

September 19, 2017

From: Chris Jerdonek, OSVTAC Chair

To: Open Source Voting System Technical Advisory Committee (OSVTAC)

RE: Agenda Item #8 – Proposed text re: Committee Recommendations

3. Assumptions

This section lists certain assumptions the committee has made while drafting this document.

- The Department of Elections does not have the expertise to conduct the day-to-day management of the development and certification of an open source voting system.
- The Department of Elections has expressed a preference [link to relevant Director's report] for the GNU General Public License version 3 (GPLv3). This is consistent with the copyleft preference stated in the Elections Commission's Open Source Voting Systems Resolution.

...

5. Recommendations

...

5.2. Requirements gathering

This section contains recommendations ~~about~~ related to gathering requirements. For committee recommendations in relation to of specific requirements, see the Requirements section below.

5.2.1. Key Decisions

The following are some key decisions about requirements that should be made early in the process of designing and developing the voting system.

5.2.1.1. Pre-printed versus on-demand ballots, including how selections are marked

For in-person voting, the question of pre-printed ballots versus on-demand ballots, combined with how ballots are marked (for both accessible voting and not-necessarily-accessible voting) will greatly affect what type of precinct hardware needs to be developed. It also greatly affects how many units would need to be purchased and deployed per precinct.

This decision needs to be made separately for accessible voting and not-necessarily-accessible voting. However, the decisions for the two scenarios are not independent. They are related.

For not-necessarily-accessible voting, options include—

1. Pre-printed ballots with selections marked by hand
2. On-demand ballots printed without selections and marked by hand
3. On-demand ballots printed together with selections using an accessible device

For accessible voting, options include—

1. Pre-printed ballots marked using an accessible device (e.g. by inserting the ballot)
2. On-demand ballots printed without selections and marked using an accessible device
3. On-demand ballots printed together with selections using an accessible device

Some considerations include—

1. The more that the accessible and not-necessarily-accessible scenarios are similar to one another, the more consistent the voter experience will be. The most similar would be if both scenarios are conducted with option (3), “on-demand ballots printed together with selections using an accessible device.” Different but still similar would be if both groups use pre-printed ballots or on-demand ballots printed without selections, with the only difference being how the ballot is marked (by hand versus using an accessible device). The least similar would be, for example, option (1) for not-necessarily-accessible voting and option (3) for accessible voting. The latter happens to be how San Francisco conducts its elections today.
2. To preserve ballot secrecy during the count, it is preferable if the voted ballots “look” the same across the accessible and not-necessarily-accessible methods. An example of the ballots looking different would be if accessible voting results in voted ballots that contain only the voters’ selections and not other ballot choices, whereas the not-necessarily-accessible approach results in voted ballots containing all ballot choices but with the voters’ selections marked.
3. Requiring ballots to be printed on-demand for all voters (either with or without selections) would require using a printer for every voter in the polling place. This would likely require more electronic devices at each polling place, which in turn would increase costs, complexity, and the possibility of something breaking or going wrong. These printing requirements would be even greater for the case of printing not just blank ballots for all voters, but ballots *with their selections* for all voters. This is because voters would likely need to be occupying a machine while they are making their selections.
4. Using pre-printed ballots allows voters without disabilities to vote using the “low-tech” solution of only using a marker or pen (with the exception of the precinct ballot scanner that normally scans and counts the ballot). This would reduce the polling place’s overall

dependency on technology and possible things that can go wrong (e.g. power outages, one or more machines breaking, etc.).

5. Using pre-printed ballots results in increased paper usage and printing costs, since the Department needs to prepare extras of every ballot type (including every language, party preference, and combination thereof).
6. Printing ballots on-demand would theoretically allow voters to get the correct ballot type even if they go to the wrong precinct. Currently, a voter going to the wrong precinct can only choose among the ballot types pre-printed and made available at that precinct.
7. If ballots are printed on-demand, poll workers would not have to keep track of all the different ballot types (e.g. different languages, the various party ballots, etc.). It would instead automatically be taken care of by the ballot printer.
8. If the accessible device is a ballot-marking device, the device will be harder to use because each ballot card would need to be inserted individually into the device. Conversely, if the accessible device prints the ballot with selections, fewer physical cards would be required.

5.2.1.2. Printing unique identifiers on ballots at scan-time

One key decision is whether a unique identifier should be printed on every ballot while it is being scanned.

Pros:

- This would permit more sophisticated auditing approaches that involve selecting individual ballots at random, which could reduce time and costs (e.g. risk-limiting audits). Without this feature, auditing needs to be done in larger “batches,” or ballots need to be kept in careful order to allow accessing individual ballots.

Cons:

- It is not clear if COTS scanners support the feature of printing while scanning.
- The scanner hardware would become more complicated since there would be another “moving part” that can break.

5.2.1.3. End-to-end verifiability

It should be determined how much additional work would need to be done to make the voting process end-to-end verifiable, and whether and which designs are more compatible (e.g. among approaches listed in section 5.2.1.1. “pre-printed versus on-demand ballots”). Also, is this something that could be incorporated later on in the process, or does it need to be incorporated from the beginning?

...

5.5. Open Source

This section covers topics related to open source.

- Each software component being developed should be licensed under an OSI-approved software license (see also the Assumptions section).
- The development of the software should be done in public from the first day of development. All software development should occur in public (e.g. on GitHub), rather than, for example, waiting for the software to reach a certain level of completion before becoming public.
- All software should be licensed under an OSI approved software license from the first day of development. All software being developed in public should have an open source license when development first starts, rather than, for example, adding a license file later on. This would eliminate any confusion and uncertainty from members of the public as to whether the software will really be open source. This would encourage members of the public to start contributing to the project as early as possible.
- All software being developed should be developed using an open-source software language. For software languages, any OSI-approved license should be okay. The programming language itself need not be copyleft.

...

5.7. Software architecture and design

- When defining software components to develop, favor designs that promote reusing components. For example, a software library that can read a ballot image and return the marked “votes” (what we are calling a “ballot image interpreter” component) can be used in both precinct scanners and central scanners (as well as software applications for adjudication or auditing). Favoring component reuse can mean having less code to write and test, which in turn can reduce required time and costs.

5.8. Software development

- The project should not depend on volunteers for the successful completion or security of the project. However, useful volunteer contributions should be encouraged and not turned away.

[Note: the bullet point above is partly in response to the following quote in the Sept. 3 SF Chronicle article, “And Jack Miller, the chief information security officer at SlashNext, a cybersecurity firm, questioned whether an open-source voting project could attract enough skilled volunteers to police the software code effectively.”]

5.12. Testing

- Datasets of real election data (e.g. a couple past elections in San Francisco of different types) should be compiled in a structured format for product prototyping and testing. This includes not just vote totals but also candidate and contest data. This will help in establishing requirements and designing the system.