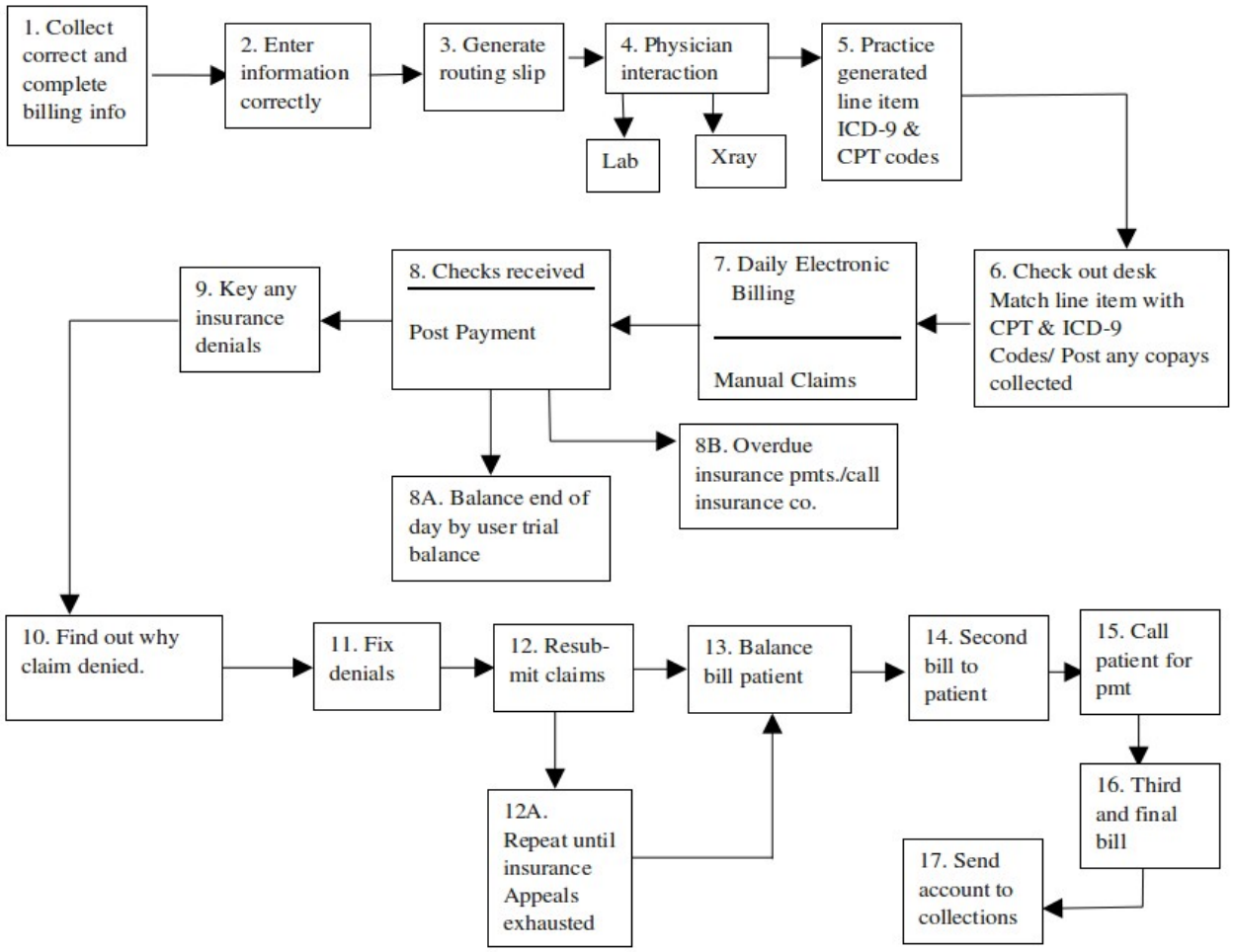
OpenEMR Practice Management Gap Review -DRAFT-

This document uses a typical medium to large practice work flow diagram for practice management and billing to analyze the gaps in OpenEMR, state requirements and recommend possible enhancements and work flow changes to bring the project up to industry standards.

Work Flow Diagram – Provided by Dr. Sam Bowen



Gaps, Identified by Block

1. Collect Correct and Complete information

1. Demographics
    1. Guarantor/Patient connection is weak. Insurance subscriber model is not adquate to describe the relationship. Demographics should support separate record type for guarantors that can also be patients and allow for a connection to be made from one demograpic record to another. The dependent patients can have connections to different guarantors for each insurance level including "self". The Insurance records should be attached to the subscriber's demographic record. This reduces the

redundant storage of information. Insurance information needs to be maintained in permanantly by effective date for claims processing. This is done in the insurance_data table now, but the historical data is not visible to the user.

2. Enter Information Correctly

1. Good validation of key fields is missing. A full definition of the requirements for each field should be available and configurable in the layout model. Some fields have rules that are insurance company, clearing house or billing service specific and should have 'transformation' rules loaded at the service providers level.
2. Fields such as postal code and telephone area codes should be validated agaist reliable tables and provide a warning that they may be incorrect.
3. Some fields are required only when other fields are filled/selected this chain of requirements must be supported in all forms
4. Free Text entries should be considered carefully with an eye toward pick lists and lookup tables whenever possible
5. Referencial integrity should be enforced where possible to prevent the removal of parts of the data in-advertantly

3. Generate Routing Slip (aka SuperBill)

1. The creation of superbill or routing slips is more complex than it needs to be. A tool more like the layout tool should be created that will allow user to create custom superbills that can be printed for offline use or used online in an intuitive way. The current Fee Sheet form works for this if configured well by a technician. A more modern UI that uses the same underlying structure is in progress. (See addendum for screen shots).

4. Physician Interaction (Orders, Lab, Xray)

1. The Superbill interface should be wrapped in the workflow in a way that includes other Physician driven orders
2. Providers need to be able to enter codes as they see the patient, after the patient visit or separately in some kind of batch mode (see 5. below)
3. Needs to add referrals support to the CPOE wrapper as part of the work flow. "Transactions" seems like the right place to use as the CPOE wrapper but needs to be extended for all types of orders
4. History, Issues, previous issues feed need to, optionally, allow the relevant Diagnosis codes to be used in the new encounters. Issues need better diagnosis code lookups to support this.
5. Level criteria for management code, map service code requirements to ROS + Diag.

Progress sheet based. Biller can see that when a specific diagnois is used a certain level of service is called for (91..,92...,93...) and code accordingly.

5. Practice Generated Line Item Coding of Services and Diagnosis

1. The Superbill interface must allow the user to easily see the medical notes so that coding can be done correctly when the Provider is not to coder. This is specific to services.
2. Alternatively the medical notes forms themselves could be preconfigured with the typical service codes as part of the form or allow the provider to pick the service from a Service search or Superbill pick list.
3. Diagnosis coding should be done at the time of the medical notes not left to billing forms.

6. Check out desk – coding review, co-pay collection

1. Front Desk and Billers need to be able to adjust and correct the Diagnosis codes and provide additional codes (up coding and HEDIS recovery). But these need to be approved by the Physician prior to being billable.
    1. Are there are any HIPPA issues around privacy? Maybe just a Not Coded warning
2. Collection of patient amounts due needs to be supported for all all kinds of payment methods and insurance eligibility, deductible and co-pay need to be readily available as part of the check out process. Payment methods include cash, check, ACH/bank draft, health savings card, prepaid plan support, etc. Interfaces to e-services for this would be helpful.
3. Payments need to able to be related to specific services and products, not just 'patient CO-PAY' need a full suite of payment types posted at the service line level
4. End day reporting is incomplete all the reports below should be batch-able (ie: run with one selection). Perhaps a 'closing' screen that provides an on screen review that can allow for corrections (logged) and then final approved printouts if desired. All official reports should be retained in an online report archive for audits.
    1. Need report that separates charges, payment by type, adjustments by type
    2. Separate cash drawer, deposit balancing report for cash
    3. Separate credit card balancing report by card type
    4. Daily report needs to include insurance payments and adjustments posted that day and method of payment (check, auto-deposit, etc)
    5. Report sales of none medical services or products
    6. Report sales of in house pharmacy separately
    7. Productivity reports by Practice and by Provider with both details and summary

options. Daily, Monthly, any other date range (daily would be the default)

8. Logging of posting information, ie who keyed the accounting data.
9. Out put of final daily transations to General Ledger in users format, provide QuickBooks, MS Money, CSV, perhaps customizable options for accounting services
10. Currently identified issues:
    1. Payments posted daily for previous visits show on previous date report, not on current day reports
    2. Adjustments, debits, credits and payments are all bundled into one column. Should be broken down individually.

7. Daily Claims and Statement Processing (Electronic, Manual, Out Sourced)

1. Claims need a preprocess that checks for obvious errors
    1. Uncoded or Not Justified (this is particially there)
    2. No Insurance or Secondary coverage carrier only
    3. ID# missing or incorrectly formatted for the target carrier
    4. Demographic data missing or incorrectly formatted for the target carrier
    5. Diag or Service codes that are not valid or typically create rejections
2. Need to be able to easily separate claims processing by claim types, (see addendum for screen shot of design for this).
    1. Electronic by Clearing House partner
    2. Electronic by Insco for direct processing
    3. Manual by Insco + Address for easy sorting and mailing
    4. Statement processing for patient portion, A/R and unisured. Needs to support payment plans as well.
    5. Interface with the "billing" company should be supported for those that don't want to do in house billing at all.
    6. Review of X12 batch (error logs, etc) should happen **PRIOR** to downloading or submitting the batch for processing.
    7. All batches should be saved and easy to access from the billing process, not just on the disk somewhere. Ideally the batch should be saved as an sql table instead of a flat file making it easy to tweak and recreate if needed.
    8. CMS1500 claims (perhaps all) should be able to be loaded into a WYSIWYG CMS1500 form for easy review, edit and print before and after submission.
    9. Issues/Questions with existing system
        1. How are manual secondary claims handled, manual/electronic Liability, Workers comp. Etc?
        2. How are NPS billed as supporting Physicians versus direct credentialed providers.
        3. Where are Medipass – Medicaid ID group MSO number kept
        4. EOB/EOP – Insco batch payments need to be recorded for the day that posted.

Related to closing the day
5. ERA posting should be automated from EDI response files (not just manually uploaded).

8. Posting of InsCo payments

1. Payment Posting EOB/EOP should reflect denials and rejections from automatic responses as well as allowing manual flagging errors for correction. This should generate a patient billing note and drive a work list. Common problems, incorrect demographics, incorrect procedure, missing medical data.
    1. Payment posting screens need to be modernized to new UI standard as modal windows. Work list needs to be refreshed after each payment post, etc.
    2. Completed payment batches need to produce a batch posting review report **PRIOR** to actually writing the payments to the billing records. This will allow form easy correction of typo's with having to enter adjustments.
    3. Payments need to show on daily reports for the day posted not the encounter/visit date

9. Posting Insurance Denials

1. Claims denial model sometimes has really good ways to NOT say what the real issue is. Some of X12 mapping to english from the programmer crud by insurance co and clearing house.
2. Need a Denial Dictionary and a way to added new information to it
3. Add the ability to create new error types/codes for future use and training
4. Add the ability to create adjustment codes from the posting module

10. Claim Denial Research

1. Easy access the the claims with errors from work list, links to patient records and encounters (like in billing report now).
2. Easy access to Insco help desk phone # contact information for harder to figure denials
3. Need a learning module related to insco "special" rules that result in rejections.

11. Fix the Denial

1. Processing loop work flow needed.
2. Claims and their status (billed, paid, errors) should be visible from the patient view for the front desk.
3. Should be able to reopen an encounter from the patient enounter mode to correcting and rebilling.
4. Error processing
    1. Eletronic response files need to be processed for errors on batch and reported with each claims and on patient notes to billers to be worked

2. Manual posting of errors and rejections during EOB/EOP posting should be possible

12. Resubmit Claims

1. Billing process improvments for reprocessing, and error/rejection learning process
2. Easy access to previous claims, statements and billing by encounter ID, Patient, Insurance Co, date ranges, error code types.
3. Loop on this until appeals are exhausted, logging of the calls and attempts to fix the issues should be kept in the patient notes including batch information on resubmits, etc.

13. Balance Billed to Patient

1. Statement Billing process needs to be defined and solidified
    1. Running statement should be a automatic result of patient balances that are not waiting on insurance payments
    2. Creation of the statement format should be template based and allow for paper formats, simple CSV and 'plug-able' custom outputs for services

14. Second billing to patient

1. Statements need a backend rule set for number of months to 'collections' and rules on locking the record if contractually required to.
2. Automatic notices in patient notes for front desk, billers etc

15. Patient Called for payment

1. Reminder system and work list needed for calling patients for payment and working out payment plans if needed
2. Payment plan model and support tools needed

16. Final Bill

1. Same as 15 need tools to work this

17. Send to Collections

1. Export to collections should produce a CSV (may already) to be sent to agency if desired.
2. Notes in Patient record should not use existing 'generic value' area as this can be stepped on or may step on some user data
3. Patient record should be locked if contractually required to do so (prevent collections at the front desk).

This document was created by Tony McCormick of Medical Information Integration, LLC with advise from Dr. Sam Bowen and Dr. Robert Schaffer and their staff.

It is intended to be a guide for formal specification of a project centered around improvements to OpenEMR's practice management systems.

First Revision: 02/17/10