



Agency Interface Validation Plan Overview

December 15, 2005



Presentation Objectives

- Provide Agencies with an understanding of both **informal** and **formal** Agency Interface Validation Testing (AIVT)
- Provide Agencies with an introduction to documentation requirements for planning and executing AIVT



Agency Interface Validation Plan

● Purpose

- Serves as a roadmap for Project Aspire and Agencies to perform activities required to complete the Agency Interface Validation Testing (AIVT).
- Identifies the prerequisites, the tasks, and the timeframe for the AIVT, as well as providing templates for documentation.
- Provides the Agencies with an opportunity to identify and train teams in preparation for the AIVT.
- Describes the approaches to the AIVT, both *informal* and *formal*.



Approaches to AIVT

- **Informal**

- AIVT provides an Agency the opportunity to test an interface without formal testing procedures such as test scripts, test results, and defect tracking and reporting but is intended to provide lessons learned.

- **Formal**

- AIVT is the prerequisite for Aspire cutover to production as part of the Wave rollout. It is to ensure that the end-to-end functionality from an Agency's system to the downstream processes of the Aspire system is acceptable to the agency.



- **Informal Testing**

- The Aspire Project Team will schedule a kickoff meeting with Agencies representatives, as needed, prior to the testing.

- **Formal Testing**

- The Aspire Project Team will schedule a kickoff meeting with the Agencies participating in each wave rollout prior to testing. Roles and responsibilities, communication plan, test plan, and documentation will be reviewed during this meeting.



Test Execution: Formal and Informal

- Execution of the ***informal*** AIVT, once the preparation is completed, is a simple process of conducting the kickoff meeting, scheduling the execution of the test, conducting follow up meetings to discuss the outcome and document lessons learned.
- *Execution of the ***formal*** AIVT essentially follows the same flow but requires the formal use of test scripts, defect tracking, defect resolution, and test acceptance.*



Agency Interface Validation Plan

● Interface Testing Execution Roles

(P) Primary; (S) Secondary; (X) Participant

<i>Aspire Testing Teams</i>	<i>Project Aspire Teams</i>					<i>Agency Representatives</i>	
	<i>Application Software Group</i>	<i>Testing Support Group</i>	<i>Technical: Application Development Team</i>	<i>Technical: Enterprise Interface Team</i>	<i>Technical: Infrastructure Team</i>	<i>Agency Technical Lead</i>	<i>Agency Advocate</i>
<i>Execute Kick-off</i>	X	P	X	X	X	X	X
<i>Build Test Environment</i>					P		
<i>Test Environment Data/Configuration Set-up</i>	P	S	S	S		X	X
<i>Execute Test Cases and Test Sets in Test Director</i>	S	S	S	P			X
<i>Analyze Results and Log Defects in Test Director</i>	S	S	S	P		S	X
<i>Defects Resolution</i>	S	S	S	P	S	X	X
<i>Reporting and Final Documentation</i>	S	S	S	P			X
<i>Exit Meeting</i>	X	P	X	X	X	X	X



The approach to accomplishing the AIVT includes:

- Organize for Testing
- Design/Build Test Scripts
- Design/Build Test Procedures
- Build Test Environment
- Execute Test
- Resolve Test Defects
- Test Acceptance



Agency Interface Validation Plan

For additional information, please contact Charles Ghini at:

charles.ghini@fldfs.com



Q&A



Chart of Accounts (COA) Redesign

December 15, 2005



- **COA Redesign Process**
- **Key Issues with Original Design**
- **New COA Structure**
- **Ensuring Data Integrity**
- **SetID Use**
- **Aspire Trees**
- **Combination Editing**
- **Questions & Answers**



COA Redesign Process



COA Redesign Process

9/1 – 10/5

10/3 – 10/28

10/17 – 11/09

11/10 – 12/20

12/5 - 12/23

**Identify Current
COA Issues**

**Validate
Requirements**

**Develop and
Present
Solutions**

**Prototype
Solution**

**Approve/
Sign-off**

- Conducted an analysis to identify current design issues
- Created the configuration analysis document
- Created a high level approach and timeline for redesign

- Conducted series of workshops
- Classified requirements into groupings: Business, Statutory and Reporting
- Validated with LAS/PBS & Capitol Stakeholders
- Completed the 'Requirements Document'

- Identified key design guidelines
- Developed alternate proposed solutions
- Validated proposed solutions with stakeholder groups
- Performed project wide technical/ functional impact analysis

- Performed prototyping
- Tested from a configuration perspective key functional and processing concepts
- Validate from a conceptual perspective, the proposed solution's reporting and coding aspects

- Complete 'Design Document'
- Stakeholder acceptance and sign-off



Key Issues – Original Design



Key Issues with Original Design

The following table summarizes key issues with the previous COA design:

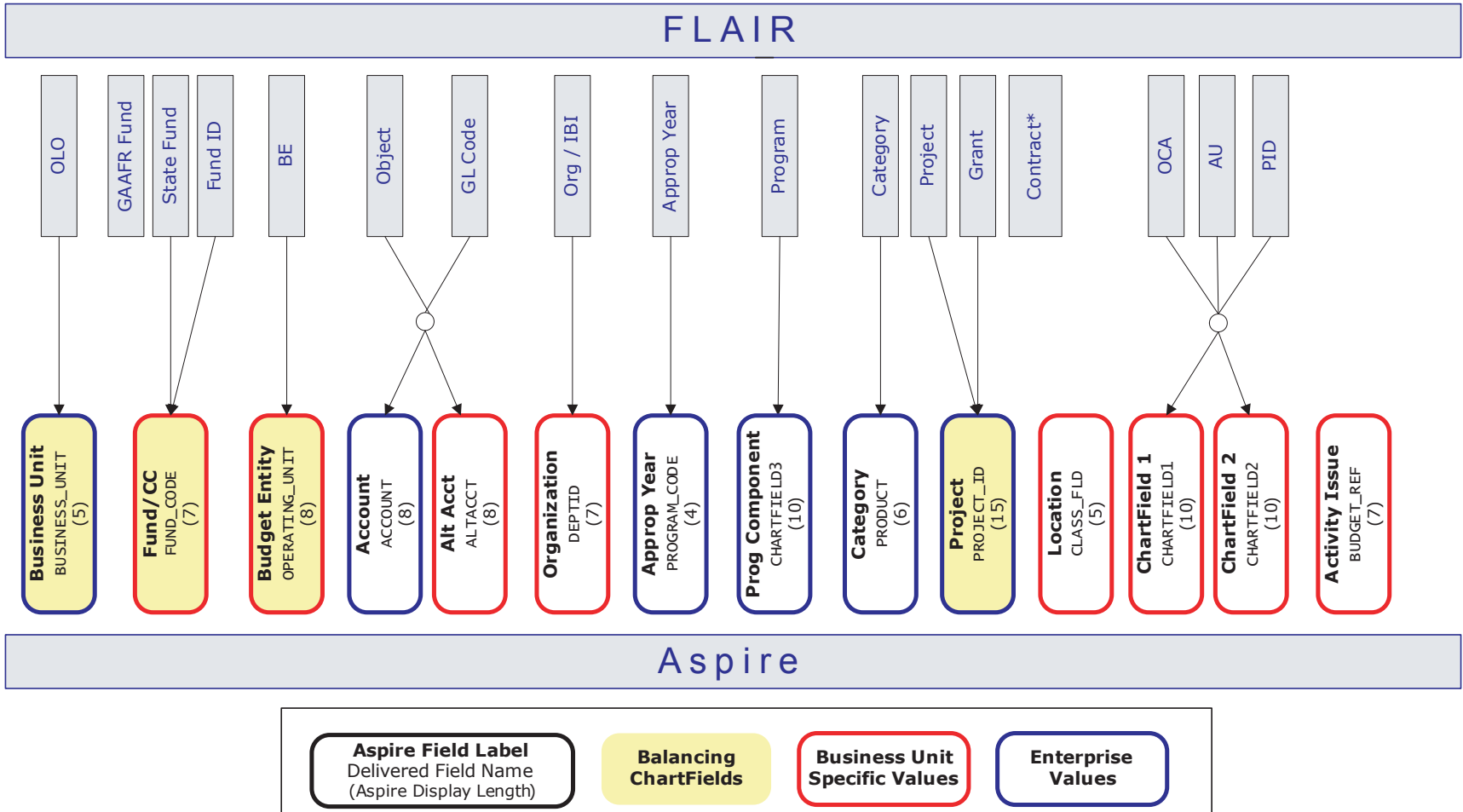
Issue	Description
Unable to meet certain key statutory requirements	COA did not share common structure with FFMIS; Unable to meet some key reporting requirements; Inability to produce financials by Budget Entity (BE) & other ChartFields
Unable to enforce data element input relationships	Enforcing data integrity extremely difficult between : BE & Fund; Fund CC & LAS/PBS Fund; BE & Program Component; Account & Category; FCO Appropriations & Year
Usability and Recognizability	Unable to easily recognize, input and report on the GAA line item components in an efficient manner
Maintenance	Original COA design increased the level of effort required to maintain values, related trees, speedcharts and combination edits
Flexibility	Original design did not provide for future COA expansion



New COA Structure



New COA Structure (spider diagram)



Notes: 1) Funding Source Indicator (FSI), Itemization of Expense (IOE) and GAAFR will be derived

2) * Procurement contracts to use Procure-to-Pay (P2P) contracts functionality. Contracts requiring full set of financials to use Projects functionality



Ensuring Data Integrity



Methods for Ensuring Data Integrity

- Use of SetID functionality to narrow choices to appropriate Agency value sets
- Combination Editing establishes valid data relationships for entry
- Effective tree design creating easy to maintain rollup structures
- Maintain recognizable fields where possible (i.e. BE, Fund)
- Avoid use of “surrogate” numbering schema, field attributes
- Use the ‘fewest possible ChartFields’ while meeting State of Florida requirements
- Minimize field transformation between business systems and Aspire



SetID Use

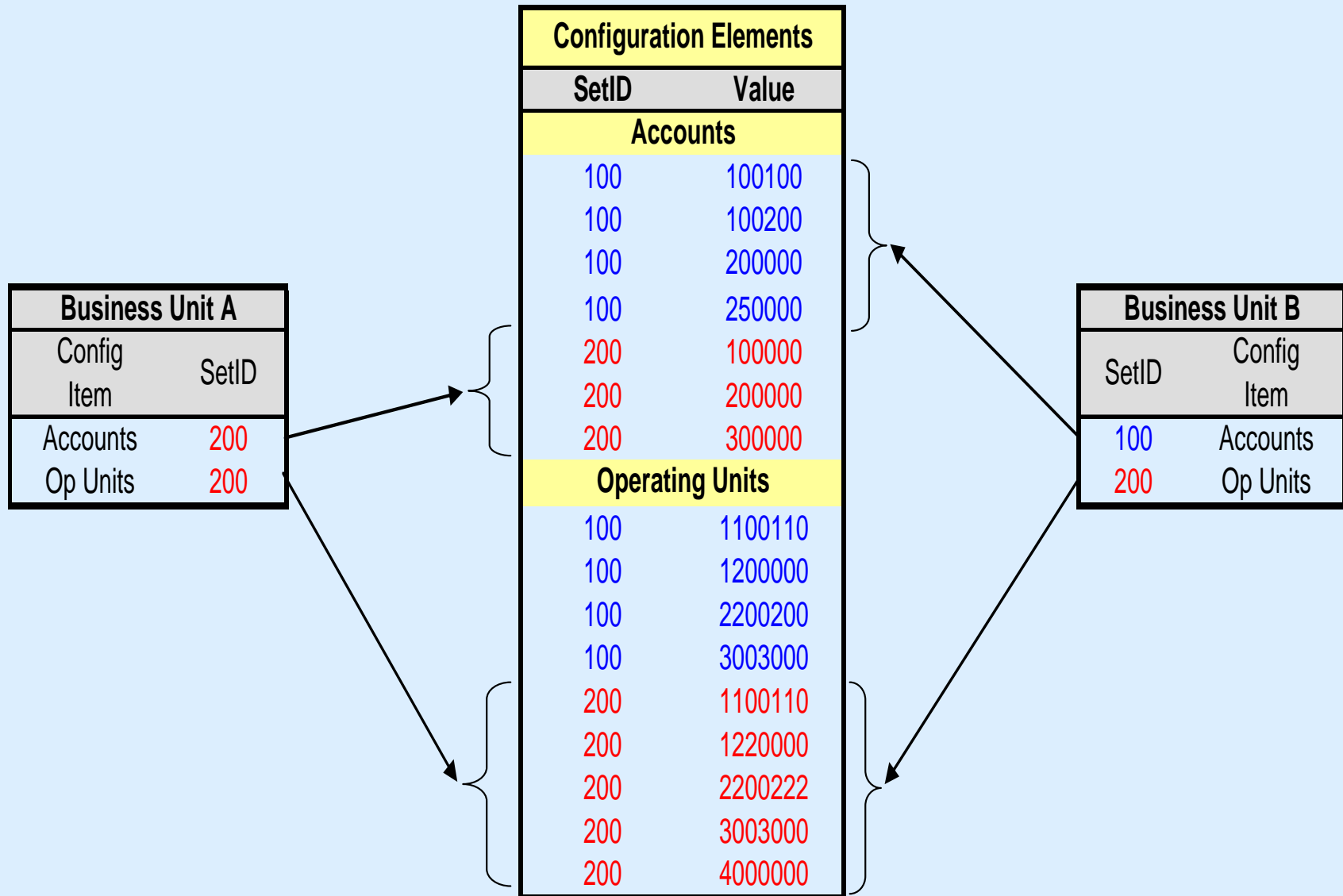


SetID Definition

- An identification code that represents a set of control table information or TableSets. A TableSet is a group of tables (records) necessary to define Enterprise or Agency structure and processing options.
- SetID configuration determines which set of values each Business Unit uses for each type of *configuration element*
 - ✓ Account ChartField
 - ✓ Fund ChartField
 - ✓ Customers
 - ✓ Vendors
 - ✓ etc. (308 total items).
- SetID's can be shared by all enterprise Business Units, several Business Units or by a single Business Unit.
- Business Units that use the same SetID for a specific type of configuration element share the same set of values for that element.



SetID Illustration





Aspire Trees



Aspire Tree Definition

- Trees represent hierarchical structures that represent a group of summarization rules for a particular database field.
- Summarization rules depicted in a tree apply to the DETAIL values of a particular field and are user definable.
- Once defined, trees can be used in a variety of ways:
 - ✓ Reporting
 - ✓ Summary ledgers
 - ✓ Security
 - ✓ Combination Editing
- There are two main types of tree structures:
 - ✓ Detail tree structures – are either detail-value (summer) or node-oriented (winter) trees
 - ✓ Summary tree structures – used primarily for PS nVision reporting using summary ledgers.
- Tree Usage in conjunction with SetID.



Fund to Budget Entity Tree Example

SetID: 64000 **Last Audit:** Valid Tree
Effective Date: 01/01/1901 **Status:** Active
Tree Name: 64000_FUND_BE BE classified by Fund

[Close](#) [Display Options](#) [Print Format](#)

[Collapse All](#) | [Expand All](#) [Find](#) First Page 17 of 17 Last Page

- ALL -**
 - 2141 - COUNTY HEALTH DEPT TF
 - 2141000 - COUNTY HEALTH DEPT TF
 - [64200700] - CTY HLTH LOC HLTH NEED
 - 2339 - GRANTS AND DONATIONS TF
 - 2339060 - GRANTS AND DONATIONS TF
 - [64200300] - FAMILY HLTH OUTPATNT/NUTRN
 - [64200400] - INFECTIOUS DISEASE CNTRL
 - [64200600] - ENVIRONMENTAL HEALTH SVCS
 - [64200800] - SW PUBLIC HLTH SUPPORT SVC
 - [64300100] - CHILD SPECL HLTH CARE
 - [64400200] - COMMUNITY HEALTH RES
 - 2352 - MEDICAL QLTY ASSURANCE TF
 - 2352001 - MEDICAL QLTY ASSURANCE TF
 - [64100200] - ADMINISTRATIVE SUPPORT
 - [64100400] - INFORMATION TECHNOLOGY
 - [64400100] - MEDICAL QUALITY ASSURANCE



Combination Editing



Combination Editing Definition

- Combination editing compares keyed ChartField combinations to user defined rules that establish valid combinations.
- ChartField combinations can be set up for either valid or invalid combination checking dependant on which is more efficient at excluding unwanted entries.
- Combination definitions establish the ChartFields involved in combination editing.
- Combination rules define the valid ChartField values for the combination definitions.
- Combination groups require that all rules within a group must have the same definition.
- Combinations that match the rules and definitions can be marked as valid or invalid based on processing efficiency.
- Combinations that do not match rules and definitions are invalid and can not be marked as valid.



Combination Rule Definition

Rule Definition

ChartField Combinations

SetID: 64000

Combination Rule: FUND_BE

Description:

Budget Entity to Fund

Long Description:

Empty text area with scrollbars

Effective Date From:

01/01/1901



Open Effective Date To

Effective Date To:

01/01/2099

Combination Definition:

FUND_BE

Effective Date for Prompting:

12/12/2005

Non-Anchor ChartField Option

ChartField

Value Required

Budget Entity



ChartField Combination Rules

Rule Definition

ChartField Combinations

SetID: 64000

Combination Rule: FUND_BE

Descr: Budget Entity to Fund

Anchor ChartFields

Find | View All | First 1 of 3 Last

ChartField:

Tree:

Level:



Seq: 1

How Specified

- Selected Detail Values
- Selected Tree Nodes

Chartfield values / Tree nodes

Find | View All | First 1 of 1 Last

Node/Value

2141000



Non-Anchor ChartFields

Find | View All | First 1 of 1 Last

Non-Anchor ChartField:

Tree:



Level:



Budget Entity

How Specified

- Selected Detail Values
- Selected Tree Nodes

Chartfield values / Tree nodes

Find | View All | First 1 of 1 Last

Node/Value

2141000





ChartField Combination Rules, cont.

Rule Definition

ChartField Combinations

SetID: 64000

Combination Rule: FUND_BE

Descr: Budget Entity to Fund

Anchor ChartFields

Find | View All | First 2 of 3 Last

ChartField:

Tree:

Level:



Fund/CC

Seq: 2

How Specified

- Selected Detail Values
- Selected Tree Nodes

Chartfield values / Tree nodes

Find | View All | First 1 of 1 Last

Node/Value

2339060



Non-Anchor ChartFields

Find | View All | First 1 of 1 Last

Non-Anchor ChartField:

Tree:

64000_FUND_BE



Level:



Budget Entity

How Specified

- Selected Detail Values
- Selected Tree Nodes

Chartfield values / Tree nodes

Find | View All | First 1 of 1 Last

Node/Value

2339060





ChartField Combination Rules, cont.

Rule Definition | **ChartField Combinations**

SetID: 64000 **Combination Rule:** FUND_BE **Descr:** Budget Entity to Fund

Anchor ChartFields [Find](#) | [View All](#) First ◀ 3 of 3 ▶ Last

ChartField: Tree: Level: + -

Fund/CC **Seq:** 3

How Specified

- Selected Detail Values
- Selected Tree Nodes

Chartfield values / Tree nodes [Find](#) | [View All](#) | First ◀ 1 of 1 ▶ Last

Node/Value

+ -

Non-Anchor ChartFields [Find](#) | [View All](#) First ◀ 1 of 1 ▶ Last

Non-Anchor ChartField: Tree: + - Level: + -

Budget Entity

How Specified

- Selected Detail Values
- Selected Tree Nodes

Chartfield values / Tree nodes [Find](#) | [View All](#) | First ◀ 1 of 1 ▶ Last

Node/Value

+ -



Questions & Answers