

7.1.0 Schema Changes

Last Modified on 03/12/2024 12:14 pm ACDT

Schema changes between CareRight V7.0.0 and 7.1.0

New Tables/Views

- service_provider_external_identifiers
- appointment_shared_assessments

New Columns

- person_detail_setups.description
- saved_reports.category
- saved_reports.read_only
- refunds.reversal_user_id
- receipts.reversal_user_id
- f_statement.reversal_user_id
- person_detail_setups.primary_meteor_code
- letters.letter_content_id
- data_sets.read_only
- default_templates.width
- default_templates.height
- default_templates.designed_for
- default_templates.product
- custom_patient_labels.description
- custom_patient_labels.width
- custom_patient_labels.height
- custom_patient_labels.designed_for
- custom_patient_labels.product
- tools.show_on_patient_header
- attachments.url
- documents.expiry_date
- general_services.cost
- general_services.item_id
- refunds.reversal_timestamp, :date
- receipts.reversal_timestamp, :date
- f_statement.reversal_timestamp, :date
- admissions.episodes_count
- data_collection_elements.domain
- data_collection_elements.loinc_code
- data_collection_values.loinc_code

- metrics.description
- branding_settings.online_payment_link_expiry_in_hours
- branding_settings.document_link_expiry_in_hours
- branding_settings.signature_link_expiry_in_hours
- branding_settings.pathology_and_radiology_link_expiry_in_hours
- branding_settings.paid_invoice_link_expiry_in_hours

New Indexes

- patient_tracking(patient_id, location_id, deleted) idx_pt_pld
- appointments(patient_id, location_id) idx_patient_location_id

Changed Views

- None

Changed Columns/Indexes

- f_patient.pt_num index is now unique
- metrics.description renamed to metrics.title
-

Changed Tables

-

Deleted Tables/Views

- dassa_client_episodes
- medicare_ree_records
- medicare_ren_records
- medicare_res_records
- medicare_rle_records
- medicare_rfc_records
- medicare_rli_records
- medicare_rox_records
- problem_suggestions

Deleted Columns/Indexes

- attachments.note_id

Other

Clinical notes migration to attachment mappings

```
ClinicalNote.where("text LIKE '%note_editor/attachment%'").each do |cn|
  cn.update_links
end
```

```
insert into attachment_mappings (attachment_id, record_type, record_id)
  select id as attachment_id, 'ClinicalNote' as record_type, note_id as record_id
  from attachments where note_id is not NULL
```

Letter attachment migration to attachment mappings

```
Letter.where("letter_text LIKE '%/attachment%'").each do |letter|
  letter.update_links!
end
```

Reversal timestamps

```
update f_statement set reversal_timestamp = stat_date where reversal_timestamp < stat_date
update receipts set reversal_timestamp = date where reversal_timestamp < date
update refunds set reversal_timestamp = date where reversal_timestamp < date
```

Force SQL Server compatibility level to minimum of 2014

```
if value.first["compatibility_level"] < 120
  sql = <<~SQL
  ALTER DATABASE #{database_name} SET COMPATIBILITY_LEVEL = 120;
  SQL
  execute(sql)
end
```

CareRight 7.1.3 Changes

Receipt Numbers

We have swapped from referring to the *f_counters* table and the *ct_rec_no* field to a proper database sequence (*receipt_number*).

- Removed column *ct_counters.ct_rec_no*
- Create sequence *receipt_numbers*

This does not change the definition of affected columns that referenced this number. However it changed the **behaviour**. Importantly, where a transaction rolled back; **previously this would discard any attempts to assign a new receipt number**.

From this release, this will **continually increment the receipt numbers even if the transaction is rolled back** This may manifest as "missing" receipt numbers; but is an expected behaviour.

See also: <https://learn.microsoft.com/en-us/sql/relational-databases/sequence-numbers/sequence-numbers?view=sql-server-ver16>

```
existing_data = CounterSet.where(ct_number: 1).first
start = 1
start = CounterSet.next_counter_value("CT_REC_NO") if existing_data
# historically went into:
# f_statement.stat_ref
# refunds.number
# receipt.number
check = (Statement.maximum(:stat_ref) || 0) + 1
start = check if check > start
check = (Receipt.maximum(:number) || 0) + 1
start = check if check > start
check = (Refund.maximum(:number) || 0) + 1
start = check if check > start
case ActiveRecord::Base.connection.adapter_name.downcase
when "sqlserver"
  execute <<~END_OF_SQL
    CREATE SEQUENCE receipt_number START WITH #{start} INCREMENT BY 1
  END_OF_SQL
when "postgresql"
  execute <<~END_OF_SQL
    CREATE SEQUENCE receipt_number START #{start}
  END_OF_SQL
else
  raise "Unhandled database type"
end
remove_column :f_counters, :ct_rec_no
end
```

While this change is small and should be routine, we are recommending customers have:

- Appropriate database backups in place
 - A regression test plan to validate receipt creation post deployment functions as expected.
-