



Original Article

# Carc/Rarc Mapping Using AI Tools

Justin Pathrose James  
Sr Technical Analyst, LA Care, Los Angeles, CA.

Received On: 17/12/2025

Revised On: 19/01/2026

Accepted On: 26/01/2026

Published On: 08/02/2026

**Abstract:** CARC and RARC are two of the core code sets used in healthcare claims adjudication. They explain why a claim was paid, denied, or adjusted. Your search results confirm this across multiple authoritative sources. CARCs explain the reason for a financial adjustment on a claim or service line. They tell you why the payer changed the billed amount (denied, reduced, bundled, not covered, etc.). They appear on ERAs (Electronic Remittance Advice) and EOBs. They are standardized across the industry and maintained by X12. Remittance Advice Remark Codes (RARCs) are used to provide additional explanation for an adjustment already described by a Claim Adjustment Reason Code (CARC) or to convey information about remittance processing. Each RARC identifies a specific message as shown in the Remittance Advice Remark Code List. There are two types of RARCs, supplemental and informational. The majority of the RARCs are supplemental; these are generally referred to as RARCs without further distinction. Supplemental RARCs provide additional explanation for an adjustment already described by a CARC. The second type of RARC is informational.

**Keywords:** Healthcare Analytics, X12, EDI Transactions, Claim Adjudication, Artificial Intelligence.

## 1. Introduction

CARC (Claim Adjustment Reason Codes) and RARC (Remittance Advice Remark Codes) are vital for HIPAA as they standardize the communication of claim adjustments, denials, and payment information from payers to providers, facilitating efficient electronic processing and ensuring alignment with HIPAA's Administrative Simplification regulations for clear, uniform communication throughout the U.S. healthcare system. They offer universal codes (CARCs for reasons, RARCs for context) on remittance advice, simplifying revenue cycles by assisting providers in comprehending payment discrepancies, implementing corrective measures, and minimizing confusion caused by proprietary denial messages.

CARC and RARC codes play a vital role in the healthcare revenue cycle. Without these standardized codes, claim processing would become problematic as providers would find it difficult to comprehend the reasons behind claim denials or adjustments. There are numerous reasons that highlight its significance. The main reason is that it helps providers comprehend why a claim wasn't handled as expected, avoiding misunderstandings and delays in payment.

Billing specialists can likewise implement corrective measures and resubmit the claims correctly after interpreting the adjustment reasons. CARC and RARC codes assist providers in comprehending the reasons for denial, along with the essential supporting documents that were absent in the initial claim submission. Numerous billing software systems utilize these codes to streamline claim monitoring and produce reports on denial patterns. This assists providers in pinpointing areas where the workflow and claim

submissions can be enhanced. Moreover, comprehending these codes guarantees that healthcare providers comply with payer regulations.

## 2. CARC/RARC Reporting

### 2.1. Challenges

Linking Claim Adjustment Reason Codes (CARC) to internal denial codes and managing Recovery Audit Contractor (RAC) audits are very tough for the healthcare providers, mainly in terms of data consistency, administrative workload, and different interpretations of billing regulations. Difficulties in CARC Mapping To accurately map CARCs onto a provider's internal systems for denial management is challenging for the following reasons:

**Code Specificity and Variation:** There are multiple CARCs and related Remittance Advice Remark Codes (RARCs) which make it hard to interpret and categorize correctly. Using a general CARC code (like CARC 299 for an ineligible provider) instead of a specific one (such as CARC 147 for an expired contract) leads to not only insufficient information but also difficulty in automated posting.

**Payer Discrepancies:** The payers might have different CARC/RARC combinations for the same denial reasons or may use non-standard codes which would make it even more difficult for the providers to maintain consistent mapping logic among all the payers.

**Absence of Standardization:** Despite HIPAA-compliant coding mandates, there may be conflicts in advance between providers and auditors/payers as to the specific guidelines used to assess suitable payments which result in inconsistencies in the use of codes.

**System and Reporting Constraints:** The success of denial mapping depends on the strength of a provider's internal systems for workflow, reporting, and analytics, and the provider's system may not be powerful enough to manage the intricate CARC data.

**Difficulties in RAC Audits** RAC audits do face the problem of being a unique challenge, as they are aimed at uncovering the improper Medicare payments (overpayments and underpayments alike):

**High Volume and Tight Deadlines:** Providers frequently find themselves in a situation where they must handle a sudden rise in medical record requests along with strict submission deadlines which may result in documentation submission delays or errors.

**Documentation Errors:** Mistakes in documents could lead to denials and loss of revenue even in the case of necessary treatments if the documentation is incomplete, missing, or hard to read.

**Aggressive Audit Stances and Incentives:** In the past, RACs worked on a contingency fee basis which led to them becoming quite aggressive in their search for overpayments. Although there is some supervision, the worry about the use of aggressive techniques is still there. Burden on

**Administrative Resources:** The administrative time and resources spent in responding to audits, collecting huge amounts of documentation, and going through the long multi-stage appeals process are so huge that they can even take the focus away from patient care activities.

**Cash Flow Disruption:** A healthcare organization may face a sudden, major cash flow disruption due to recoupments of claims that are years old because of post-payment audits.

**Lack of Clear Guidelines:** One of the most difficult things to deal with has been the persistent lack of a common understanding between auditors and providers on the exact payment and coverage standards which makes it hard to avoid denials proactively.

**Evolving Audit Topics:** New RAC audit topics are being continuously approved by CMS, which means that providers must keep up with the new audit areas. Providers can check the CMS website for the approved topics for preparation.

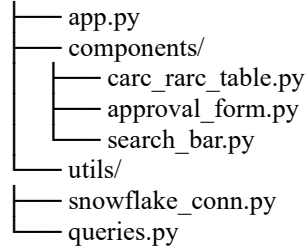
2.1.1. Review and Approval Console using AI

The Console built in AI will help

- Search by edit code
- View AI-generated CARC/RARC
- Compare with existing mappings
- Approve or override
- Write back to a curated Snowflake table

2.1.2. Streamlit App structure

streamlit\_app/  
|



2.1.3. Snowflake Connection Utilit

```
import streamlit as st
from snowflake.snowpark.context import get_active_session
```

```
@st.cache_resource
def get_session():
    return get_active_session()
```

2.1.4. SQL Queries

```
GET_EDIT = """
SELECT
    edit_code,
    edit_description,
    carc_code,
    carc_description,
    rarc_code,
    rarc_description,
    ai_result:carc_code::string AS ai_carc_code,
    ai_result:carc_description::string AS ai_carc_description,
    ai_result:rarc_code::string AS ai_rarc_code,
    ai_result:rarc_description::string AS ai_rarc_description
FROM CARC_RARC_REVIEW
WHERE edit_code = ?
"""
```

```
UPDATE_MAPPING = """
UPDATE CARC_RARC_REVIEW
SET
    carc_code = ?,
    carc_description = ?,
    rarc_code = ?,
    rarc_description = ?,
    approved_by = ?,
    approved_at = CURRENT_TIMESTAMP()
WHERE edit_code = ?
"""
```

2.1.5. Streamlit UI (Full App)

```
import streamlit as st
from utils.snowflake_conn import get_session
from utils.queries import GET_EDIT, UPDATE_MAPPING

st.set_page_config(page_title="CARC/RARC Review Console", layout="wide")

session = get_session()

st.title("CARC/RARC Review Console")
st.caption("AI-assisted mapping review for claim edit codes")

# Search bar
```

```

edit_code = st.text_input("Enter Edit Code",
placeholder="e.g., E1002")

if edit_code:
df = session.sql(GET_EDIT,
params=[edit_code]).to_pandas()

if df.empty:
st.warning("No record found for this edit code.")
else:
row = df.iloc[0]

st.subheader("Edit Code Details")
st.write(f"***Description:**
{row['EDIT_DESCRIPTION']}")

col1, col2 = st.columns(2)

with col1:
st.markdown("### Existing Mapping")
st.write({
"CARC": f"{row['CARC_CODE']}",
{row['CARC_DESCRIPTION']}",
"RARC": f"{row['RARC_CODE']}",
{row['RARC_DESCRIPTION']}"
})

with col2:
st.markdown("### AI Suggested Mapping")
st.write({
"CARC": f"{row['AI_CARCODE']}",
{row['AI_CARCODE_DESCRIPTION']}",
"RARC": f"{row['AI_RARCODE']}",
{row['AI_RARCODE_DESCRIPTION']}"
})

st.divider()

st.subheader("Approve or Override Mapping")

new_carc = st.text_input("CARC Code",
value=row["AI_CARCODE"])
new_carc_desc = st.text_area("CARC Description",
value=row["AI_CARCODE_DESCRIPTION"])

new_rarc = st.text_input("RARC Code",
value=row["AI_RARCODE"])
new_rarc_desc = st.text_area("RARC Description",
value=row["AI_RARCODE_DESCRIPTION"])

approver = st.text_input("Approver Name")

if st.button("Approve Mapping"):
session.sql(
UPDATE_MAPPING,
params=[
new_carc,
new_carc_desc,
new_rarc,
new_rarc_desc,

```

```

approver,
edit_code
]
).collect()

st.success("Mapping approved and saved.")

```

### 2.1.6. How the UI Work

#### User Flow

1. Enter an edit code
2. UI fetches:
  - Existing CARC/RARC
  - AI-generated CARC/RARC
3. User compares side-by-side
4. User approves or overrides
5. Mapping is written back to Snowflake

### 2.1.7. Data Requirements

You need a table like:

```

CREATE TABLE CARC_RARC_REVIEW (
edit_code STRING,
edit_description STRING,
carc_code STRING,
carc_description STRING,
rarc_code STRING,
rarc_description STRING,
ai_result OBJECT,
approved_by STRING,
approved_at TIMESTAMP
);

```

## 3. Usage of the UI

This UI provides the user:

- A governed review workflow
- A clean interface for SMEs
- A feedback loop to improve AI mappings
- A single source of truth for CARC/RARC logic
- A Snowflake-native solution (no external hosting)

## 4. Conclusion

The usage of latest technologies will help health plans to create better mapping for CARC/RARC thus help provide more accurate data for audits. Providers will find it more meaningful to use more streamlined approach rather than existing options used to create the mapping.

## 5. Conflicts Of Interest

No conflicts of interest

## References

- [1] X12.org, "Remittance Advice Remark,". [Online]. Available: Remittance Advice Remark Codes | X12.
- [2] CMS.Gov,"Health Care Payment and Remittance Advice".[Online] Available : <https://www.cms.gov/medicare/coding-billing/electronic-billing/health-care-payment-remittance-advice>.
- [3] Esteva, A., Kuprel, B., Novoa, R. A., Ko, J., Swetter, S. M., Blau, H. M., & Thrun, S. (2017). Dermatologist-level classification of skin cancer with deep neural

- networks. *Nature*, 542(7639), 115–118. <https://doi.org/10.1038/nature21056>
- [4] Kourou, K., Exarchos, T. P., Exarchos, K. P., Karamouzis, M. V., & Fotiadis, D. I. (2015). Machine learning applications in cancer prognosis and prediction. *Computational and Structural Biotechnology Journal*, 13, 8–17. <https://doi.org/10.1016/j.csbj.2014.11.005>
- [5] Litjens, G., Kooi, T., Bejnordi, B. E., Setio, A. A. A., Ciompi, F., Ghafoorian, M., ... Sánchez, C. I. (2017). A survey on deep learning in medical image analysis. *Medical Image Analysis*, 42, 60–88. <https://doi.org/10.1016/j.media.2017.07.005>
- [6] Coudray, N., Ocampo, P. S., Sakellaropoulos, T., Narula, N., Snuderl, M., Fenyö, D., ... Tsigos, A. (2018). Classification and mutation prediction from non-small cell lung cancer histopathology images using deep learning. *Nature Medicine*, 24(10), 1559–1567. <https://doi.org/10.1038/s41591-018-0177-5>
- [7] Bibault, J.-E., Giraud, P., & Burgun, A. (2016). Big data and machine learning in radiation oncology: State of the art and future prospects. *Cancer Letters*, 382(1), 110–117. <https://doi.org/10.1016/j.canlet.2016.05.033>
- [8] Ching, T., Himmelstein, D. S., Beaulieu-Jones, B. K., Kalinin, A. A., Do, B. T., Way, G. P., ... Greene, C. S. (2018). Opportunities and obstacles for deep learning in biology and medicine. *Journal of The Royal Society Interface*, 15(141), 20170387. <https://doi.org/10.1098/rsif.2017.0387>
- [9] LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, 521(7553), 436–444. <https://doi.org/10.1038/nature14539>
- [10] Miotto, R., Wang, F., Wang, S., Jiang, X., & Dudley, J. T. (2017). Deep learning for healthcare: Review, opportunities and challenges. *Briefings in Bioinformatics*, 19(6), 1236–1246. <https://doi.org/10.1093/bib/bbx044>
- [11] Topol, E. J. (2019). High-performance medicine: The convergence of human and artificial intelligence. *Nature Medicine*, 25(1), 44–56. <https://doi.org/10.1038/s41591-018-0300-7>
- [12] Kather, J. N., Krisam, J., Charoentong, P., Luedde, T., Herpel, E., Weis, C.-A., ... Halama, N. (2019). Predicting survival from colorectal cancer histology slides using deep learning: A retrospective multicenter study. *PLoS Medicine*, 16(1), e1002730. <https://doi.org/10.1371/journal.pmed.1002730>