined as

$$OC = \{OC_{A_{m,i} \rightarrow A_{m+1,j}} \mid m \in \{0, \dots, M-1\}, \\ i \in \{0, \dots, N_m\}, \\ j \in \{0, \dots, N_{m+1}\}\}, \quad (\text{B.7})$$

where $OC_{A_{m,i} \rightarrow A_{m+1,j}}$ represents the cost paid by agent $A_{m,i}$ to purchase each unit of feed materials from upstream agent $A_{m+1,j}$. It is equal to the sale price offered by upstream supplier $A_{m+1,j}$. I assume the order cost to be 0 (i.e., $OC_{A_{M-1,i}} = 0$) for all manufacturers $A_{M-1,i}$.

2. Lead time is defined as

$$L = \{L_{A_{m,i} \rightarrow A_{m+1,j}} \mid m \in \{0, \dots, M-1\}, \\ i \in \{0, \dots, N_m\}, \\ j \in \{0, \dots, N_{m+1}\}\}, \quad (\text{B.8})$$

where $L_{A_{m,i} \rightarrow A_{m+1,j}}$ represents the number of periods required to deliver products from agent $A_{m,i}$ to agent $A_{m+1,j}$. $L_{A_{m,i} \rightarrow A_{m+1,j}}$ is determined by the distance between two agents as follows:

$$L_{ij} = \lfloor \sqrt{(x_{m,i} - x_{m+1,j})^2 + (y_{m,i} - y_{m+1,j})^2} \rfloor \quad (\text{B.9})$$

Here, (x_i, y_i) and (x_j, y_j) represent the locations of agent $A_{m,i}$ and agent $A_{m+1,j}$, respectively. x_i, y_i, x_j, y_j follow $\text{Uniform}(0, 10)$.

Appendix B.3. Update Rules

At the end of each period, the variables in the simulated environment are updated to reflect the outcomes of the agents' decisions. Several rules are defined to govern these updates, which serve to maintain the consistency, track the agents' states, and propagate the effects of their decisions throughout the environment.

1. The actual order fulfillment $R_{A_{m+1,j} \rightarrow A_{m,i}}^t$ is constrained jointly by the upstream supplier $A_{m+1,j}$'s production capacity $C_{m+1,j}$, the inventory level $I_{m+1,j}$, the backlog $B_{m+1,j}$, and the order quantity $O_{m,t}$ placed by the agent $A_{m,i}$ itself. It can be formulated as:

$$R_{m,i}^t = \sum_{A_{m+1,j} \in \mathcal{S}_{m,i}} \min(B_{m+1,j}^{t-1} + O_{m,i}^t, C_{m+1,j}, I_{m+1,j}^{t-1} + R_{m,i}^{t-L_{A_{m+1,j} \rightarrow A_{m,i}}}), \quad (\text{B.10})$$

where $m \in \{0, \dots, M-2\}$.

Supply of raw materials to the manufacturers are assumed to be unlimited, thus have:

$$R_{M-1,i}^t = O_{M-1,i}^t \quad (\text{B.11})$$

2. The inventory level $I_{m,i}^t$ is updated to account for the deliveries received from the suppliers after the lead time $L_{A_{m+1,j} \rightarrow A_{m,i}}$ and the sales during the period t as follows:

$$I_{m,i}^t = I_{m,i}^{t-1} + R_{m,i}^{t-L_{A_{m+1,j} \rightarrow A_{m,i}}} - S_{m,i}^t, \quad (\text{B.12})$$

where $m \in \{0, \dots, M-1\}$.

3. The sales $S_{m,i}^t$ is determined by the order quantity it can fulfilled to the downstream customers.

$$S_{m,i}^t = R_{m-1,j}^t \quad (\text{B.13})$$

where $m \in \{1, \dots, M-1\}$, $A_{m-1,j} \in \mathcal{D}_{m,i}^t$.

For retailers, their sales $S_{0,i}^t$ is determined by the customer demand as

$$S_{0,i}^t = \min(B_{0,i}^{t-1} + D_i^t, C_{0,i}, I_{0,i}^{t-1} + R_{0,i}^{t-L_{A_{1,j} \rightarrow A_{0,i}}}), \quad (\text{B.14})$$

where $A_{1,j} \in \mathcal{S}_{0,i}^t$.

4. The backlog $B_{m,i}^t$ at period t depends on the previous backlog, the order $O_{m,i}^t$ placed by downstream customers during the current period t , and the actual sales $S_{m,i}^t$. It can be formulated as:

$$B_{m,i}^t = B_{m,i}^{t-1} + O_{m,i}^t - S_{m,i}^t, \quad m \in \{1, \dots, M-1\}, \quad (\text{B.15})$$

For retailers (*i.e.*, $m = 0$), the backlog depends on the customer demand $D_i(t)$ instead of downstream orders. The backlog is express as:

$$B_{0,i}^t = B_{0,i}^{t-1} + D_i(t) - S_{0,i}^t \quad (\text{B.16})$$

5. The profit $Pf_{m,i}$ is a combination of factors. It depends on the sale at the current period t and the price. Additionally, it depends on the order costs incurred to purchase materials from the suppliers during the same period. Furthermore, the inventory costs

and backlog costs also contribute to the profit calculation. Overall, it is formulated as:

$$\begin{aligned}
Pf_{i,m}^t &= P_{m,i} \cdot S_{m,i}^t \\
&- \sum_{A_{m+1,j} \in S_{m,i}^t} OC_{A_{m,i} \rightarrow A_{m+1,j}} \cdot R_{m,i}^t \\
&- BC_{m,i} \cdot B_{m,i}^t \\
&- HC_{m,i} \cdot I_{m,i}^t,
\end{aligned} \tag{B.17}$$

where $m \in \{0, \dots, M - 1\}$.

Appendix C. Semi-Structured Interview Questions for the User Study

Part 1: Engagement

- The Timeline Tree Map and its color encoding at each state motivates me to improve my decisions for achieving better consequences.
- The tiered level of supply chain challenges and the supply chain customization make the simulation feel dynamic and challenging.
- I would like to use and create events (e.g., storms, strikes) in the simulation to make it fun and realistic.
- The background story and the narratives of the different companies made the experience more immersive.
- The advice and encouragement from my AI co-manager (LLM agent) helped me stay motivated, especially when I was struggling.
- What was the most engaging or fun feature of the system for you?
- Can you describe a moment when you felt particularly motivated or, conversely, demotivated? What caused it?

Part 2: Learning Support

- The ‘Performance Quadrant’ (on the Operations tab) was an intuitive way to visualize the trade-off between my Fill Rate and Inventory Turns.
- The Timeline Tree Map was very helpful for exploring and comparing the outcomes of different decisions by doing time-reverse.
- The real-time feedback on profit and loss (e.g., visual encoding on entity icon, the data in analytics dashboard) helps me monitor my performance and motivates me to improve my decisions.
- The Analytics Dashboard (Financials) effectively helped me analyze trends and understand the performance of my supply chain regarding inventory and cost management.
- The Analytics Dashboard (Operations) effectively helped me analyze trends and understand the performance of my supply chain regarding inventory and order fulfillment.
- The Analytics Dashboard (Logistics) effectively helped me analyze trends and understand the performance of my supply chain regarding logistics and supplier reliability.
- The reasoning bubbles showing why AI agents made certain decisions were useful for understanding their strategies.
- The advice from the LLM co-manager helped me think more critically about my decisions.
- Which feature was most helpful for you to analyze the situation and plan your strategy? Why?
- Was there any information you needed for your analysis that was missing or hard to access?
- Using this system improved my understanding of fundamental supply chain dynamics (e.g., cause-and-effect relationships).
- I have a better understanding of key supply chain metrics (like inventory, backlog, profit) and how they are related after using the simulation.
- The system effectively helped me connect textbook theories with practical decision-making scenarios.
- By looking at the charts, I have a much better understanding of the trade-off between holding costs and the risk of stock-outs.
- By playing the intermediate level and advanced level, I have a more clear understanding of supply chain resilience and a competing market.
- Please describe one key insight or lesson you learned about supply chain management while using SupplyNet.
- How does the learning experience with SupplyNet compare to other methods you have used (e.g., lectures, case studies, other simulation games)?

SUS

- I think I would use this interface frequently.
- I found the interface unnecessarily complex.
- I thought the interface was easy to use.
- I would need help from a technical person to use this interface.
- I found the various functions in the interface well-integrated.

- I thought there was too much inconsistency in the interface.
- I would imagine most people would learn to use this interface quickly.
- I found the interface very cumbersome to use.
- I felt confident using the interface.
- I needed to learn a lot of things before I could get going with this interface.

Appendix D. Mathematical Formulas for Performance Metrics

This appendix provides the detailed mathematical formulas used in the Task-Oriented Analysis Console to calculate financial, operational, and logistics performance metrics.

Appendix D.1. Financial Performance Metrics

The financial metrics for an agent $A_{m,i}$ at time t are calculated as follows:

- **Total Profits ($Pf_{m,i}^t$):** The net gain after accounting for sales revenue and all operational costs:

$$Pf_{m,i}^t = \left(\sum_{A_{m-1,j} \in D_{m,i}} P_{m,i} \cdot S_{m,i}^t \right) - \text{Total Cost}_{m,i}^t$$

- **Total Sales ($\text{Sales}_{m,i}^t$):** The total quantity of products ordered by all downstream customers $D_{m,i}$ at time t :

$$\text{Sales}_{m,i}^t = \sum_{A_{m-1,j} \in D_{m,i}} O_{m-1,j}^t$$

- **Total Cost ($\text{Total Cost}_{m,i}^t$):** The sum of ordering, production, holding, and backlog costs:

$$\text{Total Cost}_{m,i}^t = \text{OrderCost}_{m,i}^t + \text{ProdCost}_{m,i}^t + \text{HoldCost}_{m,i}^t + \text{BacklogCost}_{m,i}^t$$

- **Ordering Cost ($\text{OrderCost}_{m,i}^t$):** The cost of purchasing materials from upstream suppliers $S_{m,i}$, where $OC_{A_{m,i} \rightarrow A_{m+1,j}}$ is the unit cost and $O_{m,i}^t$ is the order quantity:

$$\text{OrderCost}_{m,i}^t = \sum_{A_{m+1,j} \in S_{m,i}} OC_{A_{m,i} \rightarrow A_{m+1,j}} \cdot O_{m,i}^t$$

- **Production Cost ($\text{ProdCost}_{m,i}^t$):** Calculated based on the unit production cost $PC_{m,i}$ and the production capacity $C_{m,i}$:

$$\text{ProdCost}_{m,i}^t = PC_{m,i} \cdot C_{m,i}$$

- **Holding Cost ($\text{HoldCost}_{m,i}^t$):** The cost of storing inventory $I_{m,i}^t$ with unit holding cost $h_{m,i}$:

$$\text{HoldCost}_{m,i}^t = h_{m,i} \cdot I_{m,i}^t$$

- **Backlog Cost ($\text{BacklogCost}_{m,i}^t$):** The penalty for unfulfilled orders $B_{m,i}^t$ with unit backlog cost $b_{m,i}$:

$$\text{BacklogCost}_{m,i}^t = b_{m,i} \cdot B_{m,i}^t$$

Appendix D.2. Operations Performance Metrics

The operations performance metrics are calculated as averages over a selected time window $[t_0, t]$ within the timeline tree map:

- **Avg Inventory On Hand ($\bar{I}_{m,i}$):** The average inventory level maintained during the period, calculated based on the selected decision path in the timeline tree map:

$$\bar{I}_{m,i} = \frac{1}{t - t_0 + 1} \sum_{\tau=t_0}^t I_{m,i}^\tau$$

- **Avg Upstream Fill Rate ($\overline{FR}_{m,i}$):** The average ratio of received materials $R_{m,i}^\tau$ to the total quantity ordered $O_{m,i}^\tau$ from all suppliers $S_{m,i}$:

$$\overline{FR}_{m,i} = \frac{1}{t - t_0 + 1} \sum_{\tau=t_0}^t \left(\frac{R_{m,i}^\tau}{O_{m,i}^\tau} \right)$$

- **Avg Upstream On-Time Delivery (\overline{OTD}_S):** The percentage of deliveries from suppliers that arrived within the expected lead

time $L_{A_{m+1,j} \rightarrow A_{m,i}}$. Let $\hat{L}_{A_{m+1,j} \rightarrow A_{m,i}}^k$ be the actual delivery time for the k -th order:

$$\overline{OTD}_S = \frac{\text{Count}(\hat{L}^k \leq L)}{\text{Total Deliveries}} \times 100\%$$

- **Avg Downstream On-Time Delivery (\overline{OTD}_D):** The percentage of the target agent's own deliveries to downstream customers $D_{m,i}$ that were fulfilled within the expected lead time $L_{A_{m,i} \rightarrow A_{m-1,j}}$:

$$\overline{OTD}_D = \frac{\text{Count}(\hat{L}^k \leq L)}{\text{Total Deliveries}} \times 100\%$$

Appendix D.3. Logistics Performance Metrics

The logistics performance metrics provide a high-level overview of the supply chain's efficiency and responsiveness over the period $[t_0, t]$:

- **Avg Fill Rate ($FR_{m,i}$):** The average ratio of fulfilled sales $S_{m,i}^\tau$ to the total order quantity $O_{m,i}^\tau$ received from downstream customers:

$$FR_{m,i} = \frac{1}{t - t_0 + 1} \sum_{\tau=t_0}^t \frac{S_{m,i}^\tau}{O_{m,i}^\tau}$$

- **Average Inventory ($\bar{I}_{m,i}$):** The average quantity of products held in stock over the time period:

$$\bar{I}_{m,i} = \frac{1}{t - t_0 + 1} \sum_{\tau=t_0}^t I_{m,i}^\tau$$

- **Average Inventory Turns ($IT_{m,i}$):** A standard measure of how many times the inventory is sold and replaced over the period. It is calculated as the total Cost of Goods Sold (COGS) divided by the average inventory value. The unit cost is defined as the sum of unit ordering cost, unit production cost, and unit holding cost ($PC_{m,i} + \max_{j \in S_{m,i}} OC_{A_{m,i} \rightarrow A_{m+1,j}} + h_{m,i}$):

$$IT_{m,i} = \frac{\sum_{\tau=t_0}^t \text{Total Cost}_{m,i}^\tau}{\bar{I}_{m,i} \cdot (PC_{m,i} + \max_{j \in S_{m,i}} OC_{A_{m,i} \rightarrow A_{m+1,j}} + h_{m,i})}$$

- **Average Weeks of Supply ($WOS_{m,i}$):** A standard measure of how many weeks (or time periods) the current inventory will last based on average demand:

$$WOS_{m,i} = \frac{\bar{I}_{m,i}}{(\sum_{\tau=t_0}^t S_{m,i}^\tau) / (t - t_0 + 1)}$$

Appendix E. LLM Prompt Infrastructure for Decision Making

This appendix details the prompt construction and LLM integration within the SupplyNet framework. It clarifies the role of the "Golden Rules," provides examples of prompt templates used for the textualization of extracted subgraphs, and demonstrates how the exact natural language outputs from the LLM are converted into numerical simulation decision variables.

Appendix E.1. Content and Role of the "Golden Rules"

The "Golden Rules" serve as foundational operational principles that equip the LLM agents with essential domain knowledge in supply chain management. By embedding these rules into the prompt, we ensure that the agents' decision-making processes are grounded in established economic and operational logic, such as inventory equilibrium, profitability margins, and the mitigation of the bullwhip effect.

```

1 GOLDEN_RULES = (
2     "Operational Principles for Inventory Optimization:\n"
3     "1. Inventory Equilibrium: Open Orders must equal 'Expected Downstream Demand + Backlog' to
4     ensure stability.\n"
5     "2. Profitability Margin: Sales Price must exceed the aggregate of Production and
6     Procurement Costs.\n"
7     "3. Procurement Constraints: Order placement is restricted to identified upstream suppliers
8     only.\n"

```

```

6     "4. Lead-Time Mitigation: Proactive order placement is required to account for supplier lead
       times.\n"
7     "5. Strategic Sourcing: Optimize supplier selection based on lead-time efficiency vs.
       procurement costs.\n"
8     "6. Bullwhip Effect Suppression: Distribute order volumes across multiple periods to
       minimize variance amplification."
9 )

```

Appendix E.2. Textualization of Extracted Subgraphs

To provide the LLM with the current supply chain status, we extract the relevant subgraph for each agent and convert it into a structured natural language format (Textualization). This graph-based environment status is represented as two markdown tables: a Node Table for agent attributes (e.g., inventory, costs) and an Edge Table for relational properties (e.g., lead times, delivery status).

```

1 NODE_TABLE_TEMPLATE = """
2 ### Node Table (Agent Attributes)
3 | Node ID | Attributes (<node label>, <role>, <inventory level>, <sale price>, <costs>) |
4 |:-----|:-----|
5 | <id>    | Inventory: <value>, Price: <value>, Production Cost: <value>, Order Cost: <value> |
6 """
7
8 EDGE_TABLE_TEMPLATE = """
9 ### Edge Table (Supply Chain Relations)
10 | Source | Destination | Attributes (<lead time>, <order placement>, <delivery status>) |
11 |:-----|:-----|:-----|
12 | <id>   | <id>       | Lead Time: <value> |
13 | <id>   | <id>       | On-the-way delivery: <value> |
14 """

```

Appendix E.3. LLM Output and Conversion to Simulation Variables

To ensure that the natural language generated by the LLM can be seamlessly integrated back into the simulation engine, we enforce a strict JSON output schema. The LLM is instructed to output exact numerical parameters (e.g., integer values for order quantities) and specific string identifiers (e.g., supplier IDs for network configuration). The simulation engine directly parses this JSON object to update the environment state for the next timestep.

```

1 JSON_OUTPUT_FORMAT = {
2     "supplier_selection": {
3         "reasoning": "<1-2 sentence justification>",
4         "removals": ["<id>"],
5         "additions": ["<id>"]
6     },
7     "order_placement": {
8         "orders": [
9             {"supplier_id": "<id>", "quantity": "<int>"}
10        ]
11    },
12    "performance_review": {
13        "business_performance": "<string_analysis_max_15_words>",
14        "feeling": "<satisfied/concerned>"
15    }
16 }

```

Appendix E.4. Integrated Prompt Example

The final prompt assembled for the LLM integrates the system role, the textualized graph data, contextual updates (events and downstream demands), the Golden Rules, and the strict output instructions.

```
1 TASK_DESCRIPTION = (  
2     "Task I: Supply Network Configuration - Evaluate upstream edges and modify the supplier list  
3     .\n"  
4     "Task II: Procurement Execution - Determine optimal order quantities and assess round  
5     performance."  
6 )  
7  
8 prompt = f"""  
9 ### SYSTEM ROLE  
10 You are the lead Supply Chain Manager for {agent_role}.  
11 Simulation Round: {round_period}.  
12  
13 ### ENVIRONMENT STATUS (GRAPH REPRESENTATION)  
14 {node_data}  
15 {edge_data}  
16  
17 ### CONTEXTUAL UPDATES  
18 - Emergent Events: {event_msg}  
19 - Downstream Requirements: {downstream_msg}  
20  
21 ---  
22 ### OPERATIONAL CONSTRAINTS  
23 {GOLDEN_RULES}  
24 ---  
25 ### DECISION TASKS  
26 {TASK_DESCRIPTION}  
27  
28 ### OUTPUT INSTRUCTIONS  
29 Return your decisions strictly in the following JSON format. Ensure all IDs match the provided  
30 tables.  
31 {JSON_OUTPUT_FORMAT}  
32 """
```

Appendix F. Implementation Details

The prototype of SupplyNet is implemented using a modern web technology stack, ensuring a responsive and interactive user experience.

Frontend and Visualization: The user interface is built with Vue 3.5 and Vite 6.1, utilizing Element Plus for UI components. The interactive supply chain network is rendered using Vue Flow (@vue-flow/core), which supports drag-and-drop interactions and dynamic node updates. Data visualization in the Analysis Console is powered by Chart.js and ECharts, providing real-time interactive charting capabilities. The onboarding tutorial is implemented using Driver.js, and internationalization is supported via vue-i18n.

Backend and Simulation Core: The backend simulation engine is implemented in Python (version 3.14+). It manages the global state of the supply chain, calculates performance metrics based on standard formulas, and orchestrates the timestep progression. Communication between the frontend and the backend is handled via Axios over RESTful APIs.

LLM Agents: We developed a custom multi-agent framework tailored specifically for this supply chain simulation. All agents in the system are powered by OpenAI's gpt-4o model to ensure consistent and high-quality reasoning capabilities across the simulation. The agents receive contextualized subgraphs and task descriptions, and return structured JSON responses that are directly parsed by the simulation core.

Database and State Management: The supply chain state for each timestep is persisted as JSON files. These files store comprehensive agent properties, including costs, profits, inventory levels, and supply relationships. This lightweight approach allows for efficient state retrieval and supports the branching mechanism in the Timeline Tree Map.

System Requirements and Token Costs: Running the system locally requires Node.js v22 and Python 3.14+, with a minimum of 8GB RAM recommended. Since the LLM reasoning is performed via cloud APIs, no dedicated GPU is required. An active OpenAI API Key is necessary to run the simulation. Based on current pricing, a typical simulation session processing 100k to 200k tokens costs approximately \$0.25 to \$2.00. While the system can be configured to use smaller local LLMs to eliminate costs, the quality of complex supply chain reasoning and simulation stability may be degraded compared to gpt-4o.