# Sharing community data

I've been thinking lately about how open data and community data fit together. Actually, I've been thinking about it for a long time - since we launched OpenCrisis to try to get more data tools and ideas into the hands of crisismappers, and started the long work of trying to archive and share as much mapping data and information as we could. Here's some first thoughts on restarting this work.

Some of the big questions that came up then, and continue to come up, are about ownership, risk and responsibility. For instance:

- Ownership. If a community of people add reports to a site, and that site also sucks in data from social media, who owns that data? That question's already been asked and answered in many open data and social media sites, often involving much work and lost data as licenses are transferred (see OpenStreetMap's license moves, for example). Having your mappers sign a contributor agreement, and having a data license statement on any dataset you put out is a good start here.
- Risk. Is something we've also always dealt with. The types of information that mappers handle mean you need to do data risk analysis that covers population, mappers, organizations and leads.
- Responsibility. If you make a dataset public, you have a responsibility to, to the best of your knowledge, skills and advice you can obtain, do no harm to the people connected to that dataset. That's a big responsibility, and one that has to do a balancing act between making data available to people who can do good with it and protecting the data subjects, sources and managers.

This week, I used some Python code to share all the geolocations (latitude, longitude and location name) from 2013 Ushahidi instances for Philippines typhoons. That was a pretty clear-cut example: it wasn't a conflict situation, none of the locations were sensitive (e.g. personal homes or addresses of places like rape crisis centers that need to be kept secret), and the data was gleaned from public sites created by large NGOs and open data teams. Life isn't always that clear-cut.

## Datasets: where and what

Some of the places that mappers hold data are:

- Google spreadsheets (tables of information),
- Skypechats (informal information transfers) and googledocs/emails (e.g. annotated highlights from a day's Skype discussions),
- OpenStreetMap
- Micromappers data (often visible as google spreadsheets) and
- Ushahidi instances (unfortunately afaik, there weren't any Ushahidi instances updated for

Typhoon Ruby, so I couldn't compare the two sets of data).

Some of the data collected by those mappers includes:

- Geolocation: Latitude and longitude for each location name in a list. These are usually either a) found automatically by the platform using a gazetteer like Nominatim, b) input by site administrators, or c) input by the person submitting a direct report.
- Direct reports: Messages sent by reporters or general public via SMS or web form. These generally have the standard set of Ushahidi report entries (title, description, category list etc), but might also include custom form fields to capture items of specific interest to the map owner.
- Indirect reports: Messages scraped from Twitter, Facebook etc.
- Category lists: The categories that reports are tagged with; these lists are usually created by the site administrator.
- API data: data input into a platform by another system, using the platform's API. This includes sensor data, and datasets scraped and aggregated from another platform instance.
- Media: Images, video, audio files added to a report by the reporter or an administrator.

Not all of this data is owned by the site owner. For example, third party data (e.g. Twitter messages) has restrictions on storage and ownership that even with the original sender's permission could make it illegal for you to distribute or keep on your site.

## Who and why?

Open data is, by its nature, open, and it's difficult to anticipate all the uses that people have for a dataset you release. Some examples from experience are:

- Academics - analysis of social media use, group dynamics etc.
- People in your organization - for lessons learned reports, for illustration, for visualizations, for analysis of things like report tempo (how frequently did information arrive for processing, translation etc)
- Data subjects - to check veracity of data about them, and to ask for data removal (sometimes under laws designed for this purpose, e.g. EU privacy laws). I haven't seen this happen yet, but I really really want it to.

If you release a dataset to anyone, you have to assume a risk that that dataset will make its way into the public domain. We've seen too many instances of datasets that should have been kept private making it into the public domain (and, to be fair, also instances of datasets that should have become public, and datasets that have been carefully pruned being criticized for release too). Always have the possibility of accidental release in mind when you assess the risks of opening up data.

## How?

Sharing data shouldn't just be about clicking on a "share" button. There are processes to think about, and risk analysis to do:

- Ethical process: Assessing the potential risks in sharing a dataset; selecting which data you should and should not share. Alway think what the potential harms from sharing information from the deployment is, versus the potential good. If you're not sure, don't share, but if you've checked, cleaned, double-checked and the risk is minimal (and ethical: you're working with other people's information here), seriously consider it. If it's come from a personal source (SMS, email etc), check it. At least twice. I generally do a manual investigation by someone who already has access to the deployment dataset first, with them weeding out all the obvious PII and worrisome data points, then ask someone local to the deployment area to do a manual investigation for problems that aren't obvious to people outside the area (see under: Homs bakery locations).
- Legal process: choosing who to share with, writing nondisclosure agreements, academic ethics agreements etc. You might want to share data that's been in the media because it's already out there, but you could find yourself in interesting legal territory if you do (see under: GDELT). In some countries, slander and libel laws could also be a consideration.
- Physical process: where to put cleaned data, how to make it available. There are many "data warehouses" which specialise in hosting datasets. Data warehouses include the Humanitarian Data Exchange (HDX), which specialises in disaster-related data. You can also share data by making it public on an Ushahidi instance (e.g. crowdmap), or by making data available to people on request. See crowdmap.com's api and csv public download button.

Some of the things I look for on a first pass include:

- Identification of reports and subjects: Phone numbers, email addresses, names, personal addresses
- Military information: actions, activities, equipment
- Uncorroborated crime reports: violence, corruption etc that aren't also supported by local media reports
- Inflammatory statements (these might re-ignite local tensions)
- Veracity: Are these reports true - or at least, are they supported by external information.

Things that will mess up your day doing this include untranslated sections of text (you'll need a native speaker or good auto translate software), codes (e.g. what does "41" mean as a message?) and the amount of time it takes to check through every report by hand. But if you don't do these things, you're not doing due diligence on your data, and that needs to happen.

Other questions that you might want to ask (and that could make your checking easier) include:

- How geographically accurate does your data release have to be?  E.g. would it be okay/ better to release data at a lower geographical precision (e.g. to town level rather than street)?
- Do you need to release every report?  Most deployments have a lot of junk messages (usually tagged unverified) - remember, the smaller amount of data you release, the less you have to manage (and worry about).
- Would aggregated data match the need of the person asking for data?  e.g. if they just need dates, locations and categories, do you need to release text too?
- Time. You might want to leave time after your deployment for the dataset to be less potentially damaging. When you release a dataset, you should also have a "data retirement" plan in place, detailing whether there's a last date that you want that data to be available, and a process to archive it and any associated comments on it.

## Further reading

There's a worldwide discussion going on right now about how private individuals, groups and companies can release open data, and on the issues and considerations needed to do that. More places to look include:

- Responsibledata.io page on responsible development data
- Responsibledata.io resource page
- Ushahidi Data Cleaning Guidelines