SUNGARD PUBLIC SECTOR
CHART OF ACCOUNTS SETUP

IFAS
Integrated Financial &
Administrative Solution

NOTICE

Should you wish to make a comment, or if you find an error in the text, please contact us via email:

doc@bi-tech.com

Document Change Log

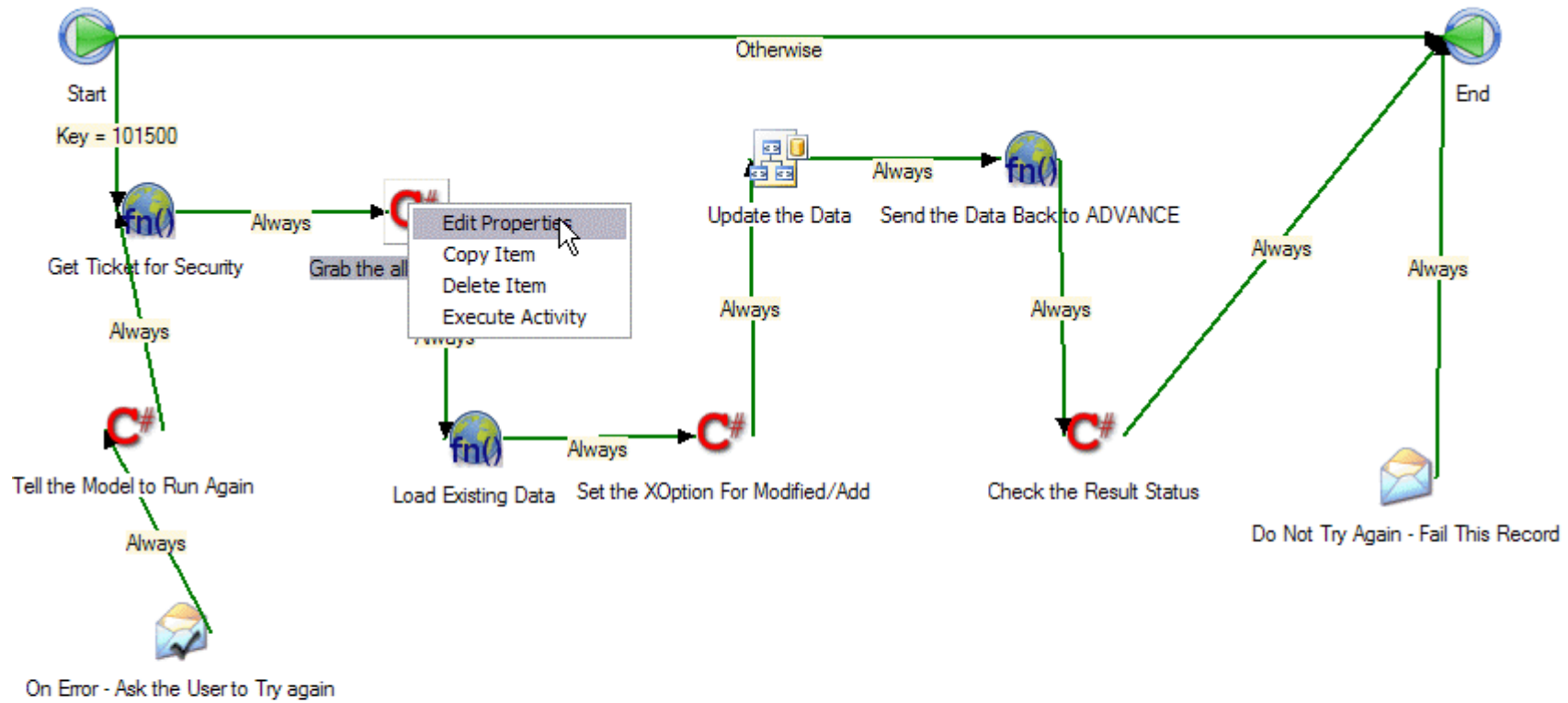| Version	Date	Change Description
7.9.4	April 2009	7.9.4 Version

Contents

1 Introduction to SunGard HigherEd Advance Chart of Accounts Interface

The SunGard HigherEd Advance Chart of Accounts Interface allows interaction between Advance and IFAS. This interface updates the Advance Chart of Account allocation table with the data from the GLK_KEY_MSTR table in IFAS. When a user inserts or updates records on the GLUPKY screen, workflow will trigger call into web services hosted by Advance to update the chart of accounts.

1.1 Setup

To activate the interface between IFAS and Advance, you must enable the CHRT_ACCTS_UPDT workflow model and setup the mapping between the Advance chart of accounts and the IFAS GLK_KEY_MSTR tables.

Once the model is activated, there are two activities that determine the mapping. Typically a background part will be used to link the GL Key to the allocation code; however any field available to the model through the data description for GLK_KEY_MSTR in dbcustom.xml can be used to link the values. To set the allocation code, modify the settings of the activity described as "Grab the allocation code" by right clicking on the activity and selecting "Edit Properties." Once the properties dialog appears, click settings to modify the allocation code.

Start

Key = 101500

Otherwise

End

Get Ticket for Security

Always

Grab the all

Edit Properties
Copy Item
Delete Item
Execute Activity

Update the Data

Always

Send the Data Back to ADVANCE

Always

Always

Always

Always

Always

Always

Tell the Model to Run Again

Always

Load Existing Data

Always

Set the XOption For Modified/Add

Check the Result Status

Do Not Try Again - Fail This Record

On Error - Ask the User to Try again

AllocationTable   AllocationTicket   AllocationCode   AdvanceAppCode   Result

General | **Who, Action, and Wait** | Transitions

Who
- ○ Person [▼]
- ○ Role   [▼]
- ○ Many   [Set Conditions...   ]
- ● None

Execution Options:
- ☐ Do not skip activity if assignee notified already

Role Assignment Options (1 only):
- ☐ Assign to another if assignee notified already
- ☐ Assign to creator of instance if in role

Action
[C#   ▼]

[Settings...   ]

When
- ○ Once
- ● Each Change in Data
- ○ Each Flow Life
- ○ Condition   [Set Data Filter   ]

Expiration Handling
Wait: [2] [Days   ▼]
After
Expiration: [Retry Forever   ▼]

Restrict Access
[No change in access   ▼]

External States
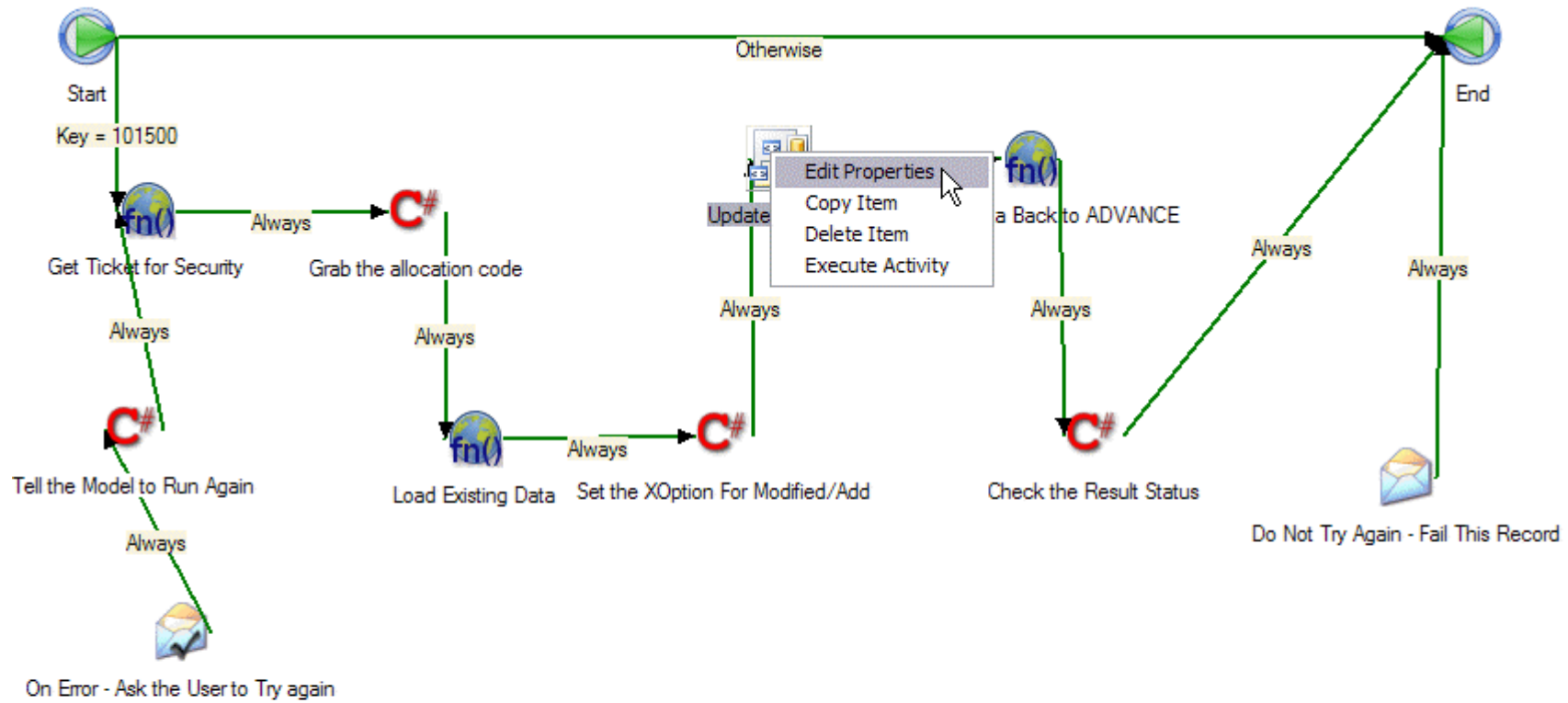Set State to: [   ]

[OK   ]   [Cancel   ]

Initialization
Edit Initialization...

Assembly References:
System.dll
System.Xml.dll

Source Code

Method: void Run()   Delete   New Method: void Method()   Add

```
Variables.AllocationCode = Source.GrpPart02;
```

Validate   Cancel   OK

Once the allocation code has been mapped from the GLK_KEY_MSTR table or other linked source, the remaining fields can be mapped by modifying the settings on the activity with the description "Update the Data." Do not modify the xoption column or the allocation_code column, they are maintained by the workflow model. To select the type of mapping, you can select a value from the binding type drop down. The options are null, a hard coded value, the value from a variable, the value from an SQL list result, or a value from the source table. Click on the association column to set the value of the binding.

Start

Key = 101500

Get Ticket for Security

Always

Grab the allocation code

Otherwise

End

Edit Properties
Copy Item
Delete Item
Execute Activity

Update

a Back to ADVANCE

Always

Always

Always

Always

Always

Tell the Model to Run Again

Always

Load Existing Data

Always

Set the XOption For Modified/Add

Check the Result Status

Do Not Try Again - Fail This Record

On Error - Ask the User to Try again

AllocationTable   AllocationTicket   AllocationCode   AdvanceAppCode   Result

General | Who, Action, and Wait | Transitions

Who
- ○ Person [▼]
- ○ Role [▼]
- ○ Many [Set Conditions...   ]
- ● None

Execution Options:
- ☐ Do not skip activity if assignee notified already

Role Assignment Options (1 only):
- ☐ Assign to another if assignee notified already
- ☐ Assign to creator of instance if in role

Action

[Update DataSet   ▼]

[Settings...   ]

When
- ○ Once
- ● Each Change in Data
- ○ Each Flow Life
- ○ Condition [Set Data Filter   ]

Expiration Handling

Wait: [2] [Days   ▼]

After
Expiration: [Retry Forever   ▼]

Restrict Access

[No change in access   ▼]

External States

Set State to: [   ]

[OK   ]   [Cancel   ]

Setup

Action: UpdateRows   Data Set Variable: AllocationTable

Table: coa_data

Where Clause/Filter

Filter:

New Data

	Column	DataType	BindingType	Association
▶	xoption	Char	Null	
	allocation_code	String	Null	
	status_code	Char	Table	GLKKeyMas...
	account	String	Table	GLKKeyMas...
	short_name	String	Table	GLKKeyMas...
	long_name	String	Table	GLKKeyMas...
	long_name2	String	Null	
	cfae_purpose_code	String	Null	
	cfae_sub_code	String	Null	
	restrict_code	String	Null	

*Only the first record of lists and tables will be bound.

The error and rejection handlers should be setup to email an appropriate user. If an error occurs, an email will be sent to the specified user explaining that an error has occurred and which key it occurred because of. To view a detailed report of the error, the user can view the report in the task list. Approving the email will trigger the model to make another attempt.