Knowledge Base — CNTR/VS/EA

**General Guidelines**

1. Execute steps in order. Sub-steps appear as 1.1, 1.2 where deeper action is required.

2. Prefer re-sync/replay workflows over manual edits unless SOP explicitly allows it.

3. Capture evidence (before/after) and update the case with sanitized details.

**CNTR: Trying to create Container Range From CONTAINER\_ID to BSIU 323099, but we are hit with Container Range Error**

|  |  |
| --- | --- |
| Module | Container Report |

**Overview**

We are attempting to create a container range from CONTAINER\_ID to CONTAINER\_ID, but encountered an error: 'Overlapping container range(s) found.' However, the specified range could not be located. A query for the range from CONTAINER\_ID to CONTAINER\_ID returned a different range. The concern is that the range isn't visible in the system, despite being created by the relevant process. A search on our end also returned a different range. Why is this discrepancy not visible on their end?

**Resolution**

1. How to check container parameter range and whether it overlap?

2. System check the serial number range excluding Check Digit

3. For this case, the serial number overlap. BSIU 3430001

4. Confirm scope and reproduction on a safe test entity.

5. Check recent deployments/config toggles around the timestamp.

6. Apply compliant fix and document the change.

6.1 Capture before/after evidence (screenshots/queries).

6.2 Update the case with sanitized details.

**Verification**

1. Run the end-to-end journey again; confirm success.

2. No new errors for 30 minutes in monitoring.

3. Attach evidence and close the case.

**CNTR: Discrepancy between customer portal and TOS for container CM**

|  |  |
| --- | --- |
| Module | Container Booking |

**Overview**

Discrepancy between the customer portal and TOS for container CONTAINER\_ID on voyage VOYAGE\_ID. The portal shows 'Loaded @ LOCATION' while TOS indicates 'In Yard - BLOCK'. An audit trail reveals a late yard move event, possibly overwritten by an older milestone. Booking and seal information (placeholders) are included. The last move was handled by YARD\_CRANE at LOCATION.

**Resolution**

1. Investigate Data Discrepancy:

1.1 Executed Authentication & Token Refresh to resolve the issue.

1.2 Ran cross-system comparison and invalidated cache to synchronize data.

2. Ran cross-system compare and forced cache invalidation.

3. Re-ordered events by eventTime; replayed 'Yard Move Completed'.

4. Corrected size/type to manifest truth and locked the attribute for 6 hours.

5. Published reconciliation summary with sanitized diffs.

6. Confirm scope and reproduction on a safe test entity.

7. Check recent deployments/config toggles around the timestamp.

8. Apply compliant fix and document the change.

8.1 Capture before/after evidence (screenshots/queries).

8.2 Update the case with sanitized details.

**Verification**

1. Run the end-to-end journey again; confirm success.

2. No new errors for 30 minutes in monitoring.

3. Attach evidence and close the case.

**CNTR: Translator rejected EDIFACT COARRI for CONTAINER\_ID: segment count mismatch.**

|  |  |
| --- | --- |
| Module | Container Report |

**Overview**

Translator rejected EDIFACT COARRI for CONTAINER\_ID: segment count mismatch, sequence skipped 018→019→021. Missing public milestone though warehouse shows correct move.

**Resolution**

1. Engaged partner; shared sanitized samples; added pre-ingest normalizer for qualifiers; updated monitoring for repeated segment anomalies.

1.1 Run schema validator (segments/qualifiers/max-occurrence).

1.2 Fix qualifier mapping or split oversized occurrences.

2. Locate the problematic EDI message/file and quarantine it.

3. Validate structure and partner-specific rules.

3.1 Run schema validator (segments/qualifiers/max-occurrence).

3.2 Fix qualifier mapping or split oversized occurrences.

4. Correct normalization/mapping and reprocess safely.

4.1 Use idempotency token to prevent duplicates.

4.2 Re-ingest from quarantine and monitor translator logs.

5. Verify downstream sequencing and deduplication.

**Verification**

1. Confirm next run processed successfully; no duplicates.

2. Verify partner ACK/CONTRL is archived.

3. Attach validator logs and sanitized sample.

**CNTR: Discrepancy between customer portal and TOS for CONTAINER\_ID on voyage VA61**

|  |  |
| --- | --- |
| Module | Container Booking |

**Overview**

Discrepancy between the customer portal and TOS for container CONTAINER\_ID on voyage VOYAGE\_ID. The portal shows 'Loaded @ LOCATION' while TOS indicates 'In Yard - BLOCK'. An audit trail shows a late yard move event, possibly overwritten by an older milestone. Booking and seal information (placeholders) are included. The last move was handled by YARD\_CRANE at LOCATION.

**Resolution**

1. Investigate Data Discrepancy:

1.1 Executed Duplicate Message De-duplication to resolve the issue.

1.2 Ran cross-system comparison and invalidated cache to synchronize data.

2. Ran cross-system compare and forced cache invalidation.

3. Re-ordered events by eventTime; replayed 'Yard Move Completed'.

4. Corrected size/type to manifest truth and locked the attribute for 6 hours.

5. Published reconciliation summary with sanitized diffs.

6. Confirm scope and reproduction on a safe test entity.

7. Check recent deployments/config toggles around the timestamp.

8. Apply compliant fix and document the change.

8.1 Capture before/after evidence (screenshots/queries).

8.2 Update the case with sanitized details.

**Verification**

1. Run the end-to-end journey again; confirm success.

2. No new errors for 30 minutes in monitoring.

3. Attach evidence and close the case.

**CNTR: 500 responses and timeouts on '/yard/locations' when querying CONTAINER\_ID**

|  |  |
| --- | --- |
| Module | Container Report |

**Overview**

500 responses and timeouts on '/yard/locations' when querying CONTAINER\_ID. slowest 5% of requests latency > 10s; retries exhausted—gaps visible in customer timeline. Coincides with partner load test near PSA Keppel.

**Resolution**

1. Introduced token-leeway (+60s); purged problematic cache entries; rebuilt hot keys.

1.1 Check DB slowest 5% of requests latency; look for lock waits.

1.2 Purge problematic cache keys; warm hot keys if needed.

2. Shaped traffic during partner tests and published sanitized RCA.

3. Reproduce the failing call on a safe test entity.

3.1 Use read-only token; capture status code and latency.

3.2 Record request ID/correlation ID for log search.

4. Check API gateway/service health around the incident window.

5. Review auth and permissions for the calling user/client.

5.1 Refresh/rotate token; confirm scopes.

5.2 Verify NTP/clock skew; resync if >2 seconds.

**Verification**

1. Re-run the endpoint; expect 2xx and correct JSON fields.

2. Observe error rate and latency for 30 minutes; under baseline.

3. Attach request/response evidence to the case.

**CNTR: Weight and size attributes for CONTAINER\_ID do not align with manifest data**

|  |  |
| --- | --- |
| Module | Container Booking |

**Overview**

Weight and size attributes for CONTAINER\_ID do not align with manifest data. OMS lists 22G1 whereas the EDI intake translated to 45R1—slot planning at PSA Pasir Panjang misallocated. Voyage QS19; last known location Block B3. Suspect duplicate CODECO from yesterday caused field overwrite.

**Resolution**

1. Ran de-duplication on EDI queue and restored the correct CODECO snapshot.

2. Set precedence to prefer TOS for equipment attributes; added a rule to reject backdated updates within 120 minutes.

2.1 Archive ACK with timestamp and execution ID.

2.2 Notify partner with sanitized sample if needed.

3. Customer notified using a masked ticket reference.

4. Locate the problematic EDI message/file and quarantine it.

5. Validate structure and partner-specific rules.

5.1 Run schema validator (segments/qualifiers/max-occurrence).

5.2 Fix qualifier mapping or split oversized occurrences.

6. Correct normalization/mapping and reprocess safely.

6.1 Use idempotency token to prevent duplicates.

6.2 Re-ingest from quarantine and monitor translator logs.

**Verification**

1. Confirm next run processed successfully; no duplicates.

2. Verify partner ACK/CONTRL is archived.

3. Attach validator logs and sanitized sample.

**CNTR: Discrepancy between customer portal and TOS for container CONTAINER\_ID on voyage V001**

|  |  |
| --- | --- |
| Module | Container Booking |

**Overview**

Discrepancy between customer portal and TOS for container CONTAINER\_ID on voyage V001. Portal shows 'Loaded @ BP02' while TOS indicates 'In Yard - Block J4'. Audit trail reveals a late yard move event, potentially overwritten by an older milestone. Booking BK-CC5O7UUB and seal S2343-PVS included (randomized placeholders). Yard crane QC13 handled the last move at PSA Tanjong Pagar.

**Resolution**

1. Fixed API mapping for party roles and flushed CDN.

1.1 Capture before/after evidence (screenshots/queries).

1.2 Update the case with sanitized details.

2. Repaired affected records (voyage V001); set monitors for attribute flip-flop bursts.

3. All identifiers are randomized placeholders.

4. Confirm scope and reproduction on a safe test entity.

5. Check recent deployments/config toggles around the timestamp.

6. Apply compliant fix and document the change.

6.1 Capture before/after evidence (screenshots/queries).

6.2 Update the case with sanitized details.

**Verification**

1. Run the end-to-end journey again; confirm success.

2. No new errors for 30 minutes in monitoring.

3. Attach evidence and close the case.

**CNTR: Reefer CONTAINER\_ID telemetry gap**

|  |  |
| --- | --- |
| Module | Container Booking |

**Overview**

Reefer CONTAINER\_ID telemetry gap. Temperature feed offline briefly; backfill later within acceptable range. Yard block G7, handled by YC22.

**Resolution**

1. Reset telemetry bridge; enabled store-and-forward; alert for >5m gaps; confirmed stability post-recovery; tagged for periodic review.

2. Confirm scope and reproduction on a safe test entity.

3. Check recent deployments/config toggles around the timestamp.

4. Apply compliant fix and document the change.

4.1 Capture before/after evidence (screenshots/queries).

4.2 Update the case with sanitized details.

5. Validate end-to-end user flow.

5.1 Capture before/after evidence (screenshots/queries).

5.2 Update the case with sanitized details.

**Verification**

1. Run the end-to-end journey again; confirm success.

2. No new errors for 30 minutes in monitoring.

3. Attach evidence and close the case.

**CNTR: Auth token rejection for CONTAINER\_ID on '/containers/IRE8QJ/events'**

|  |  |
| --- | --- |
| Module | Container Report |

**Overview**

Auth token rejection for CONTAINER\_ID on '/containers/IRE8QJ/events'. Node clock skew surfacing as intermittent 401/403; partial data for ~18 minutes prior to mitigation.

**Resolution**

1. Introduced token-leeway (+60s); purged problematic cache entries; rebuilt hot keys.

1.1 Check DB slowest 5% of requests latency; look for lock waits.

1.2 Purge problematic cache keys; warm hot keys if needed.

2. Shaped traffic during partner tests and published sanitized RCA.

3. Reproduce the failing call on a safe test entity.

3.1 Use read-only token; capture status code and latency.

3.2 Record request ID/correlation ID for log search.

4. Check API gateway/service health around the incident window.

5. Review auth and permissions for the calling user/client.

5.1 Refresh/rotate token; confirm scopes.

5.2 Verify NTP/clock skew; resync if >2 seconds.

**Verification**

1. Re-run the endpoint; expect 2xx and correct JSON fields.

2. Observe error rate and latency for 30 minutes; under baseline.

3. Attach request/response evidence to the case.

**CNTR: Duplicate ANSI X12 214 received with conflicting**   
**eventTime for EMCU4728808CONTAINER\_ID**

|  |  |
| --- | --- |
| Module | Container Report |

**Overview**

Duplicate ANSI X12 214 received with conflicting eventTime for EMCU4728808CONTAINER\_ID. Later message had older timestamp; last-writer-wins regressed status

**Resolution**

1. Engaged partner; shared sanitized samples; added pre-ingest normalizer for qualifiers;   
updated monitoring for repeated segment anomalies.

1.1 Run schema validator (segments/qualifiers/max-occurrence).

1.2 Fix qualifier mapping or split oversized occurrences.

2. Locate the problematic EDI message/file and quarantine it.

3. Validate structure and partner-specific rules.

3.1 Run schema validator (segments/qualifiers/max-occurrence).

3.2 Fix qualifier mapping or split oversized occurrences.

4. Correct normalization/mapping and reprocess safely.

4.1 Use idempotency token to prevent duplicates.

4.2 Re-ingest from quarantine and monitor translator logs.

5. Verify downstream sequencing and deduplication.

**Verification**

1. Confirm next run processed successfully; no duplicates

2. Verify partner ACK/CONTRL is archived.

3. Attach validator logs and sanitized sample.

**CNTR: Mismatched shipment party codes for CONTAINER\_ID**

|  |  |
| --- | --- |
| Module | Container Booking |

**Overview**

Mismatched shipment party codes for CONTAINER\_ID. Shipper/consignee swapped after a backfill job. Customer raised billing risk; TOS remains accurate but public API cached the wrong roles. Affected terminal: Tuas Port; booking BK-KMKZEBBK.

**Resolution**

1. Ran de-duplication on EDI queue and restored the correct CODECO snapshot.

1.1 Check DB slowest 5% of requests latency; look for lock waits.

1.2 Purge problematic cache keys; warm hot keys if needed.

2. Set precedence to prefer TOS for equipment attributes; added a rule to reject backdated updates within 120 minutes.

3. Customer notified using a masked ticket reference.

4. Reproduce the failing call on a safe test entity.

4.1 Use read-only token; capture status code and latency.

4.2 Record request ID/correlation ID for log search.

5. Check API gateway/service health around the incident window.

6. Review auth and permissions for the calling user/client.

6.1 Refresh/rotate token; confirm scopes.

6.2 Verify NTP/clock skew; resync if >2 seconds.

**Verification**

1. Re-run the endpoint; expect 2xx and correct JSON fields.

2. Observe error rate and latency for 30 minutes; under baseline.

3. Attach request/response evidence to the case.

**CNTR: 403 responses and timeouts on '/events/subscribe' when querying CONTAINER\_ID**

|  |  |
| --- | --- |
| Module | Container Report |

**Overview**

403 responses and timeouts on '/events/subscribe' when querying CONTAINER\_ID. High latency (slowest 5% of requests > 10s) and exhausted retries resulted in gaps in the customer timeline. This coincided with partner load testing near LOCATION.

**Resolution**

1. Rolled back gateway buffering; enabled keep-alive upstream.

2. Raised SLI alerts for error-rate/latency with auto-page.

3. Coordinated cipher suite update with partner.

4. Reproduce the failing call on a safe test entity.

4.1 Use read-only token; capture status code and latency.

4.2 Record request ID/correlation ID for log search.

5. Check API gateway/service health around the incident window.

6. Review auth and permissions for the calling user/client.

6.1 Refresh/rotate token; confirm scopes.

6.2 Verify NTP/clock skew; resync if >2 seconds.

**Verification**

1. Re-run the endpoint; expect 2xx and correct JSON fields.

2. Observe error rate and latency for 30 minutes; under baseline.

3. Attach request/response evidence to the case.

**CNTR: Auth token rejection for CONTAINER\_ID on '/auth/token'**

|  |  |
| --- | --- |
| Module | Container Booking |

**Overview**

Auth token rejection for CONTAINER\_ID on '/auth/token'. Node clock skew surfacing as intermittent 401/403; partial data for ~18 minutes prior to mitigation.

**Resolution**

1. Investigate Auth Token Rejection:

1.1 Applied Data Mismatch Escalation Workflow to resolve token rejection.

1.2 Tuned retry backoff, scaled read replicas, and synchronized NTP to mitigate intermittent 401/403 errors.

2. Tuned retry backoff and timeouts; scaled read replicas; enabled request coalescing.

3. Rotated OAuth keys and hard-synced NTP; added idempotency on callbacks.

3.1 Refresh/rotate token; confirm scopes.

3.2 Verify NTP/clock skew; resync if >2 seconds.

4. Reproduce the failing call on a safe test entity.

4.1 Use read-only token; capture status code and latency.

4.2 Record request ID/correlation ID for log search.

5. Check API gateway/service health around the incident window.

6. Review auth and permissions for the calling user/client.

6.1 Refresh/rotate token; confirm scopes.

6.2 Verify NTP/clock skew; resync if >2 seconds.

**Verification**

1. Re-run the endpoint; expect 2xx and correct JSON fields.

2. Observe error rate and latency for 30 minutes; under baseline.

3. Attach request/response evidence to the case.

**CNTR: OCR misread of container ID in the gate system, triggering a lane hold**

|  |  |
| --- | --- |
| Module | Container Report |

**Overview**

Gate OCR misread ID for CONTAINER\_ID, triggering a lane hold. Seal verification (placeholders) showed a mismatch with the manifest, causing a delayed dispatch. Queue was built at LOCATION.

**Resolution**

1. Investigate OCR Misread:

1.1 Manual verification and applied Duplicate Message De-duplication.

1.2 Corrected seal entry, routed OCR misreads to manual validation, and retrained OCR with low-light samples.

2. Corrected seal entry; routed doubtful OCR reads to manual validation for 24h; retrained OCR with low-light samples; issued ops bulletin.

3. Compare authoritative sources and capture the exact delta.

3.1 Check status/location/size-type fields.

3.2 Capture last-updated timestamps from each system.

4. Check event sequencing; identify missing/late/out-of-order events.

4.1 Sort milestones by eventTime.

5. Trigger a targeted re-sync or corrective update per SOP.

5.1 Sort milestones by eventTime.

5.2 Replay missing yard move/completion if allowed by SOP.

**Verification**

1. Re-open both systems and confirm fields align (status/location/size-type).

2. Ensure event ordering is correct; no stale updates reappear.

3. Attach before/after screenshots or queries.

**CNTR: Discrepancy between customer portal and TOS for container CONTAINER\_ID on voyage VA61**

|  |  |
| --- | --- |
| Module | Container Booking |

**Overview**

Discrepancy between customer portal and TOS for container CONTAINER\_ID on voyage VOYAGE\_ID. The portal shows 'Loaded @ LOCATION' while TOS indicates 'In Yard - BLOCK'. An audit trail reveals a late yard move event, possibly overwritten by an older milestone. Booking and seal information (placeholders) are included. The last move was handled by YARD\_CRANE at LOCATION.

**Resolution**

1. Investigate Data Discrepancy:

1.1 Executed Manual Override & Audit Trail to resolve the discrepancy.

1.2 Ran cross-system comparison and invalidated cache to synchronize data.

2. Ran cross-system compare and forced cache invalidation.

3. Re-ordered events by eventTime; replayed 'Yard Move Completed'.

4. Corrected size/type to manifest truth and locked the attribute for 6 hours.

5. Published reconciliation summary with sanitized diffs.

6. Confirm scope and reproduction on a safe test entity.

7. Check recent deployments/config toggles around the timestamp.

8. Apply compliant fix and document the change.

8.1 Capture before/after evidence (screenshots/queries).

8.2 Update the case with sanitized details.

**Verification**

1. Run the end-to-end journey again; confirm success.

2. No new errors for 30 minutes in monitoring.

3. Attach evidence and close the case.

**CNTR: Translator rejected EDIFACT COARRI for CONTAINER\_ID: segment count mismatch, sequence skipped**

|  |  |
| --- | --- |
| Module | Container Report |

**Overview**

Translator rejected EDIFACT COARRI for CONTAINER\_ID: segment count mismatch, sequence skipped 018→019→021. Missing public milestone though warehouse shows correct move.

**Resolution**

1. Enabled conflict resolution preferring max(eventTime); replayed from quarantine; validated against TOS/warehouse scans; added anomaly detection for >30m skews.

1.1 Run schema validator (segments/qualifiers/max-occurrence).

1.2 Fix qualifier mapping or split oversized occurrences.

2. Locate the problematic EDI message/file and quarantine it.

3. Validate structure and partner-specific rules.

3.1 Run schema validator (segments/qualifiers/max-occurrence).

3.2 Fix qualifier mapping or split oversized occurrences.

4. Correct normalization/mapping and reprocess safely.

4.1 Use idempotency token to prevent duplicates.

4.2 Re-ingest from quarantine and monitor translator logs.

5. Verify downstream sequencing and deduplication.

**Verification**

1. Confirm next run processed successfully; no duplicates.

2. Verify partner ACK/CONTRL is archived.

3. Attach validator logs and sanitized sample.

**CNTR: Callback delivery failures for CONTAINER\_ID**

|  |  |
| --- | --- |
| Module | Container Booking |

**Overview**

Callback delivery failures for CONTAINER\_ID. Webhook sender faced 502 due to peer TLS renegotiation. Queue grew and breaker opened. Affected endpoint '/bookings/search'.

**Resolution**

1. Investigate Callback Delivery Failure:

1.1 Applied API Timeout & Retry Policy to mitigate 502 errors.

1.2 Tuned retry backoff, scaled read replicas, and enabled request coalescing to reduce latency and improve reliability.

2. Tuned retry backoff and timeouts; scaled read replicas; enabled request coalescing.

3. Rotated OAuth keys and hard-synced NTP; added idempotency on callbacks.

3.1 Refresh/rotate token; confirm scopes.

3.2 Verify NTP/clock skew; resync if >2 seconds.

4. Reproduce the failing call on a safe test entity.

4.1 Use read-only token; capture status code and latency.

4.2 Record request ID/correlation ID for log search.

5. Check API gateway/service health around the incident window.

6. Review auth and permissions for the calling user/client.

6.1 Refresh/rotate token; confirm scopes.

6.2 Verify NTP/clock skew; resync if >2 seconds.

**Verification**

1. Re-run the endpoint; expect 2xx and correct JSON fields.

2. Observe error rate and latency for 30 minutes; under baseline.

3. Attach request/response evidence to the case.

**CNTR: Callback delivery failures for CONTAINER\_ID**

|  |  |
| --- | --- |
| Module | Container Booking |

**Overview**

Callback delivery failures for CONTAINER\_ID. Webhook sender faced 504 due to peer TLS renegotiation. Queue grew and breaker opened. Affected endpoint '/auth/token'.

**Resolution**

1. Investigate Callback Delivery Failure:

1.1 Applied Manual Override & Audit Trail to address callback issues.

1.2 Tuned retry backoff, scaled read replicas, and enabled request coalescing to mitigate 504 errors.

2. Tuned retry backoff and timeouts; scaled read replicas; enabled request coalescing.

3. Rotated OAuth keys and hard-synced NTP; added idempotency on callbacks.

3.1 Refresh/rotate token; confirm scopes.

3.2 Verify NTP/clock skew; resync if >2 seconds.

4. Reproduce the failing call on a safe test entity.

4.1 Use read-only token; capture status code and latency.

4.2 Record request ID/correlation ID for log search.

5. Check API gateway/service health around the incident window.

6. Review auth and permissions for the calling user/client.

6.1 Refresh/rotate token; confirm scopes.

6.2 Verify NTP/clock skew; resync if >2 seconds.

**Verification**

1. Re-run the endpoint; expect 2xx and correct JSON fields.

2. Observe error rate and latency for 30 minutes; under baseline.

3. Attach request/response evidence to the case.

**CNTR: Inbound EDIFACT IFTSTA failed schema validation: unexpected qualifier in EQD for CONTAINER\_ID**

|  |  |
| --- | --- |
| Module | Container Booking |

**Overview**

Inbound EDIFACT IFTSTA failed schema validation: unexpected qualifier in EQD for CONTAINER\_ID. Downstream event not created; timeline incomplete.

**Resolution**

1. Enabled conflict resolution preferring max(eventTime); replayed from quarantine; validated against TOS/warehouse scans; added anomaly detection for >30m skews.

2. Reproduce the failing call on a safe test entity.

2.1 Use read-only token; capture status code and latency.

2.2 Record request ID/correlation ID for log search.

3. Check API gateway/service health around the incident window.

4. Review auth and permissions for the calling user/client.

4.1 Refresh/rotate token; confirm scopes.

4.2 Verify NTP/clock skew; resync if >2 seconds.

5. Check upstream dependencies (DB/cache/queue) for latency or errors.

5.1 Check DB slowest 5% of requests latency; look for lock waits.

5.2 Purge problematic cache keys; warm hot keys if needed.

**Verification**

1. Re-run the endpoint; expect 2xx and correct JSON fields.

2. Observe error rate and latency for 30 minutes; under baseline.

3. Attach request/response evidence to the case.

**CNTR: Mismatched shipment party codes for CONTAINER\_ID**

|  |  |
| --- | --- |
| Module | Container Booking |

**Overview**

Mismatched shipment party codes for CONTAINER\_ID. Shipper/consignee swapped after a backfill job. Customer raised billing risk; TOS remains accurate but public API cached the wrong roles. Affected terminal: LOCATION; booking BOOKING\_NUMBER.

**Resolution**

1. Investigate Mismatched Shipment Party Codes:

1.1 Executed Customer Notification & RCA Communications to address the issue.

1.2 Ran cross-system comparison, invalidated cache, and purged problematic cache keys to synchronize data.

2. Ran cross-system compare and forced cache invalidation.

2.1 Check DB slowest 5% of requests latency; look for lock waits.

2.2 Purge problematic cache keys; warm hot keys if needed.

3. Re-ordered events by eventTime; replayed 'Yard Move Completed'.

4. Corrected size/type to manifest truth and locked the attribute for 6 hours.

5. Published reconciliation summary with sanitized diffs.

6. Reproduce the failing call on a safe test entity.

6.1 Use read-only token; capture status code and latency.

6.2 Record request ID/correlation ID for log search.

7. Check API gateway/service health around the incident window.

8. Review auth and permissions for the calling user/client.

8.1 Refresh/rotate token; confirm scopes.

8.2 Verify NTP/clock skew; resync if >2 seconds.

**Verification**

1. Re-run the endpoint; expect 2xx and correct JSON fields.

2. Observe error rate and latency for 30 minutes; under baseline.

3. Attach request/response evidence to the case.

**CNTR: Auth token rejection for CONTAINER\_ID on '/bookings/search'**

|  |  |
| --- | --- |
| Module | Container Report |

**Overview**

Auth token rejection for CONTAINER\_ID on '/bookings/search'. Node clock skew surfacing as intermittent 401/403; partial data for ~18 minutes prior to mitigation..

**Resolution**

1. Rolled back gateway buffering; enabled keep-alive upstream.

2. Raised SLI alerts for error-rate/latency with auto-page.

3. Coordinated cipher suite update with partner.

4. Reproduce the failing call on a safe test entity.

4.1 Use read-only token; capture status code and latency.

4.2 Record request ID/correlation ID for log search.

5. Check API gateway/service health around the incident window.

6. Review auth and permissions for the calling user/client.

6.1 Refresh/rotate token; confirm scopes.

6.2 Verify NTP/clock skew; resync if >2 seconds.

**Verification**

1. Re-run the endpoint; expect 2xx and correct JSON fields.

2. Observe error rate and latency for 30 minutes; under baseline.

3. Attach request/response evidence to the case

**CNTR: Translator rejected EDIFACT IFTSTA for CONTAINER\_ID: segment count mismatch**

|  |  |
| --- | --- |
| Module | Container Report |

**Overview**

Translator rejected EDIFACT IFTSTA for CONTAINER\_ID: segment count mismatch, sequence skipped 018→019→021. Missing public milestone though warehouse shows correct move.

**Resolution**

1. Enabled conflict resolution preferring max(eventTime); replayed from quarantine; validated against TOS/warehouse scans; added anomaly detection for >30m skews.

1.1 Run schema validator (segments/qualifiers/max-occurrence).

1.2 Fix qualifier mapping or split oversized occurrences.

2. Locate the problematic EDI message/file and quarantine it.

3. Validate structure and partner-specific rules.

3.1 Run schema validator (segments/qualifiers/max-occurrence).

3.2 Fix qualifier mapping or split oversized occurrences.

4. Correct normalization/mapping and reprocess safely.

4.1 Use idempotency token to prevent duplicates.

4.2 Re-ingest from quarantine and monitor translator logs.

5. Verify downstream sequencing and deduplication.

**Verification**

1. Confirm next run processed successfully; no duplicates.

2. Verify partner ACK/CONTRL is archived.

3. Attach validator logs and sanitized sample.

**CNTR: Mismatched shipment party codes for CONTAINER\_ID**

|  |  |
| --- | --- |
| Module | Container Booking |

**Overview**

Mismatched shipment party codes for CONTAINER\_ID. Shipper/consignee swapped after a backfill job. Customer raised billing risk; TOS remains accurate but public API cached the wrong roles. Affected terminal: LOCATION; booking BOOKING\_NUMBER.

**Resolution**

1. Fixed API mapping for party roles and flushed CDN.

2. Repaired affected records (voyage ZX77); set monitors for attribute flip-flop bursts.

3. All identifiers are randomized placeholders.

4. Reproduce the failing call on a safe test entity.

4.1 Use read-only token; capture status code and latency.

4.2 Record request ID/correlation ID for log search.

5. Check API gateway/service health around the incident window.

6. Review auth and permissions for the calling user/client.

6.1 Refresh/rotate token; confirm scopes.

6.2 Verify NTP/clock skew; resync if >2 seconds.

**Verification**

1. Re-run the endpoint; expect 2xx and correct JSON fields.

2. Observe error rate and latency for 30 minutes; under baseline.

3. Attach request/response evidence to the case.

**VSL: HI Team, Similar with previous case: TICKET\_NUMBER where we have duplicate Abbreviated Vessel Name.**

|  |  |
| --- | --- |
| Module | Vessel |

**Overview**

Case: TICKET\_NUMBER where we have duplicate Abbreviated Vessel Name.  
 Please assist to change Abbreviated Vessel Name for new VESSEL\_NUMBER with callsign CALLSIGN\_ID to be VESSEL\_NAME\_SHORT i/o VESSEL\_NAME\_LONG to avoid same name with callsign CALLSIGN\_ID2 as showing error on below screenshot while creating new Vessel Advice.

**Resolution**

1. SQLB patch and Please ask user to retry

2. Confirm scope and reproduction on a safe test entity.

3. Check recent deployments/config toggles around the timestamp.

4. Apply compliant fix and document the change.

4.1 Capture before/after evidence (screenshots/queries).

4.2 Update the case with sanitized details.

5. Validate end-to-end user flow.

5.1 Capture before/after evidence (screenshots/queries).

5.2 Update the case with sanitized details.

**Verification**

1. Run the end-to-end journey again; confirm success.

2. No new errors for 30 minutes in monitoring.

3. Attach evidence and close the case.

**VSL: BAPLIE inconsistency for VESSEL\_ID: COARRI shows discharge completed for bay BAY\_NUMBER, but BAPLIE still lists units in those slots. Older timestamp regressed the plan**

|  |  |
| --- | --- |
| Module | Vessel |

**Overview**

BAPLIE inconsistency for VESSEL\_ID: COARRI shows discharge completed for bay BAY\_NUMBER, but BAPLIE still lists units in those slots. Older timestamp regressed the plan.

**Resolution**

1. Investigate Timestamp Discrepancy

1.1 Check timestamps across BAPLIE, stowage plan, and discharge records.

1.2 Identify where older timestamps are regressing newer discharge data.

2. Enabled conflict resolution preferring max(eventTime); reprocessed from quarantine; verified stowage/discharge alignment across systems.

2.1 Use idempotency token to prevent duplicates.

2.2 Re-ingest from quarantine and monitor translator logs.

3. Communications used masked ticket references where applicable.

4. Locate the problematic EDI message/file and quarantine it.

5. Validate structure and partner-specific rules.

5.1 Run schema validator (segments/qualifiers/max-occurrence).

5.2 Fix qualifier mapping or split oversized occurrences.

6. Correct normalization/mapping and reprocess safely.

6.1 Use idempotency token to prevent duplicates.

6.2 Re-ingest from quarantine and monitor translator logs.

**Verification**

1. Confirm next run processed successfully; no duplicates.

2. Verify partner ACK/CONTRL is archived.

3. Attach validator logs and sanitized sample.

**VSL: ETA/ETB mismatch for VESSEL\_ID on voyage VOYAGE\_ID**

|  |  |
| --- | --- |
| Module | Vessel |

**Overview**

ETA/ETB mismatch for VESSEL\_ID on voyage VOYAGE\_ID. Berth plan shows ETB at LOCATION while TOS differs by >90 minutes due to tidal constraints. Partner portal still displays outdated ETA; yard resource plan at LOCATION reflects older figures. Pilotage request pending; quay CRANE\_LOCATION crane allocation misaligned.

**Resolution**

1. Investigate ETA/ETB Mismatch:

1.1 Check ETA/ETB discrepancies between berth plan and TOS.

2. Reconciled ETA/ETB across systems; normalized times to port TZ and re-published.

3. Triggered berth plan recalculation and crane profile refresh.

4. Added drift guard to auto-alert on >30 min divergence.

5. Communications used masked ticket references where applicable.

6. Compare authoritative sources and capture the exact delta.

6.1 Check status/location/size-type fields.

6.2 Capture last-updated timestamps from each system.

7. Check event sequencing; identify missing/late/out-of-order events.

8. Trigger a targeted re-sync or corrective update per SOP.

8.1 Sort milestones by eventTime.

8.2 Replay missing yard move/completion if allowed by SOP.

**Verification**

1. Re-open both systems and confirm fields align (status/location/size-type).

2. Ensure event ordering is correct; no stale updates reappear.

3. Attach before/after screenshots or queries.

**VSL: Schedule API intermittently returns 403 for VESSEL\_ID; slowest 5% of requests latency > 12s**

|  |  |
| --- | --- |
| Module | Vessel |

**Overview**

Schedule API intermittently returns 403 for VESSEL\_ID; slowest 5% of requests latency > 12s. External subscribers missed webhook pushes; fallback polling exposed gaps in berth window at LOCATION.

**Resolution**

1. Investigate API 403 Failure:

1.1 Check the Schedule API for intermittent 403 errors and high latency.

1.2 Increase timeout, tune backoff settings, and enable request coalescing.

2. Increased timeout, tuned backoff, enabled request coalescing.

3. Rotated OAuth keys; enforced clock-skew tolerance; added SLI alerts for error-rate/latency.

3.1 Refresh/rotate token; confirm scopes.

3.2 Verify NTP/clock skew; resync if >2 seconds.

4. Communications used masked ticket references where applicable.

5. Reproduce the failing call on a safe test entity.

5.1 Use read-only token; capture status code and latency.

5.2 Record request ID/correlation ID for log search.

6. Check API gateway/service health around the incident window.

7. Review auth and permissions for the calling user/client.

7.1 Refresh/rotate token; confirm scopes.

7.2 Verify NTP/clock skew; resync if >2 seconds.

**Verification**

1. Re-run the endpoint; expect 2xx and correct JSON fields.

2. Observe error rate and latency for 30 minutes; under baseline.

3. Attach request/response evidence to the case.

**VSL: Schedule API intermittently returns 500 for VESSEL\_ID; slowest 5% of requests latency > 12s**

|  |  |
| --- | --- |
| Module | Vessel |

**Overview**

Schedule API intermittently returns 500 for VESSEL\_ID; slowest 5% of requests latency > 12s. External subscribers missed webhook pushes; fallback polling exposed gaps in berth window at LOCATION.

**Resolution**

1. Investigate API 500 Failure:

1.1 Check the Schedule API for intermittent 500 errors and high latency.

1.2 Increase timeout, tune backoff settings, and enable request coalescing.

2. Increased timeout, tuned backoff, enabled request coalescing.

3. Rotated OAuth keys; enforced clock-skew tolerance; added SLI alerts for error-rate/latency.

3.1 Refresh/rotate token; confirm scopes.

3.2 Verify NTP/clock skew; resync if >2 seconds.

4. Communications used masked ticket references where applicable.

5. Reproduce the failing call on a safe test entity.

5.1 Use read-only token; capture status code and latency.

6. Check API gateway/service health around the incident window.

7. Review auth and permissions for the calling user/client.

7.1 Refresh/rotate token; confirm scopes.

7.2 Verify NTP/clock skew; resync if >2 seconds.

**Verification**

1. Re-run the endpoint; expect 2xx and correct JSON fields.

2. Observe error rate and latency for 30 minutes; under baseline.

3. Attach request/response evidence to the case.

**VSL: Schedule API intermittently returns 504 for VESSEL\_ID; slowest 5% of requests latency > 12s**

|  |  |
| --- | --- |
| Module | Vessel |

**Overview**

Schedule API intermittently returns 504 for VESSEL\_ID; slowest 5% of requests latency > 12s. External subscribers missed webhook pushes; fallback polling exposed gaps in berth window at LOCATION.

**Resolution**

1. Investigate API 504 Failure:

1.1 Check the Schedule API for intermittent 504 errors and high latency.

1.2 Increase timeout, tune backoff settings, and enable request coalescing.

2. Increased timeout, tuned backoff, enabled request coalescing.

3. Rotated OAuth keys; enforced clock-skew tolerance; added SLI alerts for error-rate/latency.

3.1 Refresh/rotate token; confirm scopes.

3.2 Verify NTP/clock skew; resync if >2 seconds.

4. Communications used masked ticket references where applicable.

5. Reproduce the failing call on a safe test entity.

5.1 Use read-only token; capture status code and latency.

5.2 Record request ID/correlation ID for log search.

6. Check API gateway/service health around the incident window.

7. Review auth and permissions for the calling user/client.

7.1 Refresh/rotate token; confirm scopes.

7.2 Verify NTP/clock skew; resync if >2 seconds.

**Verification**

1. Re-run the endpoint; expect 2xx and correct JSON fields.

2. Observe error rate and latency for 30 minutes; under baseline.

3. Attach request/response evidence to the case.

**VSL: EDIFACT COPRAR inconsistency for VESSEL\_ID; COARRI shows discharge completed for bay BAY\_NUMBER, but BAPLIE still lists units in those slots. Older timestamp regressed the plan.**

|  |  |
| --- | --- |
| Module | Vessel |

**Overview**

EDIFACT COPRAR inconsistency for VESSEL\_ID: COARRI shows discharge completed for bay BAY\_NUMBER, but BAPLIE still lists units in those slots. Older timestamp regressed the plan.

**Resolution**

1. Investigate BAPLIE Discrepancy:

1.1 Check for older timestamps causing regression in the discharge plan.

2. Enabled conflict resolution preferring max(eventTime); reprocessed from quarantine; verified stowage/discharge alignment across systems.

2.1 Use idempotency token to prevent duplicates.

2.2 Re-ingest from quarantine and monitor translator logs.

3. Communications used masked ticket references where applicable.

4. Locate the problematic EDI message/file and quarantine it.

5. Validate structure and partner-specific rules.

5.1 Run schema validator (segments/qualifiers/max-occurrence).

5.2 Fix qualifier mapping or split oversized occurrences.

6. Correct normalization/mapping and reprocess safely.

6.1 Use idempotency token to prevent duplicates.

6.2 Re-ingest from quarantine and monitor translator logs.

**Verification**

1. Confirm next run processed successfully; no duplicates.

2. Verify partner ACK/CONTRL is archived.

3. Attach validator logs and sanitized sample.

**VSL: Weather-Related Delay & Scheduling Adjustment SOP**

|  |  |
| --- | --- |
| Module | Vessel |

**Overview**

Pilotage delay due to weather; VESSEL\_ID missed ETB LOCATION slot. Towage reschedule pending; ripple effects on crane sequencing at quay CRANE\_LOCATION in LOCATION.

**Resolution**

1. Investigate Weather Delay Impact on System:

1.1 Triggered weather diversion workflow in the scheduling system to adjust ETA and re-sequence berth windows.

1.2 Automated reallocation of cranes and system update for updated scheduling.

2. Applied system updates to adjust scheduling based on weather input from external data feeds.

3. Ensure communication to stakeholders using sanitized system-generated updates.

4. Confirm scope and reproduce the issue on a safe test system.

5. Review recent deployments/config toggles around the incident timestamp to ensure no configuration drift.

6. Apply a compliant fix to the scheduling system and document the change.

6.1 Capture before/after evidence (system logs, test queries, etc.).

6.2 Update the case with sanitized details for tracking.

**Verification**

1. Run the end-to-end system journey again, verifying that the scheduling system adjusts for weather delays as expected.

2. Monitor system logs for no new errors for 30 minutes after fix deployment.

3.Attach evidence (system logs, test results) and close the case.

**VSL: ANSI X12 315 inconsistency for VESSEL\_ID: COARRI shows discharge completed for bay BAY\_NUMBER**

|  |  |
| --- | --- |
| Module | Vessel |

**Overview**

ANSI X12 315 inconsistency for VESSEL\_ID: COARRI shows discharge completed for bay BAY\_NUMBER, but BAPLIE still lists units in those slots. Older timestamp regressed the plan

**Resolution**

1. Investigate EDI Message Inconsistency:

1.1 Followed Vessel API Timeout & Retry process to resolve timestamp regression.

2. Enabled conflict resolution preferring max(eventTime); reprocessed from quarantine; verified stowage/discharge alignment across systems.

2.1 Use idempotency token to prevent duplicates.

2.2 Re-ingest from quarantine and monitor translator logs.

3. Communications used masked ticket references where applicable.

4. Locate the problematic EDI message/file and quarantine it.

5. Validate structure and partner-specific rules.

5.1 Run schema validator (segments/qualifiers/max-occurrence).

5.2 Fix qualifier mapping or split oversized occurrences.

6. Correct normalization/mapping and reprocess safely.

6.1 Use idempotency token to prevent duplicates.

6.2 Re-ingest from quarantine and monitor translator logs.

**Verification**

1. Confirm next run processed successfully; no duplicates.

2. Verify partner ACK/CONTRL is archived.

3. Attach validator logs and sanitized sample.

**VSL: Schedule API intermittently returns 502 for VESSEL\_ID; slowest 5% of requests latency > 12s**

|  |  |
| --- | --- |
| Module | Vessel |

**Overview**

Schedule API intermittently returns 502 for VESSEL\_ID; slowest 5% of requests latency > 12s. External subscribers missed webhook pushes; fallback polling exposed gaps in berth window at LOCATION.

**Resolution**

1. Investigate API 502 Failure:

1.1 Applied Berth Window Coordination process to mitigate impact.

1.2 Increased timeout, tuned backoff, and enabled request coalescing to address latency and error rate.

2. Increased timeout, tuned backoff, enabled request coalescing.

3. Rotated OAuth keys; enforced clock-skew tolerance; added SLI alerts for error-rate/latency.

3.1 Refresh/rotate token; confirm scopes.

3.2 Verify NTP/clock skew; resync if >2 seconds.

4. Communications used masked ticket references where applicable.

5. Reproduce the failing call on a safe test entity.

5.1 Use read-only token; capture status code and latency.

5.2 Record request ID/correlation ID for log search.

6. Check API gateway/service health around the incident window.

7. Review auth and permissions for the calling user/client.

7.1 Refresh/rotate token; confirm scopes.

7.2 Verify NTP/clock skew; resync if >2 seconds.

**Verification**

1. Re-run the endpoint; expect 2xx and correct JSON fields.

2. Observe error rate and latency for 30 minutes; under baseline.

3. Attach request/response evidence to the case.

**VSL: ANSI X12 301 inconsistency for VESSEL\_ID: COARRI shows discharge completed for bay BAY\_NUMBER**

|  |  |
| --- | --- |
| Module | Vessel |

**Overview**

ANSI X12 301 inconsistency for VESSEL\_ID: COARRI shows discharge completed for bay BAY\_NUMBER, but BAPLIE still lists units in those slots. Older timestamp regressed the plan.

**Resolution**

1. Investigate EDI Message Inconsistency:

1.1 Followed BAPLIE/COPRAR Verification process to resolve timestamp regression.

1.2 Reconcile discharge data and reprocess from quarantine to ensure alignment across systems.

2. Enabled conflict resolution preferring max(eventTime); reprocessed from quarantine; verified stowage/discharge alignment across systems.

2.1 Use idempotency token to prevent duplicates.

2.2 Re-ingest from quarantine and monitor translator logs.

3. Communications used masked ticket references where applicable.

4. Locate the problematic EDI message/file and quarantine it.

5. Validate structure and partner-specific rules.

5.1 Run schema validator (segments/qualifiers/max-occurrence).6. Correct normalization/mapping and reprocess safely.

6.1 Use idempotency token to prevent duplicates.

6.2 Re-ingest from quarantine and monitor translator logs.

**Verification**

1. Confirm next run processed successfully; no duplicates.

2. Verify partner ACK/CONTRL is archived.

3. Attach validator logs and sanitized sample.

**VSL: ETA/ETB mismatch for VESSEL\_ID on voyage VOYAGE\_ID**

|  |  |
| --- | --- |
| Module | Vessel |

**Overview**

ETA/ETB mismatch for VESSEL\_ID on voyage VOYAGE\_ID. Berth plan shows ETB at LOCATION while TOS differs by >90 minutes due to tidal constraints. Partner portal still displays outdated ETA; yard resource plan at LOCATION reflects older figures. Pilotage request pending; quay CRANE\_LOCATION crane allocation misaligned.

**Resolution**

1. Investigate ETA/ETB Mismatch:

1.1 Reconcile ETA/ETB discrepancies across systems, normalizing to port timezone.

1.2 Recalculate berth plan and refresh crane profiles to align with updated times.

2. Reconciled ETA/ETB across systems; normalized times to port TZ and re-published.

3. Triggered berth plan recalculation and crane profile refresh.

4. Added drift guard to auto-alert on >30 min divergence.

5. Communications used masked ticket references where applicable.

6. Compare authoritative sources and capture the exact delta.

7. Check event sequencing; identify missing/late/out-of-order events.

7.1 Sort milestones by eventTime.

7.2 Replay missing yard move/completion if allowed by SOP.

8. Trigger a targeted re-sync or corrective update per SOP.

8.1 Sort milestones by eventTime.

8.2 Replay missing yard move/completion if allowed by SOP.

**Verification**

1. Re-open both systems and confirm fields align (status/location/size-type).

2. Ensure event ordering is correct; no stale updates reappear.

3. Attach before/after screenshots or queries.

**VSL: BAPLIE inconsistency for VESSEL\_ID: COARRI shows discharge completed for bay BAY\_NUMBER**

|  |  |
| --- | --- |
| Module | Vessel |

**Overview**

BAPLIE inconsistency for VESSEL\_ID: COARRI shows discharge completed for bay BAY\_NUMBER, but BAPLIE still lists units in those slots. Older timestamp regressed the plan.

**Resolution**

1. Investigate BAPLIE Inconsistency:

1.1 Followed BAPLIE/COPRAR Verification process to resolve timestamp regression.

2. Enabled conflict resolution preferring max(eventTime); reprocessed from quarantine; verified stowage/discharge alignment across systems.

2.1 Use idempotency token to prevent duplicates.

2.2 Re-ingest from quarantine and monitor translator logs.

3. Communications used masked ticket references where applicable.

4. Locate the problematic EDI message/file and quarantine it.

5. Validate structure and partner-specific rules.

5.1 Run schema validator (segments/qualifiers/max-occurrence).

5.2 Fix qualifier mapping or split oversized occurrences.

6. Correct normalization/mapping and reprocess safely.

6.1 Use idempotency token to prevent duplicates.

6.2 Re-ingest from quarantine and monitor translator logs.

**Verification**

1. Confirm next run processed successfully; no duplicates.

2. Verify partner ACK/CONTRL is archived.

3. Attach validator logs and sanitized sample.

**VSL: ANSI X12 301 inconsistency for VESSEL\_ID: COARRI shows discharge completed for bay BAY\_NUMBER**

|  |  |
| --- | --- |
| Module | Vessel |

**Overview**

ANSI X12 301 inconsistency for VESSEL\_ID: COARRI shows discharge completed for bay BAY\_NUMBER, but BAPLIE still lists units in those slots. Older timestamp regressed the plan.

**Resolution**

1. Investigate EDI Inconsistency:

1.1 Followed Duplicate EDI De-duplication process to resolve timestamp regression.

2. Enabled conflict resolution preferring max(eventTime); reprocessed from quarantine; verified stowage/discharge alignment across systems.

2.1 Use idempotency token to prevent duplicates.

2.2 Re-ingest from quarantine and monitor translator logs.

3. Communications used masked ticket references where applicable.

4. Locate the problematic EDI message/file and quarantine it.

5. Validate structure and partner-specific rules.

5.1 Run schema validator (segments/qualifiers/max-occurrence).

5.2 Fix qualifier mapping or split oversized occurrences.

6. Correct normalization/mapping and reprocess safely.

6.1 Use idempotency token to prevent duplicates.

6.2 Re-ingest from quarantine and monitor translator logs.

**Verification**

1. Confirm next run processed successfully; no duplicates.

2. Verify partner ACK/CONTRL is archived.

3. Attach validator logs and sanitized sample.

**VSL: Schedule API intermittently returns 401 for VESSEL\_ID; slowest 5% of requests latency > 12s**

|  |  |
| --- | --- |
| Module | Vessel |

**Overview**

Schedule API intermittently returns 401 for VESSEL\_ID; slowest 5% of requests latency > 12s. External subscribers missed webhook pushes; fallback polling exposed gaps in berth window at LOCATION.

**Resolution**

1. Investigate API 401 Failure:

1.1 Followed BAPLIE/COPRAR Verification process to address authentication issues.

1.2 Increase timeout, tune backoff settings, and rotate OAuth keys to resolve 401 errors.

2. Increased timeout, tuned backoff, enabled request coalescing.

3. Rotated OAuth keys; enforced clock-skew tolerance; added SLI alerts for error-rate/latency.

3.1 Refresh/rotate token; confirm scopes.

3.2 Verify NTP/clock skew; resync if >2 seconds.

4. Communications used masked ticket references where applicable.

5. Reproduce the failing call on a safe test entity.

5.1 Use read-only token; capture status code and latency.

5.2 Record request ID/correlation ID for log search.

6. Check API gateway/service health around the incident window.

7. Review auth and permissions for the calling user/client.

7.1 Refresh/rotate token; confirm scopes.

7.2 Verify NTP/clock skew; resync if >2 seconds.

**Verification**

1. Re-run the endpoint; expect 2xx and correct JSON fields.

2. Observe error rate and latency for 30 minutes; under baseline.

3. Attach request/response evidence to the case.

**VSL: ETA/ETB mismatch for VESSEL\_ID on voyage VOYAGE\_ID**

|  |  |
| --- | --- |
| Module | Vessel |

**Overview**

ETA/ETB mismatch for VESSEL\_ID on voyage VOYAGE\_ID. Berth plan shows ETB at LOCATION while TOS differs by >90 minutes due to tidal constraints. Partner portal still displays outdated ETA; yard resource plan at LOCATION reflects older figures. Pilotage request pending; quay CRANE\_LOCATION crane allocation misaligned.

**Resolution**

1. Investigate ETA/ETB Mismatch:

1.1 Executed Manual Override & Audit Trail to reconcile ETA/ETB across systems.

1.2 Normalize times to port timezone and trigger berth plan recalculation.

2. Reconciled ETA/ETB across systems; normalized times to port TZ and re-published.

3. Triggered berth plan recalculation and crane profile refresh.

4. Added drift guard to auto-alert on >30 min divergence.

5. Communications used masked ticket references where applicable.

6. Compare authoritative sources and capture the exact delta.

6.1 Check status/location/size-type fields.

7. Check event sequencing; identify missing/late/out-of-order events.

7.1 Sort milestones by eventTime.

8. Trigger a targeted re-sync or corrective update per SOP.

8.1 Sort milestones by eventTime.

8.2 Replay missing yard move/completion if allowed by SOP.

**Verification**

1. Re-open both systems and confirm fields align (status/location/size-type).

2. Ensure event ordering is correct; no stale updates reappear.

3. Attach before/after screenshots or queries.

**VSL: ETA/ETB mismatch for VESSEL\_ID on voyage VOYAGE\_ID**

|  |  |
| --- | --- |
| Module | Vessel |

**Overview**

ETA/ETB mismatch for VESSEL\_ID on voyage VOYAGE\_ID. Berth plan shows ETB at LOCATION while TOS differs by >90 minutes due to tidal constraints. Partner portal still displays outdated ETA; yard resource plan at LOCATION reflects older figures. Pilotage request pending; quay CRANE\_LOCATION crane allocation misaligned.

**Resolution**

1. Investigate ETA/ETB Mismatch:

1.1 Executed Pilotage & Towage Delay Workflow to address mismatch.

1.2 Reconcile ETA/ETB across systems, normalize times to port timezone, and trigger recalculation of berth plan and crane profiles.

2. Reconciled ETA/ETB across systems; normalized times to port TZ and re-published.

3. Triggered berth plan recalculation and crane profile refresh.

4. Added drift guard to auto-alert on >30 min divergence.

5. Communications used masked ticket references where applicable.

6. Compare authoritative sources and capture the exact delta.

6.1 Check status/location/size-type fields.

6.2 Capture last-updated timestamps from each system.

7. Check event sequencing; identify missing/late/out-of-order events.

7.1 Sort milestones by eventTime.

8. Trigger a targeted re-sync or corrective update per SOP.

**Verification**

1. Re-open both systems and confirm fields align (status/location/size-type).

2. Ensure event ordering is correct; no stale updates reappear.

3. Attach before/after screenshots or queries.

**VSL: BAPLIE inconsistency for VESSEL\_ID: COARRI shows discharge completed for bay BAY\_NUMBER**

|  |  |
| --- | --- |
| Module | Vessel |

**Overview**

BAPLIE inconsistency for VESSEL\_ID: COARRI shows discharge completed for bay BAY\_NUMBER, but BAPLIE still lists units in those slots. Older timestamp regressed the plan.

**Resolution**

1. Investigate BAPLIE Inconsistency:

1.1 Followed Pilotage & Towage Delay Workflow to address timestamp regression.

1.2 Reconcile discharge data and reprocess from quarantine, ensuring alignment across systems.

2. Enabled conflict resolution preferring max(eventTime); reprocessed from quarantine; verified stowage/discharge alignment across systems.

2.1 Use idempotency token to prevent duplicates.

2.2 Re-ingest from quarantine and monitor translator logs.

3. Communications used masked ticket references where applicable.

4. Locate the problematic EDI message/file and quarantine it.

5. Validate structure and partner-specific rules.

5.1 Run schema validator (segments/qualifiers/max-occurrence).

5.2 Fix qualifier mapping or split oversized occurrences.

6. Correct normalization/mapping and reprocess safely.

6.1 Use idempotency token to prevent duplicates.

**Verification**

1. Confirm next run processed successfully; no duplicates.

2. Verify partner ACK/CONTRL is archived.

3. Attach validator logs and sanitized sample.

**VSL: ETA/ETB mismatch for VESSEL\_ID on voyage VOYAGE\_ID**

|  |  |
| --- | --- |
| Module | Vessel |

**Overview**

ETA/ETB mismatch for VESSEL\_ID on voyage VOYAGE\_ID. Berth plan shows ETB at LOCATION while TOS differs by >90 minutes due to tidal constraints. Partner portal still displays outdated ETA; yard resource plan at LOCATION reflects older figures. Pilotage request pending; quay CRANE\_LOCATION crane allocation misaligned.

**Resolution**

1. Investigate ETA/ETB Mismatch:

1.1 Executed Berth Window Coordination to resolve ETA/ETB discrepancies.

1.2 Reconcile ETA/ETB across systems, normalize times to port timezone, and trigger recalculation of berth plan and crane profiles.

2. Reconciled ETA/ETB across systems; normalized times to port TZ and re-published.

3. Triggered berth plan recalculation and crane profile refresh.

4. Added drift guard to auto-alert on >30 min divergence.

5. Communications used masked ticket references where applicable.

6. Compare authoritative sources and capture the exact delta.

6.1 Check status/location/size-type fields.

6.2 Capture last-updated timestamps from each system.

7. Check event sequencing; identify missing/late/out-of-order events.

7.1 Sort milestones by eventTime.

7.2 Replay missing yard move/completion if allowed by SOP.

8. Trigger a targeted re-sync or corrective update per SOP.

8.1 Sort milestones by eventTime.

8.2 Replay missing yard move/completion if allowed by SOP.

**Verification**

1. Re-open both systems and confirm fields align (status/location/size-type).

2. Ensure event ordering is correct; no stale updates reappear.

3. Attach before/after screenshots or queries.

**VSL: Pilotage delay due to weather; VESSEL\_ID missed ETB LOCATION slot**

|  |  |
| --- | --- |
| Module | Vessel |

**Overview**

Pilotage delay due to weather; VESSEL\_ID missed ETB LOCATION slot. Towage reschedule pending; ripple effects on crane sequencing at quay CRANE\_LOCATION in LOCATION.

**Resolution**

1. Investigate Pilotage Delay:

1.1 Triggered Pilotage & Towage Delay Workflow to manage the weather-related delay.

1.2 Re-sequenced berth windows, reallocated cranes, and communicated sanitized updates to stakeholders.

2. Re-sequenced berth windows and reallocated cranes; communicated sanitized update to stakeholders.

3. Communications used masked ticket references where applicable.

4. Confirm scope and reproduction on a safe test entity.

4.1 check Towage/location/size-type fields.

4.2 Capture last-updated timestamps from each system.

5. Check recent deployments/config toggles around the timestamp.

6. Apply compliant fix and document the change.

6.1 Capture before/after evidence (screenshots/queries).

6.2 Update the case with sanitized details.

**Verification**

1. Run the end-to-end journey again; confirm success.

2. No new errors for 30 minutes in monitoring.

3. Attach evidence and close the case.

**VSL: ETA/ETB mismatch for VESSEL\_ID on VOYAGE\_ID**

|  |  |
| --- | --- |
| Module | Vessel |

**Overview**

ETA/ETB mismatch for VESSEL\_ID on voyage VOYAGE\_ID. Berth plan shows ETB at LOCATION while TOS differs by >90 minutes due to tidal constraints. Partner portal still displays outdated ETA; yard resource plan at LOCATION reflects older figures. Pilotage request pending; quay CRANE\_LOCATION crane allocation misaligned.

**Resolution**

1. Investigate ETA/ETB Mismatch:

1.1 Executed Stowage Plan Alignment process to resolve ETA/ETB discrepancies.

1.2 Reconcile ETA/ETB across systems, normalize times to port timezone, and trigger recalculation of berth plan and crane profiles.

2. Reconciled ETA/ETB across systems; normalized times to port TZ and re-published.

3. Triggered berth plan recalculation and crane profile refresh.

4. Added drift guard to auto-alert on >30 min divergence.

5. Communications used masked ticket references where applicable.

6. Compare authoritative sources and capture the exact delta.

6.1 Check status/location/size-type fields.

6.2 Capture last-updated timestamps from each system.

7. Check event sequencing; identify missing/late/out-of-order events.

7.1 Sort milestones by eventTime.

7.2 Replay missing yard move/completion if allowed by SOP.

8. Trigger a targeted re-sync or corrective update per SOP.

8.1 Sort milestones by eventTime.

8.2 Replay missing yard move/completion if allowed by SOP.

**Verification**

1. Re-open both systems and confirm fields align (status/location/size-type).

2. Ensure event ordering is correct; no stale updates reappear.

3. Attach before/after screenshots or queries.

**VSL: Schedule API intermittently returns 500 for VESSEL\_ID; slowest 5% of requests latency > 12s**

|  |  |
| --- | --- |
| Module | Vessel |

**Overview**

Schedule API intermittently returns 500 for VESSEL\_ID; slowest 5% of requests latency > 12s. External subscribers missed webhook pushes; fallback polling exposed gaps in berth window at LOCATION.

**Resolution**

1. Investigate API 500 Failure:

1.1 Applied ETA/ETB Reconciliation to address API failures.

1.2 Increased timeout, tuned backoff, and enabled request coalescing to reduce latency and error rate.

2. Increased timeout, tuned backoff, enabled request coalescing.

3. Rotated OAuth keys; enforced clock-skew tolerance; added SLI alerts for error-rate/latency.

3.1 Refresh/rotate token; confirm scopes.

3.2 Verify NTP/clock skew; resync if >2 seconds.

4. Communications used masked ticket references where applicable.

5. Reproduce the failing call on a safe test entity.

5.1 Use read-only token; capture status code and latency.

5.2 Record request ID/correlation ID for log search.

6. Check API gateway/service health around the incident window.

7. Review auth and permissions for the calling user/client.

7.1 Refresh/rotate token; confirm scopes.

7.2 Verify NTP/clock skew; resync if >2 seconds.

**Verification**

1. Re-run the endpoint; expect 2xx and correct JSON fields.

2. Observe error rate and latency for 30 minutes; under baseline.

3. Attach request/response evidence to the case.

**VSL: ETA/ETB mismatch for VESSEL\_ID on voyage VOYAGE\_ID**

|  |  |
| --- | --- |
| Module | Vessel |

**Overview**

ETA/ETB mismatch for VESSEL\_ID on voyage VOYAGE\_ID. Berth plan shows ETB at LOCATION while TOS differs by >90 minutes due to tidal constraints. Partner portal still displays outdated ETA; yard resource plan at LOCATION reflects older figures. Pilotage request pending; quay CRANE\_LOCATION crane allocation misaligned.

**Resolution**

1. Investigate ETA/ETB Mismatch:

1.1 Executed Manifest Closure & Rollover to address discrepancies.

1.2 Reconcile ETA/ETB across systems, normalize times to port timezone, and trigger recalculation of berth plan and crane profiles.

2. Reconciled ETA/ETB across systems; normalized times to port TZ and re-published.

3. Triggered berth plan recalculation and crane profile refresh.

4. Added drift guard to auto-alert on >30 min divergence.

5. Communications used masked ticket references where applicable.

6. Compare authoritative sources and capture the exact delta.

6.1 Check status/location/size-type fields.

6.2 Capture last-updated timestamps from each system.

7. Check event sequencing; identify missing/late/out-of-order events.

7.1 Sort milestones by eventTime.

7.2 Replay missing yard move/completion if allowed by SOP.

8. Trigger a targeted re-sync or corrective update per SOP.

8.1 Sort milestones by eventTime.

8.2 Replay missing yard move/completion if allowed by SOP.

**Verification**

1. Re-open both systems and confirm fields align (status/location/size-type).

2. Ensure event ordering is correct; no stale updates reappear.

3. Attach before/after screenshots or queries.

**VSL: ETA/ETB mismatch for VESSEL\_ID on voyage VOYAGE\_ID**

|  |  |
| --- | --- |
| Module | Vessel |

**Overview**

ETA/ETB mismatch for VESSEL\_ID on voyage VOYAGE\_ID. Berth plan shows ETB at LOCATION while TOS differs by >90 minutes due to tidal constraints. Partner portal still displays outdated ETA; yard resource plan at LOCATION reflects older figures. Pilotage request pending; quay CRANE\_LOCATION crane allocation misaligned.

**Resolution**

1. Investigate ETA/ETB Mismatch:

1.1 Executed Vessel API Timeout & Retry to resolve discrepancies.

1.2 Reconcile ETA/ETB across systems, normalize times to port timezone, and trigger recalculation of berth plan and crane profiles.

2. Reconciled ETA/ETB across systems; normalized times to port TZ and re-published.

3. Triggered berth plan recalculation and crane profile refresh.

4. Added drift guard to auto-alert on >30 min divergence.

5. Communications used masked ticket references where applicable.

6. Compare authoritative sources and capture the exact delta.

6.1 Check status/location/size-type fields.

6.2 Capture last-updated timestamps from each system.

7. Check event sequencing; identify missing/late/out-of-order events.

7.1 Sort milestones by eventTime.

7.2 Replay missing yard move/completion if allowed by SOP.

8. Trigger a targeted re-sync or corrective update per SOP.

8.1 Sort milestones by eventTime.

8.2 Replay missing yard move/completion if allowed by SOP.

**Verification**

1. Re-open both systems and confirm fields align (status/location/size-type).

2. Ensure event ordering is correct; no stale updates reappear.

3. Attach before/after screenshots or queries.

**VSL: EDIFACT COPRAR inconsistency for VESSEL\_ID: COARRI shows discharge completed for bay BAY\_NUMBER**

|  |  |
| --- | --- |
| Module | Vessel |

**Overview**

EDIFACT COPRAR inconsistency for VESSEL\_ID: COARRI shows discharge completed for bay BAY\_NUMBER, but BAPLIE still lists units in those slots. Older timestamp regressed the plan.

**Resolution**

1. Investigate EDI Inconsistency:

1.1 Followed Stowage Plan Alignment to resolve timestamp regression.

1.2 Reconcile discharge data and reprocess from quarantine, ensuring alignment across systems.

2. Enabled conflict resolution preferring max(eventTime); reprocessed from quarantine; verified stowage/discharge alignment across systems.

2.1 Use idempotency token to prevent duplicates.

3. Communications used masked ticket references where applicable.

4. Locate the problematic EDI message/file and quarantine it.

5. Validate structure and partner-specific rules.

5.1 Run schema validator (segments/qualifiers/max-occurrence).

5.2 Fix qualifier mapping or split oversized occurrences.

6. Correct normalization/mapping and reprocess safely.

6.1 Use idempotency token to prevent duplicates.

6.2 Re-ingest from quarantine and monitor translator logs.

**Verification**

1. Confirm next run processed successfully; no duplicates.

2. Verify partner ACK/CONTRL is archived.

3. Attach validator logs and sanitized sample.

**VSL: Schedule API intermittently returns 429 for VESSEL\_ID; slowest 5% of requests latency > 12s**

|  |  |
| --- | --- |
| Module | Vessel |

**Overview**

Schedule API intermittently returns 429 for VESSEL\_ID; slowest 5% of requests latency > 12s. External subscribers missed webhook pushes; fallback polling exposed gaps in berth window at LOCATION.

**Resolution**

1. Investigate API 429 Failure:

1.1 Applied Manifest Closure & Rollover to manage rate-limiting.

1.2 Increased timeout, tuned backoff settings, and enabled request coalescing to reduce latency and error rate.

2. Increased timeout, tuned backoff, enabled request coalescing.

3. Rotated OAuth keys; enforced clock-skew tolerance; added SLI alerts for error-rate/latency.

3.1 Refresh/rotate token; confirm scopes.

3.2 Verify NTP/clock skew; resync if >2 seconds.

4. Communications used masked ticket references where applicable.

5. Reproduce the failing call on a safe test entity.

5.1 Use read-only token; capture status code and latency.

5.2 Record request ID/correlation ID for log search.

6. Check API gateway/service health around the incident window.

7. Review auth and permissions for the calling user/client.

7.1 Refresh/rotate token; confirm scopes.

7.2 Verify NTP/clock skew; resync if >2 seconds.

**Verification**

1. Re-run the endpoint; expect 2xx and correct JSON fields.

2. Observe error rate and latency for 30 minutes; under baseline.

3. Attach request/response evidence to the case.

**VSL: Schedule API intermittently returns 500 for VESSEL\_ID; slowest 5% of requests latency > 12s**

|  |  |
| --- | --- |
| Module | Vessel |

**Overview**

Schedule API intermittently returns 500 for VESSEL\_ID; slowest 5% of requests latency > 12s. External subscribers missed webhook pushes; fallback polling exposed gaps in berth window at LOCATION.

**Resolution**

1. Investigate API 500 Failure:

1.1 Applied Stowage Plan Alignment to address the issue.

1.2 Increased timeout, tuned backoff, and enabled request coalescing to reduce latency and error rate.

2. Increased timeout, tuned backoff, enabled request coalescing.

3. Rotated OAuth keys; enforced clock-skew tolerance; added SLI alerts for error-rate/latency.

3.1 Refresh/rotate token; confirm scopes.

3.2 Verify NTP/clock skew; resync if >2 seconds.

4. Communications used masked ticket references where applicable.

5. Reproduce the failing call on a safe test entity.

5.1 Use read-only token; capture status code and latency.

5.2 Record request ID/correlation ID for log search.

6. Check API gateway/service health around the incident window.

7. Review auth and permissions for the calling user/client.

7.1 Refresh/rotate token; confirm scopes.

7.2 Verify NTP/clock skew; resync if >2 seconds.

**Verification**

1. Re-run the endpoint; expect 2xx and correct JSON fields.

2. Observe error rate and latency for 30 minutes; under baseline.

3. Attach request/response evidence to the case.

**API: Rate limiter throttled legitimate traffic after load surge**

|  |  |
| --- | --- |
| Module | EDI/API |

**Overview**

Rate limiter throttled legitimate traffic after load surge. Hot keys on '/edi/upload/UPLOAD\_ID' created cache stampede; error budget burned for the window. Context tokens: CONTEXT\_TOKEN1, CONTEXT\_TOKEN2.

**Resolution**

1. Introduced per-partner rate shaping; added idempotency keys for webhook retries; validated TLS and cipher compatibility; published sanitized RCA.

2. Communications used masked ticket references where applicable.

3. Reproduce the failing call on a safe test entity.

3.1 Use read-only token; capture status code and latency.

3.2 Record request ID/correlation ID for log search.

4. Check API gateway/service health around the incident window.

5. Review auth and permissions for the calling user/client.

5.1 Refresh/rotate token; confirm scopes.

5.2 Verify NTP/clock skew; resync if >2 seconds.

**Verification**

1. Re-run the endpoint; expect 2xx and correct JSON fields.

2. Observe error rate and latency for 30 minutes; under baseline.

3. Attach request/response evidence to the case.

**EDI: Intermittent 408 on '/auth/token' during webhook deliveries to PARTNER\_ID**

|  |  |
| --- | --- |
| Module | EDI/API |

**Overview**

Intermittent 408 on '/auth/token' during webhook deliveries to PARTNER\_ID. slowest 5% of requests latency > 12s triggered breaker; retries hit max; delivery backlog observed. Downstream customer timeline showed gaps; LOCATION ops unaffected but visibility degraded.

**Resolution**

1. Scaled read replicas and split hot key namespaces; added half-open probes to breaker; restored normal delivery and rebuilt backlog safely.

2. Communications used masked ticket references where applicable.

3. Reproduce the failing call on a safe test entity.

3.1 Use read-only token; capture status code and latency.

3.2 Record request ID/correlation ID for log search.

4. Check API gateway/service health around the incident window.

5. Review auth and permissions for the calling user/client.

5.1 Refresh/rotate token; confirm scopes.

5.2 Verify NTP/clock skew; resync if >2 seconds.

**Verification**

1. Re-run the endpoint; expect 2xx and correct JSON fields.

2. Observe error rate and latency for 30 minutes; under baseline.

3. Attach request/response evidence to the case.

**EDI: Time zone drift caused eventTime to serialize in UTC+0 for partner PARTNER\_ID, while TOS computed local UTC+8.**

|  |  |
| --- | --- |
| Module | EDI/API |

**Overview**

Time zone drift caused eventTime to serialize in UTC+0 for partner PARTNER\_ID, while TOS computed local UTC+8. Container CONTAINER\_ID milestones appeared out-of-order on the portal; BAPLIE 'Gate Out' overshadowed later 'Loaded' event. Booking BOOKING\_ID not impacted operationally but customer view inconsistent. Context tokens: CONTEXT\_TOKEN1, CONTEXT\_TOKEN2.

**Resolution**

1. Investigate Time Zone Drift:

1.1 Executed Duplicate Interchange De-duplication to resolve serialization mismatch.

1.2 Set precedence to TOS for equipment attributes, purged cache, and re-indexed data to correct event ordering.

2. Set precedence to TOS for equipment attributes; forced cache purge and re-index.

3. Backfilled corrected attributes, then locked fields for 6h to prevent drift.

3.1 Archive ACK with timestamp and execution ID.

3.2 Notify partner with sanitized sample if needed.

4. Added rule to reject backdated EDI updates if newer move exists within 120 minutes.

4.1 Archive ACK with timestamp and execution ID.

4.2 Notify partner with sanitized sample if needed.

5. Communications used masked ticket references where applicable.

6. Locate the problematic EDI message/file and quarantine it.

7. Validate structure and partner-specific rules.

7.1 Run schema validator (segments/qualifiers/max-occurrence).

7.2 Fix qualifier mapping or split oversized occurrences.

8. Correct normalization/mapping and reprocess safely.

8.1 Use idempotency token to prevent duplicates.

8.2 Re-ingest from quarantine and monitor translator logs.

**Verification**

1. Confirm next run processed successfully; no duplicates.

2. Verify partner ACK/CONTRL is archived.

3. Attach validator logs and sanitized sample.

**EDI: Unexpected qualifier 'QUALIFIER' in EQD segment from Partner\_ID; schema validation failed...**

|  |  |
| --- | --- |
| Module | EDI/API |

**Overview**

Unexpected qualifier 'QUALIFIER' in EQD segment from PARTNER\_ID; schema validation failed; containers in the message not reflected on the portal timeline. Context tokens: CONTEXT\_TOKEN1, CONTEXT\_TOKEN2.

**Resolution**

1. Implemented pre-ingest normalizer for qualifiers; updated partner IG notes; enabled anomaly detection for repeated segment errors exceeds threshold.

1.1 Run schema validator (segments/qualifiers/max-occurrence).

1.2 Fix qualifier mapping or split oversized occurrences.

2. Communications used masked ticket references where applicable.

3. Locate the problematic EDI message/file and quarantine it.

4. Validate structure and partner-specific rules.

4.1 Run schema validator (segments/qualifiers/max-occurrence).

4.2 Fix qualifier mapping or split oversized occurrences.

5. Correct normalization/mapping and reprocess safely.

5.1 Use idempotency token to prevent duplicates.

5.2 Re-ingest from quarantine and monitor translator logs.

**Verification**

1. Confirm next run processed successfully; no duplicates.

2. Verify partner ACK/CONTRL is archived.

3. Attach validator logs and sanitized sample.

**API: OAuth token rejection spikes on '/events/subscribe'**

|  |  |
| --- | --- |
| Module | EDI/API |

**Overview**

OAuth token rejection spikes on '/events/subscribe'. Clock skew across nodes yielded 401/403 bursts; clients observed partial responses before circuit opened. Context tokens: CONTEXT\_TOKEN1, CONTEXT\_TOKEN2.

**Resolution**

1. Investigate OAuth Token Rejection:

1.1 Applied API Timeout & Retry Policy to address 401/403 errors.

1.2 Increased timeouts, tuned exponential backoff, and enforced token leeway to mitigate clock skew issues.

2. Increased timeouts, tuned exponential backoff, enabled request coalescing.

3. Rotated OAuth keys and enforced ±60s token leeway.

3.1 Refresh/rotate token; confirm scopes.

3.2 Verify NTP/clock skew; resync if >2 seconds.

4. Warmed cache for hot routes and raised SLI alerts.

4.1 Check DB slowest 5% of requests latency; look for lock waits.

4.2 Purge problematic cache keys; warm hot keys if needed.

5. Communications used masked ticket references where applicable.

6. Reproduce the failing call on a safe test entity.

6.1 Use read-only token; capture status code and latency.

6.2 Record request ID/correlation ID for log search.

7. Check API gateway/service health around the incident window.

8. Review auth and permissions for the calling user/client.

8.1 Refresh/rotate token; confirm scopes.

8.2 Verify NTP/clock skew; resync if >2 seconds.

**Verification**

1. Re-run the endpoint; expect 2xx and correct JSON fields.

2. Observe error rate and latency for 30 minutes; under baseline.

3. Attach request/response evidence to the case.

**EDI: Spike in DLQ messages after routine maintenance; consumer group lag increased across EDI topic**

|  |  |
| --- | --- |
| Module | EDI/API |

**Overview**

Spike in DLQ messages after routine maintenance; consumer group lag increased across EDI topic. Partner PARTNER\_ID reported delayed acknowledgements; LOCATION ops not impacted. Context tokens: CONTEXT\_TOKEN1, CONTEXT\_TOKEN2.

**Resolution**

1. Corrected cron schedule to UTC; added mutual exclusion lock and run-idempotency; published sanitized incident note and fixed dashboards.

2. Communications used masked ticket references where applicable.

3. Reproduce the failing call on a safe test entity.

3.1 Use read-only token; capture status code and latency.

3.2 Record request ID/correlation ID for log search.

4. Check API gateway/service health around the incident window.

5. Review auth and permissions for the calling user/client.

5.1 Refresh/rotate token; confirm scopes.

5.2 Verify NTP/clock skew; resync if >2 seconds.

**Verification**

1. Re-run the endpoint; expect 2xx and correct JSON fields.

2. Observe error rate and latency for 30 minutes; under baseline.

3. Attach request/response evidence to the case.

**EDI: Translator rejected ANSI X12 301: missing BGM count alignment (UNH/UNT mismatch)**

|  |  |
| --- | --- |
| Module | EDI/API |

**Overview**

Translator rejected ANSI X12 301: missing BGM count alignment (UNH/UNT mismatch). Partner PARTNER\_ID interchange quarantined; downstream milestone not created.

**Resolution**

1. Added de-duplication keyed by (controlNumber, docType, equipmentId, eventType); instated rule to discard stale duplicates; replayed healthy messages.

2. Communications used masked ticket references where applicable.

3. Locate the problematic EDI message/file and quarantine it.

4. Validate structure and partner-specific rules.

4.1 Run schema validator (segments/qualifiers/max-occurrence).

4.2 Fix qualifier mapping or split oversized occurrences.

5. Correct normalization/mapping and reprocess safely.

5.1 Use idempotency token to prevent duplicates.

5.2 Re-ingest from quarantine and monitor translator logs.

**Verification**

1. Confirm next run processed successfully; no duplicates.

2. Verify partner ACK/CONTRL is archived.

3. Attach validator logs and sanitized sample.

**EDI: Cron for backfill job misfired after DST adjustment; two runs overlapped causing duplicate API publications for the same event batch.**

|  |  |
| --- | --- |
| Module | EDI/API |

**Overview**

Cron for backfill job misfired after DST adjustment; two runs overlapped causing duplicate API publications for the same event batch.

**Resolution**

1. Scaled consumers and rebalanced partitions; drained DLQ with rate control; enabled poison-message quarantine and added SLO alerting

2. Communications used masked ticket references where applicable.

3. Reproduce the failing call on a safe test entity.

3.1 Use read-only token; capture status code and latency.

3.2 Record request ID/correlation ID for log search.

4. Check API gateway/service health around the incident window.

5. Review auth and permissions for the calling user/client.

5.1 Refresh/rotate token; confirm scopes.

5.2 Verify NTP/clock skew; resync if >2 seconds.

**Verification**

1. Re-run the endpoint; expect 2xx and correct JSON fields.

2. Observe error rate and latency for 30 minutes; under baseline.

3. Attach request/response evidence to the case.

**API: Shipper/consignee role swap observed on API payload for CONTAINER\_ID**

|  |  |
| --- | --- |
| Module | EDI/API |

**Overview**

Shipper/consignee role swap observed on API payload for CONTAINER\_ID. Upstream denormalization reversed NAD qualifiers; billing prep failed validation. Partner: PARTNER\_ID; affected terminal: LOCATION.

**Resolution**

1. Investigate Shipper/Consignee Role Swap:

1.1 Executed Webhook Delivery & Idempotency process to resolve role swap issue.

1.2 Set precedence to TOS for equipment attributes, purged cache, and re-indexed data.

2. Set precedence to TOS for equipment attributes; forced cache purge and re-index.

2.1 Check DB slowest 5% of requests latency; look for lock waits.

2.2 Purge problematic cache keys; warm hot keys if needed.

3. Backfilled corrected attributes, then locked fields for 6h to prevent drift.

4. Added rule to reject backdated EDI updates if newer move exists within 120 minutes.

5. Communications used masked ticket references where applicable.

6. Reproduce the failing call on a safe test entity.

6.1 Use read-only token; capture status code and latency.

6.2 Record request ID/correlation ID for log search.

7. Check API gateway/service health around the incident window.

8. Review auth and permissions for the calling user/client.

8.1 Refresh/rotate token; confirm scopes.

8.2 Verify NTP/clock skew; resync if >2 seconds.

**Verification**

1. Re-run the endpoint; expect 2xx and correct JSON fields.

2. Observe error rate and latency for 30 minutes; under baseline.

3. Attach request/response evidence to the case.

**API: OAuth token rejection spikes on '/auth/token'**

|  |  |
| --- | --- |
| Module | EDI/API |

**Overview**

OAuth token rejection spikes on '/auth/token'. Clock skew across nodes yielded 401/403 bursts; clients observed partial responses before circuit opened. Context tokens: HLCUPN7241107580.

**Resolution**

1. Scaled read replicas and split hot key namespaces; added half-open probes to breaker; restored normal delivery and rebuilt backlog safely.

2. Communications used masked ticket references where applicable.

3. Reproduce the failing call on a safe test entity.

3.1 Use read-only token; capture status code and latency.

3.2 Record request ID/correlation ID for log search.

4. Check API gateway/service health around the incident window.

5. Review auth and permissions for the calling user/client.

5.1 Refresh/rotate token; confirm scopes.

5.2 Verify NTP/clock skew; resync if >2 seconds.

**Verification**

1. Re-run the endpoint; expect 2xx and correct JSON fields.

2. Observe error rate and latency for 30 minutes; under baseline.

3. Attach request/response evidence to the case.

**EDI: Rate limiter throttled legitimate traffic after load surge**

|  |  |
| --- | --- |
| Module | EDI/API |

**Overview**

Rate limiter throttled legitimate traffic after load surge. Hot keys on '/webhooks/delivery/DELIVERY\_ID' created cache stampede; error budget burned for the window.

**Resolution**

1. Scaled read replicas and split hot key namespaces; added half-open probes to breaker; restored normal delivery and rebuilt backlog safely.

2. Communications used masked ticket references where applicable.

3. Reproduce the failing call on a safe test entity.

3.1 Use read-only token; capture status code and latency.

3.2 Record request ID/correlation ID for log search.

4. Check API gateway/service health around the incident window.

5. Review auth and permissions for the calling user/client.

5.1 Refresh/rotate token; confirm scopes.

5.2 Verify NTP/clock skew; resync if >2 seconds.

**Verification**

1. Re-run the endpoint; expect 2xx and correct JSON fields.

2. Observe error rate and latency for 30 minutes; under baseline.

3. Attach request/response evidence to the case.

**EDI: Time zone drift caused eventTime to serialize in UTC+0 for partner PARTNER\_ID, while TOS computed local UTC+8.**

|  |  |
| --- | --- |
| Module | EDI/API |

**Overview**

Time zone drift caused eventTime to serialize in UTC+0 for partner PARTNER\_ID, while TOS computed local UTC+8. Container CONTAINER\_ID milestones appeared out-of-order on the portal; ANSI X12 315 'Gate Out' overshadowed later 'Loaded' event. Booking BOOKING\_ID not impacted operationally but customer view inconsistent.

**Resolution**

1. Patched mapper to respect NAD roles; added tests for role inversion; flushed CDN.

2. Sent sanitized incident summary to stakeholders; identifiers masked.

3. Communications used masked ticket references where applicable.

4. Locate the problematic EDI message/file and quarantine it.

5. Validate structure and partner-specific rules.

5.1 Run schema validator (segments/qualifiers/max-occurrence).

5.2 Fix qualifier mapping or split oversized occurrences.

6. Correct normalization/mapping and reprocess safely.

6.1 Use idempotency token to prevent duplicates.

6.2 Re-ingest from quarantine and monitor translator logs.

**Verification**

1. Confirm next run processed successfully; no duplicates.

2. Verify partner ACK/CONTRL is archived.

3. Attach validator logs and sanitized sample.

**EDI: Unexpected qualifier 'QUALIFIER' in EQD segment from PARTNER\_ID; schema validation failed; containers in the message not reflected on the portal timeline.**

|  |  |
| --- | --- |
| Module | EDI/API |

**Overview**

Unexpected qualifier 'QUALIFIER' in EQD segment from PARTNER\_ID; schema validation failed; containers in the message not reflected on the portal timeline. Context tokens: CONTEXT\_TOKEN1,

**Resolution**

1. Investigate Unexpected Qualifier:

1.1 Followed EDI Translation & Quarantine process to address schema validation failure.

1.2 Enabled conflict resolution, patched translator for partner tolerance, and reprocessed from quarantine to synchronize data.

2. Enabled conflict resolution preferring max(eventTime); patched translator for partner-specific tolerance; reprocessed from quarantine; verified pipeline end-to-end.

2.1 Use idempotency token to prevent duplicates.

2.2 Re-ingest from quarantine and monitor translator logs.

3. Communications used masked ticket references where applicable.

4. Locate the problematic EDI message/file and quarantine it.

5. Validate structure and partner-specific rules.

5.1 Run schema validator (segments/qualifiers/max-occurrence).

5.2 Fix qualifier mapping or split oversized occurrences.

6. Correct normalization/mapping and reprocess safely.

6.1 Use idempotency token to prevent duplicates.

6.2 Re-ingest from quarantine and monitor translator logs.

**Verification**

1. Confirm next run processed successfully; no duplicates.

2. Verify partner ACK/CONTRL is archived.

3. Attach validator logs and sanitized sample.

**API: OAuth token rejection spikes on '/containers/search'**

|  |  |
| --- | --- |
| Module | EDI/API |

**Overview**

OAuth token rejection spikes on '/containers/search'. Clock skew across nodes yielded 401/403 bursts; clients observed partial responses before circuit opened.

**Resolution**

1. Introduced per-partner rate shaping; added idempotency keys for webhook retries; validated TLS and cipher compatibility; published sanitized RCA.

2. Communications used masked ticket references where applicable.

3. Reproduce the failing call on a safe test entity.

3.1 Use read-only token; capture status code and latency.

3.2 Record request ID/correlation ID for log search.

4. Check API gateway/service health around the incident window.

5. Review auth and permissions for the calling user/client.

5.1 Refresh/rotate token; confirm scopes.

5.2 Verify NTP/clock skew; resync if >2 seconds.

**Verification**

1. Re-run the endpoint; expect 2xx and correct JSON fields.

2. Observe error rate and latency for 30 minutes; under baseline.

3. Attach request/response evidence to the case.

**ADI: Time zone drift caused eventTime to serialize in UTC+0 for partner PARTNER\_ID, while TOS computed local UTC+8.**

|  |  |
| --- | --- |
| Module | EDI/API |

**Overview**

Time zone drift caused eventTime to serialize in UTC+0 for partner PARTNER\_ID, while TOS computed local UTC+8. Container CONTAINER\_ID milestones appeared out-of-order on the portal; EDIFACT COARRI 'Gate Out' overshadowed later 'Loaded' event. Booking BOOKING\_ID not impacted operationally but customer view inconsistent. Context tokens: CONTEXT\_TOKEN1, CONTEXT\_TOKEN2.

**Resolution**

1. Patched mapper to respect NAD roles; added tests for role inversion; flushed CDN.

2. Sent sanitized incident summary to stakeholders; identifiers masked.

3. Communications used masked ticket references where applicable.

4. Locate the problematic EDI message/file and quarantine it.

5. Validate structure and partner-specific rules.

5.1 Run schema validator (segments/qualifiers/max-occurrence).

5.2 Fix qualifier mapping or split oversized occurrences.

6. Correct normalization/mapping and reprocess safely.

6.1 Use idempotency token to prevent duplicates.

6.2 Re-ingest from quarantine and monitor translator logs.

**Verification**

1. Confirm next run processed successfully; no duplicates.

2. Verify partner ACK/CONTRL is archived.

3. Attach validator logs and sanitized sample.

**API: Rate limiter throttled legitimate traffic after load surge**

|  |  |
| --- | --- |
| Module | EDI/API |

**Overview**

Rate limiter throttled legitimate traffic after load surge. Hot keys on '/auth/token' created cache stampede; error budget burned for the window.

**Resolution**

1. Increased timeouts, tuned exponential backoff, enabled request coalescing.

2. Rotated OAuth keys and enforced ±60s token leeway.

2.1 Refresh/rotate token; confirm scopes.

2.2 Verify NTP/clock skew; resync if >2 seconds.

3. Warmed cache for hot routes and raised SLI alerts.

3.1 Check DB slowest 5% of requests latency; look for lock waits.

3.2 Purge problematic cache keys; warm hot keys if needed.

4. Communications used masked ticket references where applicable.

5. Reproduce the failing call on a safe test entity.

5.1 Use read-only token; capture status code and latency.

5.2 Record request ID/correlation ID for log search.

6. Check API gateway/service health around the incident window.

7. Review auth and permissions for the calling user/client.

7.1 Refresh/rotate token; confirm scopes.

7.2 Verify NTP/clock skew; resync if >2 seconds.

**Verification**

1. Re-run the endpoint; expect 2xx and correct JSON fields.

2. Observe error rate and latency for 30 minutes; under baseline.

3. Attach request/response evidence to the case.

**EDI: Spike in DLQ messages after routine maintenance; consumer group lag increased across EDI topic**

|  |  |
| --- | --- |
| Module | EDI/API |

**Overview**

Spike in DLQ messages after routine maintenance; consumer group lag increased across EDI topic. Partner PARTNER\_ID reported delayed acknowledgements; LOCATION ops not impacted.

**Resolution**

1. Corrected cron schedule to UTC; added mutual exclusion lock and run-idempotency; published sanitized incident note and fixed dashboards.

2. Communications used masked ticket references where applicable.

3. Locate the problematic EDI message/file and quarantine it.

4. Validate structure and partner-specific rules.

4.1 Run schema validator (segments/qualifiers/max-occurrence).

4.2 Fix qualifier mapping or split oversized occurrences.

5. Correct normalization/mapping and reprocess safely.

5.1 Use idempotency token to prevent duplicates.

5.2 Re-ingest from quarantine and monitor translator logs.

**Verification**

1. Confirm next run processed successfully; no duplicates.

2. Verify partner ACK/CONTRL is archived.

3. Attach validator logs and sanitized sample.

**EDI: Shipper/consignee role swap observed on API payload for CONTAINER\_ID**

|  |  |
| --- | --- |
| Module | EDI/API |

**Overview**

Shipper/consignee role swap observed on API payload for CONTAINER\_ID. Upstream denormalization reversed NAD qualifiers; billing prep failed validation. Partner: PARTNER\_ID; affected terminal: LOCATION. Context tokens: CONTEXT\_TOKEN1.

**Resolution**

1. Patched mapper to respect NAD roles; added tests for role inversion; flushed CDN.

2. Sent sanitized incident summary to stakeholders; identifiers masked.

3. Communications used masked ticket references where applicable.

4. Reproduce the failing call on a safe test entity.

4.1 Use read-only token; capture status code and latency.

4.2 Record request ID/correlation ID for log search.

5. Check API gateway/service health around the incident window.

6. Review auth and permissions for the calling user/client.

6.1 Refresh/rotate token; confirm scopes.

6.2 Verify NTP/clock skew; resync if >2 seconds.

**Verification**

1. Re-run the endpoint; expect 2xx and correct JSON fields.

2. Observe error rate and latency for 30 minutes; under baseline.

3. Attach request/response evidence to the case.

**API: Unexpected qualifier 'QUALIFIER' in EQD segment from PARTNER\_ID; schema validation failed; containers in the message not reflected on the portal timeline.**

|  |  |
| --- | --- |
| Module | EDI/API |

**Overview**

Unexpected qualifier 'QUALIFIER' in EQD segment from PARTNER\_ID; schema validation failed; containers in the message not reflected on the portal timeline.

**Resolution**

1. Implemented pre-ingest normalizer for qualifiers; updated partner IG notes; enabled anomaly detection for repeated segment errors exceeds threshold.

1.1 Run schema validator (segments/qualifiers/max-occurrence).

1.2 Fix qualifier mapping or split oversized occurrences.

2. Communications used masked ticket references where applicable.

3. Locate the problematic EDI message/file and quarantine it.

4. Validate structure and partner-specific rules.

4.1 Run schema validator (segments/qualifiers/max-occurrence).

4.2 Fix qualifier mapping or split oversized occurrences.

5. Correct normalization/mapping and reprocess safely.

5.1 Use idempotency token to prevent duplicates.

5.2 Re-ingest from quarantine and monitor translator logs.

**Verification**

1. Confirm next run processed successfully; no duplicates.

2. Verify partner ACK/CONTRL is archived.

3. Attach validator logs and sanitized sample.

**API: Time zone drift caused eventTime to serialize in UTC+0 for partner PARTNER\_ID, while TOS computed local UTC+8.**

|  |  |
| --- | --- |
| Module | EDI/API |

**Overview**

Time zone drift caused eventTime to serialize in UTC+0 for partner PARTNER\_ID, while TOS computed local UTC+8. Container CONTAINER\_ID milestones appeared out-of-order on the portal; BAPLIE 'Gate Out' overshadowed later 'Loaded' event. Booking BOOKING\_ID not impacted operationally but customer view inconsistent. Context tokens: CONTEXT\_TOKEN.

**Resolution**

1. Investigate Time Zone Drift:

1.1 Executed Rate Limiting & Traffic Shaping to address eventTime serialization issue.

1.2 Set precedence to TOS for equipment attributes, purged cache, and re-indexed data to synchronize timestamps.

2. Set precedence to TOS for equipment attributes; forced cache purge and re-index.

3. Backfilled corrected attributes, then locked fields for 6h to prevent drift.

3.1 Archive ACK with timestamp and execution ID.

3.2 Notify partner with sanitized sample if needed.

4. Added rule to reject backdated EDI updates if newer move exists within 120 minutes.

4.1 Archive ACK with timestamp and execution ID.

4.2 Notify partner with sanitized sample if needed.

5. Communications used masked ticket references where applicable.

6. Locate the problematic EDI message/file and quarantine it.

7. Validate structure and partner-specific rules.

7.1 Run schema validator (segments/qualifiers/max-occurrence).

7.2 Fix qualifier mapping or split oversized occurrences.

8. Correct normalization/mapping and reprocess safely.

8.1 Use idempotency token to prevent duplicates.

8.2 Re-ingest from quarantine and monitor translator logs.

**Verification**

1. Confirm next run processed successfully; no duplicates.

2. Verify partner ACK/CONTRL is archived.

3. Attach validator logs and sanitized sample.

**EDIE Translator rejected BAPLIE: missing BGM count alignment (UNH/UNT mismatch)**

|  |  |
| --- | --- |
| Module | EDI/API |

**Overview**

Translator rejected BAPLIE: missing BGM count alignment (UNH/UNT mismatch). Partner PARTNER\_ID interchange quarantined; downstream milestone not created.

**Resolution**

1. Implemented pre-ingest normalizer for qualifiers; updated partner IG notes; enabled anomaly detection for repeated segment errors exceeds threshold.

1.1 Run schema validator (segments/qualifiers/max-occurrence).

1.2 Fix qualifier mapping or split oversized occurrences.

2. Communications used masked ticket references where applicable.

3. Locate the problematic EDI message/file and quarantine it.

4. Validate structure and partner-specific rules.

4.1 Run schema validator (segments/qualifiers/max-occurrence).

4.2 Fix qualifier mapping or split oversized occurrences.

5. Correct normalization/mapping and reprocess safely.

5.1 Use idempotency token to prevent duplicates.

5.2 Re-ingest from quarantine and monitor translator logs.

**Verification**

1. Confirm next run processed successfully; no duplicates.

2. Verify partner ACK/CONTRL is archived.

3. Attach validator logs and sanitized sample.

**EDI: Field mapping mismatch across systems for container CONTAINER\_ID**

|  |  |
| --- | --- |
| Module | EDI/API |

**Overview**

Field mapping mismatch across systems for container CONTAINER\_ID. Partner PARTNER\_ID posted ANSI X12 301 with size/type as SIZE\_TYPE, but our master lists SIZE\_TYPE2. Public API reflected partner payload, breaking yard slotting at LOCATION. Booking BOOKING\_ID shows reefer flag OFF while TOS maintains ON. Audit indicates stale backfill job re-applied older attributes.

**Resolution**

1. Investigate Field Mapping Mismatch:

1.1 Executed Partner IG Variance Handling to resolve mapping discrepancies.

1.2 Set precedence to TOS for equipment attributes, purged cache, and re-indexed data to synchronize system values.

2. Set precedence to TOS for equipment attributes; forced cache purge and re-index.

2.1 Check DB slowest 5% of requests latency; look for lock waits.

2.2 Purge problematic cache keys; warm hot keys if needed.

3. Backfilled corrected attributes, then locked fields for 6h to prevent drift.

4. Added rule to reject backdated EDI updates if newer move exists within 120 minutes.

5. Communications used masked ticket references where applicable.

6. Reproduce the failing call on a safe test entity.

6.1 Use read-only token; capture status code and latency.

6.2 Record request ID/correlation ID for log search.

7. Check API gateway/service health around the incident window.

8. Review auth and permissions for the calling user/client.

8.1 Refresh/rotate token; confirm scopes.

8.2 Verify NTP/clock skew; resync if >2 seconds.

**Verification**

1. Re-run the endpoint; expect 2xx and correct JSON fields.

2. Observe error rate and latency for 30 minutes; under baseline.

3. Attach request/response evidence to the case.

**API: ANSI X12 315 duplicate detected with conflicting control numbers; later message had older eventTime**

|  |  |
| --- | --- |
| Module | EDI/API |

**Overview**

ANSI X12 315 duplicate detected with conflicting control numbers; later message had older eventTime. Quarantine spiked; publish paused to avoid regression.

**Resolution**

1. Investigate EDI Duplicate:

1.1 Followed Token Refresh & Clock Skew Handling to resolve duplicate issue.

1.2 Enabled conflict resolution, preferring max(eventTime), and reprocessed from quarantine to ensure data accuracy.

2. Enabled conflict resolution preferring max(eventTime); patched translator for partner-specific tolerance; reprocessed from quarantine; verified pipeline end-to-end.

2.1 Use idempotency token to prevent duplicates.

2.2 Re-ingest from quarantine and monitor translator logs.

3. Communications used masked ticket references where applicable.

4. Locate the problematic EDI message/file and quarantine it.

5. Validate structure and partner-specific rules.

5.1 Run schema validator (segments/qualifiers/max-occurrence).

5.2 Fix qualifier mapping or split oversized occurrences.

6. Correct normalization/mapping and reprocess safely.

6.1 Use idempotency token to prevent duplicates.

6.2 Re-ingest from quarantine and monitor translator logs.

**Verification**

1. Confirm next run processed successfully; no duplicates.

2. Verify partner ACK/CONTRL is archived.

3. Attach validator logs and sanitized sample.

**API: Time zone drift caused eventTime to serialize in UTC+0 for partner PARTNER\_ID, while TOS computed local UTC+8.**

|  |  |
| --- | --- |
| Module | EDI/API |

**Overview**

Time zone drift caused eventTime to serialize in UTC+0 for partner PARTNER\_ID, while TOS computed local UTC+8. Container CONTAINER\_ID milestones appeared out-of-order on the portal; EDIFACT COARRI 'Gate Out' overshadowed later 'Loaded' event. Booking BOOKING\_ID not impacted operationally but customer view inconsistent.

**Resolution**

1. Patched mapper to respect NAD roles; added tests for role inversion; flushed CDN.

2. Sent sanitized incident summary to stakeholders; identifiers masked.

3. Communications used masked ticket references where applicable.

4. Locate the problematic EDI message/file and quarantine it.

5. Validate structure and partner-specific rules.

5.1 Run schema validator (segments/qualifiers/max-occurrence).

5.2 Fix qualifier mapping or split oversized occurrences.

6. Correct normalization/mapping and reprocess safely.

6.1 Use idempotency token to prevent duplicates.

6.2 Re-ingest from quarantine and monitor translator logs.

**Verification**

1. Confirm next run processed successfully; no duplicates.

2. Verify partner ACK/CONTRL is archived.

3. Attach validator logs and sanitized sample.

**CNTR: Duplicate Container information received**

|  |  |
| --- | --- |
| Module | Container |

**Overview**

Received with conflicting duplicate entry for container information. Later message had older timestamp; last-writer-wins regressed status.

**Resolution**

1. Verify in DB
   1. SELECT \* FROM container WHERE cntr\_no = :CONTAINER\_NO ORDER BY created\_at DESC;
2. Keep the latest record (highest created\_at) and delete earlier records — **with a preview and a safe, bounded delete**.
   1. SELECT c.\* FROM container c JOIN ( SELECT cntr\_no, vessel\_id, eta\_ts, MAX(created\_at) AS max\_created\_at FROM container WHERE cntr\_no = :CONTAINER\_NO AND vessel\_id = :VESSEL\_ID AND eta\_ts = :ETA\_TS GROUP BY cntr\_no, vessel\_id, eta\_ts ) keep ON keep.cntr\_no=c.cntr\_no AND keep.vessel\_id=c.vessel\_id AND keep.eta\_ts=c.eta\_ts WHERE c.created\_at < keep.max\_created\_at;
3. Safe delete (keep latest only)
   1. DELETE c FROM container c JOIN ( SELECT cntr\_no, vessel\_id, eta\_ts, MAX(created\_at) AS max\_created\_at FROM container WHERE cntr\_no = :CONTAINER\_NO AND vessel\_id = :VESSEL\_ID AND eta\_ts = :ETA\_TS GROUP BY cntr\_no, vessel\_id, eta\_ts ) keep ON keep.cntr\_no=c.cntr\_no AND keep.vessel\_id=c.vessel\_id AND keep.eta\_ts=c.eta\_ts WHERE c.created\_at < keep.max\_created\_at;

**VAS: VESSEL\_ERR\_4 Vessel Name has been used by other vessel advice**

|  |  |
| --- | --- |
| Module | Vessel |

**Overview**

Resolve duplicate key error on creating vessel advice VESSEL\_ERR\_4 (only one active advice per system\_vessel\_name).

**Preconditions**

* You have access to DB schema with tables vessel\_advice and berthing\_application
* Log file: vessel\_advice\_service.log

**Resolution**

1. Confirm Error in Logs
2. grep -i "VESSEL\_ERR\_4" vessel\_advice\_service.log
3. Inspect Existing Advice Rows
   1. SELECT vessel\_advice\_no, system\_vessel\_name, effective\_start\_datetime, effective\_end\_datetime, system\_vessel\_name\_active FROM vessel\_advice WHERE system\_vessel\_name = :system\_vessel\_name ORDER BY effective\_start\_datetime;
   2. identify active advice: row with effective\_end\_datetime IS NULL.
   3. Record down
      1. :active\_vessel\_advice\_no (if any)
4. Check Port Programs Referencing Active Advice (Active only)
   1. SELECT application\_no, vessel\_advice\_no, vessel\_close\_datetime, deleted, berthing\_status FROM berth\_application WHERE vessel\_advice\_no = :active\_vessel\_advice\_no AND vessel\_close\_datetime IS NULL AND berthing\_status = 'A' AND deleted = 'N';
   2. If result set not empty → active port programs exist.

**Decision** **Logic**

* If NO active advice (no row with effective\_end\_datetime IS NULL): the duplicate error shouldn’t occur—recheck.
* Active advice exists & NO active port programs: proceed to expire advice.
* Active advice exists & active port programs present: close/archive them first, then expire advice.

**Close Active Port Programs**

Define closure timestamp (prefer consistent UTC: e.g. '2025-10-06 00:00:00').

Status code definitions (confirm internally):

* berthing\_status: 'A' = Active, 'C' = Closed
* Deleted: 'N' = Normal, 'A' = Archived/Removed

UPDATE berth\_application SET vessel\_close\_datetime = :CLOSE\_TS, berthing\_status = 'C', deleted = 'A' WHERE vessel\_advice\_no = :vessel\_advice\_no AND vessel\_close\_datetime IS NULL AND berthing\_status = 'A' AND deleted = 'N';

* Expire the Active Advice

UPDATE vessel\_advice SET effective\_end\_datetime = :CLOSE\_TS WHERE vessel\_advice\_no = :vessel\_advice\_no AND effective\_end\_datetime IS NULL;

**EDI: EDI Message Timeout or Delay in Acknowledgment**

|  |  |
| --- | --- |
| Module | EDI/API |

**Overview**

Rough Flow of the process: Message Parsed → Acknowledgment Expected → No Acknowledgment → Message Stuck in ERROR Status → Check Logs → Inspect EDI Messages → Manually Trigger Acknowledgment → Message Status Updated to ACKED → Issue Resolved

The issue occurs when an EDI message (e.g., IFTMIN) is received and processed (status: PARSED) but is not properly acknowledged. This can be due to server delays, network issues, or other factors. As a result, the message remains in the ERROR status and doesn’t have the expected ack\_at timestamp, preventing it from being processed further, which disrupts data flow and shipment tracking.

For example, if PSA receives an IFTMIN message from LINE-PSA but the acknowledgment isn't sent to the sender (PSA-TOS), the message stays in ERROR status with a NULL value for ack\_at. The lack of acknowledgment means that the message isn’t processed or moved forward.

**Preconditions**

* Access to the EDI message table (edi\_message) in the database schema is available.
* The message has been received and processed (status: PARSED), but no acknowledgment has occurred yet.
* The message is in the ERROR status and doesn’t have the correct ack\_at timestamp. Specifically, it has the value NULL in ack\_at.
* Log file: edi\_message\_processing.log

**Resolution**

1. Confirm Error in Logs (Check the logs for any entries related to the specific message (e.g., REF-IFT-0007) to verify if the acknowledgment process was delayed or failed).
2. grep -i "REF-IFT-N" edi\_message\_processing.log
3. Inspect Existing Messages
4. Run a query on the edi\_message table to find any messages that either:  
   - Have the ERROR status or   
   - Have been received and processed (status: PARSED) but haven't been properly acknowledged (i.e., ack\_at is NULL).

For example, if a message was received by LINE-PSA and sent to PSA-TOS, but ack\_at is NULL, it indicates that the message was parsed but wasn't properly acknowledged, leading to the ERROR status.

1. Run a query on the edi\_message table to find any messages that have been acknowledged (status: ACKED) but have not been properly acknowledged (ack\_at is at NULL) after 24 hours.
2. SELECT edi\_id, message\_ref, sender, receiver, status, sent\_at, ack\_at, error\_text  
   FROM edi\_message  
   WHERE status = 'ERROR' OR (status = 'PARSED' AND ack\_at IS NULL);

#### **Decision Logic**

* If NO active message (no message stuck in ERROR status with missing ack\_at):
  + The timeout issue shouldn't occur. Recheck the acknowledgment logic or network connection to confirm if any messages were delayed.
* If active message exists and no acknowledgment (message is stuck in ACKED status but no proper acknowledgment):
  + You can manually trigger acknowledgment for the message, ensuring it gets an appropriate ack\_at timestamp
* If acknowledgment failed (even though the message was attempted for acknowledgment):
  + You must process the message again or trigger acknowledgment manually as described below.

#### **Resolve the Timeout**

1. Manually trigger the acknowledgment process for messages stuck in the ERROR status but not properly acknowledged. Update the status and set the ack\_at timestamp.
   1. -- Trigger ACKED status update manually  
      UPDATE edi\_message  
      SET status = 'ACKED', ack\_at = NOW(), error\_text = NULL  
      WHERE edi\_id = <edi\_id>; -- Replace with the actual edi\_id (1, 2, 3, 4, n)