

# Northstar Analytics

## CECL Suite v4.2

### Implementation & Production Operations

### Runbook

Document ID: NA-CECL-OPS-4.2

Release Date: February 17, 2026

*Confidential - For Client Operations Use Only*

**Prepared by:** Northstar Analytics, Delivery Engineering & Platform Ops

**Primary contacts:** ops@northstaranalytics.example |  
modelsupport@northstaranalytics.example

**Target users:** Client Model Owners, Finance Close Team, IT Integration, Model Risk Management

## Document Control

Version	Date	Author	Reviewer	Summary of Changes
2.0	2025-09-18	J. Rivera	S. Khan	Initial production runbook for CECL Suite v4.0
2.1	2025-12-10	J. Rivera	S. Khan	Added rerun procedures; expanded evidence retention checklist
2.2	2026-02-17	J. Rivera	S. Khan	Updated orchestration; clarified close calendar; added DR steps

## Distribution and Use

This runbook provides operational procedures for implementing and executing the Northstar CECL Suite. It is written in a vendor operations format and is not aligned to client model documentation templates. Clients should integrate these procedures with internal close calendars, change control, and model governance requirements.

## Table of Contents (Generated in Word)

This document includes a placeholder for a Microsoft Word-generated Table of Contents. To update: References > Table of Contents > Update Table.

1. Purpose, Scope, and Operating Principles
  2. Platform Overview and System Topology
  3. Environments, Access, and Segregation of Duties
  4. Configuration Management and Release Process
  5. Close Calendar and End-to-End Run Overview
  6. Pre-Run Data Intake and Gate Controls
  7. Model Execution Procedure (Standard Run)
  8. Post-Run Checks, QC, and Reconciliations
  9. Reporting Outputs, Exports, and Distribution
  10. Exceptions, Reruns, and Adjusted Runs
  11. Incident Management and Support
  12. Backup, Business Continuity, and Disaster Recovery
  13. Audit Evidence, Retention, and Traceability
- Appendices (Checklists, Sample Logs, Job Catalog, Glossary)

## 1. Purpose, Scope, and Operating Principles

**Purpose.** Define repeatable, controlled procedures for implementing the CECL Suite and executing monthly/quarterly allowance runs with appropriate controls, evidence, and error handling.

**Scope.** Includes data ingestion, QC gating, model execution, output delivery, and evidence retention. Excludes policy decisions such as scenario weights and overlays, which remain client-governed.

### 1.1 Operating Principles

- Deterministic processing: same inputs + configuration produce identical outputs.
- Segregation of duties: separate roles for data delivery, run execution, and approval.
- Controlled change: configuration and code changes routed through release process.
- Auditability: preserve artifacts that support re-performance and review.
- Fail-safe operation: clear rerun and rollback procedures with documented rationale.

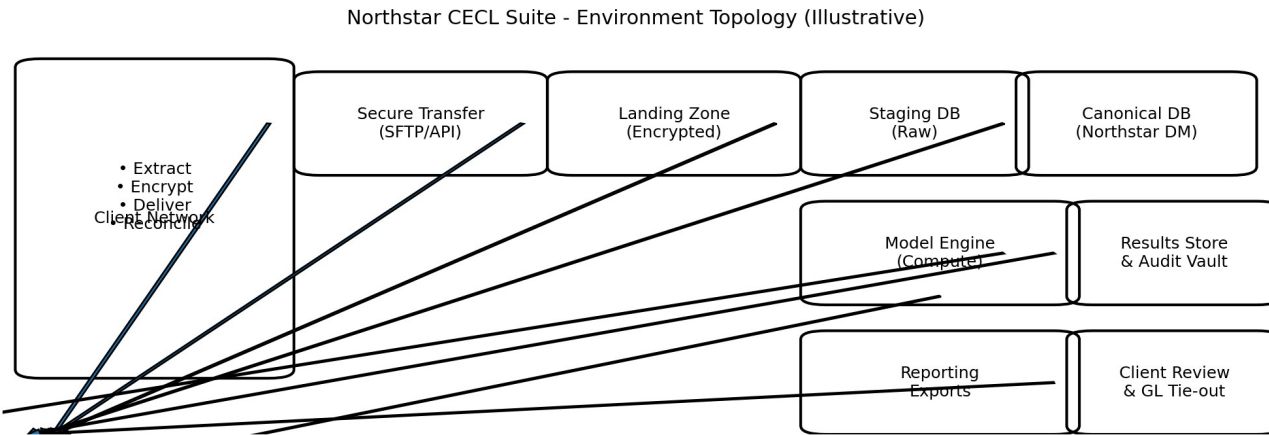
### 1.2 Roles Used in This Runbook (Vendor Terminology)

Role	Responsibilities
Run Operator	Executes the run, monitors jobs, coordinates with support; cannot approve results.
Data Steward	Owns data delivery and reconciliations; resolves data QC findings.
Finance Reviewer	Reviews outputs and reconciliations; approves reporting package.
Platform Admin	Manages access and infrastructure; does not modify model logic.
Model Support	Vendor support for defects and questions; provides hotfixes under change control.

2. Platform Overview and System Topology

The CECL Suite is deployed as a set of services: file intake, staging, canonical transformation, quality control engine, model compute engine, results store, and export/reporting layer. Implementations may be on Northstar-managed infrastructure or client-managed infrastructure depending on contracting and security requirements.

2.1 System Topology (Illustrative)



2.2 Core Components

Component	Purpose
Intake Gateway	Receives encrypted files via SFTP/API; validates package metadata; stores in landing zone.
Landing Zone	Encrypted storage location; immutable file storage with checksums.
Staging Database	Raw landing tables; minimal transformations; supports re-processing.
Canonical Database	Standardized Northstar DM tables consumed by model engine.
QC Rules Engine	Automated checks; outputs QC findings and gate status.
Model Engine	Computes ECL outputs; writes results to store.
Results Store/Audit Vault	Holds run artifacts, configuration snapshots, and outputs.

Component	Purpose
Export & Reporting	Generates client-facing files and reporting datasets.

2.3 Run Identifiers and Metadata

Each run is assigned a run\_id (e.g., 2026Q1\_SRB\_00052). Runs include metadata: as\_of\_date, scenario\_set, configuration\_version, data\_package\_id, execution timestamps, operator id, and export package checksums. Metadata is stored in the audit vault.

### 3. Environments, Access, and Segregation of Duties

The CECL Suite supports separate environments to manage change and reduce operational risk. Minimum recommended: DEV, UAT, and PROD.

#### 3.1 Environments

Environment	Use
DEV	Engineering/testing; synthetic or masked data; frequent changes
UAT	Client testing and parallel runs; approvals prior to production releases
PROD	Production close runs; controlled change; restricted access

#### 3.2 Access Controls (Typical)

Access is role-based. Write access to production configuration is limited. Run execution requires multi-factor authentication and is logged.

Permission	Run Operator	Data Steward	Finance Reviewer
Upload data package	N	Y	N
Execute model run	Y	N	N
Approve QC gate	N	Y	N
Approve final outputs	N	N	Y
Modify model configuration	N	N	N (controlled)
Access audit vault	Read	Read	Read

#### 3.3 Segregation of Duties Controls

- Run Operator cannot approve QC gates or final reporting package.
- Data Steward cannot modify production model code; can remediate and re-deliver data.
- Finance Reviewer signs off on final package and retains evidence bundle.
- Emergency access (break-glass) is logged and reviewed after-the-fact.

## 4. Configuration Management and Release Process

Configuration items include segmentation tables, scenario set definitions, model parameters (e.g., reversion horizon), export layouts, QC thresholds, and mapping metadata. Configuration is versioned and promoted through environments.

### 4.1 Configuration Artifacts (Examples)

Artifact	Description
SEGMENT_DEFINITION	Segment rules and overrides by portfolio
SCENARIO_SET	Scenario names, weights, macro series mapping
RUN_CONTROL	As-of date, run mode, toggles
QC_RULES	Rule thresholds and gate settings
EXPORT_LAYOUT	File formats and report templates

### 4.2 Release Workflow (Standard)

1. Plan: define change scope, risk, and test plan; open change ticket.
2. Build: implement change in DEV; generate release candidate.
3. Test: execute unit tests and sample run; remediate defects.
4. UAT: client runs parallel test; approves release candidate.
5. Promote: deploy to PROD during approved window; capture deployment evidence.
6. Post-deploy: perform smoke test and confirm baseline behavior.

### 4.3 Rollback Policy

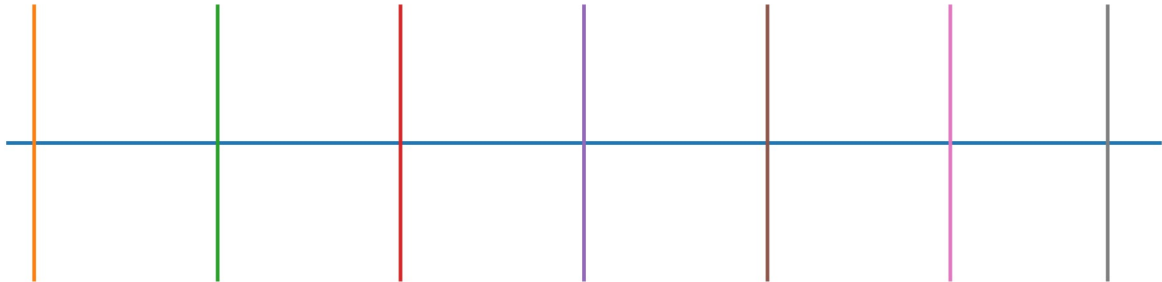
If a production deployment causes an unacceptable outcome (failed smoke test, corrupted outputs, or material unexpected change), rollback is executed by restoring the prior configuration snapshot and/or prior release package. Rollbacks are documented with root cause and follow-up actions.



## 5. Close Calendar and End-to-End Run Overview

The CECL Suite run is typically executed as part of the month-end or quarter-end close. The end-to-end process includes data extraction, delivery, QC gating, model execution, post-run checks, reporting, and archive.

### 5.1 Example Close Timeline



### 5.2 End-to-End Control Gates

Gate	Criteria
Gate A: Data Package Acceptance	Delivery complete; checksums verified; package registered.
Gate B: QC Critical Checks	Key fields present; unique keys; referential integrity; critical rules pass.
Gate C: Reconciliations	Balances and CO/recovery tie to control totals within tolerance.
Gate D: Run Completion	Model run completed successfully; outputs generated; no critical exceptions.
Gate E: Finance Sign-off	Reporting package reviewed; rollforwards and GL tie-outs approved.
Gate F: Archive	Evidence bundle sealed; inputs/config/outputs stored; read-only.

## 6. Pre-Run Data Intake and Gate Controls

### 6.1 Data Package Components

- Required: ACCOUNT\_MASTER, ACCOUNT\_SNAPSHOT, TRANSACTION (optional for some portfolios), CHARGEOFF, RECOVERY, GL\_CONTROL\_TOTALS, MACRO\_SERIES.
- Optional: COLLATERAL, SCORE, reference dimensions, and configuration lookup tables.
- Each table delivered as a file with consistent as\_of\_date and package id.

### 6.2 Intake Steps (Operator/Steward)

7. Verify expected package files are present and encrypted; confirm checksum files (if provided).
8. Register data\_package\_id and as\_of\_date in intake console.
9. Decryption and load to staging initiated; monitor job completion.
10. Run automated QC suite; review critical/high findings.
11. If critical failures, reject package and initiate remediation (see Section 10).

### 6.3 Pre-Run Checklist (Excerpt)

Item	Owner	Evidence	Completed (Y/N)
Data package delivered and registered	Data Steward	Intake console screenshot / log export	
Critical QC rules pass	Data Steward	QC report (PDF/CSV)	
Balance reconciliation within tolerance	Data Steward	Reconciliation workbook	
Scenario set selected and weights confirmed	Finance Reviewer	Scenario memo / approval	
Run controls configured (as_of_date, run mode)	Run Operator	Config snapshot export	
Change freeze confirmed (no pending deployments)	Platform Admin	Change calendar entry	

## 7. Model Execution Procedure (Standard Run)

A standard run executes the configured CECL Suite with no overlays beyond pre-approved configuration. Execution is job-orchestrated and produces an output package with deterministic identifiers.

### 7.1 Job Orchestration

Job ID	Name	Description	Success Criteria	Dependencies
JOB-100	Load & Validate	Load staging tables; validate schema; checksums	Success	None
JOB-200	Transform to Canonical	Apply mapping rules; generate Northstar DM tables	Success	JOB-100
JOB-300	QC Rules Engine	Execute QC catalog; generate gate report	Gate Pass	JOB-200
JOB-400	Model Compute	Run component models and lifetime projection	Success	JOB-300
JOB-500	Post-run QC	Output sanity checks; variance checks vs prior run	Pass/Review	JOB-400
JOB-600	Exports	Generate exports and reporting dataset	Success	JOB-500
JOB-700	Archive	Seal evidence bundle; store to audit vault	Success	JOB-600

### 7.2 Standard Run Steps

12. Confirm Gate C completion (reconciliations).
13. Set RUN\_CONTROL parameters: as\_of\_date, scenario\_set, run\_type=STANDARD, export\_profile.

14. Start JOB-400 (Model Compute) via orchestration console; record start time.
15. Monitor job status and resource utilization; capture completion time and run\_id.
16. Review JOB-500 output QC summary; route any High findings to Data Steward/Finance Reviewer.
17. If acceptable, execute JOB-600 exports; generate output package checksums.
18. Proceed to Gate E sign-off and JOB-700 archive.

### 7.3 Console Commands (Illustrative Only)

Some deployments support command-line invocation for automation. The following is illustrative and may differ by implementation.

```
ceclctl run start --as_of_date 2026-01-31 --scenario_set 2026Q1_ForecastSet_A --run_type STANDARD
```

```
ceclctl run status --run_id 2026Q1_SRB_00052
```

```
ceclctl export generate --run_id 2026Q1_SRB_00052 --profile FINANCE_CLOSE
```

## 8. Post-Run Checks, QC, and Reconciliations

### 8.1 Output Sanity Checks (Automated)

- Completeness: all expected segments present; instrument counts within tolerance vs input.
- Reasonableness: ECL totals within configured variance bands vs prior period unless explained.
- No negative ECL for assets unless explicitly allowed.
- Discount factors monotonic decreasing; term structures valid.
- Scenario weights sum to 1.0; identifiers match approved memo.

### 8.2 Variance Diagnostics (Example)

Segment	Prior ECL	Current ECL	% Change	Driver Notes (to complete)
Auto - Prime	\$12.4MM	\$12.9MM	+4.0%	Higher balance + mild PD increase
Auto - Near Prime	\$9.1MM	\$10.0MM	+9.9%	Scenario unemployment higher
Credit Card - Prime	\$18.2MM	\$17.8MM	-2.2%	Higher paydowns; utilization lower
Credit Card - Subprime	\$21.5MM	\$23.6MM	+9.8%	PD uplift; delinquency drift
HELOC	\$6.3MM	\$6.0MM	-4.8%	HPI path improved; collateral values higher

### 8.3 Reconciliations (Finance Close)

- Tie ECL totals to allowance rollforward workbook and GL posting amounts (where applicable).
- Confirm portfolio totals align to reporting hierarchy and exclude sold/held-for-sale per policy.
- Retain variance commentary and approvals as part of evidence bundle.

#### 8.4 Evidence Bundle Contents (Minimum)

Artifact	Owner	Storage Location
Input file checksums + intake logs	Run Operator	Audit Vault /inputs
QC report (critical/high findings)	Data Steward	Audit Vault /qc
Reconciliation workbook (balances, CO, recov)	Data Steward	Client repository + audit vault copy
Configuration snapshot (RUN_CONTROL, scenario set)	Run Operator	Audit Vault /config
Run metadata + job logs	Platform Admin	Audit Vault /logs
Output package checksums + exports	Run Operator	Audit Vault /outputs
Finance sign-off memo	Finance Reviewer	Client close package repository

## 9. Reporting Outputs, Exports, and Distribution

### 9.1 Standard Export Package (FINANCE\_CLOSE Profile)

File	Description
ECL_SUMMARY.csv	Portfolio/segment totals by scenario and weighted result
ECL_DETAIL.csv	Instrument-level ECL (or account-level depending on config)
ROLLFORWARD.xlsx	Allowance rollforward support tabs and checks
DRIVER_ATTRIBUTION.csv	Attribution of change vs prior run (balance, PD, LGD, scenario)
EXCEPTIONS.csv	QC exceptions and flagged accounts
RUN_METADATA.json	Configuration and audit metadata snapshot

### 9.2 Distribution Controls

- Export packages are delivered to a restricted client folder with least-privilege access.
- Emails contain no data; notifications reference package id and location only.
- Checksum file accompanies exports; recipients verify integrity before use.

## 10. Exceptions, Reruns, and Adjusted Runs

### 10.1 When to Rerun

- Critical QC failures (missing keys, duplicate snapshots, schema mismatches).
- Reconciliations outside tolerance without an approved exception.
- Material operational error identified after run (e.g., wrong scenario set selected).
- Confirmed platform defect or interrupted job resulting in partial outputs.

### 10.2 Rerun Approval Requirements

Rerun Type	Requires Approval From	Notes
Data Fix Rerun	Data Steward + Finance Reviewer	Data package re-delivered; mapping unchanged
Scenario Change Rerun	Finance Reviewer + Model Owner	Document rationale and weights
Configuration Change Rerun	Model Owner + Platform Admin	Requires change ticket and UAT unless emergency
Emergency Hotfix Rerun	Platform Admin + Model Support	Break-glass; post-incident review required

### 10.3 Rerun Procedure (Summary)

19. Open rerun ticket with root cause, impacted artifacts, and requested resolution.
20. Freeze distribution of prior outputs; mark package as superseded.
21. Deliver corrected input or configuration; capture evidence of change.
22. Execute rerun; compare outputs vs prior; document differences.
23. Publish final package; archive superseded outputs with rationale.



## 11. Incident Management and Support

### 11.1 Severity Levels (Vendor)

Severity	Definition	Typical Response	Examples
Sev 1	Production outage; close at risk	Immediate bridge; hourly updates	Jobs fail repeatedly; exports unavailable
Sev 2	Degraded service; workaround exists	Same-day response	QC engine slow; partial reports delayed
Sev 3	Minor issue; no close impact	Next business day	Cosmetic export formatting issue
Sev 4	Question/request	As scheduled	How-to guidance; access requests

### 11.2 Incident Ticket Content (Minimum)

- run\_id and as\_of\_date
- job\_id(s) affected and timestamps
- error message excerpt and log location
- impact statement (close deadline at risk?)
- steps already attempted and results

## 12. Backup, Business Continuity, and Disaster Recovery

### 12.1 Backup Strategy (Typical)

- Daily backups of configuration and metadata; weekly full backups of canonical database.
- Immutable storage for input packages and outputs (audit vault).
- Retention: inputs and outputs retained for at least 7 years unless contract specifies otherwise.

### 12.2 Disaster Recovery Procedure (Summary)

24. Declare incident and initiate DR runbook; capture incident id.
25. Restore services in secondary region/environment (if applicable).
26. Restore latest canonical DB snapshot and configuration snapshot.
27. Re-ingest last accepted data package from audit vault; validate checksums.
28. Execute smoke test; if successful, proceed with close run.
29. Document all steps and approvals; perform post-incident review.

### 13. Audit Evidence, Retention, and Traceability

Auditability is supported through immutable artifacts and deterministic identifiers. The minimum evidence bundle supports re-performance for a selected as\_of\_date and scenario set.

#### 13.1 Traceability Matrix (Illustrative)

Question	Evidence	Location	Owner
What inputs were used?	Input file checksums + intake logs	Audit Vault /inputs	Run Operator
What configuration was used?	Config snapshot (tables + version id)	Audit Vault /config	Run Operator
Did QC pass?	QC report + gate status	Audit Vault /qc	Data Steward
When did jobs run?	Job logs + timestamps	Audit Vault /logs	Platform Admin
What outputs were delivered?	Export package + checksums	Audit Vault /outputs	Run Operator
Who approved results?	Finance sign-off memo	Client close package repository	Finance Reviewer

#### 13.2 Retention and Access

Evidence bundles are retained in read-only storage. Access is limited to authorized personnel and is logged. If clients require longer retention or specific archival formats, these are defined in the statement of work.

Appendix A: Detailed Run Checklist (Printable)

- 30. Confirm close calendar date and reporting as\_of\_date.
- 31. Verify scenario package approved and stored (scenario\_set id).
- 32. Confirm data delivery complete; verify checksums.
- 33. Run QC suite; resolve all Critical findings.
- 34. Perform balance and CO/recovery reconciliation; document variances.
- 35. Lock configuration snapshot; export RUN\_CONTROL and scenario set.
- 36. Execute standard run; monitor JOB-400 and JOB-500.
- 37. Review variance diagnostics and exceptions; document drivers.
- 38. Generate exports; verify checksum and file counts.
- 39. Finance review and sign-off; archive evidence bundle.

Appendix B: Sample Run Log (Excerpt)

Run ID: 2026Q1\_SRB\_00052

As-of Date: 2026-01-31

Scenario Set: 2026Q1\_ForecastSet\_A (Baseline 60%, Adverse 30%, Severe 10%)

Data Package ID: SRB\_2026-01-31\_PKG\_00018

Operator: op\_0147

Start Time: 2026-02-03T02:15:09Z

End Time: 2026-02-03T04:42:51Z

Status: SUCCESS

Notes: One High QC finding (stale collateral >18mo) accepted with documented exception  
EXC-2026-02-03-01

Appendix C: Glossary (Operational)

Term	Definition
as_of_date	Reporting cutoff date used for snapshots and allowance run.
data_package_id	Identifier for a delivered set of input files.
gate	Control point requiring review/approval before proceeding.

Term	Definition
orchestration console	User interface for running and monitoring jobs.
audit vault	Immutable storage for inputs, outputs, configuration, and logs.