

# MRF Quality Report

s3://talon-storage-private/mrf-feed-uploads/2026-01/2026-01-01\_mc\_2\_in-network-rates.json

Size: 0 B • MD5: d41d8cd98f00b204e9800998ecf8427e

Payer: **Unknown** • File Date: **N/A** • Generated: **2026-04-24 14:12 EDT** • Tool Version: **1.0.0** • Elapsed: **20.70s**

# 51.0

**Not Usable**

score capped at 59.0 — 1 native JSON parse error(s) require payer re-export; CMS official schema validation failed (see meta for details)

Errors: 5 • Warnings: 0 • Info: 0 • **1 Raw JSON Error(s)**

## TOC Plan References

TOC: s3://talon-storage-private/mrf-feed-uploads/2026-01/2026-01\_plan\_ref\_e7cd5f5c82c1c63f41b07fb716400106\_index.json • Midlands Choice (Third Party)

Canonical: s3://talon-storage-private/mrf-feed-uploads/2026-01/2026-01-01\_mc\_2\_in-network-rates.json

Plan Name	Plan ID	Issuer / Sponsor	Market
MidlandsChoice2	2 (EIN)	—	group

## Raw JSON Parse Errors

### IncompleteJSONError

parse error: premature EOF

Context:

```
(right here) -----^
```

Full message:

```
parse error: premature EOF
```

```
(right here) -----^
```

## CMS Official Schema Validation

**FAILED** (exit code 1) — File does not conform to the CMS schema.

Validator output:

```
error Data file does not contain version information. Please run again using the --schema-version option to specify a version.
```

## Dimension Scores

Dimension	Score	Weight	Findings
Schema Integrity	70.0	30%	4
Provider Mapping	100.0	15%	0
Code Coverage	100.0	15%	0
Pricing Sanity	0.0	40%	1

## Schema Integrity — Findings

---

Score: 70.0

**ERROR** `required_header_fields`

Missing required header fields: ['reporting\_entity\_type', 'last\_updated\_on', 'reporting\_entity\_name', 'version']

---

**ERROR** `cms_schema_validation`

CMS official schema validator FAILED (exit code 1). File does not conform to the TIC in-network-rates schema.

---

**ERROR** `raw_json_error`

Native JSON parse error detected in unpatched source file: parse error: premature EOF. Scoring below reflects auto-patched data only. File must be re-exported by the payer to fully resolve.

(right here) -----^

---

**ERROR** `parse_crash`

single pass crash after 0 items: parse error: premature EOF (right here) -----^

---

## Pricing Sanity — Findings

Score: 0.0

**ERROR** no\_rates

No negotiated rates found in file

## Recommended Actions

1. **overall**

**P0**

Score below minimum threshold — do not use for pricing without remediation.

2. **schema** required\_header\_fields

**P1**

Missing required header fields: ['reporting\_entity\_type', 'last\_updated\_on', 'reporting\_entity\_name', 'version']

3. **schema** cms\_schema\_validation

**P1**

CMS official schema validator FAILED (exit code 1). File does not conform to the TIC in-network-rates schema.

4. **schema** raw\_json\_error

**P1**

Native JSON parse error detected in unpatched source file: parse error: premature EOF. Scoring below reflects auto-patched data only. File must be re-exported by the payer to fully resolve.

5. **schema** parse\_crash

**P1**

single pass crash after 0 items: parse error: premature EOF (right here) -----^

6. **pricing** no\_rates

**P1**

No negotiated rates found in file

## Provider Geographic Coverage

0 unique NPIs found — 0 geocoded (0%) — 0 zip codes represented.

## Schema Integrity — Metrics

---

header_missing_fields	reporting_entity_type, last_updated_on, reporting_entity_name, version
header_conditional_issues	
file_age_days	None
items_total	0
items_missing_required_pct	0
items_empty_rates	0
prices_total	0
prices_missing_required_pct	0
prices_missing_field_breakdown	
prices_missing_service_code	0
prices_invalid_billing_class	0
rates_without_providers	0
negotiation_arrangements	
billing_code_types	
expired_prices	0
invalid_expiration_format	0

## Provider Mapping — Metrics

---

provider_references_in_file	0
provider_group_ids_referenced	0
unresolved_references	0
resolution_rate_pct	100.0
npis_validated	0
invalid_npi_count	0
npi_validity_rate_pct	None
invalid_npi_examples	
eins_validated	0
invalid_ein_count	0

ein_validity_rate_pct	None
invalid_ein_examples	
empty_npi_groups	0
groups_without_tin	0
npi_in_multiple_groups	0

## Code Coverage — Metrics

unique_codes_total	0			
duplicate_codes	0			
duplicate_pct	0			
by_code_type				
unknown_code_types				
format_invalid_by_type				
codes_not_in_reference	reference_not_loaded			
most_frequent_codes	<table border="1"> <thead> <tr> <th>Type</th> <th>Code</th> <th>Occurrences</th> </tr> </thead> <tbody> </tbody> </table>	Type	Code	Occurrences
Type	Code	Occurrences		

## Pricing Sanity — Metrics

total_prices_checked	0
total_rates	0
per_diem_rates	0
percentage_rates	0
negative_rates	0
zero_rates	0
extreme_high_rates	0
extreme_low_rates	0

<b>rate_distribution</b>	<b>sample_n</b>	0
	<b>sample_k</b>	0
	<b>confidence</b>	low
	<b>p5</b>	None
	<b>p25</b>	None
	<b>p50</b>	None
	<b>p75</b>	None
	<b>p95</b>	None
	<b>p99</b>	None

<b>by_billing_class</b>	<b>Class / Type</b>	<b>Count</b>	<b>Median</b>	<b>p25</b>	<b>p75</b>	<b>p95</b>	<b>Confidence</b>
-------------------------	---------------------	--------------	---------------	------------	------------	------------	-------------------

**negotiated\_types**

**unique\_rate\_contexts** 0

<b>rate_key_dimension_validity</b>	<b>invalid_negotiated_type</b>	0
	<b>invalid_negotiated_types_seen</b>	{}
	<b>invalid_setting</b>	0
	<b>invalid_settings_seen</b>	{}
	<b>invalid_severity_of_illness</b>	0
	<b>severity_on_non_apr_drg</b>	0
	<b>institutional_with_service_codes</b>	0
	<b>invalid_service_code_format</b>	0
	<b>billing_code_modifier_too_long</b>	0

# Scoring Methodology

Embedded in this report at generation time.

## Overall Score

Weighted sum of four structural dimensions, normalized to a 0–100 scale.

Normalized Weights		
	Schema Integrity	30%
	Provider Mapping	15%
	Code Coverage	15%
	Pricing Sanity	40%

  

Confidence Bands		
	High	≥90
	Usable With Caution	≥75
	Limited Reliability	≥60
	Not Usable	<60

  

Score Caps		
	Raw Json Errors Only → 74.0	Native JSON syntax errors in the unpatched source file. File must be re-exported by the payer; scoring reflects auto-patched data only.
	Cms Validation Failure Only → 65.0	CMS official schema validator reports the file does not conform to the TIC spec.
	Both Raw Json Errors And Cms Failure → 59.0	Both native JSON syntax errors and CMS schema validation failure present.

## Rate Context Key — 14-Tuple Field Coverage

Every rate in a CMS TIC MRF file is described by a 14-field tuple. Fields 1–10 form the rate-context key used to group and compare rates across the system. Fields 11–12 (provider, expiration date) are validated separately and excluded from the grouping key for analytical reasons. Each of the four scoring dimensions validates a distinct slice of this tuple — together they cover all 14 fields.

Field	Validated by
1 billing_code_type	Schema (required field) + Code Coverage (enum + format validation)
2 billing_code_type_version	Schema (required field)
3 billing_code	Schema (required field) + Code Coverage (format, duplicates, reference lookup)
4 billing_code_modifier	Pricing (modifier length, key normalization)
5 service_code	Pricing (POS format, normalization, institutional-class check)
6 negotiated_type	Pricing (CMS TIC enum validation)
7 billing_class	Schema (CMS TIC enum validation) + Pricing (spread thresholds)
8 negotiation_arrangement	Schema (CMS TIC enum validation) + Pricing (FFS vs bundle/capitation gating)
9 severity_of_illness	Pricing (APR-DRG only, valid values 1–4)

Field	Validated by
10 setting	Pricing (CMS TIC enum validation)
11 provider (NPI/EIN)	Provider Mapping (Luhn checksum, IRS prefix, group resolution) — excluded from grouping key
12 expiration_date	Schema (date validity, far-future sanity) — excluded from grouping key
13 additional_generic_notes	not validated (free-text)
14 negotiated_rate	Pricing (negative/zero/extreme-value checks, spread analysis)

- Fields 1–10 are the grouping key. Each unique combination is a distinct rate context — rates with different modifiers, POS codes, or arrangements land in separate buckets and are never compared against each other.
- Provider (field 11) is excluded from the key: the spread check is cross-provider by design. Partitioning by provider produces singleton buckets and eliminates the spread signal.
- Expiration date (field 12) is excluded because it is a contract lifecycle attribute, not a clinical context. Rates for the same service should be comparable regardless of when they expire.
- service\_code (field 5) arrays are flattened and normalized before keying: '1' → '01', and a rate with ['11','22'] contributes to both the '11' and '22' buckets so rates are compared apples-to-apples by place of service.

## Schema Integrity

Validates required fields, enum values, conditional requirements, and date validity per the CMS TIC in-network-rates schema. Also checks file freshness and expiration date sanity.

**Method:** Penalty-based deductions from 100, capped per category.

per_missing_required_header_field	5
per_header_conditional_issue	2
freshness_warn	5
freshness_error	10
item_missing_fields_pct	×5 (cap 30)
empty_rates_pct	×0.5 (cap 5)
price_missing_fields_pct	×10 (cap 30)
rates_without_providers_rate	×200 (cap 20)
expired_prices_pct	×0.5 (cap 5)
file freshness warn days	45
file freshness error days	90
expiry far future years	3

## Provider Mapping

Verifies that all provider\_group\_id references in in\_network items resolve to an entry in the provider\_references array. Validates NPI integrity via Luhn checksum and EIN integrity via IRS-issued 2-digit prefix.

**Method:** Weighted component sum (not purely penalty-based).

<b>provider_resolution (60%)</b>	$\text{resolution\_rate\%} \times 0.60$
<b>npi_validity (30%)</b>	$(100 - \text{invalid\_npi\_pct} \times 5) \times 0.30$
<b>ein_validity (10%)</b>	$10 - (\text{invalid\_ein\_pct} \times 0.1)$ [0% invalid → 10 pts, 100% invalid → 0 pts, linear]

## Code Coverage

Tracks every (billing\_code\_type, billing\_code) pair and flags unrecognized CMS TIC code types, format violations for CPT/HCCPS/NDC, and duplicates (same code appearing in multiple in\_network items).

**Method:** Penalty-based deductions from 100.

<b>per_unknown_code_type</b>	3 pts each (cap 20)
<b>format_invalid_pct</b>	$\times 0.5$ (cap 10)
<b>duplicate_code_pct</b>	$\times 2$ (cap 20)
<b>codes_not_in_reference_pct</b>	$\times 0.5$ (cap 30) — only when reference set is loaded

## Pricing Sanity

Detects invalid rates (negative, zero, extreme-value) and distribution anomalies (per-class P95/P50 spread, per-code max/min ratio, flat-rate distributions). Exact counts are used for all validity checks (negative, zero, extreme, dimension validity). Percentile-based checks (spread, IQR) use reservoir sampling —  $k=5\,000$  global,  $k=1\,000$  per (billing\_class, negotiated\_type) bucket — so memory stays bounded on large files. Per-code max/min spread is exact (all rates seen, no sampling).

**Method:** Penalty-based deductions from 100.

<b>negative_rate_pct</b>	$\times 5$ (cap 20)
<b>zero_rate_pct</b>	$\times 3$ (cap 15)
<b>extreme_rate_pct</b>	$\times 5$ (cap 25) — ffs only
<b>class_spread_excess</b>	$(\text{spread} - \text{threshold}) \times 2$ , max across (billing_class, negotiated_type) buckets (cap 15)
<b>per_code_high_spread_count</b>	$\times 0.1$ (cap 15)
<b>invalid_negotiated_type_pct</b>	$\times 3$ (cap 10) — rates silently dropped
<b>invalid_setting_pct</b>	$\times 1$ (cap 5) — silently defaults to wildcard
<b>invalid_severity_pct</b>	$\times 1$ (cap 5) — silently normalised to ''
<b>institutional_with_service_codes_pct</b>	$\times 1$ (cap 5) — extra key variation
<b>invalid_service_code_pct</b>	$\times 2$ (cap 5) — encode raises ValueError
<b>extreme high by billing class</b>	professional: 25000.0, institutional: 2000000.0, both: 2000000.0, default: 500000.0

<b>extreme low</b>	0.01
<b>spread warn p95 over p50 by class</b>	professional: 5, institutional: 10, both: 10, default: 5
<b>per rate context max min ratio</b>	professional_codes: 20, facility_drg_codes: 50
<b>flat rate iqr p75 threshold pct</b>	5.0
<b>flat rate min rates to check</b>	100
<b>spread min n to flag</b>	50
<b>per code min n to flag</b>	3

## Dashboard: MRF Identity Key

(ingest-time — not stored in report JSON)

The dashboard assigns a persistent `mrf_key` to each MRF so that all validation runs of the same file are grouped together in the score-history view, even if the payer re-exports the file at a new URL.

<b>Tier 1 — entity + plan_id</b>	Used when both <code>reporting_entity_name</code> and <code>plan_id</code> are present. Key input: <code>plan &lt;entity&gt; &lt;plan_id_type&gt; &lt;plan_id&gt;</code> . Stable across monthly re-exports.
<b>Tier 2 — URL hash</b>	Fallback when <code>plan_id</code> is absent. Key input: the raw file location URL/path. Entity name alone is not used — a payer publishes multiple distinct plans under the same entity name and without <code>plan_id</code> they cannot be safely distinguished. A URL change produces a different key.

The key is a 16-character MD5 hex digest of the input string (case-insensitive, whitespace-stripped). **This run:** `mrf_key = 24784283559c3712 · entity = — · tier = 2 (URL hash)`

## Provider Geographic Coverage

(supplemental — does not affect score)

Geographic analysis is a supplemental feature computed on demand after scoring completes. It does not affect any scoring dimension — it is an observational overlay to assess the breadth and distribution of in-network providers.

<b>NPPES</b>	CMS National Plan and Provider Enumeration System — monthly full-replacement CSV. Maps each NPI to its primary registered ZIP code.
<b>ZCTA centroids</b>	GeoNames US postal code file. Maps each 5-digit ZIP to a (latitude, longitude) centroid for map placement.

**Process:** Extract all NPIs from the MRF file → resolve each NPI to its primary practice ZIP via NPPES → aggregate provider count per ZIP → map each ZIP to a lat/lon centroid via ZCTA → render as a weighted heatmap (intensity  $\propto$  provider count per ZIP).

**Limitations:** NPIs absent from NPPES (recently issued, test NPIs, EINs) are excluded and reduce the geocoding match rate. Location reflects the provider's NPPES-registered primary address, not necessarily where they accept this specific plan. Map viewport covers the bounding box of ZIP codes representing 90% of total provider count, dropping sparse geographic outliers.