
2015-03-12 - 12:30 pm - ETL Spec Group

Attendees: Jen, Erica, Mark, Michelle, Ryan

E: I only requested detail for carrier condition type.
J: That's fine.

J: I'm concerned about moving HCPCS into the measurement table. How do we track costs?
M: We should maybe skip that mapping of HCPCS => Measurement
R: We'd be OK to make a procedure occurrence from a HCPCS that would create a measurement, we should probably store the procedure_concept_id as 0 or else we'd be storing a Measurement domain concept_id in the procedure_occurrence table.
J: I don't care where things live, I just want their costs.
R: We should press CR to drop the HCPCS => Measurement mapping until we have a unified cost table.
J: Costs are highly tied to HCPCS. HCPCS aren't submitted for a physician to get a pat on the back, they are submitted so the physician can bill insurance and patients for money.
E: We don't have lab values, so measurement and observation table probably won't be used in the SynPUF ETL.

carrier_claims => device_cost
M: CR won't be mapping the ICD-9 Proc => device_exposure
Looks a lot like the carrier_claims => procedure_cost

outpatient_claims => device_cost
Looks a lot like the outpatient_claims => procedure_cost

carrier_claims => care_site
We were going to use tax_id to represent locations, but weren't sure what place_of_service_concept_id should be. Now we think "Office" suffices.

carrier_claims => fact_relationship
It is hard to represent these relationships using RabbitInAHat, so just look in the logic section of the fact_relationship table for our relationships.

carrier_claims => drug_exposure
We won't have dose information.

outpatient_claims => drug_exposure
Much like carrier_claims => drug_exposure
Attending provider will be the provider for all 45 HCPCS that might possibly be DE

DRGs in conditions??
M: I've written a paper using DRGs, but that might have been a bad approach.
M: Most people don't use these. DRG is what determines payment for an inpatient visit.
E: They look like they've been mapped to Observations
M: That might be appropriate. DRG represents a 1 line summary of why patient was in the hospital

inpatient_claims => observation
So clm_drg_cd will be mapped to observation table.
DRG codes are reused over time. 001 in 1997 isn't the same as 001 in 2012. It looks like vocabularies don't contain more than one copy of a code. We need to load in more of the historical DRGs.
We might need a better observation type than "chief complaint"

Tests
E: we should add tests for the ETL to make sure that we have valid data coming out of the ETL process.
E: There's also Achilles HEEL
E: The tests will be out in the open and help guide people.
M: We'll probably do the ETL in R, we might be able to build the tests in that, but we really need to have another all-hands meeting and see which technical people want to join in on the ETL building process.

2015-03-11 - 1:00 pm - ETL Spec Group

Attendees: Jen, Erica, Mark, Michelle, Ryan

Abstract:
We're not using DRGs yet. DRGs are date-driven so their numbers can be reused which is bad. So the vocabs are having issues handling these codes. Is the cost a visit_cost? Could DRGs be a condition concept? But we need to assign a cost to it? Can we hop from DRGs to SNOMED? Cost for DRG would go into visit_cost and you'd find the DRG condition and its link to a visit_occurrence to get the visit_cost.

J: We didn't map the line processing codes. These tell if a code was accepted/denied/etc.
M: We were going to pre-filter these lines in the ETL.
J: Where are we recording this? It's part of the ETL, but it's not technically a mapping so will it still show in the spec.
E: It's like in CPRD where there's a patient indicator flag saying if you can/can't use patient for research. If they don't have flag, they never make it into CDM.
E: RabbitInAHat is missing a good place for notes, like preprocessing notes. Like these indicators, or needing to pivot a dataset before loading it.
J: The indicators in CC file.
R: Do we reject associated condition too?
J: Yes
M: Some rejections are billing errors, some are duplicates. There is perhaps just a subgroup of rejections that we want to respect.
E: Which codes do we want to reject?
R: Let's determine them offline
E: And I still need to add in the descriptions for cohort definitions
M: I'll recalculate the percentage of mapped columns. I've gotten great feedback from the group. I'll ask for affiliations and those who don't respond won't make it on as authors.

outpatient_claims => visit_cost
Very similar to inpatient claims.
However paid_by_payer is simpler than in inpatient.
Payer plan period defaults to Part B

How do we handle it if we look for a payer_plan_period for a certain date and we don't find a record that matches? R: NULL

J/E: Seems good. If there isn't a payer plan at the time, then let the NULL reflect that.

carrier_claims => drug_cost
There are no copays, so NULL for paid_copay
This seems to be a lot like the cc => procedure_cost table mapping.
Perhaps as a repeat, the total_paid column could either be imputed, or taken from the line_alowd_chrg_amt field. If medicare is secondary, primary may allow more than medicare so total_paid may not always amount allowed from medicare.

We're forced to do imputation because SynPUF isn't providing allowed amounts as we'd find in LDS.

prescription_drug_events => drug_cost
There might be copays, but its not indicated if patient payment was a copay or coinsurance. We should go with coinsurance since it feels more generic.
We don't appear to have payer payment amount. We aren't confident that we can impute by taking total_rx_cst_amt - ptnt_pay_amt so we won't do it.
Medicare contracts out drug coverage to other insurance companies. Those companies are very secretive about their formulary.
Looks like total_rx_cst_amt is a rollup of ingredient cost, dispensing fee, sales tax, and vaccine fee.
There might be some sort of masking involved.

carrier_claims => device_exposure
The SynPUF has no quantity field for devices.

2015-03-09 - 12:30 pm - ETL Spec Group

Attendees: Jen, Erica, Mark, Michelle, Ryan

Christian delivered on the new condition types (carrier claim) but still needs to do work on the domain stuff.

Abstract for AMIA
Mark: uncomfortable saying "we looked at everything and here's what we think about it". Not so rigorous.
Erica: Patrick has yet to see it.
Mark: We need a better result than "yeah, looks like it should work".
Erica: It's interesting we did it open-source
Mark: Can we put up Don's website? Is there some quantitative something we can share? Can we say we mapped XX% of fields?
Jen: We can also point to the spec
Erica: The XX% is an unimpressive metric if the results are all garbage. We could say we covered XX% of domains
Jen: The XX% represents that we "kept" an impressive amount of data in the conversion. And SynPUF is already in the open, it makes it easy for people to know it's already pared down.
Erica: If we get through the cost tables, we can actually tell how many columns we've mapped
Mark: Says we went through this carefully and were able to map the information to a model. We'll send an example abstract. Who should be authored?
Erica: If it was me, it'd be people involved with the calls.

Mark: and data guys?
Erica: Rich Boyce will send out an abstract for feedback and only those who respond.
Mark: and there will hopefully be a manuscript that will allow us to include others. Erica should be the first author. She's managing the whole thing.
Michelle: Does white rabbit produce a percentage like this?
Erica: Nope. It also would be neat to at least see what was never mapped. I'll start another issue in WhiteRabbit on GitHub

Continuing the spec

Cohort Table
Erica: The beneficiary table comes with some flags for people. We could stick those flags in the cohort table
M/J: Agreed. And it will up our percentage.
Erica: Let's include a map from the column name to the cohort definition name (e.g. sp_chf => "Congestive Heart Failure". Looks like we have 10ish cohort-generating columns.
Had to write in the cohort_start_date/cohort_end_date logic by hand. The year source column wasn't present in our rabbit in a hat spec.

carrier_claims => procedure_cost
Will be generated at same time as procedure. Procedure info will drag along cost table and then create both occurrence and then cost so cost can use occurrence_id
Currency should be in US dollars (safe assumption)
Medicare has no copay, so all coinsurance
line_nch_pmt_amt - payer amount (what medicare paid)
line_bene_ptb_ddctble_amt - deductible
line_bene
line_coinsrnc - coins amount
line_allowed - amount allowed
We should probably take out all denied claims
    We'll talk about this later. Maybe we preprocess the data
We will keep claims that weren't covered
If I pay a copay for office visit under my plan, then the copay will most likely be covered by medicare as secondary. The copay will not be captured in the copay field. It will be captured in what medicare paid.
For medicare primary other secondary, the patient's balance will capture what secondary may or may not have paid.
line_bene_prmy_pyr shows how much the primary insurance (non-medicare) paid. Medicare will only cover the allowed amount of the remaining cost.
Jen: In case this is blowing your mind, Medicare is the easiest payer to understand
total_paid is the sum of the deductible plus coinsurance.
Jen: I need to research how a non-covered payment is reported because it wouldn't show up under deductible or coinsurance. But this is a very rare case

allowed_charge_amt - represents the total paid
payer_plan_period will draw from the part B plan (for carrier and outpatient claims)

inpatient_claims => visit_cost
No payments by line. So this means a visit_cost. We must have already thought of this. Someone might bill a lot of procedures in this context, but medicare sends back a bulk payment. Visit_cost really represents a "bulk cost"

Jen: How do we handle the separation of blood vs IP deductible? We should sum them and put them in paid_toward_deductible

Pass thru costs are paid by medicare. They are like infrastructure costs. People with more patients get a higher amount. Do we want to keep this information separate from the claim payment amount (clm_pmt_amt)? Do we want to rollup the info? Let's rollup.

Wrap up
Erica: Christian sent updated codes, so she'll add them to proper parts of spec. Draw arrows in procedure_occurrence/cost tables. Generate the document and put it on GitHub
J/M: Cohort definition names/descriptions

Next meeting: 1pm pacific (4pm eastern) on Wednesday, March 11, 2015

Tangents:
revenue codes
We have claims that have just revenue codes, but if the only place we can put a revenue code is in the procedure_cost table, we can't make a procedure_occurrence to associate the cost to. So we need to bring this up with Christian

---

2015-03-03 - 12:30 pm - ETL Spec Group

Attendees: Jen, Erica, Mark, Michelle

Discussion of cohort table and SEER Medicare data – potential issues in duplicate patients who have multiple primary tumors.

Planning for meeting with Christian about procedure code to drug exposure mapping and to discuss the structure of Medicare data and how to identify physician (carrier) claims.

Working on prescription drug events file.
- May be logic for handling different length NDC codes. Erica to find.

- Can we infer dose and dose unit from NDC (after mapping) and apply to table? (Ask Christian)

PQRS codes (as HCPCS codes)– observation tables? How do we handle HCPCS codes that map to non-procedure domains – duplicate records? Has implications for cost tables. Check with Christian.
- All procedures with no cost (or $0.01) get mapped to observations? Doesn't work because some PQRS codes appear to have costs associated with them. G8445 was the example we found in the data.
- Drugs get moved to drug exposure table

Remove denied claims at the outset?

Care site – identified by unique tax_id number. Tied to visit occurrence. Need to use fact relationship table. Need to talk to Christian about having the necessary concepts in the fact relationship table to make it work.

2015-02-27 - 11:30am - ETL Spec Group

Attendees: Mark, Ryan, Jen, Erica, Adler

Jen: SHouldn't Amy attend these meetings since she's going to create a SEER ETL with similar logic as what we are using for SynPuf?

Erica: Wants to get started on SEER and should be ok - we'll make it work.

Mark: She can get a lot of stuff started for us and we'll review it.

Adler joins for the first time and Erica gives an introduction to what we are doing.

Eria: Do we want to do anything for AMIA? Possibly put something about doing eTL in open source forum.

Mark: Focus on access to real data and put it into CDM. First public available claims dataset accessible to researchers in CDM format. Use what CMS gave us (raw data) and provide useful data. Now we have a platform, all of your tools should work. And you could develop your own tools and they will walk to the next CDM dataset. Works for policy and academic researchers as well as drug safety.

Erica: Submit something for a poster. Due March 12 at 11:59pm eastern daylight time.

Adler: generally they are one page.

Erica to suggest Mark start the draft. Mark agrees to start.

Erica: Need to get this to Patrick early to get his input.
Create Death table. Decided:

- you can have a max of two death records per patient.
    - beneficiary summary table
    - all claims files (carrier, inpatient, outpatient) using icd9 codes. If there are multiple records and dates from claims information, use the latest date in Death table.
- allows researchers to define what death information they can use (claim info or membership info).

Create Procedure Occurrence table:

- Carrier claims
- Inpatient claims
- outpatient claims

2015-02-25 - 12:00pm - ETL Spec Group

Attendees: Mark, Ryan, Jen, Erica, Amy, Michelle

Amy: There are differences between the various CMS formats. Spoke with Patrick Ryan because he wants a SEER medicare ETL from this WG. Why don't we keep a running document of differences between formats. Perhaps Amy will work up a SEER ETL concurrently with SynPUF efforts. Theoretically, the two efforts should complete around the same time. Might streamline the whole process so we have SEER Medicare ETL ready to go soon.

Erica: Once Amy is done, we, as a group, would review the work that she'd done and work out the last 20% of the data.

Jen: I've already walked through some of the SEER ETL, so you can check that out on GitHub, but this parallel approach sounds ok.

Amy: I could take the work done by Jen and incorporate it into WhiteRabbit.

Erica: Mantra should be just to keep pushing through. The hard stuff is still to come.

Amy: We have 3 cuts [of SEER]. Not sure if the structures are all the same. Can we share structures as part of the DUA? Can we compare against your cuts?

Michelle: We have many cuts and they are all the same. Sometimes the number of files changes, but not the structure of those files.

Jen: Are your cuts all from the same time frame?

Amy: We've gotten them over the span of 4-5 years.

Michelle/Jen: The variable names might change slightly.

Erica: We have a pre-ETL step, and LDS has the preferred structure? We might take SEER and restructure/rename columns so they match LDS.

Ryan: I still think the CMS family of might need some massaging.

Mark: SEER released claims up to 2011. It will not change its format. LDS did change its format as of data to 2011. SEER will maintain the old structure. SEER must have someone who is denormalizing the data. We don't know what the relationship between SEER and CMS is and what format SEER gets it's data. Maybe SEER is responsible for generating data that looks like LDS 2010 format all along.

There are other formats MCBS and such that we haven't surveyed the structure of.

Erica: Well, lets go finish out visit_occurrence.

Amy: I'll take off and go do my SEER ETL now and not follow the meetings.

Erica: So we resolved last time to use Alternative #1 at the start and then fall back to #3 if #1 doesn't work out.

Easiest case is if we see a carrier_claim or outpatient_claim subsumed by the dates of an inpatient_claim we…

Mark: Well, will an outpatient_claim every fall inside an inpatient_claim?

Mark: And what if carrier_claim coincides with outpatient_claim?

Erica: If I see carrier_claim on march 1 and outpatient_claim on march 1, are they the same visit?

Jen: Only one PoS per day is the easy case. So we'll use the visit_occurrence of the outpatient_claim.

Erica: What if you have two outpatient_claims on the same day?

Mark: There is a scenario where you go to doctor's office (carrier claim) and then get sent for MRI (outpatient_claim) then radiologist makes a claim (carrier claim). We'd lose doctor's office visit_occurrence

Jen: Well, we have the NPIs which we could match. This alternative #2.
Erica: Let's walk through some iterations: one OP, two CC, only one CC shares NPI with OP. If we have two OP and one CC. We won't combine the two OPs.

Jen/Mark: When we have LDS/SEER, we'll have PoS for carrier claims. This won't be as big an issue.

Erica: We'll need to process OP, the IP, the CC?

Michelle: I imagine you'd suck up OP into IP.

Jen: The only time you have OP stuff is in an ER visit. ER looks like OP in SynPUF. But your ER information is deleted if you are admitted. IP sucks up OP visit. Unless IP is overnight, then its flipped to OP visit. But there won't be two visits. But if you're OP observed for more than 1 days, you'll have OP and IP and OP end will be IP start.

Erica: so if OP shares end date with start of IP, it's a separate visit, otherwise it's subsumed?

J/M/M: Can you have an OP during IP? I think you'd have to be discharged. Which makes our logic work still.

Erica: Are we categorizing ER visits?

Jen: In SynPUF we cannot.

Erica: So what if there is an CC claim all by itself?

Jen: It's 0.. SynPUF doesn't provide enough detail.

Mark: This looks ok and we'll tweak this model based on reality in the data.

CC => condition_occurrence

- condition_occurrence ID is typically autogenerated
- icd9_dgns_cd_* and line_icd9_dgns_cd_* => condition_source_value
    - claim diagnoses are a shopping cart. They are the diagnoses that appear either in the line, or were extra included by the provider for billing reasons
    - line diagnoses are specifically tied to a procedure. each procedure gets a single diagnosis (in medicare...other carriers allow more) and that diagnosis is the primary reason the procedure was performed
    - Normally each procedure gets its own row. SynPUF crams 13 procedures per row. So normally each row has 8 claim diags and a single line diag. Here we have 13 line diags and 13 claim diags.
- We'll want to use the fact_relationship table to connect procedure to its line diagnosis
- Seems like we need a type on the condition that distinguishes between a claim and a line diagnosis
    - Can we use "header" for claim and "detail" for line?
    - Even if a CC visit is rolled up as an IP visit, we'd leave the CC as CC detail/header
- Start/end date are clm_from_dt/clm_thru_dt
- desynpuf_id is person_id
- No stop reason (typically NULL)
- Line diags get NPIs, it's NULL for claim
- Should we dedupe claim and line diagnoses?
    - If we're looking for chemo from a particular diagnosis, we'd first find the chemo procedure/drug exposure
    - Most algorithms enforce a gap of 30 days, so it's ok to have 2 of the same diagnosis on the same day
    - Truven has been rolled up, but cost data isn't as good
    - We can tackle rolling up later if need be
    - Raw records have their own duplication
    - Optum has tons of duplication which is where J&J started rolling up
    - Mark: If we can't simply data to be intuitive, we should be faithful to the original data

IP => condition_occurrence

- condition_occurrence ID is typically autogenerated
- There are no line diagnoses in IP claims
- These are all inpatient headers
- There's no "primary diagnosis" (maybe admitting diagnosis for SynPUF?)
    - Nope, admitting should be last number
- Provider is NPI of attending since we only have one field

OP => condition_occurrence

- condition_occurrence ID is typically autogenerated

- There are no line diagnoses in OP claims
- These are all outpatient headers
- There's no "primary diagnosis" (maybe admitting diagnosis for SynPUF?)
    - Nope, admitting should be last number
- Provider is NPI of attending since we only have one field

Erica: Asked Patrick if they have parameterized SQL to run to make an ERA. They are planning on having something.

Ryan: We don't have to ship Eras in SynPUF CDM csvs

Erica: We should, but we don't have to discuss them. They are a solved. We'll not worry about them until the ETL is ready for release.

Next meeting: Same time Friday?

2015-02-23 - 12:00pm - General Group Meeting

Atendees: Mark, Frank, Don, Ryan, Bill, Erica

Everyone is good at some language, not all the same language, how are we going to divide and conquer the development of the ETL process?

Can we agree on a set of technologies that we can all use to develop the ETL together and spread it across the group.

What is Frank's experience?

6-7 devs are all developing the WebAPI and most aren't Java developers. Taking the path where they aren't the fastest individually, but will be fast as a group. Working in a consensus-oriented fashion

ETL doesn't have to be in Java, but there are a lot of lessons already learned in that technology. There are ORM tools and CDM entity models, things that we might be able to harness and reuse.

However, off-shore dev-team is C# based and focused on CDMBuilder. Not expecting people to use C#, but Java has momentum in OHDSI community.

Mark: Some are more inclined to use Python, and that still feels on the table. Bill has a slightly different approach to CDMBuilder.

Bill: Uses OHDSI tools to build mappings and explore the data. But attempts to separate the syntactic restructuring from the semantic part of the ETL.

Uses perl/python to blaze a trail first, then dictates the lessons into a “serious” Java program.

Loading SEER into MongoDB because its hard to load into RDBMS because it is so wide.

If we’re producing something to be shared with OHDSI community, we should go Java. But in prototyping the approach, we should be free to run whatever we need to get it done.

Mark: We probably don’t want to burden people with overhead for prototype.

Is there a way to divide and conquer the ETL prototyping? Is Python a viable way forward?

Frank: Not looking to force people into uncomfortable framework, it’s been hard for him. There is a question of how much throwaway there is to the prototype? There aren’t “devs” to hand off to. We are the devs.

Mark: How much can we disassemble this process?

Frank; Today we take SAS or flat files and load them into SQL Server and that is the “raw” schema.

He’s also done some work of loading data into Mongo. Could get to a “person document” that then needs to be converted to CDM relational structure.

Maybe we preprocess on flat files to make some person chunks?

Bill: SEER data is the first try at this. Using Kettle/Python to get data into DBMS of some sort. Can each column be characterized into condition, demo, etc and drop those into Mongo in a way that looks like CDM domains?

Or you can use an RDBMS to create view to look at the columns in that category.

Frank: if goal is to load data into DBMS, is the goal to support all RDBMS for OHDSI?

Bill: Currently using Kettle so it should work like that.

Mark: slippery slope is that this becomes CDMBuilder v2.

Don: SynPUF is 20 sets. Python chomps through it with little trouble. Could put some code up that helps newcomers to OHDSI that are a bit more accessible. ETL is fairly straightforward. Let's go rapidly on it and then go back later and make it standard Java.

Frank: If goal is to make a quick ETL, don't know how much breaking it into stages is going to help. With small data, like SynPUF, you can cut corners where a 1.6TB dataset (Truven) will never finish.

Don: One size doesn't fit all in ETL.

Bill: We don't have a consistent mapping between datasets. There are lots of silos of ETL experts, but its hard to share.

Frank: Is goal to get people SynPUF, or is goal to let people make their own. Patrick Ryan only wants this ETL if also produces ETL for LDS and SEER.

Rest of community: WE NEED A SAMPLE DATASET. Even if ETL only does SynPUF, that is a huge boon to the community.

Don: Our SynPUF could be example of how to do future ETLs and why we do things in a certain order.

Mark: Is there a CDMBuider v2?

Frank: Nope. I thought I was joining that group. If we make SynPUF in CDM format, that is good enough. If we talk about an ETL that is thoughtful to the OHDSI community, we might be it.

Mark: We could have two goals, make SynPUF available, and then CDMBuilder v2? Do we have the time to do CDMBuilder v2?

Don: Should we split this up short-term/long-term.

Mark: our deliverables are left up to us. Certainly we have SynPUF in CDM as a goal, but reproducible isn't necessarily a goal? And further, should we spin off the v2 Builder into its own work group?

Frank: Hardest step in joining OHDSI is getting data into CDM format. Would love to see a more diverse CDMBuilder. But do we have a bunch of people banging on the door asking for an ETL framework? Is that what prevents people from joining OHDSI?

Mark: One of our clients is definitely feeling stymied by the ETL process. Most people don't even know this is a roadblock until they are way into it.

Frank: we'll be working with a vendor who's CDMing their data. One approach is making a conscious choice to meet with providers and convince them to do the ETL for us. Abstracting the process is horribly complicated and possibly unrealistic. Another option is that vendors can host the data in CDM in the cloud and then slap our APIs and tools on top of the data. Seeing 2 or 3 vendors adopting CDM as a delivery format.

Frank: The idea that you process on a per-person basis is key. Being able to parallelize and distribute the workload is essential.

Don: Is this more of a supply or demand issue? Does it help to provide people with a person-at-a-time example? SynPUF could be illustrative in this way.

Frank: In a file where there's 20,000 people with no sort order, how do you grab one person's data? In a DBMS, this is easy, but in a flat file, it's hard.

Lowest hanging fruit is to get the SynPUF out the door. Quickest win.

After that, do we look at Hadoop and distributed architectures? It's not the nearest-term goal.

Mark: CMS data is in very different formats. Is there an interim structure that is useful for ETLing this data. Can we find an intermediate format?

Frank: Why isn't intermediate format just CDM? If we find a different format, isn't it better than CDM?

Mark: But is there some structure we can use to dump raw data into before processing it?

Frank: But take visits. You can't parallelize visit generation because you need unique keys between visits. Some datasets come with visit identifiers, but those don't match the way CDM defines visits. It is hard to define an infrastructure that spans all the data sets. CDMBuilder was an attempt.

Mark: Feels like CMS family of data might have an intermediate representation. Or should each mapping be done raw.

Frank: Dataset is valuable. Anyone participating in ETL will learn a lot and that is valuable.

Don: We've hit our time.

Mark: I still think we find a generalization for the Medicare family. We'll continue on the spec side and then we'll brainstorm more about how to do implementation. This was great blue sky.

2015-02-23 - 10:30am - ETL Spec Group

Attendees: Mark, Ryan, Jen, Erica, Amy, Michelle

Start with Jen's slides on visit occurrence. Found glaring issues with SynPUF while prepping the slides.

Invalid date ranges: Claim date is a range that will encompass all line-item dates, so office visits can end up being 24 days long. Not right!

Physician claims == carrier claims

How do we handle claim date ranges in carrier claims?

We would like to take the line date for each procedure, but SynPUF doesn't include it.

Just use the claim_from as the line date? This isn't real data anyway.

Each clm_id will be unique to a provider.

Can't tell if we have two 99214s on the same day or on different days.

Could end up with a visit occurrence spanning 365 days.

Premier doesn't give day or month, so granularity is often missing. They use a visit order field to give an order temporally, but SynPUF doesn't even have that.

All datasets have weird things like this. SynPUF is to show off tools, not represent research-grade data.

Sometimes deduping conditions makes sense, but not procedures. We probably won't dedupe conditions here since they are spread across a claim.
Do procedures have units? Some procedures record the number of hours. We have a quantity field. Should we roll up codes and put in the quantity? Let's not. Let's just make duplicate rows.

Maybe we could find duplicate codes and then bump date on each duplicate? But 99214 and 99213 are roughly the same thing and should bump the date, but won't.

Ok, so all procedures have start date.

Did anyone analyze how often claim dates are a range as opposed to a single day? Only 5% of carrier claims have date ranges and 12% of outpatient claims. So let's keep it simple.

We'll revisit this when we get to SEER and LDS. There *are* valid date ranges with no specific visit information in SEER/LDS -- dialysis observation and home health certification. Dialysis is billed once per 30 days. These are "procedurally correct" ways to bill these services, but how we'll turn those into visit occurrences is perplexing. The codes sometimes indicate the number of times per week a patient visited e.g. 2 times or 3 times a week. How do we store this information so that we can accurately count number of dialysis visits?

This is analogous to building drug eras were sometimes the quantity/days supplied aren't reported and eras break. Sometimes you need other tricks to handle these situations.

Collapsing dates for visit occurrences:
Facility bills room and board in inpatient_claims, physician bills time and services in carrier_claims.
We discuss the three options that Jen proposed.

What is a visit? Is it an appearance of a provider at a place? Or is it a visit by a patient?

Most people are just creating visits for both inpatient and carrier claims. No one questions that an inpatient_visit is a visit and that an office visit is a visit, but when a provider visits a patient in the hospital, what is the right approach?

For alternative #2, we only have 3 NPI numbers. We run a risk of not having all the proper NPI numbers to match against the carrier claims.

OutIns did #3 for their simple ETL. We had MANY more inpatient visits than seemed appropriate at the end.

Perhaps we need to distinguish between an inpatient facility and inpatient provider visit.

Goal should be to structure the visits so we can disentangle these kinds of visits.

How do we handle outpatient_claims that overlap with inpatient_claims?

We really need to get at the philosophy of what a visit_occurrence represents in CDM.

It feels like a separate visit when a get a separate bill from the doctor after they visit me in the hospital.

The majority seem to do #3, inpatient_claims make a long inpatient_visit and provider visits are also recorded.

This starts to matter when we incorporate costs. There is now a visit cost table. But if costs are rolled up to visit level but we have mini-visit while in hospital, where does mini-visit cost go?

We all like #3, but can we label visits so we can separate out physician visits from the hospitalization? Can we label the mini-visits as outpatient? Do we need a new concept_id?

If we label mini-visits as outpatient, how do we distinguish between genuine outpatient visits and mini-visits?

There are two flavors of inpatient and outpatient: facility vs provider. How do we distinguish these four combinations?

We don't want to over-count inpatient visits in Achilles.

Kind of an artifact of US billing that facility and service are billed separately.

Maybe we can abuse the visit_type_concept_id to let us distinguish if the source was the facility file or the carrier file.

A lot of times we're trying to count utilization and we want to know how many hospitalizations happened. If we have mini-visits, we'd WAY over count hospitalizations.

In CPRD, J had to keep track of whether conditions came from the HES vs GOLD data. So granularity is really achieved at the condition/procedure level in this case.

#1 is preferable, but there will be situation where a physicians will bill a 30 day window (e.g. a home health review), but a patient might be hospitalized for a week in the middle of that. If you're in observation in an outpatient setting, you can also have these windows.

If we don't do #1, how do we avoid "outpatient" diagnoses that have been rolled into an inpatient visit? Sometimes we need to know which file a procedure/condition came from because there is a research algorithm that relies on the source file from which that condition/procedure came.

Of course, this is lost in SynPUF, but LDS and SEER provide this level of detail.

Let's kick the tires on #1 on SynPUF and see if it works. If not, we'll fall back to #3 which is easier to code that #1. Just think of the permutations of date ranges and figure out how to handle each?

We'll think through #1 with caveat that we can back off. Then we can review at next meeting and then present to Patrick Ryan for 10-15 minutes after that.

Skilled nursing facility claims are tricky too, but don't appear to be in SynPUF?

Next meeting: 2/25 at sometime.

Can Jen review at a high level the differences between SynPUF and LDS/SEER. e.g. enrollment and no SNF etc would be helpful.

Tangents:
LDS data set
Currently, only OutIns has access to the LDS dataset and will show the structure to the rest of the group.

2015-02-20 - ETL Spec Group

Attendees: Mark, Ryan, Jen, Erica, Amy, Michelle

SynPUF's Observation Period is unlike other data sources. For Patrick, thinks benefit isn't in having an ETL process that takes SynPUF all the way to Achilles. Patrick thinks the value is having a builder that is close to real-world dataset

SynPUF is close in flavor to other CMS datasets, but enrollment is just funky.

SynPUF appears to be a simplified version of CMS data. Things that are missing:

- Modifiers
- Revenue codes
- Line vs claim date
-

Still makes sense to do SynPUF before LDS and SEER. Nothing we do here will be lost, but the ETL for SynPUF is not plug-and-play against LDS/SEER.

SynPUF ETL was much simplier for Jen to implement than SEER. We'll end up asking a lot of more intricate questions with LDS/SEER.

The SynPUF process is a good learning experience for the team.

Is SynPUF as perfect subset of LDS/SEER? Well, maybe conditions/procedures, but visit information might be more detailed in LDS/SEER.

The ETL might run with light modification on other datasets, but other datasets are more rich and that ETL process won't take advantage of that.

OHDSI is trying to get ahold of 5% medicare sample (LDS)

There's also the MCBS which comes with CMS data.

There's also the chronic condition warehouse.

There's also the virtual lab by CMS and that data hasn't been seen by our group. It'd be nice if that lab supported OMOP CDM.

LDS is similar to SEER in structure.

The SynPUF data will allow people to kick the tires on OMOP's toolset.

For provider, we'll keep the Tax Num in the provider_source_value so we can track groups of doctors.

inpatient_claims => Provider
inpatient_claims.prvdr_num is the NPI for the facility. We could turn those into facilities. But can't hospitals be broken into smaller chunks? But since these are inpatient claims, generally they are all the same for a single acute hospital. This number is more likely to vary in the outpatient facility setting.

inpatient_claims => care_site
We'll set PoS for all this to 8717 - inpatient hopital and the place_of_service_source_value will be "Inpatient Facility". Theoretically, if a hospital serves as inpatient and outpatient faciity, they should have different provider numbers for different settings. Mayo Clinic might show up 15 times with 15 different numbers.

outpatient_claims => care_site
We'll set PoS to 8756 - outpatient hospital and place_of_service_source_value to Outpatient Facility

We'll link a provider to a care_site because providers can attend many care_sites. If we want to look up high vs low volume hospitals, this linkage is nice.

In SEER, need to use PoS vocabulary for Physician claims. They will list 21 for inpatient setting. Depending on how we do visits, the inpatient claims and physician claims that both refer to treatment in inpatient setting might result in doubling up of visit occurrences. We'll deal with this a bit more later.

Attending is primary doctor assigned to patient, op_physn is the one that performed procedure, ot_phsyn_npi is available for other doctor.

carrier_claims => visit_occurrence
carrier_claims == SEER's NCH file

If have three rows in this file, what represents a visit? One row might represent multiple physicians. Different clm_ids means different visits. SynPUF doesn't provide line-level detail like in SEER. So we'll need to revisit.

In SynPUF, all we have is claim date, so we can't split visits into lines. In SEER/LDS, we have claim dates and line dates, making it easy to split into separate visits.

How frequently is clm_id duplicated in SynPUF? None.

So each clm_id represents a new visit (only for SynPUF).

Autogenerate the visit_occurrence_id? Can we hash? Let's look at other sources for visits first, then see if we can't pull that off.

Looks like we have multiple providers per clm_id, how do we handle that? Most expensive visit? Most expensive is defined by line_allowed. Ideally, we'd sum charges by provider and select the one with most charges.

We don't have PoS for SynPUF, so we'll set visit_concept_id to 0

visit_type_concept_id are all "derived from claims"

care_site isn't knowable in carrier

visit_source_concept_id is 0. We don't even know what this does.

inpatient_claims => visit_occurrence
Each clm_id represents a visit. The same clm_id might appear in multiple rows. Rollup claims by clm_id.
clm_admsn_dt and nch_bene_dschg_date represents start/end of visit.
visit_concept_id is "inpatient", visit_type is claims.
at_physn_id will be the provider_id

outpatient_claims => visit_occurrence
each clm_id is a visit
provider is attending

claim to/from is start/end
care_site is prvdr_num

Tangents
SEER vs SynPUF structure looks a bit different. LDS 2010 vs 2011 is structured very differently. Little things like dates are split apart in SEER. Trying to make this “plug an play” across different CMS datasets might be difficult. Maybe we need to massage raw data into similar structure for all datasets.

Even slices of Truven CCAE are different.

Let’s focus on getting SynPUF done, then SEER, then see what we can generalize for LDS.

Claim vs line:
The claim is an invoice, and claim diagnoses are diags that appear on lines or are generally applicable to the patient. At line level, there is a single diag that appears which is principal reason for procedure. So each line might have different dates of service. Sometimes claim diagnoses are not related to a specific procedure.

This sounds like long-term care which has long dates and represents a summary of an entire month at times.

How do we reconcile visits from carrier and inpatient that both refer to inpatient setting?
If there are carrier claims that fall within the inpatient visits, switch carrier visits to the corresponding inpatient visits. We’ll lose the provider_id from carrier claims. But procedures will still carry providers and be associated to those visits.

So if inpatient visit from March 1-30, and carrier claim from March 2-10, we’d use the inpatient visit for carrier claims. Jen will refine this logic offline and share with the group. We have seen examples where outpatient visits occur while patient is inpatient. This may be an artifact of the data.

2015-02-17 - General Meeting Notes

Attendees: Mark, Ryan, Jen, Michelle, Erica, Bill, Amy, Don

ETL Spec group has most of phase 1 completed. We’ll work on Phase 2 starting on Friday.

Don wrote a python script and shipped it to Ryan. Ryan made some slight changes and uploaded it to Github.

Don has DDL for postgres (with year for beneficiary table) and combined the beneficiary table. He has a non-spec ETL for one of his developers.

Beneficiary record is goofy so Don tacked on the year to each beneficiary row

Generally if similar records are scattered across different tables, life is made hard (e.g. beneficiary info being in three separate tables).

Lee has opted to keep the beneficiary data in separate tables.

Bill discusses this post (http://forums.ohdsi.org/t/etl-approaches-for-file-data/317) on his approach for ETL

- The overall approach is to take the source data, rename/restructure most fields in the source data to the source_value columns that OMOP uses
- Jasper/Clover can interact with MongoDB and turn that into relations for an RDBMS
- Can embed more difficult logic into this process
- Working on a way to feed a configuration file into a tool and have it do all the restructuring based on the configuration
- Mongo can do MapReduce and other complicate logic
- Erica says that CDMBuilder uses a similar approach, but that most times a simple select statement doesn't end up being sufficient for recasting/restructuring data
    - Most conversion is done in C# now, not in SQL
- The "fast" part of mongo is not having to have a schema ahead of time
- Mongo can be sharded easily

Total SynPUF data is ~70GB unzipped.

Discussing problems with missing concepts. If the missing concept is in a well defined vocabulary (NDC, ICD-9) then the source value must be garbage.

Is Lee's environment our formal environment? Nope.

Who's doing the implementation of the ETL spec? Undecided for now. Still evaluating technologies and who might take on the challenge.

We might make it a group effort, led by ETL veterans (e.g. Bill, Erica, Frank?)

Perhaps potential implementers can have a separate meeting to figure out approaches and responsibilities. Ryan, Don, Bill, will participate in meeting. Will be Monday's meeting.

Primary goal is "get it done" rather than "do it reproducibly". Try to avoid expensive, proprietary software to make it work.

Don will post URL to browse the SynPUF data he's loaded.

2015-02-17 - ETL Spec Meeting Notes

Attendees: Mark, Ryan, Michelle, Jen, Erica, Amy

We might start developing on beta version of Vocab v5, but our ETL will probably not be ready before Christian has released the final version of v5. (Concept ancestor, conditions, etc are still a bit buggy).

The claims dataset is pretty vanilla, so we can just build the ETL and then plug in the fixed vocab at the end.

Location table.
We only have the county from the beneficiary file. We'll put the text of bene_county_cd into the location.county column. We'll possibly need to make a mapping for the StCM table to translate county code to county string.

sp_state_code will go to location.state

Concat sp_state_code and bene_county_cd for location_source_value

For locations, they find the universe of counties and states and often find that there are counties that belong to no state.

Person table

- sp_stat_code, bene_county_cd are lookup for location_id
- desynpuf_id will become person_id
- bene_sex_ident_cd is gender_concept_id (need to double check if male == 1)
- bene_birth_date => year, month, day of birth (YYYYMMDD format) day is always first of month in source data
    - There is probably at least one date of birth that isn't set to first of month
- bene_race_cd => race_concept_id
    - 0 unknown
    - 1 white
    - 2 black
    - 3 other
    - 4 asian
    - 5 hispanic
    - 6 NA Native

    - Hispanic is not in the Race class, so we'll set hispanic to "other" and ethnicity will be set to "hispanic"
    - ethnicity/race_source_concept_id
- For speed we'll find the proper concept_ids for each crosswalk offline
- person.gender_source_value will be translated to human-readable text (male/female vs 1/2) (see tangents)

Observation period. Now the pain begins:

- We have shoddy information for SynPUF. We have number of months of enrollment, not actual months enrolled.
- We must decide how to handle this awkwardness
- bene_hi is Part A/B, bene_smi is Part B, plan_cvrg is Part D, bene_hmo is HMO coverage
    - Do we want to split these fields into payer_plan_periods
- Part A should be superset of other coverages
- Normally we don't research HMO patients because capitation and bundling provides insufficient claims data on a patient
- What does bene_hi == 0 mean? People might opt out of Part A.
- Do we want claims to be complete within observation period?
    - If so, Part A, Part B, no HMO
- How are we using payer_plan_period? The recommendation is they should be non-overlapping.
- If only Part A, we only have inpatient claims
- If only Part B, we only have outpatient
- If we're just looking at mortality rates, OutIns sometimes doesn't require Part B data, so this is a use case against A/B/No HMO
- Observation period should represent "a period when claims data is complete" but "complete" may vary from party to party
- The rule is no overlapping observation periods
- Amy tucked HES (hospitalization) time periods into Cohort table for CPRD ETL
- We can use payer_plan_period to track enrollment periods and use some sort of super set into the observation_period table
- We could concatenate A/B/No HMO/D and create a period for each combination
- In Optum payer plan period doesn't overlap
- We have birthdate, so if given 5 months of coverage, did someone qualify in that year? Then they would have had the last five months of the year
- Or 11 and above is eligible for the year
- If you're 80 and you only have 5, what happened? Is that your death date?
- Seems like a safe assumption for 95% of people that after someone starts, they stay until they die (at least for Part A)
- This problem is unique to SynPUF, so why are we trying so hard?
- OutIns can come up with simple algorithm offline

    - Thinking of doing payer plan periods first, then rolling them up into an observation period
    - This loses information about gaps in coverage
- There is a patient that died in Feb 2009, but was credited with 12 months of coverage
- Ryan asked why we can't use first claim, last claim date for start/end of observation period
    - Reason this is a Bad Idea is someone might have coverage (e.g. opportunity to have claims data) but not actually have any claims data reported (which is informative...if you're not going to the doctor, we might assume you're healthy)
- We are allowed to have overlapping payer plan periods, so this is what we'll do
- Birth date, death date, and months in which claims were paid
    - If someone had claims in Feb, we can't "right align" the plan date to the last five months. The five months must cover Feb
- If a person has 0 for a column, they aren't enrolled
- If they have non-0 for a column, we say they are in for the whole year
- We truncate the enrollment by birth and death dates

Provider:

- Provider table will de-dupe all providers
- tax_num_* is perhaps a proxy for group number
    - They are like a "business ID"
    - Maybe could capture this in the care_site
    - We'll stash tax_num in the provider_source_value and now provider is distinct combos of tax_num and NPI
- Carrier_claims are all physician claims so these could be considered outpatient
    - But we don't have PoS info and these could contain claims from an inpatient setting

Next Meeting:
Friday afternoon: Feb 20, Noon Pacific time

Tangents:
On translating the source_values? (from discussion on translating 1 or 2 to male or female for gender_source_value)
Should it be male/female (a translation) or kept as 1 /2?
If kept as 1 or 2, it is easier to joins on those values when writing queries.
When translated, it is easier to read as a human.

Seems like this might be considered on a case-by-case basis. Translating "1" => "male" is both easy to query and handy to read. Translating "412" => "old mycoardial infarction" is going overboard and the source code of "412" should only be used.

Source and standard concept_id columns (tangent from mapping race/ethnicity)

For source values, there may be source concepts in the concept table, sometimes the race_concept_id and the race_source_concept_id might be the same. For, say, premier billing codes, there will be no concepts in the concept table, so the *_source_concept_id column would be 0. For ICD-9s, the condition_source_concept_id most likely isn't 0, but points to a source concept in the concept table (which is different from the condition_concept_id which points to a SNOMED concept (as SNOMED is a standard concept)

To handle the billing codes, premier descriptions are matched to SNOMED descriptions using a python script (this was pre-Usagi).

This was added in CDMv5 and honestly seems like it is redundant in some situations (e.g. person table's person.gender_source_concept_id) but is very handy in the condition_occurrence table (e.g. condition_occurrence.condition_source_concept_id will point to ICD-9s in the concept table. Very nice. No more worrying about if an ICD-9 source_value has decimals or not).

In CDMv4, ICD-9 lived exclusively in StCM table, but in v5 all mappings were "promoted" into the concept table and live next to "standard" vocabularies. StCM is now blank and used only for custom mappings.

2015-02-06 Meeting Notes

Attendess: Erica, Amy, Mark, Jen, Michelle, Ryan

Put new notes at the top
Use forums to call out follow up items revealed in meetings

While we discuss SynPUF for now, let's keep SEER/LDS datasets in the back of our minds as well.

White Rabbit scans the database. We'll use the scan from Addler for today. We might need a new scan once we have the data loaded up from Don/Lee

Rabbit in a Hat connects up the columns found in White Rabbit with CDM columns.

Perhaps we could build the ETL in chunks and only do a few tables at a time, make sure the tables are in order, then move on to other tables.

OutIns is uncomfortable with having SQL in the spec. Most of OutIns staff aren't fluent in SQL.

SQL in spec worked for Janssen because people speccing the ETL and people implementing SQL both speak SQL.

OutIns is ok with some examples, but still wants an english explanation to accompany any code snippets.

Some helpful examples are specifying where to fetch vocabularies.

Going to go the chunking route for development of the ETL.

Janssen builds a set of test cases and runs ETL on the test cases to ensure ETL doesn't suffer regressions as it is modified. Helpful to set up test cases that are hard/rare to find in test data.

Achilles is also useful as a way to ensure data is converted as expected.

Success is document along with implementation of ETL but OutIns would like to provide solid documentation for running of actual ETL process.

Not all data providers give the same exact cut of data. SynPUF are identical, and Don's scripts to download/set up data are an important part of setting up the raw data.

Generic CDM Builder should make these ETLs repeatable on other systems as much as possible.

WIth White Rabbit, they normally stick to 100K, 1M samples of the data. Some datasets have 118M patients!

Fake data generation is a beta thing. Avoid for now. Maybe some day it will generate a nice fake dataset that is like the real data, but for now it doesn't do that.

The result from White Rabbit is an Excel file with summary information about each column in the database. Each sheet is the table, with two columns in sheet per column in table.

Rabbit in a Hat reads the spread sheet and guides our process.

Today we'll get the table level done and maybe all of Phase 1 from agenda.

Once the document diverges from Rabbit in a Hat (RiH), there's no going back to Rabbit in a Hat for generation.

While someone drives RiH, someone else should take notes.

- beneficiary_summary table maps to:

    - Person desynpuf_id is ID across tables
    - Location (state)
    - Payer Plan Period (number of months)
    - Observations (sp_*)?
        - These are specific to this dataset. Just chronic condition warehouse flags.
        - Specific to the year
        - We could run our own algorithms to derive this information
        - Seems redundant
        - Chronic condition warehouse is an imputed value from ????
    - The payment information is a year summary
        - Erica: Data vendors generally don't do a great job with rollups
- carrier_claims
    - File is organized by procedure
        - HCPCS is matched to dgn1, prf1, tax_num1, etc
        - We refer to this as a "line". There are 13 lines per row.
    - clm_id are based on HCFA forms. They represent how a claim was builded and don't really represent anything important about the encounter
    - icd9_dgns* - diagnoses for patient
    - There is cost information in here for each line
    - Processing indicator
        - Says if claim was accepted/rejected
    - Line diagnoses are strongly correlated with procedure
        - It's the reason why the procedure should be paid
        - We need to figure out how to capture this relationship in CDM
        - e.g. rituximab is associated with cancer, we want to ensure rituximab was administered for cancer rather than arthritis
        - Maybe we capture this in a condition type
            - So far types are truven-specific
            - Just need to have a conversation and get new types blessed
    - There are no modifiers in this dataset?
        - Seemingly not
    - Also no units
    - For NPI, they pull all distinct NPIs and shove them into the provider table
        - The provider information has yet to be valuable
        - Providers end up having many specialties
    - Could be useful to map a physician claim to a facility claim
        - This would "link" facility and physician claims
    - In the past, dates have been used to match up visits.
        - If someone is in hospital, they aren't leaving to visit their GP
    - For synthetic data, dates and provider number might be sufficient to generate place of service since it is missing
    - HCPCS => procedure occurrence

    - Payments => Procedure Cost
        - How are JCodes handled (drug procedures)?
            - Does cost show up in drug_cost or procedure_cost
            - Historically they've put the cost in only one place
        - Maybe all JCodes go to drug, administration codes go to procedure/costs
        - Need to make sure vocabularies avoid having JCode create both a drug exposure and a procedure
- Inpatient Claims
    - segments need to be rolled up together to represent a single claim
    - No line-specific dates for procedures
    - claim_from_dt to claim_thru_dt represents the visit
    - There may be procedure dates
    - HCPCS aren't used and are all blank
    - Looks like they are using the same format as outpatient claims file
    - Provider number is specific to hospital
        - These will go into the care site
    - There are NPI fields that Michelle is suspicious of.
        - We could *try* to match them up to outpatient claims
    - Utilization day count is probably length of stay
    - admittng_icd9_dgns_cd is ICD-9 for admission
    - Only visit cost should be used here. There is not sufficient information for procedure cost
    - Where would discharge status go?
        - Amy puts in it observation file
- Outpatient claims
    - icd9 procedure don't apply, procedures are in HCPCS
    - In synthetic, we only have visit cost available
        - No drug costs or procedure costs
- Prescription drug events
    - pde_id is part d enrollment ID
    - [I got distracted by question about pde_id and didn't hear some things]
    - prod_svc_id is an NDC code (yay)

Now we look at which CDM tables have no arrows pointed to them.

- Observation Period
    - Probably comes from enrollment information
    - Can impute by the earliest and latest date of data
    - We could probably come up with a little algorithm that finds the start/end dates when dates are incomplete
    - As OutIns does, Amy also removes patients when they have HMO coverage
        - Means we can't infer completeness of data from looking at first/last date from data

    - OutIns enforces both Part A+B coverage and no HMO for a person to have a valid observation period
    - Erica has seen that most patients don't have Part D, but patients still end up having their drug claims appear in the data
    - For Part A, we can infer that if someone has 8 months of coverage, it's the last 8 months
    - We could put various types of coverage into the payer plan period
    - Let's remember that we're not doing _real_ research on SynPUF
    - The rollup of enrollment is very specific to SynPUF so we don't have to *really* nail this
- visit_occurrences
    - Drug events rarely get a visit_occurrence
- Device exposure is going to be generated by HCPCS, ICD-9-Proc (so carrier, inpatient, outpatient claims)
    - With device_cost, are we talking cost of device, or of procedure?
        - Or rather, from the raw data, what costs are reported?
    - These procedures will generate both a device_exposure and a procedure_occurrence

Side discussions

- SynPUF isn't a perfect proxy for LDS and other Medicare, which is a bummer
    - We'll need to redo the process for the other datasets
- For drug eras, they always build to 30-gap
    - Research supporting sensitivity
    - They sometimes remake the eras based on different windows for specific studies
- They comb the inpatient file for ICD-9s that represent death to verify/improve death information
- Ryan's notes for Martijn
    - When I select a box, hide other arrows from other boxes
    - Frequency is nice, please also show me the percentage of the total that frequency count represents
    - Clearly label that the CDM is CDMv4 or CDMv5
- Christian's dream is run data over Vocabulary and have vocabulary tell you what rows should be generated in the CDM
    - e.g. this CPT is a procedure AND a drug exposure

Follow-ups

- Mark will look at observation period straw man
- Erica will gather notes from Martijn and pass those along
- Ryan will make sure that White Rabbit is ready in-house at OutIns
- Ryan will ask Erica to migrate RiH work to CDMv5 and maybe post that to the github repo.

2015-02-04 Meeting Notes

Amy Matcho, Frank DeFalco, Erica Voss, Lee Evans, Don O'Hara, Ryan Duryea, Jen Duryea, Marc Halperin, Mark Danese, Claire Cangialose, Michelle Gleeson, Rich Boyce

Erica recommends process documented in Wiki (http://www.ohdsi.org/web/wiki/doku.php?id=documentation:etl_best_practices)
They were able to complete the document in 3 days
The team needed longer because they needed to get up to speed with OMOP
No programming until Document is done
Divide and conquer is tricky because tables call back to each other
There is a natural progression to walking through the tables

Don agrees
Observation periods and visits come first

White Rabbit/Rabbit In A Hat to drive the process

Jen is a good candidate for specification process. People with experience with the raw data are important in this process.

Need researchers as well (Mark) who know what they want to ask the data

Probably can knock it out in a few meetings. (7-8 hours) You also want the implementors present so they hear decisions being made.

We need a document owner. Erica? Mark? Probably Mark.

Tools are designed to be in a room together to do first iteration, then present that result to larger group.

Lee: At point we're ready to start ETL, what does that look like?
Don: If we had data in cloud, we could all look at it. We might end up with several different implementations.

Mark: Can we hear about ETL implementations that currently exist?

Frank: 5-6 years ago, CDMBuilder was built in .Net and SQL-Server based. Reality is that CDMBuilder is not for everyone. Only works with SQL Server and that's a hangup for some. CDMBuilder works well in an environment that mirrors Janssen

But if we want a more agnostic solution, we're looking at a rewrite though that would invalidate the battle-tested nature of CDMBuilder

CDMBuilder's architecture is a good template that we could mimic in a more universal environment.

JDBC would let us connect to other DBs. Maybe/maybe not use SqlRender.

Rich: has CDMv4 agnostified in Hibernate. Using that to save/query tables.

Frank: using an ORM is too transactional to be performant. Output of CDMBuilder should be flat files in the future. Then people can use bulk loader for their specific DB.

We still need an object model to represent the data within the CDM. Person is built as an object, then we spit them out into specific rows for the OMOP tables.

Mark: would be nice to have ETL in an accessible language that others can use. Implementers can have a discussion on approach.

SAS? Python? Java?

Mark: would rather do it well, rather than fast.

Frank: It's possible to get a solid starting implementation in a few days of effort and refine as we go along. Janssen has resources they can put towards development

Erica: getting something up and running early lets us see the data and make refinements sooner. We have an opportunity to define how builders should be done.

Implementation group: Michelle, Don, Frank, Lee, Ryan

Lee: Deliverables? Are we shipping the data to people itself?

Mark: Spec, implementation, and sure, ship the data.

Don: Sees four deliverable, [Ryan's note: I didn't catch what it was]

Erica: Once it's in CDMv5, run it against Achilles so we can throw that up on the web.

Lee: I can host the Achilles results.

Don: Beneficiary summary is missing year on some file names

Ryan: Source data needs some massaging before main ETL

Don: I can massage and load into PostgreSQL. Would have a download script that does a set of wgets to get all the files and load up the data.

Erica: Addler might have already done the loading of the raw data. No other way to do the White Rabbit process.

Mark: Let's contact Addler and integrate his/her efforts. Don, Lee, Ryan, get the data loaded post-haste. Get the white rabbit process started. Jen, Erica, Mark, Amy, Michelle,

Amy: sometimes SQL is a good way to concisely describe a business rule

Mark: game time decision, let's favor english unless it's too clumsy

Amy: Maybe use SQL as short hand and someone else can english-ify it later. Helpful for the implementers.

Everyone please email mark@outins.com so he has your email address.

Don: where do we put documents/notes/etc?

Lee: we could use the wiki?

Ryan: we could use github?

Don: GitHub ok with me.

Erica: GitHub ok with me.

Are we doing this in CDMv5? Yes!

Mappings already exist in Vocab v5, so that's nice.

Erica: At meeting I can walk through the tools and show people how they work. We might need to re-run white rabbit on Don's data.

Mark: walk us through a done one to see where we're headed

Erica: Don, Lee, let me know about database to use

Lee: I can install white rabbit after database is up.

Mark: Meet this week, then a meeting or two when Jen gets back (Feb 16th). Mid-March is AMIA submission.

Ryan will setup github and coordinate ETLers.

End 2014-02-04 meeting
