# Introducing a few firsts

- FIRST: Open Source project sponsored by CCSF/DT
- FIRST: Implementation of a Risk-Limiting Audit on a Ranked-Choice Voting contest
- FIRST: Independent validation of Dominion's RCV Tabulation

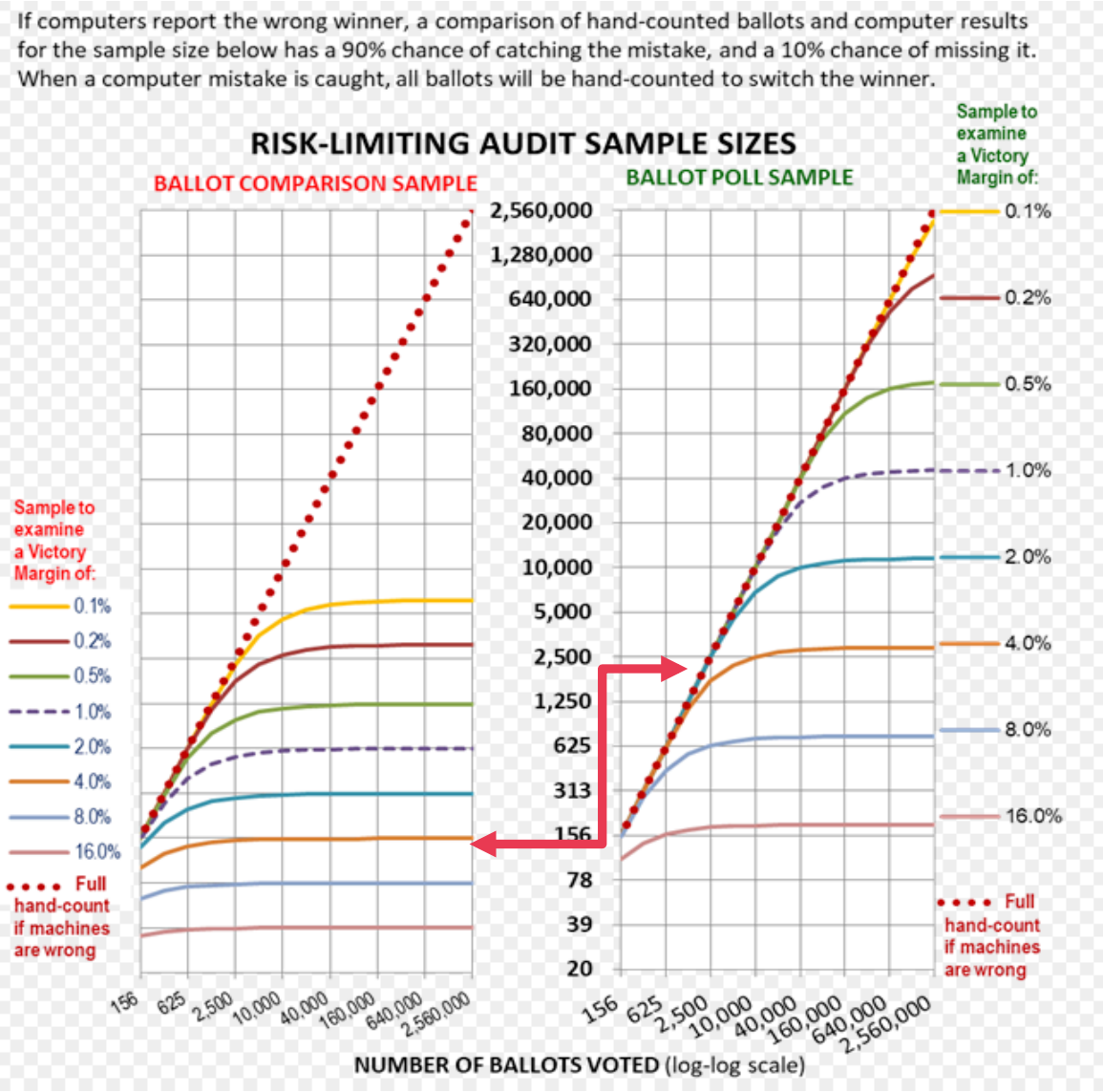
# What is a Risk-Limiting Audit?

- A Risk-Limiting Audit (RLA) offers a statistical guarantee:

  *"If a full manual tally of the paper ballots would show that the reported election outcome is wrong, an RLA has a known minimum chance, the RLA limit, of leading to a full manual tally"* – Philip B. Stark

  *"As with other elections audit, the goal is to identify not only intentional alterations of ballots and tallies, but also bugs in election machines, such as software errors, scanners with blocked sensors or scanners skipping some ballots. "* – Wikipedia

# What is a Ballot-Comparison audit ?




Two main types of RLA:

- **Ballot Polling**: Humans count a random sample of ballots and report any difference between manual percentage and computer percentage
- **Ballot Comparison**: Election system provide a Cast Vote Record(CVR). Humans check physical ballots in a random sample of ballots. RLA system check results of human count against machine count.
  - Ballot comparison is more efficient than ballot polling due to its smaller sample size, and, arguably, less error-prone on an RCV.

# Anyone is welcome to check the math

audit-irv($\mathcal{C}$, $\mathcal{B}$, $c_w$, $\alpha[, \gamma]$)

```
audits ← ∅
F ← ∅ ▷ F is a set sequences to expand (the frontier)
LB ← 0
▷ Populate F with single-candidate sequences
for each(c ∈ C \ {c_w}):
    π ← [c]
    h ← FindBestAudit(π, C, B, α[, γ])
    hy[π] ← h ▷ Record best hypothesis for π
    ba[π] ← π ▷ Record best ancestor sequence for π
    F ← F ∪ {π}
▷ Repeatedly expand the sequence with largest ASN in F
while(|F| > 0):
    π ← argmax{ASN(hy[π]) | π ∈ F}
    F ← F \ {π}
    if(ASN(hy[ba[π]]) ≤ LB):
        audits ← audits ∪ {hy[ba[π]]}
        F ← F \ {π' ∈ F | ba[π] is a suffix of π'}
        continue
    for each(c ∈ C \ π):
        π' ← [c] ++π
        h ← FindBestAudit(π', C, B, α[, γ])
        hy[π'] ← h
        ba[π'] ← if ASN(h) < ASN(hy[ba[π]]) then π' else ba[π]
        if(|π'| = |C|):
            if(ASN(hy[ba[π']]) = ∞):
                terminate algorithm, full recount necessary
            else:
                audits ← audits ∪ {hy[ba[π']]}
                LB ← max(LB, ASN(hy[ba[π']]))
                F ← F \ {π' ∈ F | ba[π] is a suffix of π'}
                continue
        else:
            F ← F ∪ {π'}
return audits with maximum ASN equal to LB
```

# Overview

- SHANGRLA is an Open Source RCV-RLA project sponsored by CCSF
  - https://github.com/pbstark/SHANGRLA
  - Implementation of academic research on RCV-RLA
  - Six key contributors across the globe on the project
    - We have yet to meet each other and we all spoke different languages:
      - *English, American, Australian, Java, C++, Python, JSON, CSV, and Statistics*
- SHANGRLA pilot is NOT an RLA
  - It is a ballot-comparison audit of Vote By Mail ballots (~68% of total)
  - A full RLA would require inclusion of all ballots casted
    - And a separate audit method for non VBM Ballots

# Acknowledgements - CCSF -

CCSF would like to acknowledge the gargantuan efforts of the team:

CCSF Team:

Linda J. Gerull CCSF, City CIO & Executive Director, Dept of Technologies

Members of the San Francisco Dept of Elections

RCV Team:

Dr. Michelle Blom:

Research Fellow, School of Computing and Information Systems, The University of Melbourne, Australia

Dr. Andrew Conway:

CEO, Silicon Econometrics Pty. Ltd., Australia

Peter Stuckey:

Professor, Data Science & AI, Monash University, Melbourne, Australia

Vanessa Teague:

Associate Professor, School of Computing and Information Systems, The University of Melbourne, Australia

RLA Team:

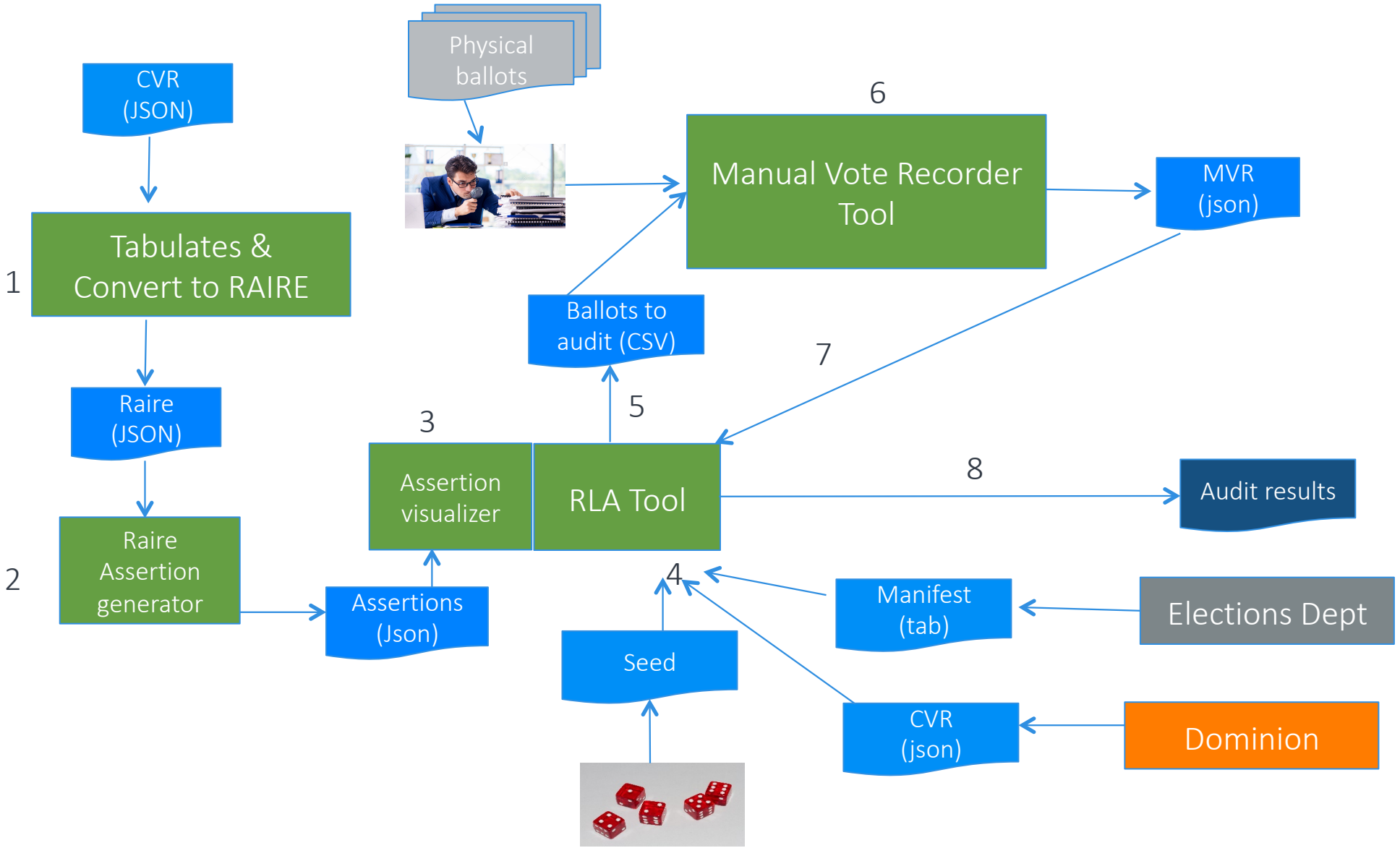
Dan King:

CCSF Dept of Technology Consultant, CEO ViewPoint Technology, San Diego

Philip B. Stark:

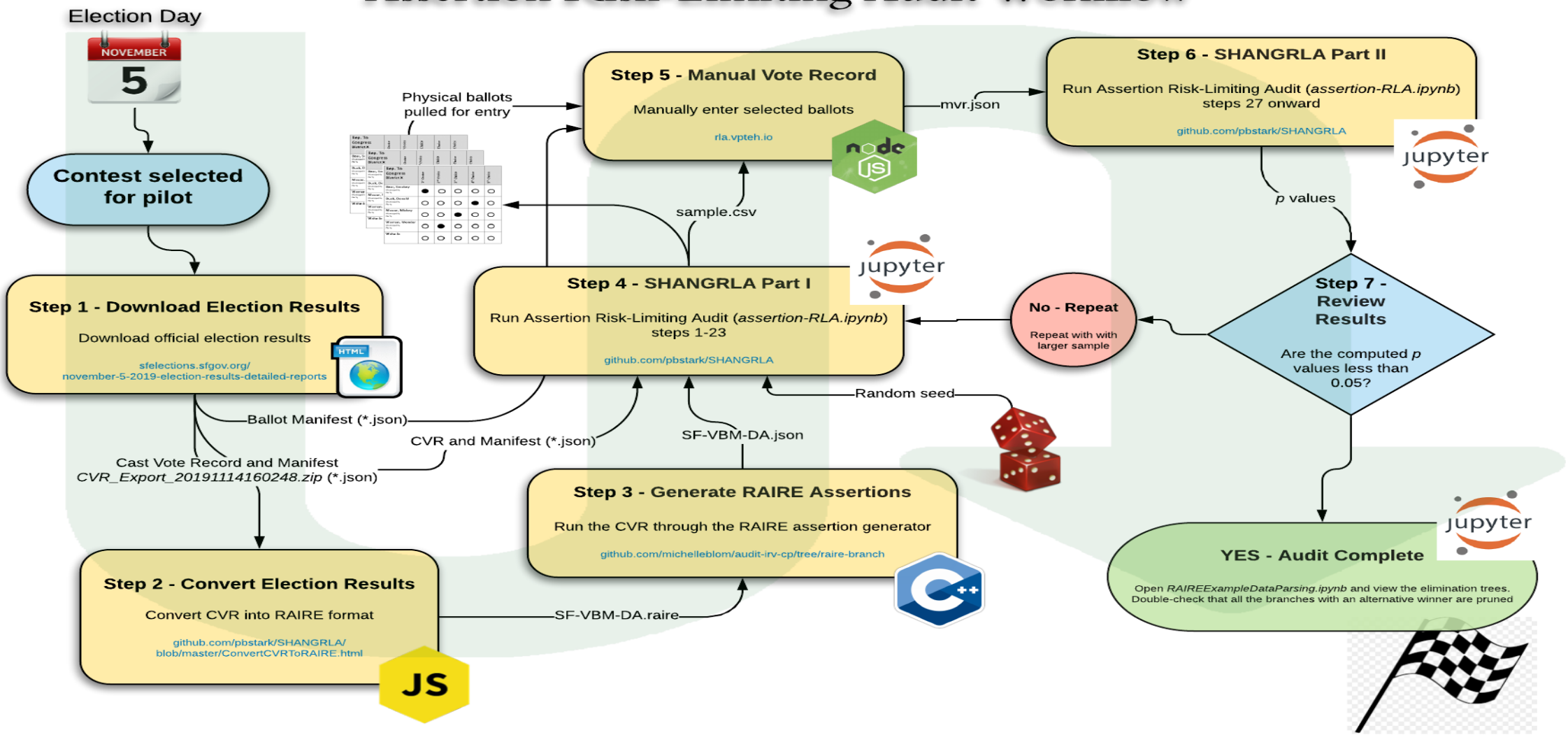
Professor of Statistics, Associate Dean, Division of Mathematical and Physical Sciences , Regional Associate Dean (Interim), College of Chemistry and Division of Mathematical and Physical Sciences, University of Berkeley, CA

# ShangRLA flow overview



# RLA Pilot – November 2019

## Assertion Risk-Limiting Audit Workflow



# RAIRE Tabulator/Converter – Andrew Conway - Aus

## Independent verification of Dominion's tabulation

### RAIRE Tabulator (Report 8)

| Report 8 | Round 1 | Round 2 | Round 3 |
|---|---|---|---|
| Loftus | 59,762 | 66,228 | 83,511 |
| Dautsch | 26,901 | | |
| Tung | 37,161 | 46,403 | |
| Boudin | 68,145 | 72,860 | 85,950 |

### DOE Report (Report8)

City and County of San Francisco

| Candidate | Round 1 | | | Round 2 | | | Round 3 | | |
|---|---|---|---|---|---|---|---|---|---|
| | Votes | Percentage | Transfer (Elimination) | Votes | Percentage | Transfer (Elimination) | Votes | Percentage | Transfer |
| SUZY LOFTUS | 59,762 | 31.13% | 6,466 | 66,228 | 35.70% | 17,283 | 83,511 | 49.28% | 0 |
| LEIF DAUTCH | 26,901 | 14.01% | -26,901 | 0 | 0.00% | 0 | 0 | 0.00% | 0 |
| NANCY TUNG | 37,161 | 19.36% | 9,242 | 46,403 | 25.02% | -46,403 | 0 | 0.00% | 0 |
| CHESA BOUDIN | 68,145 | 35.50% | 4,715 | 72,860 | 39.28% | 13,090 | 85,950 | 50.72% | 0 |
| Continuing Ballots Total | 191,969 | | | 185,491 | | | 169,461 | | |

# Assertions Generator – Michelle Blom - Aus

## Many of the the 29 audits steps can be found here:

michelleblom

Follow

michelle.blom@gmail.com

Block or report user

Overview | Repositories 7 | Projects 1 | Stars 0 | Followers 0 | Following 0

Popular repositories

**audit-irv-cp**
Code for generating and running ballot-level comparison audits for IRV elections.
C++ ★ 2

**audit-irv-bp**
Code for generating and running ballot polling audits for IRV elections.
C++ ★ 1

**margin-irv**
Margin computation for IRV elections
C++

**NSW2015**
Data files for use with a range of code available in my repositories. Can be used with: audit-irv-bp; audit-irv-cp.

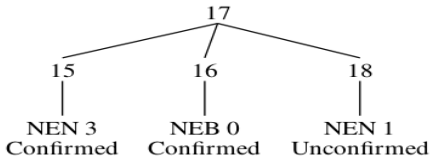
**STV-manipulator**
An implementation of two heuristics for computing candidate manipulations of Single Transferable Vote (STV) elections. These manipulations provide an upper bound on the margin of victory for such e...
C++

**AZUL**
Framework to support policy learning for the boardgame AZUL. The purpose of this framework is to allow students to implement algorithms for learning AI players for the game and evaluate the perform...
Python

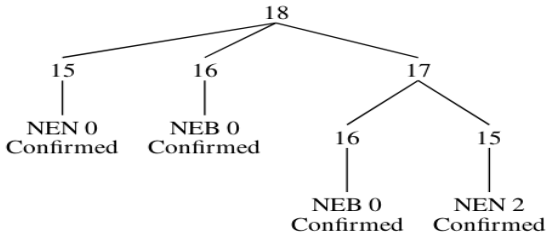
# Assertion Visualizer - Vanessa Teague – Aus.

[2]:


*Pruned tree in which 16-LEIF DAUTCH wins.*




*Pruned tree in which 17-NANCY TUNG wins.*




*Pruned tree in which 18-CHESA BOUDIN wins.*

*Trees showing how other winners are excluded.*

Now print all the assertions. This gives you an explanation of the meaning of each one.

[3]:

```
printAssertions(WOLosers,IRVElims)
```

```
Not-Eliminated-Before assertions:
Confirmed:   NEB  0: Candidate 15 cannot be eliminated before 16.


Not-Eliminated-Next assertions:
Confirmed:   NEN  0: Candidate 15 cannot be eliminated next when {'16', '17'} are eliminated.
Unconfirmed: NEN  1: Candidate 18 cannot be eliminated next when {'16', '15'} are eliminated.
Confirmed:   NEN  2: Candidate 15 cannot be eliminated next when {'16'} are eliminated.
Confirmed:   NEN  3: Candidate 15 cannot be eliminated next when {'16', '18'} are eliminated.
```

# RLA – Philip B. Stark – Berkeley

**Philip B. Stark**
pbstark

Follow

Faculty at UC Berkeley / Consultant
United States
stark@risklaw.org
www.stat.berkeley.edu/~stark

PRO

Block or report user

Overview | Repositories 22 | Projects 0 | Stars 0 | Followers 50 | Following 0

Popular repositories

**Padova15**
Materials for a 30-hour course in Statistics for Engineers, given at University of Padova
CSS ★13 2

**MX14**
Short course on nonparametric inference in auditing and litigation, XXIX Foro Internacional de Estadistica, Puebla, MX
Jupyter Notebook ★12 5

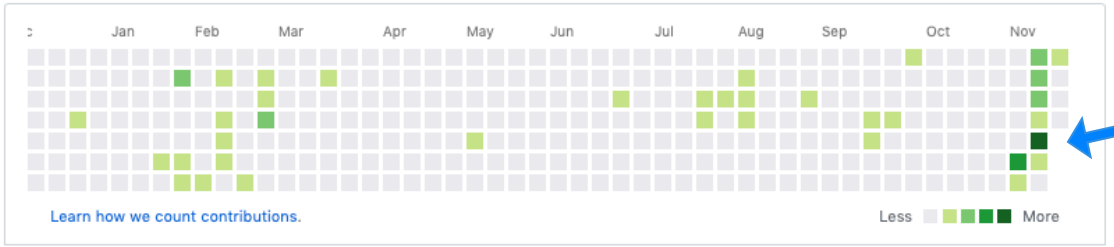
**S157F17**
Statistics 157, Fall 2017, UC Berkeley
Jupyter Notebook ★10 10

**Nonpar**
Teaching materials for nonparametric statistics
CSS ★7 3

**pseudorandom**
Jupyter Notebook ★5

**CORLA18**
theory and code for RLAs: Colorado and San Francisco
TeX ★3 4

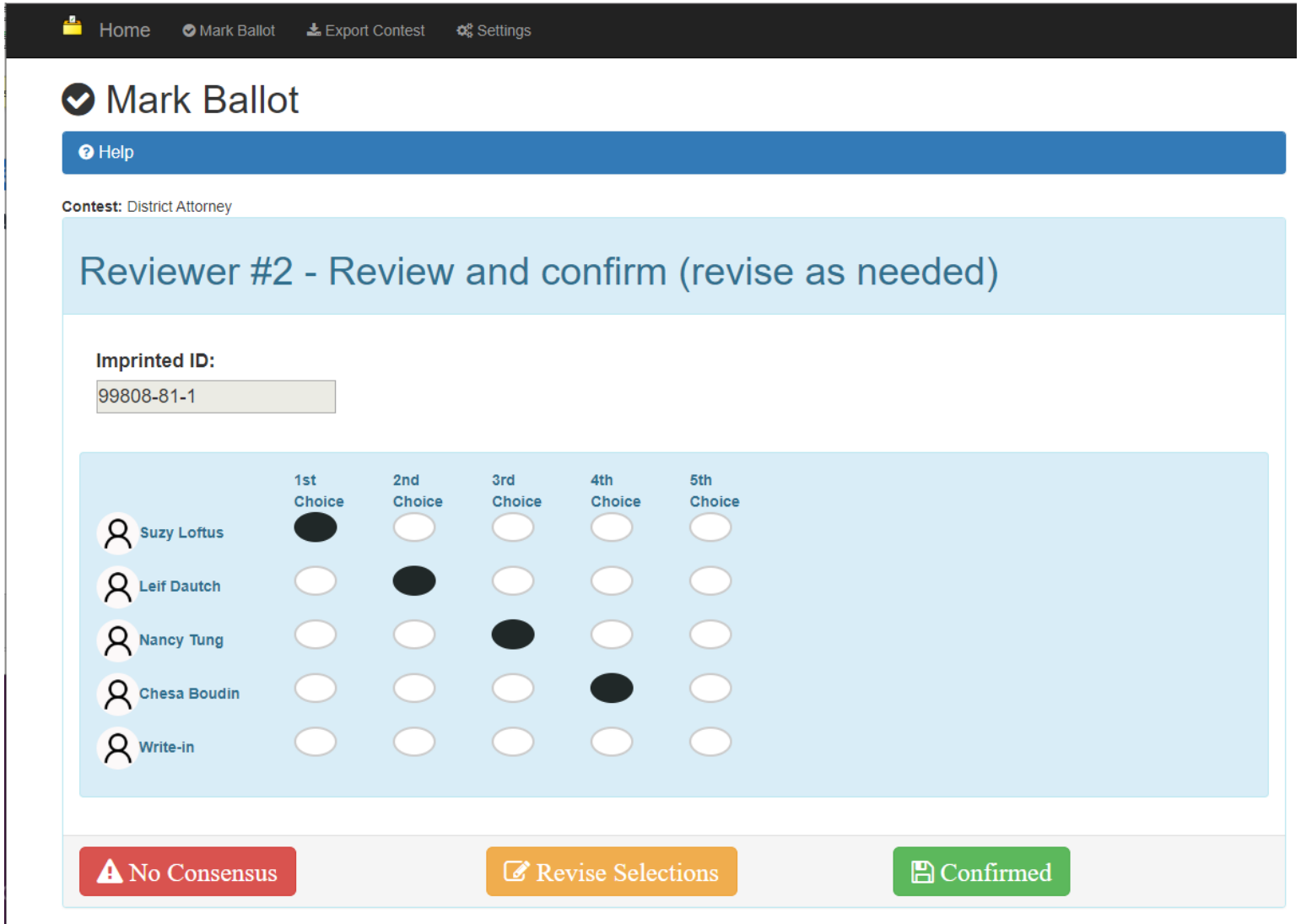
95 contributions in the last year




- Inventor of RLA
- Has participated in dozens of RLA
- His work is the basis for CORLA
- ShangRLA is a substantial update to CORLA's methods

Was "busy" in November

# Auditor Tool - Dan King - CCSF/San Diego



# Next Step: Taking ShangRLA from pilot to product

- Phase I:
  - Standardize on languages
    - Transition out of Jupyter notebook
  - Migrate from files to an RDBMS
    - JSON is ill-suited for a system that has a natural entity-relationship model
  - Build a test suite above and beyond unit tests
  - Document
- Phase II:
  - Support for Multi-Contest auditing
  - Integrate non-VBM Ballot auditing
  - Enhance the UI
  - ShangRLA is engineered to support various forms of contest beyond RCV
    - …but “official support” may require further development and testing