**OpenHIE Community Call 13 May 2014**

**Attendees:**

    Ryan

    Linda

    Joan

    Derek

    Larry

    Carl

Meeting recording available online here for 30 days: <http://www.conferenceplayback.com/stream/58597351/92787801.mp3>

**Agenda**

* New OpenHIM development update
* Review new save encounter workflow
* [https://wiki.ohie.org/display/documents/Save+encounter+workflow](https://wiki.ohie.org/display/documents/Save%2Bencounter%2Bworkflow)
* Validating terminology in the save encounter workflow
* Starting development of the save encounter mediator

**Minutes:**

RC - Thanks to Derek for running the two previous calls in my absence - have listened to both recordings

RC - Any other items for the agenda

DR - we need to clean up the diagram of the save encounter workflow to include the IL Infoman

RC - will include in discussion on the save encounter workflow

**New OpenHIM development update**

RC- spoke previously about coding some of the functionality in Node this has been going ahead. Progress is being made we have rebuilt the OpenHIM core to be more flexible and configurable, still in development but we have the core functions working in the test environment. Web application configuration for the core. Configure what has access to send messages to the HIM. Development of the consul has started. Have a web app that is hooked into the HIM Core and can do the functionality.

May have something to show in a month's time.

RC - will forward information for anyone who has specific questions.

DR - moving to an interpreter?

RC - not quite an interpreter but what would you see as interpreting.

DR - consul developed to see how the Core will be seeing certain messages.

RC - not how it is being built the consul is an actual web page that allows HIM configuration and once done moved into a scalable database, not a interpreter.

DR - test messages against this design compared to the previous design, is it faster? slower?

RC - not tested yet have preproduction servers up but it needs to be a bit more stable before we do that kind of testing.

RC - expecting it to be faster as well as being able to scale out and this is one of the core reasons we went with the newer tool

DR - difference between scale up and scale out?

RC - scale up still a single server, is increasing the power of that server increasing RAM, increasing speed. There is a limit that you would reach when it would be better to buy the latest new technology

Scaling out looking at spreading across servers to perform as one unit so you don't have the limit of reaching the end of technology ability

RC - have a load balancer in the front and using the MongoDB database

RC - all the routing will be stateless, you can hit any node and get the same result.

LL - if your PoC when you fire up in the morning with load balancing will you get one node that you get to all day or will it be a round robin kind of thing

RC - won't always talk to the same node, the load balance will figure out what is best for you

DR - our transactions will be stateless if we ever had stateful it would require the same node or connection.

LL - does that mean every time you send to the HIM you are going to break connection? Should stay connected if you're a high volume PoC instead of breaking and reconnecting

DR - the load balancer extracts from behind the curtain, the load balancer purports to be the address you are connecting to

RC - load balancer keeps the connection, not to a single node but to the balancer can keep the connection open for periods of time and the load balancer will figure that out on the backend.

LL - using mongoDB for this wondering if we have a common graft database but realise how difficult that may be.

DR - will have an inRAM DB in future.

RC - there are options to do that in future

RC -comes down to what is needed mongoDB is more document based which is better for what we want to do

RC - not sure it's possible to have the same across registries

DR - is the database our audit log

RC - it will be, it is doing a bunch of generic meta data but not conforming to the profile yet but it will

DR - implementing jurisdiction would want a generation of certificates

DR - challenge of ensuring that certificates are renewed to prevent the breakdown of the system

RC - good mechanism to have to prevent that situation

**Save encounter workflow**

RC- Email thread has been going around through various communities

Discussions about how certain validations happen within the save encounter workflow.

RC - Captured consensus in the workflow diagram.

There is now a new actor, the Infoman, that the IL would manage, it's purpose is to provide interlinked registry info.

In a transaction there wouldn't be a reaching out to the different registries but instead the infomanager will have a cached set of information.

RC - what sort of validation do we want to do, they are listed in the diagram and the discussion section some items listed;  is a facility is valid and operational, provider is current and practicing and works at the facility referred to in the encounter and provides the service and the service is allowed at the facility.

RC - also noted that we might want to make them configurable so that it can fit with implementations that may not have the high level of information.

RC - added or updated them to be more advanced and also configurable.

RC - are we happy with this approach?

LL - need to add the date within range not future and not too far in the past but at least a check at the date of the encounter.

RC - internal IL check could do that

DR - number of validity checks need to check the performance effect.

LL - infoman doing validation of facility and provider at the end of the day you need a date within a range you are expecting.

DR - may lead to many exceptions with 11, 12 and 13 listed on the diagram, Achilles heel for the infoman is the services registries is very weak

RC - this is an ideal scenario but not always possible, could turn more of the checks on at the later stages as data quality improves

DR - could have the PoC monitor the exceptions queue every few hours to fix and resend exceptions

RC - the interoperability layer could handle that well, the IL could receive the message and later do the work

RC - the save encounter workflow would be the most comprehensive and it is a good candidate for asynchronous validating

DR - can decide on the type of document to do synchronously some need real time responses especially in drugs and allergies always treat as synchronous

LL - for terminology validation we need to add units, is it the right term and is it sent in the right units, also a feature can it be translated and are you allowed to translate

DR - blood pressure reading can be done in two readings must be sure we can translate between the two options.

LL - have machinery to look up all the possible units and then translated to the standard unit that is used in the system

DR - work done in ISO for what are the units used in health care HL7 updated to this standards based units of measure

LL - dictionary is older so we adapt what we have therefore- can we translate what is coming in to the units that we have for the default units for that term

DR - recommend that this is something to escalate that units is also what they need to ensure it is in the standard for medical units of measure.

RC - add a step in the term validation to standardise the measurements of units, do we want to try doing normalisation?

DR - code set to code set is a different conversion.

RC - may be out of scope for a terminology server

RC - Note this and pick up with the Terminology servers community as well

LL - after acceptance by the HIM and save into the SHR doesn't OpenMRS do some of the same checks

RC - not sure if it will check if a value conforms to a specific unit

LL beyond units you have five terms that come in it would do the same checks before persisting to the SHR

DR - are the checks in the API that we are using

RC - we are going straight to the API but the checks will still happen in the SHR, not sure how comprehensive

LL - just bringing up the question that with all that goes on in the HIM there will be checks in the SHR as part of persistence to the SHR

DR - shouldn't have checks and then checker checks we shouldn't have that. If we are adhering to the IHE spec, a repository just saves what was sent to it.

LL if you do clinical support you must do some validation of the units

LL - you have to get the data from the SHR to do the clinical support

DR - we are doing that at the PoC

LL - it would be in the HIM and the SHR

DR - design we are using for our HIE is not an EMR in the sky

LL - if you do it at the PoC you will have to have it originate in the SHR

LL - to do the support you must have identifiable discrete data that can be checked for valid terminology and so forth

DR - to do the clinical support at the PoC you would have to do that

LL - started with not checking the checker but with the OpenMRS may not be good at accepting verbatim what comes from the SHR

RC - previously indicated we want to break it down to some discrete data that is understood by the SHR for reuse, it should be in a form we expect it to be in.

DR - right now there seems to be an expectation that the SHR will be doing logic and validations; don't think this is the right place to do these...

LL - the data for clinical support will be coming from the SHR, do we agree on that. If the PoC asks give me all that patients data so trying to make sure we think of the SHR as being the source of that data

RC - in agreement there.

ICP workflow and where we can fit it into the save encounter diagram, perhaps put a note onto a different page. Split it out to two diagrams one as current and another with the workflow

DR - posted a PowerPoint presentation in the architecture group; will develop a PPT re: potential workings of the ICP service and post it to the Google group to start a discussion before we make changes to the diagram.

RC - good way to go forward.

RC - start the discussion from the powerpoint and start building another page to add these items to, to start fleshing out the issues.

RC - will go through the powerpoints in the architecture group.

RC - Pick up the discussions in the next call. Will contact the terminology server community around the units discussions