# Test Report

**Report Number** *HRT-3026-TR-01*

# Verity Data for State of Colorado

## Phase II

### Test Report Release 1.0

July 27th, 2015

Prepared for:

| | |
|---|---|
| **Vendor Name** | *Hart InterCivic* |
| **Vendor System** | *Verity Data 1.3.3* |
| **Vendor Address** | 15500 Wells Port Drive<br>Austin, TX 78728 |

Prepared by:


216 16th St.
Suite 700
Denver, CO 80202
303-575-6881
www.SLIGlobalsolutions.com


NVLAP LAB CODE 200733-0

Accredited by the National Institute of Standards and Technology (NIST) National Voluntary Lab Accreditation Program (NVLAP), and accredited by the Election Assistance Commission (EAC) for VSTL status.

# Revision History

| Release | Author | Revisions |
|---|---|---|
| 1.0 | M. Santos | Initial Revision |

## 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.



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.

# TABLE OF CONTENTS

# 1 Introduction

SLI Global Solutions is submitting this report as a summary of the testing efforts for the Hart Verity Data State of Colorado Phase II testing. The purpose of this document is to provide an overview of the testing effort and the resultant findings for the **Verity Election Office** suite component, **Verity Data**. The review and testing was performed at SLI's Denver, Colorado facility.

SLI is a full service third party testing facility, founded in May 1996, from a software test-consulting firm. The specific system testing services offered include:

- Test Planning and Test Management
- eBusiness, Client-Server and Stand-alone Application Functional, Compatibility and Regression Testing
- eBusiness and Client-Server Load and Performance Testing
- Automated Regression Test Development, Consulting, Scripting and Execution
- Complex, Integrated Test Solutions and Automated Test Harnesses
- Independent Verification and Validation
- EAC approved and NIST NVLAP accredited Voting System Test Laboratory

## 1.1 References

1. SLI Quality System Manual, Revision v1.15, prepared by SLI, dated June 18th, 2013

## 1.2 Document Overview

This document contains:

- The Introduction which discusses the applications tested/reviewed
- The Test Background which discusses the testing process
- The System Identification which identifies hardware and software for the Hart Verity Data configuration
- The System Overview which discusses the functionality of Hart Verity Data management software
- The Testing Performed section which is a summary of the testing effort
- The Test Results summary section which contains the final analysis of the testing effort

# 2 Test Background

## 2.1 FCA - Functional & System Testing

SLI's standard test suites were customized for the Hart Verity Data State of Colorado Phase II configuration. Simulations were conducted to demonstrate an integrated business use case process for the Hart Verity Data configuration.

## 2.2 Terms and Abbreviations

The following terms and abbreviations will be used throughout this document:

**Table 1 – Terms and Abbreviations**

| Term | Abbreviation | Description |
|---|---|---|
| Commercial Off the Shelf | COTS | Commercial, readily available hardware devices (such as card readers, printers or personal computers) or software products (such as operating systems, programming language compilers, or database management systems) |

| Term | Abbreviation | Description |
|---|---|---|
| Election Management System | EMS | Typically a database management system used to collect jurisdiction information (district, precincts, languages, etc.) as well as election specific information (races, candidates, voter groups (parties), etc.). In addition, the EMS is also used to layout the ballots, download the election data to the voting devices, upload the results and produce the final results reports. |
| Independent Test Authority | ITA | This is a test lab that is not connected with the vendor or manufacturer of the voting system. |
| Chevron | No Abbreviation | **Verity** components use workflow chevrons. Workflow chevrons, arranged along the top of the screen, identify the function the user is currently viewing. |
| National Institute of Standards and Technology | NIST | A non-regulatory federal agency within the U.S. Dept. of Commerce. Its mission is to promote U.S. innovation and industrial competitiveness by advancing measurement science, standards, and technology in ways that enhance economic security and improve our quality of life. |
| National Voluntary Laboratory Accreditation Program | NVLAP | A division of NIST that provides third-party accreditation to testing and calibration laboratories. |
| Standard Lab Procedure | SLP | SLI's quality system documentation is made up of standard lab procedures (SLPs), which are procedures required to ensure a systematic, repeatable and accurate approach to voting systems testing and governing the actual performance of SLI's work. |
| Validation | No Abbreviation | Confirmation by examination and through provision of objective evidence that the requirements for a specific intended use or application have been fulfilled (ISO 9000) |

# 3 System Identification

The Hart Verity Data State of Colorado Phase II testing was submitted for testing with the hardware and software listed below. Other than the components listed below, no other Hart Verity product was reviewed in this test effort.

## 3.1 Software

Table 2 below details each application employed by the Hart Verity Data State of Colorado Phase II testing

**Table 2 – Hart Verity Voting 1.0 Software and Firmware**

| Application | Application type | Version |
|---|---|---|
| Verity Data | Data Management software | 1.3.3 |
| Verity Build | EMS software | 1.0.3 |
| Verity Desktop | EMS software | 1.0.3 |
| Verity User Manager | EMS software | 1.0.3 |
| Verity Election Manager | EMS software | 1.0.3 |

Table 3 below details the Commercial Off The Shelf software and firmware utilized within the **Verity Data** system.

**Table 3 – COTS Software/Firmware**

| Manufacturer | Application | Version | Verity Voting 1.0 Component |
|---|---|---|---|
| Microsoft | Windows 7, Service Pack 1 | 6.1.7601 | Data, Build |
| Microsoft | SQL Server | 6.1 | Data, Build |
| Adobe | Acrobat | 10.0 | Data, Build |

## 3.2 Equipment

The following equipment was required for the execution of the hardware, software and security tests. This includes system hardware, general purpose data processing and communications equipment, and any test instrumentation required.

### 3.2.1 COTS Equipment

The following Commercial Off-the-Shelf equipment was used in testing:

- Desktops

**Table 5 – COTS Equipment**

| Manufacturer | Hardware | Model |
|---|---|---|
| Various<br>(for **Verity Data and Build**) | **Intel-Windows Workstation (Minimum Requirements)**<br>Processor – Intel Celeron D 420 3.06GHz Dual Core<br>Memory – 2GB upgradable to 4GB<br>Hard Drive – 120 GB<br>Removable Storage – 8xDVD+/-RW Slim line<br>USB Ports – 4 ports<br>Video Card - Integrated Graphics<br>Keyboard - USB Keyboard<br>Mouse - USB Mouse | |
| Various<br>(for **Verity Data and Build**) | **Monitor (Minimum Requirements)**<br>Panel Size - 50.8 cm<br>Aspect Ratio - Widescreen (16:9)<br>Optimal Resolution - 1600 x 900 at 60 Hz<br>Contrast Ratio - 1000: 1<br>Brightness - 250 cd/m2 (typical) | |

# 4 System Overview

## 4.1 Analysis of Verity Data

An evaluation was performed on Verity Data and the requirements prescribed by the State of Colorado, for certification of use within the state.

Based off review of the Colorado Requirements Matrix, which consists of requirements from the 2002 Voting System Standards (VSS) as well as Colorado specific requirements, a list of requirements was compiled, as per the "Verity Data Colorado Phase II Test Plan v1.3". A listing of the requirements can be found in "Appendix A – Requirements Verified".

# 5 Testing Performed

## 5.1 Testing Executed

The configuration employed within the Hart Verity Data State of Colorado Phase II testing test suites included a Verity Data workstation and a Verity Build workstation.

The following Testing was performed for this project:

### 5.1.1 Trusted Build

A Trusted Build was performed, where SLI created the build environment from pristine components, integrated the Verity Data source code into the environment and then proceeded to compile the executable code.

The build process addressed requirements listed in items 62-81 of "Appendix A – Requirements Verified".

- *62. (EAC Program Manual, 5.5.1)*
- *63. (EAC Program Manual, 5.5.2)*
- *64. (EAC Program Manual, 5.5.3)*
- *65. (EAC Program Manual, 5.5.4)*
- *66. (EAC Program Manual, 5.6.1.1)*
- *67. (EAC Program Manual, 5.6.1.2)*
- *68. (EAC Program Manual, 5.6.1.3)*
- *69. (EAC Program Manual, 5.6.2)*
- *70. (EAC Program Manual, 5.6.2.1)*
- *71. (EAC Program Manual, 5.6.2.2)*
- *72. (EAC Program Manual, 5.6.2.3)*
- *73. (EAC Program Manual, 5.6.2.4)*

- *74.(EAC Program Manual, 5.6.3)*
- *75.(EAC Program Manual, 5.6.3.1)*
- *76. (EAC Program Manual, 5.6.3.2)*
- *77. (EAC Program Manual, 5.6.3.3)*
- *78. (EAC Program Manual, 5.6.3.4)*
- *79. (EAC Program Manual, 5.8.2)*
- *80.(EAC Program Manual, 5.9)*
- *81. (Colorado req. 21.4.8)*

### 5.1.2 Verity Data Application Level test suite

Each Chevron was navigated, with each underlying task, and task component, being exercised in order to validate that functionality within the application is able to perform the expected task for which it is designed. Each field within each screen was verified to meet the component implementation intention. Note that the application's feature was not verified, as this is not applicable to the State of Colorado's requirements.

This testing addressed requirements listed in items 1-8 and 36-61 of "Appendix A – Requirements Verified".

- *1. Create a Provisional Voting Type in Data*
- *2. (21.5.2(f)(5))*
- *3. (21.5.2.(f)(7))*
- *4. (1-5-404 and 1-5-406)*
- *5. (1-5-407(2))*
- *6. (1-5-407 (3) and 1-5-611 (1) (b))*
- *7.(1-5-407 (4.5) and 4.8.2)*
- *8. Verification of the Hart Verity Data technical reference manual, including all documented functionality and capabilities.*
- *36.(2002 VSS 2.2.6.a)*
- *37. (2002 VSS 2.2.6.b)*
- *38. (2002 VSS 2.2.6.c)*
- *39. (2002 VSS 2.2.6.d)*
- *40. (2002 VSS 2.2.6.e)*
- *41. (2002 VSS 2.2.6.f)*
- *42. Section 4.4 (2002) / 5.4 (2005). (2002 VSS 2.2.6.i)*
- *43.(2002 VSS 2.2.8.2.l)*
- *44. (2002 VSS 2.3.1.1.1.b.1)*
- *45. (2002 VSS 2.3.1.1.1.b.2)*
- *46. (2002 VSS 2.3.1.1.1.b.3)*
- *47. (2002 VSS 2.3.1.1.1.c)*
- *48. (2002 VSS 2.3.1.1.1.d)*
- *49. (2002 VSS 2.3.1.1.1.e)*
- *50. (2002 VSS 2.3.1.2.a)*
- *51. (2002 VSS 2.3.1.2.b)*
- *52. (2002 VSS 2.3.1.2.c)*

- *53. (2002 VSS 2.3.1.2.d)*
- *54. (2002 VSS 2.3.1.2.e)*
- *55. (2002 VSS 2.3.1.2.f)*
- *56. (2002 VSS 2.3.1.2.g)*
- *57.(2002 VSS 2.3.2.a)*
- *58. (2002 VSS 2.3.2.b)*
- *59. (2002 VSS 2.3.2.c)*
- *60. (2002 VSS 2.3.2.d)*
- *61. (Colorado req. 21.4.7(d)(1))*

### 5.1.3 Error and Audit test suite

Testing of Error messaging focused on the appropriate error messages being generated in response to specific errors, and content of the message. Methods employed to generate errors included attempting to access functions out of order or without authorization, erroneous responses to error messages, as well as invalid inputs.

Testing of Audit logging consisted of verifying that **Verity Data** kept an accurate record of events that occurred within the application. This included when each item type was created, modified and/or deleted. It also noted when associations between item types were created, modified and/or removed. As a support type application to the voting system proper, this was deemed to be sufficient as the election definition created within **Verity Voting 1.0 Build** documents what the items, and their relationships to other items, acts as a final log of what was done within **Verity Election Office** suite component, **Verity Data**.

This test addressed requirements listed in items 8-35 of "Appendix A – Requirements Verified".

- *8. Verification of the Hart Verity Data technical reference manual, including all documented functionality and capabilities.*
- *9. (2002 VSS 4.4.1.a)*
- *10. (2002 VSS 4.4.1.b)*
- *11. (2002 VSS 4.4.1.c)*
- *12. (2002 VSS 4.4.1.d)*
- *13. (2002 VSS 4.4.1.e)*
- *14. (2002 VSS 2.2.5.2)*
- *15. (2002 VSS 2.2.5.2.1.a)*
- *16. (2002 VSS 2.2.5.2.1.b)*
- *17. (2002 VSS 2.2.5.2.1.c)*
- *18. (2002 VSS 2.2.5.2.1.d)*
- *19. (2002 VSS 2.2.5.2.1.e)*
- *20. (2002 VSS 2.2.5.2.1.f)*
- *21. (2002 VSS 2.2.5.2.1.g.1)*
- *22. (2002 VSS 2.2.5.2.1.g.2)*
- *23. (2002 VSS 2.2.5.2.1.g.3)*

- *24. (2002 VSS 2.2.5.2.2.a)*
- *25. (2002 VSS 2.2.5.2.2.b)*
- *26. (2002 VSS 2.2.5.2.2.c)*
- *27. (2002 VSS 2.2.5.2.2.e)*
- *28. (2002 VSS 2.2.5.2.2.f)*
- *29. (2002 VSS 2.2.5.2.2.g)*
- *30. (2002 VSS 2.2.5.2.3 1st paragraph)*
- *31. (2002 VSS 2.2.5.2.3 2nd paragraph)*
- *32. (2002 VSS 2.2.5.2.3 3rd paragraph)*
- *33. (2002 VSS 2.2.5.3 3rd paragraph)*
- *34. (2002 VSS 2.2.5.3 4th paragraph)*
- *35. (2002 VSS 2.2.5.3 5th paragraph)*

### 5.1.4 Desktop test suite

**Verity Desktop** is the **Verity** application used for setting workstation date/time, accessing the desktop and gathering hash codes for **Verity Data**

- **Verity Desktop** was tested first as an individual component in order to verify that all declared functionality is present and working as documented,
- **Verity Desktop** was tested as an integrated piece of the larger application where it resides (**Verity Data**), verifying that it performs the appropriate functions for the workstation.

This testing addressed requirements listed in item 8 of "Appendix A – Requirements Verified".

- *8. Verification of the Hart Verity Data technical reference manual, including all documented functionality and capabilities.*

### 5.1.5 User Management test suite

**Verity User Manager** is the **Verity** application used for creating and managing all user roles and accounts within **Verity Data**.

- **Verity User Manager** was tested first as an individual component in order to verify that all declared functionality is present and working as documented. Different applicable user roles for the workstation were created, modified, managed and deleted.
- **Verity User Manager** was tested as an integrated piece of the larger application where it resides (**Verity Data**), verifying that it performs the appropriate functions

and manages the pertinent role for the parent application. Different applicable user roles for the workstation were verified as to role authorization.

This test addressed requirements listed in item 8 of "Appendix A – Requirements Verified".

- *8. Verification of the Hart Verity Data technical reference manual, including all documented functionality and capabilities.*

### 5.1.6 Election Manager test suite

**Verity Election Manager** is the **Verity** application used for adding, copying, importing, exporting, archiving and restoring election data into and from **Verity Data**.

- **Verity Election Manager** was tested first as an individual component in order to verify that all declared functionality is present and working as documented. Functionality was verified for importing, exporting, archiving and restoring election data sets
- **Verity Election Manager** was then tested as an integrated piece of the larger application where it resides (**Verity Data**), verifying that it performs the appropriate functions for the parent application. Functionality was verified that a newly added election data set (which is basically an empty election structure), an existing election data set imported and an archived election data set were able to be used by **Verity Data**. Outputs from **Verity Data** were also verified to be able to be exported, copied and archived by **Verity Election Manager**.

This testing addressed requirements listed in item 8 of "Appendix A – Requirements Verified".

- *8. Verification of the Hart Verity Data technical reference manual, including all documented functionality and capabilities.*

### 5.1.7 Integrate Data Output into Verity Voting (General) test suite

This test created a new General election data set, then took the output from **Verity Data** and imported the generated xml data into the **Verity Voting 1.0** voting system, and included the verification of ballot layouts for paper ballots, electronic ballots, and audio ballots by ballot style in both English and Spanish languages.

This test addressed requirements listed in items 1-8, 39-41 and 49 of "Appendix A – Requirements Verified".

- *1. Create a Provisional Voting Type in Data*
- *2. (21.5.2(f)(5))*
- *3. (21.5.2.(f)(7))*
- *4. (1-5-404 and 1-5-406)*
- *5. (1-5-407(2))*
- *6. (1-5-407 (3) and 1-5-611 (1) (b))*
- *7. (1-5-407 (4.5) and 4.8.2)*
- *8. Verification of the Hart Verity Data technical reference manual, including all documented functionality and capabilities.*
- *39. (2002 VSS 2.2.6.d)*
- *40. (2002 VSS 2.2.6.e)*
- *41. (2002 VSS 2.2.6.f)*
- *49. (2002 VSS 2.3.1.1.1.e)*

### 5.1.8 Integrate Data Output into Verity Voting (Primary) test suite

This test created a new Primary election data set, then took the output from **Verity Data** and imported the generated xml data into the **Verity Voting 1.0** voting system, and included the verification of ballot layouts for paper ballots, electronic ballots, and audio ballots by ballot style in both English and Spanish languages.

This test addressed requirements listed in items 1-8, 39-41 and 49 of "Appendix A – Requirements Verified".

- *1. Create a Provisional Voting Type in Data*
- *2. (21.5.2(f)(5))*
- *3. (21.5.2.(f)(7))*
- *4. (1-5-404 and 1-5-406)*
- *5. (1-5-407(2))*
- *6. (1-5-407 (3) and 1-5-611 (1) (b))*
- *7. (1-5-407 (4.5) and 4.8.2)*
- *8. Verification of the Hart Verity Data technical reference manual, including all documented functionality and capabilities.*
- *39. (2002 VSS 2.2.6.d)*
- *40. (2002 VSS 2.2.6.e)*
- *41. (2002 VSS 2.2.6.f)*
- *49. (2002 VSS 2.3.1.1.1.e)*

# 6 Test Results Summary

## 6.1 Functional Testing Summary

SLI executed the testing as identified in Section 5.1. The testing incorporated two different election scenarios, one General election and one Closed Primary election, as well as testing of each applications specific feature set, and specific attention to error messaging and audit logging, testing the functionality supported by **Verity Data** in its kiosk environment.

All requirements were tested and verified to be met by the **Verity Data** environment.

## 6.2 Deficiencies

SLI has determined that only a few minor issues were found in the Verity Data environment. These included:

- In the Select Election screen, when more than 34 elections are listed, the screen may blankly display, showing no elections. This can be resolved by archiving and then removing elections, via **Verity Election Manager**.
- In the Districts screen there is a “Help” screen that incorrectly states that 600 districts can be created, but the correct number is 75. Hart has updated documentation to advise users of this fact.
- In the Polling Place screen, when creating over 20 places, the new records being added display only the top half of the fields displayed, which makes it a bit inconvenient to read what is being input, but does not prevent continued creation of polling places.

# 7 Appendix A - Requirements Verified

1. Create a Provisional Voting Type in Data
2. Include a recall question (21.5.2(f)(5))
3. Include a ballot question of at least 200 words (21.5.2.(f)(7))
4. Be able to demonstrate that the order of candidates can be set within the software as needed (1-5-404 and 1-5-406)
5. Include contest instruction (the dataset will include this) (Vote for not more than one, etc.) (1-5-407(2))
6. Include write-in lines (1-5-407 (3) and 1-5-611 (1) (b))
7. Include “there are no candidates filed for this election” (1-5-407 (4.5) and 4.8.2)
8. Verification of the Hart Verity Data technical reference manual, including all documented functionality and capabilities.
9. Pre-election Audit record, log shall include allowable number of selection for an office or issue (2002 VSS 4.4.1.a)
10. Pre-election Audit record, log shall include combinations of voting patterns permitted or required by jurisdiction (2002 VSS 4.4.1.b)
11. Pre-election Audit record, log shall include inclusion or exclusion of offices or issues as result of multiple districting with polling place (2002 VSS 4.4.1.c)
12. Pre-election Audit record, log shall include Any other characteristics that may be peculiar to jurisdiction, election or polling place (2002 VSS 4.4.1.d)
13. Pre-election Audit record, log shall include Manual data maintained by election personnel (2002 VSS 4.4.1.e)
14. Audit records shall be prepared of ballot preparation and election definition phase (2002 VSS 2.2.5.2)

15. Audit records shall provide capability to create and maintain a real-time audit record (2002 VSS 2.2.5.2.1.a)
16. All systems shall include a real-time clock and shall maintain an absolute record of the date and time or a record relative to some event whose time and date are known and recorded. (2002 VSS 2.2.5.2.1.b)
17. On all systems, audit record entries shall include the time-and-date stamp (2002 VSS 2.2.5.2.1.c)
18. The audit records shall be active whenever the system is in an operating mode. This record shall be available at all times, though it need not be continually visible. (2002 VSS 2.2.5.2.1.d)
19. The generation of audit record entries shall not be terminated or altered by program control, or by the intervention of any person. The physical security and integrity of the record shall be maintained at all times. (2002 VSS 2.2.5.2.1.e)
20. Once the system has been activated for any function, the system shall preserve the contents of the audit record during any interruption of power to the system until processing and data reporting have been completed. (2002 VSS 2.2.5.2.1.f)
21. The system shall be capable of printing a copy of the audit record. A separate printer is not required for the audit record, and the record may be produced on the standard system printer if all the following conditions are met: The generation of the audit trail records does not interfere with the production of output reports. (2002 VSS 2.2.5.2.1.g.1)
22. The system shall be capable of printing a copy of the audit record. A separate printer is not required for the audit record, and the record may be produced on the standard system printer if all the following conditions are met: The entries can be identified to facilitate recognition, segregation and retention. (2002 VSS 2.2.5.2.1.g.2)
23. The system shall be capable of printing a copy of the audit record. A separate printer is not required for the audit record, and the record may be produced on the standard system printer if all the following conditions are met: The audit record entries are kept physically secure. (2002 VSS 2.2.5.2.1.g.3)
24. The system shall generate, store and report to the use all error messages as they occur (2002 VSS 2.2.5.2.2.a)
25. All error messages requiring intervention by an operator or precinct official shall be displayed or printed unambiguously in easily understood language text, or by means of other suitable visual indicators. (2002 VSS 2.2.5.2.2.b)
26. When the system uses of numerical error codes for trained technician maintenance or repair, the text corresponding to the code shall be

self-contained, or affixed inside the unit device. (2002 VSS 2.2.5.2.2.c)

27. The message cue for all systems shall clearly state the action to be performed in the event that voter or operator response is required. (2002 VSS 2.2.5.2.2.e)
28. System design shall ensure that erroneous responses will not lead to irreversible error. (2002 VSS 2.2.5.2.2.f)
29. Nested error conditions shall be corrected in a controlled sequence such that system status shall be restored to the initial state existing before the first error occurred. (2002 VSS 2.2.5.2.2.g)
30. Jurisdictions may require some status and information messages to be displayed and reported in real-time. Messages that do not require operator intervention may be stored in memory to be recovered after ballot processing is completed. (2002 VSS 2.2.5.2.3 1st paragraph)
31. The system shall display and report of critical status messages using unambiguous indicators or English language text. The voting system need not display non-critical status messages at the time of occurrence. Voting systems may display non-critical status messages (i.e., those that do not require operator intervention) by means of numerical codes for subsequent interpretation and reporting as unambiguous text. (2002 VSS 2.2.5.2.3 2nd paragraph)
32. Systems shall provide a capability for the status messages to become part of the real-time audit record. The system shall provide a capability for a jurisdiction to designate critical status messages. (2002 VSS 2.2.5.2.3 3rd paragraph)
33. Authentication shall be configured on the local terminal (display screen and keyboard) and on all external connection devices (network cards and ports). (2002 VSS 2.2.5.3 3rd paragraph)
34. The operating system audit shall be enabled for all session openings and closings, for all process executions and terminations, and for the alteration or deletion of any memory or file object. (2002 VSS 2.2.5.3 4th paragraph)
35. The system shall be configured to execute only intended and necessary processes during the execution of election software. The system shall also be configured to halt election software processes upon the termination of any critical system process (such as system audit) during the execution of election software. (2002 VSS 2.2.5.3 5th paragraph)
36. An EMS shall generate and maintain a database, or one or more interactive databases, that enables election officials or their designees to perform the following: Define political subdivision boundaries and multiple election districts as indicated in the system documentation. (2002 VSS 2.2.6.a)

37. An EMS shall generate and maintain a database, or one or more interactive databases, that enables election officials or their designees to perform the following: Identify contests, candidates, and issues. (2002 VSS 2.2.6.b)
38. An EMS shall generate and maintain a database, or one or more interactive databases, that enables election officials or their designees to perform the following: Define ballot formats and appropriate voting options. (2002 VSS 2.2.6.c)
39. An EMS shall generate and maintain a database, or one or more interactive databases, that enables election officials or their designees to perform the following: Generate ballots and election-specific programs for vote recording and vote counting equipment. (2002 VSS 2.2.6.d)
40. An EMS shall generate and maintain a database, or one or more interactive databases, that enables election officials or their designees to perform the following: Install ballots and election-specific programs. (2002 VSS 2.2.6.e)
41. An EMS shall generate and maintain a database, or one or more interactive databases, that enables election officials or their designees to perform the following: Test that ballots and programs have been properly prepared and installed. (2002 VSS 2.2.6.f)
42. An EMS shall generate and maintain a database, or one or more interactive databases, that enables election officials or their designees to perform the following: Process and produce audit reports of the data indicated in Section 4.4 (2002) / 5.4 (2005). (2002 VSS 2.2.6.i)
43. Recall issues with options. (2002 VSS 2.2.8.2.l)
44. All systems shall be capable of collecting and maintaining the following data: Offices and their associated labels and instructions. (2002 VSS 2.3.1.1.1.b.1)
45. All systems shall be capable of collecting and maintaining the following data: Candidate names and their associated labels. (2002 VSS 2.3.1.1.1.b.2)
46. All systems shall be capable of collecting and maintaining the following data: Issues or measures and their text. (2002 VSS 2.3.1.1.1.b.3)
47. All systems shall be capable of supporting the maximum number of potentially active voting positions as indicated in the system documentation. (2002 VSS 2.3.1.1.1.c)
48. All systems shall be capable of for a primary election, generating ballots that segregate the choices in partisan races by party affiliation. (2002 VSS 2.3.1.1.1.d)

49. All systems shall be capable of generating ballots that contain identifying codes or marks uniquely associated with each format. (2002 VSS 2.3.1.1.1.e)
50. All systems shall be capable of creation of newly defined elections. (2002 VSS 2.3.1.2.a)
51. All systems shall be capable of rapid and error-free definition of elections and their associated ballot layouts. (2002 VSS 2.3.1.2.b)
52. All systems shall be capable of uniform allocation of space and fonts used for each office, candidate, and contest such that the voter perceives no active voting position to be preferred to any other. (2002 VSS 2.3.1.2.c)
53. All systems shall be capable of Simultaneous display of the maximum number of choices for a single contest as indicated by the vendor in the system documentation. (2002 VSS 2.3.1.2.d)
54. All systems shall be capable of retention of previously defined formats for an election (2002 VSS 2.3.1.2.e)
55. All systems shall be capable of prevention of unauthorized modification of any ballot formats (2002 VSS 2.3.1.2.f)
56. All systems shall be capable of modification by authorized persons of a previously defined ballot format for use in a subsequent election. (2002 VSS 2.3.1.2.g)
57. All systems shall provide for the: Logical definition of the ballot, including the definition of the number of allowable choices for each office and contest. (2002 VSS 2.3.2.a)
58. All systems shall provide for the: Logical definition of political and administrative subdivisions, where the list of candidates or contests varies between polling places. (2002 VSS 2.3.2.b)
59. All systems shall provide for the: Exclusion of any contest on the ballot in which the voter is prohibited from casting a ballot because of place of residence, or other such administrative or geographical criteria. (2002 VSS 2.3.2.c)
60. All systems shall provide for the: Ability to select from a range of voting options to conform to the laws of the jurisdiction in which the system will be used. (2002 VSS 2.3.2.d)
61. Provide a facility for the definition of the ballot, including the definition of the number of allowable choices for each office and contest and for special voting options such as write-in candidates (Colorado req. 21.4.7(d)(1))
62. Specifically, the trusted build will do the following: Demonstrate that the software was built as described in the Technical Data Package. (EAC Program Manual, 5.5.1)

63. Specifically, the trusted build will do the following: Show that the tested and approved source code was actually used to build the executable code used on the system. (EAC Program Manual, 5.5.2)
64. Specifically, the trusted build will do the following: Demonstrate that no elements other than those included in the Technical Data Package were introduced in the software build. (EAC Program Manual, 5.5.3)
65. Specifically, the trusted build will do the following: Document for future reference the configuration of the system certified. (EAC Program Manual, 5.5.4)
66. The VSTL shall construct the build environment in an isolated environment controlled by the VSTL, as follows: The device that will hold the build environment shall be completely erased by the VSTL to ensure a total and complete cleaning of it. The VSTL shall use commercial off-the-shelf software, purchased by the laboratory, for cleaning the device. (EAC Program Manual, 5.6.1.1)
67. The VSTL shall construct the build environment in an isolated environment controlled by the VSTL, as follows: The VSTL, with vendor consultation and observation, shall construct the build environment. (EAC Program Manual, 5.6.1.2)
68. The VSTL shall construct the build environment in an isolated environment controlled by the VSTL, as follows: After construction of the build environment, the VSTL shall produce and record a file signature of the build environment. (EAC Program Manual, 5.6.1.3)
69. After successful source code review, the VSTL shall load source code onto the build environment (EAC Program Manual, 5.6.2)
70. The VSTL shall check the file signatures of the source code modules and build environment to ensure that they are unchanged from their original form. (EAC Program Manual, 5.6.2.1)
71. The VSTL shall load the source code onto the build environment and produce and record the file signature of the resulting combination. (EAC Program Manual, 5.6.2.2)
72. The VSTL shall capture a disk image of the combination build environment and source code modules immediately before performing the build. (EAC Program Manual, 5.6.2.3)
73. The VSTL shall deposit the disk image into an authorized archive to ensure that the build can be reproduced, if necessary, at a later date. (EAC Program Manual, 5.6.2.4)
74. Creating the Executable Code. Upon completion of all the tasks outlined above, the VSTL shall produce the executable code. (EAC Program Manual, 5.6.3)
75. The VSTL shall produce and record a file signature of the executable code. (EAC Program Manual, 5.6.3.1)

76. The VSTL shall deposit the executable code into an EAC-approved software repository and create installation disk(s) from the executable code.EAC Program Manual, 5.6.3.2)
77. The VSTL shall produce and record file signatures of the installation disk(s) in order to provide a mechanism to validate the software before installation on the voting system in a purchasing jurisdiction. (EAC Program Manual, 5.6.3.3)
78. The VSTL shall install the executable code onto the system submitted for testing and certification before completion of system testing. (EAC Program Manual, 5.6.3.4)
79. Software operating on a host computer will typically be verified by providing a self-booting compact disk (CD) or similar device that verifies the file signatures of the voting system application files AND the signatures of all nonvolatile files that the application files access during their operation. Note that the creation of such a CD requires having a file map of all nonvolatile files that are used by the voting system. Such a tool must be provided for verification using the file signatures of the original executable files provided for testing. (EAC Program Manual, 5.8.2)
80. Manufacturers shall provide documentation to the Program Director verifying that the trusted build has been performed, software has been deposited in an approved repository, and system identification tools are available to election officials. The Manufacturer shall submit a letter, signed by both its management representative and a VSTL official, stating (under penalty of law) that it has (1) performed a trusted build consistent with the requirements of Section 5.6 of this Manual, (2) deposited software consistent with Section 5.7 of this Manual, and (3) created and made available system identification tools consistent with Section 5.8 of this Manual. This letter shall also include (as attachments) a copy and description of the system identification tool developed under Section 5.8 above. (EAC Program Manual, 5.9)
81. The voting system must allow the operating system administrative account to verify that the software installed is the certified software by comparing it to the trusted build or other reference information (Colorado req. 21.4.8)

## End of Test Report