



Secrets of the Account Generator

Session Leader
Karen Brownfield, Solution Beacon

Release 11/Workshops

Orlando, FL • Chicago, IL • St. Louis, MO • Los Angeles, CA
San Ramon, CA • Boston, MA • Dallas, TX • Atlanta, GA

www.solutionbeacon.com

ORACLE CERTIFIED
PARTNER

© 2006, 2006 Solution Beacon, LLC. All Rights Reserved.





Are you an OAUG Member?

Member Benefits include:

- **Advocacy** opportunities to influence Oracle on product enhancements, usability, new features, Oracle support, pricing and quality
- **Knowledge** that showcases the latest trends and techniques used by industry leaders through our national and regional events and our publications, such as OAUG Insight magazine
- **Communication** with other OAUG members worldwide through participation in OAUG committees, leadership positions, interaction with Oracle Corporation's user initiatives, frequent member surveys, and Oracle management briefings
- **Education** through the hundreds of career-enhancing presentations in our conference paper database archive, as well as discounts to conferences and Oracle education
- **Networking** with Oracle customers, industry experts, third-party software firms, and other Oracle Applications specialists through our Member Database and Online Vendor Directory



Karen Brownfield Profile

- **25+ yrs experience in IT Industry**
- **14+ yrs experience Oracle Applications**
- **7 yrs experience Workflow**
- **OAUG Board of Directors 12 years, Former President**
- **Presenter of Training/Presentations multiple clients, conferences**
- **Solution Beacon** is a leading provider of expert-level resources to Government Agencies and Fortune 1000 companies specializing in the Oracle Applications marketplace, providing leading expertise in the following areas:
 - ◆ Release 11/Upgrades/Implementations
 - ◆ Oracle Apps Database Administration
 - ◆ Oracle Systems Architecture
 - ◆ Oracle Performance Tuning
 - ◆ Oracle Workflow





Objectives

- As a result of this presentation, you will understand:
 - ◆ How Account Generators are initiated
 - ◆ How to use the seeded packages
 - ◆ The hidden attributes and how to use them to send a notification
 - ◆ The rules to follow when customizing an account generator



Agenda

- Seeded Account Generators.
- Customization Rules.
- How Account Generators are Started.
- Standard Flexfield Activities.
- Customization Process
- Notifications of Errors
- Debugging



Account Generator

- Generate meaningful flexfield code derived from a series of concatenated segments pulled from variable tables
- Pass back completed flexfield codes to the calling applications (or information about failure to generate flexfield)



Seeded Account Generators

- OM: Generate Cost of Goods Sold Account (OECOGS) – generates COGS account when interfacing invoices to AR
- PSB Account Generator for OLD Integration (PSBLDMAG) – Public Sector Budgeting – derives accounts for positions with POETA charging instructions – used to import salary distribution info from LDC



Seeded Account Generators

- ITR Account Generator (ITRWKFAG) – builds creation and receiving accounts for ITR service lines
- IAC Account Generator (IGIIACWF) – Public sector assets Inflation accounting
- MHCA Account Generator (IGIAMAWF) – Public sector assets
- FA Account Generator (FAFLEXWF) – Fixed Assets accounts for various transactions



Seeded Account Generators

- Project Budget Account Generator (PABDACWF) – generate combinations for budget items
- Project Supplier Invoice Account Generation (PAAPIN VW) – generates combinations when invoice distribution references a project
- Inventory Cost of Goods Sold Account (INVFLXWF) – used when processing intercompany transactions



Seeded Account Generators

- PO Account Generator (POWFPOAG) – generates PO charge, encumbrance, variance, and accrual accounting distributions
- PO Requisition Account Generator (POWFRQAG) – generates requisition charge, encumbrance, variance, and accrual accounting distributions



Seeded Account Generators

- IGC Charge Account Generator (IGCACGNC) – generate charge account for contract commitment in Public Sector Contracts
- IGC Budget Account Generator (IGCACGNB) - generate budget account for contract commitment in Public Sector Contracts



Seeded Account Generators

- AR: Substitute Balancing Segment (ARSBALSG) – updates balancing segment during various accounting activities against transactions/receipts
- Generate Cost of Goods Sold Account (SHPFLXWF) – Pre 11i COGS account for Order Entry – see Metalink note 260697.1



Special Rules

- Start Generating Code Combinations
 - ◆ Must be the "start" node
- End generating Code Combination
 - ◆ Must be an "end" node
 - ◆ Must have result set to 'Success'
- Abort generating Code Combination
 - ◆ Must be an "end" node
 - ◆ Must have result set to 'Failure'



Special Rules

- Process must have result type of 'Flexfield Result'
- Must use 'Validate Code Combination' activity before ending workflow with Success result



Special Rules

- No Notification Activities
- On Revisit behaves as if set to Loop
- No Master/Detail
- No deferring of activities to background engine
- No parallel flows
 - ◆ No <Any> transitions



Special Rules

- Following std activities not allowed
 - ◆ And
 - ◆ Defer Thread
 - ◆ Wait
 - ◆ Continue Flow / Wait for Flow
 - ◆ Role Resolution
 - ◆ Voting
 - ◆ Compare Execution Time
 - ◆ Notify



Special Rules

- Following std activities not allowed (cont)
 - ◆ Block
 - ◆ Can use this if calling program issues start to workflow, and call to continue workflow in single thread
 - ◆ Example – Requisition Account Generator



Special Rules

- No data written to workflow tables
 - ◆ Can't view from Monitor
 - ◆ No auditing of process
- Following API's only ones allowed
 - ◆ CreateProcess
 - ◆ GetItemAttribute
 - ◆ GetActivityAttribute
 - ◆ AddItemAttribute
 - StartProcess
 - SetItemAttribute
 - CompleteActivity
 - LaunchProcess



Starting Account Generators

■ FND_FLEX_WORKFLOW.INITIALIZE

◆ Parameters

- ◆ Application Short Name – 'SQLGL'
- ◆ Code – 'GL#' or 'GLAT'
- ◆ Num – id of the particular accounting flexfield
- ◆ ItemType – which account generator

◆ Fetches number of segments in COA

◆ Determines whether running in Debug mode

- ◆ Sets itemkey appropriately



Starting Account Generators

- FND_FLEX_WORKFLOW.INITIALIZE
 - ◆ Calls Wf_Engine.CreateProcess
 - ◆ Adds 'hidden' attributes
 - ◆ FND_FLEX_APPSNAME – 'SQLGL'
 - ◆ FND_FLEX_CODE – 'GL#' or 'GLAT'
 - ◆ FND_FLEX_NUM – id of accounting flexfield
 - ◆ FND_FLEX_APPLID – 101
 - ◆ FND_FLEX_NSEGMENTS – number of segments
 - ◆ FND_FLEX_CCID – will be the code combination id build by the account generator



Starting Account Generators

■ FND_FLEX_WORKFLOW.INITIALIZE

◆ 'Hidden' attributes continued

- ◆ FND_FLEX_SEGMENTS – concatenated segments
- ◆ FND_FLEX_DATA – concatenated id's of each segment
- ◆ FND_FLEX_DESCRIPTIONS – concatenated descriptions of each segment
- ◆ FND_FLEX_MESSAGE – if the generator fails, will be the error message
- ◆ FND_FLEX_STATUS – validation status



Starting Account Generators

- FND_FLEX_WORKFLOW.INITIALIZE
 - ◆ 'Hidden' attributes continued
 - ◆ FND_FLEX_INSERT – whether new combinations are allowed
 - ◆ FND_FLEX_NEW – whether generated combination is new
 - ◆ FND_FLEX_SEGMENTn – one attribute for each segment
 - ◆ Sets values for FND_FLEX_APPSNAME, FND_FLEX_CODE, FND_FLEX_NUM, FND_FLEX_APPLID, FND_FLEX_NSEGMENTS



Starting Account Generators

- FND_FLEX_WORKFLOW.INITIALIZE
 - ◆ Calls FND_FLEX_WORKFLOW.GENERATE
 - ◆ From Form passes via IN/OUT variables
 - ◆ Code combination id
 - ◆ Concatenated segments
 - ◆ Concatenated descriptions
 - ◆ Concatenated ids
 - ◆ Error message
 - ◆ From Form returns TRUE/FALSE as to success of generation



Starting Account Generators

- FND_FLEX_WORKFLOW.INITIALIZE
 - ◆ Calls FND_FLEX_WORKFLOW.GENERATE
 - ◆ From Batch passes via IN/OUT variables
 - ◆ Code combination id
 - ◆ Concatenated segments
 - ◆ Concatenated descriptions
 - ◆ Concatenated ids
 - ◆ Whether id is new combination
 - ◆ Error message
 - ◆ From Batch returns TRUE/FALSE as to success of generation



Starting Account Generators

■ Assumptions

- ◆ Form uses returned value to place in form and form calls standard flex checking routines
- ◆ Batch program is responsible for insertion of any new combinations and issuing commit
 - ◆ New combination is signified by `code_combination_id = -1` and `new combination Boolean = TRUE`



Starting Account Generators

- Form Generate calls Batch Generate
- Batch Generate sets workflow engine threshold to 999999
- Batch Generate issues call to `Wf_Engine.StartProcess`



Standard Flexfield Activities

Navigator Control Properties [X]

Function | Details | Roles | Access | Node | Node Attributes

Attribute

Name: Value

Type: Item Attribute

Value: Lookup Set Value [Edit]

Name	Value Type	Value	Type	Description
Segment Identifier	Constant	Name	Lookup	
Segment	Constant	Fund	Text	Segment
Value	Item Attribute	Lookup Set Value	Text	Value
Replace existing ...	Constant	True	Lookup	Should the new

[OK] [Cancel] [Apply] [Help]



Standard Flexfield Activities

- Common Activity Attributes
 - ◆ Segment Identifier – whether segment is identified by 'Qualifier' (like Balancing Segment) or 'Name'
 - ◆ Segment – either the specific qualifier or specific name
 - ◆ Value – value to be assigned



Standard Flexfield Activities

- Common Activity Attributes (cont)
 - ◆ Structure Number – of an accounting flexfield (other than SOB one)
 - ◆ Replace existing value – use 'True' to copy value regardless of existing value, 'False' to only copy value if current value is null



Standard Flexfield Activities

- Assign Value to Segment - assigns value in to specific segment
 - ◆ Attributes – Segment Identifier, Segment, Value, Replace existing value
- Copy Values from Code Combination – copies all values from given code combination to combination being generated
 - ◆ Attributes – Code Combination ID, Replace existing value



Standard Flexfield Activities

- Copy Segment Value from Code Combination – copies a segment value from specific code combination to a segment in the combination being built
 - ◆ Attributes – Code Combination ID, Segment Qualifier, Segment, Replace existing value



Standard Flexfield Activities

- Copy Segment Value from Other Structure Code Combination – copies segment value from given code combination in another accounting flexfield to combination being generated
 - ◆ Attributes – Structure Number, Code Combination ID, Segment Identifier, Segment, Replace existing value



Standard Flexfield Activities

- Get Value from Code Combination – retrieves value from specific segment and stores result in item attribute
 - ◆ Attributes – Code Combination ID, Segment Identifier, Segment, Attribute to assign value



Standard Flexfield Activities

- Is Code Combination Complete – checks to see if all segments have values
 - ◆ Attribute – Check only for required segments – if 'True', only validates if required segments have values, if 'False', validates all segments have values



Standard Flexfield Activities

- Validate Code Combination (Continued)
 - ◆ Validation Type – set to 'Generate Code Combination ID' to do full validation and return ccID, set to 'Validate Segments with Values only' to validate that value in each segment is a valid value
 - ◆ Note: Does not do insert into GL_CODE_COMBINATIONS, returns -1 if can be inserted and combo is otherwise valid



Standard Flexfield Activities

- Validate Code Combination (Continued)
 - ◆ Sets FND_FLEX_SEGMENTS
 - ◆ If Invalid...
 - ◆ FND_FLEX_CCID = 0
 - ◆ FND_FLEX_DATA, FND_FLEX_DESCRIPTIONS set to NULL
 - ◆ FND_FLEX_STATUS = 'INVALID'
 - ◆ FND_FLEX_MESSAGE set to failure reason
 - ◆ FND_FLEX_NEW = 'N'



Standard Flexfield Activities

- Validate Code Combination (Continued)
 - ◆ If valid...
 - ◆ FND_FLEX_CCID = -1 (new combination) or id of existing combination
 - ◆ FND_FLEX_DATA set to concatenated id's of each segment
 - ◆ FND_FLEX_DESCRIPTIONS set to concatenated description of each segment
 - ◆ FND_FLEX_STATUS = 'VALID'
 - ◆ FND_FLEX_NEW = 'N' for existing combo, 'Y' for new combo



Additional Projects Activity

- Segment Lookup Set value – returns value from lookup into named set using named intermediate value
 - ◆ Parameters
 - ◆ Lookup Set Name
 - ◆ Intermediate Value
 - ◆ Requires setup of Item Attribute 'Lookup Set Value' with internal name 'LOOKUP_SET_VALUE'
 - ◆ Following node will be 'Assign value to Segment'



Customization Process

- Open Account Generator workflow seeded by Oracle (it will automatically copy in Standard Flexfield Workflow itemtype also)
- Ensure customization level set to 100 or higher



Customization Process

- Copy default process and rename to your custom name
 - ◆ Right click default process, copy
 - ◆ Right click 'Processes', paste
 - ◆ Change Internal Name, display name, description



Customization Process

- Open newly created process in diagrammer window
- Add and configure appropriate nodes
- Save to database



Register the Customization

- Sign into applications, Setup | Flexfields | Key | Accounts
 - ◆ Query records, arrow down until correct accounting flexfield shows
 - ◆ Select appropriate account generator, change process to new custom process



Register the Customization

Account Generator Processes

Application: Oracle General Ledger Flexfield Title: Accounting Flexfield
 Structure: Operations Accounting Flex Description: Vision Operations Accounting Flexie

Item Type	Process Name
MHCA Account Generator	Generate Default Account
IGC Charge Account Generator	Generate Default Account
IGC Budget Account Generator	Generate Default Account
AR: Substitute Balancing Segment	Replace Balancing Segment
FA Account Generator	Generate Default Account
Project Supplier Invoice Account Generation	Generate Default Account
Generate Cost of Goods Sold Account	Generate Default Account
Project Web Employees Account Generator	Default Account Generator for
PO Account Generator	Generate Default Accounts
PO Requisition Account Generator	Generate Default Accounts

+ Security
 Accounts
 + Descriptive
 + Validation
 + Calendars

Account Generation Processes

Find %

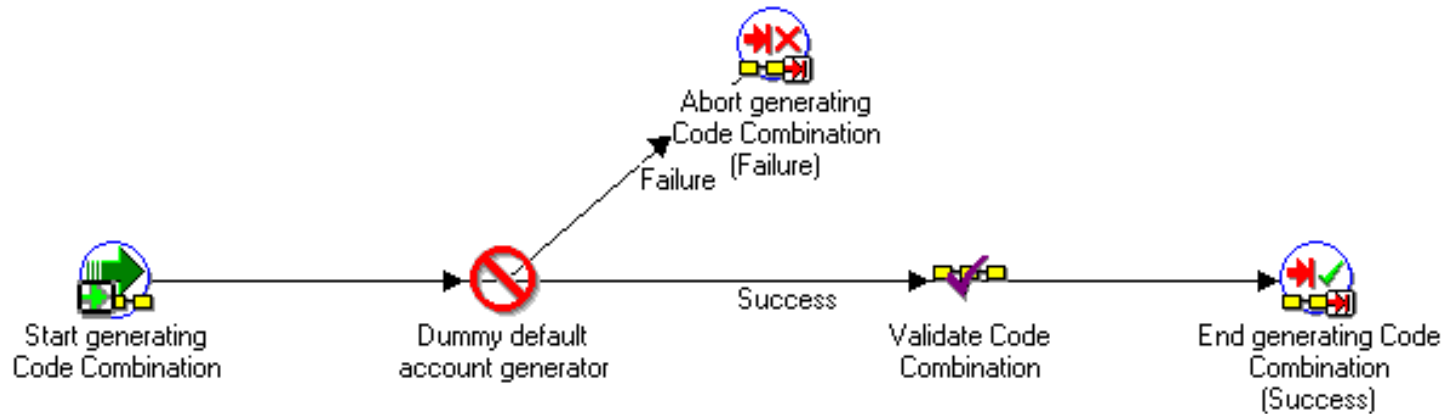
Name

- Comms - Build Supplier Invoice Project Charge Account
- Generate Default Account
- Generate Account using Flexbuilder rules
- Operations-Build AP Project Invoice Charge Account
- Progress Project Supplier Invoice Account Generator
- ProjMfg - Build Project Supplier Invoice Charge Account
- PSUK- Build Supplier Invoice Project Charge Account
- Services Build Supplier Invoice Project Charge Account
- University Project Supplier Invoice Account Generator

Find OK Cancel

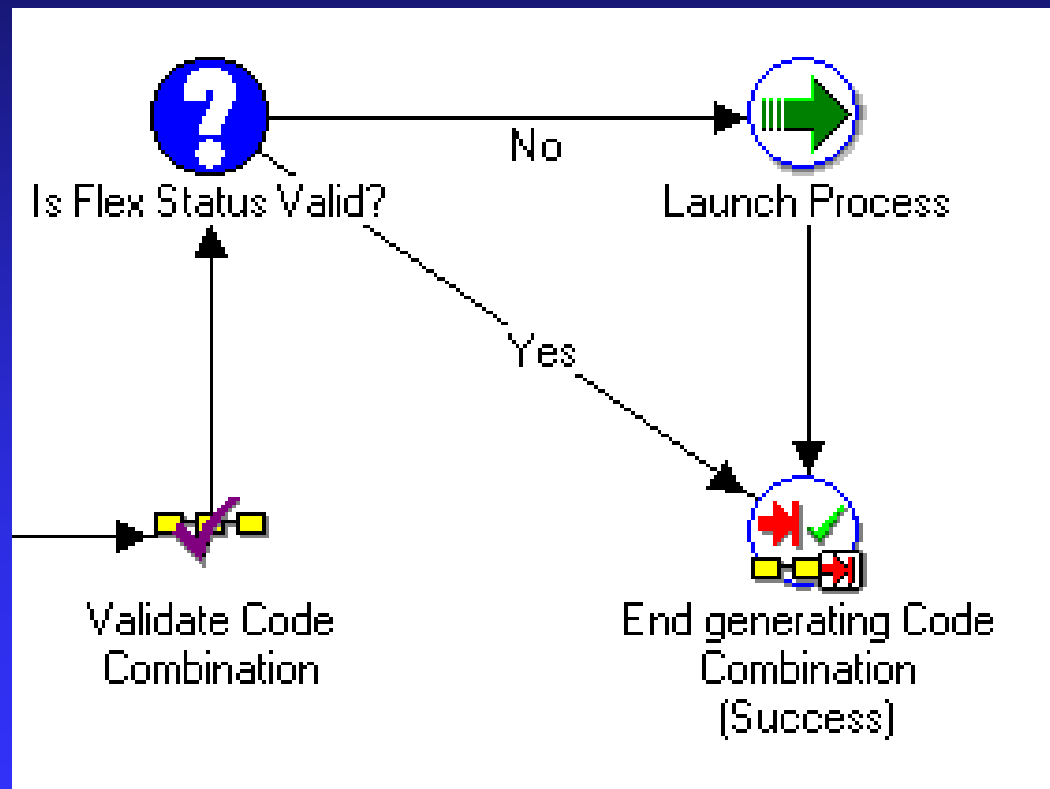


Account Generator Example



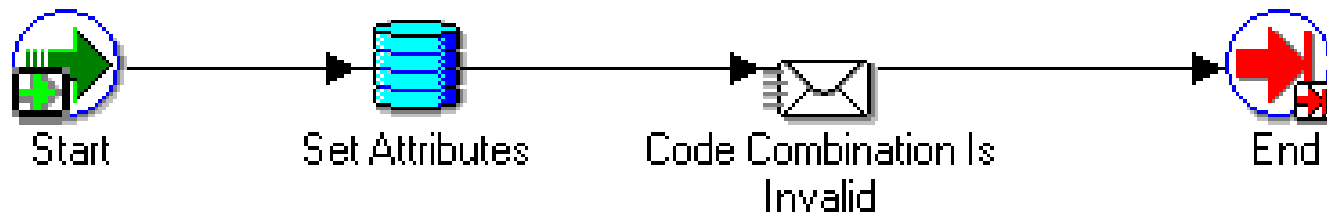


Adding a Twist





So Notification can be Sent





Passing Values

- Launched Workflow
 - ◆ Owner will be recipient of notification
 - ◆ Itemkey is concatenation of line_id, header_id, inventory_item_id, operating_unit, and SYSDATE
 - ◆ Userkey is concatenated segments (and can add error message)
 - ◆ Set Attributes breaks out keys and stores values as item attributes to be used in the message and as the performer



Debugging

- Account Generators
 - ◆ Does not add records to runtime history
 - ◆ Cannot be viewed through monitor
- Unless....
- Profile option – 'Account Generator:Run in Debug Mode' – set to 'Yes'
 - ◆ 'Hidden' attributes still not visible



Debugging

- Oracle provided scripts
 - ◆ Example – OECOGS – Metalink note 159998.1, 113492.1
 - ◆ See Top Tech Docs, Search tab
- “Breadcrumbs” – routines that create DBMS_OUTPUT file
 - ◆ Usually have to recompile the procedure to set Boolean variable to TRUE



Questions & Answers

Remember:

KISS



Keep It Simple and SupportableSM

Thank You!

Karen Brownfield
kbrownfield@solutionbeacon.com
www.solutionbeacon.com

Real Solutions for the Real World.[®]

ORACLE CERTIFIED
PARTNER

© 2005,2006 Solution Beacon, LLC. All Rights Reserved.

