**EOD Limit Check Test Scenarios - Code Modifications**

**Overview**

You need to modify the determineExposure() method in rateseodlimits.py around **line 83** (after the dataSourceFactory call) to simulate missing data scenarios.

**VTD List (Based on your YAML files):**

1. AMRS LINEAR RATES
2. APAC LINEAR RATES
3. CFD AND STRUCTURED NOTES
4. EMEA LINEAR RATES
5. GLOBAL NON-LINEAR-AMRS STRUCTURED RATES
6. GLOBAL NON-LINEAR-APAC STRUCTURED RATES
7. GLOBAL NON-LINEAR-EMEA STRUCTURED RATES
8. GLOBAL RATES (from uat\_global\_rates\_eod.yaml)

**SCENARIO 1: All VTDs have data (Baseline)**

**Purpose**: Verify normal operation when all data is present.

# No modifications needed - this is your baseline test

# Just run the script normally to verify all VTDs generate reports

# and attachments are created for all VTDs

**Sub-scenario 1a: GNLR AMRS with multiple sources**

**Purpose**: Verify GNLR AMRS VTD handles both legacy and cirt\_unified\_screen sources correctly.

# No code changes needed - this VTD naturally has multiple sources

# Verify in logs that both sources contribute data:

# - legacy source: IR01, Vega, IR Vega

# - cirt\_unified\_screen source: IR01, Vega, IR Vega

**SCENARIO 2: One VTD - All measures missing data**

**Scenario 2a: First VTD (AMRS LINEAR RATES) missing all data**

Add this code after line 83 in determineExposure():

if name == 'AMRS LINEAR RATES':

# Clear all snapshots and exposure data

snapshotsForSource = {}

expTable = qztable.Table()

fieldsDict = {

'source': 'cirt\_unified\_screen',

'measure\_names': ['IR01', 'Vega', 'Sov Spread Delta'],

'calc\_level': ['VTD+Currency'],

'measuresMissingExposures': ['IR01', 'Vega', 'Sov Spread Delta'],

'level': 'AMRS LINEAR RATES'

}

**Scenario 2b: Last VTD (GLOBAL RATES) missing all data**

if name == 'GLOBAL RATES':

snapshotsForSource = {}

expTable = qztable.Table()

fieldsDict = {

'source': 'management\_rra',

'measure\_names': ['IR Delta', 'IR Vega', 'Inflation Delta'],

'calc\_level': ['LE', 'Currency'],

'measuresMissingExposures': ['IR Delta', 'IR Vega', 'Inflation Delta'],

'level': 'GLOBAL RATES'

}

**Scenario 2c: Second VTD (APAC LINEAR RATES) missing all data**

if name == 'APAC LINEAR RATES':

snapshotsForSource = {}

expTable = qztable.Table()

fieldsDict = {

'source': 'cirt\_unified\_screen',

'measure\_names': ['IR01', 'Sov Spread Delta'],

'calc\_level': ['VTD+Currency'],

'measuresMissingExposures': ['IR01', 'Sov Spread Delta'],

'level': 'APAC LINEAR RATES'

}

**SCENARIO 3: One VTD - One measure missing, one measure present**

**Scenario 3a: First VTD (AMRS LINEAR RATES) missing one measure**

if name == 'AMRS LINEAR RATES':

# Keep some data but remove Vega measure

expTable = expTable[expTable['Measure'] != 'Vega']

# Remove Vega from snapshots

for key in snapshotsForSource.keys():

if 'Vega' in snapshotsForSource[key].columnNames():

snapshotsForSource[key] = snapshotsForSource[key][snapshotsForSource[key]['Measure'] != 'Vega']

fieldsDict.update({

'measuresMissingExposures': ['Vega']

})

**Scenario 3b: Last VTD (GLOBAL RATES) missing one measure**

if name == 'GLOBAL RATES':

# Remove IR Vega measure

expTable = expTable[expTable['Measure'] != 'Vega']

for key in snapshotsForSource.keys():

if 'Measure' in snapshotsForSource[key].columnNames():

snapshotsForSource[key] = snapshotsForSource[key][snapshotsForSource[key]['Measure'] != 'Vega']

fieldsDict.update({

'measuresMissingExposures': ['Vega']

})

**Scenario 3c: Second VTD (APAC LINEAR RATES) missing one measure**

if name == 'APAC LINEAR RATES':

# Remove IR01 measure

expTable = expTable[expTable['Measure'] != 'IR01']

for key in snapshotsForSource.keys():

if 'Measure' in snapshotsForSource[key].columnNames():

snapshotsForSource[key] = snapshotsForSource[key][snapshotsForSource[key]['Measure'] != 'IR01']

fieldsDict.update({

'measuresMissingExposures': ['IR01']

})

**SCENARIO 4: Multiple VTDs with missing data**

**Purpose**: Test combination of completely missing VTD and partially missing VTD.

# Combine two scenarios - one VTD completely missing, one partially missing

if name == 'CFD AND STRUCTURED NOTES':

# CFD completely missing all measures

snapshotsForSource = {}

expTable = qztable.Table()

fieldsDict = {

'source': 'management\_rra',

'measure\_names': ['Vega', 'IR01'],

'calc\_level': ['VTD+Currency'],

'measuresMissingExposures': ['Vega', 'IR01'],

'level': 'CFD AND STRUCTURED NOTES'

}

if name == 'EMEA LINEAR RATES':

# EMEA missing only Vega measure

expTable = expTable[expTable['Measure'] != 'Vega']

for key in snapshotsForSource.keys():

if 'Measure' in snapshotsForSource[key].columnNames():

snapshotsForSource[key] = snapshotsForSource[key][snapshotsForSource[key]['Measure'] != 'Vega']

fieldsDict.update({

'measuresMissingExposures': ['Vega']

})

**Expected Results for Each Scenario**

**Scenario 1 (Baseline)**

* ✅ EOD limit check report generated with all 8 VTDs
* ✅ All exposure detail attachments present
* ✅ No missing exposure alerts

**Scenarios 2a, 2b, 2c (Complete VTD missing)**

* ✅ EOD limit check report with 7 VTDs (missing VTD excluded)
* ✅ Missing exposure alert for the affected VTD with all measures listed
* ✅ No attachment for the missing VTD
* ✅ Attachments present for remaining 7 VTDs

**Scenarios 3a, 3b, 3c (Partial VTD missing)**

* ✅ EOD limit check report with all 8 VTDs
* ✅ Affected VTD shows data only for available measures
* ✅ Missing exposure alert for the affected VTD with missing measure listed
* ✅ Attachment for affected VTD contains only available measures
* ✅ Full attachments for other 7 VTDs

**Scenario 4 (Multiple VTDs affected)**

* ✅ EOD limit check report with 7 VTDs (completely missing VTD excluded)
* ✅ Partially affected VTD shows available measures only
* ✅ Two separate missing exposure alerts (one for each affected VTD)
* ✅ No attachment for completely missing VTD
* ✅ Partial attachment for partially missing VTD
* ✅ Full attachments for remaining 6 unaffected VTDs

**Additional Scenarios to Consider**

**Scenario 5: Data source connectivity issues**

Test when one data source is completely unavailable (simulate network/database issues):

if name == 'GLOBAL NON-LINEAR-AMRS STRUCTURED RATES':

# This VTD has both legacy and cirt sources - simulate legacy source failure

if fieldsDict.get('source') == 'legacy':

snapshotsForSource = {}

expTable = qztable.Table()

fieldsDict.update({

'measuresMissingExposures': ['CFTC-IRDelta', 'CFTC-IRVega', 'CFTC-IRVegaM1']

})

**Scenario 6: Timing edge cases**

Test EOD processing during different snap times to ensure consistent behavior.

**Testing Checklist**

For each scenario, verify:

1. **Report Generation**:
   * [ ] EOD limit check report contains expected VTDs
   * [ ] Report excludes VTDs with no data
   * [ ] Report shows partial data for VTDs with some measures
2. **Alert Generation**:
   * [ ] Missing exposure alerts generated for affected VTDs
   * [ ] Alert emails contain correct measure names
   * [ ] Alert subject follows pattern: [CFTC Error][Action Required] MissingExposures:
3. **Attachments**:
   * [ ] No attachments for completely missing VTDs
   * [ ] Partial attachments for VTDs with some missing measures
   * [ ] Full attachments for unaffected VTDs
4. **Data Integrity**:
   * [ ] No duplicate data in reports
   * [ ] GNLR AMRS properly handles multiple data sources
   * [ ] Utilization calculations correct for available data

**Implementation Notes**

* Test one scenario at a time by commenting/uncommenting the respective code blocks
* Always test against UAT environment first
* Verify email recipients receive appropriate alerts
* Check Sandra database writes for exposure data consistency
* Monitor logs for any unexpected errors or warnings

Based on your requirements and code analysis, I've created a comprehensive testing guide. Here are the key points:

**Where to Add Code**

Add the test code in the determineExposure() method in rateseodlimits.py, specifically **after line 83** where you have:

python

snapshotsForSource, expTable, fieldsDict = dataSourceFactory(self.cfg, sourceKey, dataSources, self.jobTimestamp)

**How to Use**

1. **Test one scenario at a time** - uncomment only the code block for the scenario you want to test
2. **Run the Bob job** from your UAT environment
3. **Verify the results** match the expected outcomes listed in each scenario

**Key Verification Points**

For each test scenario, check:

* **EOD Report**: Contains expected VTDs and excludes missing ones
* **Alert Emails**: Generated for missing data with correct subject line
* **Attachments**: Present/absent as expected based on data availability
* **Log Messages**: Confirm the notifyEODEmptyMeasureExposures function is called

**Additional Scenarios to Consider**

* **Data source connectivity issues** (when one of multiple sources fails)
* **Timing edge cases** for different snap times