New York State Board of Elections
Voting System Verification Testing

Clear Ballot ClearCount 2.2.5
Test Report v4.0

May 3, 2023

Document Number: NYS-22023-SCRTR-01
Prepared for:

Client Name	*New York State Board of Elections (NYSBOE)*
Address	*40 North Pearl St.* *Albany, New York 12207*

Prepared by:

SLI Compliance
4720 Independence St.
Wheat Ridge, CO 80033
(303) 422-1566
www.SLICompliance.com

SLI Compliance®, a Division of Gaming Laboratories International

Revision History

Date	Version	Author	Revision Summary
2/22/2023	1.0	M. Santos	Initial Release
3/5/2023	2.0	M. Santos	Updated for new build
4/28/2023	3.0	M. Santos	Updated for new build
5/3/2023	4.0	M. Santos	Updated for NYSTEC comments

DISCLAIMER

The test results reported herein must not be used by the client to claim product certification, approval, or endorsement by NVLAP, NIST, or any agency of the Federal Government. Results herein relate only to the items tested.

All testing conducted for this engagement has been done outside of the U.S. Election Assistance Commission's (EAC) Test and Certification Program. In no way does this test report represent an EAC certification against the Voluntary Voting System Guidelines (VVSG) or any other standard.

Trademarks: All products and company names are used for identification purposes only and may be trademarks of their respective owners.

The tests referenced in this document were performed in a controlled environment using specific systems and data sets, and results are related to the specific items tested. Actual results in other environments may vary.

Opinions and Interpretations

There are no opinions or interpretations included in this report.

Contents

Introduction

This Test Report details testing SLI Compliance performed to evaluate the **Clear Ballot ClearCount 2.2** system against an issue found dealing with oval position calculations being incorrect when the bottom of the ballot has more timing marks than the top of the ballot.

ClearCount software will make adjustments to the position of oval targets to account for ballot skew caused by a paper ballot that is not fed perfectly straight into the scanner. In this case, the ovals on the ballot may not be aligned based on their pixel coordinates (a line drawn across the ballot at a particular y-coordinate position on the image may pass through some ovals but not all the ovals in that row). Because a ballot may skew side-to side as it is pulled through the scanner, the amount by which the position of the oval target is adjusted (the offset) is not calculated based on the overall angle of the ballot from leading edge to trailing edge, but rather at each point along the length of the ballot where ovals reside. Due to an error in calculating the offset under certain conditions, the oval targets were incorrect for some ovals on some Dominion ballot formats.
The issue resided in the calculation of the offset due to ballot skew of the oval position. The original code assumed the marks to be used were the marks on the top of the ballot. However, in the case of Dominion ballots where there are, sometimes, more timing marks on the bottom (foot of the document) as compared to the top marks (head of the ballot), the bottom marks are used. The value used for the slope of the ballot is based on the slope of the side timing marks from the marks to be used location (top or bottom) and the current row. This calculation is used to account for the change in skew over the length of the ballot.
The issue may be more pronounced with longer ballots, as longer ballots may skew a bit more as they are pulled through a scanner.
Stacks of ballots that are lined up very straight (for example by using a paper jogger) with the paper guides on the scanner tightly in place would be less likely to become subject to skew.

The modification implemented correctly determines whether the top or the bottom has more timing marks and utilizes whichever has the most in the skew calculation.

1.1 References

The following documents were used in conducting this test campaign:

1. SLI VSTL Quality System Manual, v 3.1, June 28, 2019.

1.2 Attachment

- Attachment A - SW-13321 RCA + Corrective Action CC 2.2 Confidential

1.3 Project Overview

This Test Report describes the approach SLI implemented to perform verification testing of the **Clear Ballot ClearCount 2.2.5** system. This section lists the tasks required for this testing effort in more detail.

1.3.1 Project Tasks

The following tasks were executed to complete this test effort:

- The system was set up and configured with the NYSBOE ClearCount 2.2 software in accordance with the system setup documentation in Clear Ballot's TDP.
- Functional test cases were designed to address the modification made to the system.

1.3.2 Owner Assignments

- Source Code analysis was conducted by the Source Code Review (SCR) team, Security and Voting Test Engineers, with oversight by the Test Manager (TM)
- Test module development and validation was conducted by Voting Test Engineer personnel, with oversight provided by the TM
- Formal test execution was conducted by Voting Test Engineer personnel, with oversight by the TM

1.4 Modifications

The modifications that have been made to the Clear Ballot ClearCount 2.2 system can be found in the "Attachment A - SW-13321 RCA + Corrective Action CC 2.2 **Confidential**".

1.5 System Identification

This section contains a description of the Clear Ballot ClearCount 2.2.5 system specific components that were tested in this test campaign.

1.6.1 Equipment Configurations Utilized

SLI Compliance utilized the configuration:

- **ClearCount Configuration:**
  - 2 Scanning Stations
    - Latitude 5521 (scanning station)
    - Fujitsu fi-7900 (scanner)
    - Latitude 5590 (scanning station)
    - Fujitsu fi-6800 (scanner)
  - 1 Count Station
    - Optiplex XE3(Tower and Monitor combo serves as the CountStation)
  - 1 Count Server
    - Dell PowerEdge T130 (server)
  - Cisco 8 Port Switch
  - Brother Printer – HL-L2350DW

2 Materials Required for Testing

All materials were provided by Clear Ballot to SLI to facilitate testing of the system. This section outlines the required materials.

2.1 Software and Firmware

The following software and firmware were required for the execution of formal tests. This included all supporting software such as operating systems, compilers, assemblers, application software, firmware, and transmission of data.

2.1.1 Software and Firmware Information

ClearCount v2.2.5 Servers

- Ubuntu 18.04.5 LTS
- mysql 5.7.35-0

ClearCount v2.2.5 ScanStations

- Windows 10 Pro
- PaperStream Capture – 2.10.0.9

ClearCount v2.2.5 CountStations

- Windows 10 Pro
- Google Chrome – 87.0.4280.141
  (Note: version updated to correct error in ClearCount 2.2 test plan, where Google Chrome version 78.0.3904.108 was listed)

ClearCount Printers

- Brother HL-L2350DW – 1.3.0.0

2.1.2 Artifacts Provided by Clear Ballot

Pertinent artifacts needed for verification of the Clear Ballot modification were provided by Clear Ballot. This included four ballots sets of three different sizes:

-17 inches, front and back used, in landscape. (Note: This was the election where the original issue was seen) (Tioga County, NY)
-22 inches in Landscape Mode, front and back. General (Orange County, NY)
-14 inches, front and back in portrait mode, ovals as far right as possible. General. (New Port City, VT)
-14 inches, font and back in portrait mode with ovals as far left as possible. General. (Leon County, FL)

3 Test Operations Procedures

3.1 Facility Requirements

Testing was performed on-site at SLI in Colorado.

Secure labs are available with appropriate power supply and space to accommodate the various configurations defined within this Test Report. Temperature/humidity gauges were employed to determine that the appropriate conditions exist during testing.

Unless otherwise specified herein, all remaining tests, including system level functional testing, shall be performed at standard ambient conditions:

- Temperature: 64°F - 79°F (17.7°C - 26.1°C)
- Relative Humidity: 20 to 90%

Atmospheric Pressure: Local Site Pressure

4 Test Operations Procedures

4.1 Source Code Review

Source code was modified in the “tabulate/BaseBallotRegistration.py” module.

The old code was:

```
if self.slopes_dy_dx[row] is None:
        self.slopes_dy_dx[row] = \
                BaseBallotRegistration.calculate_horizontal_slope(
                        self.longest_horiz_real_marks[:row])
ya = (y + int((x - vert_marks[-1 - col].center.x) *
                        self.slopes_dy_dx[row]))
```

The new code was modified to:

```
if self.slopes_dy_dx[row] is None:
        # Calculate the slope from the vert_marks to the current row
        # Handle case when top_marks are used as the vert_marks
        if vert_marks == self.top_marks:
                self.slopes_dy_dx[row] = \
                        BaseBallotRegistration.calculate_horizontal_slope(
                                self.longest_horiz_real_marks[:row])
        # Handle case when bottom_marks are used as the vert_marks
        else:
                self.slopes_dy_dx[row] = \
                        BaseBallotRegistration.calculate_horizontal_slope(
                                self.longest_horiz_real_marks[row:])
ya = (y + int((x - vert_marks[-1 - col].center.x) *
                        self.slopes_dy_dx[row]))
```

The yellow highlighted lines indicate the newly introduced source code.
Note that the lines that start with a “#” symbol and are slightly greyed out, indicate comments that describe the action being performed, and are not functional source code.
The introduced code was found to be used to determine the appropriate set of marks to be used, either the top-of-page marks or the bottom-of-page marks.

4.2 Trusted Build

Once the code was reviewed and determined to be the expected update for resolution of the skewing issue, a Trusted Build was performed by SLI Compliance.
Source code from the SLI repository for the ClearCount 2.2.4 source code was used, incorporating the

newly updated "BaseBallotRegistration.py" module that had just been reviewed.

4.3 Functional Evaluation

SLI performed an evaluation of the Clear Ballot ClearCount 2.2.5 system against the modification implemented through the following steps:

1. The RCA was reviewed to determine exactly what the issue was and where it occurred in the ballot scanning process. In this case, the issue was found to be in the algorithm used to calculate how ballots are scanned and read by the ScanStation to determine oval positions, which would then be passed to ClearCount. If ballots were inserted at a sufficiently large angle it would lead to a miscalculation of oval detection and marking.
2. The old and new source code was reviewed for comparisons and corrections made
3. The original issue was reproduced using a 2.2.4 ScanStation/ClearCount configuration.
4. Inserting ballots into ClearCount that were heavily skewed to both the left and right, tolerance of the maximum amount a ballot can be angled when inserted were determined.
5. After the original issue was reproduced and maximum angle determined, the ClearCount configuration was upgraded from 2.2.4 to 2.2.5.
6. Using the parameters and data developed in the previous steps, a set of Test Cases were developed, utilizing ballots that were inserted into the ScanStation at an angle (both left and right), up to their max accepted angle.

   The ballots used included:

   a. 17-inch, Tioga County, NY landscape where the original issue was seen
   b. 22-inch, Orange County, NY landscape for maximum ballot size
   c. 14-inch, New Port City, VT portrait with ovals were on the right most position
   d. 14-inch, Leon County FL ballots where ovals were on the left most positions
7. Once the ballots were scanned at the ScanStation, the CVRs were brought into the ClearCount adjudication workflow where the adjudication markings were compared to the marked ballots and determined to be accurate to the expected outcome.

6 Conclusion

SLI has completed functional testing of the Clear Ballot ClearCount 2.2.5 system, examining the modification for correcting the implementation of skew adjustment for Dominion style ballots. With the modification implemented, the issue is resolved.

End of Test Report