Verity Data for State of Colorado
Phase II Test Plan

Document Number: HRT-TP-062015-04

Prepared for:

Vendor Name	Hart InterCivic
Vendor Application	Verity Data
Vendor Address	15500 Wells Port Drive, Austin, TX 78728

Prepared by:

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

Copyright © 2015 by SLI Global Solutions, LLC.
All products and company names are used for identification purposes only and may be trademarks of their respective owners.

Revision History

Date	Release	Author	Revision Summary
June 26, 2015	1.0	*M. Santos*	Initial Test Plan
July 7, 2015	1.1	*M. Santos*	Updated to include section 2.1 Software/Firmware, added note in section 2.2 Equipment, of separate environments for Verity Data and Verity Build, in section 3.3.6 included verification of ballot and audio data in Verity Build
July 10, 2015	1.2	*M. Santos*	Added in additional requirements into appendix A, as requested by the State, and updated references in testing sections. Added in section 3.3.1 Trusted Build
July 11, 2015	1.3	*M. Santos*	Updated several requirements as requested by the State. Added in 2 additional requirements, as items 61 and 81 in the appendix, and updated references in the document body.

Disclaimer

The information 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.

Trademarks

- SLI is a registered trademark of SLI Global Solutions, Inc.
- Intel and Pentium are registered trademarks of Intel Corporation.
- Microsoft, MS are registered trademarks and Windows XP, Windows Server 2003 are trademarks of Microsoft Corporation.
- All products and company names are used for identification purposes only and may be trademarks of their respective owners.

Managing technology risk

TABLE OF CONTENTS

1 INTRODUCTION 4

1.1 REFERENCES 4
1.2 TERMS AND ABBREVIATIONS 4
1.3 PROJECT OVERVIEW 5
1.4 PURPOSE 5
1.5 SCOPE OF TESTING 6
1.6 TESTING RESPONSIBILITIES 6
1.6.2 State of Colorado & Manufacturer Dependencies 7
1.7 SCOPE OF THE HART VERITY DATA DATA MANAGEMENT SOFTWARE 7
1.7.1 Application Overview 7

2 MATERIALS REQUIRED FOR TESTING 8

2.1 SOFTWARE 8
2.1.1 Verity Software 9
2.1.2 COTS Software/Firmware 9
2.2 EQUIPMENT 9
2.2.1 COTS Equipment 9

3 TEST SPECIFICATIONS 10

3.1 REQUIREMENTS 10
3.2 HARDWARE CONFIGURATION AND DESIGN 11
3.3 HOW REQUIREMENTS WILL BE TESTED 11
3.3.1 Trusted Build 11
3.3.2 Data Application Level test suite 11
3.3.3 Error and Audit test suite 12
3.3.4 Desktop test suite 12
3.3.5 User Management test suite 12
3.3.6 Election Manager test suite 13
3.3.7 Integrate Data Output into Verity Voting (General) test suite 13
3.3.8 Integrate Data Output into Verity Voting (Primary) test suite 13

4 TEST OPERATIONS PROCEDURES 13

5 APPROVAL SIGNATURES 14

6 APPENDIX A - REQUIREMENTS TO BE VERIFIED 15

Managing technology risk

1 INTRODUCTION

This Test Plan outlines the test approach SLI Global Solutions will follow when performing testing for the State of Colorado, on the ***Hart Verity Data*** data management software. The purpose of this document is to provide a clear understanding of the work SLI will conduct and a detailed plan outlining the test effort.

When the testing is complete, SLI will submit a Test Report that details all test results and findings from the Test effort.

1.1 References

The following is a listing of all documents that contain material that was used in the preparation of this test plan.

1. SLI VSTL Quality System Manual, Rev. 1.16, prepared by SLI, dated December 3, 2013.

1.2 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 products (such as operating systems, programming language compilers, or database management systems)
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.
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.

Managing technology risk

Term	Abbreviation	Description
Tab	No Abbreviation	A second, or lower, level of application functionality, as broken out by intended usage.
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)
Verification -	No Abbreviation	Confirmation by examination and through provision of objective evidence that specified requirements have been fulfilled (ISO 9000)
Voting System Test Lab	VSTL	This is the lab where the voting system is being tested.
Voting Test Specialist	VTS	A SLI employee within the Compliance division that has been qualified to perform voting system testing.

1.3 Project Overview

This test plan outlines the approach SLI will implement to perform testing of the **Hart Verity Data** data management software against the requirements in "Appendix A – Requirements to be Verified", as well as the **Hart Verity Data** Technical Reference Manual requirements to be validated for the State of Colorado certification program.

1.4 Purpose

The purpose of this Test Plan is to create clear and precise documentation of the processes that SLI, will implement throughout the course of voting system verification testing.

This Test Plan:

- Defines the overall test approach.
- Identifies required **Verity Data** software/hardware and additional **Verity Voting 1.**0 components to be tested.
- Serves as a foundation for the development of relevant **Verity Data** test suites that will verify the requirements being tested

Managing technology risk

1.5 Scope of Testing

SLI will provide testing on the ***Hart Verity Data*** data management software based on the requirements listed in "Appendix A – Requirements to be Verified", as well as the "**Hart Verity Data** Technical Reference Manual", to satisfy requirements for the State of Colorado.

SLI's major task categories for **Hart Verity Data** verification testing, as defined by SLI's Quality System Manual, include:

- Management of **Hart Verity Data** supplied deliverables
- Generation of test modules and suites to ensure the data management is tested against all applicable requirements as declared by the State of Colorado.
- Traceability of test modules to the requirements.
- Reporting of all test results.

SLI will develop and submit to the State of Colorado, a final test report for the **Hart Verity Data** data management software.

1.6 Testing Responsibilities

The following schedule describes the high level tasks and assigned personnel titles that will be involved in the test effort of the ***Hart Verity Data*** data management software.

1.6.1.1 Owner Assignments

- Test Module Development and Validation will be conducted by Voting Test Specialist (VTS) personnel, with oversight provided by the Test Manager.
- Test Suite Development and Validation will be conducted by VTS personnel, with oversight provided by the Test Manager.
- Formal Test Execution will be conducted by VTS personnel, with oversight provided by the Test Manager.

Managing technology risk

1.6.2 State of Colorado & Manufacturer Dependencies

The Test Plan will require State of Colorado approval prior to finalization.

Hart InterCivic will be required to provide all documentation, equipment and supporting materials identified as part of the data management environment.

Hart InterCivic will be required to provide all identified components of the **Verity Voting 1.0** voting system identified, as well as any supporting materials required for the voting system. SLI will verify that the output xml from Verity Data is able to be successfully imported into Verity Voting 1.0's Verity Build application.

In addition, **Hart InterCivic** is required to provide support for the duration of the project.

1.7 Scope of the Hart Verity Data data management software

This section provides a description of the scope of ***Hart Verity Data*** data management software:

- Verity Data is a data management application that allows users to manage election and jurisdiction data in a manner that is compatible for use with the Verity Voting system.

1.7.1 Application Overview

The **Hart Verity Data** data management provides the capability to create the xml input required by the **Verity Voting 1.0** voting system. This includes:

- Setting preferences for an election
- Basic election information
- Election voting types
- Political parties
- Preferred wording for specific field labels
- Special election rules
- Jurisdictional precincts
- Jurisdictional districts
- Jurisdictional polling places
- Contest titles
- Contest choices

- Ballot instructions
- Ballot additional text
- Ballot order
- Ballot header information
- Ballot stub information
- Ballot translations
- Ballot audio information
- Import of ballot data
- Export of ballot data
- Preview of ballot layout
- Data validation of ballot information
- Reports
- Export of election information for import into Verity Voting

2 MATERIALS REQUIRED FOR TESTING

Any materials that are used in creation of election information by **Verity Data**, must be provided to SLI to facilitate testing of the data management system. This section outlines these materials that are required.

2.1 Software

Any and all software that is to be used by the declared voting system whether directly or indirectly, in a production environment, must be validated during the testing process.

The following software is required for the execution of testing to verify the requirements within the scope of this testing effort. This includes all supporting software such as operating systems, compilers, assemblers, application software, any applications used for burning of media or creation/management of databases.

Managing technology risk

2.1.1 Verity Software

Verity Election Office *and* ***Verity Voting 1.0*** *components under review within this test plan consist of the following software components:*

Table 2 – Hart Verity Software

Application	Description	Version
Verity Data	**Verity Election Office** data management software	1.3.3
Verity Build	**Verity Voting 1.0** EMS software	1.0.3

2.1.2 COTS Software/Firmware

This section details the Commercial Off The Shelf software and firmware utilized within the **Verity Data** and **Verity Build** environments.

Table 3 – COTS Software/Firmware

Manufacturer	Application	Version
Microsoft	Windows 7 Embedded	Standard
Microsoft	Windows 7, Service Pack 1	6.1.7601
Microsoft	SQL Server	6.1
Adobe	Acrobat	10.0

2.2 Equipment

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

2.2.1 COTS Equipment

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

- Desktops
- Printers

Managing technology risk

Table 4 – COTS Equipment

Manufacturer	Hardware	Model
OKIDATA	Ballot Printer (Monochrome)	831
Various	**Intel-Windows Workstation (Recommended Requirements)** Processor – x86-compatible, 3.0GHz, Quad Core Memory – 8GB Hard Drive – 2 x 1 TB RAID-Level 1, Removable w/ key lock Ethernet Port – 100Mb/1Gb USB Ports – 4 ports Video Card - Integrated Graphics Keyboard - USB Keyboard Mouse - USB Mouse NO Wireless technologies allowed: WiFi, Bluetooth, Aircard, etc.	
Various	**Monitor (Minimum Requirements)** Panel Size - 50.8 cm Aspect Ratio - Widescreen (16:9) Optimal Resolution - 1600 x 900 at 60 Hz Contrast Ratio - 1000: 1 Brightness - 250 cd/m2 (typical)	

3 TEST SPECIFICATIONS

The following are the specifications for testing to be conducted on the **Hart Verity Data** data management software. The specifications contain details on the focus of testing, configuration(s), and the requirements to be tested.

3.1 Requirements

The **Hart Verity Data** will be tested to the Colorado requirements needing to be verified, as listed in “Appendix A – Requirements to be Verified”.

3.2 Hardware Configuration and Design

The **Hart Verity Data** data management software needed includes:

- A **Verity Data** workstation to generate the xml input file.

For the two integration tests to be performed, the **Hart Verity Voting 1.0** voting system needed, includes:

- A **Verity Voting Build** workstation to validate the xml file outputted by **Verity Data**.

Note that **Verity Data** resides in a separate environment from **Verity Build**. Both environments consist of identical hardware. 2 separate environments are required for this testing, 1 **Verity Data** environment and 1 **Verity Build** environment.

3.3 How Requirements Will Be Tested

This section looks at the **Hart Verity Data** data management software determines the testing strategy to be implemented in order to verify the **Verity Data** data management software and its integration into the **Verity Voting 1.0** system.

3.3.1 Trusted Build

A Trusted Build will be performed, where SLI creates the build environment from pristine components, integrates the Verity Data source code into the environment and then proceeds to compile the executable code.

This build process will address requirements listed in items 62-81 of “Appendix A – Requirements to be Verified”.

3.3.2 Data Application Level test suite

Each Chevron will be navigated, with each underlying task, and task component, being exercised in order to validate that each functionality within the application is able to perform the expected task for which it is designed. Each field within each screen will be verified to meet the design implementation intention.

This test will address requirements listed in items 1-8 and 36-61 of “Appendix A – Requirements to be Verified”.

Managing technology risk

3.3.3 Error and Audit test suite

Voting systems can be subject to various conditions and when the system exceeds limitations errors are typically found. Testing of Error messaging focus' on the appropriate error messages being generated in response to specific errors, and content of the message. Methods employed to generate errors include attempting to access functions out of order or without authorization, erroneous responses to error messages, as well as invalid inputs.
Audit records are used to track what system functions have been executed, what data has been modified, as well as by who and when. They present a concrete, indestructible archival record of all application activity, and are essential for public confidence. Additionally, audit record data content can be a key factor in identifying system anomalies and provide assistance in troubleshooting system errors.

This test will address requirements listed in items 8-35 of "Appendix A – Requirements to be Verified".

3.3.4 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**
o **Verity Desktop** will be tested first as an individual component in order to verify that all declared functionality is present and working as documented,
o **Verity Desktop** will then be 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.

This test will address requirements listed in item 8 of "Appendix A – Requirements to be Verified".

3.3.5 User Management test suite

Verity User Manager is the **Verity** application used for creating and managing all user roles and accounts within **Verity Data**.
o **Verity User Manager** will be tested first as an individual component in order to verify that all declared functionality is present and working as documented,
o **Verity User Manager** will then be 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.

This test will address requirements listed in item 8 of "Appendix A – Requirements to be Verified".

3.3.6 Election Manager test suite

Verity Election Manager is the **Verity** application used for importing, exporting, archiving and restoring elections into and from **Verity Data**.
o **Verity Election Manager** will be tested first as an individual component in order to verify that all declared functionality is present and working as documented,
o **Verity Election Manager** will then be 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.

This test will address requirements listed in item 8 of "Appendix A – Requirements to be Verified".

3.3.7 Integrate Data Output into Verity Voting (General) test suite

This test will take the output from **Verity Data** and import the generated xml data into the **Verity Voting 1.0** voting system, and will include the verification of ballot layouts for paper ballots, electronic ballots, and audio ballots by ballot style in both English and Spanish languages.

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

3.3.8 Integrate Data Output into Verity Voting (Primary) test suite

This test will take the output from **Verity Data** and import the generated xml data into the **Verity Voting 1.0** voting system.

This test will address requirements listed in items 1-8 of "Appendix A – Requirements to be Verified".

4 TEST OPERATIONS PROCEDURES

Throughout the testing effort, test suites and modules will be marked as follows:

- **Accept –** Test is accepted as successful.
- **Reject** – Test is rejected as unsuccessful.
- **NT –** Not Testable is used for test modules that cannot be followed. For example, if failure of one test modules failure precludes attempting subsequent test modules, the latter will be marked as NT.

Managing technology risk

Test results **Reject** and/or **NT** will include comments explaining the reason for the result.

Issues encountered during testing will be documented in the Discrepancy Report. Issues that do not conform to the requirements of the State of Colorado will be marked as **Documentation Discrepancies** or **Functional Discrepancies** (a discrepancy occurs when the does not meet defined requirements or specifications).

5 Approval Signatures

SLI:

Traci Mapps
VSTL Director
July 11th, 2015

Managing technology risk

6 Appendix A - Requirements to be 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 State of Colorado Verity Data Test Plan