**🔹 Additional MTA / Policy Scenarios to Consider**

| **Scenario** | **Case Description** |
| --- | --- |
| MTA – Policy on Hold | Temporarily suspend policy, resume later – validate data flow and rating impact. |
| MTA – Premium Adjustment | Manual premium override (e.g., goodwill discount) – ensure new premium is reflected. |
| MTA – Cross-product change | Change from one policy type to another (e.g., car to van) – ensure message handling. |
| MTA – Mid-term Quote Abandonment | Quote started but not completed – ensure partial interaction is logged. |

**🔹 Renewal Journey Variants**

| **Scenario** | **Case Description** |
| --- | --- |
| Renewal – Lapse | Policy not renewed after invite – validate quote expiry and data retention. |
| Renewal – Requote at Renewal | Customer re-quotes during renewal invite period – ensure updated pricing handled. |
| Renewal – Change of Payment Frequency | Switch from monthly to annual or vice versa – validate correct MQS data flow. |

**🔹 New Business (NB) Variants**

| **Scenario** | **Case Description** |
| --- | --- |
| NB – Abandoned Quote | Quote not completed – partial data stored, Cache ID issued, but no Rating Response. |
| NB – Manual Underwriting Required | Quote triggers manual review – validate MQS handles “Pending” status and data state. |

**🔹 Decline Scenarios Expansion**

| **Scenario** | **Case Description** |
| --- | --- |
| Decline – Fraud Rule Triggered | Quote declined by fraud engine – ensure Stop Quote and reason are mapped. |
| Decline – System Error | Rating failed due to system error – validate system decline category handled. |
| Decline – Eligibility Rule | E.g., driver age, conviction – verify decline code mapped correctly in EDM. |

**🔹 Claims Impact Scenarios Expansion**

| **Scenario** | **Case Description** |
| --- | --- |
| Post-MTA Claim | Claim after MTA – ensure new premium, coverage reflected in updated quote chain. |
| DVLA Match Failure | DVLA data mismatch – verify Stop Quote triggered and reason stored. |

**🔹 Aggregator-Specific Scenarios**

| **Scenario** | **Case Description** |
| --- | --- |
| Aggregator – Multiple Quotes | Customer submits multiple quotes – ensure Cache ID reuse and traceability. |
| Aggregator – Redirect to Org Site | Quote started on aggregator, completed on org website – linkage validation. |

**🔹 General Edge Cases**

| **Scenario** | **Case Description** |
| --- | --- |
| Duplicate Quote Submission | Same user submits quote twice in short time – ensure message deduplication. |
| Policy Cancellation – Non-payment | Auto-cancellation process – validate data flow and status update. |
| Cross-channel Requote | Quote started on aggregator, requoted on org site – validate consistency. |

Let me know if you want this delivered in:

* A **tabular Excel format**
* Jira **test scenario stories**
* A **Confluence-ready checklist** for traceability

I can quickly prepare it.

**You said:**

more of quote based scenario , like switch Tier, multiple quote version and time travel scenario etc

**ChatGPT said:**

Great — you're now moving into **complex quote behavior scenarios** that test real-world variations in the **quote lifecycle**, **tiering**, and **versioning logic**. Here's an expanded list of **additional quote-based test scenarios** that are **often overlooked** but critical for full validation of MQS and data platform integrations.

**🔹 Advanced Quote-Based Test Scenarios**

**1. Tier Switching Scenario**

| **Scenario ID** | **Description** |
| --- | --- |
| QBS\_01 | User gets multiple price tiers (e.g., Tier 1 / Tier 2), switches between tiers – ensure correct tier selected in Rating Response. |
| QBS\_02 | Tier initially shown, user requotes and different tier offered – validate tier history across quote versions. |
| QBS\_03 | Declined on lower tier, accepted on higher tier – ensure both quote attempts stored, correct decline flag applied. |

**2. Quote Versioning / Requote Chains**

| **Scenario ID** | **Description** |
| --- | --- |
| QBS\_04 | User submits quote → requotes → again requotes → validate all Quote IDs linked by Cache ID. |
| QBS\_05 | Version 1 accepted, later updated via MTA → validate original quote and MTA quote stored correctly. |
| QBS\_06 | Requote triggers new Risk Assessment (e.g., different vehicle) – ensure Core Risk versioning logged. |

**3. Quote Time Travel / Effective Date Testing**

| **Scenario ID** | **Description** |
| --- | --- |
| QBS\_07 | Quote created today, effective next month – validate correct timestamping and policy start date. |
| QBS\_08 | Quote updated multiple times over days – ensure all timestamped events are visible in EDM. |
| QBS\_09 | Policy backdated via MTA → validate historical data integrity and correct version reflected. |

**4. Multi-Policy / Multi-Product Quotes**

| **Scenario ID** | **Description** |
| --- | --- |
| QBS\_10 | User quotes for car + home insurance – validate separate quote journeys handled with unique identifiers. |
| QBS\_11 | One policy accepted, one declined – ensure correct separation and status in data warehouse. |

**5. Cross-Channel Quote Journey**

| **Scenario ID** | **Description** |
| --- | --- |
| QBS\_12 | Start quote on Aggregator → redirect to Org site → user accepts – validate linkage of quote records. |
| QBS\_13 | Quote started on Org site, requoted on Aggregator – confirm quote chain tracked across channels. |

**6. Quote Abandonment & Retry**

| **Scenario ID** | **Description** |
| --- | --- |
| QBS\_14 | User starts quote but abandons → returns next day to complete – ensure partial data and retry linked. |
| QBS\_15 | Retry generates new Quote ID, reuse Cache ID – validate reuse logic, no data loss. |

**7. Stop Quote & Re-Engagement**

| **Scenario ID** | **Description** |
| --- | --- |
| QBS\_16 | Initial quote declined at Precomp (Stop Quote) → user edits input → successful quote issued. |
| QBS\_17 | Both declined and accepted quote stored – ensure Stop Quote flag captured correctly. |

**8. Quote Data Refresh (MI Reporting Impact)**

| **Scenario ID** | **Description** |
| --- | --- |
| QBS\_18 | Quote data updated overnight – ensure Gold layer (EDM) refresh job reflects latest version. |
| QBS\_19 | KPI reports show correct quote counts, versions, declines – regression test required. |

**✅ Special Focus: Data Warehouse Perspective**

For each of the above scenarios, **validate**:

* Correct **Cache ID / Quote ID / PRN linkage**
* Accurate timestamps for each quote version
* Correct **Tier Level** stored
* Field-level **data consistency** across quote versions
* **Stop Quote** and **Decline Code** representation
* Visibility of **full quote chain** in EDM for MI

Would you like this list as:

* 📊 Excel Test Scenario Matrix?
* 🧾 Jira Story/Sub-task list?
* 📋 Confluence-ready table?

Let me know and I’ll generate it for you.

**You said:**

include something in the view of relation between tier & premium, essestial tier has high premium and silver tier has low premium , etc .... quote interaction in terms timeline , quote expire etc

1/2

**ChatGPT said:**

Excellent – you’re now looking for **realistic quote scenarios** tied to **tier–premium relations**, **timeline of quote interactions**, and **quote expiry behavior**, which are critical for accurate validation in SIT and downstream MI reporting.

Here's an expanded **Quote Behavior Scenario Set** focused on:

**🔹 Tier vs. Premium Relationship Scenarios**

| **Scenario ID** | **Description** |
| --- | --- |
| QBS\_20 | **Essential Tier = Higher Premium** – Validate that quote with Essential Tier shows higher premium vs Silver Tier for same input. |
| QBS\_21 | **Silver Tier = Lower Premium** – User initially shown Essential, requotes to get Silver – validate premium drop is reflected in Rating Response. |
| QBS\_22 | **Tier Switch → Premium Change → Final Quote Saved** – User compares tiers, selects one → validate only final tier/premium is stored as accepted. |
| QBS\_23 | **Tier Rejected by Pricing Rules** – Quote with cheaper tier declined (e.g., not eligible) → user accepts costlier tier – validate both outcomes logged. |

✅ **Test Tip:** In EDM, verify Tier\_Level, Premium\_Amount, Rating\_Response\_ID all linked correctly.

**🔹 Quote Interaction Timeline & Expiry Scenarios**

| **Scenario ID** | **Description** |
| --- | --- |
| QBS\_24 | Quote created at 10 AM → Requoted at 4 PM same day → validate both quotes timestamped, latest marked “current”. |
| QBS\_25 | Quote created → not accepted within 24 hours → quote expires – ensure status flagged “expired” in data. |
| QBS\_26 | Requote submitted after quote expiry – validate new Quote ID, same Cache ID, new timestamp. |
| QBS\_27 | Quote viewed multiple times by user – validate **interaction count** and **last viewed timestamp**. |

✅ **Test Tip:** Verify EDM captures Created\_Timestamp, Last\_Updated, Expiry\_Flag, Interaction\_Count.

**🔹 Combined Scenarios: Tier + Time + Outcome**

| **Scenario ID** | **Description** |
| --- | --- |
| QBS\_28 | User quotes Essential Tier today, declines → requotes Silver Tier next day, accepts – validate time gap, premium difference, and quote chain. |
| QBS\_29 | Multiple quotes over 7 days – ensure MI reports capture accurate **quote lifecycle** per customer (create → update → expire → requote → accept). |

**✅ Data Warehouse Focus Points for Each Scenario**

* **Fields to validate:**
  + Tier\_Level, Premium\_Value, Quote\_Status, Quote\_Created\_Date, Quote\_Expiry\_Date
* **Linkage via**:
  + Cache\_ID, Quote\_ID, PRN
* **Historical Tracking**:
  + Store both **accepted and expired quotes** in EDM for auditability.
* **Premium Consistency**:
  + Premium matches tier logic as per pricing config.

**🟢 Summary Additions for Your Testing Plan:**

* Simulate **Essential vs Silver tier premiums** → validate in EDM
* Validate **quote expiry handling** and correct flagging downstream
* Ensure **MI reports** can differentiate **latest vs. historical quotes**
* Validate **interaction timestamps** for clickstream and quote journey analysis

Let me know if you want this delivered as:

* A **Confluence page format**
* A **testing checklist or scenario tracker**
* A **diagram of Tier ↔ Premium ↔ Outcome flow** for presentation use.

**You said:**

MTA cancel with 14 days of purchase , auto renewal opt in / out, cancel the policy before it starts, include something from Radar engine testing prespective as well

**ChatGPT said:**

Perfect – you’re now diving into **policy lifecycle edge cases**, **regulatory conditions (cooling-off, auto-renewal)**, and **Radar engine testing triggers**, which are often missed but critical for **compliance, pricing accuracy**, and **customer experience validation**.

Here’s a **refined set of scenarios** to cover these cases in **SIT** from a **data warehouse + Radar engine** validation perspective:

**🔹 Policy Cancellation & Cooling-Off Period Scenarios**

| **Scenario ID** | **Description** |
| --- | --- |
| QBS\_30 | **MTA Cancellation within 14 Days (Cooling-off)** – Customer cancels MTA within 14 days → validate cancellation recorded, premium adjustment reversed, Stop Quote if needed. |
| QBS\_31 | **Policy Cancelled Before Start Date** – Customer cancels after quote acceptance but before policy starts → validate policy status = cancelled, no rating response needed. |
| QBS\_32 | **Cancellation by Org (e.g., fraud flag)** – Organisation cancels → validate forced cancellation flag & reason code stored. |
| QBS\_33 | **Auto-renewal Opt-In → Cancelled before renewal date** – Validate flag for opted-in customers cancelling before renewal, ensure renewal quote voided. |
| QBS\_34 | **Auto-renewal Opt-Out → No quote generated** – Ensure no renewal quote journey exists, confirm in MI data gaps for opted-out. |

✅ Validate in EDM:

* Policy\_Status, Cancellation\_Reason\_Code, Cooling\_Off\_Flag, Auto\_Renewal\_OptIn\_Flag, Start\_Date, Cancellation\_Date

**🔹 Radar Engine Testing Scenarios (Pricing Rules + Declines)**

| **Scenario ID** | **Description** |
| --- | --- |
| QBS\_35 | **Radar triggers rating decline (U/W rule)** – Quote declined due to Radar rule (e.g., driver age) – validate Rating Response = declined, proper reason stored. |
| QBS\_36 | **Radar applies discount / adjustment** – Pricing adjustment applied (e.g., loyalty discount) → validate discount reflected in premium, rating response. |
| QBS\_37 | **Radar fails → System Error** – Radar engine fails to rate → Stop Quote with system decline code – ensure error category mapped. |
| QBS\_38 | **Radar Tier Adjustment** – Radar switches tier based on real-time rules (e.g., low-risk customer) – validate tier shift and premium match. |
| QBS\_39 | **Radar vs MI Premium Discrepancy Check** – Simulate premium from Radar vs what is stored in EDM → validate they match for audit. |

✅ Validate:

* Radar\_Rule\_Applied, Rating\_Outcome, Discount\_Code, Premium\_Amount, Tier\_Level, Rating\_Decline\_Code

**✅ General Validation Points for These Scenarios**

| **Area** | **What to Validate** |
| --- | --- |
| **Timestamps** | Accurate Quote\_Created, Cancellation\_Date, Policy\_Start |
| **Flags** | Auto\_Renewal\_Flag, Cooling\_Off\_Flag, Radar\_Rule\_Applied, Decline\_Reason\_Code |
| **Premium Handling** | Premium refund on cancellation, no duplication post-renewal |
| **Radar Testing** | Engine-triggered outcomes, premium/tier matching MI |

**✅ Business Impact & MI Reporting Coverage**

* Compliance reporting for **cooling-off cancellations**
* **Auto-renewal opt-out rates** in MI reports
* MI accuracy for **Radar-triggered declines / discounts**
* Policy lifecycle **traceability** in EDM

Would you like these:

* As **Jira test cases**?
* In **Confluence checklist**?
* With a **diagram of policy lifecycle events**? Let me know!