arXiv:2611.25542v2 [cs.SE] 23 Nov 2026

# Technology-Enabled Fuel Supply Chain Resilience: An Integrated Information System for Real-Time Monitoring and Decision Support

Mohamed Loutis*
Sorbonne Université, Paris, Île-de-France, France
mohamed.loutis@etu.sorbonne-universite.fr; ORCID 0009-0001-6491-9193

Esma Aïmeur
Université de Montréal, Montreal, Quebec, Canada
esma.aimeur@umontreal.ca; ORCID 0000-0001-7414-5454

**Abstract**

**Purpose** – To evaluate and extend the transactional information system described in an earlier fuel logistics study, focusing on data reliability, monitoring and decision support.

**Design/methodology/approach** – An author-led re-evaluation examined available Delphi/SQL code and tested a reconstructed source-derived baseline (R0) against a revised SQL Server artefact (R1) in twelve scenarios. Synthetic transactions and public U.S. energy series were used for technical assessment; organizational work and supply delays were simulated.

**Findings** – R1 improved integration and reporting accuracy in controlled SQL Server tests and added stock-movement records, alerts and recovery controls. On comparable feeds, integration defects fell from 2.90–3.46 % in the reconstructed R0 to 0.14–0.49 % in R1. For national weekly demand, selected forecasts did not significantly outperform the previous-year mean. Process and disruption simulations suggested benefits for manual work and replenishment delay, but these estimates varied with modelling assumptions and are not field-observed resilience outcomes.

**Research limitations/implications** – R0 combines original T-SQL with reconstructed tables and emulated client rules; it is not a verified historical deployment. Operational benefits depend on an author-specified manual comparator and require field testing.

**Practical implications** – Transaction traceability, early exception detection and recoverable records can support decisions when movement data are entered reliably and acted upon promptly.

**Originality/value** – The study evaluates a source-derived legacy design, implements a revised transactional architecture, and distinguishes experimentally observed information-system behaviour from model-predicted organizational effects.

**Keywords** – Fuel supply chain resilience; Information system evaluation; Design science research; Real-time monitoring; Inventory visibility; Software audit

**Paper type** – Research paper

## 1 Introduction

Fuel distribution depends on timely knowledge of refinery production, depot stocks, replenishment orders, receipts and outbound deliveries. Delays or errors in these records can impair decisions

*Corresponding author.

before a physical shortage occurs. Research associates digital integration and information visibility with supply-chain resilience (Ponomarov and Holcomb, 2009; Brandon-Jones et al., 2014; Jiang et al., 2024), while petroleum monitoring studies mainly address equipment and control systems at individual sites (Rashad et al., 2022; Fang et al., 2025; Waqas and Jamil, 2024). This study focuses instead on the integrity and timeliness of transactional information across a fuel distribution network (Figure 1).

In a conference chapter published in 2026, we described the architecture of a fuel logistics information system and reported simulation-based outcomes (Loutis and Aïmeur, 2026). That account discussed IoT, PLC/FPGA, cloud integration and real-time capabilities. In this follow-up, we distinguish the chapter's architectural scope from functions established by the released Delphi/SQL material. We do not present the earlier simulated outcomes as field measurements or claim an independently verified operational deployment of those integrations.

This paper is an *author-led re-evaluation and extension*, not an independent third-party audit. It compares the available database logic, reconstructed as R0, with a newly implemented R1 artefact under specified test conditions. We ask: RQ1, which information capabilities can be established from the released code? RQ2, how do a revised design and its controls perform in the tested scenarios? RQ3, what contribution to resilience can be inferred from observed information-system behaviour and from explicitly modelled operational processes?

The contributions are a source-level characterization of the released logic, a revised transactional architecture grounded in five design principles, and twelve technical and model-based evaluations. We separate comparable database operations from new R1 capabilities, and measured system outcomes from simulated effects on manual work and replenishment. Detailed defect cases and scenario protocols appear in the appendices.

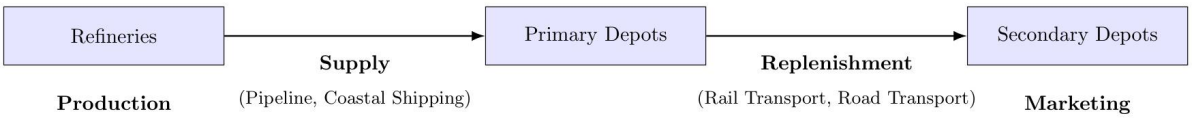


Figure 1. Fuel logistics process: production at refineries, supply of primary depots by pipeline or coastal shipping, replenishment of secondary depots by rail or road, and marketing deliveries to service stations and customers.

Source: Reproduced from the Author's Original Manuscript of Loutis and Aïmeur (2026), Figure 1. https://doi.org/10.1201/9781003647300-28

Section 2 reviews related work and Section 3 the research design. Section 4 presents the artefact and the audit of its implementation, Section 5 the corrections and extensions, Section 6 the evaluation and Section 7 the discussion; Section 8 concludes. Detailed protocols, data dictionaries and additional results are provided in the appendices at the end of this document.

# 2 Literature review

## 2.1 Digital monitoring in petroleum logistics

Digital monitoring of petroleum facilities typically emphasizes PLC/SCADA, IoT sensing and fault detection at terminals or pipelines (Rashad et al., 2022; Fang et al., 2025; Waqas and Jamil, 2024; Ahmed et al., 2023). Related work has explored terminal scheduling and digital twins (Gonçalves and da Cunha, 2021; Mazzuto et al., 2025). These contributions complement, but do not directly evaluate, the consistency and latency of fuel-movement records across depots.

## 2.2 Digital technologies, information integration and supply chain resilience

Visibility, collaboration and information integration are commonly associated with resilience (Christopher and Peck, 2004; Brandon-Jones et al., 2014; Jiang et al., 2024; Wu et al., 2025). Much of this evidence comes from organizational surveys and reviews (Iftikhar et al., 2024; Hosseini Shekarabi et al., 2025; Wei et al., 2026); it does not itself verify the behaviour of a particular logistics database. Our evaluation links these concepts to traceable movements, reporting, alerting and recovery.

## 2.3 Inventory information, forecasting and responsiveness in fuel distribution

Inventory control depends on stock position and replenishment policies (Chopra and Meindl, 2016; Silver et al., 2017). In fuel distribution, responsiveness has been linked to forecast quality (Mardiana, 2023) and to organizational coordination (Manggiasih and Hidayatno, 2023). The present work evaluates forecast accuracy on held-out public series, separately from the effectiveness of information flows in simulated replenishment. MAPE is accompanied by scaled-error metrics because percentage errors can mislead near zero (Hyndman and Koehler, 2006).

## 2.4 Modelling methods

Merise specifies conceptual and relational views of information systems (Avison, 1991); it may be combined with object-oriented design (Arab et al., 2018). Here, entity relationships and business constraints define the model under test, with the detailed data dictionary and logical tables provided in Supplementary Section S1.

## 2.5 Research gap

The research gap is the controlled evaluation of the *transactional information layer* of a multi-site fuel distribution network, rather than a general claim that centralised digitalisation produces resilience. SCADA/PLC primarily concerns physical process monitoring and control, whereas enterprise resource planning (ERP) and supply-chain systems may already provide purchasing, inventory, workflow and reporting functions. The proposed R1 artefact is not claimed to replace or experimentally outperform those systems. Its narrower, testable contribution is the coupling of provenance-aware stock events, database-level validation, exception notifications and operator-reviewed replenishment suggestions with explicit evidence boundaries. Table I positions the research relative to prior studies; Table Ia distinguishes functional scope without presenting a commercial-system benchmark.

Table I. Positioning relative to prior work

| **Strand** | **Representative studies** | **Focus and evidence** | **Boundary** | **Contribution of this study** |
|---|---|---|---|---|
| Control-level monitoring | Rashad et al. (2022); Fang et al. (2025); Waqas and Jamil (2024); Ahmed et al. (2023); Mazzuto et al. (2025) | PLC/SCADA, IoT, FPGA, digital twin; prototypes and laboratory plants | One terminal, pipeline or plant; physical signals | Network-level transactional records; latency and data quality measured |

*Table I (continued)*

| Strand | Representative studies | Focus and evidence | Boundary | Contribution of this study |
|---|---|---|---|---|
| Digital technologies, information integration and resilience | Jiang et al. (2024); Wu et al. (2025); Abourokbah et al. (2026); Iftikhar et al. (2024) | Firm-level capabilities; surveys and reviews | No artefact; no test of information delivered | Mechanisms built into an artefact and tested |
| Fuel supply chain analysis | Mardiana (2023); Manggiasih and Hidayatno (2023); Terab et al. (2023) | System dynamics; soft systems; survey of IT use | Policies and actors; no information system evaluated | Forecasting and information delays evaluated on real and synthetic data |
| Logistics information systems | Biswas and Narahari (2004); Trebuna et al. (2024); Loutis and Aïmeur (2026) | Object models; logistics IS design; artefact description | Design, prototype demonstration or deployment; no evaluation of the information delivered | Audit, correction and experimental evaluation |

Source: Authors' own work

Table Ia. Functional positioning; no head-to-head ERP/SCADA benchmark

| Approach | Primary scope | Relation to the present work | Evidence boundary |
|---|---|---|---|
| PLC/SCADA-oriented monitoring | Field signals and equipment control | Complementary to transactional stock and order records; field integration not evaluated here | Literature-level positioning only |
| ERP and supply-chain suites | Enterprise transactions, inventory and workflows; feature sets vary by product | Potential functional overlap; no claim that R1 implements capabilities unavailable in ERP systems | No commercial product inspected or benchmarked |
| Reconstructed R0 | Legacy Delphi/T-SQL records and reports | Source-derived reference for the shared database operations | Reconstructed schema and emulated forms, not a historical installation |
| Revised R1 | Movement ledger, validation, notifications, access control and order suggestions | Implemented and assessed under specified database tests and process models | No industrial deployment or user study |

Source: Authors' functional classification and the evidence defined in this study; this table is not an independent comparison of commercial products.

# 3 Research design

## 3.1 Design science research approach

The study follows design science research (DSR): diagnose an existing design, define requirements, implement a revised artefact and evaluate its behaviour (Hevner et al., 2004; Peffers et al., 2007). The evaluation is controlled and artificial in the FEDS classification (Venable et al., 2016). An organizational deployment or operator trial has not been performed; the latter is necessary to test the proposed operational benefits.

Three conditions are used. *R0* is a *source-derived comparator*: the released stored procedures, triggers and views are executed as written, but the tables are reconstructed from the available

schema descriptions and the missing Delphi form resources require emulation of client checks. Thus R0 is not the original deployed installation, and R0–R1 differences cannot be interpreted as verified improvements over a historical deployment. For two recovery and load tests a separately identified numbering-controlled R0 arm permits continued operation. *R1* is the revised SQL Server implementation developed for this evaluation. *MANUAL* is an author-specified reference workflow used only in the process and replenishment simulations. SQL Server 2022 Developer (16.0.4325.4) ran on Linux with a 2 GB memory limit and twelve logical CPUs; testing used synthetic transactions generated and submitted by Python programs.

Tests distinguish *inspected* code behaviour, *measured* SQL Server or public-series outcomes, *modelled* process and discrete-event estimates, and functions *not assessed* without users or field data. The principal R1 results were evaluated with synthetic seeds 101–105 where stated, scaled networks and an out-of-sample forecasting interval; T8 one-at-a-time sensitivity checks used only seeds 101–103. The evaluation protocols and numerical summaries are presented in the article and appendices.

### 3.2 Requirements and design principles

Resilience is treated as the capability to prepare for, respond to and recover from disruptions (Ponomarov and Holcomb, 2009). Following the distinction between resource-oriented and action-oriented digital technologies (Wu et al., 2025), the study assumes that an information system contributes to these phases through the information it provides and the actions it enables: visibility of stock and flows, an ingredient of supply chain agility (Christopher and Peck, 2004) that has been found to support resilience (Brandon-Jones et al., 2014), supports readiness; in the study's reasoning, early and correct signals support response, and recoverable records support recovery of the information function itself. Seven requirements follow from these phases and from the capabilities claimed for the system: REQ1 complete and consistent records of every stock movement; REQ2 timely visibility of stock and stock in transit; REQ3 correct reports generated without manual consolidation; REQ4 anticipation of demand and depot autonomy; REQ5 early detection of critical stocks, anomalies and late deliveries; REQ6 few manual interventions and few remaining errors; REQ7 controlled access, recoverability and stable performance as the network grows. Table II states five design principles that answer them, each with its resilience mechanism, its implementation in R1 and its test.

Table II. Design principles of the revised artefact

| **Design principle (requirements)** | **Mechanism** | **Resilience rationale** | **Implementation (Table VII)** | **Tests** |
|---|---|---|---|---|
| DP1 One ledger of physical movements (REQ1–REQ3) | Every stock change is recorded once with its source document; balances, occupancy and reports are derived from it | Readiness: visibility of stock and flows (Christopher and Peck, 2004; Brandon-Jones et al., 2014) and traceability of each change to its document | C2, C3, E1, E2, E3 | T1, T3, T12 |
| DP2 Validate at entry, reconcile at the count (REQ1, REQ6) | Invalid and contradictory entries are rejected; physical events are always recorded and, when inconsistent, flagged; counts are compared with the book | Decisions rest on data that can be trusted; losses and wrong records become visible | C4, C5, E3, E6 | T1, T7, T12 |

*Table II (continued)*

| Design principle (requirements) | Mechanism | Resilience rationale | Implementation (Table VII) | Tests |
|---|---|---|---|---|
| DP3 Push exceptions as events (REQ2, REQ5) | Alerts on the inventory position, abnormal movements, discrepancies and overdue deliveries, notified when the triggering record is committed | Response: shorter information delay between a condition and the decision maker | E4, E5 | T2, T5, T8 |
| DP4 Propose what stored records determine (REQ6) | Orders pre-filled from routes, policy parameters and the inventory position; the operator reviews and confirms | Response: shorter decision delay and fewer manual interventions, without automatic commitment | E8 | T6, T8 |
| DP5 Separate write rights and keep a recoverable history (REQ7) | Writes only through operations granted per role; hashed credentials with lockout; full and frequent transaction-log backups | Recovery: continuity of the information function after failures and misuse | E7 | T9, T11, T12 |

Source: Authors' own work

The five principles operate as a coupled, context-specific design mechanism rather than five independent software features. DP1 and DP2 link a physical movement to a unique source record and distinguish an issued order from a completed receipt; an apparently timely balance is not reliable if that distinction is lost. DP3 and DP4 connect an alert to an operator-reviewed action, but their effectiveness is conditional on when the underlying physical event is entered and when a responsible operator can respond. DP5 preserves the integrity and recoverability of the informational state; it does not restore disrupted physical transport. These constraints yield testable predictions about record correctness (T1, T3), record-to-display delay (T2, T5), and event-to-order delay (T8), while setting a boundary on claims about organizational resilience (T12).

### 3.3 Evaluation framework

Table III defines the twelve scenarios, comparators and measures. Only shared functionality is compared directly between R0 and R1; new R1 capabilities are evaluated as additional coverage. Technical counts and latencies come from controlled software tests. Human actions, entry errors, decision times and disruption outcomes are model estimates relative to the specified MANUAL workflow. The appendices contain scenario definitions, parameters and more detailed outputs.

Table III. Evaluation scenarios, units, arms and measures

| Scenario | Unit and procedure | Arms | Measure | Reported as |
|---|---|---|---|---|
| T1 Data integration | Feed row; six feeds of 14 days with seeded defects of ten types; one year of operation (T1b) | R0 (form, bulk import), R1 (direct, staged) | Loss of valid rows, contamination by defective rows, integration defect rate, balance consistency | Rates with numerators and denominators, per seed and defect rate |
| T2 Monitoring | Committed stock change; paced replay with push and polling dashboards | R1 (R0 has no refresh) | Submission-to-display latency | Median, 95th percentile and maximum |

*Table III (continued)*

| Scenario | Unit and procedure | Arms | Measure | Reported as |
|---|---|---|---|---|
| T3 Reporting | Report field; nine reports against independent reference values | R0 (published, logic only), R1 | Field accuracy; generation time | Correct fields of all fields; maximum time |
| T4 Forecasting | Forecast origin; public weekly data and synthetic depot data, rolling origin | Previous-year aggregate (R0) and nine methods | MAPE, MAE, RMSE, MASE | Test-period error per series and method; paired comparisons on the same forecast origins (block-bootstrap intervals, Diebold–Mariano tests, Holm adjustment) |
| T5 Alerting | Expected alert; same replay as T2 with seeded disruptions | R1 (R0 has no alerts) | Recall, precision, notification latency | Raised, missed and false alerts; median and maximum latency |
| T6 Manual interventions | Business action per operating day; process model | MANUAL, R0, R1 | Actions and time per day | Per day; change relative to MANUAL |
| T7 Manual errors | Seeded-error probe (measured); erroneous value reaching the record (modelled) | MANUAL, R0, R1 | Interception share; remaining errors | Intercepted probes; remaining errors per day with Monte Carlo intervals |
| T8 Supply delay | Replenishment order; discrete-event model | MANUAL, R0, R1 | Position below reorder point to arrival, by component | Mean and components; change relative to MANUAL with range over seeds and sensitivity |
| T9 Security | Access attempt (role × operation); authentication cases | R0 (forms, database), R1 | Unauthorized operations permitted | Counts of permitted and refused attempts |
| T10 Scalability | Request; 1×, 2×, 3×, 5× network, history and users | R0 (logic), R1 | 95th-percentile response time, failures | Per load level |
| T11 Backup and recovery | Restoration trial after simulated media failure | R0 strategy, R1 strategy | Restoration time, data lost | Per trial |
| T12 Resilience | Disruption scenario | MANUAL, R0, R1 | Supply delay, stock-out hours, unmet demand, time to detect | Mean over seeds per scenario |
| Usability | Task attempt by an operator | – | Unaided task completion | Not testable here (no participants) |

Source: Authors' own work

# 4 The artefact and the audit of its implementation

## 4.1 Architecture and roles

The available system uses a Delphi VCL client and SQL Server through ADO components (Embarcadero Technologies, 2025; Microsoft Corporation, 2026); its code includes 46 form units, three stored procedures, two triggers and eleven SQL views. Functions documented in the earlier chapter but not evidenced by the released code are identified separately in Supplementary Table V. Figure 2 presents the legacy application roles and shared server, while Figure 3 presents a schematic network of refineries and multiple depots sharing logistics transaction records.

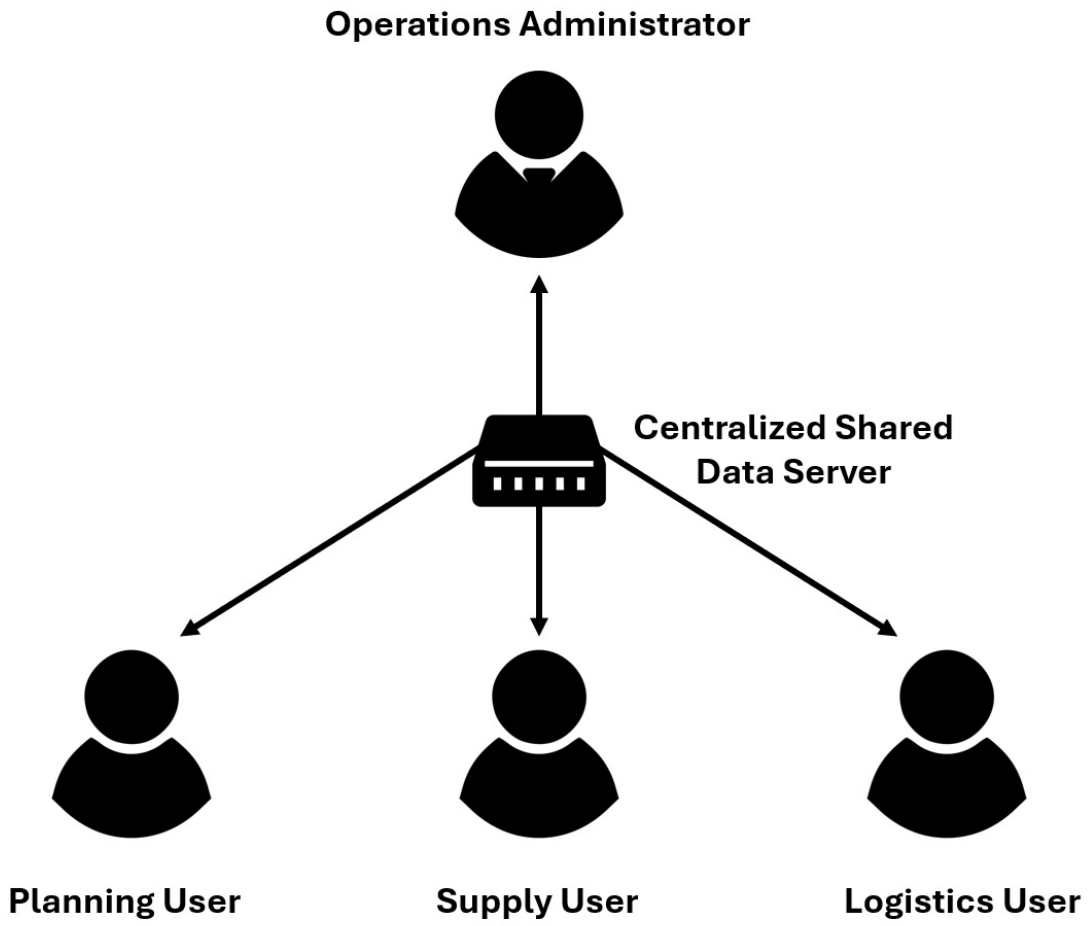


Figure 2. Legacy client–server arrangement with four application roles and a shared database server.

Source: Adapted from the Author's Original Manuscript of Loutis and Aïmeur (2026), Figure 2. https://doi.org/10.1201/9781003647300-28

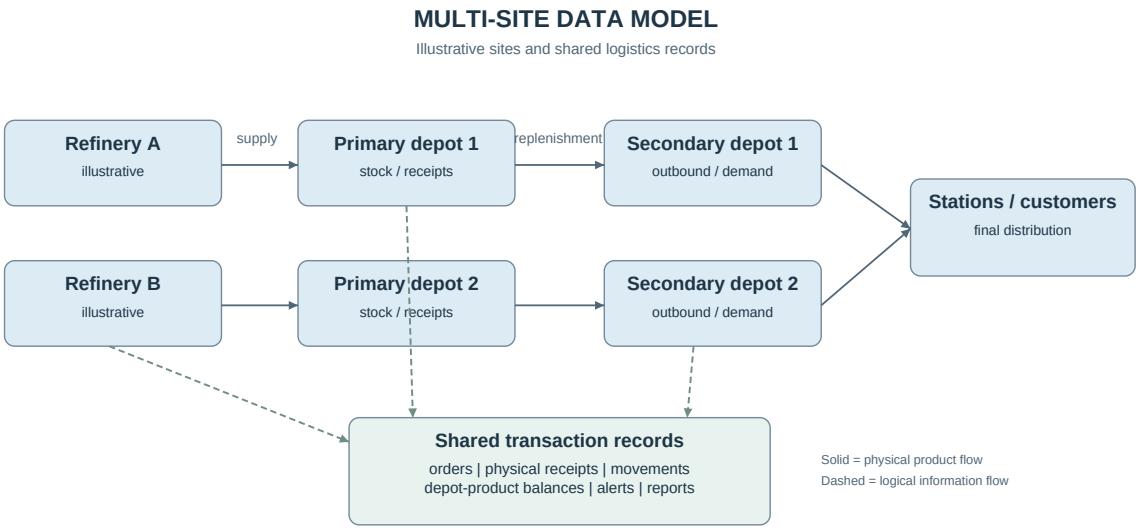


Figure 3. Schematic multi-site fuel distribution network and shared transactional records. Solid arrows indicate possible physical logistics flows; dashed arrows indicate information flows to the shared system.

Source: Authors' own schematic, based on the process model in Loutis and Aïmeur (2026) and the requirements of this study.

## 4.2 Conceptual data model

The Merise conceptual data model (CDM) contains thirteen entities and seventeen associations (Supplementary Table IV); the data dictionary is in Supplementary Tables S1a–S1b. It describes refineries, depots, products, transporters, production records and supply or replenishment orders. Figure 4 presents an EER-style representation of the legacy relational model derived from the published conceptual descriptions and Delphi/T-SQL declarations. The diagram identifies the distinct emitting and receiving depot roles of replenishment orders. R1 extends this model with separate records for outbound deliveries, physical receipts and stock counts.

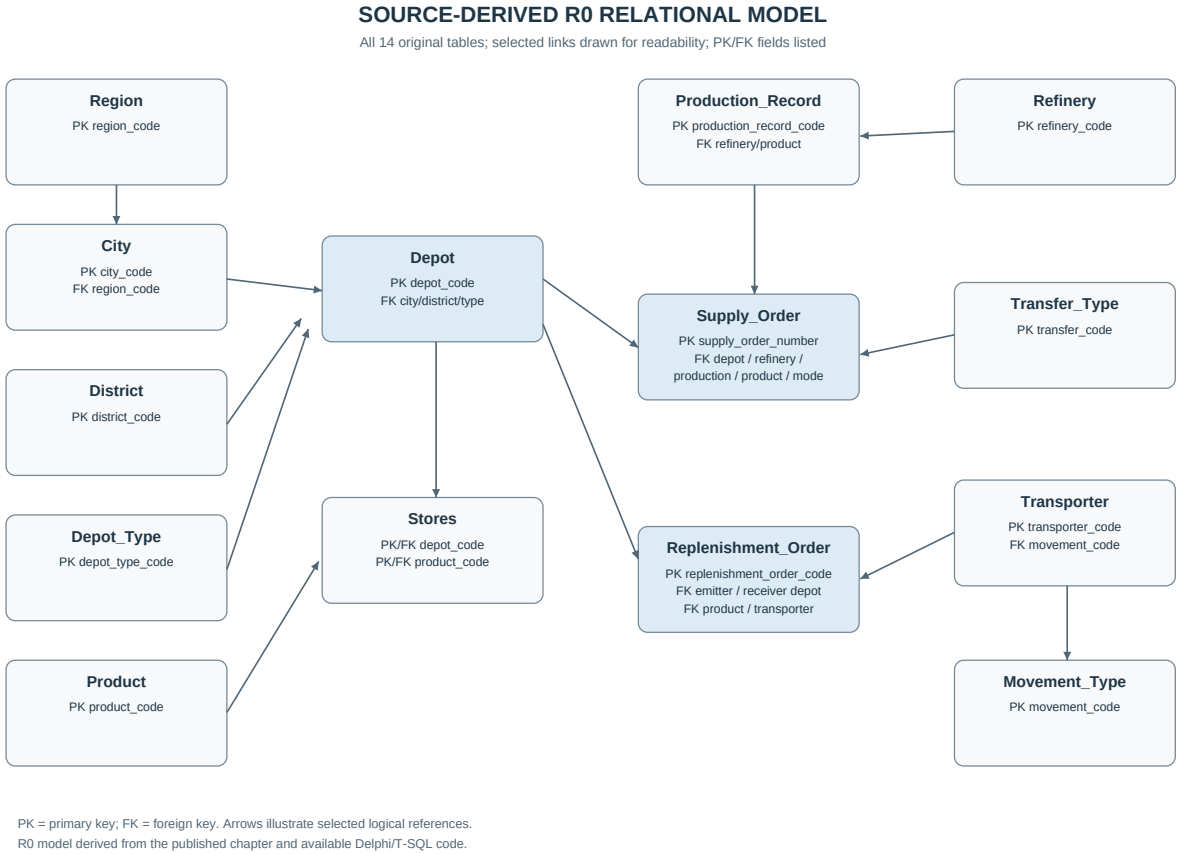


Figure 4. EER-style representation of the source-derived R0 relational model. Lines show selected logical key references and associations; attribute lists are restricted to identifying and linking fields.

Source: Authors' reconstruction of the model in Loutis and Aïmeur (2026); adapted data-model concepts, newly drawn visualisation.

### 4.3 Logical data model and demonstration scenario

The CDM maps to fourteen original relational tables (Supplementary Table S1c). R1 extends the model with stock movements, outbound deliveries, receipt states, counts, alerts, supply routes and staging records (Supplementary Table S1d). A worked synthetic transaction and the calculation rules are documented in S1e and S2. Quantities are expressed in cubic metres.

### 4.4 Audit of the published implementation

The released material was inspected at commit `ac3d536` (14 November 2026). It provides executable T-SQL and Delphi units but not the complete form resources or installation schema needed to reproduce the graphical application. The available SQL logic can therefore be tested only with a reconstructed data schema, while UI checks must be inferred from source. As authors of the 2026 chapter, we clarify that its descriptions of IoT, PLC/FPGA, cloud, alerts and forecasting were broader than the capabilities evidenced by this released implementation; no present experiment relies on an operational field-device or cloud installation. The comparison below concerns the material available for this study, not an independently verified historical deployment (Supplementary Table V).

Key observed defects include document numbering based on a fixed-width `COUNT(*)+1` suffix, triggers that process only one row of a multi-row insert, and a report that cross-joins order tables before aggregation. A one-year synthetic replay exposed numbering rejection, while focused SQL tests reproduced the trigger and reporting failures (Supplementary Table VI). These findings characterize the released logic in the reconstructed R0 environment and motivate the controls incorporated into R1.

The published design also has strengths that R1 keeps: codes are selected from lists rather than typed, document numbers and total costs are computed, stocks are updated by the database rather than by hand, the supply form checks the remaining production quantity, and order dates cannot be set before the entry date.

## 5 Corrections and extensions: the revised artefact

R1 replaces fragile document numbering, records stock movements in a ledger, separates orders from physical receipts, validates feeds and flags discrepancies. Additional functions comprise stock alerts, update notification, role-based database permissions, backup/recovery controls and operator-reviewed order suggestions (Supplementary Table VII). The five design principles in Table II connect these mechanisms to information visibility, accuracy and response.

R1 was evaluated as a single defined configuration. Database controls, feed staging, an event-driven notification channel and decision-support functions were exercised in twelve scenarios; only shared operations permit like-for-like comparisons with reconstructed R0.

Compatibility-oriented SQL corrections were also exercised on the reconstructed legacy schema in 57 tests, including integrity, reports, regression, migration and backup/restore. This tests database-level corrections, not operation of the original Delphi client; extensions requiring new entities or interfaces were evaluated only as part of R1. Figure 5 depicts the principal R1 data-model extensions. It distinguishes stock movements, outbound deliveries, physical counts, alert rules and alert records as separate entities rather than treating every order as an immediate physical stock change. The relationships are derived from the R1 design specified in Supplementary Table S1d; the diagram is not an independently audited production schema.

*Illustrative operator decision.* Consider the synthetic worked transaction in Supplementary Table

S1e: a primary depot receives 6,000 $m^3$ and dispatches 2,500 $m^3$ to a secondary depot. In R1, the order itself does not establish physical receipt; the corresponding recorded movement changes the stock ledger. Inventory position and days of cover can then be calculated from the current recorded state, and a threshold exception can prompt an operator to review a proposed replenishment order. This sequence explains the decision mechanism without claiming a real-world operator trial.

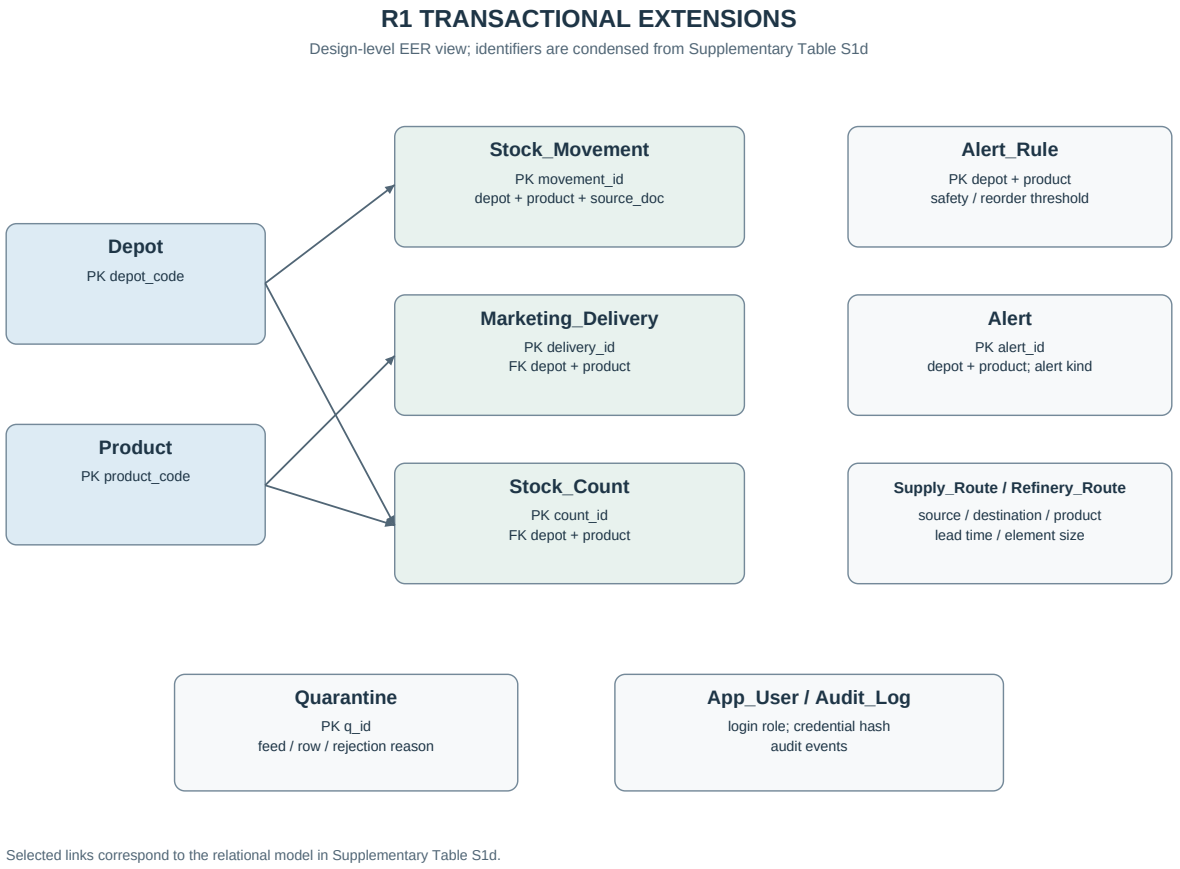


Figure 5. Selected relational extensions of R1, illustrating links between depot-product records, movements, deliveries, counts, alert rules and notifications. Additional columns and constraints are documented in Supplementary Table S1d.

Source: Authors' own work.

# 6 Evaluation results

The following results concern controlled SQL Server tests of the final R1 design, synthetic validation seeds 101–105, and public forecasting observations. Detailed counts and distributions are reported in the supplementary tables. Table XIV summarizes the evidence by method.

## 6.1 Data integration and sustained operation (T1)

Fourteen-day synthetic feeds contained valid rows and seeded defects. On the four feeds represented by both designs, the integration defect rate in R0 was 2.90–3.46 %, versus 0.14–0.49 % with R1 staging. The additional outbound and receipt feeds were assessed only for R1. The staged validator retained unresolved valid rows in quarantine; balance consistency was not a like-for-like outcome because the two systems use different posting rules (Supplementary Table VIII).

In one year of nominal synthetic operation, reconstructed R0 accepted 3,009 of 12,351 production records and orders after the document-number suffix overflowed; R1 accepted all 12,351. The result applies to the tested R0 reconstruction and is not an observation of the historical installation.

## 6.2 Automated reporting (T3)

On the synthetic report inventory, seven of nine assessed legacy report views returned correct fields; the depot status was affected by stock-occupancy arithmetic, and the general supply forecast failed with an SQL integer overflow. For the eight shared report layouts, R0 produced 4,697/4,733 correct fields and R1 produced 4,733/4,733. The ninth report, a general supply forecast, has different layouts: 128 reference fields in R0 (the query failed) and 50 in R1 (all correct). Thus R1 also attained 4,783/4,783 fields across its own nine-report inventory (4,733 shared-layout fields plus 50 forecast fields); R0's corresponding inventory had 4,861 fields (4,733 plus 128). The two total inventories are not field-for-field comparable. R1 generated its reports in at most 37 ms in the nominal test (Supplementary Table S3c).

## 6.3 Monitoring and alerting (T2, T5)

R1 added stock-update notification and alert functions absent from the examined R0 code. These functions were assessed independently of the R0 comparison (Supplementary Table IX).

In the controlled replay, R1 raised all 244 expected alerts at nominal load and all 1,144 at fivefold load, with no false positives in the seeded cases. Median push-display latency remained below 0.01 s, but fivefold-load maximum delays reached 7.327–9.061 s during the early part of the test window; the full-window push P95 was 0.806–1.306 s across runs. These spikes remain unexplained and must not be omitted from responsiveness claims. Additional notification and polling results are in Supplementary Table IX. The tested readout is a database-backed monitoring client rather than the original Delphi GUI: its operational indicators comprise depot-product stock position, stock in transit, days of cover (stock divided by the preceding 28-day mean outbound quantity) and open alert count (Supplementary Section S2.2). The controlled replay verifies record visibility and notification latency; it does not validate dashboard usability, operator interpretation or a field installation. No screenshot of an independently tested graphical interface is provided, and none of the schematic figures should be described as one.

### 6.4 Stock and demand forecasting (T4)

Rolling-origin forecasting used public EIA series, with 2021–2023 for method selection and 2024–2026 for testing (Hyndman and Athanasopoulos, 2021). M0 is the previous-year mean and M1 a one-step naive comparator; the candidate set also includes Holt–Winters (M4), harmonic regression with ARIMA(1,1,1) errors (M5), and a variant of Holt–Winters with the 2020 pandemic observations replaced by the corresponding 2019 observations (M9), as specified in Supplementary Section S3 and Table X. The comparative results appear in Supplementary Table X. The interpretation does not infer a significant gain in national weekly demand forecasting from differences in raw MAPE alone.

Selected methods did not establish an improvement over the previous-year mean for any of the three national weekly demand series after multiple-testing adjustment. They were more accurate on gasoline and distillate stock series. One-week test MAPEs for selected methods were 2.96 %, 5.99 % and 6.84 % on the three demand series, and 0.83 % and 1.59 % on the two stock series. Paired forecast comparisons, uncertainty intervals and depot-level synthetic results are in Supplementary Section S3.

### 6.5 Security, scalability and recovery (T9, T10, T11)

*Access control.* Of the 25 role–operation pairs that the role policy forbids and that the published model can express, seven were reachable from the menus of the corresponding role (for example, the logistics role can open the supply-order form and edit master data), and all 25 succeeded when executed through the single application login that the client code implies. Passwords were stored in clear text, the role was the user name, and 1,000 wrong guesses were rejected without any lockout, after which the correct password was still accepted. In R1, none of 36 forbidden attempts succeeded (none of the 25 that the published model can also express) and all 32 permitted ones did; passwords were stored as salted PBKDF2 hashes (210,000 iterations, an implementation parameter rather than a requirement of NIST guidance; Temoshok et al., 2025), an account was locked after five failures for 15 minutes, and database logins with wrong or empty passwords were rejected. No forbidden operation succeeded in the tested matrix. This is not a penetration test, transport encryption was not tested, and neither version underwent an information security risk assessment following ISO/IEC 27005 (International Organization for Standardization and International Electrotechnical Commission, 2022) or a review against the control catalogue of NIST SP 800-53 (Joint Task Force, 2020); R1's separation of database roles, operation-level grants and hashed credentials with lockout is an instance of the layered protection that the catalogue calls defense in depth (control PL-8(1)), not an assessed implementation of it.

*Scalability.* Supplementary Table XI gives the response times per load level (additional load levels summarized in Supplementary Table S3d). Both versions were measured on SQL Server on the same date, with the same calendar of 686 days. With the published logic, the general supply forecast failed in every request at every level (33 of 33 at the nominal load, 99 of 99 at 5×), always with the arithmetic overflow of Section 6.2, after up to 3.6 s at the nominal load and up to 54.5 s at 3× in T10's concurrent mixed-request workload, which is not directly comparable to T3's isolated report timing; the highest 95th percentile of the other operations was 1.309–1.359 s (the previous-year reports at 5×). In R1, the slowest operation at 5× was the general supply forecast, with a 95th-percentile response time of 0.145–0.178 s across three runs; the tabulated run had a 99th percentile of 0.181 s and a maximum of 0.200 s. Over all 3,509 requests at 5×, the mean was 0.039 s, the median 0.047 s, the 95th percentile 0.095 s, the 99th percentile 0.128 s and the maximum 0.285 s, and no request failed; at the nominal load, R1's slowest operation (transaction entry) had a 95th percentile of 0.050 s. The request mixes have the same shares but not the same content for one class: R1's stock read is its dashboard, which also counts open alerts and

stock in transit, whereas R0 reads the stock table; reports, transaction entry and the general supply forecast are common operations. Without R1's indexes, the 95th percentile of the slowest operation at 5× was 0.160 s (the general supply forecast; 3,494 requests, no failure): at this size, the indexes changed little, and the forecast's gain comes from removing the cross join.

*Recovery.* Under three simulated media failures, full and 60-s transaction-log backups in R1 reduced the recovery-point interval compared with a once-daily full-backup strategy for the reconstructed R0. R1 restored in 0.84–0.97 s and lost 58–106 recently committed rows; the full-backup strategy lost all 40 documents entered since its backup. These small-database tests establish recovery-point behaviour, not operational recovery at scale (Supplementary Table S3g).

### 6.6 Manual interventions and errors (T6, T7)

*Seeded errors (measured).* R1 intercepted 24 of 30 designed error probes, versus 7 of 30 in the reconstructed R0; on the 27 error types supported by both, the counts were 21 versus 7. Neither detected erroneous values consistent with all other records (Supplementary Table XII).

*Process outcomes (modelled).* Under the specified synthetic workload and author-defined MANUAL workflow, the estimated business-action count was 1,670.6 per day manually versus 526.0 with R1 (68.5 % fewer). Modelled working-time and error reductions also depended on assumed typing, checking and error rates. These figures are simulation outputs, not reductions observed in an organization; the full task decomposition and uncertainty ranges appear in Supplementary Tables S3a and S3h.

### 6.7 Supply delay and operational resilience (T8, T12)

Supplementary Table XIII decomposes the supply delay. In the manual procedure, a depot-product whose inventory position fell below its reorder point received its replenishment on average 73.7 h later: 24.0 h before the information reached the decision maker, 6.0 h to prepare, approve and transmit the order, 10.8 h waiting for a loading slot and 33.0 h in transit. The published system shortened this to 63.6 h (13.7 % less), mainly by faster order entry; information still arrived through the next morning's counts. R1 with outbound deliveries posted at 20:00 reached 55.3 h (25.0 % less): the alert follows the posting immediately, but an evening alert waits for the next working morning, which moves part of the delay from information to decision. Posting deliveries also at 11:00, the second iteration, gave 46.5 h, a reduction of 36.9 % (36.0–38.3 % over five seeds), and reduced the order delay from the reorder point to the issued order by 56.1 % (29.9 to 13.1 h). An hourly meter feed, which is not implemented, would reach 44.3 h (39.8 %). Transit, which no information system shortens, is 45 % of the manual delay: removing every information, decision and dispatch delay would leave transit alone and bound the reduction at about 55 %, a bound that only continuous recording of outbound flows and around-the-clock response could approach. A one-at-a-time sensitivity analysis on three of the five validation seeds (101–103) gave reductions of 30.6–40.3 % with two postings across the stated preparation, approval, response and report-time assumptions; assuming the manual morning report missed the daily review raised that estimate to 75.8 %. Moving the single evening posting to 16:00 or 22:00 gave 15.7 % or 26.9 %, respectively, versus 25.0 % at 20:00. These sensitivity bounds describe only the three tested seeds and must not be attributed to all five validation seeds. With the same reorder points, the modelled manual procedure produced 423.6 stock-out depot-product hours over the entire 150-day horizon across 96 depot-products (fill rate 99.82 %), compared with 7.6 for reconstructed R0 and none for R1. The 150-day totals include the simulation warm-up period; they are not post-warm-up steady-state estimates.

Across specified demand and transport disruptions, the discrete-event model predicted fewer

stock-outs and less unmet demand with R1 than with the author-specified manual process, under the chosen inventory, reporting and decision policies. For example, a 60 % demand surge raised mean manual stock-out time to 502.6 h in the simulated network while R1 recorded none; such outcomes are conditional on these model settings, not measured organizational resilience. Full scenarios appear in Supplementary Table S3h.

### 6.8 Summary of results

Table XIV separates measured information-system results from modelled operational estimates. The revised system improved record completeness and the tested controls; the extent to which this yields faster replenishment in practice depends on reporting frequency, human response and physical transit. Public-series demand forecast accuracy did not significantly exceed the previous-year-mean benchmark.

Table XIV. Core findings and evidential scope

| **Evaluation** | **Core finding** | **Basis** |
|---|---|---|
| Data integration (T1) | Shared-feed integration defect rate: 2.90–3.46 % (R0) versus 0.14–0.49 % (R1) | Measured; synthetic, reconstructed comparator |
| Reporting (T3) | R1: 4,733/4,733 fields in eight common layouts; 4,783/4,783 in its full nine-report inventory | Measured; synthetic; layouts differ |
| Monitoring and alerts (T2, T5) | All seeded alerts raised; push maxima up to 9.061 s at fivefold load | Measured; synthetic |
| Forecasting (T4) | No established improvement over previous-year mean for national weekly demand | Public-series tests |
| Manual work and errors (T6, T7) | Fewer actions and residual errors predicted under the study's process assumptions | Modelled; author-defined MANUAL |
| Supply delay (T8, T12) | Directional model benefit; magnitude strongly sensitive to reporting schedules | Modelled; scenario-dependent |
| Security, load and recovery (T9–T11) | Tested R1 controls, response times and log-backup recovery | Measured; development environment |

Note: For T3, 4,733 fields belong to the eight directly comparable layouts. The ninth report has 128 fields in R0 and 50 in R1, so the total inventories are 4,861 and 4,783 fields, respectively; total-inventory percentages are not direct field-level comparisons. Source: Authors' own work

## 7 Discussion

### 7.1 Answers to the research questions

*RQ1.* The author-led source review identified a relational core and several implementation weaknesses, including numbering, trigger and reporting defects. Some capabilities described in the earlier chapter could not be established from the released implementation. This finding concerns source evidence and is not an independently verified assessment of its historical installation.

*RQ2.* R1 improved data validation and report correctness and provided separately tested notification, security and recovery functions (Table XIV). Measured results apply to controlled SQL Server experiments and to the reconstructed R0 comparator. Organizational effects remain model estimates under defined scenarios.

*RQ3.* The design can support readiness through reliable stock records, response through timely notifications and decision support, and information recovery through retained logs. These are information-system mechanisms, not direct evidence that a fuel supply chain recovered from an actual disruption.

## 7.2 Contribution

In the design-science contribution framework, this is an *improvement* in the sense of Gregor and Hevner (2013): it does not propose a new general theory of resilience. Its theoretical contribution is a conditional account of how transactional controls may translate into readiness and response at network level (Jiang et al., 2024; Wu et al., 2025), beyond equipment-level monitoring (Rashad et al., 2022; Waqas and Jamil, 2024). The principles identify a chain of mechanisms with identifiable failure boundaries, rather than assuming that digitalization itself causes resilience.

*P1, event provenance and valid state (DP1–DP2):* when receipts, deliveries and corrections are recorded as distinct events linked to source documents, ledger-derived balances and reports should exhibit fewer untraceable discrepancies than overwritten stock totals; this prediction depends on complete physical-event entry and can be checked by reconciled depot-product balances (T1, T3). *P2, recording cadence and actionable signals (DP3–DP4):* rapid notifications should shorten record-to-display delay, but will shorten event-to-order delay only when physical events are entered before the relevant review window and an operator responds; T2/T5 test the first link technically, while T8 tests the conditional second link only in a model. *P3, recoverable information versus physical constraints (DP5):* access controls and log backups can maintain or restore usable records, but cannot remove dispatch and transit time; T9/T11 assess information controls, whereas T12 models the operational consequence. These propositions are analytically derived and partly illustrated, not established as field-level causal effects.

## 7.3 Size and limits of the effects

*Supply delay.* The estimate is particularly sensitive to reporting schedules. The model predicted a 25.0 % reduction with outbound deliveries posted once daily and 36.9 % when posted twice daily, relative to the author-defined manual workflow. One-at-a-time schedule and parameter checks on three validation seeds, rather than the five used for the principal result, yielded estimates from 15.7 % to 75.8 % across different assumptions and scenarios. These are conditional sensitivity results, not a five-seed uncertainty interval or an observed operational benefit. The reported stock-out totals also include the model warm-up period.

*Forecasting.* Selected methods performed well on two stock series, but their improvements for national weekly demand were not established against the previous-year mean after adjustment. Aggregation and series variability materially affected error levels; the full paired comparisons appear in Supplementary Table S3i.

*Manual work and errors.* Modelled reductions reflect assumed entry rates, checking probabilities and process design (Wahi et al., 2008; Panko, 2000). They must be validated with operators; the technical error probes do not cover plausible but internally consistent mistakes.

## 7.4 Implications for practice

Three practical lessons follow. First, sustained-volume tests and set-based transaction checks expose defects that small demonstrations may miss. Second, a coherent movement ledger enables traceable stocks and reports, provided physical events are entered. Third, notification performance alone is insufficient: review schedules, responsibility for open alerts and physical transit determine

when decisions can change outcomes. For example, the T8 discrete-event experiment gave a modelled mean replenishment interval of 73.7 h under MANUAL and 46.5 h under R1 with two daily postings; the difference is conditional on the specified reporting and decision timetable, not evidence of shorter physical transport or a deployment benefit. In practical terms, the system is intended to make a recorded shortage and the corresponding source documents visible before the next decision window, rather than to remove the transport constraint.

### 7.5 Limitations and future research

R0 uses original SQL on reconstructed tables and inferred Delphi checks; it does not reproduce an authenticated historical deployment. Technical tests were performed on SQL Server 2022 Developer under synthetic loads on a memory-limited workstation. The author-defined MANUAL workflow, simulated recording delays and error probabilities underpin the estimated business actions, error reductions and supply delays. A field or operator study has not been conducted. Public weekly U.S. data support only the forecasting comparisons, not the operational resilience claims.

Push-display delays peaked at 9.061 s during the early load window. Future work should evaluate user interaction and performance in operational settings, and extend the system to automatic measurements and multi-product shipments.

## 8 Conclusion

This author-led re-evaluation and extension of an earlier fuel logistics information system presents a revised transaction architecture and twelve evaluations. Controlled SQL Server experiments on synthetic data found improved integrity, correct reports and functioning alerts and recovery controls relative to a *reconstructed* source-derived comparator. The public-series forecasting tests did not establish better accuracy than a previous-year-mean baseline for national demand. Process and disruption models indicate possible benefits for work and replenishment, including estimated reductions in business actions, residual errors and delay, but the magnitudes depend strongly on the defined manual workflow and reporting assumptions. The study therefore supports the value of reliable information systems as potential resilience enablers; whether the predicted organizational benefits occur requires operator and field validation.

## Data, code and supplementary material

**Figure permissions and acknowledgements.** Figures 1 and 2 are from an Author's Original Manuscript of a book chapter published by Routledge/Taylor & Francis in *Moving Towards Sustainable Businesses and Developing More Resilient Economies* on May 13, 2026, available online at `https://doi.org/10.1201/9781003647300-28`. This licence does not allow the use of the Publisher's version/PDF (this is the version of record that is published on the publisher's website) to be posted online. From: *Moving Towards Sustainable Businesses and Developing More Resilient Economies*, 1st edition, edited by Aamir Rashid and Fenio Annansingh-Jamieson, Copyright © 2026 by CRC Press. Reproduced by permission of Taylor & Francis Group.

**Relation to prior publication.** This article extends the authors' 2026 conference chapter (Loutis and Aïmeur, 2026). The earlier chapter presented a broad architectural design and simulation-based performance statements. The present study reports an author-led re-evaluation of released source material and a separately evaluated revised design; neither the prior simulated outcomes nor a historical field deployment are treated as independently reproduced.

**Data availability.** The study used synthetic operational inputs and public weekly U.S. Energy Information Administration series (U.S. Energy Information Administration, 2026, downloaded 13 November 2026). The public EIA series are available from the original provider; the synthetic input specifications, evaluation methods and aggregate results are documented in the article and appendices.

**Code availability.** The project repository is at `https://github.com/mloutis/fuel-information-system`. The link identifies the project and should not be interpreted as an archive of the R1 evaluation harness or its generated run records.

**Supplementary material.** The appendices contain the MCD/MLD and data dictionary (S1), calculation and business rules (S2), evaluation protocols and detailed tables (S3), sensitivity analyses (S4), and reproducibility specifications and boundaries (S5).

# Supplementary Material

The appendices present the conceptual and logical models, experimental protocols and detailed numerical results supporting the main text.

## S1. Entities, data dictionary and relational tables

Table S1a. Entities of the conceptual data model

| Entity | Identifier | Attributes |
|---|---|---|
| Depot | depot_code | depot_name, depot_address, storage_capacity, occupied_volume (derived in R1) |
| City | city_code | city_name |
| District | district_code | district_name |
| Depot_Type | depot_type_code | depot_type_description (tier in R1: primary or secondary) |
| Product | product_code | product_name, product_abbreviation, unit_volume |
| Refinery | refinery_code | refinery_name, refinery_address |
| Region | region_code | region_name |
| Supply_Order | supply_order_number | supplied_quantity, supplied_elements_count, start_date, end_date, notes_supply_order |
| Transfer_Type | transfer_code | transfer_description (supply modes: pipeline, coastal shipping) |
| Production_Record | production_record_code | production_date, quantity_produced, remaining_quantity |
| Replenishment_Order | replenishment_order_code | quantity_delivered, delivered_elements_count, cost_per_element, total_cost, loading_date, unloading_date, notes_replenishment_order |
| Transporter | transporter_code | transporter_name |
| Movement_Type | movement_code | movement_description (replenishment modes: rail, road) |

Source: Adapted from Loutis and Aïmeur (2026), Table 1, *Design and Deployment of a Fuel Logistics Information System*. https://doi.org/10.1201/9781003647300-28. Reproduced by permission of Taylor & Francis Group.

Table S1b. Data dictionary: fields, types and formats

| Code | Description | Type | Length | Observation |
|---|---|---|---|---|
| depot_code | Depot code | AN | 4 | – |
| depot_name | Depot name | AN | 30 | – |
| depot_address | Depot address | AN | 50 | – |
| storage_capacity | Storage capacity | N | – | $m^3$, all products |
| occupied_volume | Occupied volume | N | – | $m^3$; derived in R1 |
| city_code | City code | N | 6 | – |
| city_name | City name | AN | 30 | – |
| district_code | District code | AN | 2 | stored as text in the published client |
| district_name | District name | AN | 30 | – |
| depot_type_code | Depot type code | AN | 5 | – |
| depot_type_description | Depot type description | AN | 30 | – |
| product_code | Product code | AN | 5 | – |
| product_name | Product name | AN | 30 | – |
| product_abbreviation | Product abbreviation | AN | 5 | – |

*Table S1b (continued)*

| Code | Description | Type | Length | Observation |
|---|---|---|---|---|
| unit_volume | Volume per unit of stock | DEC | 10,6 | $m^3$ per unit; integer in the published code (defect D4) |
| refinery_code | Refinery code | AN | 4 | – |
| refinery_name | Refinery name | AN | 30 | – |
| refinery_address | Refinery address | AN | 50 | – |
| region_code | Region code | AN | 4 | – |
| region_name | Region name | AN | 30 | – |
| supply_order_number | Supply order number | AN | 12 | 16 in R1 |
| supplied_quantity | Supplied quantity | N | – | $m^3$ |
| supplied_elements_count | Supplied elements | N | – | batches or parcels |
| start_date, end_date | Supply start and end dates | D | – | yyyy-mm-dd; end $\geq$ start in R1 |
| notes_supply_order | Supply order notes | AN | 50 | – |
| transfer_code | Transfer code | AN | 4 | – |
| transfer_description | Transfer description | AN | 20 | – |
| stock_quantity | Stock quantity | N | – | $m^3$ |
| production_record_code | Production record code | AN | 12 | 16 in R1 |
| production_date | Production date | D | – | yyyy-mm-dd |
| quantity_produced | Quantity produced | N | – | $m^3$ |
| remaining_quantity | Remaining quantity | N | – | $m^3$; $\geq 0$ in R1 |
| replenishment_order_code | Replenishment order code | AN | 12 | 16 in R1 |
| depot_code_emitter, depot_code_receiver | Emitting and receiving depots | AN | 4 | different in R1 |
| quantity_delivered | Quantity delivered | N | – | $m^3$ |
| delivered_elements_count | Delivered elements | N | – | rail cars or road tankers |
| cost_per_element, total_cost | Transport cost per element, total | N | – | currency not specified; total computed |
| loading_date, unloading_date | Loading and unloading dates | D | – | yyyy-mm-dd; unloading $\geq$ loading in R1 |
| notes_replenishment_order | Replenishment order notes | AN | 50 | – |
| transporter_code | Transporter code | AN | 4 | – |
| transporter_name | Transporter name | AN | 15 | – |
| movement_code | Movement code | AN | 4 | – |
| movement_description | Movement description | AN | 10 | – |

AN: alphanumeric; N: integer; DEC: decimal (precision, scale); D: date. Types follow the persistent-field declarations of the Delphi units and the procedure parameters; The field descriptions follow the conceptual model and the available declarations. Source: Authors' own work

Table S1c. Relational tables of the logical data model

| Table | Attributes |
|---|---|
| Region | *region_code, region_name |
| City | *city_code, city_name, **region_code |
| District | *district_code, district_name |
| Depot_Type | *depot_type_code, depot_type_description |
| Depot | *depot_code, depot_name, depot_address, storage_capacity, occupied_volume, **city_code, **district_code, **depot_type_code |
| Product | *product_code, product_name, product_abbreviation, unit_volume |
| Refinery | *refinery_code, refinery_name, refinery_address |
| Transfer_Type | *transfer_code, transfer_description |
| Movement_Type | *movement_code, movement_description |
| Transporter | *transporter_code, transporter_name, **movement_code |

*Table S1c (continued)*

| Table | Attributes |
|---|---|
| Production_Record | *production_record_code, production_date, quantity_produced, remaining_quantity, **refinery_code, **product_code |
| Supply_Order | *supply_order_number, supplied_quantity, supplied_elements_count, start_date, end_date, notes_supply_order, **transfer_code, **refinery_code, **production_record_code, **product_code, **depot_code |
| Replenishment_Order | *replenishment_order_code, quantity_delivered, delivered_elements_count, cost_per_element, total_cost, loading_date, unloading_date, notes_replenishment_order, **depot_code_emitter, **depot_code_receiver, **product_code, **transporter_code |
| Stores | *depot_code (**), *product_code (**), stock_quantity |

Notes: *Primary key (component); **Foreign key. The repository does not publish SQL table-creation statements; the keys and attributes listed here reconstruct the relational model from the source code and conceptual associations. Source: Adapted from Loutis and Aïmeur (2026), Table 3, *Design and Deployment of a Fuel Logistics Information System*. `https://doi.org/10.1201/9781003647300-28`. Reproduced by permission of Taylor & Francis Group.

Table S1d. Tables added in the revised artefact R1

| Table | Attributes | Purpose |
|---|---|---|
| Marketing_Delivery | *delivery_id, **depot_code, **product_code, delivery_date, quantity, source_reference (unique) | Outbound deliveries (E1) |
| Stock_Count | *count_id, **(depot_code, product_code), count_ts, measured_quantity, book_quantity, discrepancy, source_reference | Physical counts and reconciliation (E3) |
| Stock_Movement | *movement_id, depot_code, product_code, movement_date, quantity (signed), movement_kind, source_doc, entered_by, entered_at | Ledger of every stock change (E1) |
| Alert_Rule | *(depot_code, product_code), safety_stock, reorder_point, max_level, max_movement, count_tolerance | Alert thresholds (E4) |
| Alert | *alert_id, alert_ts, alert_kind, depot_code, product_code, observed, threshold, source_doc, acknowledged | Alerts (E4) |
| Supply_Route, Refinery_Route | receiver or depot, product, emitter or refinery, mode, default transporter, lead time, element size, cost per element, order-up-to level | Order suggestions (E8) |
| Quarantine | *q_id, batch_id, feed, src_row, reason, payload | Rejected feed rows kept for correction (E6) |
| App_User, Audit_Log | username, password hash, salt, iterations, role, failed attempts, lock time; audit records | Authentication (E7) |

In R1 the order data model also records status, receipt or delivery quantity and time, and unique source references; the extensions are not part of the original fourteen-table model.

Source: Authors' own work

Table S1e. Synthetic demonstration of the logical data model

| Table | Values (attribute order of Supplementary Table S1c) |
|---|---|
| Region | ***RG02**, "Coastal Region" |
| City | ***100012**, "Port City", ****RG02** |
| District | ***07**, "East District" |
| Depot_Type | ***DT1**, "Primary depot"; ***DT2**, "Secondary depot" |
| Depot | ***D001**, "East Terminal", "1 Terminal Road", 60000, 17500, ****100012, **07, **DT1**; ***D101**, "Inland Depot 1", "4 Depot Lane", 9000, 3600, ****100012, **07, **DT2** |

*Table S1e (continued)*

| Table | Values (attribute order of Supplementary Table S1c) |
|---|---|
| Product | ***P03**, “Diesel”, “DSL”, 1 |
| Refinery | ***R02**, “Coastal Refinery”, “2 Refinery Road” |
| Transfer_Type | ***PIPE**, “Pipeline” |
| Movement_Type | ***RAIL**, “Rail” |
| Transporter | ***T02**, “Carrier 2”, ****RAIL** |
| Production_Record | ***P03/25/001**, 2025-08-10, 9600, 3600, ****R02, ****P03** |
| Supply_Order | ***P03/25/001**, 6000, 12, 2025-08-10, 2025-08-12, “Scheduled pipeline batch”, ****PIPE, ****R02, ****P03/25/001, ****P03, ****D001** |
| Replenishment_Order | ***P03/25/001**, 2500, 25, 2400, 60000, 2025-08-12, 2025-08-14, “Rail transfer to inland depot”, ****D001, ****D101, ****P03, ****T02** |
| Stores | ***D001, *P03**, 17500; ***D101, *P03**, 3600 |

Notes: *Primary key (component); **Foreign key. Document codes are illustrative; source references and production, supply and replenishment identifiers belong to distinct tables. Stocks before the chain: D001 14,000 and D101 1,100. Then: remaining production 9,600 − 6,000 = 3,600; D001 14,000 + 6,000 − 2,500 = 17,500; D101 1,100 + 2,500 = 3,600; total cost 25 × 2,400 = 60,000 (25 rail cars of 100 $m^3$; twelve pipeline batches of 500 $m^3$); occupied volume = sum of stocks × unit volume (each example depot holds one product). The example distinguishes an order on 12 August from its receipt on 14 August; R1 records stock changes on the corresponding physical events. All records are synthetic. Source: Authors’ own work

## S1.1 Full conceptual association table

Supplementary Table IV. Associations and cardinalities of the conceptual data model

| Association | Entities involved | Cardinality | Attribute / role |
|---|---|---|---|
| Belongs_to | Depot × City | (1,1) : (0,N) | – |
| Managed_by | Depot × District | (1,1) : (0,N) | – |
| Has_type | Depot × Depot_Type | (1,1) : (0,N) | – |
| Includes | City × Region | (1,1) : (0,N) | – |
| Benefits_from | Depot × Supply_Order | (0,N) : (1,1) | – |
| Executed_by | Supply_Order × Transfer_Type | (1,1) : (0,N) | – |
| Supplies | Supply_Order × Product | (1,1) : (0,N) | – |
| Sent_by | Supply_Order × Refinery | (1,1) : (0,N) | – |
| Linked_to | Supply_Order × Production_Record | (1,1) : (0,N) | – |
| Produced_by | Production_Record × Refinery | (1,1) : (0,N) | – |
| Contains | Production_Record × Product | (1,1) : (0,N) | – |
| Stores | Depot × Product | (0,N) : (0,N) | stock_quantity |
| Issues | Depot × Replenishment_Order | (0,N) : (1,1) | emitting depot |
| Receives | Replenishment_Order × Depot | (1,1) : (0,N) | receiving depot |
| Refuels | Replenishment_Order × Product | (1,1) : (0,N) | – |
| Carried_by | Replenishment_Order × Transporter | (1,1) : (0,N) | – |
| Performs | Transporter × Movement_Type | (1,1) : (0,N) | – |

Note: Cardinalities follow the order of the entities in the second column. Source: Revised from Loutis and Aïmeur (2026), Table 2, *Design and Deployment of a Fuel Logistics Information System*. `https://doi.org/10.1201/9781003647300-28`. Cardinalities were re-examined for the present study.

## S2. Action rules and calculation rules

### S2.1 Action rules

Each item gives the two rules of one association of Supplementary Table IV (34 rules).

- **Belongs_to.** A depot belongs to one and only one city; a city may contain zero, one or several depots.
- **Managed_by.** A depot is managed by one and only one district; a district may manage zero, one or several depots.
- **Has_type.** A depot has one and only one type; a type may group zero, one or several depots.
- **Includes.** A city belongs to one and only one region; a region may group zero, one or several cities.
- **Benefits_from.** A depot may benefit from zero, one or several supply orders (secondary depots from none); a supply order benefits one and only one depot.
- **Executed_by.** A supply order is carried out with one and only one transfer type; a transfer type may serve zero, one or several supply orders.
- **Supplies.** A supply order concerns one and only one product; a product may appear in zero, one or several supply orders.
- **Sent_by.** A supply order is sent by one and only one refinery; a refinery may send zero, one or several supply orders.
- **Linked_to.** A supply order draws on one and only one production record; a production record may feed zero, one or several supply orders.
- **Produced_by.** A production record is produced by one and only one refinery; a refinery may produce zero, one or several production records.
- **Contains.** A production record concerns one and only one product; a product may appear in zero, one or several production records.
- **Stores.** A depot may store zero, one or several products; a product may be stored in zero, one or several depots.
- **Issues.** A depot may emit zero, one or several replenishment orders; a replenishment order has one and only one emitting depot.
- **Receives.** A replenishment order has one and only one receiving depot; a depot may receive zero, one or several replenishment orders.
- **Refuels.** A replenishment order concerns one and only one product; a product may appear in zero, one or several replenishment orders.
- **Carried_by.** A replenishment order is carried by one and only one transporter; a transporter may carry zero, one or several replenishment orders.
- **Performs.** A transporter performs one and only one movement type; a movement type may be performed by zero, one or several transporters.

Business rules beyond the cardinalities, enforced in R1 only: a supply order delivers a primary depot and draws on a production record of the same refinery and product; a replenishment order runs from a primary to a different secondary depot; quantities and element counts are positive; end and unloading dates are not earlier than start and loading dates; the remaining production quantity cannot become negative; an order cannot take more than the emitter's stock or push the receiver above its storage capacity (checked when the order is issued; physical receipts and

deliveries are always recorded and raise an alert when they create a negative book stock or exceed capacity).

### S2.2 Calculation rules

The calculation rules use the quantity and timing of the recorded physical movements. Occupied volume is calculated over all products stored in a depot.

- Total cost of a replenishment order = cost_per_element × delivered_elements_count.
- Remaining production quantity = quantity_produced − sum of supplied_quantity of the supply orders linked to the record.
- Stock of a primary depot after a supply = previous stock + supplied (R0: when the order is created; R1: delivered quantity at receipt).
- Stock of the emitting depot after a recorded replenishment dispatch (R1) = previous stock − dispatched quantity; the published logic reduces the emitter stock when the order is created.
- Stock of the receiving depot after replenishment = previous stock + received quantity (R1, when receipt is recorded); the published logic credits the receiver when the order is created.
- Stock after an outbound (marketing) delivery (R1 only) = previous stock − delivered quantity.
- Occupied volume of a depot = $\sum_{\text{products}}$ stock_quantity × unit_volume.
- Inventory position (R1) = stock + planned inbound supply and replenishment quantities.
- Elements of an order = ⌈quantity / element size⌉ (100 $m^3$ rail car, 36 $m^3$ road tanker, 500 $m^3$ pipeline batch and 1,000 $m^3$ coastal parcel in the synthetic network).
- Days of cover (R1 dashboard) = stock / mean daily outbound delivery over the last 28 days.
- Count discrepancy (R1) = measured quantity − book quantity; an alert is raised when its absolute value exceeds the tolerance of the depot-product.

Worked example (Table S1e): remaining 9,600 − 6,000 = 3,600 $m^3$; D001 14,000 + 6,000 − 2,500 = 17,500 $m^3$; D101 1,100 + 2,500 = 3,600 $m^3$; total cost 2,400 × 25 = 60,000; occupied volume of D001 17,500 × 1 = 17,500 $m^3$ out of 60,000 $m^3$.

### S2.3 Design and source-review tables

Supplementary Table V. Capabilities described in the earlier chapter and evidence in the released code

| Described capability | Status in released code | Evidence (inspected) |
|---|---|---|
| 46 windows, RAD Studio and SQL Server | Implemented, not executable as published | 46 form units; missing form files, project file and schema |
| Relational database with procedures and triggers | Partially implemented | 3 procedures, 2 triggers, 11 views; defects D1–D8 |
| Optimized indexing | Not found | No index definition; non-sargable year predicates |
| Automated daily reports | Implemented with defects | Nine report views assessed (of eleven SQL views); two have incorrect output or fail (D3, D7) |
| Real-time monitoring, dashboard updates | Not evidenced in released code | No refresh or notification; the only timers update a clock label |
| Stock forecasting | Not evidenced in released code | “Forecast” views return previous-year totals |
| Alert generation | Not evidenced in released code | No alert entity, rule or notification |

*Supplementary Table V (continued)*

| Described capability | Status in released code | Evidence (inspected) |
|---|---|---|
| Role-based access control | Partially implemented | Role equals user name; menus differ by role; clear-text passwords; no database-level separation in the code |
| IoT endpoints, PLC/FPGA controllers, cloud services | Not evidenced in released code | No interface, library or configuration in the code |
| Backup and recovery documented in the repository | Partially implemented | Maintenance-plan logs of full backups; no restore procedure |
| ISO/IEC 27005 risk management, defense in depth | Not evidenced | No risk assessment or layered controls beyond the login |

Source: Authors' own work

Supplementary Table VI. Defects of the published logic and their measured consequences

| ID | Defect | Consequence (scenario) |
|---|---|---|
| D1 | Document number COUNT(*)+1 formatted with STR(n,3) | The 1,000th record is numbered P01/24/***, every later one fails (T0); 21.8–24.4 % of one year's documents accepted (T1b) |
| D2 | Triggers process one row per insert statement | Three orders inserted in one statement: 1 of 3 receivers updated, emitter decreased by 100 instead of 300 $m^3$ (T0) |
| D3 | Occupied volume increased by the new stock instead of the delivered quantity; not updated by stock entry | A 100 $m^3$ delivery raised the receiver's occupied volume by 3,081 $m^3$ (T0), and the trigger subtracts the same amount from the emitter's (inspected); depot status 0 of 36 fields correct (T3) |
| D4 | Unit volume stored and computed as an integer | 0.003785 stored as 0 (T0) |
| D5 | Stock update skipped when the depot-product row is missing | A 1,000 $m^3$ supply created no stock record but raised the occupied volume by 1,000 $m^3$ (T0) |
| D6 | No check of quantities, dates, emitter $\neq$ receiver, emitter stock, receiver capacity, depot tiers or production provenance | 57–64 % of defective feed rows accepted (T1); 7 of 30 seeded errors intercepted (T7) |
| D7 | General supply forecast cross-joins supply and replenishment orders | Fails with an arithmetic overflow once a year of orders has accumulated: 0 of 128 fields correct (T3), failure in every request at every load level (T10) |
| D8 | Year predicates not usable by indexes; no secondary index | Previous-year reports slower as data grow: 95th percentile 0.24 s at the nominal load, 1.31 s at 5× (T10) |
| D9 | No record of outbound deliveries, receipts or counts; stocks overwritten | 42–43 % of feed rows cannot be represented (T1); losses invisible (T7, T12) |
| D10 | Receiver credited when the order is created | Stock shown before it arrives (inspected) |
| D11 | Clear-text passwords, role = user name, no lockout, no database role separation | 25 of 25 forbidden operations succeed through the application login; no lockout after 1,000 guesses (T9) |
| D12 | The supply-order form sets the procedure parameters by position: the transfer mode ('cana' or 'cabo') goes to the fourth parameter, which the procedure takes as the refinery code, and the value commented as the refinery code to the production-record parameter | Wrong codes stored or rejected by a key; which query feeds each value is in the missing form file, so the client was not corrected (inspected) |
| D13 | "Forecast" views are previous-year totals | Baseline M0 in T4 |
| D14 | No alert, notification, refresh, IoT, cloud or audit function | Claims not supported (Table V) |
| D15 | Full backups only; no log backups or restore procedure | Data since the last full backup lost after a media failure (T11) |

Note: T0 reproduces each defect directly on R0, with the published T-SQL executed unchanged on SQL Server, by

a minimal transaction (Supplementary Section S3.0); the other scenarios are those of Table III. Source: Authors' own work

Supplementary Table VII. Corrections (C) and extensions (E) in the revised artefact R1

| ID | Change | Reason | DP | Test | SQL Server scripts |
|---|---|---|---|---|---|
| C1 | Document numbers from sequences (product/year/six digits) | D1 overflow and race of COUNT(*)+1 | DP1 | T1b, T10 | yes |
| C2 | Set-based posting of every movement through a ledger trigger; missing stock rows created | D2, D5 | DP1 | T1, T3 | yes |
| C3 | Occupied volume derived from stocks | D3 | DP1 | T3 | yes (trigger on the stock table) |
| C4 | Positive quantities, date order, depot tiers, emitter $\neq$ receiver, production provenance, remaining quantity $\geq 0$; emitter stock and receiver capacity checked when an order is issued | D6 | DP2 | T1, T7 | yes, except depot tiers |
| C5 | Unique source reference of documents received from feeds | Duplicate transactions | DP2 | T1, T7 | no (no source reference in the published model) |
| C6 | General supply forecast rebuilt without cross join; range predicates; indexes on dates and keys | D7, D8 | DP1 | T3, T10 | yes |
| C7 | Decimal unit volume | D4 | DP1 | – | optional (needs a client change) |
| E1 | Outbound (marketing) deliveries and stock-movement ledger | D9 | DP1 | T1, T4 | no |
| E2 | Order life cycle (planned, delivered) with received quantity; receiver credited on receipt | D10 | DP1 | T1, T3 | no |
| E3 | Stock counts compared with the book; discrepancy alert beyond a tolerance | D9 | DP2 | T7, T12 | no |
| E4 | Alerts: critical stock, reorder point on the inventory position, overfill, abnormal outbound delivery, count and transit discrepancies, negative book, capacity, overdue delivery | D14 | DP3 | T5 | no |
| E5 | Dashboard view and change notification to clients | D14 | DP3 | T2 | no |
| E6 | Feed staging: validation, unit normalization, duplicate and receipt cross-checks, quarantine | Data integration | DP2 | T1 | no |
| E7 | Role logins with grants on operations only; PBKDF2 password hashes and lockout; full and transaction-log backups | D11, D15 | DP5 | T9, T11 | no |
| E8 | Supply routes and order suggestions | Decision delay, manual interventions | DP4 | T6, T8 | no |

Note: The evaluated R1 database logic was defined in T-SQL; three parts of R1 ran in the client layer of the test harness: the feed validator of E6 (with the quarantine table in the database), the application-login check of E7 (PBKDF2 hashes and lockout stored in the database), and the creation of the role logins and their grants (E7). Last column: corrections back-ported to the SQL Server scripts of the published repository (see the end of this section). Source: Authors' own work

## S3. Evaluation protocols and supporting results

### S3.0. Environment, comparators and measurement rules

The evaluation defines three conditions: MANUAL, the modelled fragmented procedure; R0, the reconstructed published logic with the original procedures, triggers and report views; and R1, the revised artefact. For performance and recovery tests only, a distinct R0 numbering-controlled arm prevents the known identifier overflow from stopping the other operations. Database measurements use Microsoft SQL Server 2022 Developer in disposable databases on Linux, limited to 2 GB of memory, with twelve logical processors. The application forms are emulated from available Delphi source, not compiled as a running Delphi application. The synthetic generator and tests are executed by Python scripts. The development data use seed 1; the principal replenishment simulations report seeds 101–105, whereas their one-at-a-time sensitivity analysis covers only seeds 101–103. The 150-day stock-out totals include the warm-up interval and are not separately estimated for a post-warm-up horizon. Tests of forecasting methods use the public weekly EIA series and a held-out 2024–2026 test interval.

Rates count the exact valid, invalid, accepted or rejected operations of each designed test. Machine timings report medians, 95th and 99th percentiles or maxima, according to the scenario. The process and discrete-event models report values derived from their stated assumptions, separately from database measurements. Scenarios T1 to T12 are defined in Table III; a concise operational protocol for each is below.

### S3.1. T1: Data integration and sustained operation

An independent generator labels each input row as valid or defective before submitting fourteen days of production records, supply orders, replenishment orders, stock reports, outbound deliveries and receipts. The feed experiments include 1 %, 5 % and 10 % injected defect rates and compare the operations supported by both R0 and R1 separately from R1's additional coverage. Integration defect rate = (valid rows lost + defective rows accepted silently) / input rows; contamination is counted only among defective input rows. The sustained-operation arm submits 365 days of production records and orders to identify the effects of the original numbering logic.

### S3.2. T2 and T5: Monitoring and alerts

Committed stock changes are replayed to push and polling dashboards at nominal and fivefold event/client loads. Submission-to-display latency is measured until a client observes the new row version. Alert expectations are computed independently from stock thresholds, abnormal deliveries, reconciliation discrepancies and overdue orders. Recall counts raised expected alerts; false positives count alerts without a corresponding expected event. Time-based overdue alerts are checked by a periodic monitor; their latency definition differs from that of event-driven notifications. R0 lacks these functions.

### S3.3. T3: Reports

Nine report views are evaluated against independently computed fields. The eight report layouts shared by R0 and R1 are compared like for like; the general supply forecast is evaluated separately because its layout differs in R1. Results count matching fields out of all reference fields, the occurrence of query errors, and maximum query duration under an isolated request protocol. The report scope and totals are summarized in Section 6.2.

### S3.4. T4: Demand and stock forecasting

One-week-ahead forecasts are computed at rolling weekly origins. Method selection is performed on the 2021–2023 validation period and then frozen before evaluating the 2024–2026 test observations. Forecast scores include MAPE, MAE, RMSE and MASE. Comparisons with M0 use weekly paired absolute-percentage-error differences, circular moving-block bootstrap intervals, Diebold–Mariano tests with finite-sample correction, and Holm adjustment across the five series. Synthetic daily depot demand is a separate experiment at three noise levels. Table S3i isolates the selected-method comparisons.

### S3.5. T6–T8: Work, errors and supply delay

T6 counts per-day business actions under the same synthetic workload in MANUAL, R0 and R1 using a process model. T7 first submits designed invalid, inconsistent and derived-value probes to the evaluated logic, and then uses a separate process model to estimate errors reaching the record after the controls. T8 uses a discrete-event model of replenishment after the inventory position crosses a reorder point. Information delay, decision delay, dispatch waiting time and transport time are accounted for separately; the two-postings arm changes the availability of outbound information rather than transport time. The modelled arms use literature-informed assumptions and sensitivity ranges, not field observations.

### S3.6. T9–T11: Security, scalability and recovery

T9 attempts allowed and forbidden role–operation combinations and tests authentication and lockout behaviour. T10 replays a fixed request mix under one-, two-, three- and fivefold load, with R0 and R1 on the same server and calendar; failures include database query errors. The general supply forecast in R0 is reported as failed rather than assigned an artificial successful latency. T11 simulates loss of database data and log files and restores from each strategy's available backups. R1 uses full and transaction-log backups at a 60-second interval, unlike the full-backup-only published strategy. Transaction loss is measured relative to the committed state at failure.

### S3.7. T12: Disruption scenarios

The discrete-event model also evaluates a temporary 60 % demand surge, a two-day rail delay, a four-hour database outage and a media-failure recovery, reporting mean supply delay, stock-out hours and unmet demand over the stated seeds. The loss-detection thought experiment contrasts record-based discrepancy alerts with a paper-book count comparison. Simulated effects are distinct from physical operating-site tests.

### S3.8. Supplementary results

Table S3a. Business actions per operating day by transaction type (process model, final workload)

| **Transaction** | **Per day** | **Manual** | **R0** | **R1** | **R1 + meter feed** | **R1 + meter + gauging** |
|---|---|---|---|---|---|---|
| Production record | 10.0 | 100.0 | 90.0 | 20.0 | 20.0 | 20.0 |
| Supply order | 10.6 | 169.6 | 159.0 | 21.2 | 21.2 | 21.2 |
| Supply receipt | 10.5 | 105.0 | 21.0 | 21.0 | 21.0 | 21.0 |
| Replenishment order | 13.2 | 251.5 | 185.3 | 36.3 | 36.3 | 36.3 |

*Table S3a (continued)*

| Transaction | Per day | Manual | R0 | R1 | R1 + meter feed | R1 + meter + gauging |
|---|---|---|---|---|---|---|
| Replenishment receipt | 13.2 | 158.3 | 26.4 | 26.4 | 26.4 | 26.4 |
| Outbound deliveries | 90.1 | 360.2 | 180.1 | 180.1 | 0.0 | 0.0 |
| Stock count | 110.0 | 473.8 | 583.8 | 220.0 | 220.0 | 0.0 |
| Daily reports | 4.0 | 16.0 | 8.0 | 0.0 | 0.0 | 0.0 |
| Stock monitoring | 1.0 | 36.0 | 37.0 | 1.0 | 1.0 | 1.0 |
| Total | | 1670.6 | 1290.7 | 526.0 | 345.9 | 125.9 |

Meter feed and automatic tank gauging are not implemented; they show where the remaining actions lie. Source: Authors' own work

Table S3b. Summary of seeded-error probes (T7)

| Probe group | Number of probes | Published logic (R0) | Revised artefact (R1) |
|---|---|---|---|
| Designed error probes, full scope | 30 | 7 intercepted; 3 cases unsupported by the model | 24 intercepted |
| Common-scope error probes | 27 | 7 intercepted | 21 intercepted |
| Control transaction | 1 | Accepted without flag | Accepted without flag |

Source: Authors' own work

Table S3c. Independent report checks in T3

| Measure | Published logic (R0) | R1 |
|---|---|---|
| Fields matching reference, eight common report layouts, nominal network | 4,697/4,733 | 4,733/4,733 |
| Fields matching reference, common layouts, twofold network | 6,875/6,941 | 6,941/6,941 |
| General supply forecast | Arithmetic overflow (error 8115); 0/128 reference fields | 50/50 fields in the revised layout |
| Full nine-report inventory, nominal network (version-specific layouts) | 4,697/4,861 fields correct in isolated logic checks | 4,783/4,783 fields correct |
| Maximum report generation duration, nominal network | Up to 1.4 s before failure of the general forecast | 0.037 s |
| Maximum report generation duration, twofold network | Up to 3.9 s before failure | 0.049 s |

Note: Eight common report layouts contain 4,733 fields (6,941 in the twofold network). Adding the version-specific general supply forecast yields 4,861 fields for R0 (128 forecast fields) and 4,783 for R1 (50 forecast fields). These totals are distinct output inventories, not matched-field denominators. The R0 full-inventory figure refers to isolated report-logic checks; separately, sustained-operation numbering failures reduced its end-to-end result to 1,105/4,861. Source: Authors' own work

Table S3d. Load-level results in T10 (SQL Server 2022)

| Load | R0 failed forecast requests | R0 slowest other P95 (s) | R1 slowest operation P95 (s) | R1 requests / failures |
|---|---|---|---|---|
| 1× | 33/33 | 0.236 | 0.050 | 697 / 0 |
| 2× | 37/37 | 0.642 | 0.064 | 1,436 / 0 |
| 3× | 54/54 | 1.054 | 0.088 | 2,210 / 0 |
| 5× | 99/99 | 1.309–1.359 | 0.145–0.178 | 3,509 / 0 |

*Table S3d (continued)*

| Load | R0 failures | R0 other P95 | R1 worst P95 | R1 requests / failures |
|---|---|---|---|---|
| 5×, R1 without indexes | – | – | 0.160–0.166 | 3,494 / 0 |

Note: R0 forecast failures were query overflows, not time-outs. P95 is the operation-specific 95th percentile, not the P95 aggregated across all request types. Source: Authors' own work

Table S3e. Test MAPE (%) of all methods, one week ahead, EIA data 2024–2026

| Series | M0 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | M9 |
|---|---|---|---|---|---|---|---|---|---|---|
| gasoline_demand | 3.31 | 3.74 | 3.17 | 3.18 | 3.12 | 2.95 | 3.35 | 2.84 | 3.02 | 2.96 |
| distillate_demand | 6.81 | 8.82 | 7.49 | 7.54 | 7.02 | 5.99 | 7.16 | 6.35 | 6.98 | 7.15 |
| jet_demand | 6.81 | 9.36 | 8.34 | 7.06 | 6.79 | 6.84 | 8.29 | 6.59 | 6.78 | 6.61 |
| gasoline_stocks | 5.26 | 1.13 | 3.15 | 2.13 | 0.93 | 0.83 | 1.62 | 0.93 | 1.38 | 0.96 |
| distillate_stocks | 6.81 | 1.85 | 5.81 | 3.06 | 1.98 | 1.59 | 3.11 | 1.77 | 2.26 | 1.99 |

Source: Authors' own work

Table S3f. Synthetic depot-level demand forecasting by noise level (T4)

| Noise CV | M0 previous-year mean MAPE | Selected model MAPE |
|---|---|---|
| 0.05 | 7.67 % | 4.07 % |
| 0.10 | 10.41 % | 8.18 % |
| 0.20 | 17.76 % | 16.63 % |

The differences are 3.60, 2.23 and 1.13 percentage points, respectively. These are results of an experiment on generated depot-level series, not measured depot demand. Source: Authors' own work

Table S3g. Simulated-media-failure restoration results (T11)

| Backup strategy | Trials | Restore time | Documents lost, trials 1/2/3 |
|---|---|---|---|
| R0, start-of-day full backup | 3 | 0.42–0.43 s | 40 / 40 / 40 |
| R1, full plus 60-s log backups | 3 | 0.84–0.97 s | 30 / 0 / 0 |

R1 additionally lost 58–106 rows committed since the last log backup across the runs (at most 32 stock movements). These quantities are distinct from the counts of documents. R0 used a numbering-controlled arm. Source: Authors' own work

Table S3h. Disruption scenarios (T12, modelled, mean of seeds 101–105)

| Scenario | Arm | Supply delay (h) | Fill rate (%) | Stock-out depot-product hours | Unmet demand ($m^3$) |
|---|---|---|---|---|---|
| S0 no disruption | MANUAL | 73.7 | 99.82 | 423.6 | 5476.8 |
| S0 no disruption | R0 | 63.62 | 100.0 | 7.6 | 130.0 |
| S0 no disruption | R1 | 55.28 | 100.0 | 0.0 | 0.0 |
| S0 no disruption | R1 two postings | 46.52 | 100.0 | 0.0 | 0.0 |

*Table S3h (continued)*

| Scenario | Arm | Supply delay (h) | Fill rate (%) | Stock-out depot-product hours | Unmet demand ($m^3$) |
|---|---|---|---|---|---|
| S0 no disruption | R1+meter feed (not implemented) | 44.34 | 100.0 | 0.0 | 0.0 |
| S1 demand surge (+60 %, 7 days, 1/6 of depots) | MANUAL | 73.74 | 99.78 | 502.6 | 6759.0 |
| S1 demand surge (+60 %, 7 days, 1/6 of depots) | R0 | 63.7 | 99.97 | 45.0 | 783.8 |
| S1 demand surge (+60 %, 7 days, 1/6 of depots) | R1 | 55.26 | 100.0 | 0.0 | 0.0 |
| S1 demand surge (+60 %, 7 days, 1/6 of depots) | R1 two postings | 46.56 | 100.0 | 0.0 | 0.0 |
| S1 demand surge (+60 %, 7 days, 1/6 of depots) | R1+meter feed (not implemented) | 44.44 | 100.0 | 0.0 | 0.0 |
| S2 unforeseen rail delay (+2 days, 3 days) | MANUAL | 74.16 | 99.81 | 481.6 | 5820.4 |
| S2 unforeseen rail delay (+2 days, 3 days) | R0 | 64.22 | 99.98 | 46.4 | 622.0 |
| S2 unforeseen rail delay (+2 days, 3 days) | R1 | 55.82 | 100.0 | 1.0 | 11.6 |
| S2 unforeseen rail delay (+2 days, 3 days) | R1 two postings | 47.24 | 100.0 | 0.0 | 0.0 |
| S2 unforeseen rail delay (+2 days, 3 days) | R1+meter feed (not implemented) | 44.82 | 100.0 | 0.0 | 0.0 |
| S3 database outage 4 h (day 60, 09:00-13:00) | MANUAL | 73.7 | 99.82 | 423.6 | 5476.8 |
| S3 database outage 4 h (day 60, 09:00-13:00) | R0 | 63.9 | 99.99 | 10.4 | 158.8 |
| S3 database outage 4 h (day 60, 09:00-13:00) | R1 | 55.28 | 100.0 | 0.0 | 0.0 |
| S3 database outage 4 h (day 60, 09:00-13:00) | R1 two postings | 46.56 | 100.0 | 0.0 | 0.0 |
| S3 database outage 4 h (day 60, 09:00-13:00) | R1+meter feed (not implemented) | 44.4 | 100.0 | 0.0 | 0.0 |
| S4 media failure (day 60, 15:00) | MANUAL | 73.7 | 99.82 | 423.6 | 5476.8 |
| S4 media failure (day 60, 15:00) | R0 | 63.68 | 100.0 | 6.8 | 107.4 |
| S4 media failure (day 60, 15:00) | R1 | 55.28 | 100.0 | 0.0 | 0.0 |
| S4 media failure (day 60, 15:00) | R1 two postings | 46.52 | 100.0 | 0.0 | 0.0 |
| S4 media failure (day 60, 15:00) | R1+meter feed (not implemented) | 44.38 | 100.0 | 0.0 | 0.0 |

The modelled values use a common disruption schedule and are not observations from operating depots. Stock-out depot-product hours, unmet demand and fill rates cover the entire 150-day simulation, including its warm-up/initialization period. No separately computed post-warm-up steady-state outcome is reported; the totals should not be interpreted as such. Source: Authors' own work

Table S3i. Paired forecast comparison: selected method minus M0

| Weekly series | MAPE difference (points) | Paired 95 % interval (points) | Holm-adjusted p |
|---|---|---|---|
| Gasoline demand | −0.35 | [−0.76, 0.05] | 0.18 |
| Distillate demand | −0.83 | [−1.57, −0.11] | 0.068 |
| Jet demand | 0.03 | [−0.68, 0.72] | 0.94 |
| Gasoline stocks | −4.43 | [−6.03, −2.87] | < 0.001 |
| Distillate stocks | −5.22 | [−6.93, −3.58] | < 0.001 |

Differences are computed on the same forecast weeks. Intervals are paired moving-block bootstrap intervals; adjusted p-values use the five-series Holm procedure. Source: Authors' own work

## S3.9 Detailed evaluation tables

Supplementary Table VIII. Integration of multi-source feeds with seeded defects (T1, final validation)

| Arm | Rows (valid / defective) | Loss of valid rows (retained) | Defective rows accepted | Integration defect rate | Consistent balances |
|---|---|---|---|---|---|
| *Defect rate 5 %, seeds 101 / 102 / 103* | | | | | |
| R0 forms | 1,887–2,139 / 95–118 | 0.37 / 0.14 / 0.51 % (none) | 62.5 / 64.2 / 56.8 % | 3.38 / 2.90 / 3.46 % | 100.0 / 98.4 / 97.6 % |
| R0 bulk import | same | 40.0 / 35.8 / 48.4 % (none) | 44.8 / 55.8 / 43.2 % | 40.2 / 36.7 / 48.1 % | 98.2 / 0.0 / 96.7 % |
| R1 direct | 3,279–3,680 / 172–193 | 0.64 / 0.50 / 0.63 % (none) | 7.0 / 8.5 / 5.7 % | 0.96 / 0.89 / 0.88 % | 89.1 / 86.1 / 87.8 % |
| R1 staged | same | 0.58 / 0.36 / 0.52 % (all) | 0.0 / 0.0 / 0.0 % | 0.55 / 0.34 / 0.49 % | 96.4 / 99.2 / 96.7 % |
| R1 staged, published scope | as R0 | 0.37 / 0.14 / 0.51 % (all) | 0.0 / 0.0 / 0.0 % | 0.35 / 0.14 / 0.49 % | – |
| *Defect rate 10 % (seed 104) and 1 % (seed 105)* | | | | | |
| R0 forms | | 0.46 / 0.05 % | 62.3 / 68.2 % | 6.55 / 0.74 % | 93.3 / 99.2 % |
| R1 staged | | 1.07 / 0.05 % (all) | 0.3 / 0.0 % | 0.99 / 0.05 % | 95.8 / 99.2 % |

Notes: R0 rows exclude the outbound deliveries and receipts that the published model cannot represent (1,468–1,616 rows per seed, 42–43 % of all rows); the row "R1 staged, published scope" restricts R1 to the same four feeds and rows. Balance consistency is measured under each system's own posting rule and is not comparable across systems. Integration defect rate = (valid rows lost + defective rows accepted) / rows. Consistent balances: depot-product stocks equal to the reference computed from the valid rows under each system's posting rule (R0: last count plus orders; R1: ledger); R0 bulk shows 0.0 % for seed 102 because the stock-report batches of the last two days were each rejected in full by one defective value, leaving earlier counts in place. Source: Authors' own work

Supplementary Table IX. Monitoring and alerting on R1 (T2, T5, final validation, SQL Server)

| Measure | Nominal (1×, seed 101) | 5× network, events and clients (seed 102) |
|---|---|---|
| Paced window | 3,103 events at 20/s; 1,543 stock changes | 14,016 events at 100/s; 7,668 stock changes |
| Push dashboard latency: median / p95 / max | 0.009–0.010 / 0.016–0.018 / 0.079–0.107 s (4 clients) | 0.007–0.008 / 0.806–1.306 / 7.327–9.061 s (20 clients) |
| Polling dashboard (every 5 s): median / max | 2.96 / 5.00 s | 3.01 / 5.13 s |

*Supplementary Table IX (continued)*

| **Measure** | **Nominal (1×, seed 101)** | **5× network, events and clients (seed 102)** |
|---|---|---|
| Client observations of stock changes | 6,172 of 6,172 per mode | 153,360 of 153,360 per mode |
| Final dashboard equal to the ledger | 110 / 110 depot-products | 502 / 502 depot-products |
| Expected alerts raised / missed / false positives | 244 / 0 / 0 | 1,144 / 0 / 0 |
| by kind (expected = raised) | reorder 195, critical 10, abnormal 6, count 7, negative book 6, overdue 20 | reorder 909, critical 62, abnormal 30, count 32, negative book 30, overdue 81 |
| Event-driven alert latency: median / max | 0.009 / 0.018 s | 0.007 / 0.415–0.663 s |
| Overdue-delivery alert latency: median / max | 6.4 / 8.1 s (monitor every 10 s) | 8.5 / 8.5 s (81 overdue alerts) |

Notes: SQL Server final runs. Latency measured from the submission of the triggering transaction (dashboard) or from the alert timestamp written in it (alerts); overdue alerts from the moment the simulated date passed the planned unloading date. Notifications through a Service Broker queue (Microsoft Corporation, 2025) read by one dispatcher connection, which forwards stock messages to the push clients; each push client then re-reads the changed dashboard rows. Load generator, clients (in four worker processes) and server on one workstation. At 5×, 37,245 of 37,265 push observations above 0.1 s in the tabulated run occurred in the first three tenths of the window; the timing ranges also include the independent repeat. R0 has no refresh, notification or alert function. Source: Authors' own work

Supplementary Table X. Test MAPE (MASE) one week ahead, EIA weekly data, January 2024 to November 2026

| **Series** | **Previous-year mean (M0)** | **Naive (M1)** | **Holt-Winters (M4)** | **Selected method** | **Selected: MAPE [95 % interval] (MASE)** | **Paired difference from M0, points [95 % interval] (Holm $p$)** |
|---|---|---|---|---|---|---|
| Gasoline demand | 3.31 % (0.75) | 3.74 % (0.85) | 3.12 % (0.71) | M9 | 2.96 % [2.53, 3.42] (0.68) | −0.35 [−0.76, 0.05] (0.18) |
| Distillate demand | 6.81 % (0.82) | 8.82 % (1.05) | 7.02 % (0.85) | M5 | 5.99 % [5.33, 6.75] (0.73) | −0.83 [−1.57, −0.11] (0.068) |
| Jet fuel demand | 6.81 % (0.60) | 9.36 % (0.82) | 6.79 % (0.59) | M5 | 6.84 % [6.09, 7.68] (0.60) | 0.03 [−0.68, 0.72] (0.94) |
| Gasoline stocks | 5.26 % (1.47) | 1.13 % (0.31) | 0.93 % (0.26) | M5 | 0.83 % [0.73, 0.92] (0.23) | −4.43 [−6.03, −2.87] (<0.001) |
| Distillate stocks | 6.81 % (0.52) | 1.85 % (0.14) | 1.98 % (0.15) | M5 | 1.59 % [1.34, 1.79] (0.12) | −5.22 [−6.93, −3.58] (<0.001) |

Notes: 146 weekly forecast origins for demand and 149 for stocks; methods as in Supplementary Section S3.4 (M5 harmonic regression with ARIMA(1,1,1) errors; M9 Holt-Winters on a training series with the 2020 pandemic weeks replaced by 2019). MASE: mean absolute error divided by the in-sample one-step error of the seasonal naive method over 2010–2020 (Hyndman and Koehler, 2006; seasonal form as in Hyndman and Athanasopoulos, 2021), a scale rather than a test. Selected-method intervals: moving-block bootstrap of that method's MAPE with a fixed block length of 8 weeks (2,000 resamples), unlike the paired differences, whose block length follows the rule of Politis and White (2004); they describe one method and do not test differences. Paired difference: selected minus M0 on the same weeks, mean difference of the absolute percentage errors with a circular moving-block bootstrap interval; Holm $p$: Diebold–Mariano test with the Harvey–Leybourne–Newbold correction, Holm-adjusted over the five series (Supplementary Table S3i). Source: Authors' own work, from U.S. Energy Information Administration (2026)

Supplementary Table XI. Response time in seconds by load level (T10, SQL Server)

| Load level | R0: general supply forecast | R0: slowest other operation, P95 | R1: slowest operation, P95 / maximum | R1: all requests, P95 / P99 / maximum | R1 requests / failures |
|---|---|---|---|---|---|
| 1× (12 users) | failed 33 of 33 | 0.236 | 0.050 / 0.077 | 0.040 / 0.051 / 0.077 | 697 / 0 |
| 2× (24 users) | failed 37 of 37 | 0.642 | 0.064 / 0.181 | 0.052 / 0.064 / 0.181 | 1,436 / 0 |
| 3× (36 users) | failed 54 of 54 | 1.054 | 0.088 / 0.100 | 0.060 / 0.070 / 0.100 | 2,210 / 0 |
| 5× (60 users) | failed 99 of 99 | 1.309–1.359 | 0.145–0.178 / 0.166–0.210 | 0.095 / 0.128 / 0.285 | 3,509 / 0 |
| 5×, R1 without its indexes | – | – | 0.160–0.166 / 0.176 | 0.110 / 0.139 / 0.320 | 3,494 / 0 |

R0: published logic with unique numbering; R1: frozen design; both seed 101, final validation on SQL Server, run on 17 November 2026 with the same calendar (686 days from 1 January 2025). Level $L$ multiplies the network, and with it the volume of its history, and the number of users of the nominal load (emulated users in worker processes of five); statement time-out 60 s; two repetitions of 60 s per run. The 5× slowest-operation P95 and maximum summarize three runs; the remaining request statistics refer to the tabulated run. P95, P99: 95th and 99th percentiles over all requests of the operation (or of all operations) in both repetitions; the slowest operation is the one with the highest P95. R0 general supply forecast: requests that failed; every failure was the arithmetic overflow of the cross-joined sums (error 8115), none a time-out. Further statistics: Supplementary Table S3d. Source: Authors' own work

Supplementary Table XII. Seeded-error probes by class (T7, measured, seed 101)

| Class | Probes | R0: prevented / flagged / not enterable / not flagged / not in model | R0 intercepted | R1: prevented / flagged / not enterable / not flagged / not in model | R1 intercepted |
|---|---|---|---|---|---|
| Invalid | 7 | 3 / 0 / 0 / 4 / 0 | 3/7 | 7 / 0 / 0 / 0 / 0 | 7/7 |
| Cross-record | 16 | 3 / 0 / 0 / 10 / 3 | 3/16 | 8 / 6 / 0 / 2 / 0 | 14/16 |
| Wrong but consistent | 4 | 0 / 0 / 0 / 4 / 0 | 0/4 | 0 / 0 / 0 / 4 / 0 | 0/4 |
| Derived | 3 | 0 / 0 / 1 / 2 / 0 | 1/3 | 0 / 0 / 3 / 0 / 0 | 3/3 |
| All probes | 30 | 6 / 0 / 1 / 20 / 3 | 7/30 | 15 / 6 / 3 / 6 / 0 | 24/30 |

Notes: Intercepted = prevented, flagged or not enterable. The correct control transaction was accepted and not flagged by both versions. Source: Authors' own work

Supplementary Table XIII. Mean supply delay of secondary-depot replenishment and its components in hours (T8, modelled, seeds 101–105)

| Arm | Information | Decision | Dispatch wait | Transit | Total | Reduction vs manual, % (seeds) | Order delay |
|---|---|---|---|---|---|---|---|
| Manual procedure | 24.0 | 6.0 | 10.8 | 33.0 | 73.7 | – | 29.9 |
| R0 (published logic) | 21.5 | 2.6 | 8.1 | 31.5 | 63.6 | 13.7 (12.7–14.4) | 24.1 |
| R1, deliveries posted at 20:00 | 7.8 | 13.3 | 2.0 | 32.1 | 55.3 | 25.0 (24.4–25.8) | 21.2 |
| R1, deliveries posted at 11:00 and 20:00 | 4.1 | 9.0 | 2.3 | 31.2 | 46.5 | 36.9 (36.0–38.3) | 13.1 |
| R1 with an hourly meter feed (not implemented) | 0.0 | 5.0 | 6.7 | 32.7 | 44.3 | 39.8 (38.8–41.5) | 5.0 |

Supply delay: from the moment the inventory position falls below the reorder point to the arrival of the replenishment; order delay: to the issued order. Means summarize the specified model horizon; stock-out totals reported separately

include the warm-up period. Components are means over all replenishments and may not add exactly to the total because of rounding. Transit times follow the same distributions in every arm; their means differ slightly with the mix of rail and road orders. The hypothetical hourly-meter scenario yielded a mean dispatch wait of 6.7 h, compared with 2.0–2.3 h in the other R1 scenarios; these values are outputs of the discrete-event model, which does not isolate the contribution of order-submission timing and loading-slot cut-offs. Source: Authors' own work

Supplementary Table S3j. Summary of results: published logic or manual procedure and revised artefact

| **Measure** | **Published logic (R0) or manual procedure** | **Revised artefact (R1)** | **Evidence** |
|---|---|---|---|
| Integration defect rate of multi-source feeds (lost valid + accepted defective rows / rows; 5 % seeded defects) | R0 form entry 2.90–3.46 % of the four feeds it can represent (42–43 % of all rows not representable); R0 bulk import 36.7–48.1 % | R1 staged, same four feeds 0.14–0.49 %; all six feeds 0.34–0.55 % | Measured |
| Production records and orders accepted over one year of nominal operation | R0 3,009/12,351 (24.4 %) | R1 12,351/12,351 (100.0 %) | Measured |
| Report fields equal to independent reference values (eight views with the same fields in both versions; the general supply forecast separately) | R0 logic 4,697/4,733 (99.24 %) (nominal network), 6,875/6,941 (99.05 %) (2× network); general supply forecast fails (arithmetic overflow, error 8115); published end to end 1,105/4,861 (22.73 %) of all fields | R1 4,733/4,733 (100.00 %) and 6,941/6,941 (100.00 %); general supply forecast 50/50 | Measured |
| Maximum report generation time | R0 logic: general supply forecast fails after up to 1.4 s (nominal) and 3.9 s (2×), isolated T3 reports (T10 concurrent nominal: 3.6 s); other reports at most 0.019 s and 0.025 s | R1 0.037 s (nominal), 0.049 s (2×) | Measured |
| Display latency of committed stock changes (push dashboard: median / max) | R0: no refresh or notification (not measurable) | R1 median 0.009–0.010 s, maximum 0.079–0.107 s (1×); median 0.007–0.008 s and maximum 7.327–9.061 s (5×); 5-s polling max 5.13 s | Measured |
| Expected alerts raised (false positives); notification latency | R0: no alert function (not measurable) | R1 244/244 (1×) and 1,144/1,144 (5×), 0 false positives; event-driven max 0.415–0.663 s, overdue deliveries max 8.5 s (monitor every 10 s) | Measured |
| Test MAPE one week ahead, EIA weekly series 2024–2026 (gasoline / distillate / jet demand; gasoline / distillate stocks) | Previous-year mean of the published views (M0) 3.31 / 6.81 / 6.81 / 5.26 / 6.81 % | Method selected on validation 2.96 / 5.99 / 6.84 / 0.83 / 1.59 %; paired difference from M0 established for gasoline stocks and distillate stocks (Holm-adjusted $p < 0.001$, $p < 0.001$), not for gasoline demand, distillate demand, jet demand (Holm-adjusted $p = 0.18$, $p = 0.068$, $p = 0.94$) | Measured |
| Seeded errors intercepted (prevented, flagged or not enterable); probes both versions can represent, and all probes | R0 7/27 | R1 21/27; all probes 24/30 | Measured |
| Business actions per operating day | Manual procedure 1,670.6; R0 1,290.7 (−22.7 %) | R1 526.0 (−68.5 %) | Modelled |

*Supplementary Table S3j (continued)*

| Measure | Published logic (R0) or manual procedure | Revised artefact (R1) | Evidence |
|---|---|---|---|
| Remaining errors (reduction relative to the manual procedure; mean and 2.5–97.5 percentiles over 20 Monte Carlo replications) | R0 −57.4 % [52.6, 64.0] | R1 −93.1 % [91.8, 94.4] | Modelled |
| Mean supply delay, reorder point crossed to arrival (seeds 101–105) | Manual procedure 73.7 h; R0 63.6 h (−13.7 %) | R1 55.3 h (−25.0 %); deliveries also posted at 11:00 46.5 h (−36.9 %, seeds 36.0–38.3 %) | Modelled |
| Forbidden role–operation attempts that succeeded | R0 25/25 through the application login (7/25 reachable from the menus) | R1 0/36, of which 0/25 on the pairs the published model can express (permitted operations 32/32) | Measured |
| Response time at five times the nominal load (seconds; 95th percentile of the slowest operation, 99th percentile and maximum of all requests) | R0 logic: general supply forecast failed in 99 of 99 requests (arithmetic overflow, error 8115; 0 time-outs); other operations highest P95 1.309–1.359 s | R1: slowest operation (general supply forecast) P95 0.145–0.178 s; all requests P99 0.128 s, maximum 0.285 s (0 failures in 3,509 requests); without its indexes, slowest operation P95 0.160–0.166 s | Measured |
| Data lost after a simulated media failure (three trials) | R0 strategy (full backup at the start of the day): all documents of the day (40, 40, 40 in the three trials; 67–150 committed transactions) | R1 (full backup + log backups every 60 s): 58–106 rows committed after the last log backup, at most 32 stock movements; documents lost 30, 0, 0 in the three trials (the first trial lost 30 of that day's 40 documents, all within the 60-s log-backup interval in the paced replay) | Measured |
| Unaided task completion by operators | Not tested (no participants) | Not tested (no participants) | Not evaluated |

Note: Results are classified as measured or modelled. Comparisons are restricted to capabilities shared by both implementations; additional R1 capabilities are reported separately. Source: Authors' own work

## S4. Evaluation settings and sensitivity checks

The numerical results in the main article refer to the evaluated final configuration and to the scenario variants stated in each table. The following checks clarify the scope of the comparisons; they do not refer to separate deposited experiment files.

Table S4. Evaluation settings and sensitivity checks

| Scenario | Evaluated setting | Interpretation |
|---|---|---|
| T4 | Rolling-origin validation followed by evaluation on a held-out public-data test interval | Model selection and held-out assessment are distinct |
| T8 | Outbound deliveries posted at 20:00 versus twice daily at 11:00 and 20:00 | Discrete-event simulation variants, not field measurements; mean delays 55.3 versus 46.5 h |
| T8 sensitivity | One-at-a-time parameter and reporting-schedule changes on seeds 101–103 only | A three-seed sensitivity subset of the five-seed main evaluation; ranges are conditional and are not five-seed uncertainty intervals |

*Table S4 (continued)*

| Scenario | Evaluated setting | Interpretation |
|---|---|---|
| T12 stock-outs | Entire 150-day simulation, including the warm-up/initialization period | Cumulative stock-out hours and fill rates are not post-warm-up steady-state estimates |
| T10 | R1 evaluated with and without its indexes under fivefold load | Slowest-operation P95 0.145–0.178 versus 0.160–0.166 s in the reported measurements |

Source: Authors' own work

## S5. Experimental specifications and scope

– **Software environment.** Microsoft SQL Server 2022 Developer, version 16.0.4325.4, on Linux with a 2 GB memory limit; twelve logical processors on the workstation running the Python client harness. SQL Server stores the reconstructed R0 tables and the R1 implementation. The missing Delphi form resources prevent execution of the original graphical client.

– **Generator and validation.** Development seed 1; independent synthetic validation seeds 101–105 for the indicated main analyses, but seeds 101–103 for the T8 one-at-a-time sensitivity checks; one-, two-, three- and fivefold load as described in T10. The test run calendar and data scope are specified in the main results. Input records, source references and probe labels are generated before the associated test.

– **Forecasting series.** Public U.S. weekly product-supplied and stock observations from the U.S. Energy Information Administration; validation through 2023 and held-out test observations from January 2024 through November 2026. All comparisons preserve the corresponding forecast-origin dates.

– **Modelled outcomes.** Process-action rates and error probabilities, replenishment timing and physical transport distributions are model parameters; the resulting quantities are not field performance measurements.

## References


Safinaz H. Abourokbah, Mohammad Asif Salam, and Nada Saleh Badawi. Supply chain information integration in the digital era leading to performance outcomes: A systematic review. *Sustainability*, 18(6):3098, 2026. doi: 10.3390/su18063098.

Shoaib Ahmed, Aneela Pathan, Fareesa Khan Sohu, Syed Haseeb Shah, Danish Munir Arain, and Rizwan Aziz Mangi. FPGA-based implementation of SCADA system for fuel management. *Quaid-e-Awam University Research Journal of Engineering, Science & Technology*, 21(2):21–28, 2023. doi: 10.52584/QRJ.2102.03.

Issar Arab, Safae Bourhnane, and Fatiha Kafou. Unifying Modeling Language-Merise integration approach for software design. *International Journal of Advanced Computer Science and Applications*, 9(4):6–12, 2018. doi: 10.14569/IJACSA.2018.090402.

David E. Avison. MERISE: A European methodology for developing information systems. *European Journal of Information Systems*, 1(3):183–191, 1991. doi: 10.1057/ejis.1991.33.

S. Biswas and Y. Narahari. Object oriented modeling and decision support for supply chains. *European Journal of Operational Research*, 153(3):704–726, 2004. doi: 10.1016/S0377-2217(02)00806-8.

Emma Brandon-Jones, Brian Squire, Chad W. Autry, and Kenneth J. Petersen. A contingent resource-based perspective of supply chain resilience and robustness. *Journal of Supply Chain Management*, 50(3):55–73, 2014. doi: 10.1111/jscm.12050.

Sunil Chopra and Peter Meindl. *Supply Chain Management: Strategy, Planning, and Operation*. Pearson, Boston, MA, 6th edition, 2016.

Martin Christopher and Helen Peck. Building the resilient supply chain. *The International Journal of Logistics Management*, 15(2):1–14, 2004. doi: 10.1108/09574090410700275.

Embarcadero Technologies. RAD Studio 12.3 released (press release). `https://www.embarcadero.com/press-releases/rad-studio-12-3-released`, 2025. Accessed 13 November 2026.

Fanxiu Fang, Tieliang Sun, Zheng Mei, Feng Yan, Lin Zhang, Lingyi Sun, and Wei Wang. Enhancing SCADA systems in oil and gas pipelines through electronics and full data-type publishing technology. *Australian Journal of Electrical and Electronics Engineering*, pages 1–11, 2025. doi: 10.1080/1448837X.2025.2459491. Published online 9 March 2025.

Marcos Vinícios Gonçalves and Antonio Eduardo Carrilho da Cunha. Scheduling tank trucks at a fuel distribution terminal using max-plus model-based predictive control. *Journal of Process Control*, 103:8–18, 2021. doi: 10.1016/j.jprocont.2021.05.005.

Shirley Gregor and Alan R. Hevner. Positioning and presenting design science research for maximum impact. *MIS Quarterly*, 37(2):337–355, 2013. doi: 10.25300/MISQ/2013/37.2.01.

Alan R. Hevner, Salvatore T. March, Jinsoo Park, and Sudha Ram. Design science in information systems research. *MIS Quarterly*, 28(1):75–105, 2004. doi: 10.2307/25148625.

Seyed Ashkan Hosseini Shekarabi, Reza Kiani Mavi, and Flavio Romero Macau. Supply chain resilience: A critical review of risk mitigation, robust optimisation, and technological solutions and future research directions. *Global Journal of Flexible Systems Management*, 26(3):681–735, 2025. doi: 10.1007/s40171-025-00458-8.

Rob J. Hyndman and George Athanasopoulos. *Forecasting: Principles and Practice*. OTexts, Melbourne, 3rd edition, 2021. URL `https://otexts.com/fpp3/`. Online edition, accessed 14 November 2026.

Rob J. Hyndman and Anne B. Koehler. Another look at measures of forecast accuracy. *International Journal of Forecasting*, 22(4):679–688, 2006. doi: 10.1016/j.ijforecast.2006.03.001.

Anas Iftikhar, Imran Ali, Ahmad Arslan, and Shlomo Tarba. Digital innovation, data analytics, and supply chain resiliency: A bibliometric-based systematic literature review. *Annals of Operations Research*, 333(2–3):825–848, 2024. doi: 10.1007/s10479-022-04765-6.

International Organization for Standardization and International Electrotechnical Commission. ISO/IEC 27005:2022 information security, cybersecurity and privacy protection – guidance on managing information security risks. `https://www.iso.org/standard/80585.html`, 2022.

Yisa Jiang, Taiwen Feng, and Yufei Huang. Antecedent configurations toward supply chain resilience: The joint impact of supply chain integration and big data analytics capability. *Journal of Operations Management*, 70(2):257–284, 2024. doi: 10.1002/joom.1282.

Joint Task Force. Security and privacy controls for information systems and organizations. NIST Special Publication 800-53 Rev. 5, National Institute of Standards and Technology, 2020.

Mohamed Loutis and Esma Aïmeur. Design and deployment of a fuel logistics information system. In Aamir Rashid and Fenio Annansingh-Jamieson, editors, *Moving Towards Sustainable Businesses and Developing More Resilient Economies*, pages 234–241. CRC Press, London, 2026. doi: 10.1201/9781003647300-28. Proceedings of the International Conference on Business, Management, Economics, and Information Systems (ICBMEIS 2025).

Anggi Manggiasih and Akhmad Hidayatno. Evaluating the conceptual model of the fuel supply chain from terminal to small scale fuel stations using Soft System Methodology approach. In *Proceedings of the 4th African International Conference on Industrial Engineering and Operations Management*, pages 166–173, Lusaka, Zambia, 2023. IEOM Society International. doi: 10.46254/AF04.20230054.

Sylvia Mardiana. Gasoline policy simulation to increase responsiveness using system dynamics: A case study of Indonesia's gasoline downstream supply chain. *International Journal of Energy Economics and Policy*, 13(6):109–118, 2023. doi: 10.32479/ijeep.14933.

Giovanni Mazzuto, Ilaria Pietrangeli, Marco Ortenzi, Filippo Emanuele Ciarapica, and Maurizio Bevilacqua. Leveraging digital twin for operational resilience in the oil and gas industry. *Array*, 27:100443, 2025. doi: 10.1016/j.array.2025.100443.

Microsoft Corporation. SQL Server Service Broker. `https://learn.microsoft.com/en-us/sql/database-engine/configure-windows/sql-server-service-broker`, 2025. Microsoft Learn, page dated 23 September 2025; accessed 16 November 2026.

Microsoft Corporation. Microsoft SQL Server 2022. `https://www.microsoft.com/en-us/sql-server/sql-server-2022`, 2026. Product page; accessed 14 November 2026.

Raymond R. Panko. Spreadsheet errors: What we know. What we think we can do. In *Proceedings of the Spreadsheet Risk Symposium, European Spreadsheet Risks Interest Group (EuSpRIG)*, pages 7–17, Greenwich, England, 2000. ISBN 1-86166-158-4. arXiv:0802.3457.

Ken Peffers, Tuure Tuunanen, Marcus A. Rothenberger, and Samir Chatterjee. A design science research methodology for information systems research. *Journal of Management Information Systems*, 24(3):45–77, 2007. doi: 10.2753/MIS0742-1222240302.

Dimitris N. Politis and Halbert White. Automatic block-length selection for the dependent bootstrap. *Econometric Reviews*, 23(1):53–70, 2004. doi: 10.1081/ETC-120028836.

Serhiy Y. Ponomarov and Mary C. Holcomb. Understanding the concept of supply chain resilience. *The International Journal of Logistics Management*, 20(1):124–143, 2009. doi: 10.1108/09574090910954873.

Ossama Rashad, Omneya Attallah, and Iman Morsi. A smart PLC-SCADA framework for monitoring petroleum products terminals in industry 4.0 via machine learning. *Measurement and Control*, 55(7–8):830–848, 2022. doi: 10.1177/00202940221103305.

Edward A. Silver, David F. Pyke, and Douglas J. Thomas. *Inventory and Production Management in Supply Chains*. CRC Press, Boca Raton, FL, 4th edition, 2017.

David Temoshok, James L. Fenton, Yee-Yin Choong, Naomi Lefkovitz, Andrew Regenscheid, Ryan Galluzzo, and Justin P. Richer. Digital identity guidelines: Authentication and authenticator management. NIST Special Publication 800-63B-4, National Institute of Standards and Technology, 2025.

Baba Shehu Terab, Umar Ali Umar, and Ayodeji Nathaniel Oyedeji. Information technology in supply chain management: Perspectives from key actors in the Nigerian petroleum sector. *African Journal of Science, Technology, Innovation and Development*, 15(7):955–961, 2023. doi: 10.1080/20421338.2023.2221876.

Peter Trebuna, Miriam Pekarcikova, and Matus Matiscsak. Logistic-information system based on object-oriented approach. *Acta Logistica*, 11(4):537–547, 2024. doi: 10.22306/al.v11i4.538.

U.S. Energy Information Administration. Weekly U.S. product supplied of finished motor gasoline, distillate fuel oil and kerosene-type jet fuel; weekly U.S. ending stocks of total gasoline and distillate fuel oil (series WGFUPUS2, WDIUPUS2, WKJUPUS2, WGTSTUS1, WDISTUS1). `https://www.eia.gov/dnav/pet/hist/LeafHandler.ashx?n=PET&s=WGFUPUS2&f=W`, 2026. Data to the week ending 6 November 2026; downloaded 13 November 2026.

John Venable, Jan Pries-Heje, and Richard Baskerville. FEDS: A framework for evaluation in design science research. *European Journal of Information Systems*, 25(1):77–89, 2016. doi: 10.1057/ejis.2014.36.

Monika M. Wahi, David V. Parks, Robert C. Skeate, and Steven B. Goldin. Reducing errors from the electronic transcription of data collected on paper forms: A research data case study. *Journal of the American Medical Informatics Association*, 15(3):386–389, 2008. doi: 10.1197/jamia.M2381.

Muhammad Waqas and Mohsin Jamil. Smart IoT SCADA system for hybrid power monitoring in remote natural gas pipeline control stations. *Electronics*, 13(16):3235, 2024. doi: 10.3390/electronics13163235.

Shuang Wei, Jingyi Jiang, Juan Chen, and Qiang Wang. Digital technology and supply chain resilience: A literature review and emerging themes. *International Transactions in Operational Research*, 33(4):2167–2189, 2026. doi: 10.1111/itor.70119.

Lin Wu, Zhewei Zhang, Jimmy Huang, I-Hsuan Su, and Shuang Tian. Digital technologies and supply chain resilience: A resource-action-performance perspective. *Information Systems Frontiers*, 27(5):2137–2158, 2025. doi: 10.1007/s10796-025-10595-1.