Semester 1 – Final Report

Project Name	Reinforcement Learning on Sales & Distributions
Faculty Supervisor Name	Mr. Samir Mammadov
Team Lead Name & ID	Laman Panakhova 16882 Contribution: 1, 4, 9
Team member #1 Name & ID What sections did this person contribute to?	Aysu Azammadova 16113 Contribution: 2, 7, 9
Team member #2 Name & ID What sections did this person contribute to?	Nargiz Aghayeva 16042 Contribution: 3, 6, 9
Team member #3 Name & ID What sections did this person contribute to?	Leyla Eynullazada 18033 Contribution: 5, 8, 9

Word count guide: ~3000

1. Executive Summary

Overview of project goals and motivation

The primary goal of this project is to explore the application of Reinforcement Learning (RL) in dynamic sales and distribution scenarios within supply chain management (SCM). Traditional SCM methods struggle with unpredictable demand, stockouts, and overstock situations, and RL offers a data-driven adaptive solution through sequential decision-making modeled as a Markov Decision Process (MDP). The system developed in Semester 1 focuses on a simulation environment for three products (Chocolate, Milk, Sugar) and three clients (A, B, C) (for simplicity) to demonstrate RL dynamics for stock management, pricing, product recommendation, and supply allocation. After the MDP formulation is finalized the model will be trained on extensive full dataset.

Beyond operational efficiency, this project is motivated by the growing importance of intelligent decision-making systems in modern supply chains. With increasing market volatility, shorter product life cycles, and heightened customer expectations, supply chain decisions must be made continuously and under uncertainty. Reinforcement Learning is particularly well-suited to such environments, as it allows an agent to learn from interaction rather than relying solely on predefined rules or static forecasts. By framing sales and distribution decisions as a sequential learning problem, the project aims to demonstrate how adaptive policies can emerge from experience, even when demand patterns and system dynamics are partially stochastic.

From an academic perspective, the project emphasizes understanding and transparency over pure performance optimization. Rather than treating reinforcement learning as a black-box solution, the system is intentionally designed to expose the relationship between state representation,

reward design, and agent behavior. This makes the project suitable not only for experimentation but also for analysis and explanation, which are essential in an educational and research-oriented setting. The simplified scale of the simulation allows controlled experimentation while still capturing core challenges present in real-world supply chain management.

Additionally, the project aims to bridge theoretical reinforcement learning concepts with practical SCM use cases. Concepts such as delayed rewards, trade-offs between short-term profit and long-term service level, and exploration versus exploitation are explicitly reflected in the modeled environment. By doing so, the project serves as a structured case study demonstrating how reinforcement learning techniques discussed in academic literature can be operationalized within a supply chain context. This alignment between theory and practice forms the central motivation of the work and guides the design decisions made throughout Semester 1.

Summary of Semester 1 achievements

During Semester 1, the team successfully:

- Conducted a detailed literature review on RL applications in SCM, identifying gaps in real-world applicability (Rolf et al., 2023; Deng et al., 2025).
- Defined the problem space, research questions, and MDP representation of the supply chain environment.
- Developed Python simulation code for RL experiments, including state, action, reward, and transition definitions.
- Designed journey maps and data models to support experimentation, including MongoDB-based collections for experiments, actions, states, and rewards.
- Prepared preliminary evaluation metrics for algorithm performance, focusing on inventory, stockout rates, and cost optimization.

In addition to the core technical outcomes, Semester 1 also focused on ensuring that the project scope remained realistic and achievable within the academic timeline. Several iterations of problem refinement were carried out to avoid overgeneralizing the supply chain problem. Initially, the project aimed to address pricing, distribution, and recommendation simultaneously; however, based on feasibility analysis and supervisor feedback, the team decided to narrow the focus to inventory management and distribution optimization using reinforcement learning. This decision helped reduce implementation risk and allowed the team to design a more controlled and explainable simulation environment. As a result, the Semester 1 work emphasizes correctness, transparency, and reproducibility rather than overly complex modeling.

Preview of Semester 2 development plan

Semester 2 will build on this foundation by:

- Implementing Deep Q-Networks (DQN) with hyperparameter tuning.

- Conducting systematic experiments across multiple RL configurations and scenarios.
- Extending performance metrics to include comparative benchmarks against baseline heuristics.
- Validating the simulation results and preparing detailed dashboards for KPIs.

2. Team Structure and Role Allocation

Team member profiles / specializations

- *Laman Panakhova (Team Lead, 16882):* Project coordination, Literature review, overall architecture design, RL algorithm guidance.
- *Aysu Azammadova (16113):* Risk assessment, ethical considerations, validation strategy, and simulation testing.
- *Nargiz Aghayeva (16042):* Work decomposition, tool selection, database design, and data pipeline management.
- *Leyla Eynullazada (18033):* Team coordination support, documentation, semester planning, and final report integration.

Roles and responsibilities during Semester 1

- Literature review, MDP formulation, and preliminary Python environment: Laman & Nargiz
- User journey mapping and database design (MongoDB simulation logs): Aysu
- Documentation, report drafting, and diagram creation (MDP, RL training loop, data pipelines): Leyla

Proposed role allocation for Semester 2

- *Laman:* Lead DQN implementation, oversee experiments, guide RL hyperparameter optimization.
- *Nargiz:* Execute testing and evaluation protocols, maintain simulation integrity, analyze results.
- *Aysu:* Database management, implement dashboards, handle data preprocessing and logging.
- *Leyla:* Compile results, coordinate report drafting, manage Gantt charts and milestone tracking.

Rationale for role assignments

Roles were assigned based on previous experience, familiarity with tools, and contributions in Semester 1, ensuring continuity and minimizing bottlenecks. The role allocation was also influenced by workload balance and continuity between semesters. During Semester 1, tasks were intentionally distributed to allow each team member to gain exposure to both technical and documentation-related responsibilities. This approach

reduced the risk of knowledge silos, where only one member understands a critical system component. For Semester 2, while roles are more specialized, cross-review responsibilities will remain in place. For example, implementation decisions made by the reinforcement learning developer will be reviewed by the documentation and evaluation lead to ensure consistency with the research objectives. This structure supports collaboration while maintaining accountability.

3. Work Breakdown

Major tasks identified from use cases

- Stockout/Overstock Management
- Dynamic Competitive Pricing
- New Product Recommendation/Promotion
- Supply Allocation under Constraints

Sub-task decomposition

- *Stockout/Overstock Management:* Define state vectors (inventory, backlog, demand history), define reward structure, implement step function in Python, run simulations, log KPIs.
- *Dynamic Pricing:* Incorporate competitor prices, demand fluctuations, pricing actions, reward as profit and market share.
- *Product Recommendation:* Model client purchase behavior, action as recommendation selection, reward based on conversions.
- *Supply Allocation:* Determine allocation vectors, define constraints, reward based on service levels and revenue.

Each major task identified in the work breakdown was directly mapped to a corresponding reinforcement learning component to ensure traceability between system requirements and implementation. For example, the use case “Optimize Inventory Levels” was decomposed into defining the state space (inventory, backlog, time features), action space (ordering and distribution decisions), and reward function (profit-based objective). Similarly, the use case “Handle Demand Uncertainty” led to the inclusion of stochastic demand generation and moving average demand history within the environment. This mapping ensured that every technical task had a clear purpose and directly contributed to the learning objective of the agent.

Task dependency discussion

- RL algorithm design depends on accurate state, action, and reward definitions.
- Data preprocessing must precede agent training and simulation.
- Dashboard visualization requires logs from all experiment runs.
- Evaluation and analysis are dependent on successful experiment execution.

Task dependencies were carefully managed to avoid rework during later stages. Environment design was treated as a prerequisite for all learning-related tasks, since changes in state representation or reward structure would directly affect training stability. Similarly, data logging and experiment tracking were scheduled before large-scale training to ensure that results could be analyzed retrospectively. By explicitly identifying these dependencies early, the team minimized the risk of invalid experimental results caused by late-stage architectural changes.

4. Milestone Development and Timeline

For the purposes of milestone planning, you all work must be completed by **April 30, 2026**

Semester 2 planned milestones – including testing (see below)

The April 30, 2026 deadline was used as a hard constraint when designing the milestone timeline. All milestones were planned with buffer periods to accommodate debugging, re-training models, and revising documentation based on testing outcomes. This approach reflects the uncertainty inherent in reinforcement learning projects, where convergence and performance cannot always be guaranteed within a fixed number of training iterations. Therefore, milestones prioritize functional correctness and interpretability over marginal performance improvements.

Semester 2 Planned Milestones (to April 30, 2026):

Milestone	Description	Deadline	Success Criteria	Responsible
M1	DQN Implementation & Environment Integration	Feb 15	Environment correctly updates states and rewards; initial runs stable	Laman & Nargiz
M2	Hyperparameter Tuning	Mar 1	Learning convergence achieved; reward variance minimized	Laman
M3	Extended Experiments (Use Cases 2-4)	Mar 20	All use cases logged; KPIs collected	Aysu & Nargiz
M4	Dashboard & Visualization	Apr 5	Dashboards display service levels, stockouts, revenue metrics	Nargiz & Leyla
M5	Report Draft & Peer Review	Apr 15	Report complete, diagrams included	Leyla
M6	Final Testing & Validation	Apr 25	All tests executed, results reproducible	Aysu & Laman

Gantt chart or timeline projection

Timeline Projection: A Gantt chart with weeks on the X-axis and tasks on the Y-axis; milestones highlighted in color.

Task / Milestone	Feb (W1–W2)	Feb (W3–W4)	Mar (W1–W2)	Mar (W3–W4)	Apr (W1)	Apr (W2)	Apr (W3)	Apr (W4)	Responsible
M1: DQN Implementation & Environment Integration	■■■■■■	■■■■■■							Laman, Nargiz
M2: Hyperparameter Tuning		■■■■■■	■■■■■■						Laman
M3: Extended Experiments (Use Cases 2–4)			■■■■■■	■■■■■■					Aysu, Nargiz
M4: Dashboard & Visualization				■■■■■■	■■■■■■				Nargiz
M5: Report Draft & Peer Review					■■■■■■	■■■■■■			Leyla
M6: Final Testing & Validation						■■■■■■	■■■■■■		Aysu, Laman
Final Submission & Presentation Prep							■■■■■■	■■■■■■	All

The Gantt chart illustrates parallel development paths where possible, particularly between environment refinement, evaluation metric design, and documentation updates. For example, while reinforcement learning agents are being trained, the evaluation framework and visualization scripts can be developed independently. This parallelization is intended to reduce idle time and allow earlier identification of performance issues. Dependencies are explicitly marked to ensure that critical tasks, such as final system testing, are not initiated before all prerequisite components are stable.

Success criteria for each milestone

Progress is considered successful when the reinforcement learning agent can interact with the simulation environment reliably, with state transitions and reward calculations behaving as expected and initial training runs completing without runtime errors. Learning stability is demonstrated when tuning efforts lead to reduced reward variance and consistent training behavior across multiple runs.

Experimental success is achieved when all defined use cases execute fully, simulation data are logged correctly, and key performance indicators such as service level, stockouts, and revenue are available for analysis. Visualization and dashboard components are considered complete when they accurately and clearly represent these metrics and support comparison between different policies.

The reporting phase is successful when a complete and well-structured draft is produced, including diagrams and explanations, and feedback is incorporated effectively. Final validation is achieved when results are reproducible under different simulated scenarios and the overall system demonstrates stable and reliable behavior.

5. Risk Assessment and Mitigation Planning

Risk assessment and mitigation for Semester 2

Identified Risks:

1. *Data Inconsistency:* Simulated data may not reflect realistic demand patterns.

   Mitigation: Introduce random noise, calibrate against historical averages.

2. *Algorithm Convergence Issues:* DQN may fail to stabilize.

   Mitigation: Use target networks, experience replay, and stepwise learning rate tuning.

3. *High Computational Load:* Simulations may run slowly.

   Mitigation: Use smaller batch experiments initially, optimize code vectorization.

4. *Software/Library Bugs:* Errors in Python environment or MongoDB logging.

   Mitigation: Unit testing each module, version control, code review.

A key technical risk specific to reinforcement learning is training instability, where the agent may fail to converge or exhibit highly variable performance across runs. This risk is mitigated by starting with simple baseline policies and gradually increasing model complexity. Additionally, fixed random seeds and controlled demand distributions will be used during early testing to isolate implementation errors from stochastic effects. If instability persists, the team will prioritize demonstrating a correct environment and reward formulation overachieving optimal performance.

Contingency planning

- Maintain backup of all simulation logs.
- Prepare fallback baseline heuristics for comparison if RL fails.
- Parallelize experiments to ensure timely completion.

Beyond technical risks, academic workload and scheduling conflicts pose a potential challenge, particularly during peak assessment periods. To mitigate this, internal deadlines will be set earlier than official milestones, and progress will be tracked through weekly check-ins. In case a team member becomes unavailable for a period of time, task ownership documentation will allow another member to temporarily assume responsibility without significant disruption.

6. Final / Confirmed Selection of Tools, Technologies, and Resource Planning

Software, hardware, and platforms needed/planned

Software & Platforms:

- Python 3.10 for simulations

- NumPy, Pandas for data handling
- PyTorch for DQN implementation
- Gymnasium for simulation environment
- MongoDB for logging experiment data
- Figma for journey maps
- Jupyter Notebook for experiment orchestration

Hardware:

- Standard laptop for initial experiments
- Optional cloud GPU (Colab Pro or local GPU server) for DQN training

Licensing, budget constraints, and procurement plan

- Open-source software used; no additional costs
- Cloud computing costs estimated at $20-50 for GPU time (if needed)

Infrastructure setup timeline

- Python, libraries, and MongoDB: Completed Semester 1
- Cloud/GPU setup: First week of Semester 2

The selected tools were chosen primarily for transparency and educational value rather than cutting-edge performance. Python and standard reinforcement learning libraries allow the implementation to remain readable and modifiable, which is important for assessment and future extension. Cloud-based platforms such as Google Colab reduce hardware dependency and ensure that all team members can reproduce experiments under identical conditions. This choice also aligns with budget constraints, as no paid licenses or proprietary platforms are required.

7. Evaluation and Validation Strategy

Testing approaches for Semester 2:

- *Unit Testing:* Functions like step(), reward calculation, and pipeline updates
- *Integration Testing:* RL agent integrated with simulation and database logging
- *System Testing:* Complete use case runs over multiple weeks of simulated data
- *Performance Testing:* Track computation time per step and memory usage
- *User Acceptance Testing:* Review by faculty/researcher team
- *Security Testing:* Ensure no sensitive data exposure in logs

Testing in reinforcement learning systems differs from traditional software testing because correct behavior is often probabilistic rather than deterministic. Therefore, evaluation will focus not only on correctness of code execution but also on consistency of learning trends across multiple runs. Performance testing will analyze reward stability and variance over time, while system testing will ensure that the agent's actions remain within valid operational limits (e.g., no

negative inventory). These evaluation strategies ensure that observed improvements are meaningful and not due to random fluctuations.

Metrics for evaluating project success

- Service Level: % of demand met
- Stockout Reduction: # of stockouts and backlog
- Cost Metrics: Total ordering, holding, and stockout costs
- RL Training Metrics: Reward convergence, stability, and computational efficiency

In addition to numerical performance metrics, interpretability will be considered a success criterion. This includes analyzing action distributions, inventory trajectories, and reward component breakdowns. Such analysis allows the team to explain *why* the agent behaves in a certain way, which is especially important for academic evaluation and ethical considerations. A slightly lower-performing but interpretable policy may be preferred over a marginally better but opaque solution.

User testing plan (if applicable)

- Conduct scenario simulations for all four use cases
- Compare RL outcomes to baseline heuristic policies
- Collect logs and analyze KPI improvements

8. Ethical, Legal, and Professional Considerations

Data protection and privacy concerns

Simulated data does not contain real customer data; if real datasets are introduced, GDPR compliance will be ensured.

Ethical implications of design decisions

Avoid RL policies that prioritize profits at the cost of sustainability; reward functions include service level and fairness metrics.

Compliance and standards

Code follows clean coding practices (Martin, 2008); proper documentation and version control maintained.

Although the project uses simulated data, the design choices reflect real-world supply chain scenarios. Poorly designed reward functions in real systems could incentivize unethical behavior, such as intentionally undersupplying certain clients to maximize short-term profit. By explicitly penalizing stockouts and unmet demand, the project emphasizes fairness and service quality

alongside profitability. This design choice reflects responsible engineering practice, even within a simulated academic context.

9. Conclusion and Forward Strategy

Summary of Semester 1 outcomes

- Literature review completed
- MDP environment defined
- Python simulation developed and tested for core RL logic
- Database and journey maps designed
- Preliminary metrics established

Summary Semester 2 action plan

- DQN implementation and training
- Hyperparameter optimization
- Extended experiments across use cases
- Dashboard creation for performance monitoring
- Complete report drafting and final presentation

Long-term vision beyond the capstone

- Extend RL platform to larger product sets and multiple clients
- Explore hybrid RL methods for improved convergence
- Publish findings on practical RL in SCM research
- Potential deployment for local warehouses or SMEs for sales optimization

Semester 1 primarily served as a foundation-building phase, where emphasis was placed on understanding the problem space, designing a valid reinforcement learning environment, and aligning technical decisions with academic expectations. While no production-level system has been developed yet, the outcomes of this semester significantly reduce uncertainty for Semester 2. The project is now positioned to focus on refinement, evaluation, and clear demonstration of learning outcomes rather than exploratory development.

References:

Martin, R. C. (2008). *Clean code: A handbook of agile software craftsmanship*. Prentice Hall.

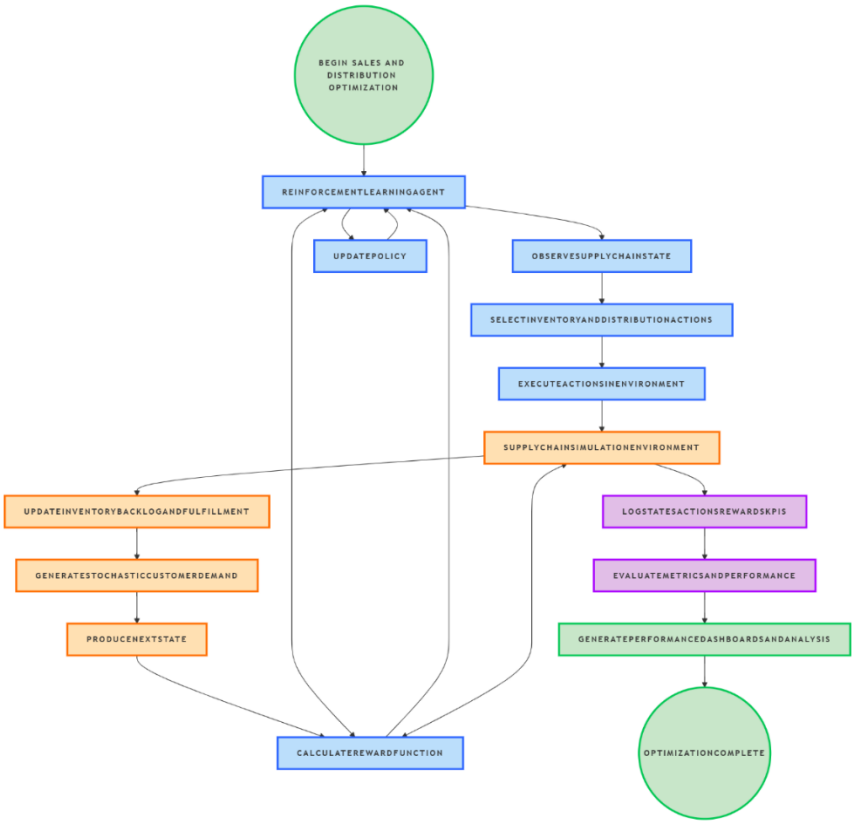
Deng, Y., Zhang, H., Li, X., & Shen, W. (2025). Reinforcement learning for supply chain optimization: A comprehensive survey of models, methods, and applications. *International Journal of Production Economics, 259*, 108812. https://doi.org/10.1016/j.ijpe.2024.108812

Rolf, E., Prokhorenkova, L., & Paninski, L. (2023). Decision-focused learning and reinforcement learning for operations and supply chain management. *Manufacturing & Service Operations Management, 25*(4), 1352–1370. https://doi.org/10.1287/msom.2022.1145

Appendices

- A. Your Research Report
- B. Your User Design
- C. Any Supplementary diagrams
- D. Preliminary code or prototypes

Overall Logic of the Project:



Initial Simulative Examples:

```
Starting simulation with LEAD_TIME = 1 and seed = 42

--------------------------------------------------------------------------------------------------------------
STEP 1  |  WoY used for demand sampling: 48
--------------------------------------------------------------------------------------------------------------
Orders placed (total units per item):
  Choc  : 0.0 ADT
  Milk  : 1300.0 ADT
  Sugar : 0.0 ADT

Pipeline state (oldest -> newest) and arrivals this step:
  Choc  : arrivals=0.0  |  pipeline=[np.float64(0.0)]
  Milk  : arrivals=0.0  |  pipeline=[np.float64(1300.0)]
  Sugar : arrivals=0.0  |  pipeline=[np.float64(0.0)]

Inventory at end of step (Inv_wh after shipments and arrivals):
  Choc  : 4200.0 ADT
  Milk  : 70.0 ADT
  Sugar : 2730.0 ADT

Backlog at end of step (per Item-Client):
  Choc  : A:126.5  B:0.0  C:80.4
  Milk  : A:0.0  B:0.0  C:0.0
  Sugar : A:13.7  B:10.4  C:0.0

Detailed per Item-Client (ShipReq, ShipAct, Demand, Sales, Shortage):
Item   Client  ShipReq  ShipAct     Demand      Sales   Shortage
Choc   A         350.0    350.0      526.5      400.0      126.5
Choc   B         300.0    300.0      411.3      411.3        0.0
Choc   C         150.0    150.0      230.4      150.0       80.4
Milk   A         400.0    400.0      491.4      491.4        0.0
Milk   B         380.0    380.0      366.7      366.7        0.0
Milk   C         350.0    350.0      337.7      337.7        0.0
Sugar  A         100.0    100.0      123.7      110.0       13.7
Sugar  B          90.0     90.0      100.4       90.0       10.4
Sugar  C          80.0     80.0       74.4       74.4        0.0

Financial & objective breakdown this step:
  Revenue:        16542.19
  Ordering Cost:  1365.00
  Holding Cost:   350.00
  Stockout Cost:  1154.80
  Reward (Profit):13672.40
--------------------------------------------------------------------------------------------------------------


--------------------------------------------------------------------------------------------------------------
STEP 2  |  WoY used for demand sampling: 49
--------------------------------------------------------------------------------------------------------------
Orders placed (total units per item):
  Choc  : 0.0 ADT
  Milk  : 2430.0 ADT
  Sugar : 0.0 ADT

Pipeline state (oldest -> newest) and arrivals this step:
  Choc  : arrivals=0.0  |  pipeline=[np.float64(0.0)]
  Milk  : arrivals=1300.0  |  pipeline=[np.float64(2430.000003)]
  Sugar : arrivals=0.0  |  pipeline=[np.float64(0.0)]

Inventory at end of step (Inv_wh after shipments and arrivals):
  Choc  : 3383.9 ADT
  Milk  : 1300.0 ADT
  Sugar : 2459.6 ADT

Backlog at end of step (per Item-Client):
  Choc  : A:55.9  B:94.2  C:0.0
  Milk  : A:398.8  B:246.7  C:237.1
  Sugar : A:0.0  B:0.0  C:3.7

Detailed per Item-Client (ShipReq, ShipAct, Demand, Sales, Shortage):
Item   Client  ShipReq  ShipAct     Demand      Sales   Shortage
Choc   A         355.0    355.0      537.4      481.5       55.9
Choc   B         311.1    311.1      405.3      311.1       94.2
Choc   C         150.0    150.0      195.3      195.3        0.0
Milk   A          25.2     25.2      424.0       25.2      398.8
Milk   B          23.3     23.3      270.0       23.3      246.7
Milk   C          21.5     21.5      258.5       21.5      237.1
Sugar  A         101.0    101.0       92.5       92.5        0.0
Sugar  B          90.0     90.0       76.3       76.3        0.0
Sugar  C          79.4     79.4       83.2       79.4        3.7

Financial & objective breakdown this step:
  Revenue:        15222.40
  Ordering Cost:  2551.50
  Holding Cost:   357.17
  Stockout Cost:  5181.83
  Reward (Profit):7131.89
--------------------------------------------------------------------------------------------------------------
```

```
--------------------------------------------------------------------------------------------------------------
STEP 3  |  WoY used for demand sampling: 50
--------------------------------------------------------------------------------------------------------------
Orders placed (total units per item):
  Choc  : 616.1 ADT
  Milk  : 1200.0 ADT
  Sugar : 0.0 ADT

Pipeline state (oldest -> newest) and arrivals this step:
  Choc  : arrivals=0.0  |  pipeline=[np.float64(616.1289411026219)]
  Milk  : arrivals=2430.0  |  pipeline=[np.float64(1200.0)]
  Sugar : arrivals=0.0  |  pipeline=[np.float64(0.0)]

Inventory at end of step (Inv_wh after shipments and arrivals):
  Choc  : 2550.6 ADT
  Milk  : 2700.1 ADT
  Sugar : 2191.3 ADT

Backlog at end of step (per Item-Client):
  Choc  : A:21.0  B:0.0  C:109.4
  Milk  : A:0.0  B:0.0  C:0.0
  Sugar : A:0.0  B:1.5  C:0.0

Detailed per Item-Client (ShipReq, ShipAct, Demand, Sales, Shortage):
Item   Client  ShipReq  ShipAct     Demand      Sales   Shortage
Choc   A         367.7    367.7      444.6      423.6       21.0
Choc   B         311.1    311.1      343.3      343.3        0.0
Choc   C         154.5    154.5      263.9      154.5      109.4
Milk   A         370.7    370.7      358.2      358.2        0.0
Milk   B         343.1    343.1      346.6      346.6        0.0
Milk   C         316.0    316.0      248.5      248.5        0.0
Sugar  A         100.1    100.1       92.0       92.0        0.0
Sugar  B          88.6     88.6       90.1       88.6        1.5
Sugar  C          79.4     79.4       65.7       65.7        0.0

Financial & objective breakdown this step:
  Revenue:        15546.95
  Ordering Cost:  7421.29
  Holding Cost:   372.10
  Stockout Cost:  659.33
  Reward (Profit):7094.23
--------------------------------------------------------------------------------------------------------------


--------------------------------------------------------------------------------------------------------------
STEP 4  |  WoY used for demand sampling: 51
--------------------------------------------------------------------------------------------------------------
Orders placed (total units per item):
  Choc  : 1449.4 ADT
  Milk  : 0.0 ADT
  Sugar : 0.0 ADT

Pipeline state (oldest -> newest) and arrivals this step:
  Choc  : arrivals=616.1  |  pipeline=[np.float64(1449.4416796091518)]
  Milk  : arrivals=1200.0  |  pipeline=[np.float64(0.0)]
  Sugar : arrivals=0.0  |  pipeline=[np.float64(0.0)]

Inventory at end of step (Inv_wh after shipments and arrivals):
  Choc  : 2324.6 ADT
  Milk  : 2877.8 ADT
  Sugar : 1925.3 ADT

Backlog at end of step (per Item-Client):
  Choc  : A:157.7  B:86.1  C:0.0
  Milk  : A:0.0  B:95.4  C:0.0
  Sugar : A:0.0  B:9.5  C:0.0

Detailed per Item-Client (ShipReq, ShipAct, Demand, Sales, Shortage):
Item   Client  ShipReq  ShipAct     Demand      Sales   Shortage
Choc   A         373.2    373.2      552.0      394.3      157.7
Choc   B         314.3    314.3      400.4      314.3       86.1
Choc   C         154.5    154.5      206.9      206.9        0.0
Milk   A         369.5    369.5      336.1      336.1        0.0
Milk   B         343.5    343.5      438.9      343.5       95.4
Milk   C         309.3    309.3      308.7      308.7        0.0
Sugar  A          99.3     99.3       83.6       83.6        0.0
Sugar  B          88.6     88.6       99.6       90.1        9.5
Sugar  C          78.1     78.1       63.8       63.8        0.0

Financial & objective breakdown this step:
  Revenue:        15499.66
  Ordering Cost:  14494.42
  Holding Cost:   356.39
  Stockout Cost:  1743.56
  Reward (Profit):-1094.71
--------------------------------------------------------------------------------------------------------------
```

```
--------------------------------------------------------------------------------------------------------------
STEP 5  |  WoY used for demand sampling: 52
--------------------------------------------------------------------------------------------------------------
Orders placed (total units per item):
  Choc  : 1675.4 ADT
  Milk  : 0.0 ADT
  Sugar : 74.7 ADT

Pipeline state (oldest -> newest) and arrivals this step:
  Choc  : arrivals=1449.4  |  pipeline=[np.float64(1675.436991242174)]
  Milk  : arrivals=0.0  |  pipeline=[np.float64(0.0)]
  Sugar : arrivals=0.0  |  pipeline=[np.float64(74.6822803741602)]

Inventory at end of step (Inv_wh after shipments and arrivals):
  Choc  : 2924.5 ADT
  Milk  : 1859.0 ADT
  Sugar : 1662.1 ADT

Backlog at end of step (per Item-Client):
  Choc  : A:8.9  B:0.0  C:19.3
  Milk  : A:10.8  B:0.0  C:7.9
  Sugar : A:0.0  B:0.0  C:0.0

Detailed per Item-Client (ShipReq, ShipAct, Demand, Sales, Shortage):
Item   Client  ShipReq  ShipAct     Demand      Sales   Shortage
Choc   A         375.3    375.3      541.9      533.1        8.9
Choc   B         314.3    314.3      310.7      310.7        0.0
Choc   C         159.8    159.8      179.1      159.8       19.3
Milk   A         366.2    366.2      377.0      366.2       10.8
Milk   B         343.5    343.5      381.5      381.5        0.0
Milk   C         309.2    309.2      317.2      309.2        7.9
Sugar  A          97.8     97.8       96.1       96.1        0.0
Sugar  B          88.8     88.8       84.8       84.8        0.0
Sugar  C          76.6     76.6       59.6       59.6        0.0

Financial & objective breakdown this step:
  Revenue:        16927.41
  Ordering Cost:  16814.12
  Holding Cost:   322.28
  Stockout Cost:  234.89
  Reward (Profit):-443.88
--------------------------------------------------------------------------------------------------------------
```

```
 ===== SUMMARY TABLE (compact) =====
 |   Step |   WoY_used |   Revenue |   OrderCost |   HoldCost |   StockoutCost |    Reward |   Inv_End_Choc |   Inv_End_Milk |   Inv_End_Sugar |   Bac
klog_Choc_A |   Backlog_Milk_A |   AvgSales_Choc_A |
|-------:|-----------:|----------:|------------:|-----------:|---------------:|----------:|---------------:|---------------:|----------------:|------
-----------:|-----------------:|------------------:|
|      1 |         48 |   16542.2 |     1365    |    350     |       1154.8   | 13672.4   |        4200    |          70    |         2730    |
126.508    |          0       |          355     |
|      2 |         49 |   15222.4 |     2551.5  |    357.172 |       5181.83  |  7131.89  |        3383.87 |        1300    |         2459.56 |
55.9394    |         398.789  |          367.651 |
|      3 |         50 |   15547   |     7421.29 |    372.099 |        659.332 |  7094.23  |        2550.56 |        2700.08 |         2191.35 |
21.0155    |           0      |          373.245 |
|      4 |         51 |   15499.7 |    14494.4  |    356.385 |       1743.56  | -1094.71  |        2324.56 |        2877.83 |         1925.32 |
157.73     |           0      |          375.346 |
|      5 |         52 |   16927.4 |    16814.1  |    322.283 |        234.886 |  -443.876 |        2924.54 |        1858.97 |         1662.15 |
8.87153 |         10.8122 |          391.119 |
```