Semantic Mediation: An LLM-Based Approach for Ground Truth in Human-in-the-Loop Automated Planning in Supply Chain Networks

Ryan R. Farish[1,2]**, Emanuele De Pellegrin** [2] **Ronald P. A. Petrick**[1]

[1]Department of Computer Science, Heriot-Watt University, Edinburgh, Scotland, UK
[2]School of Informatics, University of Edinburgh, Edinburgh, Scotland, UK
R.Farish@ed.ac.uk, E.DePellegrin@ed.ac.uk, R.Petrick@hw.ac.uk

Abstract

Modern supply chain networks (SCN) operate in fast paced dynamic environments where planning models can suffer from an inaccurate state, a disconnect between the domain representation and physical reality. We present a proof of concept approach for a semantic mediation framework utilising Large Language Models (LLMs) as a communication layer between the physical world and the ground truth of the automated planner. We adopt a human-in-the-loop approach for our framework, with telemetry checks for ground truth verification. By using a supply chain illustrative example to guide the paper, we introduce a pipeline of constraint validation, where the LLM verifies the operator directives against the telemetry data to generate validated predicates for the automated planner. By positioning the LLM as a communication layer between the operator, ground truth, and automated planner, we aim to use the approach to mitigate the misunderstanding of incorrect interpretations between the operator, real world, and planner. We also acknowledge the open challenges that this framework aims to address in future work.

Introduction

A Supply Chain Network (SCN) is a data-driven strategic process to configure and optimise a supply chain, including actors, resources, and interdependent activities (Harland 1996). For effective management, automated planners require explicit domain knowledge formalised as symbolic action and constraints to generate valid solutions. Reliable SCNs operate under hard constraints (Yildiz et al. 2016). While automated planners such as temporal and contingent planning allow alternative action branches to be built for complex time-based and branching logic (Ghallab, Nau, and Traverso 2004), the efficacy depends on the initial formulation of symbolic models, validation, and maintenance. Humans often pursue outcomes with self-defined goals and domains, and how constraints affect the steps to achieve these goals (Deci and Ryan 2000). Temporal and contingent planning cannot easily adapt to human nuances such as values and contexts.

A bottleneck arises from the reality gap, a disconnect between the dynamic real world and the symbolic representations. While Large Language Models (LLMs) excel at natural language processing, their tenancy for hallucination and lack of formal guarantees makes them unreliable in safety-critical domains (Vallati et al. 2025). This requires manual intervention, costing time and potential errors in with scaling to ensure a planner's internal state remains consistent wit the ground truth. Mismatches between the planning problem and the real world could raise severe safety and trust concerns. This has negative consequences for the SCN and stakeholders involved: including erosion of trust, financial penalties, safety risks, resource wastage, and operational paralysis.

We address these risks by proposing a semantic mediation layer by leveraging LLMs to support human operators through a human-in-the-loop method in acquisition of real-world environmental telemetry information, proactively grounding symbolic predicates with the ground truth before a solver begins. We demonstrate this through a SCN illustrative example, illustrating how using an LLM as a communication layer, rather than for plan generation, aligns strategic human intent and grounded reality. We follow this with background and related work, an outline of the proposed framework, and then conclude with the next stages of future research for this framework.

Background and Related Work

Supply chain networks (SCNs) are a highly dynamic and open world environment, often subjected to unexpected events. SCN systems must continuously adapt to the rapid convergence of different scenarios (Dash et al. 2019). The initial state of the world may become obsolete before a solution is reached, or even before starting the search for a planning problem (Cashmore et al. 2018; Coles et al. 2024). It is crucial for an SCN to process abstract data and translate them to their real-world meaning for operational effectiveness. One approach is to use semantic reasoning through Large Language Models (LLMs) to bridge planning models and SCN requirements by providing contextual reasoning (Song et al. 2026). Although these dynamic states can be monitored through the physical ground truth, because the state is always changing in SCNs, there is a risk that planner runs outdated information, resulting in an unreliable solution. Traditional Monitor & Execution (M&E) architectures employ hard-coded translations to sync reality with planning models (Ghallab, Nau, and Traverso 2004). M&E frameworks require rigid telemetry schemas and monitoring manual code updates to map out the new reality as traditional execution layers lack contextual repair of unmodeled state mismatches (Fox et al. 2006). Our proposed framework bypasses this rigidity by acting as a flexible layer by allowing an LLM to observe abstract contexts without the manual structural modifications.

Automated planning is the process of finding a sequence of actions through decision making to achieve an end goal (Ghallab, Nau, and Traverso 2004), with logistics representation referring back to the logistics benchmark planning problem operating under a closed world assumption (McDermott 2000). Knowledge engineering has relied on manual domain modelling methods, where the skills of a knowledge engineer significantly impact the quality of the resulting application (Studer, Benjamins, and Fensel 1998). One of the most common approaches, based on the STRIPS planning language (Fikes and Nilsson 1971), is the manual modelling of a domain through specifications based on textual or structural descriptions using the Planning Domain Definition Language (PDDL) (McDermott et al. 1998): containing predicates and actions/operators A, and a problem file mapping the initial state I, and goals G. PDDL allows us to distinguish between hard constraints, absolute preconditions that must be satisfied , and soft constraints, preferences a system should attempt to satisfy. Manual domain modelling may result in errors (Bhatnagar et al. 2022), in addition to the increased time it takes to find a solution scaling with domain complexity (Wójtowicz, Puszyński, and Gałuszka 2024).

Various planning frameworks can be applied. While an approach such as classical planning is sufficient for deterministic closed-world environments through full observation (Ghallab, Nau, and Traverso 2004), the dynamic demands of SCNs require both numeric fluents to track resource constraints and temporal planning to meet strict delivery windows. A classical planning approach struggles due to incomplete knowledge (Kaelbling, Littman, and Cassandra 1998). Numeric planning introduces quantitative fluents to track continuous variables (Scala et al. 2016), and temporal planning uses durative actions granting reason about exogenous events (Jiménez, Jonsson, and Palacios 2015).

Previous work has explored domain generation, the automated process of the state spaces and action models from high-level requirements (Ghallab, Nau, and Traverso 2004). Natural Language Processing (NLP) based techniques, such as LLMs, are a potential option over domain modelling to automate the creation of conceptual models from textural requirements (Guan et al. 2023; Oswald et al. 2024). SCNs are moving towards collaborative intelligence, where humans provide the goals and the AI manages the information (Elgeddawy 2026). Although LLMS can play a valuable role within knowledge engineering, it is argued they will not substitute human experts (Vallati et al. 2025). LLMs often fail because they frequently hallucinate tokens and lack a world model that is not grounded in reality as they scale (Tantakoun, Muise, and Zhu 2025). Recall and complex domains beyond a simple domain description remain a concern (Chen et al. 2023), leading to to decision paralysis (Stein et al. 2025).

Researchers have used non-LLM techniques in earlier efforts, through Action Model Learning (AML), where systems induce domain rules and world constraints from observation traces to alleviate manual input by operators (Jiménez et al. 2012), replanning and plan repair with recovery of a mismatched state of worlds (Fox et al. 2006), aligning user-defined constraints based on using automated planning to increase task engagement (James et al. 2025), and deep reinforcement learning. The downsides to these approaches include computational complexity and lack of explainability (Vallati et al. 2025). By assuming a fixed PDDL domain model, this allows us to bypass the operational costs for AML by focusing on the translation of unstructured real-world data into precise I updates.

It is important to maintain proactive approach, where the human is involved within the system process for additional resilience and oversight - known as a Human-in-the-Loop (HITL) approach (Ferguson and Allen 2007). HITL within automated planning can be limited by the representation gap of the knowledge engineer. Where LLMs can be used for conversion of a planning domain into natural language prompts, the logic of the planner can be translated into a human-readable explanation (Stein et al. 2025). Including user-defined goals through an LLM interface, a system can adjust for potential failures through a proactive approach with the operator (Merlo, Lagomarsino, and Ajoudani 2025). By mediating between the operator and the formal solver, the LLM can contribute towards the solutions of a plan being both strategically aligned with the human operator, but also grounded in the physical ground truth of reality with human oversight.

Illustrative Example

To demonstrate the feasibility of our proposed system, we discuss a real-world SCN scenario which the system could be applied. This scenario involves the transport of high-value fragile goods in Scotland, from the Isle of Skye to a distribution hub in Glasgow. The primary objective of the automated planner in this domain is to generate a sequence of valid transport actions to move cargo from a central depot to a distribution centre. While a classical symbolic solver generates an optimal plan under static conditions, it remains blind to real-time telemetry information. We simulate a multi-event disruption when the ground truth shifts unexpectedly. Ordinarily, a planner would either rely on an outdated model, leading to safety concerns, or require a human to manually reformulate the domain. This pipeline translates these raw alerts into predicates to ensure the planner remains grounded in reality.

Scenario Setup

The SCN domain setup (Figure 1) contains routes that involve ferries and a mixture of main and side roads. Under optimal conditions, all routes are passable without delays. However, we simulate a disruption event where severe weather has resulted in an impact for some routes.

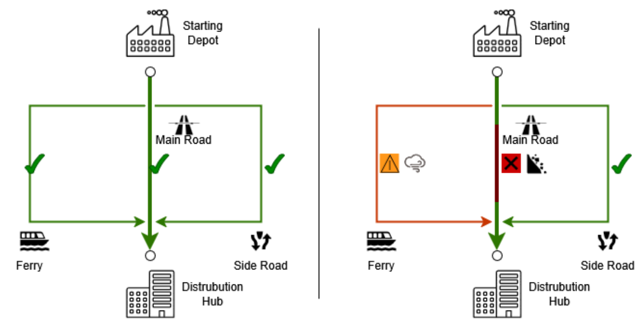

Figure 1: Scenario domain. The left shows optimal conditions, while the right shows the disruption event with a landslide on the main road, and high winds for sea traffic.

Application

The following demonstrates the pipeline of the framework, and the interaction between the human operator, ground truth, and planner.

- Intent Capture: The operator provides a strategic directive: "The road south, and maybe others, could be blocked according to news reports. Please reroute shipments to Glasgow, prioritising the 7:00PM export window. Check if the ferries are still an option, or if the roads are still viable."
- Ground Truth Validation Check: The LLM triggers a tool to call the ground truth via an API call to the national traffic and weather database. The Ground Truth Confirms:

  Status(South-Road): CLOSED
  Status(West-Road): OPEN
  Status(Ferry): DELAYED (High Winds)

- Constraint Injection: The LLM identifies that the southern road is impassable, the western side-road road is open, and the ferry is not reliable due a delay from an API check on crossing times. It introduces three constraints into the problem:

  (not(path-available south-road))
  (increase (travel-time ferry) 120 minutes)
  (preference p1 (exists (?v) (traversed ?v west-road)))

- Planner Execution: The constraints are injected into the planning problem. The planner is now aware of the constraints and finds a solution to the most optimal route ensuring the deadline is still feasible, with the following example output:

  QUERY-WEATHER-STATUS (sml-mediator, ferry-terminal) (Result: Unsafe/High Winds)
  DRIVE-TRUCK (truck01, depot, side-road-entry)
  DRIVE-TRUCK (truck01, side-road-entry, side-road-segment-A)
  DRIVE-TRUCK (truck01, side-road-segment-A, distro-centre-inland-route)
  ARRIVE-AT-DESTINATION (truck01, distro-centre)

  For the planner to complete the process in adequate time, a closed-loop approach is applied to verify the initial state has not shifted, monitoring constantly of the environment against the problem state used by the planner. If a mismatch is detected, the system triggers a re-plan to account for the new reality.

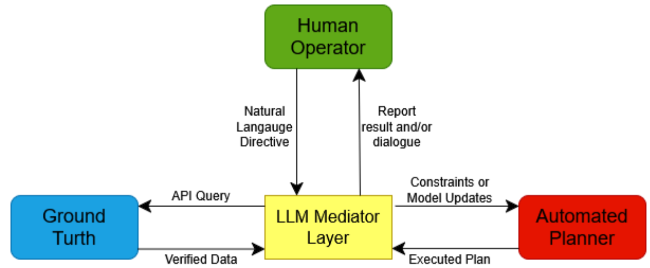

Figure 2: Proposed framework diagram of the system outline.

Observations

The illustrative example highlights two main benefits of the framework. First, the human operator did not need to manually identify the affected nodes in the network, saving time and potential error. They provide the intent, while the LLM crossed reference the intent to reality with the ground truth. Second, a stand-alone LLM may have either ignored the information of a possible ferry delay or taken it as absolute truth. By checking and validating the ground truth first, the framework ensured the planner only received valid data based on real-world telemetry.

System Outline

We propose a semantic mediation layer to bridge the gap between the three modules of the human operator, the automated planner, and the physical ground truth. In this framework, the purpose of the LLM is to serve as a non-generative layer that allows for the flow of knowledge between these three modules.

The framework (Figure 2) is designed to prevent miscommunication between these modules while ensuring logical grounding. We can breakdown this framework into three interfaces:

- Human-to-LLM, capturing natural language from the operator.
- Ground Truth-to-LLM, using tool calling to query real-time supply chain network telemetry and status.
- LLM-to-Planner, translating verified information into symbolic representations to solve.

By isolating the LLM as a communication later instead of a generative planner, we can maintain a safety buffer for symbolic representation against known LLM pitfalls, in addition to collaboration with the human operator for cross checking. The LLM may communicate the need for an alternative plan, but the ground truth, planner, and the human operator remain the final validations of the solution. The LLM layer only dynamically updates the numeric fluents and initial state I object fluents based on real-time data, preventing structural logic changes.

Pipeline

Unlike traditional systems where a human must provide input and manually update the initial state I, our proposed framework translates unstructured observations into planning constraints. Between the three modules stated above, this is structured in the following pipeline.

First, when a human operator provides a directive, the LLM identifies the relevant domain objects within the statement provided. For example, it may recognise a specific node within the supply chain has an effect, and may impact a set of actions. Second, before replanning, the LLM triggers a tool-call to the network's telemetry data (for instance, an API weather check) and performs a truth query. If the telemetry data shows the opposite to be true, the LLM flags a mismatch to the operator instead of a re-plan being required. Finally, once the ground truth has been verified, the framework presents the proposed PDDL updates to the human operator for explicit validation and approval via the interface dialogue. If approved, the LLM injects a formal constraint within the domain through a script, modifying the initial state I or the cost function within the problem file - ensuring constraints do not flow directly from the LLM to the planner without HITL oversight, mitigating the risk of unvetted or hallucinated constraints entering the solution. When the planner runs, its internal logic will now be presented with the new state, and then presented to the operator.

If an operator provides an intent that contradicts the ground truth, the system does not update the planner. Instead, the LLM generates a feasibility report explaining to the user why the strategy cannot be implemented based on current telemetry. This allows the operator the chance to refine their strategy based on the ground truth of the system, before any resources are spent on an outcome that could result in mismanagement of resources. The end goal is to provide the operator with an explanation of the system's behaviour, while the system remains logically consistent and accurate within the world model.

Discussion and Open Challenges

One of the core intentions of knowledge engineering is the explainability gap of trust, where a human must trust the system's output as reliable and accurate. If the framework routes a vehicle from one road to another, it requires justification for all human stakeholders involved. This raises the question of how we can ensure that the explanations of an LLM are grounded in factual data instead of hallucinated logic of the organisation and supply chain structure. A proposed solution would be the implementation of traceable reasoning, where the LLM must include citations to the specific ground truth data point in its response to both the human and the planner. We can also ask how can the LLM decide the threshold for safe assumptions in the case of different logic due to the ambiguous nature of natural language. Further work can explore this aspect, where the LLM proactively engages with the human for clarification, where qualitative information is converted into a numerical constraint before the planner provides a solution.

Within supply chain management, strategic intent can sometimes conflict with the ground truth. For example, an operator may want to meet a strict deadline, but the ground truth reports severe disruption. In cases like these, it would be prudent to include how we can define a fair negotiation within the system, where the LLM can suggest trade-offs and become an active negotiator with the options presented.

As the supply chain network grows larger, so does the noise of information within it (Harland 1996). The system must be able to determine which ground truth information changes are deemed of relevance to the world, the operator's intentions, and the current plan. If the system considers every minor delay to the operator for example, it could cause an overabundance of information that can negatively impact both the end user and the plan as time progresses. Therefore, the system must remain attentive to the real time situation.

Additional open challenges remain, including issues of latency and synchronisation where telemetry data may be either delayed, unavailable, or outdated. Although the current framework focuses on a single simplistic illustrative example, global supply chains involve a much larger scale of concurrent operators and actors. Therefore, scaling the framework may become a challenge and may require multiple LLM instances where the ground truth is synchronised to prevent conflicting constraints (i.e., two operators unknowingly are trying to use the same limited vehicle capacity). As the domain grows larger and evolves overtime with new real world changes, users are required to update a static domain file with this information, and thus grows in complexity. Instead of updating the ground truth, we can ask how long-term trends observed within the framework from the ground truth can be implemented within the rules of the world.

Conclusion

This paper proposed a semantic mediation framework to support automated planning in the context of supply chain networks. By positioning a large language model as a mediator between the operator, ground truth, and planner, we aim to mitigate the risk of hallucinated and untrue domains to their real world counterparts though a human-in-the-loop approach. The primary contributions of such a framework include:

- Aligning human strategic intentions with the ground truth of the environment.
- Processing unstructured real-world disruptions into formal predicates.
- Granting a cooperative approach where the human operator is kept in the loop, and contributes towards, the inputs and outputs of the automated planner.

Our illustrative example demonstrates the feasibility of the framework, addressing the gap of ensuring automated plans are not only optimal, but are also validated in the reality of the ground truth though human cooperation. We also discuss and present several open challenges to the implementation of such a framework. In our future work, our aim is to address the core challenges in testing the validation of our proposed framework.

References

Bhatnagar, S.; Mund, S.; Scala, E.; McCabe, K.; McCluskey, T. L.; Vallati, M.; et al. 2022. On-the-Fly Knowledge Acquisition for Automated Planning Applications: Challenges and Lessons Learnt. In *ICAART (2)*, 387–397.

Cashmore, M.; Coles, A.; Cserna, B.; Karpas, E.; Magazzeni, D.; and Ruml, W. 2018. Temporal planning while the clock ticks. In *Proceedings of the International Conference on Automated Planning and Scheduling*, volume 28, 39–46.

Chen, K.; Yang, Y.; Chen, B.; López, J. A. H.; Mussbacher, G.; and Varró, D. 2023. Automated domain modeling with large language models: A comparative study. In *2023 ACM/IEEE 26th International Conference on Model Driven Engineering Languages and Systems (MODELS)*, 162–172. IEEE.

Coles, A.; Karpas, E.; Lavrinenko, A.; Ruml, W.; Shimony, S. E.; and Shperberg, S. 2024. Planning and acting while the clock ticks. In *Proceedings of the International Conference on Automated Planning and Scheduling*, volume 34, 95–103.

Dash, R.; McMurtrey, M.; Rebman, C.; and Kar, U. K. 2019. Application of artificial intelligence in automation of supply chain management. *Journal of strategic innovation and sustainability*, 14(3): 43–53.

Deci, E. L.; and Ryan, R. M. 2000. The" what" and" why" of goal pursuits: Human needs and the self-determination of behavior. *Psychological inquiry*, 11(4): 227–268.

Elgeddawy, Y. 2026. Human–AI Collaboration Models in Supply Chain Decision-Making. *International Journal of Scientific Research in Science, Engineering and Technology*, 13: 246–261.

Ferguson, G.; and Allen, J. 2007. Mixed-initiative systems for collaborative problem solving. *AI magazine*, 28(2): 23–23.

Fikes, R. E.; and Nilsson, N. J. 1971. STRIPS: A new approach to the application of theorem proving to problem solving. *Artificial intelligence*, 2(3-4): 189–208.

Fox, M.; Gerevini, A.; Long, D.; Serina, I.; et al. 2006. Plan Stability: Replanning versus Plan Repair. In *ICAPs*, volume 6, 212–221.

Ghallab, M.; Nau, D.; and Traverso, P. 2004. *Automated Planning: theory and practice*. Elsevier.

Guan, L.; Valmeekam, K.; Sreedharan, S.; and Kambhampati, S. 2023. Leveraging pre-trained large language models to construct and utilize world models for model-based task planning. *Advances in Neural Information Processing Systems*, 36: 79081–79094.

Harland, C. M. 1996. Supply chain management: relationships, chains and networks. *British Journal of management*, 7: S63–S80.

James, L.; De Pellegrin, E.; Genga, L.; Montagne, B.; and Van Gorp, P. 2025. Automated Planning of Entertainment Content for Digital Health Interventions: A Technical Feasibility Study on Balance VS Fun. In *International Conference on Entertainment Computing*, 193–206. Springer.

Jiménez, S.; De La Rosa, T.; Fernández, S.; Fernández, F.; and Borrajo, D. 2012. A review of machine learning for automated planning. *The Knowledge Engineering Review*, 27(4): 433–467.

Jiménez, S.; Jonsson, A.; and Palacios, H. 2015. Temporal planning with required concurrency using classical planning. In *Proceedings of the International Conference on Automated Planning and Scheduling*, volume 25, 129–137.

Kaelbling, L. P.; Littman, M. L.; and Cassandra, A. R. 1998. Planning and acting in partially observable stochastic domains. *Artificial intelligence*, 101(1-2): 99–134.

McDermott, D.; et al. 1998. Pddl—the planning domain definition language. *Technical Report, Tech. Rep.*

McDermott, D. M. 2000. The 1998 AI planning systems competition. *AI magazine*, 21(2): 35–35.

Merlo, E.; Lagomarsino, M.; and Ajoudani, A. 2025. A human-in-the-loop approach to robot action replanning through LLM common-sense reasoning. *IEEE Robotics and Automation Letters*.

Oswald, J.; Srinivas, K.; Kokel, H.; Lee, J.; Katz, M.; and Sohrabi, S. 2024. Large language models as planning domain generators. In *Proceedings of the International Conference on Automated Planning and Scheduling*, volume 34, 423–431.

Scala, E.; Haslum, P.; Thiébaux, S.; and Ramirez, M. 2016. Interval-based relaxation for general numeric planning.

Song, Z.; Xie, Y.; Yang, L.; and Zhao, Y. 2026. Large language models in supply chain management: a systematic literature review and application framework. *International Journal of Production Research*, 1–41.

Stein, K.; Fišer, D.; Hoffmann, J.; and Koller, A. 2025. Automating the generation of prompts for llm-based action choice in pddl planning. In *Proceedings of the International Conference on Automated Planning and Scheduling*, volume 35, 250–259.

Studer, R.; Benjamins, V. R.; and Fensel, D. 1998. Knowledge engineering: Principles and methods. *Data & knowledge engineering*, 25(1-2): 161–197.

Tantakoun, M.; Muise, C.; and Zhu, X. 2025. LLMs as planning formalizers: A survey for leveraging large language models to construct automated planning models. In *Findings of the Association for Computational Linguistics: ACL 2025*, 25167–25188.

Vallati, M.; Barták, R.; Chrpa, L.; McCluskey, T. L.; and Petrick, R. P. 2025. Knowledge engineering for planning and scheduling in the LLM era. In *Proceedings of the International Conference on Automated Planning and Scheduling*, volume 35, 391–395.

Wójtowicz, D.; Puszyński, K.; and Gałuszka, A. 2024. PDDL optimal planning as the Linear Programming Problem in a seaport container terminal. In *2024 28th International Conference on Methods and Models in Automation and Robotics (MMAR)*, 76–81. IEEE.

Yildiz, H.; Yoon, J.; Talluri, S.; and Ho, W. 2016. Reliable supply chain network design. *Decision sciences*, 47(4): 661–698.