Session 3: Friday, April 8th - 10:15-11:15am

Location: Stetson

Session Title: Data Management, APIs, Catalogs

Organizer: Tim Reardon

Primary Notetaker: Katya Abazajian

Participants: Nate Ron-Ferguson, Sheila Martin, Ryan Gerety, Laura McKieran, Jodi Petersen, Stephanie Martinez, Laura Simmons, Sharukh Farooq, Tim Reardon, Amanda Macune, Bob Gradeck

Ryan; What tech do you use for the internal database component?

Sheila: Postgres

Laura S: Use MySQL and access using Toad

Tim: We handle data well internally, but we want to know how to share the data across multiple websites and get some advice to overcome challenges

Shahrukh & Jodi: Both looking to listen from people w solutions, see if need to learn new languages

Tim: Does anyone have a data catalog?

Nate & Shahrukh: Building one at the moment

Nate: Mostly spatial data, de facto data repository for a lot of city projects. Using GeoNode and it also handles nonspatial data.

Tim: Geonode have built in data catalog?

Nate: It does have indexing, Lucene which is an Apache indexing application

Laura M: San Antonio moving that way and is moving out of Weave. Would be nice to have a technology that lets people download what they’re looking at. Use Oracle for public reports aspect in sufficiently aggregated format, but programing that is extremely expensive to give the robust querying capability. .

Laura S: Web developer built a system for researchers looking to pull out complex data, but nothing for the rest of data.

Tim: has anyone done summary tables out of IDS that can query on?

Laura S: That’s a goal. Closest we get is that we get some of our nbhd data from our IDS.

Tim: We’ve done some individual web applications on specific topic areas, but creating the data to support that then requires a lot of custom tabulations and then that lives on its own as a separate application database that is hard to update and it lags behind.

Nate: Using Census API but we thought that it was a little slow because you have to grab each of the text files for each state. Wrote it in Python, split them out. Backend database creates applications for different projects. Store the API data in PostgreS and pull out in SQL queries.

Sheila: Why does it make sense to use API rather than getting the crosstab for each different project, does it really save time?

Ryan: Well we had the capacity to build from scratch to build profiles, so when there’s less technical capacity, it’s harder to make the larger decisions about databases.

Laura S: For the API, you still have to design the call for each particular table.

Ryan: If you had a computer scientist, they would say if you do it twice, you should automate it.

Tim: We download things and transform the fields into a standard that we use across years and append it over time. So having the API is good for comparing year to year. That’s all maintained in a PostgreS database. Our data librarian makes that into a tool that other people can use. So what we’ve done is basically replicating FactFinder. But then there’s an internal web browser that is for our staff. It’s a windows application.

Tim: How do other folks deal with duplicating vs. downloading FactFinder data

Jodi: We have one person and we go ask him. Which is not sustainable.

Laura M: We spend a lot of time manually calculating margins of error. Collapsing income groups, etc. to reduce MOE. We also link to results of search queries in FactFinder. It might be easier if you have a finite set of indicators.

Tim: To get to Shahrukh’s questions about new skills, R is really useful for doing MOE custom calculations or for standardizing. We also have a tool called KnowPlace that lets you pick different reports and gives you a set of MOE for different geographies, but that was tricky. The technical solutions then limit because it is a lot of work to add new tabulations.

*Bob joined*

Bob: Talk to your university librarians, they have a ton of experience in data management. We just rolled out new portal last year. Hired a company that has expertise in managing specific product to fix broken things or set up metadata standards. Use a consultant. On our end it’s training people who have the data to write metadata. Have also taken it further with data user guides. We want you to pull your data out and use it in Tableau or map. Add the metadata in Google Docs. Also looked at Open source and proprietary things. Did an RFQ to find info about open data products. Put all of this in a blog post. Also wanted to have data across more organizations without scaling out price too much. Had people answer with 8 pages or less, 10 questions. Asked about API capability, user base size, developer community, etc. Wound up choosing CCAN. Every quarter will allocate $5,000 for annual updates. Have a monthly call with developers, do a little sprint. Developed a whole open source ETL library using CfA fellowship so he had down time. Managed whole ETL process, wrote a white paper, wrote a little script. Have a library they can use. If the county posts a new API, can have the data updated.

Tim: Does anyone else publish API data? Has anyone used Pittsburgh’s API?

Bob: The city is actually pulling it in for the Mayor’s Dashboards. Would like to have the County Human Services system pull county level data from our API into their IDS. Dataset needs to have parcel numbers, though, and standardizing names is an issue. Documentation blog post: <http://www.wprdc.org/news/taking-data-documentation-up-to-11/>

Bob: PII is a huge issue because the IDS wants to release data but we want to make sure it won’t be a breach. Trying to figure out what software/processes are right for testing it now. Agencies for 311 data also have legal staff on this.

Tim: We found that CCAN had more stability issues. The model of having a consultant would be ideal because there were things that they were able to fix.

Bob: Consultant worked out because software is a service but we’re hosting it.

Tim: We have a fairly organized internal PostgreS database, but what we need is an internal interface for people to find data. Seems like Pittsburgh model is nicer for other people to push into it and it’s not from a lot of different sources, though it is nice that you get an API.

Bob: What we do is push the data to a server and then we just have a script that looks for it.

Laura M: But purposes are different. Tim’s database needs to be combining data to query it.

Tim: We have two purposes though, we need internal where people can pull a certain table, and then external needs to be a downloadable.

Laura M: But what about external dashboards to pull down data across multiple datasets?

Bob: We need to do that next.

Jodi: Everyone’s asking us for dashboards and then eventually will want to download the data.

Laura M: There is a huge misunderstanding about what a dashboard means. Some users want to “analyze” the dashboard data, but they don’t understand that they need the actual data for that.

Tim: So there’s a separate application for the Pittsburgh data profiles?

Bob: Yeah, need to figure that out too. Not a good publisher because we need to update the data more than once or twice a year, and it needs to be automated, so that’s the baseline, and need it to be scripted. Only able to do that now because we have the capacity because we have the big grant.

Tim: Some of what we do is gathering many to 1 data on assessors, for example, and collapsing around spatial data. Allows you to have a JSON file that describes relationships and you can package everything together into a zip format that it’s stored in. Metadata record, etc.