



# *Putting the Power of Configuration in the hands of the users*

*ODTUG Kscope11 Abstract #: 237549  
June 26-30, 2011*

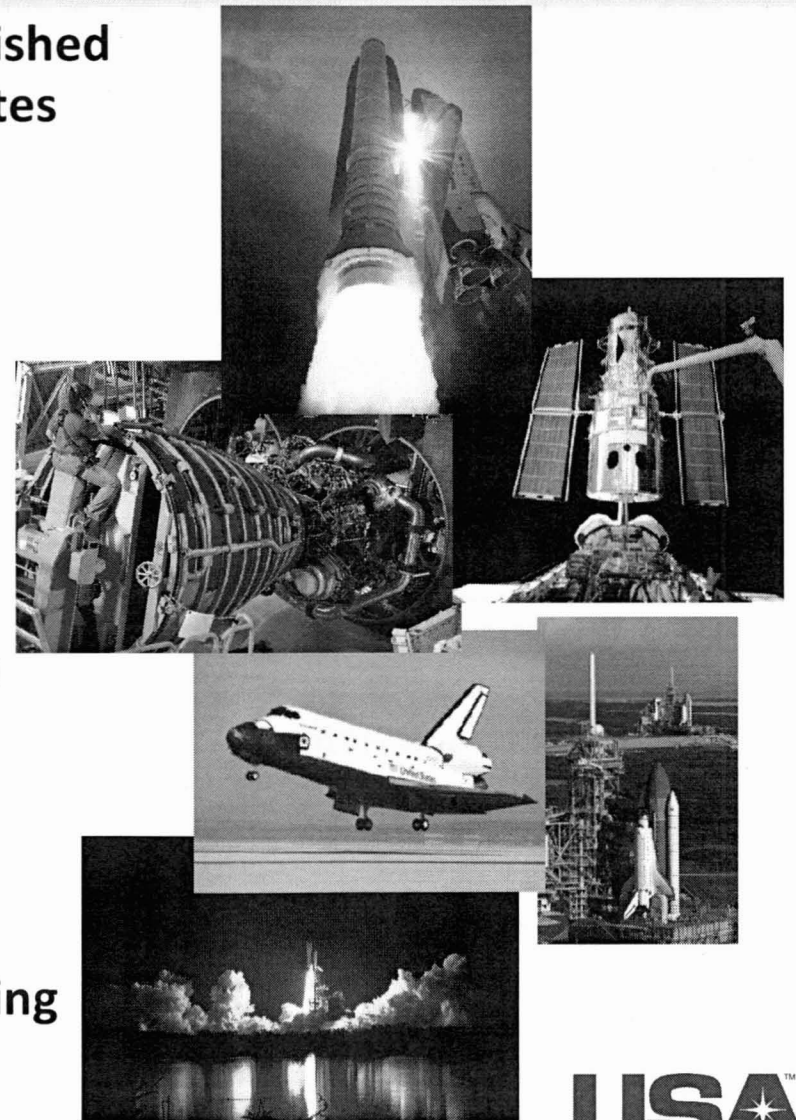
*MaryJo Al-Shihabi  
Mark Brown  
Marianne Rigolini*

Copyright © 2011 by United Space Alliance. All rights reserved. The Government retains a paid-up, nonexclusive, irrevocable and exclusive license in this work to reproduce, publish, perform publicly and display publicly, by or on behalf of the Government.

**USA**<sup>TM</sup>  
United Space Alliance

# United Space Alliance, LLC (USA)

- Headquartered in Houston, USA was established in 1995 with 8,800 employees working at sites in Texas, Florida, and Alabama
- Our mission is to provide safe, high-quality, best-value space operations, services and technologies to our customers
- World leader in Space Operations
  - Space Flight Mission Operations
  - Space Systems Integrated Logistics
  - Space Systems Integration and Program Management
  - Space Systems Ground Operations and Processing
  - Space Systems Flight Software
- Successfully completed end-to-end processing for 56 Space Shuttle Missions



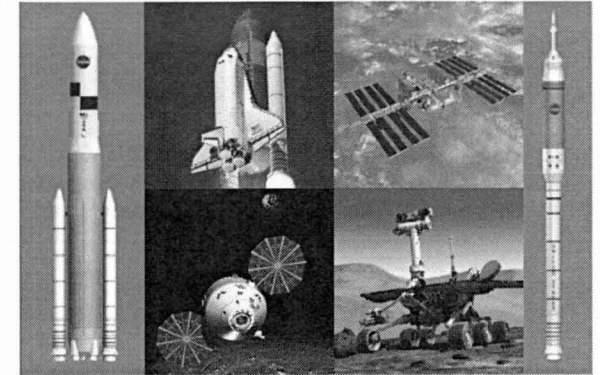
# Background

- **USA's Space Flight Operations Contract represented the integration of 12 NASA contracts that were previously managed by dozens of independent and diverse contractors**
  - **Goal was to reduce the overall cost of human space flight while maintaining the most demanding standards for safety and mission success**
- **In support of this goal, a project team was chartered to replace 18 legacy Space Shuttle nonconformance processes and systems with one fully integrated system**
  - **Problem Reporting and Corrective Action (PRACA) processes provide a closed-loop system for the identification, disposition, resolution, closure, and reporting of all Space Shuttle hardware/software problems**
  - **PRACA processes are integrated throughout the Space Shuttle organizational processes and are critical to assuring a safe and successful program**

# Challenge

## ➤ Primary Project Objectives

- Develop a fully integrated system that provides automated workflow with electronic signatures
- Support multiple NASA programs and contracts with a single “system” architecture
- Define standard processes, implement best practices, and minimize process variations
- Reduce system operating costs and supportability risks

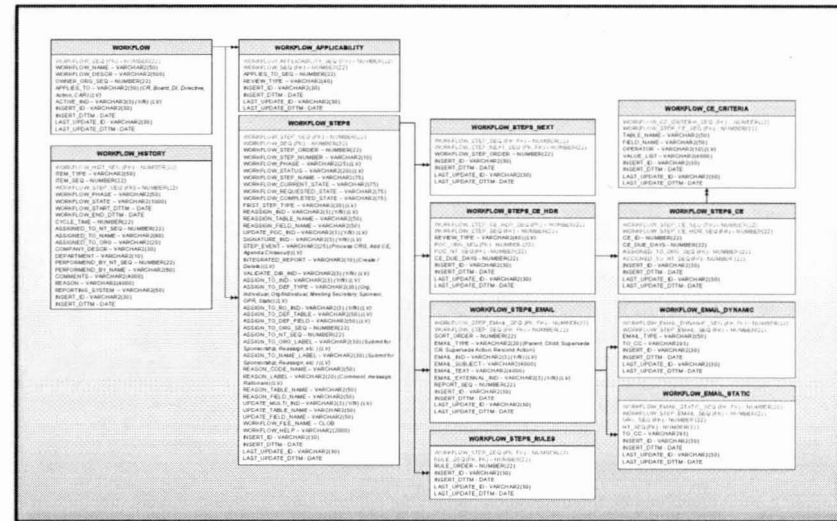


**How do you integrate multiple diverse systems when the business and system requirements are not approved or even fully defined?**

**How do you get multiple customers to buy-in to a common process and system when historically they had complete independence and control?**

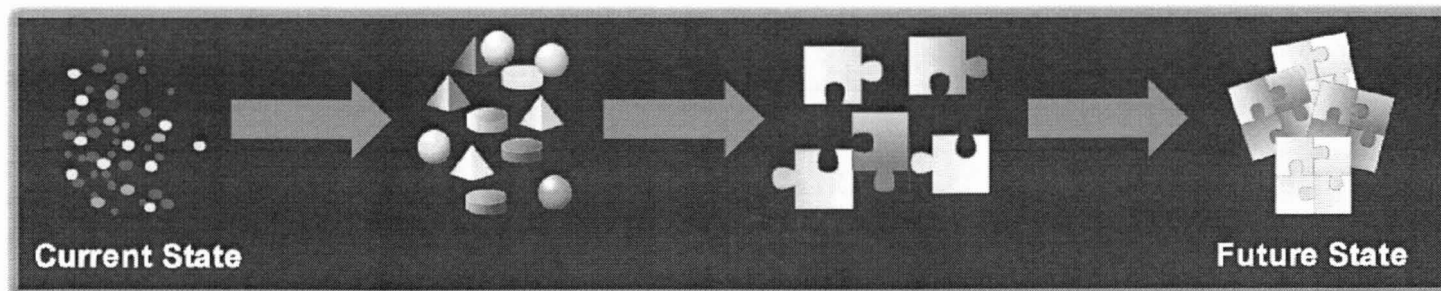
# Solution – Design for Flexibility

- Develop a configurable system that can be tailored by the user community as new business processes are defined and customer buy-in evolves
- A configurable system allows many functions to be data driven, allowing users to configure and implement changes without software code impacts
  - Data Integrity Drivers – Value Lists, Validation Rules, and Security
  - Automated Workflow, Electronic Signatures, and E-Mail Notifications
  - Data Entry Screens
  - Change Tracking and Audit
  - In Box and Assignment Lists
  - Page and Field Level Help
  - System Interfaces
  - Search and Reporting
  - Fail-Over Architecture



# Advantages

- **Configurable systems allow development to proceed without a finalized business process**
- **Highly flexible and robust design that can support Enterprise, Contract, Program, and/or Organizational processing**
- **Users can control process change implementations more efficiently**
- **Software code changes are minimized\eliminated**
  - **Increased reliability and reduced downtime**
  - **Reduced testing and verification**
  - **Reduced system maintenance cost**
- **Reusable Software Code**



# Configuration Design Approach

## ➤ Design for Flexibility and Growth

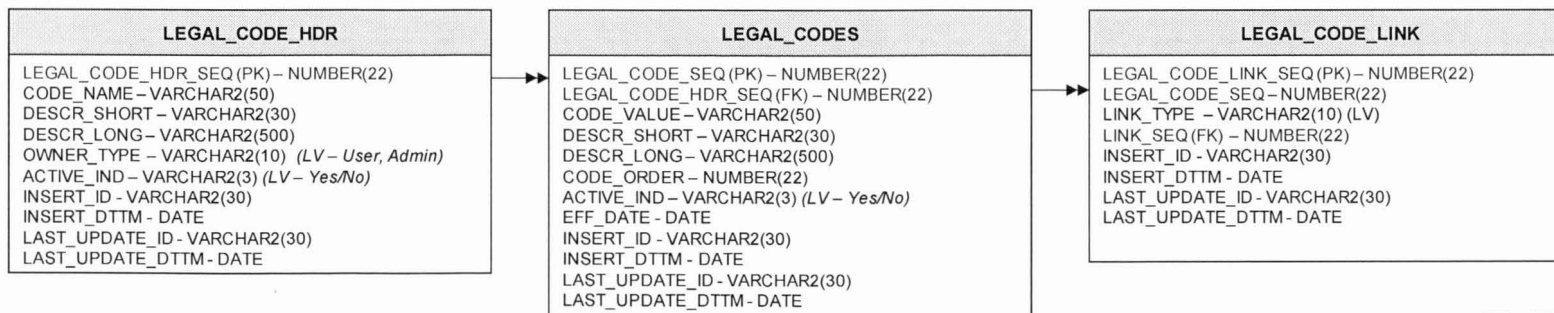
- Determine the primary configuration drivers (i.e. Contract, Organization, and/or Process Variations)
- Evaluate requirements to identify items that would benefit from a configurable design
  - High Risk requirements – details are not fully defined
  - Unique functionality required to support different contracts, organizations or processes
  - Areas where you anticipate change as the business process evolves
  - Reusable functions

## ➤ Consider hard-coding functions that will be applied across the application and flexibility is not required or necessarily desired such as core business requirements

**Cost benefits maximized with  
a Balanced approach**

# Example 1 – LOV Design

- Legal Values (or LOV) represents a list of valid values a user may select from a pre-defined list
- LOV maintained in a central table and dynamically derived
  - LEGAL\_CODE\_HDR– Groups the different legal value lists
    - CODE\_NAME - Equates to the data field on a screen that is linked to the LOV
    - OWNER\_TYPE – Defines who can update\maintain LOV
  - LEGAL\_CODES – Contain the actual individual values
    - ACTIVE\_IND – Allows the user to deactivate a value without removing it from the list
    - EFF\_DATE – Identifies when the value became effective
  - LEGAL\_CODE\_LINK – Allows values to be filtered based on another value\field
    - LINK\_TYPE and LINK\_SEQ – Identifies the source field for the sub-filtering





# Example 1 – LOV Configuration

Order	Value	Value Short Description	Value Active	Owner Type	Applies To Forms	Applies To Boards
10	Mission	Space Shuttle Mission Number	Yes	Form Type	SSP CR SSP Closeout CR	
20	Equipment	Equipment end-item or hardware serial number	Yes	Form Type	SSP CR SSP Closeout CR	
30	Facility	Manufacturing/processing building or work area	Yes	Form Type	SSP CR SSP Closeout CR	
40	Non-Flight Specific	Not associated with any specific flight, end-item, or facility	Yes	Form Type	SSP CR SSP Closeout CR	
50	Not Applicable		Yes	Form Type	SSP Closeout CR	

Order	Value	Value Short Description	Value Active	Owner Type	Applies To Forms	Applies To Boards
10	Assigned To		Yes	Workflow		
20	CE Evaluator		Yes	Workflow		
30	CE Org Evaluators		Yes	Workflow		
40	CE Approver		Yes	Workflow		
50	CE POC		Yes	Workflow		
60	CR Initiator		Yes	Workflow		
70	CR OPP		Yes	Workflow		

Code Name: CATEGORY

Code Name: CATEGORY    Type: Form Type    Active: Yes

Short Description: Effectivity category  
20 of 30

Long Description: Defines the mission, equipment, facility, etc affected and/or impacted by the change request  
92 of 500

Audit Information  
Inserted By/On: USAFL/FLEMMIMJ 06/04/2009 07:35:48 AM  
Last Updated By/On: USATX/BROWNMA 04/25/2011 02:50:38 PM

Code Name: Values

Create/Update Legal Value    Add Another Value    Cancel    Save

Value: Mission    Order: 10    Active: Yes    Effective: 06/04/2009

Short Description: Space Shuttle Mission Number  
28 of 75

Long Description: Defines Effectivity by mission or STS cycle  
43 of 500

Applies to Forms

Available: SSP ICB CR

Selected: SSP CR  
SSP Closeout CR

Values are filtered by Form Type

Audit Information  
Inserted By/On: USAFL/FLEMMIMJ 06/04/2009 07:36:05 AM  
Last Updated By/On: USATX/BROWNMA 04/25/2011 02:42:58 PM

# Example 1 – APEX LOV Definition

**List of Values**

Named LOV:

Display Extra Values:

Display Null Value:

Null Display Value:  Null Return Value:

Cascading LOV Parent Item(s):

List of values definition

```
SELECT CODE_VALUE "D", CODE_VALUE "R"  
FROM CMST_LEGAL_CODES_ACTIVE_VW V, cmst_legal_code_link L WHERE CODE_NAME  
= 'CATEGORY' and v.legal_code_seq=l.legal_code_seq  
and L.link_type='CR_FORM' and L.link_seq=:P2004_CR_FORM_SEQ  
ORDER BY CODE_ORDER
```

Create or edit static List of Values  
Create Dynamic List of Values  
List of Values Examples

**Change Request CR000790**

Next Step:  Go

SSP CR Page 1 | SSP CR Page 2 | Attachments (1) | Evaluations | CR Processing | Disposition | Workflow | Save Updates

CR Number: CR000790

Space Shuttle Program Change Request

Page 2 of 2

Elements Affected:

- Orbiter
- External Tank
- GSE
- Payloads

Change Impact:

- Performance
- Schedule
- Transition

SAFE

Delta SFR

**Add CR Effectivity - Windows Internet Explorer**

Add Effectivity

Category:  Display Order:

Mission:  And Subs

Comments:

Cancel Create

Add

Screen: 2004 User: USATXBROWNMA Current Build: 0.06.00 3/14/2011 Curator: IRDM: Mark A. Brown

# Example 2 – Validation Rules

- Validation Rules are used to ensure compliance with customer, company, organizational, and system data requirements
- Validation Rule configuration design generates an SQL statement that is evaluated when a Button is selected or a Workflow step is processed
- System will not commit changes until all validation errors are resolved

The screenshot displays a web-based user management interface. A grey arrow points to a validation error message box at the top left, which contains the following text:

- Please enter a Last Name
- Please enter a First Name
- Please select an Organization

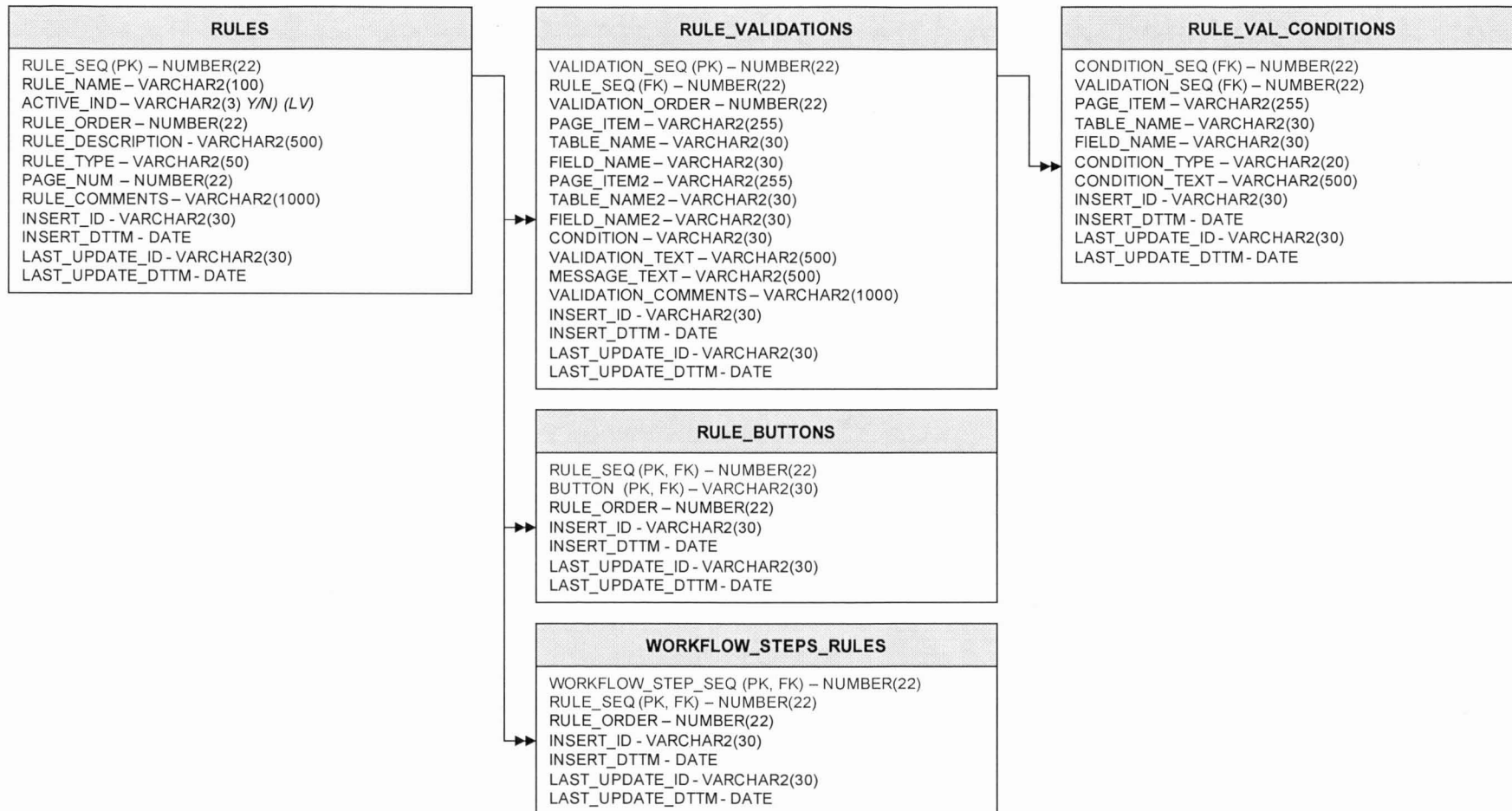
The main interface is titled "Create / Update User" and includes tabs for "User", "Organization", "Security", and "Email Preference". The "User" tab is active. The form contains the following fields and values:

- Active: Yes (dropdown)
- CURF Required: Yes (dropdown)
- CURF Received: No (dropdown)
- Approved: Yes (dropdown)
- User ID: USATXBROWNMA (text field)
- Employee ID: 020102 (text field)
- Name: Brown Mark A (text field)
- Last Name: (empty text field)
- First Name: (empty text field)
- Middle Init: A (text field)
- Email: Mark.A.Brown@usa-spaceops.com (text field)
- Phone: 281/282-6530 (text field)
- Fax Number: 281/282-6704 (text field)
- Company: United Space Alliance, LLC (dropdown menu)
- Organization: - Select - (dropdown menu)
- Department: 62500 (text field)
- Building: HOUSTON006 (text field)
- Grid: N20W1260 (text field)
- Mail Code: USH-701A (text field)
- Floor: 2 (text field)
- Room: (empty text field)

Below the main form is a "User Preference" section with the following settings:

- User Homepage: Inbox (dropdown menu)
- Inbox For View: Brown Mark A (dropdown menu)
- Default Tab: Change Requests (dropdown menu)
- Max Items: 250 (dropdown menu)

# Example 2 – Validation Rule Design



# Example 2 – Validation Rule Configuration

**Rule** **Validations**

Create/Edit Validation Rule Cancel Save Delete

Rule Name: 2-01 Active: Yes Order: 10

Description: Validate User Information Mandatory Data

40 of 500

Rule Type: Admin

Screen: 2 - Create/Update User

Comments:

---

**Applies to Buttons**

Available:	Selected:
Add Another User (ADD)	Create (CREATE)
Cancel (CANCEL)	Save (SAVE)
Retrieve User Info (ACTIVE_DIRECTORY_UPDATE)	

---

**Audit Information**

Inserted By/On: USATXBROWNMA 04/06/2011 01:04:11 PM

Last Updated By/On: USATXBROWNMA 04/21/2011 02:27:47 PM

**Validation Rules** Add Validation Copy Validation(s) Delete Validation(s) Save

Select	Order	Item	Condition	Validation Text	Conditions
<input type="checkbox"/>	10	User ID (P2_NT_ID)	is not null		Show(0)
<input type="checkbox"/>	20	Name (P2_NAME)	is not null		Hide
<input type="checkbox"/>	30	Last Name (P2_			Show(1)
<input type="checkbox"/>	40	First Name (P2_FR			Show(1)
<input type="checkbox"/>	50	Email (P2_EMAILID)			Show(1)
<input type="checkbox"/>	60	Phone (P2_PHONE			Show(1)
<input type="checkbox"/>	70	Company (P2_COM			Show(1)
<input type="checkbox"/>	80	Organization (P2_OI			Show(1)
<input type="checkbox"/>	90	User Homepage (P2_			Show(1)
<input type="checkbox"/>	100	Inbox For View (P2_			Show(1)
<input type="checkbox"/>	110	Default Tab (P2_DEF			Show(1)
<input type="checkbox"/>	120	Max Items (P2_INRC			Show(1)
<input type="checkbox"/>	130	Dashboard For View			Show(1)
<input type="checkbox"/>	140	Search Type (P2_SE			Show(1)
<input type="checkbox"/>	150	Max Items per Page (P2_SEARCH_ITEMS)	is not null		Show(1)
<input type="checkbox"/>	160	Advanced Search (P2_ADVANCED_SEARCH_LIST)	is not null		Show(1)

**Rule Validations - Windows Internet Explorer**

Create/Edit Validation Cancel Apply Changes

Order: 20

Page Item: Name (P2\_NAME)

Condition: Item is not null

Validation Text:

Message Text: Please enter a User's Name

Comments:

---

**Audit Information**

Inserted By/On: USATXBROWNMA 04/06/2011 01:04:45PM

Last Updated By/On: USATXBROWNMA 04/06/2011 03:05:58PM

Screen: 49 Current Build: 0.06.00 3/14/2011

User: USATXBROWNMA Curator: RDM: Mark A. Brown

1 - 16

---

**Conditions** Add Condition Copy Condition(s) Delete Condition(s)

Select	Item	Condition	Condition Text
<input type="checkbox"/>	Active (P2_ACTIVE_INDICATOR)	is equal to	Yes

1 - 1

Conditions may be applied to a Validation Rule to limit the applicability of the rule

# Example 2 – APEX Validation Function

Page Validation: 1 of 1 Name: Get Screen Error Messages Cancel Delete Apply Changes

Show All Validation Error Message Conditions Security Configuration Comments

**Validation**

Page: 2 Create/Update User

\* Name

\* Sequence

Type

\* Validation Expression 1

```
BEGIN
RETURN CMST_VALIDATION_PACK.GET_ERROR_MESSAGES(null, :APP_PAGE_ID, :REQUEST,null, null);
END;
```

Please enter a Last Name  
Please enter a First Name  
Please select an Organization

User Organization Security Email Preference

Create / Update User Add Another User Cancel Save

Active: Yes CURF Required: Yes CURF Received: No Approved: Yes

User ID: USATXBROWNIA Retrieve User Info

Employee ID: 020102

Name: Brown Mark A

Last Name:  First Name:  Middle Init:

Email:

Phone:  000/000-0000

Fax Number:  000/000-0000

Company:

Organization:

Department:  Mail Code:

Building:  Floor:

Grid:  Room:

User Preference

User Homepage:

Inbox For View:  Default Tab:  Max Items:

- Function (GET\_ERROR\_MESSAGES) processes each configured validation rule and identifies errors
- Errors are combined into a single message and displayed on the corresponding screen header
- System will not commit changes until all validation errors are resolved

# Example 2 – APEX Validation Behind the Scenes

```
FOR validations IN (SELECT V.* FROM CMST_RULES R, CMST_RULE_VALIDATIONS V, CMST_RULE_BUTTONS RB
WHERE R.RULE_SEQ = RB.RULE_SEQ AND R.PAGE_NUM = p_page_number AND RB.BUTTON = p_request AND R.RULE_SEQ = V.RULE_SEQ
AND V.PAGE_ITEM IS NOT NULL AND NVL(R.ACTIVE_IND,'No') = 'Yes' AND (v_cr_form_seq IS NULL OR v_cr_form_seq
IN (SELECT CR_FORM_SEQ FROM CMST_RULE_CR_FORMS WHERE RULE_SEQ = R.RULE_SEQ)) ORDER BY R.RULE_ORDER, V.VALIDATION_ORDER)
```

LOOP

```
v_run_val := TRUE;
```

```
--CHECK ALL CONDITIONS TO SEE IF THE VALIDATION SHOULD RUN
```

```
FOR condition IN (SELECT PAGE_ITEM, TABLE_NAME, FIELD_NAME, CONDITION_TYPE, CONDITION_TEXT
FROM CMST_RULE_VAL_CONDITIONS
WHERE VALIDATION_SEQ = validations.VALIDATION_SEQ
AND PAGE_ITEM IS NOT NULL)
```

LOOP

```
v_sql := 'SELECT 1 FROM DUAL WHERE ' || CMST_SQL_PACK.GET_SQL_COMPARE_TEXT(GET_DATATYPE(condition.CONDITION_TEXT)
, GET_PAGE_ITEM_QUERY_TEXT(condition.PAGE_ITEM), condition.CONDITION_TYPE, condition.CONDITION_TEXT);
```

```
BEGIN
```

```
execute immediate v_sql INTO v_count;
```

```
EXCEPTION
```

```
WHEN NO_DATA_FOUND THEN v_count := 0;
```

```
WHEN OTHERS THEN v_count := 1;
```

```
END;
```

```
IF v_count = 0 THEN v_run_val := FALSE;
```

```
EXIT; --NO REASON TO GO ON, EXIT THE LOOP AND DON'T RUN THE VALIDATION
```

```
END IF;
```

```
END LOOP;
```

```
IF v_run_val THEN
```

```
IF UPPER(validations.VALIDATION_TEXT) = 'SYSDATE' THEN
```

```
v_sql := 'SELECT 1 FROM DUAL WHERE TO_DATE(V('' || validations.PAGE_ITEM || ''),'MM/DD/YYYY') ' || validations.CONDITION ||
' TRUNC(' | validations.VALIDATION_TEXT || ')';
```

```
ELSE
```

```
IF validations.PAGE_ITEM2 IS NOT NULL THEN
```

```
v_sql := 'SELECT 1 FROM DUAL WHERE '
```

```
||
```

```
CMST_SQL_PACK.GET_SQL_COMPARE_TEXT(GET_DATATYPE(validations.VALIDATION_TEXT), GET_PAGE_ITEM_QUERY_TEXT(validations.PAGE_ITEM)
validations.CONDITION, GET_PAGE_ITEM_QUERY_TEXT(validations.PAGE_ITEM2));
```

```
ELSE
```

```
v_sql := 'SELECT 1 FROM DUAL WHERE ' || CMST_SQL_PACK.GET
, GET_PAGE_ITEM_QUERY_TEXT(validations.PAGE_ITEM
```

```
END IF;
```

```
END IF;
```

```
BEGIN
```

```
execute immediate v_sql into v_count;
```

```
EXCEPTION
```

```
WHEN OTHERS THEN v_count := 0;
```

```
END;
```

```
IF v_count = 0 THEN v_errors := v_errors || validations.MESS
```

```
END IF;
```

```
END IF;
```

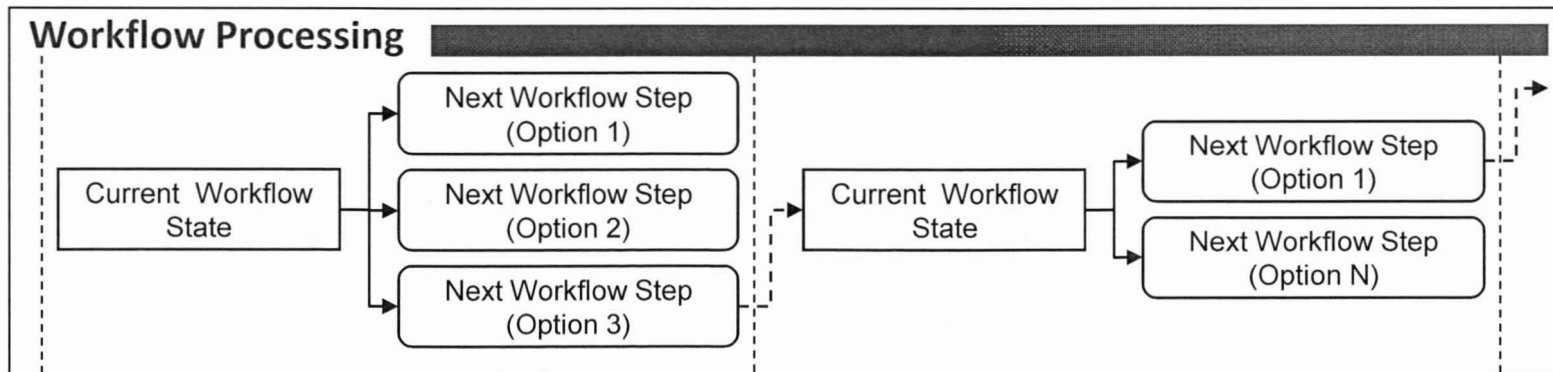
```
END LOOP;
```

**Loop through Rules to generate SQL**

```
SELECT 1 FROM DUAL WHERE V('P2_NT_ID') IS NOT NULL
SELECT 1 FROM DUAL WHERE V('P2_NAME') IS NOT NULL
SELECT 1 FROM DUAL WHERE V('P2_LAST_NAME') IS NOT NULL
SELECT 1 FROM DUAL WHERE V('P2_FIRST_NAME') IS NOT NULL
SELECT 1 FROM DUAL WHERE V('P2_EMAILID') IS NOT NULL
SELECT 1 FROM DUAL WHERE V('P2_PHONE') IS NOT NULL
SELECT 1 FROM DUAL WHERE V('P2_COMPANY_DESCR') IS NOT NULL
```

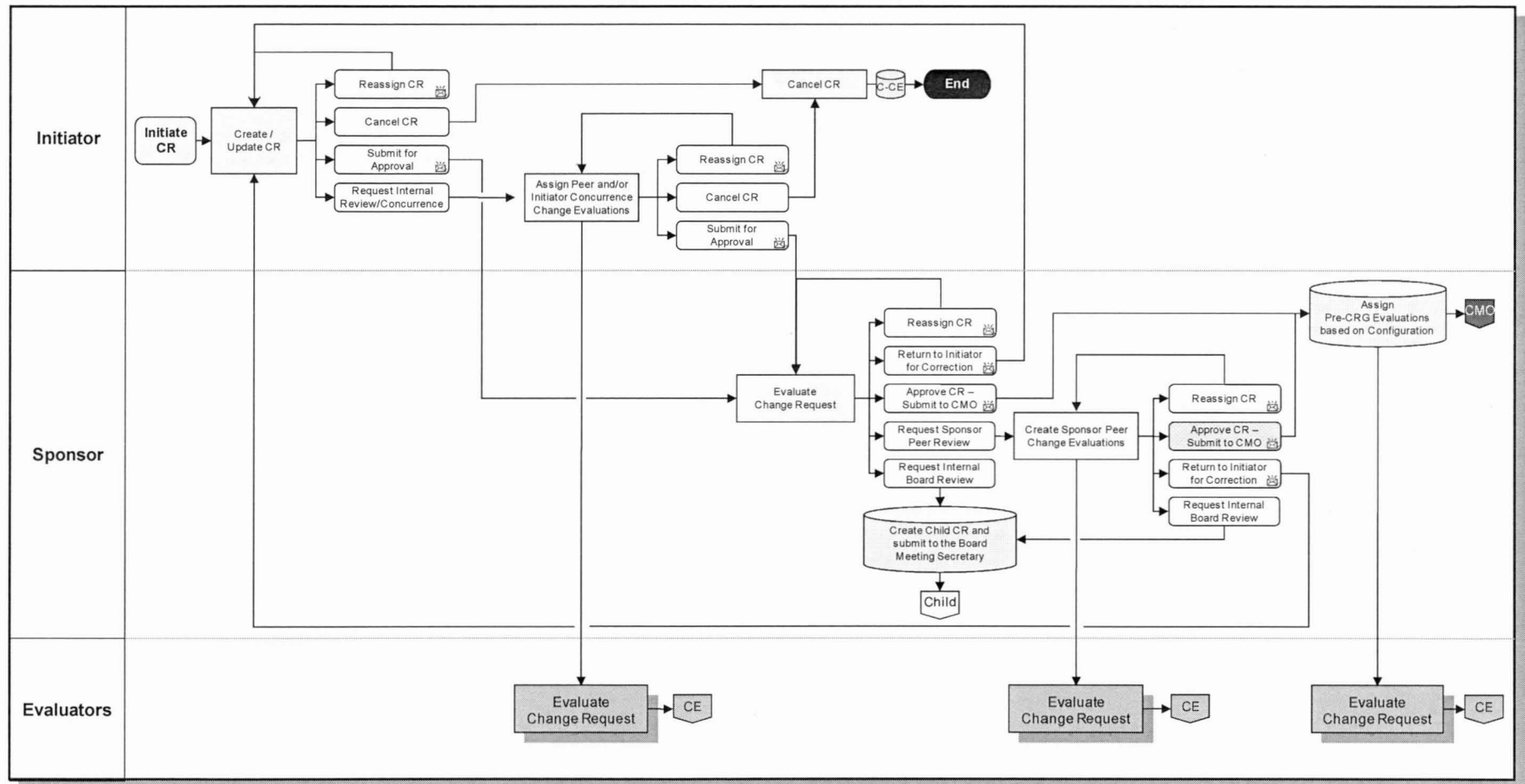
# Example 3 – Automated Workflow

- **Workflow is the automation of a business process where information is passed from one state to another for action in accordance with a set of pre-defined business rules**
  - **Configurable workflow design includes a current workflow step, potential next step(s), assignee information, e-mail notifications, and automated process activities**
  - **Workflow is accomplished by the user selecting the next appropriate step from a list of pre-defined\configured options and the system performing the associated activities**





# Example 3 – Sample Workflow



Current Step

Next Step Options

Step Event Functions

Spawned Workflow

# Example 3 – Workflow User Interface

Change Request CR000773

Next Step: Approve CR - Submit to CMO Go

SSP CR Page 1 SSP CR Page 2 Attachments (3) Evaluations CR Processing Disposition

CR Number  
CR000773

**Space Shuttle Program Closeout  
Change Request**

Initiated By: Brown Mark A Organization: PI TIS Phone: 281/282-6530

Change Title:  
Update NSTS 07700 Volume XX - Orbiter End State Requirements  
61 of 150

Change Proposal/Request Numbers: Add

SSP Documents Affected:

CR000002

NSTN 07700, Volume XX Transition and Re Requirements

Additional Information

Reason: Expedite

Requested Processing:

Password:

Are you sure you would like to Approve CR - Submit to CMO?

Ok Cancel

- Based on the workflow configuration, the user is presented with one or more Next Step options
- System determines the processing requirements and opens a pop-up that is dynamically generated based on the selected Next Step
- System runs the Validation Rules, processes the required updates, and notifies the affected users

# Example 3 – Workflow Design

WORKFLOW
WORKFLOW_SEQ (PK) – NUMBER(22)
WORKFLOW_NAME – VARCHAR2(50)
WORKFLOW_DESCR – VARCHAR2(500)
OWNER_ORG_SEQ – NUMBER(22)
APPLIES_TO – VARCHAR2(30) (LV)
ACTIVE_IND – VARCHAR2(3) (LV – Yes/No)
INSERT_ID – VARCHAR2(30)
INSERT_DTTM – DATE
LAST_UPDATE_ID – VARCHAR2(30)
LAST_UPDATE_DTTM – DATE

WORKFLOW_HISTORY
WORKFLOW_HIST_SEQ (PK) – NUMBER(22)
ITEM_TYPE – VARCHAR2(50)
ITEM_SEQ – NUMBER(22)
WORKFLOW_STEP_SEQ (FK) – NUMBER(22)
WORKFLOW_PHASE – VARCHAR2(50)
WORKFLOW_STATE – VARCHAR2(1000)
WORKFLOW_START_DTTM – DATE
WORKFLOW_END_DTTM – DATE
CYCLE_TIME – NUMBER(22)
ASSIGNED_TO_NT_SEQ – NUMBER(22)
ASSIGNED_TO_NAME – VARCHAR2(60)
ASSIGNED_TO_ORG – VARCHAR2(20)
COMPANY_DESCR – VARCHAR2(30)
DEPARTMENT – VARCHAR2(10)
PERFORMEND_BY_NT_SEQ – NUMBER(22)
PERFORMEND_BY_NAME – VARCHAR2(60)
COMMENTS – VARCHAR2(4000)
REASON – VARCHAR2(4000)
REPORTING_SYSTEM – VARCHAR2(50)
INSERT_ID – VARCHAR2(30)
INSERT_DTTM – DATE

Custom tables/fields can be added to support application unique requirements

WORKFLOW_STEPS
WORKFLOW_STEP_SEQ (PK) – NUMBER(22)
WORKFLOW_SEQ (FK) – NUMBER(22)
WORKFLOW_STEP_ORDER – NUMBER(22)
WORKFLOW_STEP_NUMBER – VARCHAR2(10)
WORKFLOW_PHASE – VARCHAR2(25) (LV)
WORKFLOW_STATUS – VARCHAR2(20) (LV)
WORKFLOW_STEP_NAME – VARCHAR2(75)
WORKFLOW_CURRENT_STATE – VARCHAR2(75)
WORKFLOW_REQUESTED_STATE – VARCHAR2(75)
WORKFLOW_COMPLETED_STATE – VARCHAR2(75)
FIRST_STEP_TYPE – VARCHAR2(20) (LV)
REASSIGN_IND – VARCHAR2(3) (LV – Yes/No)
REASSIGN_TABLE_NAME – VARCHAR2(50)
REASSIGN_FIELD_NAME – VARCHAR2(50)
UPDATE_POC_IND – VARCHAR2(3) (LV – Yes/No)
SIGNATURE_IND – VARCHAR2(3) (LV – Yes/No)
STEP_EVENT – VARCHAR2(25) (LV)
INTEGRATED_REPORT – VARCHAR2(10) (LV)
VALIDATE_DIR_IND – VARCHAR2(3) (LV – Yes/No)
ASSIGN_TO_IND – VARCHAR2(3) (LV – Yes/No)
ASSIGN_TO_DEF_TYPE – VARCHAR2(20) (LV)
ASSIGN_TO_RO_IND – VARCHAR2(3) (LV – Yes/No)
ASSIGN_TO_DEF_TABLE – VARCHAR2(50) (LV)
ASSIGN_TO_DEF_FIELD – VARCHAR2(50) (LV)
ASSIGN_TO_ORG_SEQ – NUMBER(22)
ASSIGN_TO_NT_SEQ – NUMBER(22)
ASSIGN_TO_ORG_LABEL – VARCHAR2(30) (LV)
ASSIGN_TO_NAME_LABEL – VARCHAR2(30) (LV)
REASON_CODE_NAME – VARCHAR2(50)
REASON_LABEL – VARCHAR2(20) (LV)
REASON_TABLE_NAME – VARCHAR2(50)
REASON_FIELD_NAME – VARCHAR2(50)
UPDATE_MULTI_IND – VARCHAR2(3) (LV – Yes/No)
UPDATE_TABLE_NAME – VARCHAR2(50)
UPDATE_FIELD_NAME – VARCHAR2(50)
WORKFLOW_FILE_NAME – CLOB
WORKFLOW_HELP – VARCHAR2(2000)
INSERT_ID – VARCHAR2(30)
INSERT_DTTM – DATE
LAST_UPDATE_ID – VARCHAR2(30)
LAST_UPDATE_DTTM – DATE

WORKFLOW_APPLICABILITY
WORKFLOW_APPLICABILITY_SEQ (PK) – NUMBER(22)
WORKFLOW_SEQ (FK) – NUMBER(22)
APPLIES_TO_SEQ – NUMBER(22)
REVIEW_TYPE – VARCHAR2(40)
INSERT_ID – VARCHAR2(30)
INSERT_DTTM – DATE
LAST_UPDATE_ID – VARCHAR2(30)
LAST_UPDATE_DTTM – DATE

WORKFLOW_STEPS_NEXT
WORKFLOW_STEP_SEQ (PK, FK) – NUMBER(22)
WORKFLOW_STEP_NEXT_SEQ (PK, FK) – NUMBER(22)
WORKFLOW_STEP_ORDER – NUMBER(22)
INSERT_ID – VARCHAR2(30)
INSERT_DTTM – DATE
LAST_UPDATE_ID – VARCHAR2(30)
LAST_UPDATE_DTTM – DATE

WORKFLOW_STEPS_CE_HDR
WORKFLOW_STEP_CE_HDR_SEQ (PK) – NUMBER(22)
WORKFLOW_STEP_SEQ (FK) – NUMBER(22)
REVIEW_TYPE – VARCHAR2(40) (LV)
POC_ORG_SEQ (FK) – NUMBER(22)
POC_NT_SEQ (FK) – NUMBER(22)
CE_DUE_DAYS – NUMBER(22)
INSERT_ID – VARCHAR2(30)
INSERT_DTTM – DATE
LAST_UPDATE_ID – VARCHAR2(30)
LAST_UPDATE_DTTM – DATE

WORKFLOW_STEPS_EMAIL
WORKFLOW_STEP_EMAIL_SEQ (PK, FK) – NUMBER(22)
WORKFLOW_STEP_SEQ (PK, FK) – NUMBER(22)
SORT_ORDER – NUMBER(22)
EMAIL_TYPE – VARCHAR2(20) (LV)
EMAIL_IND – VARCHAR2(3) (LV – Yes/No)
EMAIL_SUBJECT – VARCHAR2(4000)
EMAIL_TEXT – VARCHAR2(4000)
EMAIL_EXTERNAL_IND – VARCHAR2(3) (LV – Yes/No)
REPORT_SEQ – NUMBER(22)
INSERT_ID – VARCHAR2(30)
INSERT_DTTM – DATE
LAST_UPDATE_ID – VARCHAR2(30)
LAST_UPDATE_DTTM – DATE

WORKFLOW_STEPS_RULES
WORKFLOW_STEP_SEQ (PK, FK) – NUMBER(22)
RULE_SEQ (PK, FK) – NUMBER(22)
RULE_ORDER – NUMBER(22)
INSERT_ID – VARCHAR2(30)
INSERT_DTTM – DATE
LAST_UPDATE_ID – VARCHAR2(30)
LAST_UPDATE_DTTM – DATE

WORKFLOW_CE_CRITERIA
WORKFLOW_CE_CRITERIA_SEQ (PK) – NUMBER(22)
WORKFLOW_STEP_CE_SEQ (FK) – NUMBER(22)
TABLE_NAME – VARCHAR2(50)
FIELD_NAME – VARCHAR2(50)
OPERATOR – VARCHAR2(10) (LV)
VALUE_LIST – VARCHAR2(4000)
INSERT_ID – VARCHAR2(30)
INSERT_DTTM – DATE
LAST_UPDATE_ID – VARCHAR2(30)
LAST_UPDATE_DTTM – DATE

WORKFLOW_STEPS_CE
WORKFLOW_STEP_CE_SEQ (PK) – NUMBER(22)
WORKFLOW_STEP_CE_HDR_SEQ (FK) – NUMBER(22)
CE_ID – NUMBER(22)
CE_DUE_DAYS – NUMBER(22)
ASSIGNED_TO_ORG_SEQ (FK) – NUMBER(22)
ASSIGNED_TO_NT_SEQ (FK) – NUMBER(22)
INSERT_ID – VARCHAR2(30)
INSERT_DTTM – DATE
LAST_UPDATE_ID – VARCHAR2(30)
LAST_UPDATE_DTTM – DATE

WORKFLOW_EMAIL_DYNAMIC
WORKFLOW_EMAIL_DYNAMIC_SEQ (PK, FK) – NUMBER(22)
WORKFLOW_STEP_EMAIL_SEQ (FK) – NUMBER(22)
EMAIL_TYPE – VARCHAR2(50)
TO_CC – VARCHAR2(93)
INSERT_ID – VARCHAR2(30)
INSERT_DTTM – DATE
LAST_UPDATE_ID – VARCHAR2(30)
LAST_UPDATE_DTTM – DATE

WORKFLOW_EMAIL_STATIC
WORKFLOW_EMAIL_STATIC_SEQ (PK, FK) – NUMBER(22)
WORKFLOW_STEP_EMAIL_SEQ (FK) – NUMBER(22)
ORG_SEQ (FK) – NUMBER(22)
NT_SEQ (FK) – NUMBER(22)
TO_CC – VARCHAR2(93)
INSERT_ID – VARCHAR2(30)
INSERT_DTTM – DATE
LAST_UPDATE_ID – VARCHAR2(30)
LAST_UPDATE_DTTM – DATE

Application specific tables

# Example 3 – Configuration Design

**Description** / **Workflow Steps**

Create / Edit Workflow Cancel Save Copy Delete

Workflow Name:  Active:  Yes

Description:

50 of 500

Owner Organization:  Applies To:

**Applies To Forms**

Available:

Selected:

**Audit Information**

Inserted: USAFLHARTJR1 07/08/2009 08:07:05 AM  
Last Updated: USATXBROWNMA 04/28/2011 10:15:56 AM

**Description** / **Workflow Steps**

Go Actions Reorder Copy Add Step Delete Save

Select	Order	Step	Phase	Current State	Requested State	Step Event	Step Type
<input type="checkbox"/>	10	1	Initiation	Initiating Change Proposal	Initiate Change Proposal		Main Workflow
<input type="checkbox"/>	20	2	Internal Review	In Internal Review	Request Internal Review / Concurrence		
<input type="checkbox"/>	30	3	Sponsorship	In Sponsor Approval	Submit for Approval		
<input type="checkbox"/>	40	4	Sponsor Review	In Sponsor Peer Review	Request Sponsor Peer Review		
<input type="checkbox"/>	50	5	Sponsorship	In Internal Board Review	Request Internal Board Review	Create Child CR	
<input type="checkbox"/>	60	6	CMO Processing	Processing Change Proposal	Approve CR - Submit to CMO	Add CE	
<input type="checkbox"/>	70	6.5	Board Scheduling	Scheduling Child CR	N/A - Child CR First Step		Child CR
<input type="checkbox"/>	80	7	Initiation	Returned for Update	Return to Initiator for Correction		
<input type="checkbox"/>	90	10	Disposition	Processing Outside of Board	Process Outside of Board	Process CR	
<input type="checkbox"/>	100	10.5	In Board Review	Processing Outside of Board	Process Outside of Board	Process CR	Process Child CR OSB
<input type="checkbox"/>	110	11	Disposition	In Board Review	Schedule to Board	Process CR	
<input type="checkbox"/>	120	11.5	In Board Review	Child CR Scheduled for Board Review	Schedule to Board	Process CR	Process Child CR
<input type="checkbox"/>	130	7	Rework	Returned for Update	Return to Initiator for Correction		
<input type="checkbox"/>	140	12	Initiation	Updating per CMO Comments	Update CR - Remove Sponsor's Approval	Reset Signatures	
<input type="checkbox"/>	150	13	CMO Processing	Resubmitted for Processing	Return to CMO		
<input type="checkbox"/>	160	999	Closure	Change Proposal Cancelled	Cancel Change Proposal		
<input type="checkbox"/>	170	999	Closure	Reject Change Proposal	Reject / Close CR	Reject CR	
<input type="checkbox"/>	180	999.5	Closure	Review Complete	Review Complete - Close Child CR	Close Child CR OSB	
<input type="checkbox"/>	190	999.5	Closure	CR Superseded - Review for Closure	Parent CR Superseded - Close Child CR	Supersede CR	
<input type="checkbox"/>	200	999.5	Closure	CR Withdrawn - Review for Closure	Parent CR Closed - Withdraw Close CR	Withdraw CR	
<input type="checkbox"/>	210	9999.0	Initiation	Reassigning Task	Reassign Task	Reassign	

1 - 21

- Workflow header identifies the type of workflow and applicability
- Workflow steps include the specific processing instructions for each step
  - Workflow Assignments
  - Step Event Processing
  - Dynamic Field Updates
  - E-Mail Notifications
  - Report Generation

# Example 3 – Configuration Design

Workflow Step   Next Steps   Add CEs   Data Validation   E-Mail

Create/Edit Workflow Step   Copy   Delete   Cancel   Save

Step Name:

Workflow Phase:    Workflow Status:

Step Number:    Sort Order:    Step Type:

Current State:

Requested State:

Completed State:

Step Event:

Printable Version:

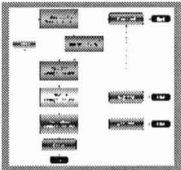
Include Reassign Option?:    Update CE POC?:

Update User in Table:

Reassign Field Name:

E-Signature Required?:

Step File Name:



Assign To Information

Assignment Type:

Display Assigned To:

Read Only:

Table:

Field:

Organization:

Individual:

Assigned To Org Label:

Assigned To Name Label:

- Step Events represent PL\SQL procedures that run specific functions when the workflow step is processed
- Assigned To Information can be static (read-only) or dynamically derived
- Help information for the Next Step options are displayed from the Workflow label

Response Comments / Rationale

Legal Value Code Name:

Message Label:

Load Response into Table:

Load Response into Field:

Workflow Completion Updates (Applied when Step is Selected)

Update Multiple Fields:

Update Table:

Update Field:

Workflow Step Help Description

Source:

The CR is approved by the sponsor and submitted to the Configuration Management Office for community review and disposition. An automated e-mail notification is sent to the CMO office and a task will appear in their inbox until the CR is processed.

Additional help is located in the CMST Project Website: <https://usa1.usa-spaceops.com/>

Audit Information

Inserted: USATXBROWNMA07/10/2009 10:09:13AM

Last Updated: USATXBROWNMA04/28/2011 10:44:20AM

# Example 3 – Sample Step Events

```
-- Step Event
CASE NVL(p_workflow_step.STEP_EVENT, 'NULL')
  WHEN 'Add CE' THEN
    ADD_CE_FROM_WORKFLOW(
      p_cr_processing_seq => p_cr_processing_seq
    ,p_workflow_step_seq => p_workflow_step.WORKFLOW_STEP_SEQ
    ,p_poc_org => v_assign_to_org
    ,p_poc_user => v_assign_to_user);

  WHEN 'Process CR' THEN
    UPDATE CMST_CR_PROCESSING
      SET CRG_DATE = SYSDATE,
          CRG_CHAIR_NT_SEQ = p_user.NT_SEQ,
          CRG_CHAIR_NAME = p_user.NAME
    WHERE CR_PROCESSING_SEQ = p_cr_processing_seq;
    IF v_cr.PROCESSED_OSB_IND = 'Yes' THEN
      DELETE FROM CMST_CR_CRG
        WHERE CR_PROCESSING_SEQ = p_cr_processing_seq;
    ELSE
      FOR crg IN (SELECT CR_CRG_SEQ
                  FROM CMST_CR_CRG
                  WHERE CR_PROCESSING_SEQ = p_cr_processing_seq)
      LOOP
        CMST_AGENDA_PACK.PROCESS_CRG(
          p_cr_crg_seq => crg.CR_CRG_SEQ
        ,p_cr_seq      => v_cr.CR_SEQ
        ,p_cr_processing_seq => p_cr_processing_seq);
      END LOOP;
    END IF;

  WHEN 'Reset Signatures' THEN
    UPDATE CMST_CR
      SET SPONSOR_ORG_SEQ = NULL,
          SPONSOR_ORG = NULL,
          SPONSOR_ORG_DESCR = NULL,
          SPONSOR_NT_SEQ = NULL,
          SPONSOR_NAME = NULL,
          SPONSOR_COMPANY = NULL,
          SPONSOR_DTTM = NULL
    WHERE CR_SEQ = v_cr.CR_SEQ;
```

**“Add CE” creates new Change Evaluations based on the Workflow CE configuration**

**“Process CR” updates the Change Review Group Information and adds the CR to the Review Board Agenda**

**“Reset Signatures” removes the applied Signature Information**

# Example 3 – Configuration Design

Workflow Step | Next Steps | Add CEs | Data Validation | E-Mail

Next Steps Modify Save

Order	Step Name	Requested State
10	Schedule to Board	Schedule to Board
20	Processing Outside of Board	Process Outside of Board
30	Return for Rework (from CMO)	Return to Initiator for Correction
40	Reject Change Proposal	Reject / Close CR

1 - 4

Previous Steps

Step Name	Requested State
Request Internal Board Review	Request Internal Board Review
Request Sponsor Peer Review	Request Sponsor Peer Review
Sponsor Change Proposal	Submit for Approval

1 - 3

**Workflow steps are linked together to create an end-to-end integrated process**

**Linked steps can spawn other workflow sub-processes**

Workflow Step | Next Steps | Add CEs | Data Validation | E-Mail

CE Reviews Add Review Delete Save

Review Type: CRG Pre-Review Add CE

Select	Order	Assigned To	Due Day(S)	Conditions	Updated
<input type="checkbox"/>	10	USA.MJ	1	Show(1)	03/18/2010 Flemming Melody J

**Automatically generate new Tasks**

Workflow Step | Next Steps | Add CEs | Data Validation | E-Mail

Data Validation Rules Modify Save

Order	Rule Name	Description
10	CR Sponsorship	Validate Mandatory data exists to Sponsor \ Submit CR
20	Submit CR to CMO	Validate Risk Assessment Exists for USA Sponsored CRs

1 - 2

**Multiple Validation Rules can be linked to each workflow steps**

# Example 3 – Configuration Design

Workflow Step   Next Steps   Add CEs   Data Validation   E-Mail

Workflow E-Mail Add Delete Save

Select	Order	Send E-Mail	Type	Subject	Updated
<input type="checkbox"/>	1	Yes	Current Record	[ITEM NUMBER] has been submitted for Processing	02/18/2010 Brown Mark A
<input type="checkbox"/>	20	Yes	Superseded CR	[ITEM NUMBER] was Superseded by [PARENT NUMBER]	04/27/2011 Brown Mark A

Creator/Edit Workflow E-Mail Add Another E-Mail Cancel Save Delete

Order: 1   Type: Current Record   Active: Yes

Copy From: - Select - Copy

Email Subject: [ITEM NUMBER] has been submitted for Processing

Email Text:

[ITEM NUMBER] has been approved by the sponsor and is ready for CMO processing.

[ITEM NUMBER] - [TITLE]	
Item Number	[ITEM NUMBER]
Initiator	[INITIATOR]
Title	[TITLE]

Attach Report: - Select -

E-Mail Recipients

Dynamic Recipients:

CR OPR	To ->	Assigned To
CR Sponsor	Delete	
Watch Items		

Type Name or Select from List

Static Recipients:

Al-Shihabi MaryJo Y	CC ->	CR Initiator
AS	Delete	

Screen: 143   User: USATXBROWNMA   Current Build: 0.08.00 3/14/2011   Curator:RDM: Mark A. Brown

Each Workflow Step can generate multiple e-mail notifications

E-Mails are configured using a library of dynamic tags

Recipients can either be dynamically generated based on the linked record or defined statically



# Example 4 – Audit Tracking Design

- Audit tracking at the field level is a balance between business need and system performance
- Configurable audit tracking allows business owner to decide which fields require traceability and alter as business processes change
- Database trigger is built based on Field Selections dynamically

Columns

Change Tracking Cancel Save

Table Name: **CMST\_ATTACHMENTS**

Available Columns:

- ATTACHMENT\_SEQ
- ATTACH\_LINK\_TYPE
- ATTACH\_LINK\_SEQ
- FILE\_TITLE
- FILE\_ORDER
- PROPRIETARY
- FILE\_ORIGINAL
- FILE\_PDF
- FILE\_STATUS
- FILE\_SIZE

Assigned Columns:

- ATTACHMENT\_TYPE
- FILE\_NAME

AUDIT_TABLES
AUDIT_TABLE_SEQ(PK) – VARCHAR2(100)
TABLE_NAME – VARCHAR2(30)
COLUMN_NAME – VARCHAR2(30)
DISPLAY_ORDER – NUMBER(22)
TRIGGER_COMPILED – DATE
INSERT_ID – VARCHAR2(30)
INSERT_DTTM – DATE
LAST_UPDATE_ID – VARCHAR2(30)
LAST_UPDATE_DTTM – DATE

AUDIT_TEMPLATE
AUDIT_SEQ(PK) – NUMBER(22)
AUDIT_DATE – DATE
AUDIT_ACTION – VARCHAR2(15)
COLUMN_NAME – VARCHAR2(30)
NEW_VALUE – VARCHAR2(4000)
OLD_VALUE – VARCHAR2(4000)
DESCRIPTOR – VARCHAR2(100)
REPORTING_SYSTEM – VARCHAR2(50)
LAST_UPDATE_ID – VARCHAR2(30)

# Example 4 – Dynamic Trigger Generation

## Step 1: Find the Audit Columns

```
cursor col_cursor is
  select column_name, rownum
  from CMST_audit_tables
  where table_name = audit_table_name;

cursor cons_cursor is
  select column_name, position
  from all_cons_columns
  where table_name = audit_table_name
  and (constraint_name like 'PK_%' OR constraint_name like '%_PK')
  and owner = 'APPL_CMST'
  order by position;

cursor CHILD_CONS_CURSOR IS
SELECT DISTINCT child_cc.COLUMN_NAME --, rownum, child_cc.position
  FROM all_cons_columns child_cc,
  all_cons_columns parent_cc,
  all_tab_columns atc
 WHERE child_cc.OWNER = 'APPL_CMST'
  AND child_cc.table_name = audit_table_name
  AND atc.COLUMN_ID < 4
  AND child_cc.CONSTRAINT_NAME LIKE 'FK_%'
  AND (PARENT_CC.constraint_name like 'PK_%' OR PARENT_CC.constraint_name like '%_PK')
  AND parent_cc.column_name = child_cc.COLUMN_NAME
  AND atc.TABLE_NAME = child_cc.TABLE_NAME
  AND atc.COLUMN_NAME = child_cc.COLUMN_NAME
```

Selects Columns identified  
in the Panel

Finds Primary Key to store  
with audit records

Finds Foreign Keys  
associated with record

## Step 2: Create the Trigger

```
v_trigger_name := trim(substr('A_AUD_'||audit_table_name,1,30));
sql_hdr       := 'CREATE OR REPLACE TRIGGER ' ||v_trigger_name || '
                AFTER UPDATE or DELETE on ' ||audit_table_name || '
                for each row DECLARE v_descriptor VARCHAR2(100); BEGIN ';
```

# Example 4 – Dynamic Trigger Generation

## ➤ Step 3: Add Dynamic Body

```
:= trigger_body || ' IF UPDATING THEN ';
```

```
sql_cols := sql_seq_name||', AUDIT_ACTION, AUDIT_DATE, LAST_UPDATE_ID, COLUMN_NAME, OLD_VALUE, NEW_VALUE, DESCRIPTOR';  
open cons_cursor;  
loop  
  fetch cons_cursor into v_con_col, v_con_cnt;  
  exit when cons_cursor%notfound;  
  sql_cols := sql_cols||', '||v_con_col;  
  v_insert_values_suffix := v_insert_values_suffix||', :old.'||v_con_col;  
end loop;  
close cons_cursor;  
open col_cursor;  
loop  
  fetch col_cursor into v_col, v_cnt;  
  exit when col_cursor%notfound;  
  v_if_value_compare := ' IF (nvl(to_char(:new.'||v_col||'),'A') <> nvl(to_char(:old.'||v_col||'),'A')) then ';  
  select data_type  
  into v_data_type  
  from all_tab_columns  
  where table_name = audit_table_name  
  and column_name = v_col  
  AND OWNER = 'APPL_CMST';  
  If v_data_type = 'DATE' then  
    sql_valuesI := ' VALUES ('||sql_seq_name||'.NEXTVAL,'U',sysdate, nvl(wv_flow.g_user,user),'||v_col||',  
to_char(:old.'||v_col||','DD-Mon-YYYY HH:Mi:SS am'), to_char(:new.'||v_col||','DD-Mon-YYYY HH:Mi:SS am'), v_descriptor';  
  elsif v_data_type = 'CLOB' then  
    v_if_value_compare := ' IF (nvl(:new.'||v_col||'),'A') <> nvl(:old.'||v_col||'),'A')) then';  
    sql_valuesI := ' VALUES ('||sql_seq_name||'.NEXTVAL,'U',sysdate, nvl(wv_flow.g_user,user),'||v_col||',:old.'||v_col||',  
:new.'||v_col||', v_descriptor';  
  else  
    sql_valuesI := ' VALUES ('||sql_seq_name||'.NEXTVAL,'U',sysdate,  
nvl(wv_flow.g_user,user),'||v_col||',to_char(:old.'||v_col||'), to_char(:new.'||v_col||'), v_descriptor';  
  end if;  
  sql_valuesI := sql_valuesI || v_insert_values_suffix || ');  
  trigger_body := trigger_body || ' ' ||  
    v_if_value_compare || ' ' || --if new value != old_value then  
    v_insert_stmt_prefix || sql_valuesI || ' ' || ' ' || --insert into audit table  
    ' end if; ';
```

**Loop through columns and  
build values statement**

```
end loop;  
trigger_body := trigger_body || ' end if; '; --end of 'IF UPDATING THEN'  
close col_cursor;  
trigger_body1 := trigger_body1 || ' end;'; --end of sql_hdr begin  
execute immediate SQL_HDR || ' ' || trigger_body || ' ' || trigger_body1;
```

# No Limits to the Power of Configuration

Configurable systems allow for future growth with minimal cost

Users appreciate the control and power of configurable systems

The image displays a collage of various software configuration windows and toolbars. Key elements include:

- Automated E-Mail Notifications:** A window for configuring email actions, with fields for Name, Description, Copy From, and E-Mail Subject.
- CR Configuration:** A window for configuring Change Request (CR) forms, including CR Name, Form Title, and Description.
- External File:** A window for managing external files, with fields for Name, Description, and Menu Item URL.
- Audit Information:** A window showing system logs, including 'Inserted By/On' and 'Last Updated By/On'.
- Field Help:** A window providing help text for a specific field, such as 'Display Order'.
- External Help File:** A window for uploading external help files.