

Interview with Michael Byrne

Home Mortgage Disclosure Act Operations Lead at Consumer Financial Protection Bureau

<https://www.linkedin.com/profile/view?id=10690378>

Interviewed January, 2015

Interviewer: Bill Dollins

1. *Please describe your background in terms of education, training, professional experience, and roles. How has your background informed your leadership style?* I keep my bio relatively open and reasonably well documented at [LinkedIn](#). There are a couple of things that this bio lacks, however. I will say at a high level, there are a couple of very influential times in my life which were the catalyst for me professionally. In order of how I think of them, they are post-secondary education, first geospatial job, and staff to executives.
  - Post-secondary education - I have a bachelor degree in environmental planning and management from the University of California at Davis. While an undergrad there, I met an ecology professor who 10 years later became my major advisory when I went back to grad school at Davis. I have a masters degree in geography from UC Davis as well. Neither of these experiences were GIS specific (while my grad degree is in geography, I took no GIS classes - the education I got there was really about problem solving, critical thinking and geographic theory)
  - California Department of Forestry - my first job as a state employee in California (second job out of undergraduate school) was with a think tank looking at natural resource conservation at the ca department of forestry and fire protection. It was 1993. It was Esri license number 46. At that job I met many of the people who became instrumental in California land use policy and more importantly received classic GIS technical training from Esri. Back in those days, Esri version 4, everything was line command. In order to get through Esri sponsored training there was significant real problems solving to be had. When Esri version 8 came out, and things moved to heavy wizard based, I think a significant problem solving background was lost.
  - My first job out of graduate school, which I got because of people I had met ten years earlier from that first job with ca dept. Of forestry, was as a staff member to the California resources secretary a California cabinet member. During this position, which was short lived, because of the governor recall election that shepherded gov. Schwarzenegger in, gave grounding in working with senior executives.
2. *From an early point in your career, you have held a number of leadership positions in the geospatial*



**field. What was your motivation for starting down this path?** Well, I am not sure it was early in my career, but I do feel fortunate to have held positions as staff to two different state government cabinet members, and senior staff in federal government. I have always had a significant interest in supporting senior level policy agendas. I find that engaging in the policy discussions to be the most rewarding opportunities because they the most opportunity to affecting change across a broad range of people; basically it is the opportunity for the most influence. Really the motivation is all about ensuring data is infused at the policy discussion. A colleague of mine has this saying, 'the issue is the issue' which roughly translates to follow the policy agenda for today (e.g. Health care, education, climate change etc). The corollary to this issue is, it is almost never about technology. Technology, like geospatial tech, is always a supporting issue. We just support the policy functions that administrations want to develop.

3. ***Twice in your career, you have held the position of geographic information officer, first with the state of California and second with the federal communications commission. How did these roles differ at the state and federal levels? What prompted you to seek the FCC position?*** They were actually radically different positions, and I was very fortunate to have both. I was very interested in doing both, for different reasons. In California there was an extreme need for coordination across agencies, state, federal and local government. I spent most of my time on the phone building relationships, and trying to get people to talk with each other. I worked directly for the ca CIO (a cabinet position) but had little actual power and no real funding. It was a very challenging and very rewarding position all in one. In particular there was a huge need for the roll, but w/o funding there was a constant struggle to meet goals. For the FCC, the role was really two primary things; a) lead the software/technical development of the national broadband map and b) infuse data visualization and geospatial collateral into policy discussions. In both cases, I nearly wrote the role before getting the job. In California I helped lead a team to establish a strategic plan for California geospatial data coordination that called for the creation of the GIO, and then was lucky enough to get hired for that job. At the FCC, while the role was really about the national broadband map, I negotiated the title and other duties into the position. I felt it was necessary to give weight to the role to have the title and these other duties. I had spent quite a bit of time in California working on broadband policy, and my family and I were interested in a large move prompting looking at jobs in dc. This came up, and again I was lucky enough to be in the right time and place.
4. ***You have served on a number of boards and councils related to geospatial technologies, coinciding with your state and federal positions. How has your participation in these bodies, and the work of these bodies overall, affected your work? What have been the benefits and the drawbacks?*** This is an excellent and difficult question, as I both have received wonderful personal gain from these positions while sincerely questioning their value in terms of getting things done (e.g. Councils don't write code). I will say, since moving to dc I have spent less time with these. I have found significant personal and professional gain



from the positions held with the national states GIS council, and the national geospatial advisory committee. I cannot really say enough from the personal enrichment I have received from being involved in these activities. These types of things have required really tremendous teamwork, meeting and interacting with diverse stakeholders, listening and even writing some policy documents. The struggle is that in nearly all of these opportunities (except perhaps GreenInfo Network) the outcomes were very difficult to measure. Often, particularly w/ the national geospatial advisory committee, there are almost no outcomes. This is very frustrating.

5. ***You are known as a prominent advocate for open-source technologies, most visibly in your work with the FCC's national broadband map. When did you become interested in open-source and what prompted that interest?*** This is a good and interesting question. I am going to answer it more in the terms of the lineage of adopting open source, than it was this one moment. Around 2006 I was working at an agency that collected and published health statistics in California. During that time I started to get involved in broadband deployment policy (from the tele-health aspect). We needed to do two things, a) tell the story of lack of access to healthcare (e.g. promote the tele-health policy business driver for broadband access) and b) publish maps for tele-health and medically underserved areas. In a normal environment, trying to publish this kind of material would have taken it service requests, budgets and a truckload of overhead/management stratosphere.

Instead, an agency in California was hosting a metadata catalog off of a code repository knock-off of github. For free it allowed us to enter content, with links and in some cases data (e.g. zipped shapefiles), and then have these rows a) published and b) linked from office sites (e.g. the agency I worked for). So in this case is stumbled upon the opportunity to publish (remember this is 2006) without asking for permission. There were consumers of this data / content instantly, and it took me 0 budget, 0 overhead and next to now time to prepare.

Later as the GIO for California we ran one project with a contractor and the University of California where we asked them to develop something (some datasets and these had to be served). We left the means of doing so open, and the UC guys came back and said, let's do this w/ open source. I said sure. And it worked for a fraction of the cost and time. I remember asking my colleague why is no one else doing this? We just couldn't figure out why adoption wasn't off the charts.

Finally, at FCC we started experimenting w/ macs. Very tiny projects with extraordinarily fast turnarounds. There are two things we latched on to. First we realized we could use gh-pages (Github Pages) branches to serve static data. This may not sound ground breaking, but with simple iframes in these static data, it meant that you could turn on a micro site in say 1 minute, and use data services, say WMS or APIs, from the mother ship, in less than a day of development. And it was published! What we identified there was no need



for management overhead (ok, not exactly, but in the cots software model there are truckloads of steps which are required; in this OSS model there isn't). Second, was the notion that the computing infrastructure has changed radically. Now my laptop is a super computer. With a handful of very small libraries (python, pre-installed on all Mac's, PostGIS, and Psycopg), I could process, analyze and or model in any way possible any question at any scale w/o every asking permission to do so. We regularly spent weekends working on all of us census data or all of the national broadband map data. In the cots world, everyone we talked too would have spun up data management infrastructure, and servers and bla bla bla. We did it on laptops. Crazy fast time to market.

It is really this lineage that has proven to me time and time again, that the paradigm is really not about OSS vs proprietary software or open vs closed its about doing vs asking if I can do. I prefer the former.

**6. *Please describe the high-level benefits, as you see them, of open-source technologies for government? What are the drawbacks? Do the benefits and drawbacks differ at the state and federal levels?***

I think I see several benefits to the use of open source software; in general I see them as a) total cost of ownership (in particular cost to exit), b) speed of feature enhancement and c) fostering a culture of publishing/peer review

In use of open source many people to tout the cost of entry; that is that open source is generally very inexpensive to start with, and I agree for sure. However, this is not the primary cost driver for me. The biggest cost driver is something I tend to call the barrier to exit. The barrier to exit is the cost (barrier) to exit using one stack/architecture/software and switch to another. In nearly every proprietary software, particularly enterprise software, the barrier to exit is extremely high. Imagine trying to switch a companies use of SharePoint to something else, or outlook to something else etc. Open source typically offers a very unique opportunity because they are mostly build on common framework libraries. This means that there is a generally easier notion of migrating from one open source software to another. To say it another way, there is a lower cost of switching or exiting.

I have been to so many events where I hear presenters of proprietary software say, "Just wait until the conference, that is when this new enhancement is coming out." this doesn't happen with open source. Releases of enhancements are driven much more organically and typically much faster. It's not the largest client who necessarily drives what the next major releases are.

I believe that the happiest employees on my team are the ones who have the opportunity to publish and interact with others in the community, not just within the building. Those who can write a component, or a



visualization, or a map or a formal paper or whatever, and get broader recognition than just me or internal staff saying 'nice job' are much happier and much more motivated. Open source generally allows staff to publish their work. We in fact in our office have an open by default policy for internal software development, such that we have to argue to make software we are writing (whether is a large system or simple library) closed. So every developer we have writes in the open by default. We generally have happier folks.

**7. *What have been the major impediments to implementing open-source technologies in government?***

***How have you overcome these impediments in your work?*** I will be honest, I have had little impediments in the things I have personally implemented. I would say though, that we need to be very careful in a couple things. I am not running a campaign for open source in government. I am implementing technology to help solve policy issues. In the former I would say open source every sentence. In the later I say what data and systems provide the best architecture for the question. In this debate there are many variables and we need to be careful that there are no panacea answers here. Any time you hear someone claiming they have a panacea answer, be careful; I'll wager they don't. We need to make sure we are solving policy issues w/ the right technology.

All this being said, we have overcome impediments by just moving forward. I tend to like to do pilots with very small components then take these to increasingly larger implementations. Often this approach is hard to argue with if you have done things appropriately. That is if you can have a significant ROI, then it governance will likely implement the solution. Other than that, I also like to not ask for permission and just implement. This often changes the tide.

**8. *Aside from just releasing source code, open-source invites worldwide community participation. Is the federal government currently in a position to support this aspect of open-source development?***

***If not, are there policy changes that can better position it to do so?*** I think the federal government is coming around to the idea of accepting open source additions from all over the world, but it is not widespread yet. I am not sure it is even infant yet. I think generally speaking we are still in a world where it is predominantly one way; the direction of the federal government pushing open code out for others to see. What I mean by this comment is that the federal government now has a pretty good track record of having some open code, pushing code to github and the like. With some simple google searches you can find a number of fantastic repositories out there for code development. However, most of this is from the standpoint of 'here is my code, see I am open' as opposed to 'here is my code, make it better'. This later part, the real contribution of open code development, is still a ways off for the federal government. We generally have taken a good approach with efforts like [Project Open Data](#), the [US Digital Service Playbook](#) and the like, but I still don't see widespread adoption of a two way street yet. I think the federal government is open



to it, but time will tell how adoption works in this regard.

The open source adoption function might be construed as a cultural shift. If one wants a cultural shift, then yes, a campaign is often the right approach; it's just not the one I subscribe to. In this vein, the federal government today seems to be well poised to support a campaign. There are a couple of groups who have taken the roll of espousing open source use/adoption and agile development.

Let me caution some though here, as I return to another principle; there are no panaceas. Policies and campaigns help for sure, but it is not the only ingredient and not the ingredient I am focusing on today. Today I tend to focus on leading by example; I think we did this exceptionally in the national broadband map, and if one looks at what we did there one will see the playbook repeated long after we took that approach in many other things (the US Digital Strategy, Project Open Data, the USDS Playbook). My focus is making sure technology helps reduce cost and answer questions for policy issues (in my case now, the home mortgage landscape).

9. ***As you have led the implementation of open-source technologies in government settings, what developments have occurred that you found unexpected or surprising?*** I think government custom code does something that solves a problem, but often times it is highly specific and it is not intended to create a market (compare let's say the national broadband map code base, with say PostGIS; clearly the latter has a market interest, and former is crazy specific). With this light, I have been surprised how much we have not seen our code have pull requests or spawn other opportunities. This is likely to change as we get smarter about developing libraries, classes and more generic services that solve many problems. This is probably my greatest surprise. That open source development in government, while good, hasn't reached what the commercial advantages of open source has. Again, this is likely to change. On the negative side, the greatest surprise is that it often costs the federal government the same amount to procure something that is free, vs something that costs money. That's right it costs the same to procure either, so often the procurement establishment will gravitate to the something that costs something just because it is a more known commodity, rather than something that is free. As an example, let's say I need to get approval to download and install QGIS. In some places, I would have to tell the software approval guys to do that, and say go to this site called kyng chaos. Culturally, they struggle with how that is better than say going to Esri.
10. ***What are the key challenges to keeping systems built on open-source technologies sustainable over the long term? What strategies are you attempting to implement to mitigate these challenges in your own work? How do you see the challenges evolving over the next five to ten years?*** You know, I have found little problems with sustainability, or I guess I should say, I have found nothing worse or better than sustaining other proprietary solutions. It's all software. Let me illustrate with two stories;





- At FCC we implemented an open source NoSQL database to handle some metrics we were dealing with. The intent of the data was to demonstrate API usage, something we couldn't get from say google analytics. So we quickly stood up some servers, and let loose with having APIs insert rows into this database for every call to the APIs. When I left the FCC I think we were cruising to something like 700 million inserts. So that's kind of a lot. That system, the database, the servers, the web calls etc., had 0 upgrades, 0, patches, and 0 server restarts in a time period of about 3 years. That is just unheard of in other solutions. Okay, I would call this a non-critical system (no one dies if it goes off, it doesn't handle pay roll, etc.), but it was crazy, crazy cheap overhead.
- In another solution, we were running a version of PHP on Solaris. So in this case, Solaris is the proprietary system. PHP doesn't make compilers out of the gate for Solaris (their primary market is Linux), so we had to buy or develop our own compilers. The solutions we came up with required an older version of PHP to do this, because that was the only way we could compile on Solaris. This was predominantly b/c of the Solaris versions we were running, some older version, because that is what the IT governance structure decided on. In this case, I would say that versioning can be a problem.

11. *Are there use cases or situations where open-source may not be the ideal approach to solving a technical problem?* Sure, look at that Solaris case I mention in question 10. There are all kinds of good reasons (or to restate reasons upon which well-meaning smart people come to the conclusion of) for having a certain technology stack. Because of these reasons, a technology approach is often dictated. I am pretty sure that the department of defense has all kinds of oversight, protocols, network availability and the like which dictates a solution which isn't conducive to open source. This may not be bad, but it is just the way it is. I think we need to look at problems, and our investment approach and try to figure out how to reduce new capital expense (CAPEX) and operating and maintenance (O&M) strategies. I personally believe these are likely to include open source, but it is not always the case. We need to be smart enough to choose our battles and recognize where our return is most effective. This being said, I will argue 100% of the time, that the return is always higher when we deal w/ back end systems rather than face lift. By this I mean, that putting a new design on a web site, doesn't give you nearly the return as beginning to re-architect the back end to smaller more effective more scalable solutions.

12. *As a user and implementer of open-source technologies, what attributes do you look for in an open-source project in order to decide to use it in the development of a government system?* The common attributes I look for in implementing technologies are a) size, b) ability to implement and c) who else is using it. Let me dive into these a little:

- Size - I tend to want to use utilities that are becoming smaller and smaller. I realize this is not



exactly the case for PostGIS/QGIS as these are not exactly small things. However, these things offer a size that is highly scalable. These are small enough and easy enough to implement on my own laptop, and tend to have the exact same implementations as my laptop as they do in the server environment. That notion transcends size. More and more I am looking for very small very discrete libraries, services or APIs which offer highly focused, easily integratable solutions for scalability. Leaflet is a pretty interesting example, as it is tiny and has a pretty good API. Mapbox seems to be knocking this one out of the park as they are releasing all kinds of solutions via their API and API infrastructure. The size of the library is critical to me, b/c I want the same implementation to test as I do to implement in an enterprise way. Open source can be very good at this, and in general proprietary solutions fail at this. Proprietary solutions tend to fail at this b/c they tend to want all things in their solution (it makes them more money). The internet has and in my humble opinion will continue to prove this idea wrong (one cannot wall in the internet).

- Ability to implement - Here I mean a combination of things. Is the install or registration very easy; do I have to procure services; what kind of license is there; is there good documentation; is the development path pretty straightforward? All of these questions are important, and all of them should be asked whether I am using open source or proprietary software. I am a fan of REST, so if there is a REST end point API, I am usually happy. If there is a soap end-point, forget it. I tend to like very easy registration, in particular no accounts or no API keys. Or if there are both, then they should be super easy to implement. We in government have a problem with some contract terms and conditions. While not software, it is worthwhile to talk about the 'share a-like' clauses in things like OpenStreetMap. At first it seems very cool, but in the end it also has very serious problems for government derivative works. Documentation for use and development is key; poor docs will get loose adoption. Finally, if my development path requires 3-5 layers of other services, you can bet I won't pick it up quickly. One of my favorite is the git API, because it allows all kinds of flexibility (check out bl.ocks.org if you are curious; uses the git API from open gists - thanks to Mike Bostock).
- Who else is using it - I follow a core group of people; Stamen Designs, Mapbox, Code for America, GreenInfo Network, New York Times (Mike Bostock). If I see these people using something, you can bet my interest is piqued. If I see something that looks and sounds like a campaign for use, you can bet I will be more hesitant.

13. *Based on your experience, what lessons can traditional organizational structures (such as corporations or governments) draw from successful open-source projects, especially in terms of governance and operations, in order to improve outcomes? How could these lessons potentially be*





*applied beyond technology in such organizations?* I gave this talk a while back on this topic and my number one conclusion was, the derivative is more important than the original work. This concept (the derivative concept) is what I think is the most important lesson traditional organizations can learn from successful open source projects. Here is why; the real return on investment (ROI) is a combination of all the things output from the project. In particular in developing the national broadband map, we had to develop 50 or so APIs. About 6 months prior to release, we released one of these, a look up on census blocks; I pass it a latitude/longitude it gives back the block that contains the latitude/longitude. To this day it is the most utilized API in the FCC (FWIW APIs at the NBM have had some 700 million requests - we are pretty proud of that). This was a derivative API; it doesn't have anything to do with broadband. It does deliver a bit of data that is required in looking up broadband information stored by census block, but this derivative has more use than the structure for broadband data. These are the kinds of solutions, these derivatives, which provide crazy huge additions to the ROI of the investment. If I had to do NBM again, I doubt we would have changed much to this approach. The real lesson here, is be open to gaining value where you don't expect it; the derivative is more valuable than the original work.

Lessons can be applied really ubiquitously. These are really not technology lessons, but relatively universal in my mind. I think the key thing is decision makers and leaders in government need to be aware, that the big thing the internet has done for us is provide us with a non-walled garden, where everyone can produce and consume. In this world, there are great gains to be made by being open, and great costs to be borne by trying to establish walled gardens inside the internet.

