**🟢 Jira Task – RL1.1 Ingestion Test Script Development & Execution**

**Description**  
As part of Radar Live SIT scope, the BICOE QA team will design and execute test scripts to validate ingestion of messages into the Azure Data Platform as per the ingestion requirements documented in Confluence.  
The task covers:

* Ingestion validation across Landing (DLT) → DV (Silverish layer).
* Data quality validation scenarios for ingestion tables.
* Test data coverage to ensure representative records for all ingestion paths.
* Technical table-level checks (schema, constraints, datatypes, field mapping).
* Reconciliation against the source payloads.
* Correlation and reconciliation against interaction data for full traceability (Quote ID, Cache ID, PRN).

**Validation Scope**

* Validate ingestion mapping for all required message types (Quote Request, Rating Response, Cache ID, Stop Quote, PreComp, Rating Request).
* Ensure all data quality checks are designed and executed (null handling, data type mismatches, mandatory vs optional fields).
* Confirm test data is available for all ingestion tables, with coverage across NB, Requote, MTA, Renewal, and Decline flows.
* Perform reconciliation between source event hub payloads and ingested records.
* Perform correlation checks across ingestion tables and interaction data for linkage (Quote → Rating → Feature Set → Core Risk).
* Validate CDC handling and schema drift changes (pre/post change scenarios).
* Ensure ingestion aligns with requirements specified in Confluence for RL1.1.

**Acceptance Criteria / Testing Requirements**

* ✅ Test scripts created for all ingestion message types.
* ✅ Ingestion mapping validated against Confluence requirements.
* ✅ All ingestion-related data quality scenarios executed with documented results.
* ✅ Test data coverage confirmed for all ingestion tables across quote journeys (NB, Requote, MTA, Renewal).
* ✅ Technical table-level validations executed (schema, datatype, key constraints, joins).
* ✅ Successful reconciliation of ingested data with source payloads.
* ✅ Successful correlation and reconciliation across interaction data (end-to-end quote flow traceability).
* ✅ CDC and schema drift validation completed.
* ✅ All high/critical defects tracked and closed before SIT exit.

**Dependencies**

* Event hub details for sample test data (with Nate).
* Schema drift change request analysis (with Mike).
* Business alignment on pre- and post-schema drift testing scope (with Pricing QA / Rahul).
* Continuity of Service scenarios for in-scope views (Shashi with Paul).
* Dynamic Pricing & Radar results requirement clarification (with Nikki).
* DevOps/IT support for Landing → DV ingestion jobs.

**Definition of Done (DoD)**

* All ingestion scripts designed, reviewed, and executed.
* Data quality, reconciliation, and correlation test scenarios completed with evidence.
* Ingestion mapping validated end-to-end with no critical gaps.
* Reconciliation results documented and signed off.
* CDC/schema drift handling validated with test results attached.
* All defects triaged, retested, and closed or accepted with mitigation.
* SIT sign-off note shared with BICOE QA Lead.

# ----------------------------------------------------------------------------------------------------------- 🟢 Jira Task – RL1.3 DVLA Data Retention Validation

**Description**  
As part of SIT for Radar Live, BICOE QA will validate the **DVLA Data Retention implementation** (RL1.3) against the defined organisational policy for data storage and deletion.  
Testing will be executed in the **time travel environment**, ensuring compliance with retention requirements (storage duration, deletion, archiving) across different scenarios.  
This requires close **coordination with Source Athena, E2E team, and BICOE Data Engineering** to confirm correct design, ingestion, and lifecycle handling.  
The correct set of **test data scenarios** must be included in the Test Data Scenario Sheet to support coverage.

**Validation Scope**

* Validate that DVLA data is ingested, stored, and retained according to organisational retention rules.
* Validate **time travel queries** for DVLA data across multiple points (T0, T1, T2) to confirm correct retention and deletion.
* Test **retention scenarios**:
  + Active storage duration.
  + Scheduled deletion after retention period.
  + Archival handling (if applicable).
  + Restricted access after expiry.
* Confirm alignment of retention with **DVLA requirements** and internal compliance policy.
* Reconcile BICOE DVLA tables with **source Athena** to ensure correct retention implementation.
* Validate correlation with E2E interaction data to ensure deleted/expired DVLA data is not visible in reporting or downstream usage.
* Ensure scenarios are tracked in **test data scenario sheet** for full traceability.

**Acceptance Criteria / Testing Requirements**

* ✅ DVLA data storage validated in SIT against retention requirements.
* ✅ Time travel environment confirms retention rules (historical vs expired data).
* ✅ Deletion/expiry rules correctly applied as per DVLA requirement.
* ✅ Data not accessible after expiry, confirming policy enforcement.
* ✅ Reconciliation completed with source Athena system for retention handling.
* ✅ E2E validation confirms expired DVLA data not propagated downstream.
* ✅ Test evidence attached in Jira for all scenarios.
* ✅ No open high/critical issues at SIT exit.

**Dependencies**

* Time travel environment setup (DevOps/IT).
* Retention policy clarification from Architecture/Compliance team.
* Source Athena support for reconciliation snapshots.
* E2E team alignment on downstream validation of expired DVLA data.
* BICOE Data Engineering support for table design and lifecycle configuration.
* Test data preparation covering multiple retention scenarios.

**Definition of Done (DoD)**

* All DVLA retention scenarios validated (storage, deletion, access restriction).
* Source vs BICOE reconciliation completed and signed off.
* Time travel queries executed with evidence attached.
* Expired data not visible in SIT downstream validation.
* Compliance with DVLA and organisational retention rules confirmed.
* Test data scenario sheet updated and approved.
* SIT sign-off note provided to stakeholders.

**🟢 Jira Task – RL1.4 Raw Enrichment Data Retention Validation**

**Description**  
As part of SIT for Radar Live, BICOE QA will validate the **Raw Enrichment Data Retention implementation** (RL1.4) against organisational policies for storage and deletion.  
Testing will leverage the **time travel environment** to validate lifecycle rules across ingestion, storage, and expiry for Raw Enrichment datasets.  
Validation will also require **coordination with Source Athena, E2E team, and BICOE Data Engineering** to ensure correct test data preparation, mapping, and alignment with data governance requirements.  
The correct set of **test data scenarios** must be included in the Test Data Scenario Sheet for SIT execution and compliance tracking.

**Validation Scope**

* Validate ingestion and storage of Raw Enrichment data into DV/EDM layers.
* Validate **time travel queries** to confirm historical records are maintained and expired data is correctly purged.
* Test retention lifecycle scenarios:
  + Active storage period validation.
  + Deletion after expiry in accordance with retention rules.
  + Archival or restricted access handling (if applicable).
* Confirm expired Raw Enrichment data is not visible in downstream reporting or analytics.
* Reconcile **source Athena Raw Enrichment data vs BICOE tables** to confirm correct retention implementation.
* Validate that correlation with interaction data remains intact for valid records and is removed for expired ones.
* Ensure retention test scenarios are captured in the **Test Data Scenario Sheet** with clear mapping to requirements.

**Acceptance Criteria / Testing Requirements**

* ✅ Raw Enrichment data lifecycle validated (storage, expiry, deletion).
* ✅ Time travel environment queries confirm historical vs expired states.
* ✅ Expiry/deletion rules applied per organisational and Raw Enrichment design.
* ✅ Reconciliation completed with source Athena for retention handling.
* ✅ Expired Raw Enrichment data not propagated to reporting/EDW.
* ✅ Technical table-level validation completed for schema and lifecycle metadata.
* ✅ Test evidence captured and attached in Jira.
* ✅ All high/critical issues resolved before SIT closure.

**Dependencies**

* Time travel environment setup (DevOps/IT).
* Raw Enrichment retention requirement confirmation with Data Governance team.
* Source Athena Raw Enrichment snapshots for reconciliation.
* E2E team validation to confirm expired data is excluded from downstream processes.
* BICOE Data Engineering support for retention configuration and validation.
* Test data preparation covering lifecycle (active, expired, archived).

**Definition of Done (DoD)**

* Raw Enrichment retention validated across ingestion lifecycle.
* Source vs BICOE reconciliation completed with evidence.
* Time travel validation executed and signed off.
* Expired Raw Enrichment records confirmed as inaccessible in SIT.
* Test data scenario sheet updated to include Raw Enrichment retention cases.
* Compliance with organisational retention policy confirmed.
* SIT sign-off note delivered to stakeholders.

**🟢 Jira Task – RL2.1 Decompress Files and Store within Raw Enrichment Area**

**Description**  
As part of SIT for Radar Live, BICOE QA will validate the **decompression and storage process for Raw Enrichment files** (RL2.1).  
The validation will focus on ensuring that compressed files are correctly decompressed, stored within the Raw Enrichment area, and linked with the correct **Key Identifier** to enable traceability and downstream processing.  
Testing scope covers ingestion pipeline checks, data quality verification, and reconciliation with source file structures.

**Validation Scope**

* Validate that compressed Raw Enrichment files are successfully decompressed during ingestion.
* Ensure decompressed files are stored correctly within the **Raw Enrichment area**.
* Confirm that each record is assigned the correct **Key Identifier** for linkage with Quote/Interaction data.
* Validate schema structure post-decompression (field mapping, datatype integrity, null handling).
* Perform **data quality checks** on decompressed data (completeness, duplicates, truncation).
* Validate reconciliation between **original compressed file content vs decompressed stored content**.
* Ensure records are visible in downstream DV/EDM layers with correct Key Identifier linkage.

**Acceptance Criteria / Testing Requirements**

* ✅ Decompression process validated successfully across multiple Raw Enrichment files.
* ✅ Records stored in Raw Enrichment area match source file content post-decompression.
* ✅ Key Identifier applied correctly and consistently across all records.
* ✅ Schema and datatype integrity maintained post-decompression.
* ✅ No loss of data, truncation, or duplicates after decompression.
* ✅ Reconciliation evidence attached (source compressed file vs decompressed output).
* ✅ Downstream linkage with Quote ID/Interaction ID confirmed.
* ✅ No open high/critical defects before SIT sign-off.

**Dependencies**

* Source system to provide sample compressed Raw Enrichment files.
* DevOps/IT setup for decompression process and Raw Enrichment storage path.
* Confirmation of Key Identifier rules (mapping, uniqueness, mandatory fields) from Data Engineering/Architecture.
* Coordination with E2E team for downstream validation.
* Test data preparation to include multiple compressed file scenarios (valid, corrupt, large file).

**Definition of Done (DoD)**

* Decompression validated end-to-end from source file to Raw Enrichment storage.
* Key Identifier assignment confirmed against design specification.
* Schema validation and data quality checks completed with evidence.
* Reconciliation completed between compressed source file and decompressed Raw Enrichment data.
* No high-severity defects open.
* SIT sign-off note shared with stakeholders.

**🟢 Jira Task – RL2.2 Create/Identify PV Groups for Access Control**

**Description**  
As part of SIT for Radar Live, BICOE QA will validate the creation and identification of **PV (Permission/Privilege) groups** for controlled access to data within the Azure Data Platform.  
The groups must provide appropriate access segregation for **Pricing, Engineering, and Underwriting teams**, ensuring compliance with data governance and security requirements.  
Testing scope will cover access control validation, role-based data visibility, and negative testing to confirm restricted data is not accessible by unauthorized groups.

**Validation Scope**

* Validate that PV groups are correctly created/identified for:
  + **Pricing team** (access to pricing data and views).
  + **Engineering team** (access to ingestion pipeline and technical views).
  + **Underwriting team** (access to underwriting-related reporting and data).
* Confirm **role-based access control (RBAC)** is applied according to organisational policy.
* Validate that each PV group can access only its intended datasets and views.
* Perform negative testing → ensure groups cannot access datasets outside their scope.
* Validate that **audit logs** capture access events for compliance.
* Reconcile test data scenarios across groups to ensure consistency in permissions.

**Acceptance Criteria / Testing Requirements**

* ✅ PV groups created/identified for Pricing, Engineering, and Underwriting teams.
* ✅ Access control validated for each group against in-scope datasets.
* ✅ Negative testing confirms access restrictions are enforced.
* ✅ Data governance and compliance requirements met (no PII leakage outside allowed groups).
* ✅ Evidence of role-based access validation attached in Jira.
* ✅ Audit logs confirm proper capture of access attempts.
* ✅ No open high/critical security defects before SIT closure.

**Dependencies**

* Confirmation of **access policy definitions** (who can access what) from Data Governance/Architecture.
* DevOps/IT setup of PV groups within Azure environment.
* Test data preparation covering Pricing, Engineering, and Underwriting datasets.
* Coordination with respective business teams to validate real-world access needs.
* Security/compliance team sign-off on RBAC setup.

**Definition of Done (DoD)**

* PV groups created and validated against requirement.
* Role-based access testing executed with positive/negative scenarios.
* Audit trail validated for access compliance.
* Evidence captured and attached in Jira.
* No high-severity access/security issues open.
* SIT sign-off note shared with stakeholders.

.

# 🟢 ****Jira Task – RL3.2 Populate Core Risk Structure (Silver Layer – Data Vault)****

**Description**  
As part of SIT for Radar Live, BICOE QA will validate the implementation of the **Core Risk structure** to support PreComp Features and Motor Rating Factor views within the Silver Layer (Data Vault).  
The validation scope includes parsing Core Risk into structured Data Vault entities (Hub, Satellite, Link), validating multiple insured objects per quote, ensuring referential integrity, CDC handling, and lineage tracking.

**Validation Scope**

* Parse Core Risk JSON into structured **Hub, Satellite, and Link models** within the Data Vault.
* Validate support for **multiple insured objects per quote** (e.g., multiple vehicles on the same policy).
* Validate **referential integrity** at the quote level using MQS-provided **Unique Quote ID**.
* Confirm correct linking of **Interaction ID** and other related business keys.
* Validate **timestamp and versioning strategy** to support CDC and full auditability.
* Confirm **metadata lineage** is preserved for each field from **source → Data Vault**.
* Verify that enrichment logic (e.g., PreComp values, DVLA attributes) is accurately retained in the Core Risk layer.
* Perform reconciliation of Core Risk records with source MQS messages to confirm accuracy.

**Acceptance Criteria / Testing Requirements**

* ✅ Core Risk successfully parsed into structured Data Vault tables (Hub, Satellite, Link).
* ✅ Multiple insured object support validated across quotes.
* ✅ Referential integrity confirmed using Unique Quote ID.
* ✅ Interaction ID and related keys correctly linked.
* ✅ CDC/versioning tested and historical versions retrievable.
* ✅ Metadata lineage verified from MQS → DV (Silver Layer).
* ✅ Enrichment values (PreComp/DVLA) retained and accessible in Core Risk tables.
* ✅ Source vs DV reconciliation completed for Core Risk payloads.
* ✅ Test evidence attached in Jira.
* ✅ No high/critical defects remain open.

**Dependencies**

* Confirmation of Core Risk Data Vault schema design (Hub/Sat/Link model) from Data Engineering.
* MQS Core Risk sample payloads for SIT test data.
* Time travel environment setup for CDC/versioning validation.
* Coordination with Pricing QA to align on enrichment fields required for PreComp and Motor Rating Factor views.
* Input from E2E team for interaction-level correlation checks.

**Definition of Done (DoD)**

* Core Risk structure validated in Silver Layer Data Vault.
* Hub/Satellite/Link models populated and reconciled with MQS Core Risk payloads.
* CDC and audit/versioning confirmed with SIT evidence.
* Referential integrity validated across Quote ID, Interaction ID, and business keys.
* Enrichment logic confirmed as retained.
* All defects triaged, fixed, retested, and closed.
* SIT sign-off note provided to stakeholders.

# 🟢 ****Jira Task – RL4.1 Populate EDM Structures with Core Risk Data (Gold Layer – EDM – Core Risk)****

**Description**  
As part of SIT for Radar Live, BICOE QA will validate the **population of Core Risk data into existing EDM structures** to support **PreComp Features** and **Motor Rating Factor Views**.  
This validation ensures dimensional models are correctly populated, quote-level traceability is maintained, compliance requirements (e.g., DVLA, PII) are enforced, and data is aggregated correctly for downstream consumption.

**Validation Scope**

* Validate the **dimensional model representation of Core Risk** in EDM.
* Confirm EDM Core Risk objects **aggregate correctly per quote**.
* Validate **quote linking** using Cache ID, Interaction ID, and related business keys.
* Verify flags for **orphaned Core Risk records** are set (e.g., null-premium transactions).
* Confirm **quote declines** are traceable (e.g., via decline reason codes, null-premium flags).
* Validate **PII masking and RBAC** on sensitive fields (no exposure for lower-permission roles).
* Confirm **DVLA compliance** by ensuring only customer self-declared values exist; DVLA data is **not shared with Pricing team**.
* Validate **semantic correctness** of risk metrics, segmentation, and categorization in EDM views.
* Test **quote linking via Cache ID** for full quote chain traceability (original → requote → MTA).
* Validate **bound transaction linking** to support PreComp features and Motor Rating Factor views.
* Confirm **multi-vehicle quotes** are supported per record (insured object granularity).
* Validate **referential integrity** between Silver (DV) and Gold (EDM) Core Risk records.
* Negative validation → Ensure **not intended records** do not appear in EDM Core Risk views, in line with requirement documentation.

**Acceptance Criteria / Testing Requirements**

* ✅ EDM dimensional models for Core Risk populated and validated.
* ✅ Quote-level aggregation confirmed across NB, MTA, Requote, Renewal flows.
* ✅ Cache ID / Interaction ID linking verified for full quote journey traceability.
* ✅ Orphaned Core Risk flagged correctly (null-premium).
* ✅ Declines represented accurately with rule-based indicators.
* ✅ Multi-vehicle support validated in EDM structures.
* ✅ PII masking enforced and RBAC validated for sensitive fields.
* ✅ DVLA compliance confirmed (no leakage of DVLA-derived data to Pricing).
* ✅ Risk metrics and segmentations validated for semantic correctness.
* ✅ Negative scenarios validated: no unintended data present in Gold views.
* ✅ Source-to-EDM reconciliation completed with test evidence.

**Dependencies**

* EDM model design and mapping specs from Data Engineering.
* Test data scenarios covering multi-vehicle, decline, orphaned records, and quote chains (NB → MTA → Renewal).
* Confirmation of PII masking & RBAC requirements with Data Governance team.
* DVLA compliance requirements from Governance/Legal team.
* Coordination with Pricing QA for PreComp / Motor Rating Factor view consumption.
* E2E team for end-to-end traceability validation.

**Definition of Done (DoD)**

* Core Risk populated correctly into EDM and validated against Confluence requirements.
* Aggregations, linking, and risk metrics confirmed across all quote journeys.
* Compliance validated (DVLA, PII).
* Negative scenarios tested and confirmed as excluded.
* Reconciliation completed across Silver → Gold for Core Risk.
* No high/critical defects open at SIT sign-off.
* Test evidence and sign-off shared with stakeholders.

# 🟢 ****Jira Task – RL5.3 Populate Radar Feature Set with Data Vault Model (Success)****

**Description**  
As part of SIT for Radar Live, BICOE QA will validate the **population of Radar Feature Set data into the Data Vault model** (success path).  
This involves validating transformation from Bronze JSON payloads into structured, dynamic tables within the Data Vault, ensuring schema flexibility, CDC readiness, and integrity of grouped feature categories.

**Validation Scope**

* Validate transformation of Radar Feature Set JSON from Bronze → Data Vault structured tables.
* Confirm feature grouping integrity (Driver, Vehicle, Policy, Proposer, Cover, Metadata).
* Validate schema flexibility for:
  + Addition of new feature sets.
  + Embedded arrays within feature records.
* Validate CDC and versioning readiness (soft deprecation, historical record handling).
* Confirm no renaming or breaking changes in field names occur without a new version ID.
* Validate record traceability back to source MQS Radar Feature Set payloads.

**Acceptance Criteria / Testing Requirements**

* ✅ Feature Set parsed successfully into Data Vault Hub, Satellite, Link models.
* ✅ Feature categories validated for accuracy and completeness.
* ✅ New feature fields added without breaking schema.
* ✅ Embedded arrays stored correctly in Data Vault tables.
* ✅ CDC/versioning tested with historical feature set versions retained.
* ✅ No unintended renaming/breaking changes detected.
* ✅ Reconciliation with Bronze JSON source completed with test evidence.

**Dependencies**

* Radar Feature Set schema definition and mapping specs.
* Sample test data with multiple feature categories, arrays, and new fields.
* Time travel environment for CDC/versioning validation.
* Input from Pricing QA for expected feature grouping behavior.

**Definition of Done (DoD)**

* Radar Feature Set successfully populated in Data Vault.
* Grouping, schema flexibility, and CDC readiness validated.
* Source-to-DV reconciliation completed with no high-severity defects.
* Evidence attached and SIT sign-off obtained.

# 🟢 ****Jira Task – RL5.4 Populate PreComp Response Errors (SVRs) with Data Vault Model (Error)****

**Description**  
As part of SIT for Radar Live, BICOE QA will validate the **population of PreComp Response Error messages (SVRs)** into the Data Vault model.  
This involves validating the transformation of error JSON payloads into structured Data Vault entities, ensuring schema flexibility for new error types, CDC readiness, and preservation of error codes for downstream MI.

**Validation Scope**

* Validate transformation of PreComp Response Error JSON from Bronze → Data Vault.
* Confirm all error attributes are parsed correctly (error code, description, timestamp, impacted field).
* Validate schema flexibility to handle:
  + New error categories or error codes.
  + Nested/embedded error records.
* Validate CDC and versioning readiness for error records.
* Confirm no breaking schema changes in field names without versioning.
* Validate lineage from source PreComp error payloads into DV tables.

**Acceptance Criteria / Testing Requirements**

* ✅ PreComp error records parsed and stored in Data Vault tables.
* ✅ Error codes and descriptions retained accurately.
* ✅ Schema flexible enough for new error codes without redesign.
* ✅ Nested/embedded error arrays handled correctly.
* ✅ CDC/versioning validated for error records.
* ✅ No breaking schema changes observed without proper version control.
* ✅ Reconciliation with Bronze PreComp error JSON source completed.

**Dependencies**

* PreComp error schema documentation.
* Test data with various error scenarios (validation failures, system errors, multiple errors in one record).
* Time travel environment for CDC/versioning testing.
* Pricing QA input for business significance of error categories.

**Definition of Done (DoD)**

* PreComp error messages populated correctly in Data Vault.
* Schema flexibility, CDC/versioning validated with SIT evidence.
* Error lineage confirmed from Bronze to DV.
* No high-severity issues open at SIT exit.
* Test evidence shared, SIT sign-off obtained.

# 🟢 ****Jira Task – RL6.2 Populate Radar Feature Set with EDM (Success Only)****

**Description**  
As part of SIT for Radar Live, BICOE QA will validate the **population of Radar Feature Set data into EDM structures (Gold Layer)** to support downstream reporting and business KPIs.  
The validation will focus on dimensional modeling, dynamic schema handling, online/offline harmonization, feature categorization, and quote-level linking/versioning for consistent reporting across NB, MTA, Requotes, and Renewals.

**Validation Scope**

* **Dimensional Modeling**: Validate EDM structure of features supports downstream reporting needs (e.g., segmentation, decline metrics, feature effectiveness).
* **Dynamic Feature Growth**: Confirm schema flexibility supports dynamic, growing feature sets over time without breaking downstream jobs.
* **Feature Categorization**: Validate correct grouping of features by prefix/category:
  + **A** (Metadata), **C** (Existing Policy), **Cvr** (Cover), **Drv** (Driver), **Pol** (Policy), **Prp** (Proposer), **Veh** (Vehicle).
* **Business Reporting Integration**: Confirm EDM Core supports KPIs for:
  + Quote segmentation.
  + Decline metrics by feature breakdown.
  + Feature effectiveness insights.
* **Join Logic**: Validate join with quote-level metrics, rating results, and decline data for full reporting traceability.
* **Data Harmonization**: Validate online (PreComp/ML inputs) vs offline (EDM) feature values align for the same quotes.
* **Quote Linking & Versioning**: Validate Cache ID logic supports quote chain navigation (original → requote → MTA).
* **Dynamic Expansion & Governance**: Validate regression when new features/categories are introduced → no schema breaks, downstream jobs unaffected.

**Acceptance Criteria / Testing Requirements**

* ✅ EDM dimensional model populated correctly with Radar Feature Set.
* ✅ Feature categories (A, C, Cvr, Drv, Pol, Prp, Veh) validated against mapping rules.
* ✅ Schema flexibility validated with additional features introduced.
* ✅ Business reporting KPIs derived correctly (segmentation, declines, effectiveness).
* ✅ Quote-level joins validated across EDM (Quote ID, Cache ID, Interaction ID).
* ✅ Online vs offline feature harmonization validated (no divergence).
* ✅ Cache ID and version markers validated for multi-quote journeys.
* ✅ Regression testing confirms new features do not break existing pipelines.
* ✅ SIT test evidence attached in Jira.
* ✅ No high/critical issues remain open before SIT exit.

**Dependencies**

* EDM schema definition and mapping from Data Engineering.
* Radar Feature Set payloads (sample test data covering all categories).
* Input from Pricing QA on business KPIs and reporting metrics.
* E2E support for quote-level linking validation.
* DevOps/IT support for schema expansion testing.

**Definition of Done (DoD)**

* Radar Feature Set successfully populated into EDM (Gold Layer).
* Feature categorization, schema flexibility, and harmonization validated.
* Joins, KPIs, and reporting metrics validated against Confluence requirements.
* Regression testing for dynamic expansion passed with evidence.
* Source-to-EDM reconciliation complete.
* SIT sign-off note shared with stakeholders.

# 🟢 ****Jira Task – RL7.2 Populate Rating Request within Raw Rating Area****

**Description**  
BICOE QA will validate the **population of Rating Requests** into the new Raw Rating Area, ensuring correct parsing and storage.

**Validation Scope**

* Validate ingestion of Rating Request JSONs.
* Confirm schema mapping and field-level accuracy.
* Validate multiple quote journeys (NB, MTA, Requote, Renewal).
* Validate Cache ID and Interaction ID linkage across records.
* Perform reconciliation with MQS source messages.

**Acceptance Criteria**

* ✅ Rating Requests ingested successfully.
* ✅ Field mappings validated against design.
* ✅ Multiple journey types supported.
* ✅ Source vs Raw Rating Area reconciliation complete.

**Dependencies**

* MQS Rating Request sample payloads.
* Mapping specs.
* Coordination with E2E team.

**DoD**

* Population validated with SIT evidence.
* No high defects open.
* Sign-off obtained.

# 🟢 ****Jira Task – RL7.3 DVLA Data Retention Implementation****

**Description**  
BICOE QA will validate **DVLA Data Retention policies** in the Raw Rating Area, ensuring compliance with organisational and DVLA rules.

**Validation Scope**

* Validate retention lifecycle (storage, expiry, deletion).
* Validate DVLA compliance – only self-declared values retained.
* Validate time travel queries for historical records vs expired data.
* Ensure expired DVLA data not visible to Pricing users.
* Reconcile with Athena source data.

**Acceptance Criteria**

* ✅ DVLA retention implemented per policy.
* ✅ Time travel queries validated.
* ✅ Expired data inaccessible in SIT.
* ✅ Source vs Raw Rating Area reconciliation complete.
* ✅ No DVLA non-compliance.

**Dependencies**

* Retention rules from Data Governance.
* Source Athena DVLA data.
* Time travel environment setup.

**DoD**

* Retention validated with evidence.
* Compliance confirmed.
* No high/critical issues.
* Sign-off obtained.

# 🟢 ****Jira Task – RL7.4 Pricing Data Access to Raw Rating Area****

**Description**  
BICOE QA will validate **Pricing team access to Raw Rating Area** to ensure correct permission controls and data visibility.

**Validation Scope**

* Validate Pricing users can access allowed Rating Request data.
* Validate RBAC / PV group assignment.
* Negative testing → ensure restricted datasets (e.g., DVLA raw fields) not visible.
* Validate audit logging of access.

**Acceptance Criteria**

* ✅ Pricing access validated per requirements.
* ✅ RBAC roles confirmed.
* ✅ Negative tests passed (no over-permission).
* ✅ Audit logs captured.

**Dependencies**

* PV group setup by DevOps/IT.
* Governance-approved access rules.
* Pricing team participation in SIT.

**DoD**

* Pricing access validated with SIT evidence.
* No high-severity security issues.
* Sign-off obtained.

# 🟢 ****RL10A.1 – Model Rating Response Structure (Silver – PreComp Features & Motor Rating Factor)****

**Description**  
Validate Rating Response Silver structure supports PreComp features and motor\_rating\_factor\_view.

**Validation Scope**

* Validate Silver tables parse Rating Response payloads correctly.
* Confirm joins with Core Risk & Feature Set.
* Validate aggregation for rating factors per quote.

**Acceptance Criteria**

* ✅ Rating Response stored as per schema.
* ✅ PreComp features represented.
* ✅ Motor rating factor values validated.

**Dependencies**

* MQS Rating Response payloads.
* Mapping from Data Engineering.

**DoD**

* Silver structure validated with SIT evidence.

# 🟢 ****RL10A.2 – Populate Rating Response Silver (PreComp Features & Motor Rating Factor View)****

**Description**  
Validate population of Silver Rating Response structures.

**Validation Scope**

* Validate data load from Bronze → Silver.
* Confirm field-level mapping.
* Validate quote journeys (NB, Requote, MTA, Renewal).

**Acceptance Criteria**

* ✅ Silver populated correctly.
* ✅ All journeys supported.
* ✅ Source vs Silver reconciliation complete.

**DoD**

* SIT evidence attached.

# 🟢 ****RL10B.3 – Populate Dynamic UW Radar Results****

**Description**  
Validate Silver Radar Results for Underwriting.

**Scope**

* Confirm Radar UW outputs stored correctly.
* Validate schema for new Radar features.
* Confirm linkage to Quote ID / Cache ID.

**Acceptance Criteria**

* ✅ UW Radar results populated.
* ✅ Quote linkage validated.

**Dependencies**

* Radar outputs from Pricing/UW.

# 🟢 ****RL10B.4 – Populate Dynamic Pricing Radar Results****

**Description**  
Validate Silver Radar Results for Pricing.

**Scope**

* Validate Pricing Radar outputs stored.
* Confirm multiple insured object handling.

**Acceptance Criteria**

* ✅ Pricing Radar results reconciled with source.

# 🟢 ****RL10B.5 – Provide Access to Dynamic Radar Results Structures****

**Description**  
Validate access controls for Radar Results.

**Scope**

* Validate RBAC for Pricing/UW.
* Negative testing → restricted groups can’t access.

**Acceptance Criteria**

* ✅ Access rules validated.
* ✅ Audit logs confirm access attempts.

# 🟢 ****RL11A.2 – Populate Rating Response Gold (Core Result Fields)****

**Description**  
Validate population of Rating Response into EDM (Gold).

**Scope**

* Validate dimensional models.
* Confirm KPIs and reporting metrics.
* Validate joins with Core Risk & Cache ID.

**Acceptance Criteria**

* ✅ Gold layer populated correctly.
* ✅ Reporting validated.

# 🟢 ****RL12.2 – Populate Cache ID within Silver Model****

**Description**  
Validate Cache ID storage in Silver.

**Scope**

* Validate Cache ID links quotes across journeys.
* Confirm unique constraint handling.

**Acceptance Criteria**

* ✅ Cache ID linkage validated (NB → Requote → MTA).

# 🟢 ****RL17.1 – Enable Continuity of Service – Pricing Views****

**Description**  
Regression testing of Pricing views with MQS data.

**Scope**

* Validate MQS integration does not break existing Pricing views.
* Validate schema consistency.

**Acceptance Criteria**

* ✅ Existing Pricing reports remain consistent.

# 🟢 ****RL18.2 – Enable Continuity of Service – Reporting Views****

**Description**  
Regression testing of Reporting views.

**Scope**

* Validate MQS integration does not break MI dashboards.

**Acceptance Criteria**

* ✅ Existing dashboards validated.

# 🟢 ****RL19 – Vehicle Model / Area Factor Data Ingestion****

**Description**  
Validate ingestion of Vehicle Model & Area Factors.

**Scope**

* Validate file ingestion pipelines.
* Confirm schema mapping.
* Validate reconciliations with source files.

**Acceptance Criteria**

* ✅ Vehicle/Area factors ingested.
* ✅ Data reconciled.

# 🟢 ****RL21 – Quote Cache Deletion Support****

**Description**  
Validate deletion of Quote Cache IDs.

**Scope**

* Confirm Cache IDs deleted per retention rules.
* Validate blob storage cleanup.

**Acceptance Criteria**

* ✅ Cache IDs deleted as expected.

# 🟢 ****RL22 – Non-Functional Requirements (NFRs)****

**Description**  
Validate service excellence aspects for Radar Live (NFR).

**Scope**

* Reconciliation between Athena & MQS.
* Monitoring, error management, alerting.
* Ensure no data loss.

**Acceptance Criteria**

* ✅ NFR scenarios tested.
* ✅ Monitoring/alerts validated.

# 🟢 ****RL23 – Athena Interaction Data****

**Description**  
Validate Athena Interaction data correlation with MQS.

**Scope**

* Confirm Interaction ID mapping.
* Validate reconciliations (Athena → MQS → DV).

**Acceptance Criteria**

* ✅ Interaction data reconciled.
* ✅ Lineage validated.

✅ Each of these can be uploaded as **separate Jira stories** under their parent Epic (e.g., Rating Response Silver, Rating Response Gold, Cache ID, Continuity of Service).