



Banner Technical
Human Resources Technical
Training Workbook
Release 8.4 – November 2010
Updated 3/25/2011

Prepared by: Ellucian
4375 Fair Lakes Court
Fairfax, Virginia 22033
United States of America

Customer Support Center website
<http://connect.ellucian.com>

Distribution Services e-mail address
distserv@ellucian.com

Other services

In preparing and providing this publication, Ellucian is not rendering legal, accounting, or other similar professional services. Ellucian makes no claims that an institution's use of this publication or the software for which it is provided will insure compliance with applicable federal or state laws, rules, or regulations. Each organization should seek legal, accounting and other similar professional services from competent providers of the organization's own choosing.

Trademark

Banner®, Colleague®, PowerCAMPUS®, Luminis® and Datatel® are trademarks of Ellucian or its affiliates and are registered in the U.S. and other countries. Ellucian, Advance, Recruiter, and Open Digital Campus are trademarks of Ellucian or its affiliates. Other names may be trademarks of their respective owners.

Revision History Log

Publication Date	Summary
4/30/2008	New version that supports Banner Human Resources 8.0 software.
3/25/2011	Substantial reformatting and update.

Notice of rights

Copyright © Ellucian 2005-2012.

This document is proprietary and confidential information of one or more of the Ellucian companies and is not to be copied, reproduced, lent, displayed used or distributed except under license from Ellucian. Third party copy centers are advised that clients of Ellucian that have licensed the underlying technology associated with this documentation have a valid license to copy and use this documentation in accordance with that underlying license.



Think before you print.

Table of Contents

- Introduction.....5**
- Human Resources Objects.....7**
 - Banner Human Resources Objects.....8
 - Human Resources Forms.....9
 - Human Resources Tables.....13
 - Human Resources Hierarchy.....16
- APIs.....19**
 - Human Resources/Position Control APIs.....20
 - Human Resources APIs.....22
 - Important API Notes.....24
 - APIs used in Banner Human Resources.....26
 - Biographic/Demographic Information APIs.....27
 - Student-Employee FICA APIs.....29
 - Position Control APIs.....30
 - Combined Deduction Limits APIs.....34
 - Exercise 1: APIs.....35
- Human Resources Components.....36**
 - Introduction to Human Resources Components.....37
 - Biographic – Demographic.....39
 - Employment Administration.....44
 - Position Management.....48
 - Exercise 2 – Query Position Control Tables.....50
 - Compensation Administration.....51
 - Effective Dating.....53
 - Exercise 3: Effective Dating.....56
 - Benefits and Deductions.....57
 - Leave Administration.....60
 - Human Resources Administration.....62
 - Banner New Hire Process.....63
 - Exercise 4 – Enter a New Employee into Banner.....65
 - Human Resources Security.....66
- Load Order.....68**
 - Load Order.....69
 - Load Order – HR Core Tables.....70
- Human Resources / Banner System Interfaces.....71**
 - Interfaces and Integration with Other Banner Systems.....72
 - Advancement Interface.....73
 - Finance Interface.....74
 - Financial Aid Integration.....75
 - Student Integration.....76

Payroll Process	77
Key Concepts	78
Pre-Payroll Process	82
Banner Payroll Process	83
PHPTIME - Time Processing Report.....	92
PHPPROF - Pay Period Proof Process.....	94
PHPLEAV - Leave Accruals/Taken Process	95
PHPCALC - Payroll Calculation Process.....	97
PHPDOCM - Check/Direct Deposit Amounts Process	99
PHPCHKL/PHPCHEK Printing Process	100
PHPUPDT – Pay Period Update Process	101
Accumulator Tables	102
Exercise 5: Query Payroll Tables	103
Encumbrances and Budget	105
Banner Payroll Process – Budget Maintenance	106
Feed to Finance	107
Feed to Finance: Payroll Expenses.....	108
Feed to Finance - Budgets and Encumbrances	109
Feed to Finance - Finance Processes Take Over	110
Feed to Finance – Recommendations.....	111
Ellucian Help Resources	112
Sources of Help	113
Ellucian Support.....	114

Introduction

Course goal

This course is designed to provide an overview of the major tables, reports and processes included in each module of the Human Resources System, as well as providing an introduction to the Banner directory structure. It will prepare technical staff to support Human Resources in the implementation and operations of the Banner Human Resources product.

The course reviews some basic concepts such as naming conventions, use of the data dictionary, and referential integrity.

Hands-on exercises are used to demonstrate the table/form/process relationships.

Course objectives

Participants in this course will be able to

- identify Banner Human Resources forms and tables
- query the Banner HR tables.
- identify tables and fields for data conversion.
- identify tables and fields for migration to the production database.
- follow key HR processes.
- identify and read reports, processes, procedures and scripts in Banner Human Resources

Intended audience

Programmers, DBAs, and analysts who may teach others about Banner tables and processes, perform programming tasks in the Banner environment, facilitate reporting, utilize any of the Banner Human Resources System features, or perform analysis on any Banner Human Resources module.

Prerequisites

To complete this course, you should have

- Completed Banner Fundamentals
- Completed Banner General
- Familiarity with Relational Databases
- SQL Knowledge
- PL/SQL Knowledge
- Oracle Forms Knowledge

Human Resources Objects

Introduction

This section discusses tables, forms and class hierarchies in Banner Human Resources.

Banner Human Resources Objects

Object types

Objects in Banner Human Resources are grouped into several basic types:

- Forms
- Tables
- APIs
- Packages and Procedures
- Reports and Processes

Human Resources Forms

Forms

Users interact with the Banner database through the use of forms. Users enter data into Banner Forms and the data is saved in underlying Oracle tables.

Naming conventions

Banner forms, like all Banner objects, adhere to the Banner objects naming conventions.

- 1st character: defines the Owner / Schema
- 2nd character: defines the Component
- 3rd character: defines the Object type
- 4th to 7th characters: unique identifier, defines the purpose of the object

Form types

There are six types of HR Forms:

- Menu
- Application
- Validation
- Rules
- Query
- Inquiry

Menu forms

Menu forms do not follow the standard Banner object naming convention.

- List all related forms
- Outline the System
- Example: *HRSEMPLOYEE (Employment Administration Menu)

Application Forms

Application forms are actual applications that execute and accept input from users.

- Insert, Delete and Update data
- Invoke APIs
- Query the System
- 3rd character in the name is always an 'A'
 - Example: PEAEMPL (Employee Form)

Validation Forms

Validation forms allow data entry of all possible values for a given field.

- Supports the List of Values (F9) function
- Table and form names are the same
- Second and third characters are 'TV'
- Maintains only 3-4 fields:
 - Validation Code
 - Description
 - Activity date fields
- Example: PTVESKL (Employee Skills)

Rule Forms

Rule forms define variables for an application by maintaining the variable codes, descriptions and rules that configure how the system works.

- 2nd and 3rd characters are always 'TR'
 - Example: PTRECLS (Employee Class)
 - Example: PTRBDCA (Benefits and Deductions)
 - Example: NTRPCLS (Position Class)

Query Forms

Query forms allow users to query data using Query Mode functionality, and to export lists of data in mass.

- 3rd character is a 'Q'
- Must be called by another form
- Look-up information only
- Information cannot be changed
 - Example: PTQECLS (Employee Class Query Form - called via clicking the **Search** icon in key block of PTRECLS)

Inquiry Forms

Inquiry forms allow users to query data and perform functions such as mathematical calculations.

- 3rd character is a 'I'
- Query data and return to another form
- Data cannot be changed
- Form can be accessed from any menu
 - Example: PEIETOT (Employee Year To Date Totals Form)
 - Example: PEIDTOT (Employee Year to Date Deductions Form)

Human Resources Tables

Tables

Banner data is stored in Oracle tables. Users enter data into Banner Forms and the data is saved in underlying Oracle tables.

Banner tables, like all Banner objects, adhere to the Banner object naming conventions.

Naming conventions

Banner forms, like all Banner objects, adhere to the Banner objects naming conventions.

- 1st character: defines the Owner / Schema
- 2nd character: defines the Component
- 3rd character: defines the Object type
- 4th to 7th characters: unique identifier, defines the purpose of the object

Table types

The three basic types of HR tables are:

- Application Tables
 - Base Tables
 - Repeating Tables
 - Temporary Tables
- Validation Tables
- Rules Tables

Application Tables are 'data' tables that hold data about specific positions and people.

Rules and Validation tables are 'configuration' tables. They hold the codes, descriptions, and rules that configure how the system works.

Application Tables: Base Tables

There can be only one occurrence of the logical key.

Example: PEBEMPL (Employee Base Table)

- The logical key is PEBEMPL_PIDM (person identification master)
 - One record for each person (employee)

Example: NBBPOSN (Position Base Table)

- The logical key is NBBPOSN_POSN
 - One record for each position

The API creates a matching History Record to capture changes made to the Base Table data.

Application Tables: Repeating Tables

There can be multiple occurrences of the logical key.

Example: PERLEAV (Leave Balances Repeating Table)

- The logical key is PERLEAV_PIDM
- Multiple records – each employee has a record for each of their leave types

Example: NBRJOBS (Job Assignment Repeating Table)

- The logical key is NBRJOBS_PIDM, NBRJOBS_POSN, NBRJOBS_SUFF
- Multiple records – each person may have multiple job assignments
- Each job assignment may 'stack' historical changes to the job, such as Annual Salary Increases using Effective Dates

Application Tables: Temporary Tables

- System-maintained intermediate and internal holding area for Banner reports and processes
- Sometimes referred to as 'Collector Tables'
- Same naming convention as Application Tables
- Example: PHRTDED

Validation Tables

A validation table and its corresponding form will have the same name.

Invoke FK1 constraints / errors to ensure data integrity in the Banner HR system.

Validation tables have at minimum these three columns:

- *tablename_code*
- *tablename_desc*
- *tablename_activity_date*

Rules Tables

Key column

- *tablename_code*
- Example: PTRECLS_CODE

Description column

- *tablename_desc*
- Example: PTRECLS_SHORT_DESC

Numerous other fields are used to associate data throughout the HR system configuring the Banner HR hierarchy.

Human Resources Hierarchy

Hierarchy

Banner Human Resources uses a hierarchy of classes to ease data entry.

Employees and job assignments are 'classed' or 'grouped'. Data entering these classes will then default other information automatically into the forms and tables.

These classes and groupings are defined in the rules and validation tables.

Diagram 1

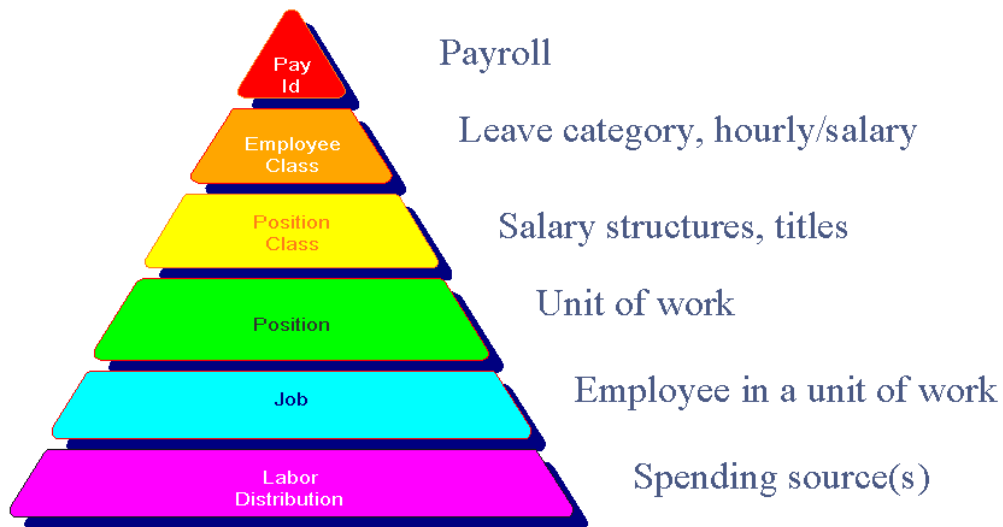


Diagram 2

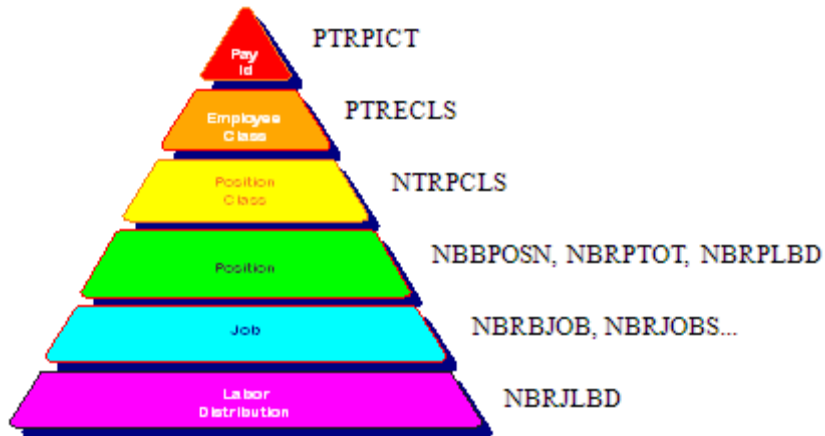
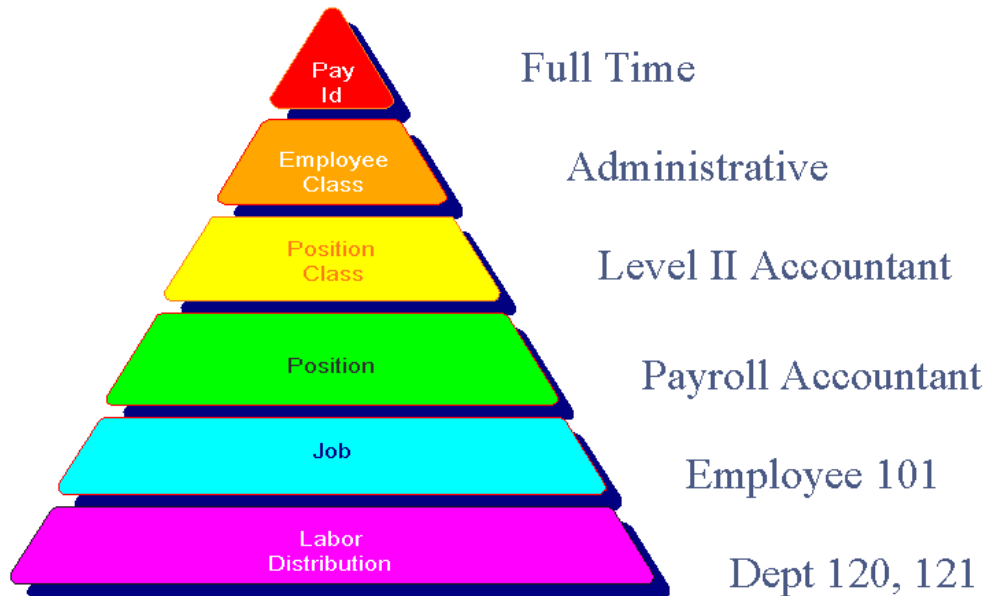


Diagram 3



Data defaults

Data defaults down (NOT up) the pyramid.

- Example: A Labor Distribution Override at the Job level will not change the labor distribution at the Budget level.

Data defaults once.

- Example: A Labor Distribution Change on a position budget will NOT change the labor distribution of employees already assigned to that position.

APIs

Introduction

This section lists the APIs used in Banner Human Resources.

Human Resources/Position Control APIs

Introduction

A *Banner API* is essentially a database package that encapsulates the business logic surrounding a Banner business entity. APIs are PL/SQL packages owned by the Banner ID *baninst1*.

A *Banner business entity* is the fundamental unit of information that a Banner application can modify (for example, a course, address, e-mail, and telephone number).

APIs leverage Banner messaging support so that Banner becomes "messaging enabled", i.e., becomes able to stimulate the production of messages based on changes to data they own. Programs within Banner as well as external systems can then manipulate a business entity or table by calling APIs.

The foundation of all Banner APIs is the ability to perform CRUD (create, retrieve, update, and delete) actions on the database. APIs contain the same validation logic and edits that are executed when performing these actions (CRUD) from a Banner Internet-Native form, self-service page, or batch process.

For example, calling the API for the "employee" business entity allows you to:

- Create employee information (insert rows into the Employee table).
- Retrieve data from the Employee table.
- Update employee information (modify rows in the Employee table).
- Delete employee information (physically remove data from the Employee table).

The advantages of using APIs include:

- APIs expose business objects that are useful to users of other applications comprising an institution's digital campus.
- APIs are bi-directional. That is, they support read and write functions.
- APIs ensure consistent edit and validation business logic (create, retrieve, update, and delete).
- APIs eliminate duplicate code that would need to be maintained in each application.
- Oracle and Banner HR security will continue to execute when data is processed through the API, no matter whether the processing source is within Banner or external to it.

Human Resources APIs

Employee API (PB_EMPLOYEE)

The Employee API (PB_EMPLOYEE) is a database package that contains the business logic (validation rules and other related processing) surrounding the table associated with Employee data. The Employee API can be called by programs within Banner (like Forms, C processes, Web packages, etc.) as well as by external systems when there is a need to insert/update/delete information into the Employee table (PEBEMPL).

The API provides for consistent processing and utilizes the same business logic whether a program in Banner calls it or an external system calls it. The table that contains Employee information is PEBEMPL.

The primary form associated with the Employee table is the Employee Form (PEAEMPL). This form retains a number of immediate edits over the data, such as the validations of Employee Class, against the Benefit Category and Leave Category codes. When inconsistent data is detected, the error messages will continue to be displayed at the bottom of the window on the Auto-help line. However, when you save the data and one or more error conditions continue to exist, a new window will open, detailing all error messages. Otherwise, you will not notice any other functional differences from prior versions of Banner Human Resources.

Other Human Resources forms and processes affect data associated with the employee table, and these have been modified to process through the Employee API. These include:

- Person Identification (Biographical Info) Form (PPAIDEN)
- New Hire Form (PEAHIRE) - This form creates Employee table data and validates using the Employee API;
- One-time Payment Form (PEA1PAY) - This form processes quick payments for employees and also impacts the Employee table (PEBEMPL);
- Employee Status Change Form (PEAESCH) - This form updates the Employee table with Leave or Termination data;
- Termination Verification Form (PEATMVF) - This form is designed to review employee records where there has been a Termination Date and Termination Reason established on PEAEMPL, but the Employee Status has not been set to 'Terminated'.
- Pay Period Update Process (PHPUPDT) - This process is the final payroll program that may affect employee records where the Employee Status is changed because Leave or Termination Dates have been future effective-dated consistent with the payroll.
- Electronic Approvals, EPAF package (NOKPLIB) - Since employee records can be created or updated through the use of EPAF, this package was modified to call the Employee API.

Please refer to the [API Reference Guide](#) for documentation related to all edits associated with the Employee API.

The PB_EMPLOYEE package provides the common business interface for the Employee API. The Employee API establishes information about an employee's terms of employment.

The following table provides a brief overview of the Employee API package. For more details, refer to the *HR API Reference Guide*.

API Object Name	API Entity Name	Associated API Packages	Associated scripts	Associated Table	Affected Objects (Forms, Processes, etc.)
PB_EMPLOYEE	EMPLOYEE	PB_EMPLOYEE_RULES PB_EMPLOYEE_STRINGS	pokb_employee0.sql pokb_employee1.sql pokb_employee_r0.sql pokb_employee_r1.sql pokb_employee_s0.sql pokb_employee_s1.sql	PEBEMPL	PEAEMPL PEAESCH PEAHIRE PEA1PAY PHPUPDT NOKPLIB

Important API Notes

Employee History Table Enhancement

Several tables are affected by the Employee API:

- PEBEMPL
- PEREHIS
- PDRBENE
- PERLEAV
- PERLHIS

The Employee History table (PEREHIS) has been expanded to include all data items associated with the Employee table (PEBEMPL). Therefore, any insert or change to any existing field on the Employee Form (PEAEMPL) will result in an audit record being posted to the Employee History table, on a going-forward basis. No changes were made to the Employee History Form (PEIEHIS) to display these new items. However, the audited elements are present on the table.

Payroll Processing Notes

An Employee table update function in the Pay Period Update Process (PHPUPDT) was modified to also utilize the Employee API. This API is called when the process determines that the Employee Status should be updated to reflect that the employee on Leave or Terminated. Through this evaluation, the Start and End Dates are evaluated against the job record in order to determine whether this update should occur.

For example, on December 1st, you indicate that the job is to go on Leave on December 31st and re-activates the job on March 1st. Next, in order to indicate the Leave information on the Employee record you enter that the Leave Start Date is January 1st and the Expected Return Date is February 28th. During the month-end payroll, since all records are in synch, the Pay Period Update Process (PHPUPDT) changes the Employee Status to *Leave*.

There is a situation within Leave and Termination processing where the Pay Period Update Process (PHPUPDT) may error, the employee will not be updated and the process will stop. This will affect the finalization of the payroll. This situation involves records where you have further modified the job status and dates without reviewing the Employee record controls. Following the above example, if the job record has been reactivated on January 15th and is extracted and attempted to be paid, the Pay Period Update Process (PHPUPDT) will error out because the employee Leave dates are inconsistent with the job dates.

In order to ensure that the Pay Period Update Process (PHPUPDT) does not error and terminate, an edit has been placed in the Pay Period Proof Process (PHPPROF). If the job dates and status are not consistent with the employee record criteria, either on the Leave or Termination windows, the employee will receive an error in the payroll and the disposition will be set to '05'. The employee must be deleted in the payroll, their records must be corrected on the Employee Form (PEAEMPL), and they must be re-extracted through the Time Processing Report (PHPTIME) to be paid.

Security

It is important to note that Oracle and Banner HR security will continue to execute when data is processed through the API against the employee table, whether the processing source is within Banner or external to it.

APIs used in Banner Human Resources

BPIs

Three BPIs exist to maintain job assignment data. These BPIs call the job related APIs and must be invoked in lieu of calling APIs directly to maintain job assignment data integrity.

- Job Labor BPI
- Job Assignment BPI
- Job Earnings BPI

Additional information on APIs used in Biographic/Demographic Information, Student Employee FICA and Position Control follows.

Biographic/Demographic Information APIs

Biographic/Demographic Information APIs

The following table provides a brief overview of the Biographic/Demographic APIs. For more details, refer to the [Banner Human Resources API Reference Guide](#), Release 7.2. These APIs replace the code in the corresponding Banner form.

All APIs packages used in Banner Human Resources will begin with a 'p,' followed by an 'b,' and an underscore '_' followed by a verb/noun describing what the API does. If the 2nd letter is a 'p' instead of a 'b,' then that database package is a Business Process API defining a 'unit of work'.

API Object Name	API Entity Name	Associated API Packages	Associated scripts	Associated Table	Affected Forms and Processes
PB_CERTIFICATION	CERTIFICATION	pb_certification pb_certification_rules pb_certification_strings	ppkb_certification0.sql ppkb_certification1.sql ppkb_certification_r0.sql ppkb_certification_r1.sql ppkb_certification_s0.sql ppkb_certification_s1.sql ppkd_pprcert0.sql ppkd_pprcert1.sql	PPRCERT	PPACERT
PB_ENDORSEMENT	ENDORSEMENT	pb_endorsement pb_endorsement_rules pb_endorsement_strings	ppkb_endorsement0.sql ppkb_endorsement1.sql ppkb_endorsement_r0.sql ppkb_endorsement_r1.sql ppkb_endorsement_s0.sql ppkb_endorsement_s1.sql ppkd_pprends0.sql ppkd_pprends1.sql	PPREND	PPACERT
PB_EXPERIENCE	EXPERIENCE	pb_experience pb_experience_rules pb_experience_strings	ppkb_experience0.sql ppkb_experience1.sql ppkb_experience_r0.sql ppkb_experience_r1.sql ppkb_experience_s0.sql ppkb_experience_s1.sql ppkd_pprexpe0.sql ppkd_pprexpe1.sql	PPREXPE	PPAEXPR

API Object Name	API Entity Name	Associated API Packages	Associated scripts	Associated Table	Affected Forms and Processes
PB_DRIVER_LICENSE	DRIVER_LICENSE	pb_driver_license pb_driver_license_rules pb_driver_license_strings	ppkb_driver_lic0.sql ppkb_driver_lic1.sql ppkb_driver_lic_r0.sql ppkb_driver_lic_r1.sql ppkb_driver_lic_s0.sql ppkb_driver_lic_s1.sql ppkd_pprdlic0.sql ppkd_pprdlic1.sql	PPRDLIC	PPAGENL
PB_HONOR_AWARD	HONOR_AWARD	pb_honor_award pb_honor_award_rules pb_honor_award_strings	ppkb_honor_award0.sql ppkb_honor_award1.sql ppkb_honor_award_r0.sql ppkb_honor_award_r1.sql ppkb_honor_award_s0.sql ppkb_honor_award_s1.sql ppkd_pprhaw0.sql ppkd_pprhaw1.sql	PPRHAW	PPAGENL
PB_PUBLICATION	PUBLICATION	pb_publication pb_publication_rules pb_publication_strings	ppkb_publication0.sql ppkb_publication1.sql ppkb_publication_r0.sql ppkb_publication_r1.sql ppkb_publication_s0.sql ppkb_publication_s1.sql ppkd_pprpubl0.sql ppkd_pprpubl1.sql	PPRPUBL	PPAGENL
PB_SKILL	SKILL	pb_skill pb_skill_rules pb_skill_strings	ppkb_skill0.sql ppkb_skill1.sql ppkb_skill_r0.sql ppkb_skill_r1.sql ppkb_skill_s0.sql ppkb_skill_s1.sql ppkd_pprskil0.sql ppkd_pprskil1.sql	PPRSKIL	PPASKIL
PB_PTVENDS	n/a	pb_ptvends	pvkb_ptvends0.sql pvkb_ptvends1.sql	PTVENDS	PB_ENDORSEMENT
PB_PTVLCSV	n/a	pb_ptvlcsv	pvkb_ptvlcsv0.sql pvkb_ptvlcsv1.sql	PTVLCSV	PB_CERTIFICATION
PB_PTVPUBT	n/a	pb_ptvpubt	pvkb_ptvpubt0.sql pvkb_ptvpubt1.sql	PTVPUBT	PB_PUBLICATION

Student-Employee FICA APIs

Student-Employee FICA APIs

The following table provides a brief overview of the student FICA APIs. For more details, refer to the [Banner Human Resources API Reference Guide](#), Release 7.2. These APIs replace the code in the corresponding Banner Human Resources forms.

API Object Name	API Entity Name	Associated API Packages	Associated scripts	Associated Table	Affected Objects (Forms, Processes, etc.)
PB_ASSOC_TERM	ASSOC_TERM	pb_assoc_term pb_assoc_term_rules pb_assoc_term_strings	phkb_assoc_term0.sql phkb_assoc_term1.sql phkb_assoc_term_r0.sql phkb_assoc_term_r1.sql phkb_assoc_term_s0.sql phkb_assoc_term_s1.sql phkd_ptratrm0.sql phkd_ptratrm1.sql	PTRATRM	PTRATRM
PB_STU_DEDUCTION	STU_DEDUCTION	pb_stu_deduction pb_stu_deduction_rules pb_stu_deduction_strings	phkb_stu_deduction0.sql phkb_stu_deduction1.sql phkb_stu_deduction_r0.sql phkb_stu_deduction_r1.sql phkb_stu_deduction_s0.sql phkb_stu_deduction_s1.sql phkd_ptrstde0.sql phkd_ptrstde1.sql	PTRSTDE	PTRSTDE
PB_STU_EMP_CRED	STU_EMP_CRED	pb_stu_emp_cred pb_stu_emp_cred_rules pb_stu_emp_cred_strings	phkb_stu_emp_cred0.sql phkb_stu_emp_cred1.sql phkb_stu_emp_cred_r0.sql phkb_stu_emp_cred_r1.sql phkb_stu_emp_cred_s0.sql phkb_stu_emp_cred_s1.sql phkd_ptrscrd0.sql phkd_ptrscrd1.sql	PTRSCRD	PTRSCRD
PB_PTVATRM	n/a	pb_ptvatrm	pvkb_ptvatrm0.sql pvkb_ptvatrm1.sql	PTVATRM	PB_ASSOC_TERM

Position Control APIs

Electronic Approvals APIs

The following table provides a brief overview of the APIs for Electronic Approvals. For more details, refer to the *Position Control API Reference Guide*.

API Object Name	API Entity Name	Associated API Packages	Associated scripts	Associated Table	Affected Objects (Forms, Processes, etc.)
NB_EA_COMMENTS	EA_COMMENTS	NB_EA_COMMENTS NB_EA_COMMENTS_RULES NB_EA_COMMENTS_STRINGS	nokb_ea_comments0.sql nokb_ea_comments1.sql nokb_ea_comments_r0.sql nokb_ea_comments_r1.sql nokb_ea_comments_s0.sql nokb_ea_comments_s1.sql nokd_norcmnt0.sql nokd_norcmnt1.sql	NORCMNT	NOAAPSM NOAEPAF
NB_EA_GROUP_CATEGORY	EA_GROUP_CATEGORY	NB_EA_GROUP_CATEGORY NB_EA_GROUP_CATEGORY_RULES NB_EA_GROUP_CATEGORY_STRINGS	ntkb_ea_group_cat0.sql ntkb_ea_group_cat1.sql ntkb_ea_group_cat_r0.sql ntkb_ea_group_cat_r1.sql ntkb_ea_group_cat_s0.sql ntkb_ea_group_cat_s1.sql ntkd_ntragrp0.sql ntkd_ntragrp1.sql	NTRAGRP	NTRAGRP
NB_EA_GROUP_USER	EA_GROUP_USER	NB_EA_GROUP_USER NB_EA_GROUP_USER_RULES NB_EA_GROUP_USER_STRINGS	ntkb_ea_group_user0.sql ntkb_ea_group_user1.sql ntkb_ea_group_user_r0.sql ntkb_ea_group_user_r1.sql ntkb_ea_group_user_s0.sql ntkb_ea_group_user_s1.sql ntkd_ntragid0.sql ntkd_ntragid1.sql	NTRAGID	NTRAGRP NOAOGRP

Jobs Information APIs

Four APIs have been created for the following tables associated with the Employee Jobs Form (NBAJOBS), namely:

- The Base Job API contains the business logic for processing base job data (NBRBJOB) on the Employee Jobs Form (NBAJOBS).
- The Job Detail API contains the business logic for processing job detail data (NBRJOBS) on the Employee Jobs Form (NBAJOBS).
- The Job Earnings API contains the business logic for processing job default earnings data (NBREARN) on the Employee Jobs Form (NBAJOBS).
- The Job Labor Distribution API contains the business logic for processing job labor distribution data (NBRJLBD) on the Employee Jobs Form (NBAJOBS).

Three BPIs have been created to maintain the employee jobs information. Each BPI, in turn, invokes one or more of the new APIs.

Note: It is important to note that the APIs alone cannot be used to maintain the jobs data. Applications must invoke the BPIs to ensure data integrity.

The new BPIs are:

- Job Labor BPI
- Job Assignment BPI
- Job Earnings BPI

The following table provides a brief overview of the Jobs APIs associated with the Employee Jobs Form (NBAJOBS). For more details, refer to the Position Control [API Reference Guide](#), Release 7.3.

API Object Name	API Entity Name	Associated API Packages	Associated scripts	Associated Table	Affected Objects (Forms, Processes, etc.)
NB_JOB_BASE	JOB_BASE	nb_job_base nb_job_base_rules nb_job_base_strings	nbkb_job_base0.sql nbkb_job_base1.sql nbkb_job_base_r0.sql nbkb_job_base_r1.sql nbkb_job_base_s0.sql nbkb_job_base_s1.sql nbkd_nbrbjob0.sql nbkd_nbrbjob1.sql	NBRBJOB	NBAJOBS PEA1PAY PEAHIRE PEAESCH
NB_JOB_DETAIL	JOB_DETAIL	nb_job_detail nb_job_detail_rules nb_job_detail_strings	nbkb_job_detail0.sql nbkb_job_detail1.sql nbkb_job_detail_r0.sql nbkb_job_detail_r1.sql nbkb_job_detail_s0.sql nbkb_job_detail_s1.sql nbkd_nbrjobs0.sql nbkd_nbrjobs1.sql	NBRJOBS	NBAJOBS PEA1PAY PEAHIRE PEAESCH
NB_JOB_EARNINGS	JOB_EARNINGS	nb_job_earnings nb_job_earnings_rules nb_job_earnings_strings	nbkb_job_earnings0.sql nbkb_job_earnings1.sql nbkb_job_earnings_r0.sql nbkb_job_earnings_r1.sql nbkb_job_earnings_s0.sql nbkb_job_earnings_s1.sql nbkd_nbrearn0.sql nbkd_nbrearn1.sql	NBREARN	NBAJOBS PEA1PAY PEAHIRE PEAESCH
NB_JOB_LABOR	JOB_LABOR	nb_job_labor nb_job_labor_rules nb_job_labor_strings	nbkb_job_labor0.sql nbkb_job_labor1.sql nbkb_job_labor_r0.sql nbkb_job_labor_r1.sql nbkb_job_labor_s0.sql nbkb_job_labor_s1.sql nbkd_nbrjld0.sql nbkd_nbrjld1.sql	NBRJLBD	NBAJOBS PEA1PAY PEAHIRE PEAESCH

Business Process Interface (BPIs) Packages

Three BPIs have been created to maintain the employee jobs information. Each BPI, in turn, invokes one or more of the Jobs APIs. It is important to note that the APIs alone cannot be used to maintain the jobs data. Applications must invoke the BPIs to ensure data integrity.

The following table provides a brief overview of the Jobs APIs associated with the Employee Jobs Form (NBAJOBS). For more details, refer to the [Position Control API Reference Guide](#), Release 7.3.

BPI Object Name	BPI Entity Name	Associated BPI Package	Associated API Packages	Associated BPI scripts	Associated Table	Affected Objects (Forms, Processes, etc.)
JOB_LABOR	JOB_LABOR	np_job_labor	nb_job_labor nb_job_labor_rules nb_job_labor_strings	nbkp_job_labor0.sql nbkp_job_labor1.sql	NBRJLBD	NBAJOBS PEA1PAY PEAHIRE PEAESCH
JOB_ASSIGNMENT	JOB_ASSIGNMENT	np_job_assignment	nb_job_detail_rules nb_job_detail_strings nb_job_base_rules nb_job_base_strings	nbkp_job_assignment0.sql nbkp_job_assignment1.sql	NBRJOBS NBRBJOB	NBAJOBS PEA1PAY PEAHIRE PEAESCH
JOB_EARNINGS	JOB_EARNINGS	np_job_earnings	nb_job_earnings_rules nb_job_earnings_strings	nbkp_job_earnings0.sql nbkp_job_earnings1.sql	NBREARN	NBAJOBS PEA1PAY PEAHIRE PEAESCH

Combined Deduction Limits APIs

Combined Deduction Limits APIs

The following table provides a brief overview of the APIs for combined deduction limits, which are new to Banner 8.0. For more details, refer to the *Banner 8 Human Resources Release Guide*.

API Object Name	API Entity Name	Associated API Packages	Associated scripts	Associated Table	Affected Objects (Forms, Processes, etc.)
PB_DEDN_COMB_LMT	DEDN_COMB_LMT	pb_dedn_comb_lmt pb_dedn_comb_lmt_rules pb_dedn_comb_lmt_strings		PTRBDCL	PTRBDCL

Human Resources Components

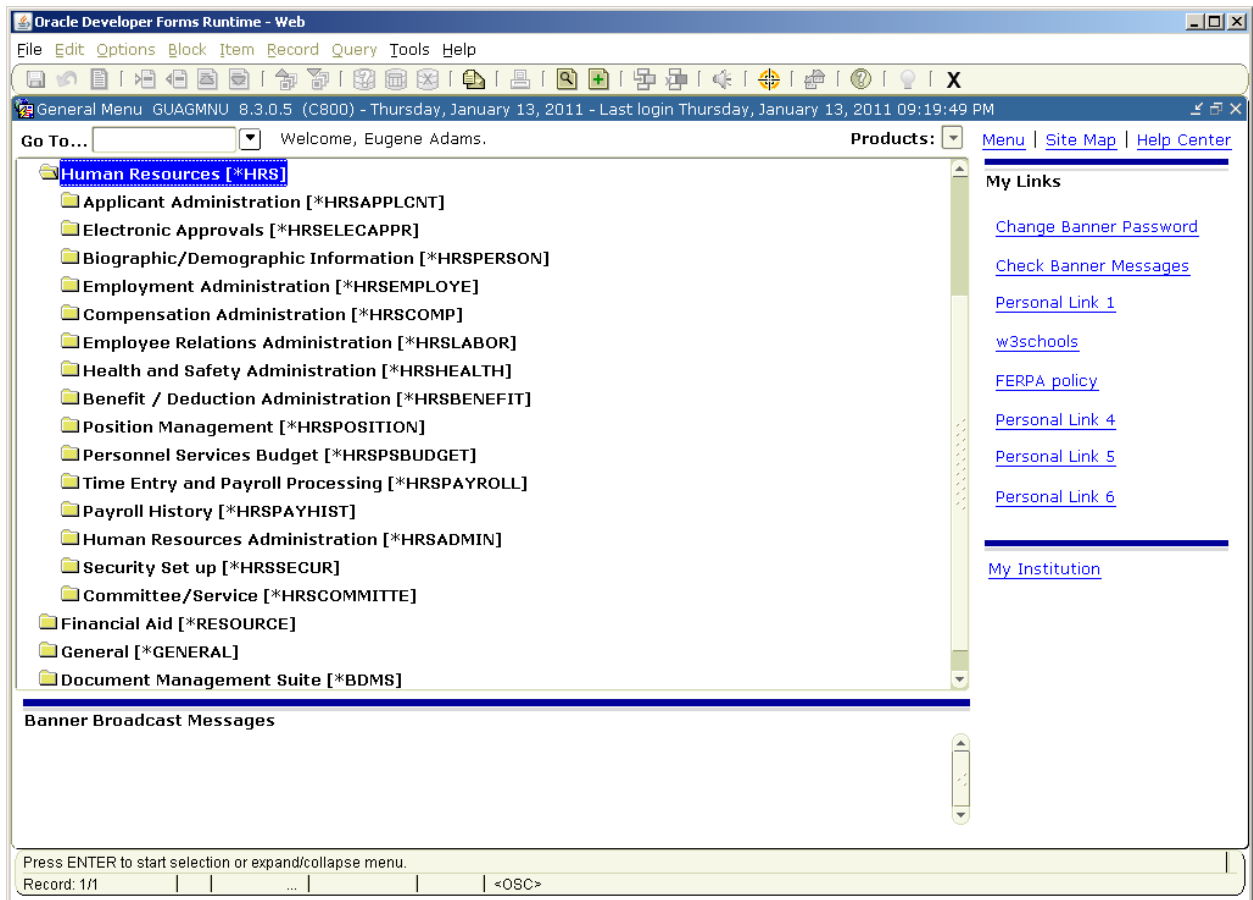
Introduction

This section provides details on the modules that make up the Banner Human Resources system.

Introduction to Human Resources Components

Introduction

Components within the Banner Human Resources system are logical groupings of data of a fully integrated HR/Payroll system. The titles relate directly to the way the HR menus break down.



Core components

Core Banner Human Resources components include:

- Biographic/Demographic Information *HRSPERSON
- Employment Administration *HRSEMPLOYEE
- Compensation Administration *HRSCOMP
- Benefit/Deductions Administration *HRSBENEFIT
- Position Management *HRSPPOSITION
- Personnel Services Budget *HRSPSBUDGET
- Human Resources Administration *HRSADMIN
- Security Set up *HRSSECUR

Optional components

Some of these latter components are not critical to the payroll process, but remain useful functionality.

- Applicant Tracking *HRSAPPLCNT
- Electronic Approvals (EPAF) *HRSELECAPPR
- Employee Relations Administration *HRSLABOR
- Health and Safety Administration *HRSHEALTH
- Committee / Service *HRSCOMMITTEE

Biographic – Demographic

Biographic/ Demographic Information

The core of the Biographic and Demographic Information is the Identification Form (PPAIDEN), which establishes a unique ID number for each individual in the system. Use this form to enter basic biographic information about an applicant or employee.

- Maintain:
 - Biographic information
 - Educational background
 - Professional qualifications
- Shared table data owned by SATURN and the Banner General module
- Some fields are REQUIRED to create Employee records

HR Required Fields – SPBPERS

- SPBPERS_SSN
- SPBPERS_BIRTH_DATE
- SPBPERS_CITZ_CODE
- SPBPERS_ETHN_CODE
- SPBPERS_SEX

HR Required Fields – SPRADDR

An employee must have at least one active address record for the address type(s) designated for payroll.

- SPRADDR_PIDM
- SPRADDR_ATYP_CODE
- SPRADDR_SEQ_NO
- SPRADDR_STREET_LINE1
- SPRADDR_CITY
- SPRADDR_STAT_CODE
- SPRADDR_ZIP

Core Application Forms

- PPAIDEN
 - Establishes a unique identifier PIDM
 - Maintains biographic and demographic information
- PPAGENL
 - Maintains professional qualifications
- GXADIRD
 - Maintains direct deposit information
- GOAINTL
 - Maintains international data

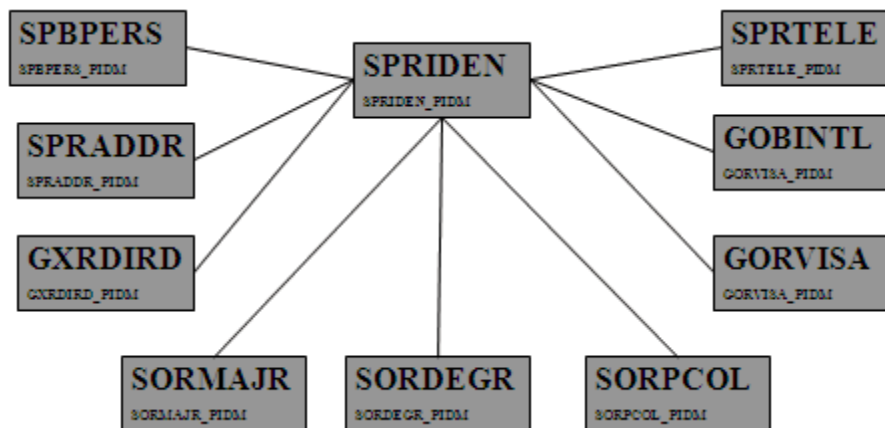
Rule Forms

- PTRCERT
 - Certification Code

Validation Forms

- STVATYP
 - Address Type Code
- STVCITZ
 - Citizen Type Code
- STVETHN
 - Ethnic Code
- STVSBGI
 - College Codes

Diagram



Person records

- Each person will have a SPRIDEN record with a SPRIDEN_CHANGE_IND of null
- Other SPRIDEN records for the person will have a value in the change_ind indicating the type of change (N)ame or (I)D
- Each person must have one SPBPERS record with the (5) required elements
- Each person must have one SPRADDR record

Employment Administration

Purpose

The Employment Administration module supports the entering and reporting of employee information.

- Maintain:
 - Employee's status
 - Hire dates
 - Benefit category (BCAT)
 - Leave category (LCAT)
 - Home Department (validates against Banner Finance)

Core Application Forms

- PEAEMPL
Establishes employee information, status, benefit and leave categories
- PEAREVW
Maintain performance review information
- PEALEAV
View and maintain leave balance records (IF leave by employee method chosen on PTRINST – populated by PEAEMPL)

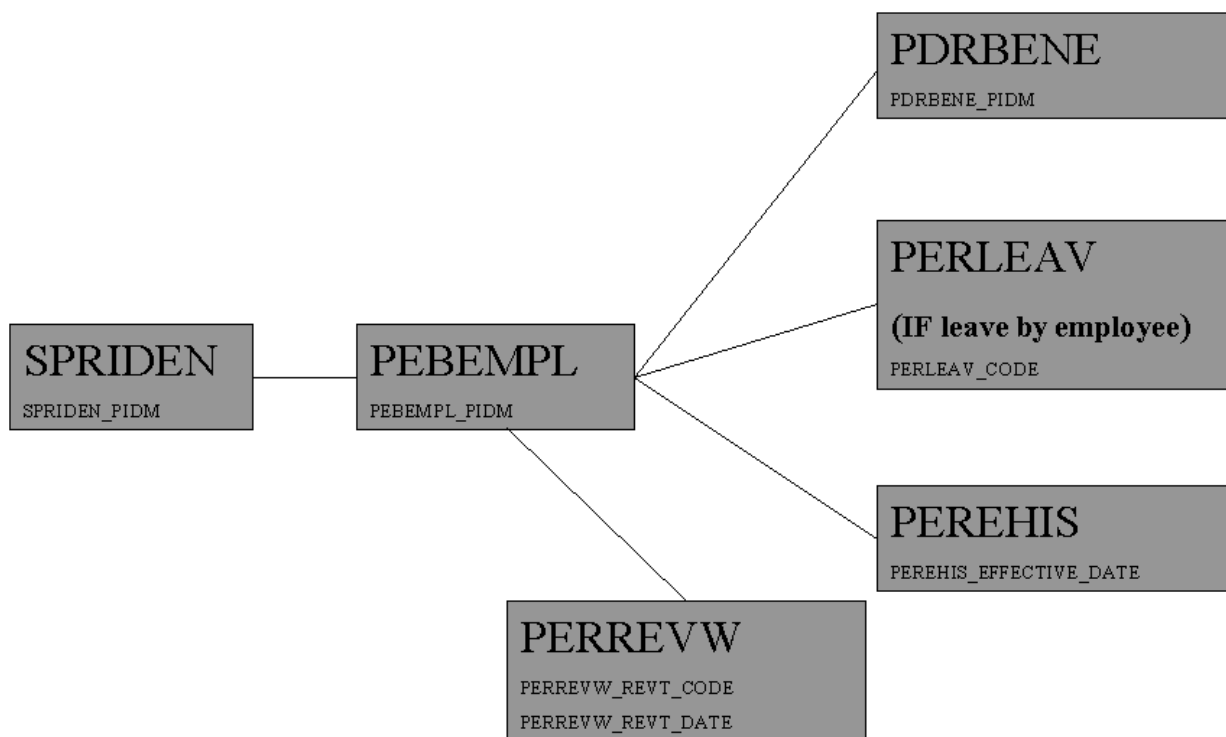
Rule Forms

- PTRLCAT
Leave Categories
- PTRECLS
Employee Class
- PTRTREA
Termination Reason

Validation Forms

- FTVCOAS
Chart of Accounts
- FTVORGN
Organization Codes

Diagram



Person records

- A row in PEBEMPL defines a person as an employee
- Each employee will have one PEBEMPL record
- A history of changes made to PEBEMPL through PEAMPLE is stored in PEREHIS
- Must have required Bio-Demo data first
- Much of the required data for the employee record defaults from the HR Hierarchy (Rules and Validation Tables)
- Other required fields are defaulted by the form PEAEMPL, but most can be overridden

HR Required Fields – PEBEMPL

- PEBEMPL_PIDM
- PEBEMPL_EMPL_STATUS
- PEBEMPL_COAS_CODE_HOME
- PEBEMPL_ORGN_CODE_HOME
- PEBEMPL_COAS_CODE_DIST
- PEBEMPL_ORGN_CODE_DIST
- PEBEMPL_ECLS_CODE
- PEBEMPL_LCAT_CODE
- PEBEMPL_BCAT_CODE
- PEBEMPL_FIRST_HIRE_DATE
- PEBEMPL_CURRENT_HIRE_DATE
- PEBEMPL_ADJ_SERVICE_DATE
- PEBEMPL_SENIORITY_DATE
- PEBEMPL_FLSA_IND
- PEBEMPL_INTERNAL_FT_PT_IND

Position Management

Purpose

Position Management is the Human Resources System module that monitors and maintains position history and incumbent history.

- Define positions
- Assign positions to budgets
- Assign positions to labor distribution
- Maintain position history

Core Application forms

- NBAPOSN
Defines all positions within a position classification and fiscal year
- NBAPBUD
Assign positions to budgets and FOAPAL
- NBAFISC
Maintain current fiscal year

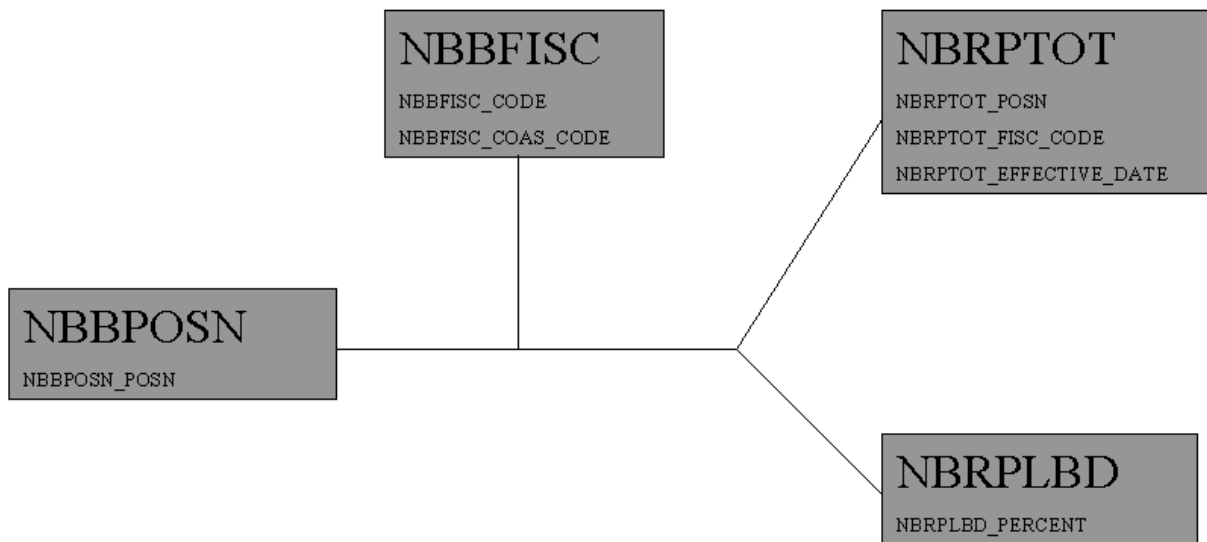
Rule forms

- NTRPCLS
Position Class
- NTRSALA
Salary rate
- NTRSGRP
Salary Group

Validation Forms

- FTVORGN
Organization
- FTVCOAS
Chart of Accounts

Diagram



Position records

- One record in NBBPOSN for each position
- At least one NBRPTOT record for each position/fiscal year combination
- At least one NBRJLBD record for each position/fiscal year combination

Compensation Administration

Purpose

Defaults come from HR hierarchy.

- Maintain:
 - Employee's Title
 - Employee's Salary
 - Compensation History

Forms

- NBAJOBS

Defines the job for a particular employee with begin and end dates, title, status, labor distribution, and salary information

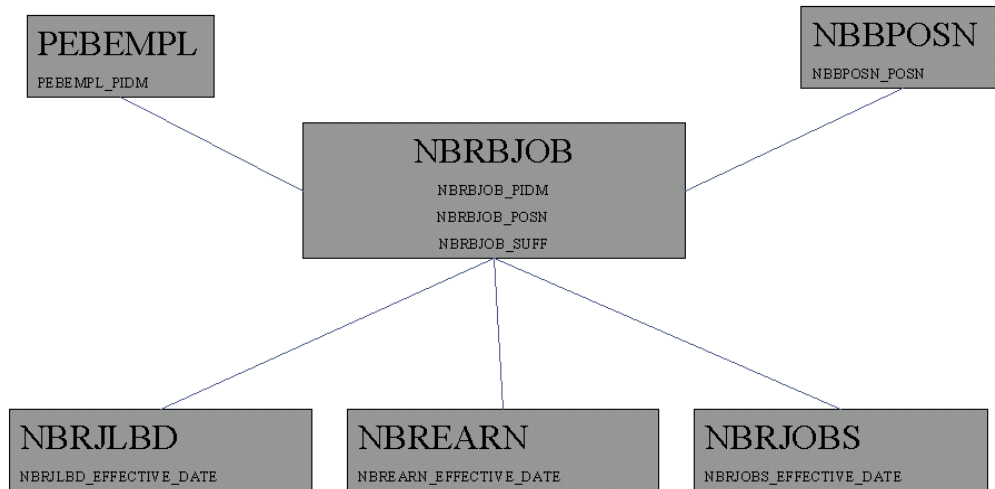
- PEIJHIS

Employee Job History Form

- PEALEAV

View and maintain leave balance records (IF leave by job method chosen on PTRINST – populated by NBAJOBS)

Diagram



Position/ employee records

- One record in NBRBJOB for each position/employee combination
- At least one record in NBRJOBS for each position/employee combination
 - NBRJOBS_EFFECTIVE_DATE = NBRBJOB_BEGIN_DATE
- Additional records for subsequent changes to job information
 - Example: Salary, Title

Position/ fiscal year / employee records

- At least one NBRJLBD record for each position/fiscal year/employee combination
 - Defaults from NBRPLBD, but can be overridden
- Records in NBREARN are defaulted depending on ECLS set up. It can be overridden

Requirements

- The person must have an active employee record to have an active job.
- Active position and budget records must exist
- Values default from Rule and Validation tables, which can be overridden

Effective Dating

Purpose

The 'interior' of Banner HR is structured based on the concept of 'Effective Dating'.

- Maintains history
- Allows for future dated personnel actions

Records are 'stacked' in the Oracle tables using the logical key, which includes the Effective Date.

Example

```
select nbrjobs_pidm, nbrjobs_posn, nbrjobs_suff,  
       nbrjobs_effective_date, nbrjobs_desc,  
       nbrjobs_status  
from   nbrjobs  
where  nbrjobs_pidm = 408  
order by nbrjobs_effective_date;
```

```
408 S00001 00 31-AUG-96 Federal Work Study (Pooled)    A  
408 S00001 00 30-SEP-96 History Dept. Office Clerk-CWS A  
408 S00001 00 11-MAY-97 History Dept. Office Clerk-CWS T  
408 S00001 00 31-AUG-97 History Dept. Office Clerk-CWS A
```

Example 1

What job record is active as of today?

```
select max(nbrjobs_effective_date)
  from nbrjobs
 where nbrjobs_effective_date <= SYSDATE
       and nbrjobs_pidm = 408;
```

31-AUG-97

Example 2

What job record is active as of Jan. 1, 2011?

```
select max(nbrjobs_effective_date)
  from nbrjobs
 where trunc(nbrjobs_effective_date) <= '01-JAN-11'
       and nbrjobs_pidm = 408;
```

31-AUG-97

A Nested Select

```
SELECT nbrjobs_pidm, nbrjobs_posn, nbrjobs_suff,  
       nbrjobs_desc, nbrjobs_status  
FROM   nbrjobs a  
WHERE  nbrjobs_pidm = 408  
       AND a.nbrjobs_status = 'A'  
       AND a.nbrjobs_effective_date =
```

```
(SELECT max(nbrjobs_effective_date)  
   FROM nbrjobs b  
  WHERE trunc(nbrjobs_effective_date) <= '01-OCT-96'  
        AND b.nbrjobs_pidm = a.nbrjobs_pidm  
        AND b.nbrjobs_posn = a.nbrjobs_posn  
        AND b.nbrjobs_suff = a.nbrjobs_suff);
```

408 S00001 00 30-SEP-96 History Dept. Office Clerk-CWS A

HR tables

Some HR tables with effective dating logic:

- NBREARN
- NBRJLBD
- NBRJOBS
- PDRDEDN

Exercise 3: Effective Dating

Directions

Use the information you have learned in this workbook to complete this self check activity.

Exercise 3

1. What is the effective date of the job record that is active as of today, for Lawrence Mitchell?
2. Is that job active as of today?
3. What was the effective date of the same job that is active as of June 1, 1994?
4. What is the annual salary of the job record for Lawrence Mitchell as of June 1, 1994?
5. List name, position number, suffix, and annual salary as of Oct. 23, 1996 for **all** employees with active jobs, in alphabetical order by last then first name.

Benefits and Deductions

Purpose

- Maintain Employee's benefits and deductions
- Eligibility administration
- Maintain Beneficiary/Dependent Information

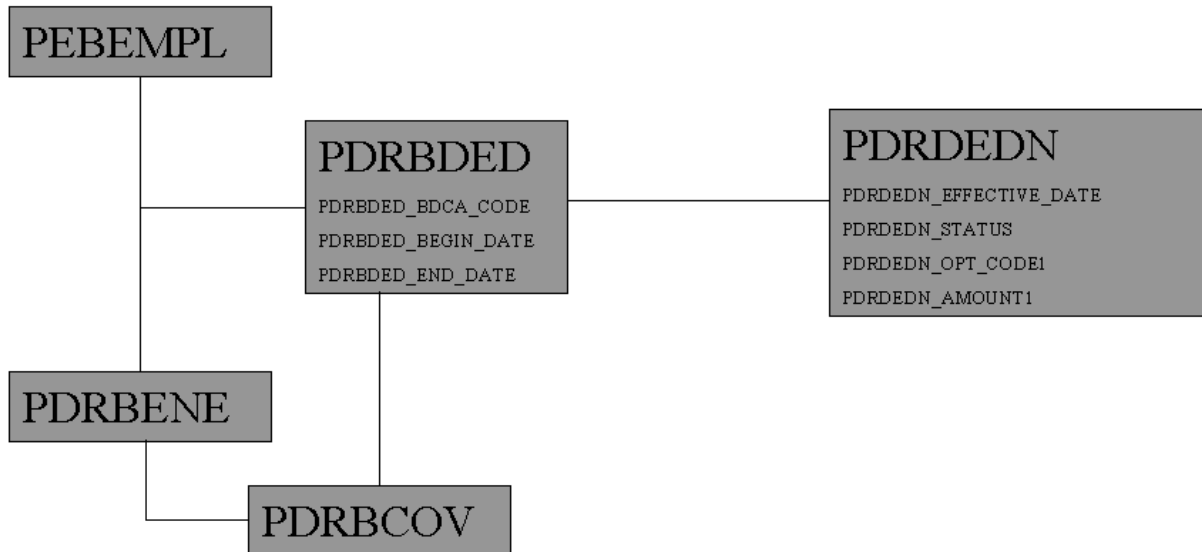
Core Application Forms

- PDAEDN
Establish/Maintain deductions for benefits, taxes and other withholdings
- PDABENE
Beneficiary Form
- PDABCOV
Beneficiary coverage Form

Rule Forms

- PTRBCAT
Benefit Category
- PTRBDCA
Benefit/Deduction Code
- PDABDSU
Benefit/Deduction Setup

Diagram



Deduction/ employee records

- One record in PDRBDED for each deduction/employee combination
- At least one record in PDRDEDN for each deduction/employee combination
 - $PDRDEDN_EFFECTIVE_DATE = PDRBDED_BEGIN_DATE$
- Additional records for subsequent changes to deduction must have a different effective date and potentially a different status, ie: future termination record
 - Example: Plans, options, amounts history tracking for mid-year go live

Requirements

- Must be an active employee to set up deductions.
- Eligibility is driven by benefit categories (BCAT) in PTRBCAT.
- 'Self' beneficiary records are created when employee record is created with PEAEMPL

Establishing Combined Deduction Limits

Banner Human Resources provided the ability to check and ensure that combinations of employee and employer contributions towards employee retirement plans were within the elective deferral maximum limit set by the IRS. This functionality has now been extended to check for the employee's

- Annual contribution limit
- Annual compensation limit
- Catch-up contributions
- 15-year rule plan.

Application forms

- PTRBDCL

Specify annual dollar limits for combinations of employee retirement plans, 50+ catch-up contributions, and the 15-year rule plan.

Altered forms

New fields related to establishing combined deduction limits have been added to:

- PTRBDCA
- PDAEDN
- PEIDHIS
- PTRBDMC

For more details on this enhancement, please refer to the *Banner 8 Human Resources Release Guide*.

Leave Administration

Purpose

- Leave can be tracked by employee or by job.
- PTRINST defines numerous Leave processing rules:
 - Leave Processing Method: By Employee vs. By Job
 - Leave Processing Order: Accrue Then Take vs. Take Then Accrue
 - Leave Balance Roll Order: Before / After Accrual
- Different set of leave tables for employee or job tracking

Leave by Employee

PEAEMPL – Leave balance records are created in PERLEAV for those leave types for which their leave category (LCAT) on PEAEMPL dictates

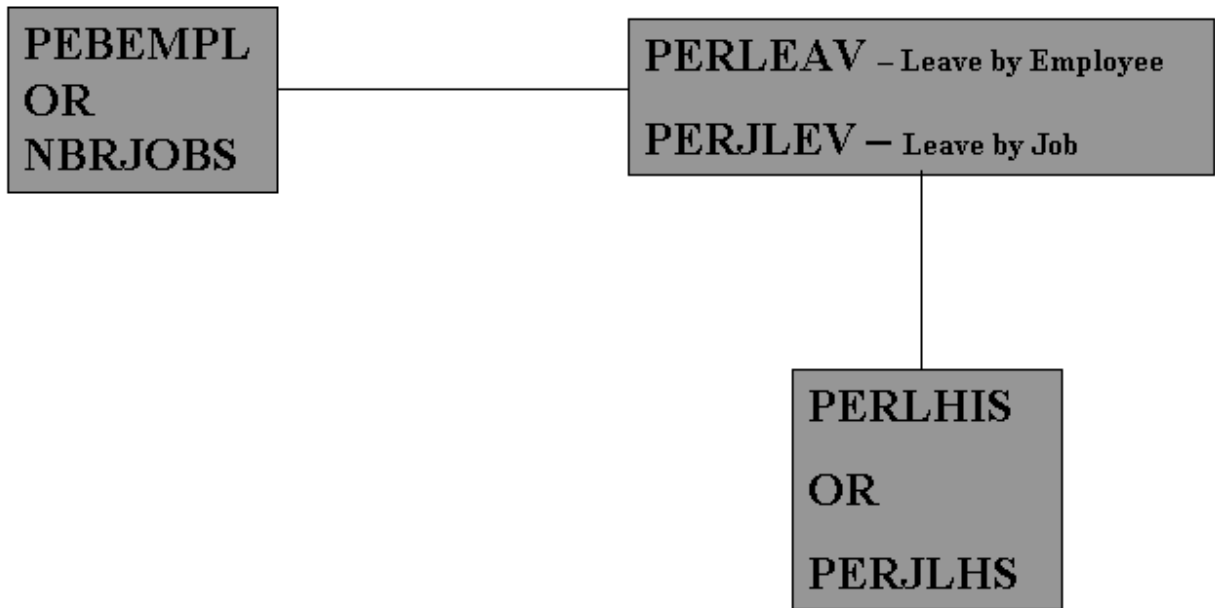
Leave by Job

NBAJOBS – Leave balance records are created in PERJLEV for those leave types for which their leave category (LCAT) on NBAJOBS dictates

Forms

- PEALEAV – View, track, and update balances for each leave code
- PEILHIS – View leave balance history
- PHIACCR – View leave accrual history

Diagram



Human Resources Administration

Administration

- Identifies Rule and Validation forms for Banner Human Resources
- 'Do Not Touch' System Required Tables
 - Tax tables, EEO reporting values, Etc
 - Usually Federally Mandated Codes
 - See: Banner Human Resources TRM Supplement
- Functional Team uses Checklists to determine which forms / tables will be populated with client-specific values

Banner New Hire Process

PPAIDEN form loads tables

- SPRIDEN PIDM, ID, Name...
- SPRADDR Address
- SPBPERS Birthdate, SSN, Ethnicity...
- SPRTELE Telephone
- SPREMRG Emergency Contact

PEAEMPL form loads tables

- PEBEMPL Employee Status, Hire Dates, Employee Classification, ...
- PERLEAV Leave Balances (if using Leave by Employee)
- PEREHIS Employee History
- PDRBENE Beneficiary / Dependent

NBAJOBS form loads tables

- NBRBJOB Position Number, Job Begin and End Date
- NBRJOBS Position Number, Effective Date, Title, Salary
- NBREARN Default Earnings Code, Default Hours
- PERJHIS Job History
- PERJLEV Leave Balances (if using Leave by Job)

PDAEDN form loads tables

- PDRBDED Deduction Code (BDCA), Deduction Begin Date, End Date
- PDRDEDN Deduction Code (BDCA), Effective Date, Deduction Plan, Deduction Amounts
- PERDHIS Deduction History

New Hire Quick Set Up

- PEAHIRE Allows you to move through a series of steps that will load tables behind PPAIDEN, PEAEMPL and NBAJOBS
Initial Set Up only
- PDABDSU Loads the tables behind PDAEDN
Initial Set Up only

Exercise 4 – Enter a New Employee into Banner

Directions

Use the information you have learned in this workbook to complete this self check activity.

Exercise

1. Log into Banner INB (TRNG), and enter PPAIDEN - the Identification Form.
2. Enter a new person into the System.
3. Identify the tables behind the form. What tables did you inset records into?
4. Make the new person an employee. What tables were records created in?
5. Give the new person a job. What tables were records created in?
6. Assign the person benefits. What tables were records created in?

Human Resources Security

Security

- In addition to BANSECR Class Security
- If ANY security types are check marked on PTRINST, HR users must be defined in PTRUSER
- Provides Row Level security for HR data
- Additionally, manages security access for SSB
 - Salary Planner
 - Faculty Load and Compensation
 - Benefits Administration
 - Time Entry Approvals
 - Leave Report Approvals
 - Effort Certification and Labor Redistributions
- If MASTER levels are not check marked on PTRUSER, values are established on subsequent forms

Four types of security limitations

- Employer
- Organization
- Employee Class
- Salary Level

The types can be used in any combination.

Application Forms

- PTRINST - Installation Rule Form
Turn on Security
- PTRUSER - User Codes Rule Form
Set up HR Users
- PSAEMPR - Banner EMPR Security Form
Set up Employer Security
- PSAORGN - Banner ORGN Security Form
Set up Organization Security
- PSAECLS - Banner ELCS Security Form
Set up Employee Class Security

Tables

- PTRUSER - User Codes Rule Form
Establishes IDs for security purposes
- PSREMPR - Banner EMPR Security Form
Defines Employer codes that users can view
- PSRORGN - Banner ORGN Security Form
Allows access to a range or ORGN codes
- PSRECLS - Banner ELCS Security Form
Allows access to individual ECLS codes

Load Order

Introduction

This section discusses the load order for Banner Human Resources components.

Load Order

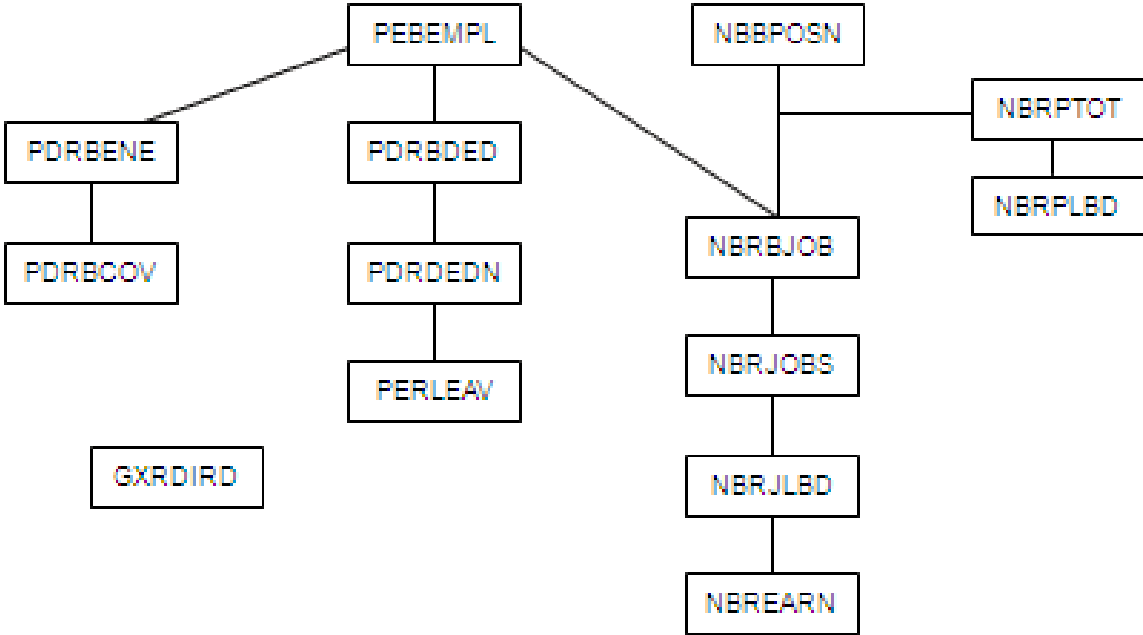
Module components

All Banner modules are comprised of two parts:

- Rule and Validation Tables
 - Must be populated first
- Module-Specific or 'PIDM related' data
 - Populated with the Converter Toolkit
 - Populated with spreadsheets
 - Manually populated

Load Order – HR Core Tables

Diagram



Human Resources / Banner System Interfaces

Introduction

This section discusses the touch points between the Banner Human Resources system and other Banner systems.

Interfaces and Integration with Other Banner Systems

Interfaces

- Advancement Interface
- Finance Interface
- Financial Aid Integration
- Student Integration

Information resources

- Technical Reference Manual Supplement (TRM)
- Banner HR User Guide: Interfaces

Advancement Interface

Advancement processing

- Advancement / Development first feeds pledges to Banner Human Resources
- Banner Human Resources feeds the deductions back in order to create the gift / installment payment records

Finance Interface

Finance processing

- Provides an automated flow of payroll expenditure, encumbrance, budget, and COBRA payment information from Human Resources to Finance
- A generic feed table, GURFEED, stores the data to be interfaced.

Financial Aid Integration

Financial Aid processing

- RJRLOAD updates Financial Aid System with actual payroll earnings
- Extracts data directly from Payroll History tables

Student Integration

Student processing

- PHPSHRS: Verification of Student Hours
 - process checks, compares, and corrects the deduction status of student-employees based on the credit hours they have enrolled during a term.
 - checks the current deduction status for each Employee Class / Deduction code combination defined on PTRSTDE
 - Corrects any discrepant records by creating a new Effective Dated deduction record in PDAEDN when appropriate
- Faculty Load and Compensation (FLAC)
 - Automated Calculation of Compensation for Faculty Assignments
 - Provide Access to information through Self Service Components
 - Provide Faculty Oriented View of Compensation Details and contract information

Payroll Process

Introduction

This section discusses the many steps involved in the Payroll Process in Banner Human Resources.

Key Concepts

Keys

- Pay Year
- Pay ID (pict code)
- Pay No

Dispositions

- Status identifiers
- Every employee is assigned a disposition at every step of the Payroll process, tracked by PHRHIST and PHRJOB
- Hard-coded into the payroll processes, so that only records ready for the next step are included

Original Payroll dispositions

- 05 Awaiting Re-Extract
- 10 Awaiting Time Entry
- 15 Awaiting Correction
- 20 Awaiting Proof
- 22 Hours Correction
- 25 Awaiting Leave Process
- 30 Awaiting Calc
- 40 Awaiting Document
- 42 Awaiting Check/Direct Deposit Run
- 43 Awaiting Direct Deposit Run
- 44 Awaiting Check Run
- 50 Awaiting Update
- 60 Finance Extract
- 70 Complete

Adjustment Pay Event Dispositions

- 27 Pre-Balance Calc
- 28 Pre-Approve Calc (preparing to calc trial register)
- 30 Awaiting Calc
- 37 Pre-Balance Check
- 38 Pre-Approve Check (preparing to cut checks)
- 40 Awaiting Document
- 42 Awaiting Check/Direct Deposit Run
- 43 Awaiting Direct Deposit Run
- 44 Awaiting Check Run
- 47 Pre-Balance Update
- 48 Pre-Approve Update (preparing to run HR system update)
- 50 Awaiting Update
- 60 Finance Extract
- 62 Finance Interface/Feed
- 70 Complete

Payroll cycle errors

- Can be fixed as they appear, or
- Print the disposition report, continue with the other employees, then return to correct the error record
 - Online Form: PHIDERR – Form to view payroll errors
 - Reports:
 - PHRDERR – Payroll Errors report
 - PHRDCON – Disposition report

Pre-Payroll Process

Optional processes

PDPLIFE – Life Insurance Calculations

- Calculates new rates based on rules in PTRLIFE
- Calculates Imputed Income

NOPEAMA – Electronic Approvals Mass Apply Process

- Applies all Electronic Approval transactions entered from the Electronic Personnel Action Form (NOAEPAF)

Banner Payroll Process

PHPTIME

PHPTIME - Time Sheet Generation

- Initializes Payroll History tables
- Initializes disposition at 05 (*Awaiting Re-extract*), 10 (*Awaiting Time Entry*), or 20 (*Awaiting Proof*)
- Open the payroll process for time entry

Tables 'initialized' by PHPTIME

- Insert into PHRHIST: Pay History Repeating Table
- Insert into PHRJOB: Pay History Jobs Repeating Table
- Insert into PHREARN: Pay History Earnings Repeating Table
- Insert into PHRHOUR: Hour Validation Table
- Insert into PHRELBD: Pay History L/D Override Repeating Table
- Insert into PHRDEDN: Pay History Deduction Repeating Table
- Insert into PHRERRL: Pay History Error Log Repeating Table

Time entry forms

- PHAHOURL – Online Time Entry
 - **Directly updates the PHR% tables**
 - Allows for data entry of Exception hours
 - Allows for adjustments to default timesheet information
- PHATIME – Time Entry with Approvals
 - Requires set up of users and routing queues
 - **Updates to PHR% tables once transaction successfully completes the routing queue**
- PHAMTIM – Mass time entry
 - Uses temporary table PHRMTIM
 - **Updates to PHR% tables after successful completion of PHPMTIM process to 'pull' hours into the payroll process**

PHPPROF

PHPPROF - Pay Period Proof Batch Process

- Validates Payroll entries
- Special rate must be entered for earn code
- PHREARN_HRS should not be 0 or NULL
- Possible insufficient leave balance
- Employee has another job at a different disposition
- Employee's total hours do not match hrs breakdown
- 'INVALID LOA END DATE'
- Manages Combined Limit Rule Codes
- Etc.

- Incoming Disposition is 20
- Outgoing Disposition is 25 (success)
 - 15 or 22 (failure)
- Tables Processed by PHPPROF
 - Update PHRHIST - Payroll History table
 - Update PHRJOB - Payroll History Jobs table
 - Update PHRERRL with errors - Pay History Error Log table

PHPLEAV

PHPLEAV - Leave Accruals and Taken Process

- Applies rules from PTRINST
- Tracks Leave Accrued / Taken in PHRACCR

- Incoming Disposition is 20
- Outgoing Disposition is 25 (success)

Tables referenced by PHPLEAV

- PERLEAV - Leave Balances
- PERJLEV - Employee Leave Balances by Jobs
- PTRLEAV - Leave Code Rule Form
- PTRLVAS - Leave Assignment Rule Form
- PTRLVAC - Leave Accrual Rule Table
- PTRLVPR - Leave Priority Code table

Tables processed by PHPLEAV

- PHRHIST – Update Disposition to 30
- PHRJOBBS – Update Disposition to 30
- PHREARN – Insert any Dock Pay Records
- PHRERRL – Insert any Errors/Warnings
- PHRACCR/PHRJACR – Insert leave Accrual Records

PHPCALC

PHPCALC - Payroll Calculation Report

- Gross to net Payroll calculation process (COBOL)
 - Calculates earnings by hours
 - Calculates Benefit / Deduction Amounts
 - Can be ran numerous times
 - Re-Calculate All Employees
- Incoming Disposition 30
- Outgoing Disposition 40 (when success)
- Updates PHRDEDN - Deduction Calculation Report
 - Calculated Deduction amounts
- Updates PHREARN - Payroll Earnings Report
 - Calculated Earnings amounts
- Updates PHRJOB - Payroll Jobs Report
 - Updates Disposition to 40
- Updates PHRHIST - Payroll History Report
 - Updates Disposition to 40 and records Gross and Net amounts
- Generates PHRHOURL - Payroll Time Entry Report
 - With calculated earnings amounts by Data Entry period
- Updates PHRACCR - Payroll History Accruals Report
 - Accrual amounts (if applicable)

PHPDOCM

PHPDOCM - Check/Direct Deposit Amount Calculation

- Creates Check and Direct Deposit Document Records
 - PHRDOCM
- Calculates Check and Direct Deposit Amounts using information from GXRDIRD : Employee Payroll Direct Deposit Information
- Incoming Disposition 40
- Outgoing Disposition 42

Considered 'the Point of No Return'

- Once PHPDOCM is ran, the event must be completely fed to Finance and adjustments made if necessary
- Updates PHRHIST- Payroll History table
- Updates PHRJOBS - Jobs History table
- Inserts Document numbers into PHRDOCM
 - starting with 1; updated by PHPCHKL
- Inserts records into temporary table PHRTDED - Temporary Payroll Deduction Record – for better performance

PHPCHKL

PHPCHKL - Check/Direct Deposit Notice Process

- PHPCHKL printing process is run twice, once for each document type
 - Once for Direct Deposits
 - Once for Checks
- Updates disposition to 43 or 44 the first run
 - Updates PHRHIST – Payroll History table
 - Updates PHRJOB – Jobs History table
 - Updates PHRDOCM – Updates document numbers
- Updates disposition to 50 after both document types have been run

PHPDIRD

PHPDIRD – Create Direct Deposit File

- Does not update disposition

PHRDIRD

- New Report in v8.3
- Displays Direct Deposit Distribution

PHPUPDT

PHPUPDT - Pay Period Update Batch Process

- Updates disposition to 60 after PHPUPDT process
- Updates Leave history and Leave balances
- Updates seniority hours
- Updates hours for adjustments
- Manages Advancement Pledge records
- Pay Advances
- Etc.

Tables updated by PHPUPDT

- Accumulators
 - PERETOT - Insert/update monthly earnings information
 - PERJTOT - Insert/update monthly earnings by position
 - PERDTOT - Insert/update monthly deduction information
- History Tables
 - PEREHIS - Insert when PEBEMPL changes
 - PERLHIS - Insert when Leave balances are updated
- Leave Tables
- Additional tables:
 - PERPADV - Insert/update with Pay Advance amounts
 - PHRHIST - Updates Disposition to 60
 - PHRJOB - Updates Disposition to 60
 - PDRBDED - Updates Arrears / delete add/replace information
 - PDRDEDN - Updates for bonds purchased
 - GXRDIRD - Updates for pre-note employees
 - NBRBJOB - Updates with deferred pay balances information
 - GURFDED /
 - GURALMP - Advancement Pledges
 - PDRFBAL - Employee Flexible Benefits Balances
 - PERPADV - Pay Advance Balances

PHPTIME - Time Processing Report

Purpose

Initializes Disposition to 05, 10, or 20

References data from the following tables to process the particular Year, Pay ID, and Pay Number combination:

- PTRCALN - Payroll Calendar Rule Table
- PTREARN - Earnings Code Rule Table
- PTRECLD - Earnings Code Labor Dist Rule Table
- PTRECLS - Employee Class Rule Table
- PTREERN - Employee Class Earn Code Rule Table
- PTREHOL - Employee Holidays Rule Table
- SPRIDEN - Identification/Name Repeating Table

Tables referenced

More tables referenced by PHPTIME:

- NBREARN - Employee Default Earnings Code Table
- NBRJLBD - Assignment Labor Dist Repeating Table
- NBRJOBS - Assignment Repeating Table
- NBRBJOB - Assignment Repeating Base Table
- PDRDEDN - Employee Deduction repeating Table
- PDRBDED - Employee Deduction Repeating Table
- NBBFISC - Fiscal Year Base Table
- PEBEMPL - Employee Base Table

Tables initialized

Tables 'initialized' by PHPTIME:

- Insert into PHRHIST: Pay History Repeating Table
- Insert into PHRJOB: Pay History Jobs Repeating Table
- Insert into PHREARN: Pay History Earnings Repeating Table
- Insert into PHRHOUR: Hour Validation Table
- Insert into PHRELBD: Pay History L/D Override Repeating Table
- Insert into PHRDEDN: Pay History Deduction Repeating Table
- Insert into PHRERRL: Pay History Error Log Repeating Table

PHPPROF - Pay Period Proof Process

Preparing for PHPPROF

- Run PHRDERR - Payroll Errors Display Report
- Run PHRDCON - Disposition Control Report
- Go to PHAHOURL - Online Time Entry Form
 - Correct any errors
 - Check default hours
 - Add exception hours
 - Change Labor Distributions
 - Re-extract, if necessary
- Run PHPPROF - Pay Period Proof Process

Purpose

- Incoming Disposition is 20
- Outgoing Disposition is 25 (success), 15 or 22 (failure)
- Tables Processed by PHPPROF
 - Update PHRHIST - Payroll History table
 - Update PHRJOB - Payroll History Jobs table
 - Update PHRERRL with errors - Pay History Error Log table

PHPLEAV - Leave Accruals/Taken Process

Preparing for PHPLEAV

- Run PHRDERR - Payroll Errors Display Report
- Run PHRDCON - Disposition Control Report
- Time Entry
 - PHAHOURL
 - PHATIME
- Web Time Entry
 - Correct any errors
 - Re-extract if necessary
- Run PHPLEAV - Leave Accruals/Taken Process

Purpose

- Incoming Disposition 25
- Outgoing Disposition 30
- Tables referenced by PHPLEAV
 - PERLEAV - Leave Balances
 - PTRLEAV - Leave Code Rule Form
 - PTRLVAS - Leave Assignment Rule Form
 - PTRLVAC - Leave Accrual Rule Table
 - PTRLVPR - Leave Priority Code table

Tables processed

- Tables processed by PHPLEAV
 - PHRHIST – Update Disposition to 30
 - PHRJOB – Update Disposition to 30
 - PHREARN – Insert any Dock Pay Records
 - PHRERRL – Insert any Errors/Warnings
 - PHRACCR – Insert leave Accrual Records

PHPCALC - Payroll Calculation Process

Preparing for PHPCALC

- Run PHRDERR - Payroll Errors Display Report
- Run PHRDCON - Disposition Control Report
- Check any Dock Pay entries
- Correct any errors
- Re-extract if necessary
- Run PHPCALC - Payroll Calculation Process

Purpose

- Calculates Gross to Net
- Calculated Benefit/Deduction Amounts
- Incoming Disposition 30
- Outgoing Disposition 40

Updated reports

- Updates PHRDEDN - Deduction Calculation Report
 - Calculated Deduction amounts
- Updates PHREARN - Payroll Earnings Report
 - Calculated Earnings amounts
- Updates PHRJOBS - Payroll Jobs Report
 - Updates Disposition to *40*
- Updates PHRHIST - Payroll History Report
 - Updates Disposition to *40* and records Gross and Net amounts
- Updates PHRHOURL - Payroll Time Entry Report
 - With calculated earnings amounts by Data Entry period
- Updates PHRACCR - Payroll History Accruals Report
 - Accrual amounts (if applicable)

PHPDOCM - Check/Direct Deposit Amounts Process

Purpose

- Incoming Disposition 40
- Outgoing Disposition 42
- Creates Check and Direct Deposit Document Records
- Checks GXRDIRD - Employee Payroll Direct Deposit Record for direct deposit information

Functions

- Updates PHRHIST- Payroll History table
- Updates PHRJOBS - Jobs History table
- Inserts Document numbers into PHRDOCM - Disposition Control Report (starting with 1)
- Inserts records into temporary table PHRTDED - Temporary Payroll Deduction Record for better performance

PHPCHKL/PHPCHEK Printing Process

PHPCHKL/ PHPCHEK

The PHPCHKL/PHPCHEK printing process is run twice, once for each document type.

- Once for Checks
- Once for Direct Deposits
- Updates disposition to 43 or 44 the first run, depending on which document type is run first
- Updates PHRHIST – Payroll History table
- Updates PHRJOB – Jobs History table
- Updates PHRDOCM – Updates document numbers
- Updates disposition to 50 after both document types have been run

Status

Now we are ready to update year to date totals and perform cleanup.

PHPUPDT – Pay Period Update Process

Tables updated by PHPUPDT

- PERETOT - Insert/update monthly earnings information
- PERJTOT - Insert/update monthly earnings by position
- PERDTOT - Insert/update monthly deduction information
- PEREHIS - Insert when PEBEMPL changes
- PERLHIS - Insert when Leave balances are updated
- PERPADV - Insert/update with Pay Advance amounts
- PHRHIST - Updates Disposition to 60
- PHRJOB - Updates Disposition to 60
- PDRBDED - Updates to delete add/replace information
- PDRDEDN - Updates for bonds purchased
- GXRDIRD - Updates for pre-note employees
- NBRBJOB - Updates with deferred pay balances information
- PERLEAV/PERJLEV - Updates Leave balance amounts

Accumulator Tables

Accumulator Tables

- Initially created to improve performance of W2s
- Payroll amounts 'accumulated' by Month
- Populated when Go Live date is NOT January 1st
 - Lump Sum Method
 - By Month Method
- Life-to-Date values for Deductions (PERDTOT)

- PERETOT - Employee History Earnings Table
- PERDTOT - Employee History Deductions Table
- PERJTOT - Employee History Earnings Table

Table comparison

Set Up Tables	History Tables	Accumulators
PEBEMPL	PHRHIST	PERETOT
NBRJOBS	PHRJOBS	PERJTOT
PDRDEDN	PHRDEDN	PERDTOT
	Etc.	

Note that the History tables are PH% tables, even those related to jobs!

Exercise 5: Query Payroll Tables

Directions

Use the information you have learned in this workbook to complete this self check activity.

Exercise

1. Retrieve the first name, last name, position number with suffix and current annual salary for all active employees (as of today). Order by last then first name and position.
2. Allowing for parameters for year, pay id and pay number, retrieve the pay period begin and end dates.
3. Retrieve the first name, last name, and position for all employees whose jobs were active during the specified payroll. Prompt for the parameters of pay year, pay id, and pay number.
4. Retrieve the payroll records which did not successfully complete the payroll process.

5. Retrieve the first and last name, gross and net pay for the original run of monthly payroll number 3 in 1995. Order by last then first name.

6. Retrieve the next available payroll year, pay id and pay number you can use as a test payroll.

Encumbrances and Budget

Introduction

This section discusses encumbrance maintenance in the Banner Payroll Process.

Banner Payroll Process – Budget Maintenance

NBPBUDM – Budget Maintenance Process

- Does not update disposition
- **Supports the Fringe Chargeback Method**
- Used during the budget development cycle
- Used throughout the year for new **or changed positions**
- **Removed from the Payroll Process in v8 with the introduction of NBPENCM**

NBPENCM – Encumbrance Maintenance Process

- Does not update disposition
- Computes Salary encumbrances amounts
- Assigns encumbrance numbers
- Ran with fiscal year end processes to release/create encumbrances
- **Ran throughout the year in sync with payroll** to pick up encumbrance changes and to liquidate encumbrances if using the recasting method
- Manages Multi-Year encumbering

Feed to Finance

Introduction

This section discusses the Feed to Finance from Banner Human Resources.

Feed to Finance: Payroll Expenses

PHPFEXP – Expenditures Finance Extract **(COBOL)**

- Prepares Payroll Expense data for the Feed to Finance
 - Updates Disposition to 62
 - Updates Expended Amounts on NBAPBUD
 - Inserts into NHRFINC (temporary)
 - Inserts into NHRDIST if option checked on NTRINST

Feed to Finance - Budgets and Encumbrances

NHPFIN1

- Prepares Encumbrance, Budget and COBRA data for the Feed to Finance process
 - Does NOT update the disposition
 - Inserts into NHRFINC (temporary)
 - Inserts into NHRDIST if option checked on NTRINST

NHPFIN2

- Updates disposition to 70
- Loads PHPFEXP and NHPFIN1 interface data into GURFEED - the Finance Interface table
- Looks to FOBSEQN table to get the next Document number, and it will start with *F*
- Prints Finance report

Feed to Finance - Finance Processes Take Over

Processes

- Refer to the Finance User Guide for instruction
 - Interfaces Chapter
 - Feeding Payroll Transactions
- Refer to the Human Resources User Guide
 - Interfaces Chapter
 - Interface to the Banner Finance System
 - Feed Payroll Transactions

Feed to Finance – Recommendations

Multiple Feeds

Run separate Feeds to Finance to provide separate reports:

- Payroll Expenditures (Finance and Payroll Office)
 - PHPUPDT
 - PHPFEXP
 - NHPFIN2
 - FURFEED.... Initiated by Finance

- Encumbrance and Budget (Finance and Budget Office)
 - NBPBUDM (if applicable)
 - NBPENCM
 - NHPFIN1
 - NHPFIN2
 - FURFEED....Initiated by Finance

Ellucian Help Resources

Introduction

This section discusses the various help resources provided by Ellucian.

Sources of Help

Banner Bookshelf

- Banner General Technical Reference Manual (TRM)
- Module-Specific
 - Technical Reference Manual (TRM) Supplement(s)
- Banner Human Resources
 - User Guide
 - Release Guides
 - Handbooks
 - Training Workbooks

Ellucian Support

Customer Support Center: www.ellucian.com

- Browse open and resolved contacts of your organization in detail
- Browse Known Issues, defect descriptions, corrections, workarounds and FAQs
- Download Banner documentation and Upgrades
- Review the Product Calendar for future product enhancements and defect resolutions
- Browse known issues

Ellucian Action Line

Phone/Fax help for time-critical issues.

To contact the Action Line:

Phone: 1-800-522-4TCP

FAX: 1-610-725-7430

The Commons

- Online community where Ellucian customers, partners and employees share, learn, explore and create.
 - share experiences
 - ask questions and find answers
 - connect to a collaborative and informed network
 - Articles, news, blogs, best practices
 - reference materials, document libraries, discussion threads
- Take the 'Getting Started in The Commons' video tour

Sign Up @ <http://commons.ellucian.com>